AIX 5L Version 5.2

Installation Guide and Reference

SC23-4389-04

AIX 5L Version 5.2

Installation Guide and Reference

SC23-4389-04

Note

Before using this information and the product it supports, read the information in Appendix B, “Notices,” on page 347.

Fifth Edition (August 2004)

This edition applies to AIX 5L Version 5.2 and to all subsequent releases of this product until otherwise indicated in new editions.

A reader’s comment form is provided at the back of this publication. If the form has been removed, address comments to Information Development, Department H6DS-905-6C006, 11501 Burnet Road, Austin, Texas 78758-3493. To send comments electronically, use this commercial Internet address: aix6kpub@austin.ibm.com. Any information that you supply may be used without incurring any obligation to you.

© Copyright International Business Machines Corporation 2002, 2004. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About This Book ix
Who Should Use This Book ix
Highlighting ix
Viewing Readme Files ix
Case-Sensitivity in AIX x
ISO 9000 x
Related Publications x

Part 1. How-To's for AIX Installation Tasks 1

Chapter 1. How-To's for Base Operating System Installation Tasks 3
Perform a New and Complete Overwrite Base Operating System Installation from CD 3
Perform a Migration Base Operating System Installation from CD 7
Perform a Nonprompted New and Complete Overwrite Base Operating System Installation from CD 10
Create and Install a Software Bundle 12
Add Open Source Applications to Your AIX System 15
Clone a rootvg Using Alternate Disk Installation 17
Configure NIM Using EZNIM 20
Install a Client Using NIM 21
Install AIX from a Linux Server Using NIM on Linux 23
Performing a Network Installation of a JS20 Blade 27
Create a System Backup to Tape 32
Clone a System Using a System Backup Tape 34
Clean Up a Failed Software Installation 35

Part 2. Operating System and Optional Software Installation 37

Chapter 2. What's New in Base Operating System Installation and Software Installation? 39
BOS Menus 39
Software Maintenance and Utilities 39
Software Service Management 39
Emergency Fix Management 40
Alternate Disk Migration Installation 40
Universal Disk Format Support for DVD-RAM 40
Installation Commands 40

Chapter 3. Introduction to Base Operating System Installation 43
BOS Menus 44
Electronic License Agreements 45

Chapter 4. Installation Options 47

Chapter 5. New and Complete Overwrite Installation / Preservation Installation 51
Step 1. Complete the Prerequisites 51
Step 2. Prepare Your System for Installation 51
Step 3. Set Up an ASCII Terminal 52
Step 4. Boot from Your Installation Media. 53
Step 5. Verify or Change the Installation Settings 53
Step 6. Finish the BOS Installation 54
Related Information. 55

Chapter 6. Migration Installation 57

© Copyright IBM Corp. 2002, 2004

Binary Compatibility Between Earlier Versions and AIX 5.2 . . . 57
Pre-Migration and Post-Migration Checking . . . 58
Migrating to AIX 5.2 . . . 59
Related Information . . . 62

Chapter 7. Configuring the Operating System . . . 63
Configuration Assistant . . . 63
Installation Assistant . . . 64
Related Information . . . 65

Chapter 8. Customized BOS Installations . . . 67
Introduction to Customized BOS Installations . . . 67
Customizing and Using a bosinst.data File . . . 68
Related Information . . . 70

Chapter 9. The bosinst.data File . . . 71
bosinst.data File Stanza Descriptions . . . 71
bosinst.data File Example . . . 78

Chapter 10. Optional Software Products and Service Updates . . . 81
Optionally Installed Software . . . 81
Software Product Identification . . . 82
Software Licensing . . . 82
Installing Optional Software Products and Service Updates . . . 83
Maintaining Optional Software Products and Service Updates . . . 87
Cleaning Up Optional Software Products and Service Updates . . . 89
Software Service Management . . . 91
InstallShield MultiPlatform Packaged Installations . . . 93
Emergency Fix Management . . . 98
Related Information . . . 116

Chapter 11. AIX Documentation and the Documentation Library Service . . . 117
Information Center . . . 117
Using the Documentation Library Service . . . 118
Installing the Online Documentation . . . 125
Related Information . . . 126

Chapter 12. Creating and Installing System Backups . . . 127
Creating System Backups . . . 127
Installing System Backups . . . 139

Chapter 13. Alternate Disk Installation . . . 147
Filesets to Install . . . 147
Alternate mksysb Disk Installation . . . 147
Alternate Disk rootvg Cloning . . . 148
Alternate Disk Migration Installation . . . 149
Data Access Between the Original rootvg and the New Alternate Disk . . . 151
Installing to an Alternate Disk using Web-based System Manager . . . 152
Running Alternate Disk Installation Using SMIT . . . 152
Alternate Disk Installation and Dynamic Logical Partitioning . . . 152
Examples . . . 152

Chapter 14. Software Product Packaging Concepts . . . 155
Software Package Formats . . . 155
Fileset Installation Packaging . . . 156
Creating Software Packages . . . 156

Bundle Packaging 157

Chapter 15. Troubleshooting Operating System and Optional Software Installation 159
Troubleshooting an Installation from a System Backup 159
Troubleshooting a Migration Installation 161
Troubleshooting an Alternate Disk Installation. 162
Troubleshooting After an Installation 162
Accessing a System That Does Not Boot 163
Troubleshooting a Full /usr File System 165
Viewing BOS Installation Logs 166
Handling System and Error Messages 166

Part 3. Network Installation 179

Chapter 16. What's New in Network Installation Management (NIM)? 181
Working with EZNIM 181
Creating Resources Simultaneously 181
LPP Source Enhancements 181
Resource Groups 182
NIM Commands 182

Chapter 17. Network Installation Management Introduction 183

Chapter 18. NIM Task Roadmap 185

Chapter 19. EZNIM 187
EZNIM Master Operations 187
EZNIM Client Operations 188

Chapter 20. Basic NIM Operations and Configuration 189
Configuring the NIM Master and Creating Basic Installation Resources 190
Adding a Standalone NIM Client to the NIM Environment 192
Using Installation Images to Install the Base Operating System (BOS) on a NIM Client 196
Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation) 197
Performing a Nonprompted BOS Installation 200
Installing to Clients on ATM Networks 201
Customizing NIM Clients and SPOT Resources 203
Configuring the NIM Master and Creating Resources to Support Diskless and Dataless Clients Only 204
Adding a Diskless or Dataless Client to the NIM Environment 207
Initializing and Booting a Diskless or Dataless Machine 209
Uninitializing Diskless and Dataless Machines 210
Installing to an Alternate Disk on a NIM Client (cloning or mksysb) 210
Network Installation Management Commands Reference 212

Chapter 21. Advanced NIM Installation Tasks 215
Controlling the Master or Client 215
Resetting Machines 216
Using Client Machines as Resource Servers 217
Defining a Machine Group 218
Adding New Members to Machine Groups 219
Removing Members from Machine Groups. 219
Including and Excluding a Group Member from Operations on the Group 220
Using Resource Groups to Allocate Related Resources to NIM Machines 221
Managing Software on Standalone Clients and SPOT Resources 222
Rebuilding Network Boot Images for a SPOT 225

Maintaining Software in an lpp_source . . . 225
Viewing Installation, Configuration, and Boot Logs . . . 227
Verifying Installation with the lppchk Operation . . . 227
Using NIM to Install Clients Configured with Kerberos Authentication . . . 228
Concurrency Control . . . 229

Chapter 22. Advanced NIM Configuration Tasks . . . 231
Removing Machines from the NIM Environment . . . 231
Creating Additional Interface Attributes . . . 232
Defining /usr versus non-/usr SPOTs . . . 233
Re-Creating SPOT Resources from Existing Directories . . . 234
Defining an lpp_source on CD-ROM versus Disk . . . 235
Establishing a Default NIM Route Between Networks . . . 235
Establishing a Static NIM Route Between Networks . . . 236
Recovering the /etc/niminfo File . . . 237
Backing Up the NIM Database . . . 238
Restoring the NIM Database and Activating the NIM Master . . . 238
Unconfiguring the NIM Master . . . 239
Booting Diagnostics . . . 239
Booting in Maintenance Mode . . . 240
Secondary Adapter Support . . . 242

Chapter 23. Additional NIM Topics . . . 247
NIM Master Management Tasks . . . 247
NIM Name Resolution . . . 249
Booting a FDDI Interface Over a Router . . . 249
Default Paging Space During BOS Installation Through NIM . . . 250
Migrating Diskless and Dataless Clients and NIM SPOTS . . . 250
Defining the NIM Environment Using the nimdef Command . . . 251
Name Requirements for NIM Object Definitions . . . 251
Interacting with the Dynamic Host Configuration Protocol . . . 251
Creating File Resources in the root Directory . . . 251
Restricting NIM Client Resource Allocation . . . 252
Preventing Machines from Adding Themselves as Clients . . . 252
Disabling Client CPU ID Validation . . . 253
Exporting NIM Resources Globally . . . 253
Creating Network Boot Images to Support Only the Defined Clients and Networks . . . 254
Updating a Spot with New Device Support for a New Level of AIX . . . 255
Tuning Client-Request Processing . . . 255

Chapter 24. Network Installation Management Concepts . . . 257
NIM Machines . . . 257
NIM Networks . . . 263
NIM Resources . . . 267
NIM Operations . . . 282
NIM Groups . . . 301

Chapter 25. Sample Files . . . 305
script Resource File . . . 305
Definition File for the nimdef Command . . . 305

Chapter 26. NIM Error and Warning Messages . . . 307

Chapter 27. NIM Troubleshooting . . . 335
Debugging a Network Boot Problem . . . 335
Producing Debug Output for NIM BOS Installations . . . 337

Port Number Conflicts with NIM and Other Applications 339

Part 4. Appendixes 341

Appendix A. Network Boot 343
Booting a Machine Over the Network 343
Creating IPL ROM Emulation Media 346

Appendix B. Notices 347
Trademarks 348

Glossary 349

Index 355

About This Book

This book provides system administrators with complete information about how to perform such tasks as installing and maintaining the AIX operating system and optional software on standalone systems and on client systems from a resource server using the Network Install Management (NIM) interface. It also includes information on how to migrate a system, manage system backups, install AIX updates, use alternate disk installation, and troubleshoot problems with installation. This publication is available on the documentation CD that is shipped with the operating system.

This edition supports the release of AIX 5L Version 5.2 with the 5200-04 Recommended Maintenance package. Any specific references to this maintenance package are indicated as *AIX 5.2 with 5200-04*.

Who Should Use This Book

This book is intended for system administrators who are installing AIX on standalone systems or networked systems.

Highlighting

The following highlighting conventions are used in this book:

Bold	Identifies commands, subroutines, keywords, files, structures, directories, and other items whose names are predefined by the system. Also identifies graphical objects such as buttons, labels, and icons that the user selects.
Italics	Identifies parameters whose actual names or values are to be supplied by the user.
`Monospace`	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or information you should actually type.

Viewing Readme Files

The Base Operating System (BOS) includes a readme file that contains information not included in other documentation. Each software product may also have its own readme file with new information specific to that product. After you install BOS, view these files to learn important changes before using your system.

Use the following procedure to view the readme files for Base Operating System (BOS) software and optional software products:

1. Log in as root user if you have not already done so.
2. Enter the following command at the system prompt:

   ```
   cd /usr/lpp
   ```
3. Type:

   ```
   ls */*README*
   ```

   The system lists readme files for each software product installed on your system.
4. To view a readme file for a specific software product, use the following command:

   ```
   pg xxx/README
   ```

   In this example, *xxx* is the directory name associated with a particular software product.

   Press Enter when the copyright screen appears. Press the following keys or key combinations to scroll through the readme file:

To page down	Press Enter key.
To page up	Type the minus (-) key, then press the Enter key.

© Copyright IBM Corp. 2002, 2004

To move forward x **pages**	Type the plus (+) key and number of pages, then press the Enter key. For example, to move forward five pages, type +5 and press the Enter key.
To move backward x **pages**	Type the minus (-) key and number of pages, then press the Enter key. For example, to move backward five pages, type -5 and press Enter.

Type q at the : (colon) prompt to exit the readme file.

Case-Sensitivity in AIX

Everything in the AIX operating system is case-sensitive, which means that it distinguishes between uppercase and lowercase letters. For example, you can use the **ls** command to list files. If you type LS, the system responds that the command is ″not found.″ Likewise, **FILEA**, **FiLea**, and **filea** are three distinct file names, even if they reside in the same directory. To avoid causing undesirable actions to be performed, always ensure that you use the correct case.

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

Related Publications

The following publications contain additional information related to the installation and management of AIX 5.2:

- *AIX 5L Version 5.2 Operating System Installation: Getting Started*
- *AIX 5L Version 5.2 System Management Guide: Operating System and Devices*
- *AIX 5L Version 5.2 System Management Guide: Communications and Networks*
- *AIX 5L Version 5.2 Commands Reference*
- *AIX 5L Version 5.2 Files Reference*
- *AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs*
- *AIX 5L Version 5.2 System User's Guide: Operating System and Devices*
- *AIX 5L Version 5.2 System User's Guide: Communications and Networks*
- *AIX 5L Version 5.2 Security Guide*

Part 1. How-To's for AIX Installation Tasks

Part 1 contains how-to instructions for performing installation tasks.

© Copyright IBM Corp. 2002, 2004

Chapter 1. How-To's for Base Operating System Installation Tasks

This chapter provides the following how-to instructions for performing common installation tasks:

- "Perform a New and Complete Overwrite Base Operating System Installation from CD"
- "Perform a Migration Base Operating System Installation from CD" on page 7
- "Perform a Nonprompted New and Complete Overwrite Base Operating System Installation from CD" on page 10
- "Create and Install a Software Bundle" on page 12
- "Add Open Source Applications to Your AIX System" on page 15
- "Clone a rootvg Using Alternate Disk Installation" on page 17
- "Configure NIM Using EZNIM" on page 20
- "Install a Client Using NIM" on page 21
- "Install AIX from a Linux Server Using NIM on Linux" on page 23
- "Performing a Network Installation of a JS20 Blade" on page 27
- "Create a System Backup to Tape" on page 32
- "Clone a System Using a System Backup Tape" on page 34
- "Clean Up a Failed Software Installation" on page 35

Perform a New and Complete Overwrite Base Operating System Installation from CD

Using this scenario, you can install the AIX operating system for the first time or overwrite an existing version of the operating system. In this scenario, you will do the following:

- Boot from the AIX product CD
- Set BOS Installation Settings
  - Perform a new and complete overwrite installation of AIX onto `hdisk0`
  - Use English as the primary language
  - Use the default options in the More Options menu
- Start the BOS Installation and Configure the System

If you are overwriting an existing system, gather the TCP/IP information from the system before you begin this scenario.

Attention: This procedure requires shutting down and reinstalling the base operating system. Whenever you reinstall any operating system, schedule your downtime when it least impacts your workload to protect yourself from a possible loss of data or functionality. Before you perform a new and complete overwrite installation, ensure you have reliable backups of your data and any customized applications or volume groups. For instructions on how to create a system backup, refer to "Creating System Backups" on page 127.

The following steps show you how to use the system's built-in CD-ROM device to perform a new and complete overwrite base operating system installation.

Step 1. Prepare Your System

- There must be adequate disk space and memory available. AIX 5.2 requires 128 MB of memory and 2.2 GB of physical disk space. For additional release information, see the *AIX 5.2 Release Notes*.

© Copyright IBM Corp. 2002, 2004

- Make sure your hardware installation is complete, including all external devices. See the documentation provided with your system unit for installation instructions.
- If your system needs to communicate with other systems and access their resources, make sure you have the information in the following worksheet before proceeding with installation:

Table 1. Network Configuration Information Worksheet

Network Attribute	**Value**
Network Interface	(For example: en0, et0)
Host Name	
IP Address	_____._____._____._____
Network Mask	_____._____._____._____
Nameserver	_____._____._____._____
Domain Name	
Gateway	_____._____._____._____

Step 2. Boot from the AIX Product CD

1. Insert the *AIX Volume 1* CD into the CD-ROM device.
2. Make sure all external devices attached to the system (such as CD-ROM drives, tape drives, DVD drives, and terminals) are turned on. Only the CD-ROM drive from which you will install AIX should contain the installation media.
3. Power on the system.
4. When the system beeps twice, press F5 on the keyboard (or `5` on an ASCII terminal). If you have a graphics display, you will see the keyboard icon on the screen when the beeps occur. If you have an ASCII terminal (also called a tty terminal), you will see the word `keyboard` when the beeps occur.

   Note: If your system does not boot using the F5 key (or the `5` key on an ASCII terminal), refer to your hardware documentation for information about how to boot your system from an AIX product CD.
5. Select the system console by pressing F1 (or `1` on an ASCII terminal) and press Enter.
6. Select the English language for the base operating system (BOS) Installation menus by typing a `1` in the **Choice** field. Press Enter to open the Welcome to Base Operating System Installation and Maintenance screen.
7. Type `2` to select **2 Change/Show Installation Settings and Install** in the **Choice** field and press Enter.

```
                    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



    88  Help ?
    99  Previous Menu
>>> Choice [1]: 2
```

Step 3. Set and Verify BOS Installation Settings

1. In the Installation and Settings screen, verify the installation settings are correct by checking the method of installation (new and complete overwrite), the disk or disks you want to install, the primary language environment settings, and the advanced options.

   If the default choices are correct, type `0` and press Enter to begin the BOS installation. The system automatically reboots after installation is complete. Go to “Step 4. Configure the System after Installation” on page 6.

   Otherwise, go to sub-step 2.

2. To change the System Settings, which includes the method of installation and disk where you want to install, type `1` in the **Choice** field and press Enter.

```
                        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.............New and Complete Overwrite
         Disk Where You Want to Install.....hdisk0

>>> Choice [0]: 1
```

3. Type `1` for New and Complete Overwrite in the **Choice** field and press Enter. The Change Disk(s) Where You Want to Install screen now displays.

```
                Change Disk(s) Where You Want to Install

Type one or more numbers for the disk(s) to be used for installation and press
Enter. To cancel a choice, type the corresponding number and Press Enter.
At least one bootable disk must be selected. The current choice is indicated
by >>>.

        Name      Location Code   Size(MB)  VG Status   Bootable

     1  hdisk0   04-B0-00-2,0      4296   none            Yes
     2  hdisk1   04-B0-00-5,0      4296   none            Yes
     3  hdisk2   04-B0-00-6,0     12288   none            Yes

>>>  0   Continue with choices indicated above

    66  Disks not known to Base Operating System Installation
    77  Display More Disk Information
    88  Help ?
    99  Previous Menu

>>> Choice [0]:
```

4. In the Change Disk(s) Where You Want to Install screen:
   a. Select `hdisk0` by typing a `1` in the **Choice** field and press Enter. The disk will now be selected as indicated by **>>>**. To unselect the destination disk, type the number again and press Enter.
   b. To finish selecting disks, type a `0` in the **Choice** field and press Enter. The Installation and Settings screen displays with the selected disks listed under `System Settings`.
5. Change the Primary Language Environment Settings to English (United States). Use the following steps to change the Cultural Convention, Language, and Keyboard to English.
   a. Type `2` in the **Choice** field on the Installation and Settings screen to select the **Primary Language Environment Settings** option.
   b. Type the number corresponding to English (United States) as the Cultural Convention in the **Choice** field and press Enter.
   c. Select the appropriate keyboard and language options.

You do not need to select the **More Options** selection, because you are using the default options in this scenario. For more information about the installation options available in AIX 5.2, see Chapter 4, “Installation Options,” on page 47.

6. Verify that the selections are correct in the Overwrite Installation Summary screen, as follows:

```
     Overwrite Installation Summary

Disks:  hdisk0
Cultural Convention:  en_US
Language:  en_US
Keyboard:  en_US
64 Bit Kernel Enabled:  No
JFS2 File Systems Created:  No
Desktop:  CDE
Enable System Backups to install any system:  Yes

Optional Software being installed:

>>> 1   Continue with Install
    88  Help ?
    99  Previous Menu

>>> Choice [1]:
```

7. Press Enter to begin the BOS installation. The system automatically reboots after installation is complete.

Step 4. Configure the System after Installation

1. On systems with a graphics display, after a new and complete overwrite installation, the Configuration Assistant opens. On systems with an ASCII display, after a new and complete overwrite installation, the Installation Assistant opens.
2. Select the **Accept Licenses** option to accept the electronic licenses for the operating system.
3. Set the date and time, set the password for the administrator (root user), and configure network communications (TCP/IP).

   Use any other options at this time. You can return to the Configuration Assistant or the Installation Assistant by typing **configassist** or **smitty assist** at the command line.
4. Select **Exit the Configuration Assistant** and select **Next**. Or, press F10 (or ESC+0) to exit the Installation Assistant.
5. If you are in the Configuration Assistant, select **Finish now, and do not start Configuration Assistant when restarting AIX** and select **Finish**.

At this point, the BOS Installation is complete, and the initial configuration of the system is complete.

Perform a Migration Base Operating System Installation from CD

Using this scenario, you can migrate a system from AIX 4.3.3 (or earlier) to AIX 5.2. In this scenario, you will do the following:

- Perform a migration installation of AIX 4.3.3 to AIX 5.2
- Use English as the primary language
- Use the default options in the Advanced Options menu

If you are overwriting an existing system, gather the TCP/IP information before you begin this scenario.

Attention: This procedure requires shutting down and reinstalling the base operating system. Whenever you reinstall any operating system, schedule your downtime when it least impacts your workload to protect yourself from a possible loss of data or functionality. Before you perform a migration installation, ensure you have reliable backups of your data and any customized applications or volume groups. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

The following steps show you how to use the system’s built-in CD-ROM device to perform a migration base operating system installation.

Step 1. Prepare for the Migration

Before starting the migration, complete the following prerequisites:

- Ensure that the root user has a primary authentication method of SYSTEM. You can check this condition by typing the following command:

  ```
  # lsuser -a auth1 root
  ```

  If needed, change the value by typing the following command:

  ```
  # chuser auth1=SYSTEM root
  ```
- Before you begin the installation, other users who have access to your system must be logged off.
- Verify that your applications will run on AIX 5.2. Also, check if your applications are binary-compatible with AIX 5.2. If your system is an application server, verify that there are no licensing issues. Refer to your application documentation or provider to verify on which levels of AIX your applications are supported and licensed. You can also check the *AIX application availability guide* at the following Web address: http://www-1.ibm.com/servers/aix/products/ibmsw/list/
- Verify that all currently installed software is correctly entered in the Software Vital Product Database (SWVPD), by using the **lppchk** command. To verify that all filesets have all required requisites and are completely installed, type the following:

  ```
  # lppchk -v
  ```
- Check that your hardware microcode is up-to-date.
- All requisite hardware, including any external devices (such as tape drives or CD/DVD-ROM drives), must be physically connected and powered on. If you need further information, refer to the hardware documentation that accompanied your system.
- Use the **errpt** command to generate an error report from entries in the system error log. To display a complete detailed report, type the following:

  ```
  # errpt -a
  ```
- There must be adequate disk space and memory available. AIX 5.2 requires 128 MB of memory and 2.2 GB of physical disk space. For additional release information, see the *AIX 5.2 Release Notes*.
- Make a backup copy of your system software and data. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

Step 2. Boot from the AIX Product CD

1. If they are not already on, turn on your attached devices.
2. Insert the *AIX Volume 1* CD into the CD-ROM device.
3. Reboot the system by typing the following command:

   ```
   # shutdown -r
   ```

4. When the system beeps twice, press F5 on the keyboard (or 5 on an ASCII terminal). If you have a graphics display, you will see the keyboard icon on the screen when the beeps occur. If you have an ASCII terminal (also called a tty terminal), you will see the word keyboard when the beeps occur.

   Note: If your system does not boot using the F5 key (or the 5 key on an ASCII terminal), refer to your hardware documentation for information about how to boot your system from an AIX product CD.

5. Select the system console by pressing F1 (or 1 on an ASCII terminal) and press Enter.
6. Select the English language for the BOS Installation menus by typing a 1 at the **Choice** field and press Enter. The Welcome to Base Operating System Installation and Maintenance menu opens.
7. Type 2 to select **2 Change/Show Installation Settings and Install** in the **Choice** field and press Enter.

```
                    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


    88  Help ?
    99  Previous Menu
>>> Choice [1]: 2
```

Step 3. Verify Migration Installation Settings and Begin Installation

1. Verify that migration is the method of installation. If migration is not the method of installation, select it now. Select the disk or disks you want to install.

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

2. Select **Primary Language Environment Settings (AFTER Install)**.
3. Type 3 and press Enter to select **More Options**. To use the Help menu to learn more about the options available during a migration installation, type 88 and press Enter in the Installation Options menu. For more information about the installation options available in AIX 5.2, see Chapter 4, “Installation Options,” on page 47.
4. Verify the selections in the Migration Installation Summary screen and press Enter.
5. When the Migration Confirmation menu displays, follow the menu instructions to list system information or continue with the migration by typing 0 and pressing Enter.

```
                                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.

    Acceptance of license agreements is required before using system.
    You will be prompted to accept after the system reboots.

>>> 0  Continue with the migration.
   88  Help ?

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


>>> Choice[0]:
```

Step 4. Verify System Configuration after Installation

After the migration is complete, the system will reboot. Verify the system configuration, as follows:

1. On systems with a graphics display, after a migration installation, the Configuration Assistant opens. On systems with an ASCII display, after a migration installation, the Installation Assistant opens.

   For more information on the Configuration Assistant or the Installation Assistant, see Chapter 7, “Configuring the Operating System,” on page 63.
2. Select the **Accept Licenses** option to accept the electronic licenses for the operating system.
3. Verify the administrator (root user) password and network communications (TCP/IP) information.

   Use any other options at this time. You can return to the Configuration Assistant or the Installation Assistant by typing **configassist** or **smitty assist** at the command line.
4. Select **Exit the Configuration Assistant** and select **Next**. Or, press F10 (or ESC+0) to exit the Installation Assistant.
5. If you are in the Configuration Assistant, select **Finish now, and do not start Configuration Assistant when restarting AIX** and select **Finish**.

When the login prompt displays, log in as the root user to perform system administration tasks.

Perform a Nonprompted New and Complete Overwrite Base Operating System Installation from CD

Using this scenario, you can perform a nonprompted new and complete overwrite base operating system installation from CD. The first time you install, the base operating system (BOS) installation program presents menus from which you must choose setup options.

For subsequent installations, you can change many aspects of the default BOS installation program by editing the **bosinst.data** file. The **bosinst.data** file directs the actions of the BOS installation program. The file resides in the **/var/adm/ras** directory on the installed machine only, and it is not accessible on the commercial tape or the CD on which you received AIX 5.2.

Also, the **bosinst.data** file can be used to replicate one set of installation settings on other machines. For example, system administrators can create a **bosinst.data** file with settings that can be used to install all the machines they support that have the same configuration.

In this scenario, you will create a **bosinst.data** file that will not prompt the user during the BOS installation menus.

In this scenario, you will do the following:

- Create a customized **bosinst.data** file
- Create a supplementary diskette
- Boot from the AIX product CD

If you are overwriting an existing system, gather the TCP/IP information from the system before you begin this scenario.

Attention: This procedure requires shutting down and reinstalling the base operating system. Whenever you reinstall any operating system, schedule your downtime when it least impacts your workload to protect yourself from a possible loss of data or functionality. Before you perform a new and complete overwrite installation, ensure you have reliable backups of your data and any customized applications or volume groups. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

Create a Customized bosinst.data File

1. Use the **cd** command to change your directory to the **/var/adm/ras** directory.
2. Copy the **/var/adm/ras/bosinst.data** file to a new name, such as **bosinst.data.orig**. This step preserves the original **bosinst.data** file.
3. Edit the **bosinst.data** file with an ASCII editor as follows:

   Note: The following example includes automatic acceptance of Software License Agreements.

```
control_flow:
    CONSOLE = /dev/lft0
    INSTALL_METHOD = overwrite
    PROMPT = no
    EXISTING_SYSTEM_OVERWRITE = yes
    INSTALL_X_IF_ADAPTER = yes
    RUN_STARTUP = yes
    RM_INST_ROOTS = no
    ERROR_EXIT =
    CUSTOMIZATION_FILE =
    TCB =
    INSTALL_TYPE = full
    BUNDLES =
    SWITCH_TO_PRODUCT_TAPE = no
    RECOVER_DEVICES = yes
```

```
    BOSINST_DEBUG = no
    ACCEPT_LICENSES = yes
    INSTALL_64BIT_KERNEL =
    INSTALL_CONFIGURATION =
    DESKTOP = CDE

target_disk_data:
    LOCATION =
    SIZE_MB =
    HDISKNAME = hdisk0

locale
    BOSINST_LANG = en_US
    CULTURAL_CONVENTION = en_US
    MESSAGES = C
    KEYBOARD = en_US
```

4. Verify the contents of the edited **bosinst.data** file using the **bicheck** command:

   ```
   /usr/lpp/bosinst/bicheck bosinst.data
   ```

5. Copy the edited file to the root directory:

   ```
   cp /var/adm/ras/bosinst.data /bosinst.data
   ```

Create a Supplementary Diskette

1. Create an ASCII file consisting of one word:

   ```
   data
   ```

2. Save the new ASCII file, naming it **signature**.
3. Copy the **signature** file to the root directory.
4. Create the supplementary diskette and use it for installation:

   Back up the edited **bosinst.data** file and the new **signature** file to the diskette with the following command:

   ```
   ls ./bosinst.data ./signature | backup -iqv
   ```

   Note: Make sure you back up the files to the diskette relative to the root path.

Boot from the AIX Product CD

1. Insert the diskette in the diskette drive of the target machine you are installing.
2. Insert the **AIX Volume 1** CD in the CD-ROM drive.
3. Boot the system.

The BOS installation program uses the customized **bosinst.data** file on the diskette, rather than the default **bosinst.data** file shipped with the installation media to answer the questions in the BOS menus. If you filled in the values correctly in the **bosinst.data** file, the BOS installation will begin. If the BOS installation program detects an error with a value in the **bosinst.data** file, the BOS menus will display with a message explaining what information needs to be corrected.

Create and Install a Software Bundle

Using this scenario, you can create a user-defined software bundle and install its contents. A user-defined software bundle is a text file ending in **.bnd** that is located in the **/usr/sys/inst.data/user_bundles** path. By creating the software bundle file in the **/usr/sys/inst.data/user_bundles** path, SMIT (System Management Interface Tool) can locate the file and display it in the bundle selection screen.

In this scenario, you will do the following:

- Create a user-defined software bundle that contains Netscape Communicator - U.S. Version, the HTTP Server Base Run-Time, and the Web-based System Manager Security application, which are located on the Expansion Pack
- Install the software bundle
- Verify the installation of the software bundle was successful

It is recommended that you first perform a system backup to ensure safe system recovery. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

The following procedure shows you how to create a software bundle and install its contents.

Step 1. Create a User-Defined Software Bundle

1. Create a text file with the extension **.bnd** in the **/usr/sys/inst.data/user_bundles** path by running the following:

   ```
   # vi /usr/sys/inst.data/user_bundles/MyBundle.bnd
   ```

2. Add the software products, packages, or filesets to the bundle file with one entry per line. Add a format-type prefix to each entry. For this example, we are dealing with AIX installp packages, so the format-type prefix is **I:**. Type the following in the *MyBundle*.bnd file:

   ```
   I:Netscape.communicator
   I:http_server.base
   I:sysmgt.websm.security
   ```

   For more information on installation format types, see Chapter 14, “Software Product Packaging Concepts,” on page 155.

3. Save the software bundle file and exit the text editor.

Step 2. Install the Software Bundle

1. Type the following at the command line:

   ```
   # smitty easy_install
   ```

2. Enter the name of the installation device or directory.
3. From the selection screen, select the name of the user-defined software bundle, *MyBundle*, you created.

```
                          Install Software Bundle

Type or select a value for the entry field.
Press Enter AFTER making all desired changes.
  +--------------------------------------------------------------------------+
  |                          Select a Fileset Bundle                         |
  |                                                                          |
  | Move cursor to desired item and press Enter.                             |
  |                                                                          |
  |   App-Dev                                                                |
  |   CDE                                                                    |
  |   GNOME                                                                  |
  |   KDE                                                                    |
  |   Media-Defined                                                          |
  |   MyBundle                                                               |
  |   ...                                                                    |
  |   ...                                                                    |
  |                                                                          |
  | F1=Help                 F2=Refresh              F3=Cancel                |
  | F8=Image                F10=Exit                Enter=Do                 |
  | /=Find                  n=Find Next                                      |
  +--------------------------------------------------------------------------+
```

4. Change the values provided in the Install Software Bundle screen as appropriate to your situation. You can change the **PREVIEW only?** option to yes to preview the installation of your software bundle before you install it. You might also need to **accept new license agreements** if the software in your bundle has an electronic license.

```
                          Install Software Bundle

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

                                                      [Entry Fields]
* INPUT device / directory for software               /cdrom
* BUNDLE                                              MyBundle               +
* SOFTWARE to install                                 [all]                  +
  PREVIEW only? (install operation will NOT occur)    no/yes                 +
  COMMIT software updates?                            yes                    +
  SAVE replaced files?                                no                     +
  AUTOMATICALLY install requisite software?           yes                    +
  EXTEND file systems if space needed?                yes                    +
  VERIFY install and check file sizes?                no                     +
  Include corresponding LANGUAGE filesets?            yes                    +
  DETAILED output?                                    no                     +
  Process multiple volumes?                           yes                    +
  ACCEPT new license agreements?                      no/yes                 +
  Preview new LICENSE agreements?                     no                     +

F1=Help             F2=Refresh          F3=Cancel           F4=List
Esc+5=Reset         F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

5. Press Enter to continue. Press Enter a second time to confirm your decision and begin the installation of your software bundle.
6. Check the installation summary at the end of the installation output by scrolling to the end of the output. The output indicates whether the installation of your user-defined software bundle was successful.

Step 3. Verify the Installation of the Software Bundle

- Check the installation summary at the end of the installation output by scrolling to the end of the output. The output indicates whether the installation of your user-defined software bundle was successful, you may see output similar to the following:

```
+-----------------------------------------------------------------------------+
                             Summaries:
+-----------------------------------------------------------------------------+

Installation Summary
--------------------
Name                        Level           Part        Event       Result
-------------------------------------------------------------------------------
sysmgt.websm.security       5.1.0.0         USR         APPLY       SUCCESS
sysmgt.websm.security       5.1.0.0         ROOT        APPLY       SUCCESS
http_server.base.source     1.3.12.2        USR         APPLY       SUCCESS
http_server.base.rte        1.3.12.2        USR         APPLY       SUCCESS
http_server.base.rte        1.3.12.2        ROOT        APPLY       SUCCESS
Netscape.communicator.com   4.7.6.1         USR         APPLY       SUCCESS
Netscape.communicator.us    4.7.6.1         USR         APPLY       SUCCESS
Netscape.communicator.com   4.7.6.1         ROOT        APPLY       SUCCESS
Netscape.msg.en_US.communic 4.7.6.1         USR         APPLY       SUCCESS
```

Add Open Source Applications to Your AIX System

The *AIX Toolbox for Linux Applications* CD that is shipped with your base operating system software contains the most commonly used open source applications that you can use with the AIX operating system. Your options for installing from this CD include:

- Using the SMIT **install_software** fast path to install **RPM** packages from the *AIX Toolbox for Linux Applications* CD.
- Using the **geninstall** command to install RPM packages from the *AIX Toolbox for Linux Applications* CD.
- Installing a bundle. Bundles group the applications you need for a basic Linux operating environment, basic desktop use, GNOME or KDE desktop use, or application development.
- Installing from a directory of packages classified by function. These directory groupings cover a broad range of applications, shell environments, network applications, development tools, application libraries, and so on.
- Installing a single package for a particular application.

The following procedures provide examples of installing RPM packages from *AIX Toolbox for Linux Applications* CD.

- To install the **cdrecord** and **mtools** RPM packages using SMIT, do the following:
  1. Run the SMIT **install_software** fast path.
  2. Enter the device name for the *AIX Toolbox for Linux Applications* CD (for example, `/dev/cd0`), and press Enter.
  3. Use the F4 key to list the contents of the device.
  4. Select the **cdrecord** and **mtools** packages, and press Enter.
  5. Accept the default values for the rest of the Install Software menu fields, and press Enter.
  6. Confirm that you do want to install the software, and press Enter.

  The software installation process begins at this point.
- To install the **cdrecord** and **mtools RPM** packages from the command line, type the following:

  ```
  # geninstall -d/dev/cd0 R:cdrecord R:mtools
  ```

  The software installation process begins at this point.
- Use the **rpm** command, which is automatically installed with the base operating system for AIX 5.1 and later versions, to install the bundles required for the GNOME desktop and the **bc** application package. Complete instructions are available on the README file of the *AIX Toolbox for Linux Applications for POWER Systems* CD.
  1. With your system powered on and AIX 5.1 or a later version running, insert the *AIX Toolbox for Linux Applications for POWER Systems* CD into the CD-ROM drive of your system.
  2. With root authority, mount the CD-ROM drive using the following command:

     ```
     mount -vcdrfs -oro /dev/cd0 /mnt
     ```

     The **-v** flag specifies the virtual file system type of **cdrfs**. The **-o** flag specifies the **ro** option, which means the mounted file is read-only. The device name is **/dev/cd0**. The directory in which you want to mount the CD-ROM is **/mnt**.
  3. Change to the **/mnt** directory by using the following command:

     ```
     cd /mnt
     ```
  4. Use the **ls** command to list the contents of the CD. The listing contains the following, which you can view or print:
     - The readme file contains complete instructions for installing from this CD.
     - The CONTENTS file lists all packages available on this CD and provides a short description of the purpose for each package.

5. In your Web browser, open the **/mnt/LICENSES/index.html** file to view software licensing information.
6. In your terminal window, change to the **ezinstall/ppc** directory by using the following command:

   ```
   cd /mnt/ezinstall/ppc
   ```

   In the next step, you use the **rpm** program to install GNOME by installing four bundles (Base, Desktop Base, GNOME Base, and GNOME Apps). Alternatively, you can install all necessary packages using the **smit install_bundle** fast path and selecting the GNOME bundle.
7. Install GNOME by using the following sequence of commands:

   ```
   rpm -Uhv ezinstall/ppc/base/*
   rpm -Uhv ezinstall/ppc/desktop.base/*
   rpm -Uhv ezinstall/ppc/gnome.base/*
   rpm -Uhv ezinstall/ppc/gnome.apps/*
   ```

   The **-U** flag updates any earlier versions of each package that you might have on your system. The **-h** flag prints hash marks (#) at timed intervals to indicate that the installation is progressing. The **-v** flag displays relevant informational or error messages that occur during the installation. Your result will look similar to the following:

   ```
   rpm -Uhv ezinstall/ppc/desktop.base/*
   gdbm            ##################################################
   libjpeg         ##################################################
   libpng          ##################################################
   libtiff         ##################################################
   libungif        ##################################################
   readline        ##################################################
   zlib            ##################################################
   ```

   If your **rpm** command returns an error, it is probably caused by one of the following:
   - Not enough space in your current file system. Resize the file system or change your mount point.
   - Package is already installed. The **rpm** program discovered an existing package of the same name and version level, so it did not replace the package. A script on the CD installs only those packages from a directory that are not already installed on your system, as shown in the following example:

     ```
     /mnt/contrib/installmissing.sh ezinstall/ppc/desktop.base/*
     ```
   - Failed dependencies. The packages listed in the error message must be installed before you can install this package or bundle.
8. Install the **bc** application package by using the following command:

   ```
   rpm -Uhv RPMS/ppc/bc-*.rpm
   ```

Clone a rootvg Using Alternate Disk Installation

Using this scenario, you can clone AIX running on **rootvg** to an alternate disk on the same system, install a user-defined software bundle, and run a user-defined script to customize the AIX image on the alternate disk. Because the **alternate disk installation** process involves cloning an existing **rootvg** to a target alternate disk, the target alternate disk must not be already assigned to a volume group.

In this scenario you will do the following:

- Prepare for the alternate disk installation
- Perform the alternate disk installation and customization
- Boot off the alternate disk
- Verify the operation

For information about alternate disk installation, see Chapter 13, “Alternate Disk Installation,” on page 147. Also, refer to the **alt_disk_install** man page.

Step 1. Prepare for the Alternate Disk Installation

1. Check the status of physical disks on your system. Type:

   ```
   # lspv
   ```

   Output similar to the following displays:

   ```
   hdisk0          0009710fa9c79877    rootvg    active
   hdisk1          0009710f0b90db93    None
   ```

   We can use `hdisk1` as our alternate disk because no volume group is assigned to this physical disk.
2. Check to see if the **alt_disk_install** fileset has been installed by running the following:

   ```
   # lslpp -L bos.alt_disk_install.rte
   ```

   Output similar to the following displays if the **alt_disk_install** fileset is not installed:

   ```
   lslpp: 0504-132  Fileset bos.alt_disk_install.rte not installed.
   ```
3. Using volume 2 of the AIX installation media, install the **alt_disk_install** fileset by running the following:

   ```
   # geninstall -d/dev/cd0 bos.alt_disk_install.rte
   ```

   Output similar to the following displays:

   ```
   +-----------------------------------------------------------------------------+
                                 Summaries:
   +-----------------------------------------------------------------------------+

   Installation Summary
   --------------------
   Name                        Level           Part        Event       Result
   -------------------------------------------------------------------------------
   bos.alt_disk_install.rte    5.2.0.0         USR         APPLY       SUCCESS
   ```
4. Create a user-defined bundle called **/usr/sys/inst.data/user_bundles/MyBundle.bnd** that contains the following filesets:

   ```
   I:bos.content_list
   I:bos.games
   ```

   For more information on how to create a user-defined software bundle, refer to “Create and Install a Software Bundle” on page 12.
5. Create the **/home/scripts** directory:

   ```
   mkdir /home/scripts
   ```
6. Create a user-defined customization script called **AddUsers.sh** in the **/home/scripts** directory:

```
touch /home/scripts/AddUsers.sh
chmod 755 /home/scripts/AddUsers.sh
```

7. Edit **/home/scripts/AddUsers.sh** to contain the following lines:

```
mkuser johndoe
touch /home/johndoe/abc.txt
touch /home/johndoe/xyz.txt
```

Step 2. Perform the Alternate Disk Installation and Customization

1. To clone the **rootvg** to an alternate disk, type the following at the command line to open the SMIT menu :

```
# smit alt_clone
```

2. Select **hdisk1** in the **Target Disk to Install** field.
3. Select the **MyBundle** bundle in the **Bundle to Install** field.
4. Insert volume one of the installation media.
5. Type `/dev/cd0` in the **Directory or Device with images** field.
6. Type `/home/scripts/AddUsers.sh` in the **Customization script** field.
7. Press Enter to start the alternate disk installation.
8. Check that the alternate disk was created, by running the following:

```
# lspv
```

Output similar to the following displays:

```
hdisk0          0009710fa9c79877    rootvg
hdisk1          0009710f0b90db93    altinst_rootvg
```

Step 3. Boot from the Alternate Disk

1. By default, the alternate-disk-installation process changes the boot list to the alternate disk. To check this run the following:

```
# bootlist -m normal -o
```

Output similar to the following displays:

```
hdisk1
```

2. Reboot the system. Type:

```
# shutdown -r
```

The system boots from the boot image on the alternate disk (`hdisk1`).

Step 4. Verify the Operation

1. When the system reboots, it will be running off the alternate disk. To check this, type the following:

```
# lspv
```

Output similar to the following displays:

```
hdisk0          0009710fa9c79877    old_rootvg
hdisk1          0009710f0b90db93    rootvg
```

2. Verify that the customization script ran correctly, by typing the following:

```
# find /home/johndoe -print
```

Output similar to the following displays:

```
/home/johndoe
/home/johndoe/.profile
/home/johndoe/abc.txt
/home/johndoe/xyz.txt
```

3. Verify that the contents of your software bundle was installed, by typing the following:

```
# lslpp -L bos.games bos.content_list
```

Output similar to the following displays:

```
  Fileset                      Level  State  Description
  ----------------------------------------------------------------------------
  bos.content_list           5.2.0.0    C    AIX Release Content List
  bos.games                  5.2.0.0    C    Games
```

Configure NIM Using EZNIM

Using this scenario, you will use the SMIT EZNIM option to configure the NIM environment for the first time. The SMIT EZNIM option installs the **bos.sysmgt.nim.master** fileset and configures the NIM environment. The configuration involves creating the NIM database and populating it with several default entries. Several basic NIM resources will then be created and defined in the NIM database.

1. Type the following:
   ```
   # smitty eznim
   ```
2. Select **Configure as a NIM Master**, and press Enter.
3. Select **Setup the NIM Master Environment**, and press Enter.
4. Verify that the default selections for software source, volume group, and file system are correct for your environment. Change the selections, if needed.
5. Press Enter to begin configuring the NIM environment.
6. To display the NIM resources that have been created, do the following:
   a. Use the SMIT **eznim_master_panel** fast path to open the EZNIM Master menu.
   b. Select **Show the NIM environment**, and press Enter.

Install a Client Using NIM

Using this scenario, you can perform a new and complete BOS (base operating system) installation on a NIM client. It is assumed that you have already configured the NIM master, defined the basic NIM resources, and defined the NIM client you want to install.

For a guide on configuring the NIM environment and defining resources, see Chapter 18, “NIM Task Roadmap,” on page 185.

In this scenario, you will do the following:

1. Perform an BOS (**rte**) installation.
2. Use a **bosinst_data** resource to perform a nonprompted installation. For information on how to create a **bosinst.data** file for nonprompted installation, see “bosinst.data File Example” on page 78.
3. Use a **resolv_conf** resource to configure the network nameserver and domain.

It is recommended that you first perform a system backup on the client to ensure safe system recovery. For instructions on how to create a system backup, see “Creating System Backups” on page 127.

1. Type the following:
   ```
   # smit nim_bosinst
   ```
2. Select a target for the BOS installation operation.
3. Select **rte** installation for the installation type.
4. Select the **lpp_source** resource for the BOS installation.
5. Select the **SPOT** resource for the BOS installation.
6. Select the **BOSINST_DATA to use during installation** option, and select a **bosinst_data** resource that is capable of performing a nonprompted BOS installation.
7. Select the **RESOLV_CONF to use for network configuration** option, and select a **resolv_conf** resource.
8. Select the **Accept New License Agreements** option, and select **Yes**.
9. Accept the default values for the remaining menu options.
10. Press Enter to confirm and begin the NIM client installation.
11. To check the status of the NIM client installation, type:
    ```
    # lsnim -l va09
    ```

    Output similar to the following displays:

    ```
    va09:
       class          = machines
       type           = standalone
       default_res    = basic_res_grp
       platform       = chrp
       netboot_kernel = up
       if1            = master_net va09 0
       cable_type1    = bnc
       Cstate         = Base Operating System installation is being performed
       prev_state     = BOS installation has been enabled
       Mstate         = in the process of booting
       info           = BOS install 7% complete : 0% of operations completed.
       boot           = boot
       bosinst_data   = bid_tty_ow
       lpp_source     = 520lpp_res
       nim_script     = nim_script
       resolv_conf    = master_net_conf
    ```

```
spot           = 520spot_res
cpuid          = 0009710F4C00
control        = master
Cstate_result  = success
```

Install AIX from a Linux Server Using NIM on Linux

Network Installation Management (NIM) on Linux (NIMOL) provides basic AIX software administration services from a Linux server. The NIMOL RPM included in the AIX product media provides basic NIM functionality for a Linux server. While using a pSeries NIM Server for AIX client management provides more functionality, NIMOL provides the following functions on Linux servers:

- Configuring services required to install AIX with NIM
- Performing a network installation of AIX
- Installing software updates on an AIX system
- Listing software installed on an AIX system
- Creating a backup of an AIX system that NIMOL can use to install other systems with NIM

NIMOL is supported on SUSE LINUX 9.0 and Redhat 9.0. Read the NIMOL readme file in the **/usr/share/doc/nimol** directory for information on updates.

Note: NIMOL supports AIX 5L Version 5.2 with the 5200-04 Recommended Maintenance package and later.

Commands

The following commands are shipped with the NIMOL RPM:

nimol_config
Configures a Linux server to install AIX with NIM

nimol_install
Sets up a client for AIX installation

nimol_lslpp
Lists software installed on an AIX client

nimol_update
Installs software updates onto an AIX client

nimol_backup
Creates an installable backup of an AIX client

Note: These commands are available on the Linux server. For additional information on these commands, run the **man** command on the Linux server with the name of the specific NIMOL command.

Configuring the Linux Server

To configure the Linux server, complete the following:

1. Insert the AIX product CD that contains the NIMOL RPM into the Linux server.
2. Install the followingRPMs, which are requisite for SUSE LINUX 9.0 and RedHat 9.0:
   - **dhcp**
   - **tftp**
   - **nfs-utils**
   - **gettext**

   The following RPMs are additional requisites for Redhat 9.0 only:
   - **dhcp-server**
   - **tftp-server**
3. Insert the AIX product CD Volume 1 into the Linux server.
4. Install the NIMOL RPM by running the following commands:

```
mount /dev/cdrom
rpm -ivh /mnt/cdrom/RPMS/linux/nimol*
```

5. Run the following command to configure the Linux server:

```
nimol_config -C
```

The **nimol_config** command configures the basic services required to install AIX on clients with NIM, such as bootp, tftp, nfs, and syslog. The command uses the dhcp server to respond to client bootp requests.

Note:

> You should not manually modify the **/etc/dhcpd.conf** file because the modifications might interfere with the ability of NIMOL to add and remove subnet and host entries.

Creating AIX Installation Resources

To create AIX installation resources, you need access to an existing AIX client.

1. Ensure that your NIMOL Linux Server has rhost permissions to the AIX client by editing the **/.rhosts** file on the AIX client.
2. Run the **nimol_backup** command on the Linux Server. For example, if the client is named **myaixclient.ibm.com** and you want to name the installation resources **aix530**, then run the following command:

   ```
   nimol_backup -c myaixclient.ibm.com -L aix530
   ```

   The **nimol_backup** command will create a **mksysb** backup of the client, as well as the other resources required to perform a network boot in the **/export/aix/aix530** directory.

Alternately, you can use the **nimol_config** command to define existing AIX installation resources that reside on your NIMOL Server. For example, if you have existing AIX installation resources located in **/export/aix/aix530**, run the following command to define the resources for use with NIMOL:

```
nimol_config -t /export/aix -L aix530
```

Setting up a NIMOL Client

You can set up a client for the installation of AIX with NIM by running the **nimol_install** command. Before you run the **nimol_install** command, you will need the following information for the client:

- Hostname
- MAC address
- Subnet mask

The following examples show how to obtain your MAC address.

To obtain the MAC address for the **ent0** network adapter on a running client, run the following command:

```
lscfg -vl ent0
```

To obtain the MAC address on an IBM eServer pSeries 690 client that is not running, complete the following:

1. Power on the system.
2. Press F1 when the keyboard icon is displayed to access the SMS menus.
3. Select **Boot Options**.
4. Select **Install or Boot a Device**.
5. Select **Network** This menu refers to your network card (for example, ethernet).
6. Select **Information**. This screen will contain the MAC address.

When specifying the MAC address to **nimol_install**, colons are required.

For example, to set up a client with a **client1** hostname, 255.255.255.0 subnet mask, 192.168.1.1 gateway, and 00:60:08:3F:E8:DF MAC address with the ″aix530″ installation resources, run the following command:

```
nimol_install -c client1 -g 192.168.1.1 -m 00:60:08:3F:E8:DF -s 255.255.255.0 -L aix530
```

After you run **nimol_install**, you can specify a network boot using the **bootlist** command on a running AIX system. For example, if the client installs AIX from the **ent0** adapter with a 192.168.1.20 client IP address, and the NIMOL server has a 192.168.1.10 IP address and is on the same subnet as the client, complete the following:

1. Run the **bootlist** command as follows:

   ```
   bootlist -m normal ent0 bserver=192.168.1.10 gateway=0.0.0.0 client=192.168.1.20
   ```

2. Run the following command to begin the installation:

   ```
   shutdown -Fr
   ```

   Note: If the client is not running, power on the system and enter the system management services (SMS) menus by pressing F1 when the keyboard icon is displayed. You can then specify the network boot parameters and select the network device as the boot device.

The **syslog** daemon is configured to receive remote status by default. This setting allows you to monitor the installation by viewing the **/var/log/nimol.log** file on the NIMOL server.

After the installation is complete, you can remove the setup for the client named **client2** by running the following command:

```
nimol_install -c client1 -r
```

Maintaining the NIMOL Client

You can use the following commands to maintain the NIMOL client:

- **nimol_update**
- **nimol_lslpp**

Note: The default option for the **nimol_install** command is intended for use on the client machine where you installed AIX. If you specify the **-n** option with the **nimol_install** command when you install AIX on the client machine, these commands will not be available.

The following scenarios are tasks involved in maintaining the NIMOL client:

- To list the software that is installed on **client1**, run the following command:

  ```
  nimol_lslpp -c client1 -f "-L"
  ```

- To install the **perl.rte** software package from the aix530 installation resources, run the following command:

  ```
  nimol_update -c client1 -L aix530 -p "perl.rte"
  ```

Notes:

1. To install additional software, copy the **installp** images into the **lpp_source/installp/ppc** sub-directory of the installation resources directory. For example, if your installation resources are located in the **/export/aix/aix530** directory, you will need to copy the **installp** images into the **/export/aix/aix530/lpp_source/installp/ppc** directory. You also must copy the **.toc** file because Linux machines cannot create this file.
2. To use a different remote access other than rhost permissions, you can specify the **-m** option with the **nimol_update** command, the **nimol_lslpp** command, or the **nimol_backup** command. The alternative remote access must be configured prior to using it with the NIMOL commands, and the remote access method cannot prompt for a password. For example, the following command will use the **/usr/bin/ssh** directory to remotely access the **client1** during a software listing:

   ```
   nimol_lslpp -c client1 -f "-L" -m /usr/bin/ssh
   ```

Unconfiguring the NIMOL Server

To unconfigure the NIMOL server, complete the following:

1. Determine if clients are set up to install by running the following command:

   ```
   nimol_install -l
   ```

2. Remove any clients that are listed as set to install by running the following command:

   ```
   nimol_install -r -c client
   ```

3. Remove the AIX installation resource labels that you created by running the following command:

   ```
   nimol_config -r -L aix530
   ```

   Removing these labels removes all resources from the NIMOL server. You can define them again if you decide to configure the NIMOL server in the future.

4. Unconfigure NIMOL by running the following command:

   ```
   nimol_config -U
   ```

   This command restores services to the state they were before you configured NIMOL. For example, if you had the **nfs** service and the **tftp** service turned off prior to configuring NIMOL, then they are turned off again.

5. Remove the NIMOL RPM by running the following command:

   ```
   rpm -e nimol
   ```

Performing a Network Installation of a JS20 Blade

Using this scenario, you can install AIX for the first time or overwrite an existing version of the operating system onto a JS20 blade. In this scenario, you will do the following:

- Gather the required TCP/IP information for your JS20 blade.
- Prepare your Network Installation Management (NIM) environment.
- Configure a NIM master.
- Create NIM installation resources.
- Define your JS20 blade as a NIM client.
- Prepare your JS20 blade for a network installation.
- Boot the JS20 blade off the network using a directed bootp or broadcast bootp method.

To perform a network install, you will need to configure a NIM master if you do not already have one configured. For instructions on how to create a NIM master, see Configuring the NIM Master and Creating Basic Installation Resources

Note: This procedure requires shutting down and reinstalling the base operating system. When you reinstall any operating system, schedule your downtime when it least impacts your workload to protect yourself from a possible loss of data or functionality. Before you perform a new and complete overwrite installation, ensure you have reliable backups of your data and any customized applications or volume groups. For instructions on how to create a system backup, see Creating System Backups.

Perform a Broadcast bootp Network Installation

To perform a broadcast bootp, ensure that the NIM server is on the same subnet as the JS20 blade that you are installing. During a broadcast bootp, the JS20 blade sends a broadcast bootp packet to its subnet. The NIM server receives and responds to this packet. The JS20 blade NIM client definition on your NIM master must include the MAC address of the JS20 blade's network adapter used during the installation, or the NIM server will not respond to a broadcast bootp. If your NIM master is on a different subnet, and you want to perform a broadcast bootp, then you must set up another system on the client's subnet to forward broadcast bootp packets.

To set up a system to forward broadcast bootp packets, complete the following steps:

1. Add the IP address of your NIM server to the **/etc/dhcprd.cnf** file on the system that forwards the packets. For example, if your NIM server's IP address is 192.24.24.1, add **server 192.24.24.1** to **/etc/dhcprd.cnf**
2. Run **startsrc -s dhcprd**.

This system will now forward broadcast bootp packets to your NIM server that is on a different subnet. You must also install the latest firmware version onto your client for the client to correctly handle the bootp response from the system that is forwarding broadcast bootp packets.

Step 1: Prepare Your NIM Server

Performing a broadcast bootp is similar to a directed bootp. The steps are the same, except you that must obtain the MAC address of the JS20 blade's network adapter that you will use to network install.

1. Obtain the MAC address from the MM Web interface by completing the following steps:
   a. Select **Monitors –> Hardware VPD**.
   b. Scroll down to **BladeCenter Server MAC Addresses**.
   c. Find the JS20 blade that you plan to install and the MAC address that corresponds to the adapter you will use to perform the installation. Write this MAC address down.
2. Specify the MAC Address when you define the JS20 blade as a NIM client. If you are using the **smitty nim_mkmac** command, specify the MAC address on the **Network Adapter Hardware Address** SMIT

screen. Do not include the colons (″:″) when you are specifying the MAC address. If the client is already defined, you can change the MAC address with the **smitty nim_chmac** command.

3. Set up your NIM master to install the JS20 blade, by completing the following steps:
   a. Run the **smitty nim_bosinst** command.
   b. Select the JS20 blade that is defined as your target .
   c. Select the type of install that you want to perform and select the installation resources that you want to use to install the JS20 blade.

   You can also prepare the JS20 blade to install using the **bos_inst** NIM operation on the command line. For more information on using the **bos_inst** operation, see bos_inst.
4. Power off the JS20 blade. If you do not want the JS20 blade to reboot automatically, set **Initiate reboot and installation now?** to **no** in the SMIT screen and press Enter.

Step 2: Initiate the Installation from the Management Module

NIM uses the bootp protocol when performing a network installation. The bootp protocol allows you to install through a directed bootp or broadcast bootp request.

1. Ensure that the JS20 blade's boot list is set to install from the network in the MM Web interface by selecting **Blade Tasks –> Configuration** and scroll down **Boot Sequence**.
2. Click the JS20 blade that you are installing and ensure the first device listed is **Network - BOOTP**. When the JS20 blade boots, it will install from the first network adapter that receives a bootp response.

   Note: You should not have a serial over LAN connection open to the JS20 blade that you are attempting to install when you power on the JS20 blade.
3. Click **save**.
4. Power on the JS20 blade from the MM Web interface by selecting **Blade Tasks –> Power/Restart**.
5. Select the JS20 blade that you are installing and click **Power On Blade**.

If you do not have an serial over LAN connection to the JS20 blade, you can view the status of the installation by running the following command from your NIM master:

```
lsnim -l js20_nim_name
```

For example, if the JS20 blade was defined as **JS20blade1**, run the following command:

```
lsnim -l JS20blade1
```

Note: If you run the AIX **bootlist** command to set the IP parameters for a network adapter and reboot the system, the IP parameters will be stored in NVRAM. When you reboot the JS20 blade from the MM with the boot sequence set to **Network-BOOTP**, the JS20 blade attempts to use the IP parameters stored in NVRAM instead of performing a broadcast bootp. To perform a broadcast bootp, run the **bootlist** command specifying 0.0.0.0 for each IP parameter and reboot from AIX using the **shutdown -Fr** command. For example, to perform a broadcast bootp over **ent1**, run the following commands.

```
# bootlist -m normal ent1 client=0.0.0.0 bserver=0.0.0.0 gateway=0.0.0.0 hdisk0
# shutdown -Fr
```

If you are unable to log into the AIX system, then follow the instructions for performing a directed bootp via the Open Firmware prompt, but specify ″0.0.0.0″ for each IP address. Once the JS20 blade installs successfully, the boot IP parameters are reset to ″0.0.0.0″.

Perform a Directed bootp Network Installation

A directed bootp can be used to install a JS20 blade from a NIM server and does not require the NIM server to be on the same subnet as the JS20 blade. This option does not require that you have the MAC address of the network adapter on the JS20 blade. To perform a directed bootp, you need a serial over LAN connection to the blade so that you can specify the IP parameters to Open Firmware. Currently you

must have 2 network adapters to perform a NIM installation if you are using serial over LAN. You cannot install AIX over the same adapter that is using serial over LAN.

Step 1: Prepare Your NIM Server

1. Create a SPOT, **lpp_source**, and any other resources that you will need at the level of AIX that you want to install on your NIM server. Your NIM server is usually the NIM master, but you can also set up a NIM client as a NIM server. For instructions on how to create NIM resources, see Configuring the NIM Master and Creating Basic Installation Resources.
2. Ensure that you have the information in the following worksheet for your JS20 blade before proceeding with the installation:

Table 2. Network Configuration Information Worksheet

Network Attribute	Value
Network Interface	(For example: ent1)
Host Name	
IP Address	_____._____._____._____
Network Mask	_____._____._____._____
Name server	_____._____._____._____
Domain Name	
Gateway	_____._____._____._____

3. Define the JS20 blade as a NIM client on your NIM master by running the **smitty nim_mkmac** command on the NIM master. This command creates a client definition for your JS20 blade. You can also define the JS20 blade using the **define** NIM operation on the command line.
4. If you want to set the JS20 blade's name server and domain name after the installation, use a **resolv_conf** resource. For more information on creating a **resolv_conf** resource, see **resolv_conf** Resource.
5. Set up your NIM master to install the JS20 blade, by running the **smitty nim_bosinst** command. Select the JS20 blade that you defined earlier as your target. Then select the type of install that you want to perform and select the installation resources that you want to use to install the JS20 blade. You can also prepare the JS20 blade to install using the **bos_inst** NIM operation on the command line.

   Notes:

   a. If the JS20 blade is powered off or has never been installed, set **Initiate reboot and installation now?** to **no** and press enter in the SMIT interface.

   b. If the JS20 blade is powered on and running AIX, set **Initiate reboot and installation now?** to **yes** in the SMIT interface. If you choose this option, a directed bootp is initiated by default and you can skip step 2. Before you run this command, ensure that the JS20 blade is a registered NIM client. To do this, run **smitty niminit** on the JS20 blade. Then specify the hostname of your NIM master and the interface you want to use for the installation. You can also initialize the JS20 blade using the **niminit** command on the command line.

Step 2: Specify a Directed bootp from the JS20 blade

1. Open a Web interface to the MM by navigating to the IP address or hostname of the MM using a Web browser.
2. Enable serial over LAN to the JS20 blade from the MM Web interface by selecting **Blade Tasks –> Serial Over LAN**.
3. Select the JS20 blade that you are installing and click **Enable Serial Over LAN**.
4. Power on the JS20 blade from the MM Web interface by selecting **Blade Tasks –> Power/Restart**.
5. Select the JS20 blade that you are installing and click **Power On Blade**.

6. Open a serial over LAN connection to the JS20 blade by telnetting into the MM and running the **console** command. For example, if the JS20 blade is in slot 3, you would run the following command:

   ```
   console -T blade[3]
   ```

   The serial over LAN connection shows a series of LED numbers.
7. Press 8 on the keyboard when you see **E1F1** to go to the Open Firmware prompt.
8. Run **boot net:bootp,server_ip,,client_ip,gateway_ip** to boot from the network.
   - If you are using a **net** type boot, you would run a command similar to the following:

     ```
     boot net:bootp,192.168.2.10,,192.168.1.11,192.168.1.1
     ```
   - If you are using **ent1**, then you would run a command similar to the following:

     ```
     boot /pci@8000000f8000000/pci@0/ethernet@1,1:bootp,192.168.2.10,,192.168.1.11,192.168.1.1
     ```

     Note: You must specify the full device path name with this command. To determine the full path to your device, list the device tree by running the **ls** command at the Open Firmware prompt. This command displays output similar to the following:

     ```
     0 > ls
     000000c87f18: /ibm,serial
     000000c88840: /chosen
     000000c88a98: /packages

     ...
     000000d31488: /vdevice
     000000d327a8:   /vty@0
     000000d32f88:   /IBM,sp@4000
     000000d33f10:   /rtc@4001
     000000d34a18: /pci@8000000f8000000
     000000d384d0:   /pci@0
     000000d4bbd0:     /ethernet@1
     000000d5af50:     /ethernet@1,1
     000000d3be00:   /pci@3
     000000d6a350:     /usb@0
     000000d845f8:       /hub@1
     000000d854b8:     /usb@0,1
     000000d9f760:       /hub@1
     000000d3f798:   /pci@1f
     000000d45ed8:   /ide@4,1
     000000d47b10:     /disk@0
     ```

     The highlighted items are the path to the second ethernet adapter. You would pass this information to the **boot** command to initiate a network boot from the second ethernet adapter
9. After you run the **boot** command, then network installation begins. Output similar to the following is displayed on the serial over LAN connection:

   ```
   BOOTP: chosen-network-type = ethernet,auto,none,auto
   BOOTP: server   IP =        192.168.2.10
   BOOTP: requested filename =
   BOOTP: client   IP =        192.168.1.11
   BOOTP: client   HW addr =   0 d 60 1e c cb
   BOOTP: gateway  IP =        192.168.1.1
   BOOTP: device    /pci@8000000f8000000/pci@0/ethernet@1,1
   BOOTP: loc-code  U8842.P1Z.23A0984-P1-T7

   BOOTP R = 1
   FILE: /tftpboot/js20blade1.austin.ibm.com
   Load Addr=0x0000000000004000, Max Size=0x0000000000bfc000
   FINAL Packet Count = 21131
   FINAL File Size = 10818623 bytes.
   load-base=0x4000
   ```

```
real-base=0xc00000

Elapsed time since release of system processors: 2 mins 28 secs
...
```

Create a System Backup to Tape

Using this scenario, you can create and verify a bootable system backup, also known as a *root volume group backup* or *mksysb image*.

Step 1. Prepare for System Backup Creation

Before creating system backups, complete the following prerequisites:

- Be sure you are logged in as root user.
- If you plan to use a backup image for installing other differently configured target systems, you must create the image *before* configuring the source system, or set the RECOVER_DEVICES variable to `no` in the **bosinst.data** file. For more information about the **bosinst.data** file, refer to The bosinst.data File in the *AIX 5L Version 5.2 Installation Guide and Reference*.
- Consider altering passwords and network addresses if you use a backup to make master copies of a source system. Copying passwords from the source to a target system can create security problems. Also, if network addresses are copied to a target system, duplicate addresses can disrupt network communications.
- Mount all file systems you want to back up. The **mksysb** command backs up only mounted JFS (journaled file systems) in the **rootvg**. To back up file systems, use the **mount** command.

  Note: The **mksysb** command does not back up file systems mounted across an NFS network.
- Unmount any local directories that are mounted over another local directory.

  Note: This backup procedure backs up files twice if a local directory is mounted over another local directory in the same file system. For example, if you mount **/tmp** over **/usr/tmp**, the files in the **/tmp** directory are then backed up twice. This duplication might exceed the number of files that a file system can hold, which can cause a future installation of the backup image to fail.
- Use the **/etc/exclude.rootvg** file to list files you do not want backed up.
- Make at least 8.8 MB of free disk space available in the **/tmp** directory. The **mksysb** command requires this working space for the duration of the backup.

  Use the **df** command, which reports in units of 512-byte blocks, to determine the free space in the **/tmp** directory. Use the **chfs** command to change the size of the file system, if necessary.

  For example, the following command adds 12 MB of disk space to the **/tmp** directory of a system with 4 MB partitions:

  ```
  # chfs -a size=+24000 /tmp
  ```
- All hardware must already be installed, including external devices, such as tape and CD-ROM drives.
- The **bos.sysmgt.sysbr** fileset must be installed. The **bos.sysmgt.sysbr** fileset is automatically installed in AIX 5.2. To determine if the **bos.sysmgt.sysbr** fileset is installed on your system, type:

  ```
  # lslpp -l bos.sysmgt.sysbr
  ```

  If the **lslpp** command does not list the **bos.sysmgt.sysbr** fileset, install it before continuing with the backup procedure. Type the following:

  ```
  # installp -agqXd /dev/cd0 bos.sysmgt.sysbr
  ```

Step 2. Create a System Backup to Tape

1. Enter the **smit mksysb** fast path.
2. Select the tape device in the **Backup DEVICE or File** field.
3. If you want to create map files, select **yes** in the **Create Map Files?** field.

   For more information, see Using Map Files for Precise Allocation in *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*.

Note: If you plan to reinstall the backup to target systems other than the source system, or if the disk configuration of the source system might change before reinstalling the backup, do not create map files.

4. To exclude certain files from the backup, select **yes** in the **Exclude Files** field.

   For more information, see Excluding Files in System Backups in *AIX 5L Version 5.2 Installation Guide and Reference*.
5. Select **yes** in the **List files as they are backed up** field.
6. Select **yes** in the **Disable software packing of backup?** field, if you are running any other programs during the backup.
7. Use the default values for the rest of the menu options.
8. Press Enter to confirm and begin the system backup process.
9. The COMMAND STATUS screen displays, showing status messages while the system makes the backup image. When the backup process finishes, the **COMMAND:** field changes to **OK**.
10. To exit SMIT when the backup completes, press F10 (or Esc+0).
11. Remove the tape and label it. Write-protect the backup tape.
12. Record any backed-up root and user passwords. Remember that these passwords become active if you use the backup to either restore this system or install another system.

You have successfully created the backup of your **rootvg**. Because the system backup contains a boot image, you can use this tape to start your system if for some reason you cannot boot from hard disks.

Clone a System Using a System Backup Tape

With a **mksysb** image, you can clone one system image onto multiple target systems. The target systems might not contain the same hardware devices or adapters, require the same kernel (uniprocessor or microprocessor), or be the same hardware platform as the source system.

Beginning in AIX 5.2, all devices and kernel support are installed by default during the base operating system (BOS) installation process. If the **Enable System Backups to install any system** selection in the Install Software menu is set to **yes**, you can create a **mksysb** image that boots and installs supported systems. Verify that your system is installed with all devices and kernel support, by typing the following:

```
# grep ALL_DEVICES_KERNELS /var/adm/ras/bosinst.data
```

Output similar to the following displays:

```
ALL_DEVICES_KERNELS = yes
```

Use this scenario if your system was not installed with all devices and kernel support during BOS installation. Be sure to boot from the appropriate product media for your system and at the same maintenance level of BOS (base operating system) as the installed source system on which the **mksysb** was made. For example, use BOS AIX 5.2 media with a **mksysb** from a BOS AIX 5.2 system. Use this how-to when installing a system backup tape to a different system.

In this scenario, you will do the following:

1. Boot the system with the *AIX Volume 1* CD in the CD-ROM drive and the system backup tape in the tape device.
2. Select **Start Maintenance Mode for System Recovery**.
3. Select **Install from a System Backup**.
4. Select the drive containing the backup tape, and press Enter.

The system reads the media and begins the installation.

You are then prompted for the BOS installation language, and the Welcome screen displays. Continue with the Prompted Installation, because cloning is not supported in nonprompted installations.

If you are cloning from the product CD to restore a backup tape, do not remove the CD from the CD-ROM drive.

After the **mksysb** installation completes, the installation program automatically installs additional devices and the kernel (uniprocessor or microprocessor) on your system, using the original product media you booted from. Information is saved in BOS installation log files. To view BOS installation log files, type `cd /var/adm/ras` and view the **devinst.log** file in this directory.

If the source system does not have the correct passwords and network information, you can make modifications on the target system now. Also, some products (such as graPHIGS) ship device-specific files. If your graphics adapter is different on the target system, verify that the device-specific filesets for graphics-related LPPs are installed.

Clean Up a Failed Software Installation

Using this scenario, you can clean up software products and service updates after an interrupted or failed installation. The cleanup procedure attempts to delete items that were partially installed or left in an incomplete state. This scenario applies only to the update or installation of optional software products. If your AIX 5.2 BOS installation was unsuccessful, see Chapter 15, “Troubleshooting Operating System and Optional Software Installation,” on page 159.

Note: It is recommended that you first perform a system backup before installing software updates to ensure safe system recovery. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

The cleanup procedure attempts to revert the update to its previous state. For example, when cleaning up an update that was interrupted in the **COMMITTING** state, the cleanup procedure attempts to return the update to its **APPLIED** state.

If an update installation is interrupted, run the **lslpp -l** command to see the current state of the update. For example, if you run **lslpp -l** on an interrupted update installation, it might report the update status as **APPLYING** rather than **APPLIED**.

If the interruption occurs during the initial state of an installation, then the cleanup procedure attempts to delete the installation entirely and restore the previous version of the product (if there is one). When the previous version is restored, it becomes the active version. When the previous version cannot be restored, the software is listed by the **lslpp -l** command as **BROKEN**.

When the product is deleted or **BROKEN**, you can attempt to reinstall the software. Any product in the **BROKEN** state cannot be cleaned up; it can only be reinstalled or removed.

To initiate a cleanup procedure using SMIT:

1. Type `smit maintain_software` on the command line.
2. Select **Clean Up After Failed or Interrupted Installation** and press Enter.

To initiate a cleanup procedure from the command line:

Type `installp -C` on the command line and press Enter.

If prompted to reboot (restart) the system after running the cleanup procedure, then do so now.

If you get a message indicating that no products were found that could be cleaned up, you may have executed the cleanup procedure when it was not needed. Try your installation again.

Part 2. Operating System and Optional Software Installation

Part 2 provides information about installing and configuring AIX on a standalone system, regardless of network connectivity. The chapters in this part describe the different base operating system installation methods, including new and complete overwrite installation, migration installation, and preservation installation.

© Copyright IBM Corp. 2002, 2004

Chapter 2. What's New in Base Operating System Installation and Software Installation?

This chapter gives an overview of the new features in the base operating system (BOS) installation and software installation procedures for AIX 5.2.

BOS Menus

Installation options are available for you to configure your system during a BOS installation process.

The installation options include the following:

- Desktop selection
- Trusted Computing Base
- Controlled Access Protection Profile (CAPP) and Evaluation Assurance Level 4+ (EAL4+)
- 64-bit kernel enablement
- Enhanced journaled file system (JFS2)
- Graphics software
- Documentation services software
- Installation of all device and kernel filesets

For information about the installation options, see Chapter 4, "Installation Options," on page 47.

Software Maintenance and Utilities

The Copy Software Bundle to Hard Disk for Future Installation option is available in the SMIT Software Maintenance and Utilities menu.

For information about the SMIT Copy Software Bundle to Hard Disk for Future Installation option, see "Copy Software Bundle to Hard Disk for Future Installation" on page 89.

Software Service Management

The SMIT Software Service Management menu allows you to manage how you update systems with fixes and preventive maintenance package levels. You can compare levels of different systems against a base system or a base set of fixes. Support is provided through the command line and SMIT to allow you to compare the filesets installed on your standalone system with the contents of an image repository or a service report that is downloaded from the IBM eServer pSeries support Web site.

This option also allows you clean up or rename software images in a repository. The following software maintenance options are available in the SMIT Software Service Management menu and the SMIT Software Maintenance and Utilities menu:

- Rename Software Images in Repository
- Clean Up Software Images in Repository

The Rename Software Images in Repository option allows you to rename software images with the PTF prefix to the fileset names generated by the **bffcreate** command. This allows you to determine exactly what updates are contained in a directory because the fileset name is recognizable.

The Clean Up Software Images in Repository option allows you to perform various tasks to clean up software image directories on standalone systems. This allows you to remove duplicate or unnecessary filesets from image repositories, easing management of the images and reducing the amount of space required to store them.

© Copyright IBM Corp. 2002, 2004

For information about the SMIT Software Service Management menu, see “Software Service Management” on page 91.

Emergency Fix Management

The emergency fix (efix) management solution allows users to track and manage efixes on a system. The efix management solution consists of: the efix packager (**epkg**) command and the efix manager (**emgr**) command.

The **epkg** command creates efix packages that can be installed by the **emgr** command. The **emgr** command installs and removes efix packages created with the **epkg** command.

The efix management solution provides the following functions:

- Emergency fixes can be packaged in either interactive or template-based modes
- List all of the efixes on a given system
- Install efixes
- Remove efixes
- Force remove efixes
- Verify efixes
- Display efix package locks

For more information about efix management, see “Emergency Fix Management” on page 98.

Alternate Disk Migration Installation

Alternate disk migration installation allows the user to create a copy of **rootvg** to a free disk (or disks) and simultaneously migrate it through Network Installation Management (NIM) to a new release level. Using alternate disk migration installation over a conventional migration provides several advantages:

- Less downtime (the migration is performed while the system is up normally, and there is no need to boot from any media).
- Quick recovery in case of migration failure.
- High degree of flexibility and customization.

For more information about alternate disk migration installation, see “Alternate Disk Migration Installation” on page 149.

Universal Disk Format Support for DVD-RAM

The Universal Disk Format (UDF) allows you to manipulate files directly on the DVD-RAM media. The system backup image is an archived file composed of many files that cannot be manipulated. However, the installation packages and any files that are not contained in the backup image can be directly manipulated on the DVD-RAM. After the DVD is mounted, the files can be changed by using an editor, or new files can be copied to the DVD using the various copy and restore commands (such as the **cp**, **mv**, and **restore** commands).

For information about the SMIT Software Service Management menu, see “DVD-RAM and Universal Disk Format” on page 134.

Installation Commands

The following commands have been added to AIX:

- The **lsmksysb** command. For further information, see “Backup Options” on page 137.

- The **install_all_updates** command. For further information, see “Update Installed Software to Latest Level from the Command Line” on page 86.
- The **lppmgr** command. For further information, see “Manage an Existing installp Image Source” on page 90.
- The **compare_report** command. For further information, see “Comparison Reports” on page 91.

Chapter 3. Introduction to Base Operating System Installation

This chapter provides information about completing an installation of the AIX 5.2 base operating system (BOS).

The BOS installation program first restores the run-time **bos** image, then installs the appropriate filesets, depending on your selections. The installation program automatically installs required message filesets, according to the language you choose.

In the BOS menus, you can also configure the following options:

- Desktop selection
- Trusted Computing Base
- Controlled Access Protection Profile (CAPP) and Evaluation Assurance Level 4+ (EAL4+)
- 64-bit kernel enablement
- Enhanced journaled file system (JFS2)
- Graphics software
- Documentation services software
- Installation of all device and kernel filesets

For more information about the installation options, refer to Chapter 4, “Installation Options,” on page 47.

The following installation methods are available on AIX:

New and Complete Overwrite
This method installs AIX 5.2 on a new machine or completely overwrites any BOS version that exists on your system.

For instructions on installing AIX 5.2 on a new machine or to completely overwrite the BOS on an existing machine, refer to Chapter 5, “New and Complete Overwrite Installation / Preservation Installation,” on page 51.

Preservation
This method replaces an earlier version of the BOS but retains the root volume group, the user-created logical volumes, and the **/home** file system. The system file systems **/usr**, **/var**, **/tmp**, and **/** (root) are overwritten. Product (application) files and configuration data stored in these file systems will be lost. Information stored in other non-system file systems will be preserved.

For instructions on preserving the user-defined structure of an existing BOS, refer to Chapter 5, “New and Complete Overwrite Installation / Preservation Installation,” on page 51.

Migration
This method upgrades from AIX 4.3 or earlier versions of the BOS to AIX 5.2. The migration installation method is used to upgrade from an existing version or release of AIX to a later version or release of AIX. A migration installation preserves most file systems, including the root volume group, logical volumes, and system configuration files. It overwrites the **/tmp** file system.

For instructions on migrating an existing version or release of AIX to a later version or release of AIX, refer to Chapter 6, “Migration Installation,” on page 57.

The following table shows the differences in the installation steps among the installation methods.

Table 3. AIX BOS Installation Methods

Installation Steps	New and Complete Overwrite	Preservation	Migration
Create **rootvg**	Yes	No	No
Create file system **/**, **/usr**, **/var**	Yes	Yes	No

© Copyright IBM Corp. 2002, 2004

Table 3. AIX BOS Installation Methods (continued)

Installation Steps	New and Complete Overwrite	Preservation	Migration
Create file system **/home**	Yes	No	No
Save Configuration	No	No	Yes
Restore BOS	Yes	Yes	Yes
Install Additional Filesets	Yes	Yes	Yes
Restore Configuration	No	No	Yes

BOS Menus

After you select the console and language to be used for the BOS menus, the Welcome to Base Operating System Installation and Maintenance screen displays, as follows:

```
                  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


    88  Help ?
    99  Previous Menu

>>> Choice [1]:
```

Note: To turn on the debug mode for the BOS installation process, type 911 in the **Choice** field and press Enter. The Welcome to Base Operating System Installation and Maintenance screen refreshes and the BOS installation process runs in debug mode when the installation occurs. Continue the procedure for selecting options and specifying data until the installation begins. Debug output is sent to the client's display as the installation proceeds.

If you select **Start Install Now with Default Settings**, the BOS command determines the default installation method to use based on your system's configuration. A summary screen displays, similar to the following, where you can confirm the installation method and installation options:

```
                  Overwrite Installation Summary

Disks:  hdisk0
Cultural Convention:  en_US
Language:  en_US
Keyboard:  en_US
64 Bit Kernel Enabled:  No
JFS2 File Systems Created:  No
Desktop:
Enable System Backups to install any system:  Yes

Optional Software being installed:
HTTP_Server (Expansion Pack)
Kerberos_5 (Expansion Pack)

>>> 1   Continue with Install
    88  Help ?
    99  Previous Menu

>>> Choice [1]:
```

If the selections are correct, press Enter to begin the BOS installation.

However, if you would like to change the installation method or options, select **Change/Show Installation Settings and Install** at the BOS welcome screen. The Installation and Settings screen displays, as follows:

```
                 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.............New and Complete Overwrite
         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)
         Keyboard Type......................Default

    3  More Options  (Desktop, Security, Kernel, Software, ...)

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

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

For more information on the BOS menu options, refer to the Help at any time by typing 88 in the **Choice** field. For conceptual information on the BOS installation procedures, refer to the following sections:

- Chapter 5, “New and Complete Overwrite Installation / Preservation Installation,” on page 51
- Chapter 6, “Migration Installation,” on page 57

For how-to instructions for performing BOS installations, refer to the following:

- “Perform a New and Complete Overwrite Base Operating System Installation from CD” on page 3
- “Perform a Migration Base Operating System Installation from CD” on page 7

Electronic License Agreements

AIX 5.2 ships with software license agreements that can be viewed electronically. If a product has an electronic license agreement, it must be accepted before software installation can continue. In the case of initial BOS installation, you can view and accept or reject license agreements in a license agreement dialog after the installation has occurred, but before the system is available for use as part of Configuration Assistant (graphics consoles) or Installation Assistant (ASCII consoles).

The AIX BOS has a license agreement, but not all software packages do. When you agree to the license agreement for BOS installation, you are also accepting all license agreements for any software installed automatically with the BOS. Some software, such as the GNOME or KDE desktops, can be optionally installed during BOS installation; the appropriate licensing information for such software is displayed separately.

If a customized **bosinst.data** file is used (usually for unattended installations, or *nonprompted installations*), the **ACCEPT_LICENSES** field in the **control_flow** stanza can be used to accept the license agreements so users are not prompted at reboot time. When performing a ″push″ installation using the Network Installation Management (NIM) environment, the licenses must be accepted, either from the

choices made when initializing the installation or in a customized **bosinst.data** file, before the installation can continue. For more information about the **bosinst.data** file, refer to Chapter 9, "The bosinst.data File," on page 71.

For additional software package installations, the installation cannot occur unless the appropriate license agreements are accepted. This option, as well as options to preview licenses, is offered in both the System Management Interface Tool (SMIT) and the Web-based System Manager installation interfaces. When using the **installp** command, use the **-Y** flag to accept licenses and the **-E** flag to view license agreement files on the media.

For more information about license manipulation, refer to the **inulag** command description in the *AIX 5L Version 5.2 Commands Reference*.

Chapter 4. Installation Options

This chapter provides information about installation options for the AIX base operating system. The installation options are available by typing 3 in the **More Options** field in the Installation and Settings screen.

On 32-bit systems, the new and complete overwrite installation options are the following:

```
                    Install Options

 1.  Desktop..................................................... NONE, CDE, GNOME, KDE
 2.  Enable Trusted Computing Base.............................. No
 3.  Graphics Software.......................................... Yes
 4.  Documentation Services Software............................ Yes
 5.  Enable System Backups to install any system................. Yes
     (Installs all devices and kernels)

>>> 6.  Install More Software

    0  Install with the current settings listed above.

    88  Help ?
    99  Previous Menu

>>> Choice [6]:
```

On 64-bit systems, the new and complete overwrite installation options are the following:

```
                    Install Options

 1.  Desktop..................................................... NONE, CDE, GNOME, KDE
 2.  Enable Trusted Computing Base.............................. No
 3.  Enable CAPP and EAL4+ Technology........................... No
     (English only, 64-bit kernel enablement, JFS2 file systems)
 4.  Enable 64-bit Kernel....................................... No
 5.  Create JFS2 File Systems................................... No
     (Requires 64-bit Kernel Enabled)
 6.  Graphics Software.......................................... Yes
 7.  Documentation Services Software............................ Yes
 8.  Enable System Backups to install any system................. Yes
     (Installs all devices and kernels)

>>> 9.  Install More Software

    0  Install with the current settings listed above.

    88  Help ?
    99  Previous Menu

>>> Choice [9]:
```

On 64-bit systems, the preservation installation options are the following:

© Copyright IBM Corp. 2002, 2004

```
                    Install Options

 1.  Desktop..................................................... NONE, CDE, GNOME. KDE
 2.  Enable Trusted Computing Base.............................. No
 3.  Import User Volume Groups.................................. Yes
 4.  Graphics Software.......................................... No
 5.  Documentation Services Software............................ No
 6.  Enable System Backups to install any system................. Yes
     (Installs all devices and kernels)

>>> 7.  Install More Software

    0  Install with the current settings listed above.

    88  Help ?
    99  Previous Menu

>>> Choice [9]:
```

On 64-bit systems, the migration installation options are the following:

```
                    Install Options

 1.  Enable Trusted Computing Base.............................. No
 2.  Import User Volume Groups.................................. Yes
 3.  Enable System Backups to install any system................. Yes
     (Installs all devices and kernels)
 4.  Remove Java 1.1.8 Software................................. No

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

    88  Help ?
    99  Previous Menu

>>> Choice [0]:
```

The following are the installation options available:

Desktop

The default is **CDE** for new and complete overwrite installations. If you select **NONE**, a minimal configuration is installed including X11, Java, perl, SMIT, and the Web-based System Manager (if **Graphics Software** is selected).

If you select **GNOME** or **KDE**, the BOS installation process prompts you for the *AIX Toolbox for Linux Applications CD*. If this CD is not available, you can type q to continue the installation without the *AIX Toolbox for Linux Applications CD*.

Enable Trusted Computing Base

The Trusted Computing Base (TCB) is the part of the system that is responsible for enforcing the information security policies of the system. All of the computer's hardware is included in the TCB, but a person administering the system should be concerned primarily with the software components of the TCB.

If you install the Trusted Computing Base option, you enable the trusted path, trusted shell, and system-integrity checking (**tcbck** command). These features can be enabled *only* during BOS installation.

The choices are **yes** and **no**. To enable the Trusted Computing Base, type 2 and press Enter. The default is **no**.

Enable CAPP and EAL4+ Technology

Available in a new and complete overwrite installation. If you enable Controlled Access Protection Profile (CAPP) and Evaluation Assurance Level 4+ (EAL4+), other restrictions exist on installation choices, such as:

- Desktop = CDE or NONE
- TCB = yes
- 64-bit kernel = yes
- JFS2 = yes
- Enable System Backups to install any system (Installs all devices and kernels) = no
- Install more software options = no

For information about CAPP and EAL4+, see Controlled Access Protection Profile and Evaluation Assurance Level 4+ in the *AIX 5L Version 5.2 Security Guide*.

Import User Volume Groups

Available in migration installation and preservation installation. You have the option to have user volume groups imported after the installation completes. These volume groups can be manually imported at a later time.

Enable 64-bit Kernel

Available only on 64-bit Common Hardware Reference Platform (CHRP) systems.
To toggle the choice between **no** (the default) and **yes**, type 3 and press Enter. If you choose **no**, the 64-bit kernel is still installed, but it is not linked to the running **/unix**. If you choose **yes**, the 64-bit kernel is installed and begins running when your system reboots.

Create JFS2 File Systems

Available in new and complete overwrite installation, as well as preservation installation with 64-bit kernel enabled option. Create enhanced journaled file systems during BOS installation.

Graphics Software

Available in new and complete overwrite installation, as well as preservation installation. Install graphics software support.

Documentation Services Software

Available in new and complete overwrite installation, as well as preservation installation. Install the documentation services software.

Remove Java 1.1.8 Software

Available in migration installation and preservation installation. You have the option to have all Java version 1.1.8 software removed.

Enable System Backups

If Enable System Backups to install any system is selected, all devices and kernels are installed, so that a system backup can be installed on a different system. For more information about installing a system backup to a different system, see “Cloning a System Backup” on page 140.

The **Install More Software** option is available in the new and complete overwrite installation method, as well as the preservation installation method. Select **Install More Software** to choose additional software to install after the BOS installation process finishes. A software bundle file corresponds to each selection that contains the required packages and filesets. The following software bundles are available:

```
                    Install More Software

 1. Netscape (Expansion Pack)..................................... No
 2. HTTP_Server (Expansion Pack).................................. Yes
 3. Kerberos_5 (Expansion Pack)................................... Yes
 4. Server  (Volume 2)............................................ No
 5. Alternate Disk Install  (Volume 2)............................ No


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

    88  Help ?
    99  Previous Menu

>>> Choice [0]:
```

Chapter 5. New and Complete Overwrite Installation / Preservation Installation

This chapter provides information on the New and Complete Overwrite Installation operation and the Preservation Installation operation.

Step 1. Complete the Prerequisites

Before starting the installation, complete the following prerequisites:

- There must be adequate disk space and memory available. AIX 5.2 requires 128 MB of memory and 2.2 GB of physical disk space. For additional release information, see the *AIX 5.2 Release Notes*.
- All requisite hardware, including any external devices (such as tape drives, CD-ROM or DVD-ROM drives), must be physically connected. If you need further information, refer to the hardware documentation that accompanied your system.
- The installation media must be loaded in the boot device.
- The system *must be* set to boot from the device in which the installation media is loaded. Refer to the hardware documentation that accompanied your system for instructions on setting the boot device.
- Before you begin the installation, other users who have access to your system must be logged off.
- If the system you are installing is currently running, create or locate a backup of the system. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.
- If your system needs to communicate with other systems and access their resources, make sure you have the information in the following worksheet before proceeding with installation:

Table 4. Network Configuration Information Worksheet

Network Attribute	Value
Network Adapter	
Host Name	
IP Address	_____._____._____._____
Network Mask	_____._____._____._____
Nameserver	_____._____._____._____
Domain Name	
Gateway	_____._____._____._____

Step 2. Prepare Your System for Installation

Prepare for a new and complete overwrite or preservation installation by doing the following:

1. Insert the *AIX Volume 1* CD into the CD-ROM device.
2. Shut down your system. If your machine is currently running, power it off now by following these steps:
   a. Log in as the root user.
   b. Type the following command:
      ```
      shutdown -F
      ```
   c. If your system does not automatically power off, place the power switch in the Off (0) position.
      Attention: You *must not* turn on the system unit until instructed to do so in “Step 4. Boot from Your Installation Media” on page 53.
3. Turn on all attached external devices. These include the following:
   - Terminals

© Copyright IBM Corp. 2002, 2004

- CD-ROM or DVD-ROM drives
- Tape drives
- Monitors
- External disk drives

Turning on the external devices first is necessary so the system unit can identify each peripheral device during the startup (boot) process.

Step 3. Set Up an ASCII Terminal

If you are using a graphics terminal, skip directly to “Step 4. Boot from Your Installation Media” on page 53.

If you are using an ASCII terminal, use the criteria listed below and your terminal reference documentation to set the communications, keyboard, and display options. The following settings are typical, but your terminal might have different option names and settings than those listed here.

Note: If your terminal is an IBM 3151, 3161, or 3164, press the Ctrl+Setup keys to display the Setup Menu and follow the on-screen instructions to set these options.

Communication Options	
Option	**Setting**
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

Keyboard and Display Options	
Option	**Setting**
Screen	normal
Row and Column	24x80
Scroll	jump
Auto LF (line feed)	off
Line Wrap	on
Forcing Insert	line (or both)
Tab	field
Operating Mode	echo
Turnaround Character	CR
Enter	return
Return	new line
New Line	CR
Send	page
Insert Character	space

Step 4. Boot from Your Installation Media

1. Turn the system unit power switch from Off (0) to On (|).
2. When the system beeps twice, press F5 on the keyboard (or 5 on an ASCII terminal). If you have a graphics display, you will see the keyboard icon on the screen when the beeps occur. If you have an ASCII terminal (also called a tty terminal), you will see the word keyboard when the beeps occur.

   Note: If your system does not boot using the F5 key (or the 5 key on an ASCII terminal), refer to your hardware documentation for information about how to boot your system from an AIX product CD.

   The system begins booting from the installation media.
3. If you have more than one console, each might display a screen that directs you to press a key to identify your system console. A different key is specified for each console displaying this screen. If this screen displays, press the specified key *only* on the console you want to use for the installation. (The system console is the keyboard and display device used for installation and system administration.)

   A screen displays, asking you to select a language to be used during installation.
4. Select the language you prefer to use during installation.
5. When the Welcome to Base Operating System Installation and Maintenance screen displays, type 2 in the **Choice** field to select **Change/Show Installation Settings and Install** and press Enter. Go to "Step 5. Verify or Change the Installation Settings" for instructions on verifying or changing installation settings.

Step 5. Verify or Change the Installation Settings

Verify the default installation settings from the Installation and Settings screen. If the installation and system settings are correct, type 0 in the **Choice** field and press Enter. Confirm that the selections on the installation summary screen are correct, and press Enter to begin the BOS installation. Go to "Step 6. Finish the BOS Installation" on page 54.

To change the installation settings, use the following procedure:

1. Select either **New and Complete Overwrite Installation** or **Preservation Installation**.
   a. Type 1 in the **Choice** field to select the **System Settings** option.
   b. When the Change Method of Installation screen displays, type the number corresponding to desired installation (either 1 for New and Complete Overwrite or 2 for Preservation) in the **Choice** field and press Enter.

      Note: Available installation methods depend on whether your system has a previous version of AIX installed.

      If you only want to install the next recommended maintenance level of AIX, see "Installing Optional Software Products and Service Updates" on page 83. You can also use the SMIT **update_all** fast path or the **install_all_updates** command to update existing filesets to the next recommended maintenance level.
   c. When the Change Disk(s) screen displays, you can change the destination disk for the installation. If you selected the preservation installation, the screen lists only disks containing a **rootvg**.

      The disk name, the location, the size of the disk, and the root volume group status is displayed for each available disk. The Bootable column indicates whether the disk is bootable. Type 77 to select **Display More Disk Information** to view additional disk attributes such as physical volume identifier, device adapter connection location, or World Wide Port Name//Lun ID.

      If the default shown is correct, type 0 in the **Choice** field and press Enter. To change the destination disk, use the following procedure:

1) Type the number for each disk you choose in the **Choice** field and press Enter. *Do not* press Enter a final time until you have finished selecting all disks. If you need to deselect a disk, type its number a second time and press Enter.
   - You can specify a supplemental disk by typing `66` and pressing the Enter key for the **Disks not known to Base Operating System Installation** option. This option opens a new menu that prompts for a device support media for the supplemental disk. The device-support media is only needed when the device cannot configure with the generic SCSI or bus-attached device drivers. BOS installation configures the system for the disk and then returns to the Change Disk screen.

2) When you have finished selecting the disks, type `0` in the **Choice** field and press Enter. The Installation and Settings screen displays with the selected disks listed under `System Settings`.

2. Change the primary language environment, if needed. Use the following steps to change the primary language used by this installation.

   Note: Changes to the primary language environment do not take effect until after BOS is installed and your system is rebooted.

   a. Type `2` in the **Choice** field on the Installation and Settings screen to select the **Primary Language Environment Settings** option.
   b. Select the appropriate set of cultural convention, language, and keyboard options. Most of the options are a predefined combination, however, you can define your own combination of options.
      - To select a predefined Primary Language Environment, type that number in the **Choice** field and press Enter.
      - To configure your own primary language environment:
        1) Select **MORE CHOICES**.
        2) Page through the choices and select the **Create Your Own Combination** option.
        3) When the Set Primary Cultural Convention screen displays, type the number in the **Choice** field that corresponds to the cultural convention of your choice and press Enter.
        4) When the Set Primary Language screen displays, type the number in the **Choice** field that corresponds to your choice for the primary language and press Enter.
        5) When the Set Keyboard screen displays, type the number in the **Choice** field that corresponds to the keyboard attached to the system and press Enter.

3. Change the installation options by typing `3` to select **More Options** and press Enter. For more information on the installation options, see Chapter 4, “Installation Options,” on page 47.
4. Verify your selections in the installation summary screen and press Enter to begin the BOS installation process.

Your system automatically reboots after installation is complete. Go to “Step 6. Finish the BOS Installation.”

Step 6. Finish the BOS Installation

1. The Installing Base Operating System screen displays the status of your installation.

   After the base run-time environment is installed, status information displays about other software that is being installed.
2. The system automatically reboots.
3. After the system has restarted, you are prompted to configure your installation. For information on configuring your system after a BOS installation process, refer to Chapter 7, “Configuring the Operating System,” on page 63.

Note: If the system being installed has 4 GB or more of memory and you have performed an overwrite installation, then a dedicated dump device is created for you. If so, the device name is **/dev/lg_dumplv**, and its size is based on the following formula:

```
4>=  RAM < 12       size of dump device= 1 GB
12>= RAM < 24       size of dump device= 2 GB
24>= RAM < 48       size of dump device= 3 GB
     RAM >= 48      size of dump device= 4 GB
```

Related Information

- For additional release information, see the *AIX 5.2 Release Notes*.
- For late-breaking information, which might include information on the configuration process and installed software, refer to readme files. For information on how to view readme files, see “Viewing Readme Files” on page ix.
- For information about installing optional software, refer to “Installing Optional Software Products and Service Updates” on page 83.

Chapter 6. Migration Installation

Migration is the default installation method to move from AIX 4.2 and later to AIX 5.2.

> **Note:** If you only want to install the next recommended maintenance level of the operating system, use the SMIT **update_all** fast path or the **install_all_updates** command to update the filesets currently installed. For more information about updating to the next recommended maintenance level of AIX, see Chapter 10, “Optional Software Products and Service Updates,” on page 81.

During a migration, the installation process determines which optional software products are installed on the existing version of the operating system. Components from previous releases that have been replaced by new software in AIX 5.2 are installed at the AIX 5.2 level.

Migration attempts to preserve all user configuration, while moving the operating system to a new level of software. The following steps are taken to achieve this objective:

- Save configuration files
- Prepare and remove old files
- Restore new files
- Remove unsupported or unnecessary filesets
- Migrate configuration data wherever possible
- Prepare VPD for install
- Update additional filesets

When performing a migration, the following occurs:

- All files in the **/usr/lib/drivers**, **/usr/lib/microcode**, **/usr/lib/methods** and **/dev** directories are removed from the system, so software support for device drivers must be reinstalled. Non-device software products and applications remain on the system, and work correctly if they are among those files described in “Binary Compatibility Between Earlier Versions and AIX 5.2.”
- The following software products are removed from the system:
  - AIXwindows Interface Composer
  - Remote Customer Services
  - AIXwindows Development Environment
  - Display PostScript functionality from AIXwindows Run-Time Environment Extensions
  - Performance Tools functionality from Extended Commands
  - OpenGL and graPHIGS
  - Xstation Manager

In most cases, user-configuration files from the previous version of a product are saved when the new version is installed during a migration installation.

Binary Compatibility Between Earlier Versions and AIX 5.2

After a migration installation, you might notice filesets on the system in the **OBSOLETE** state. Obsolete filesets were installed by earlier versions of the operating system, but they remain on the current system because the migration only replaced some, but not all, of the files they contain. These filesets remain necessary for systems running mixed levels of the operating system.

During a migration installation, the following filesets are automatically included:

- Base operating system commands

© Copyright IBM Corp. 2002, 2004

- Base operating system libraries
- Base operating system curses/termcap
- Base operating system networking
- Base operating system directories/files (symlinks)
- Messages
- X11R3
- X11R4
- X11 fonts

All operating system applications based on AIX 4.2 and those intended for use with POWER family, POWER2, and POWER-based models run compatibly on AIX 5.2 without recompilation for those same models. The only exceptions to this statement are applications using the following:

- Unsupported self-loadable kernel extensions
- Certain High-Function Terminal (HFT) control interfaces
- X11R3 input device interfaces
- CIO LAN device driver interface
- SCSI device configuration methods (IHVs)
- nlist subroutine interface
- DCE threads
- Applications compiled using POWER2 or POWER-based compiler options, but executed on models other than POWER2 or POWER-based models.

  Note: Any program that must run in all environments (POWER family, POWER2, and POWER-based models 601 and higher) must be compiled using the common mode of the compiler. Programs compiled to exploit POWER2 technology must be run on POWER2 processors. Existing code need not be recompiled to run.

A system using AIX 4.2 can operate as a server system for client machines using AIX 5.2 with the following exceptions:

- Network installation of AIX 5.2 clients
- Service SNA or X.25 to AIX 5.2 clients
- Service HCON to AIX 5.2 clients
- Service CGE extensions of PEX and PEX-PHIGS
- Use of AIX 5.2 client installation formats

Font servers might be required on the AIX 5.2 clients to reliably handle AIXwindows between server and client.

A system using AIX 5.2 might operate as a server system for client machines using AIX 4.2 or later versions as long as the necessary compatibility options are installed. All statements about binary compatibility apply in this case. AIX 5.2 applications might not execute reliably on AIX 4.2 systems using remote network mounts from an AIX 5.2 file system.

Pre-Migration and Post-Migration Checking

The **pre_migration** and **post_migration** commands perform various system checks to ensure a successful migration installation. Both commands are shipped in the **bos.rte** fileset.

In case the **pre_migration** command does not exist on a level of AIX that you want to check before performing a migration installation, the **pre_migration** command is also located in the **/usr/lpp/bos**

directory of the CD file system. Copy the **pre_migration** command from the **/usr/lpp/bos** directory of the new AIX CD version you are about to perform the migration.

The output from the **pre_migration** command is saved to the system in the **/home/pre_migration***date* directory.

The **pre_migration** command performs the following actions:

- List the device filesets being removed.
- List all other filesets being removed.
- List the saved base configuration files that will not be merged.
- List configuration files that will be merged.
- Verify fileset version consistency.
- Create a list of all filesets installed, to be used by the **post_migration** command.
- Check the size and location of the boot logical volume.
- Check the major number for rootvg is 10.
- Check for the missing DB directory for the **bos.net.ipsec.keymgt** fileset.
- Determine if Kerberos is being used.
- Check disk and memory sizes.
- If migrating from AIX 4.2, verify that the correct updates are applied.
- Verify system platform.
- Print a recommendation that a system backup be made before the migration.

The output from the **post_migration** command is saved in the **/home/post_migration***date* directory.

The **post_migration** command performs the following actions:

- Verify fileset version consistency.
- Check the installation list from before the migration, and inform the user of any filesets that might still need migrating.
- Compare saved and merged configuration scripts and save the differences.

Migrating to AIX 5.2

Notes:

1. The boot logical volume requires 12 MB of contiguous disk space. During migrations, the **inuextendblv** command runs to ensure there are contiguous partitions for `hd5`. If contiguous partitions are not present, the **inuextendblv** command attempts to create them. If the partitions are not present and the **inuextendblv** command fails to create them, the migration is stopped.
2. The settings in your bootlist are not migrated. After a migration, the bootlist is set to the primary boot device.

Use the following steps to migrate your current version of the operating system to the AIX 5.2 BOS:

- “Step 1. Complete the Prerequisites” on page 60
- “Step 2. Prepare Your System for Installation” on page 60
- “Step 3. Boot from Your Installation Media” on page 61
- “Step 4. Finish the BOS Migration” on page 62

Attention: This procedure requires shutting down and reinstalling the base operating system. Whenever you reinstall any operating system, schedule your downtime when it least impacts your workload to protect yourself from a possible loss of data or functionality. Before you perform a migration installation, ensure that you have reliable backups of your data and any customized applications or volume groups. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

Step 1. Complete the Prerequisites

Before starting the migration, complete the following prerequisites:

- All requisite hardware, including any external devices (such as tape, CD, or DVD-ROM drives), must be physically connected. If you need further information, refer to the hardware documentation that accompanied your system.
- Before migrating your BOS to AIX 5.2, ensure that the root user has a primary authentication method of **SYSTEM**. You can check this condition by typing the following command:

  ```
  # lsuser -a auth1 root
  ```

  Change the value, if needed, by typing the following command:

  ```
  # chuser auth1=SYSTEM root
  ```
- Before you begin the installation, other users who have access to your system must be logged off.
- Verify that your applications runs on AIX 5.2. Also, verify that your applications are binary-compatible with AIX 5.2. If your system is an application server, verify that there are no licensing issues. Refer to your application documentation or provider to verify on which levels of AIX your applications are supported and licensed. You can also check the *AIX application availability guide* at the following Web address: http://www-1.ibm.com/servers/aix/products/ibmsw/list/
- Verify that all currently installed software is correctly entered in the Software Vital Product Database (SWVPD), by using the **lppchk** command. To verify that all filesets have all required requisites and are completely installed, type the following:

  ```
  # lppchk -v
  ```
- Verify that your hardware microcode is up-to-date.
- All requisite hardware, including any external devices (such as tape, CD, or DVD-ROM drives), must be physically connected and powered on. If you need further information, refer to the hardware documentation that accompanied your system.
- Use the **errpt** command to generate an error report from entries in the system error log. To display a complete detailed report, type the following:

  ```
  # errpt -a
  ```
- There must be adequate disk space and memory available. AIX 5.2 requires 128 MB of memory and 2.2 GB of physical disk space. For additional release information, see the *AIX 5.2 Release Notes*.
- Make a backup copy of your system software and data. For instructions on how to create a system backup, refer to “Creating System Backups” on page 127.

Step 2. Prepare Your System for Installation

Prepare for migrating to the AIX 5.2 BOS by doing the following:

1. Insert the *AIX Volume 1* CD into the CD-ROM device.
2. Shut down your system. If your machine is currently running, power it off now by following these steps:
   a. Log in as the root user.
   b. Type the following command:

      ```
      # shutdown -F
      ```
   c. If your system does not automatically power off, place the power switch in the Off (`0`) position.

      Attention: You *must not* turn on the system unit until instructed to do so in “Step 3. Boot from Your Installation Media” on page 61.

3. Turn on all attached external devices. These include the following:
   - Terminals
   - CD or DVD-ROM drives
   - Tape drives
   - Monitors
   - External disk drives

   Turning on the external devices first is necessary so the system unit can identify each peripheral device during the startup (boot) process.

Step 3. Boot from Your Installation Media

If you are using an ASCII console that was not defined in your previous system, complete “Step 3. Set Up an ASCII Terminal” on page 52 before proceeding.

The following steps migrate your current version of the operating system to AIX 5.2:

1. Turn the system unit power switch from Off (`0`) to On (`|`).
2. When the system beeps twice, press F5 on the keyboard (or `5` on an ASCII terminal). If you have a graphics display, you see the keyboard icon on the screen when the beeps occur. If you have an ASCII terminal (also called a tty terminal), you see the word `keyboard` when the beeps occur.

   Note: If your system does not boot using the F5 key (or the `5` key on an ASCII terminal), refer to your hardware documentation for information about how to boot your system from an AIX product CD.

   The system begins booting from the installation media.
3. If your system has an LED display, the three-digit LED should display `c31`.

   If you have more than one console, each might display a screen that directs you to press a key to identify your system console. A different key is specified for each console displaying this screen. If this screen displays, press the specified key *only* on the device to be used as the system console. (The system console is the keyboard and display device used for installation and system administration.) Press a key on one console *only*.

   A screen displays, asking you to select a language to be used for installation instructions.
4. Select the language you prefer to use for installation instructions.
5. When the Welcome to Base Operating System Installation and Maintenance screen displays, either begin the migration immediately by typing `1` to select **Start Install Now with Default Settings**, or verify the installation and system settings by typing `2` to select **Change/Show Installation Settings and Install**. If you want to change any settings, follow the procedure in “Step 5. Verify or Change the Installation Settings” on page 53.

   Notes:
   - You should not have to change settings simply to select the migration installation method. If a previous version of the operating system exists, the installation method defaults to migration.
   - The available installation methods vary, depending on the version of the operating system that is currently installed (before migration). For information about the BOS installation methods, refer to Chapter 3, “Introduction to Base Operating System Installation,” on page 43. For information about the installation options available for a migration installations, refer to Chapter 4, “Installation Options,” on page 47.
6. Verify the selections in the Migration Installation Summary screen and press Enter.
7. Confirm the migration installation process in the Migration Confirmation screen, and press Enter to begin the migration installation.

Step 4. Finish the BOS Migration

After prompting for confirmation, the installation process begins. The Installing Base Operating System screen displays.

As the installation progresses, the numbers increment in the fields that show percentage complete and elapsed time to indicate the installation status. After the base run-time environment is installed, status information displays about other software that is being installed. After the BOS installation is complete, the system automatically reboots.

After the system has restarted, you are prompted to configure your installation of the BOS. Go to Chapter 7, "Configuring the Operating System," on page 63 for information on the configuration process.

> **Note:** If there is not enough space to migrate all of the usually migrated software, a collection of software called a Migration Bundle is available when you install additional software later. You must create additional disk space on the machine on which you want to install, and then you can run **smit update_all** to complete the installation, during which the Migration Bundle is installed.
>
> If you are not doing the installation from a graphics console, a Graphics_Startup bundle is created. Refer to "Installing Optional Software Products and Service Updates" on page 83 for more information about installing software bundles and for information on migrating or installing optional software products. "Maintaining Optional Software Products and Service Updates" on page 87 describes how to remove software from the system to release disk space.

Related Information

- For additional release information, see the *AIX 5.2 Release Notes*.
- For late-breaking information, which might include information on the configuration process and installed software, refer to readme files. For information on how to view readme files, see "Viewing Readme Files" on page ix.
- For information about installing optional software, refer to "Installing Optional Software Products and Service Updates" on page 83.

Chapter 7. Configuring the Operating System

This chapter provides information concerning the tasks you might need to perform after installing the base operating system (BOS). Complete all configuration tasks that apply to your newly installed system. Two configuration tools are available to assist you. Depending on which type of console you are using, one of the following usually begins automatically after installation:

- "Configuration Assistant" for graphics consoles
- "Installation Assistant" on page 64 for ASCII consoles

Notes:

- If your system was installed by a network installation server, the Configuration Assistant or Installation Assistant does not display when the BOS installation program completes.

  If your system was installed using a system backup image, or if your BOS installation was customized, or if you selected migration installation from AIX 4.2 or later, the Configuration Assistant or Installation Assistant might not display when the BOS installation program completes.
- The Configuration Assistant and the Installation Assistant do not contain the tasks needed to configure your machine as a server. If you need to configure your system for a specific resource, refer to the documentation pertaining to that resource.
- If your terminal type is not set, the first menu displayed by the ASCII Installation Assistant requires you to enter your terminal type (tty). If you enter a terminal type that is not valid, this menu redisplays until a valid type is entered.

  If you enter a valid terminal type that does not match your terminal, the next screen displayed might be unreadable. In this case, press the break key sequence to return to the Set Terminal Type screen. For most terminal types, the break key sequence is `Ctrl-C`.

Configuration Assistant

On a system with a graphical interface, the newly installed BOS reboots and the Configuration Assistant starts to guide you through the configuration tasks. For example, much of the processing on a system uses the date and time-of-day information, requiring the system have the date and time set correctly.

If there are outstanding software license agreements that must be accepted before you can continue to use the machine, the Configuration Assistant prompts you to view and accept these agreements.

The Configuration Assistant guides you through the following configuration tasks:

- Set the system date and time for your time zone.
- Set a root user account password to restrict access to system resources.
- Configure network communications.

  Note: To configure your machine as an NFS server, refer to NFS Installation and Configuration in the *AIX 5L Version 5.2 System Management Guide: Communications and Networks*.
- Configure Web-based System Manager to run in a Web browser (applet mode).
- Configure Online Documentation Library Service, which lets users conduct searches of online documentation. (It is highly recommended that you complete this configuration task, because some applications depend on the Online Documentation Library Service to conduct searches of their online manuals and helps.) Also, this configuration task optionally helps install a Web browser, Web server, and the associated documentation.
- Manage Software.
- Exit the Configuration Assistant.

© Copyright IBM Corp. 2002, 2004

The Manage Software option allows you to perform software management tasks immediately after a BOS installation. The following options are available:

- List installed software
- Install additional software
- List software licenses with license text

If you select **List installed software**, the following options are available:

- List automatically installed Software – Displays a list of all installed packages
- List optionally installed software – Displays a list of all optional software that was selected to be installed during BOS installation

If you select **Install additional software**, the following options are available:

- Install by bundles – Allows you to select from a list of software bundles to install additional software, such as the Netscape Communicator Software Bundle or a User-Defined Software Bundle
- Selective install – Allows you to select a specific package or set of packages to install

The graphical interface for the Configuration Assistant provides step-by-step instructions for completing each configuration task. The tasks are presented to you in a logical sequence. Complete all configuration tasks before you use your system.

When you exit the Configuration Assistant, the guide asks you whether you want to start Configuration Assistant again the next time you restart the operating system. After exiting the Configuration Assistant, users can begin logging in to and using AIX.

To access the Configuration Assistant later, type `configassist` on the command line.

Installation Assistant

On a system with an ASCII interface, the newly installed BOS reboots and starts the Installation Assistant to guide you through completing configuration tasks. You must have root user authority to use the Installation Assistant. To access the Installation Assistant later, type `install_assist` on the command line. You can also access it from a graphics system through the SMIT **smit assist** fast path.

If there are outstanding software license agreements that must be accepted before you can continue to use the machine, the Installation Assistant prompts you to view and accept these agreements.

The Installation Assistant guides you through the following configuration tasks:

- Set the system date and time for your time zone.
- Set a root user account password to restrict access to system resources.
- Configure network communications.
- Install software applications.
- Using SMIT (information only).
- Tasks Completed - Exit to Login.

The Install software applications option allows you to perform software management tasks immediately after a BOS installation. The following options are available:

- Add License Passwords for Applications
- Install and Update Software

If you select **Install and Update Software**, the following menu displays:

```
                        Install and Update Software

Move cursor to desired item and press Enter.

  Install Software
  Update Installed Software to Latest Level (Update All)
  Install Software Bundle
  Update Software by Fix (APAR)
  Install and Update from ALL Available Software
```

You can also access this SMIT menu by using the **install_update** fast path.

Related Information

For late-breaking information, which might include information about the configuration process and installed software, refer to readme files. For information about viewing how to view readme files, see “Viewing Readme Files” on page ix.

If you are installing from CD/DVD-ROM, or would like more information about installing optional software, refer to “Installing Optional Software Products and Service Updates” on page 83.

Chapter 8. Customized BOS Installations

This chapter describes how to customize subsequent installations of the BOS after the operating system has been installed. Customizing an installation requires you to edit the **bosinst.data** file and use it with your installation media.

Introduction to Customized BOS Installations

The first time you install, the Base Operating System (BOS) installation program presents menus from which you must choose setup options. This initial installation also automatically starts a post-installation configuration program, either the graphical Configuration Assistant or the ASCII Installation Assistant. For more information about Configuration Assistant and Installation Assistant, refer to Chapter 7, “Configuring the Operating System,” on page 63.

For subsequent installations, you can change many aspects of the default BOS install program by editing the **bosinst.data** file. For example, to install the BOS without menus, you can specify that no prompts be provided. You can also customize a BOS installation to bypass Configuration Assistant or Installation Assistant and start your own configuration script. Also, the **bosinst.data** file can be used to replicate one set of installation settings on other machines. For example, system administrators can create a **bosinst.data** file with settings that can be used to install all the machines they support that have the same configuration. For AIX 4.3.3 and later versions of the operating system, you can use the Web-based System Manager Reinstall Base Operating System wizard to install systems from product or backup media. This application lets you customize your installation by answering prompts before installation and creates a **bosinst.data** file appropriate for the type of installation wanted. For more information on the **bosinst.data** file, refer to Chapter 9, “The bosinst.data File,” on page 71.

If you run your own configuration script from a **bosinst.data** file or from the Network Installation Management (NIM) interface, the environment that is in place at the time the script is run is a *single-user environment*. This environment is not available as a multiuser environment, and thus, there are limits to what can be run from a configuration script. The **/etc/init** file is not running, so no process management can take place. All available memory cannot be made available because the RAM file system still exists, so devices that require large amounts of memory to run might fail to configure. In addition, signal handling is not available.

Because of the single-user environment, use the following guidelines for configuration scripts:

- Base devices can be configured, but devices that require daemons or more complex configuration should be started at reboot time by adding the necessary code to the end of the **/etc/firstboot** script.
- Daemons should not be started.
- Items such as NIS configuration, which uses system resource controller (SRC) commands, should be done by creating a separate entry in the **/etc/inittab** file and running a configuration script at reboot time.
- Beginning in AIX 5.2, the BOS installation process automatically creates and extends paging space based on available memory.

The **bosinst.data** file directs the actions of the BOS installation program. The file resides in the **/var/adm/ras** directory on the installed machine only, and it is not accessible on the commercial tape or the CD on which you received AIX 5.2.

The **bosinst.data** file contains stanzas with variables set to default values. Each variable is on a new line, in the *Variable*=*Value* form. A blank line separates each stanza. These stanzas provide the installation program with information such as the method and type of installation, the disks in the machine, and the language used. By editing the file with an ASCII text editor, you can substitute new values for the default variables.

© Copyright IBM Corp. 2002, 2004

Another installation file, **image.data**, can also be modified and used during BOS installation. The **image.data** file contains information describing the root volume group image created during the BOS installation process. This information includes the sizes, names, maps, and mount points of logical volumes and file systems in the root volume group. The installation program also takes input from the **image.data** file regarding defaults for the machine being installed. For a description of the **image.data** file, see *AIX 5L Version 5.2 Files Reference*. The procedure for using the **bosinst.data** file to customize BOS installation can also be used for the **image.data** file. The modified files can be used together to override BOS installation defaults.

You can also use the instructions in this chapter to create a supplemental diskette, a CD-R, or a DVD-RAM containing a modified **preserve.list** file, which is used during a preservation Installation. For more information about a preservation installation, see Chapter 3, “Introduction to Base Operating System Installation,” on page 43.

Customizing and Using a bosinst.data File

You must install the BOS before you can access and modify the default **bosinst.data** file. The Web-based System Manager **Reinstall Base Operating System** wizard can be used to prepare your next installation and in customizing the **bosinst.data** file. This file may also be retrieved and edited like any other ASCII file. If you are editing the **bosinst.data** file, use one of the following procedures:

- “Create and Use a Client File”
- “Create and Use a Supplementary Diskette”
- “Create and Use a Supplementary CD” on page 69

For information about the contents of the file and examples of edited files, refer to “bosinst.data File Stanza Descriptions” on page 71 and “bosinst.data File Example” on page 78. To verify the contents of your modified **bosinst.data** file, use the **bicheck** command.

Note: If you are customizing the **/bosinst.data** file so that it becomes part of a system backup (mksysb), beginning with AIX 4.3.3, the **mksysb** command always updates the **target_disk_data** stanzas to reflect the current disks in the **rootvg**. If you do not want this update to occur, you must create the file **/save_bosinst.data_file**. The existence of this file is checked by the **mksysb** command, before the **target_disk_data** stanzas are updated.

Create and Use a Client File

Create one customized **bosinst.data** file for each client, and using Network Installation Management (NIM), define the files as NIM resources. For more information about how to use the **bosinst.data** file as a resource in network installations, refer to Chapter 9, “The bosinst.data File,” on page 71.

Create and Use a Supplementary Diskette

To create the supplementary diskette and use it in future installations complete the following process.

1. Customize the **bosinst.data** file and create a signature file by completing the following steps:
   a. Use the **mkdir** command to create a directory called **/tmp/mydiskette**.
      ```
      mkdir /tmp/mydiskette
      ```
   b. Use the **cd** command to change your directory to the **/tmp/mydiskette** directory.
      ```
      cd /tmp/mydiskette
      ```
   c. Copy the **/var/adm/ras/bosinst.data** file to **/tmp/mydiskette**.
   d. Edit the **bosinst.data** file with an ASCII editor to customize it.
   e. Create a signature file:
      ```
      echo data > signature
      ```
2. Create the diskette and use it for installation by completing the following steps:

a. Back up the edited **bosinst.data** file and the new **signature** file to diskette with the following command:

```
ls ./bosinst.data ./signature | backup -iqv
```

OR

If you create a bundle file named `mybundle`, back up the edited **bosinst.data** file, the new **signature** file, and the bundle file to diskette with the following command:

```
ls ./bosinst.data ./signature ./mybundle | backup -iqv
```

b. Insert the diskette in the diskette drive of the target machine you are installing.

c. Boot the target machine from the installation media (tape, CD/DVD-ROM, or network) and install the operating system.

The BOS installation program uses the diskette file, rather than the default **bosinst.data** file shipped with the installation media. For more information on the **bosinst.data** file, see Chapter 9, “The bosinst.data File,” on page 71.

Create and Use a Supplementary CD

Use this procedure to create the supplementary CD and use it in future installations complete the following process.

1. Customize the **bosinst.data** file and create a signature file by completing the following steps:
   a. Use the **mkdir** command to create a directory called **/tmp/mycd**.

   ```
   mkdir /tmp/mycd
   ```

   b. Use the **cd** command to change your directory to the **/tmp/mycd** directory.

   ```
   cd /tmp/mycd
   ```

   c. Copy the **/var/adm/ras/bosinst.data** file to **/tmp/mycd**.

   d. Edit the **bosinst.data** file with an ASCII editor to customize it.

   e. Create a signature file:

   ```
   echo data > signature
   ```

   f. Change the permissions on the file using the following command:

   ```
   # chmod 777 *
   ```

2. Create the customized CD by completing the following steps:
   a. Use the **cd** command to change your directory to the **/** directory.

   b. Create the customized CD using the following command (where **/dev/cd1** varies depending on your CD writer device):

   ```
   # mkcd -d /dev/cd1 -r /tmp/mycd
   ```

3. Use the customized CD for installation by completing the following steps:
   - If you have only one CD-ROM drive and you are installing from CD, complete the following:
     a. Insert the installation CD in the CD-ROM drive of the machine where you are installing AIX.
     b. Boot the machine from the installation CD.
     c. Type `311` at the BOS welcome screen. You will be prompted to insert the customized CD.
     d. Insert the customized CD.

     The BOS installation program uses the **bosinst.data** file on the CD, rather than the **bosinst.data** file on the boot media. For more information on the **bosinst.data** file, see Chapter 9, “The bosinst.data File,” on page 71.
   - If you are performing a network installation or tape **mksysb** installation, or if you have more than one CD-ROM drive, complete the following:
     a. Insert the customized CD in the CD-ROM drive of the machine where you are installing AIX.
     b. Boot the machine from the network or a tape.
     c. Type `311` at the BOS welcome screen. The installation continues for a non-prompted installation, or the menus display for a prompted installation.

The BOS installation program uses the **bosinst.data** file on the CD, rather than the **bosinst.data** file from the boot media. For more information on the **bosinst.data** file, refer to Chapter 9, “The bosinst.data File,” on page 71.

Related Information

For more information about the **bosinst.data** file, refer to Chapter 9, “The bosinst.data File,” on page 71.

For a description of the **image.data** file, see *AIX 5L Version 5.2 Files Reference*.

Chapter 9. The bosinst.data File

This chapter provides information on the **bosinst.data** file. The chapter includes the following sections:

- "bosinst.data File Stanza Descriptions"
- "bosinst.data File Example" on page 78

bosinst.data File Stanza Descriptions

This section describes the contents of the **bosinst.data** file. Example files follow the stanza descriptions. (See "bosinst.data File Example" on page 78.)

control_flow Stanza

The control_flow stanza contains variables that control the way the installation program works:

Variable	Description
CONSOLE	Specifies the full path name of the device you want to use as the console. If this value is **Default**, and you are performing a nonprompted installation, then the console is set to **/dev/lft0**, if this device exists. If **/dev/lft0** does not exist, the console is set to **/dev/tty0**. (Instructions for which key to press are displayed on the screen.) If you change the **PROMPT** variable to **no**, you must specify a console here.
INSTALL_METHOD	Specifies a method of installation: **migrate**, **preserve**, or **new and complete overwrite**. The default value is initially blank. The installation program assigns a value, depending on which version of AIX was previously installed. See Chapter 3, "Introduction to Base Operating System Installation," on page 43 for more information. The default method of installation is **migrate** if a previous version of the operating system is on the machine. If no previous version exists, the default method is **new and complete overwrite**.
PROMPT	Specifies whether the installation program uses menus from which you make choices. The possible values are **yes** (default) and **no**. **Note:** You must fill in values for all variables in the locale stanza to uniquely identify the disk, if you set the **PROMPT** variable to **no**. Similarly, if **PROMPT** equals **no**, you must supply values for variables in the control_flow stanza, with two exceptions: the **ERROR_EXIT** and **CUSTOMIZATION_FILE** variables, which are optional. **Attention:** Fill in values for enough variables in the target_disk_data stanza if you set the **PROMPT** variable to **no**. The BOS installation program assigns target disks for blank variables. You can lose data if the installation program assigns a disk where you store data. For more information on disk selection during nonprompted installations, see 72.

© Copyright IBM Corp. 2002, 2004

EXISTING_SYSTEM_OVERWRITE

Confirms that the installation program can *overwrite* existing volume groups. This variable is applicable only for a nonprompted overwrite installation. The possible values are **no** (default), **yes**, and **any**.

no (Default) Only disks that are not part of a volume group can be used for the installation.

yes Disks that contain the root volume group is used first, and if additional disks are needed for the installation, then disks that contain no volume groups are used.

any Any disks can be used for the installation.

When the installation is nonprompted and the target_disk_data stanza is empty, the installation process uses the value of the **EXISTING_SYSTEM_OVERWRITE** field to determine the disks to install on.

When you do a prompted installation, this value is changed to `yes`, and is saved with other changes in the **/var/adm/ras/bosinst.data** file. Network Install Manager (NIM) creates a default **bosinst.data** file (NIM **bosinst_data** resource) with this value set to `yes`, and system backups use the **bosinst.data** file that is copied from the **/var/adm/ras** directory, so in most cases this value is already be set to `yes`. If this field is set to `no`, as seen in the **/usr/lpp/bosinst/bosinst.template** file, an error message informs you that there are not enough disks matching the criteria needed to complete the installation during a nonprompted install. The BOS installation is then changed to a prompted BOS installation, and the value of the **EXISTING_SYSTEM_OVERWRITE** field is set to `yes`.

INSTALL_X_IF_ADAPTER

Specifies whether a desktop should be installed or not. The possible values are **yes**, **all**, and **no**. The default value for this field is **yes**, meaning that if the system has a graphical console, and a **DESKTOP** is specified, the **DESKTOP** is installed. If set to **all** and a **DESKTOP** is specified, the desktop is installed, whether the system has a graphical console or not. If set to **no** and a **DESKTOP** is specified, the desktop is not installed.

RUN_STARTUP

Starts the Configuration Assistant on first boot after the BOS installation completes, if the system has a graphical interface. Starts Installation Assistant if the machine has an ASCII interface. The possible values are **yes** (default) and **no**. The **no** value is valid only when the **ACCEPT_LICENSES** field is set to **yes**.

RM_INST_ROOTS

Removes all files and directories in the **/usr/lpp/*/inst_roots** directories. The possible values are **no** (default) and **yes**.

The **/usr/lpp/bos/inst_roots** directories must remain if the machine is used as a network server. To save disk space, set this value to **yes** if the machine is not a network server.

ERROR_EXIT

Starts an executable program if an error occurs in the installation program. The default value is blank, which signals BOS installation to use a command that is shipped on the installation media. The command starts an error message routine when the installation program halts because of an error. As an alternative to the default, you can enter the path name of your own script or command for a customized error routine.

CUSTOMIZATION_FILE

Specifies the path name of a customization file you create. The default value is blank. The customization file is a script that starts immediately after the installation program concludes.

TCB

Specifies whether you want to install the Trusted Computing Base (TCB). When you install the TCB, the trusted path, the trusted shell, and system integrity checking are installed. The TCB must be installed and initialized when the operating system is installed. The TCB cannot be installed later. By not installing the TCB, installation time is reduced. The possible values are **no** (default) and **yes**.

INSTALL_TYPE

If set to **CC_EVAL**, then CAPP and EAL4+ technology will be enabled. This is only allowed when **INSTALL_METHOD** is overwrite. If this is set, the **CULTURAL_CONVENTION** and **MESSAGES** fields of the locale stanza can only be **en_US** or **C**. The system must be 64-bit, and have **ENABLE_64BIT_KERNEL** and **CREATE_JFS2_FS** set to **yes**. **ALL_DEVICES_KERNELS** must be set to **no**. **TCB** must be set to **yes**. **DESKTOP** can only be **NONE** or **CDE**. The additional software bundles: **NETSCAPE_BUNDLE**, **HTTP_SERVER_BUNDLE**, **KERBEROS_5_BUNDLE**, **SERVER_BUNDLE** and **ALT_DISK_INSTALL_BUNDLE**, must be set to **no**.

BUNDLES

Specifies what software bundles to install. Type the full path name of each bundle file. Be sure there is sufficient disk space and paging space on the target machine for the software you specify in the **BUNDLES** variable.

This list of bundle file names is limited to 139 bytes. If your list of bundle file names is longer than 139 bytes, use the **cat** command to combine the bundle files into a single custom bundle file and enter the name of your custom bundle file in this field.

If you are installing from tape, to specify system-defined bundles on the product media, use the full path name of each bundle file as follows:

```
/usr/sys/inst.data/sys_bundles/BundleFileName
```

If you are using a **bosinst.data** diskette to define your own bundle files, specify the full path name of each bundle file as follows: `/../`*DirectoryName*`/`*BundleFileName*. For example, if you put a bundle file named `mybundle` in the **root** directory, the full path name would be `/../mybundle`.

If you are using preservation installation, create bundle files before you start the installation. Create the files in **/home** and specify the full path name of each bundle file as follows:

```
/home/BundleFileName
```

RECOVER_DEVICES

Specifies whether to reconfigure the devices. The default value is **Default**. For **mksysb** installations, the ODM configuration database is saved in the image. The device names and attributes are automatically extracted from the database, and the BOS installation program attempts to recreate the devices the same way they were on the machine the **mksysb** was created on. This is normal procedure for regular **mksysb** restores on the same system. However, for cloning (installing the mksysb image on another system), you may not want these devices configured this way, especially for network configuration.

When the **mksysb** image is created, the CPU ID is saved. If you are reinstalling the same system, then the device information is recovered. If the **mksysb** image is used to install another system, device information is *not* recovered from the **mksysb** image.

The **Default** value can be overwritten. For example, if your system had the planar replaced, or you upgraded to another system, you might want to recover devices. In these cases, you can select **yes** in the Backup Restore menu to recover devices.

BOSINST_DEBUG

Specifies whether to show debug output during BOS installation. The value **yes** sends **set -x** debug output to the screen during BOS installation. The possible values are **no** (default) and **yes**.

ACCEPT_LICENSES

Specifies whether to accept software license agreements during the BOS installation. The default is **no**. To automatically accept them, set this value to **yes**. When the software licenses agreements are not accepted during BOS installation, Configuration Assistant or Installation Assistant prompts you to view and accept them. During a BOS installation, if this value is blank, the default of **no** is assumed.

For **mksysb** installations, when the **ACCEPT_LICENSES** field is **no**, the user is forced to accept the licenses again before continuing to use the system. When the **ACCEPT_LICENSES** field is set to **yes**, the licenses are automatically accepted for the user. If blank, the state of the licenses is the same as when the **mksysb** was created.

DESKTOP

Specifies the desktop to be installed. The choice of available desktops are CDE (the default), NONE, GNOME, and KDE. If you choose GNOME or KDE, you will be prompted for the *AIX Toolbox for Linux Applications* CD.

INSTALL_DEVICES_AND_UPDATES

When installing a **mksysb** image to a system with a different hardware configuration, boot from product media to get any missing device drivers installed. In addition, if the product media is a later level of AIX than the **mksysb**, software in the **mksysb** image will be updated. To prevent either of these additional installations from occuring, set this field to **no**. The default is **yes**.

IMPORT_USER_VGS

Specifies whether you want any user volume groups to be automatically imported after the system has been installed. The choices are **yes** and **no**.

ENABLE_64BIT_KERNEL

Specifies whether you want to enable the 64-bit kernel. The choices are **yes** and **no**.

CREATE_JFS2_FS

Specifies whether you want to create enhanced journaled file systems. The choices are **yes** and **no**.

ALL_DEVICES_KERNELS	Specifies whether to install all device and kernel filesets. The choices are **yes** and **no**. If you select **no**, your system will be installed with the devices and kernel specific to your system configuration. If you select **yes**, when you create a system backup of your system, you can use that system backup to install any system.
GRAPHICS_BUNDLE	Specifies whether to install the graphics software bundle during the BOS installation. This software bundle contains the graphics support for the Web-based System Manager and Linux desktops. The choices are **yes** and **no**.
DOC_SERVICES_BUNDLE	Specifies whether to install the documentation services software bundle during the BOS installation. This software bundle contains the Documentation Library Service software, which is used for viewing and searching online documentation. The choices are **yes** and **no**.
NETSCAPE_BUNDLE	Specifies whether to install the Netscape Communicator software bundle during the BOS installation. This software bundle contains the Netscape Communicator software. The choices are **yes** and **no**.
HTTP_SERVER_BUNDLE	Specifies whether to install the HTTP Web server software bundle during the BOS installation. This software bundle installs the HTTP Web server software. The choices are **yes** and **no**.
KERBEROS_5_BUNDLE	Specifies whether to install the Kerberos 5 client software bundle during the BOS installation. This software bundle installs the Kerberos 5 client software. The choices are **yes** and **no**.
SERVER_BUNDLE	Specifies whether to install the AIX server software bundle during the BOS installation. This software bundle installs additional networking software, performance tools, and accounting services software. The choices are **yes** and **no**.
ALT_DISK_INSTALL_BUNDLE	Specifies whether to install the alternate disk installation software during the BOS installation. The choices are **yes** and **no**.
REMOVE_JAVA_118	Specifies whether to remove the Java 1.1.8 software from the current system when performing a migration installation. The choices are **yes** and **no**.
HARDWARE_DUMP	Creates a dump logical volume to contain firmware and hardware dump data. Dump logical volumes are only create on hardware that supports creation of firmware and hardware dump data. The choices are **yes** and **no**.

target_disk_data Stanza

The target_disk_data stanza contains variables for disks in the machine where the program is to install BOS. The default **bosinst.data** file has one target_disk_data stanza, but you can add new stanzas to install BOS on multiple disks, one stanza for each disk.

Multiple target_disk_data stanzas can exist. They define the disks that are to contain the root volume group. Only one field (**PVID**, **PHYSICAL_LOCATION**, **SAN_DISKID**, **CONNECTION**, **LOCATION**, **SIZE_MB**, **HDISKNAME**) must be non-null for BOS installation to choose a disk. The order of precedence is **PVID** (Physical Volume ID), **PHYSICAL_LOCATION**, **SAN_DISKID**, then **CONNECTION** (parent attribute//connwhere attribute), then **LOCATION**, then **SIZE_MB**, and then **HDISKNAME**. The BOS installation process uses the following logic to determine how to use the target_disk_data stanza information:

- If **PVID** is set, BOS installation checks to see if a disk matches the value. If so, other attributes are ignored.
- If **PVID** is empty and **CONNECTION** is set, then BOS installation checks to see if the parent and connwhere attributes (separated by ″//″) match a disk. If they do, other attributes are ignored.
- If either **PVID** or **CONNECTION** is set, and neither value matches a disk on the target system, and no other attributes are set, an error message is generated, and a disk must be explicitly selected.

- If **PVID** is empty and **SAN_DISKID** is set, then, for fibre channel-attached disks, BOS installation interprets the **SAN_DISKID** as a World Wide Port Name and a Logical Unit ID (separated by ″//″). The World Wide Port Name (**ww_name**) and Logical Unit ID (**lun_id**) can be obtained on a running system from the **lsattr** command.

  The **SAN_DISKID** field is checked before the **CONNECTION** field.
- If the **ww_name** and **lun_id** match a disk, other attributes are ignored.
- If either **PVID** or **SAN_DISKID** is set, and neither value matches a disk on the target system, and no other attributes are set, an error message is generated and a disk must be explicitly selected.
- If **PVID** and **SAN_DISKID** are empty and **CONNECTION** is set, BOS installation verifies if the **parent** and **connwhere** attributes (separated by ″//″) match a disk. If this is true, other attributes are ignored.
- If **CONNECTION** is set, the value does not match a disk on the target system, and no other attributes are set, an error message is generated and a disk must be explicitly selected.
- If other attributes are specified, processing occurs as described below:
  - If **LOCATION** is set, BOS installation ignores **SIZE_MB** and **HDISKNAME**.
  - If **LOCATION** is not set and **SIZE_MB** is, BOS installation selects disks based on **SIZE_MB** and ignores **HDISKNAME**.
  - If **LOCATION** and **SIZE_MB** are both empty, BOS installation chooses the disk specified in **HDISKNAME**.
  - If all fields are empty, BOS installation chooses a disk for you.

For the **PVID**, **PHYSICAL_LOCATION**, **SAN_DISKID**, and **CONNECTION** fields, the BOS installation process uses the following logic to determine how to use the target_disk_data stanza information:

- Does the information in one or more of the **PVID**, **PHYSICAL_LOCATION**, **SAN_DISKID**, and **CONNECTION** fields match the disk information?
- If the disk information matches the information in one of these four fields, use that information.
- If the disk information does *not* match the information in one of these four fields, and if the **LOCATION**, **SIZE_MB**, and **HDISKNAME** fields are not set, display an error message and prompt the user for the correct disk information.

The **PHYSICAL_LOCATION** information can be retrieved using the **lsdev** command. For example:

```
# lsdev -Cc disk -l hdisk0 -F "name physloc"
```

returns the `hdisk0` diskname and the `P2/Z1-A8` physical location.

> **Attention:** If **prompt=no**, do not leave the target_disk_data stanzas empty, unless it is unimportant which disk BOS installation overwrites. This is because the algorithm that determines the default disk for the installation is not always predictable.

The **SIZE_MB** field can contain either a size or the word `largest`. If a size is listed, BOS installation does a ″best-fit″ on the disks. If the word `largest` is in that field, BOS installation selects the largest disk. If there is more than one target_disk_data stanza, BOS installation selects the two ″largest″ disks, and so on.

PVID	Specifies the 16-digit physical volume identifier for the disk.
PHYSICAL_ LOCATION	The physical location code provides a way to identify fibre channel disks during BOS Install. For fibre channel disks the **PHYSICAL_LOCATION** field includes the World Wide Port Name and Lun ID that are included in the **SAN_DISKID** field. The information in the **PHYSICAL_LOCATION** field supercedes the information in the **SAN_DISKID** field.
SAN_DISKID	Specifies the World Wide Port Name and a Logical Unit ID for fibre channel-attached disks. The **ww_name** and **lun_id** are separated by two slashes (//). This information can be obtained on a running system from the **lsattr** command.

CONNECTION Specifies the combination of the **parent** attribute and the **connwhere** attribute associated with a disk. The parent and connwhere values are separated by two slashes (//). If the **parent** value is `scsi0` and the **connwhere** value is `0,1`, then the **CONNECTION** value is `scsi0//0,1`. An example of the **CONNECTION** value for a SSA disk would be `ssar//000629CCC07300D`. In the example, the **parent** attribute is represented by `ssar` and the **ConnectionLocation** (15-character unique identity) of the disk drive `000629CCC07300D` is used for the **connwhere** attribute.

SIZE_MB Specifies the formatted size of the disk, in megabytes, where the program is to install BOS. The default value is blank. You can specify the size of your target disk by typing the number of megabytes available on the formatted disk. Also, you can type `largest` if you want to use the largest disk (that has not already been selected) found by the installation program.

LOCATION Specifies a location code for the disk where the program is to install BOS. The default value is blank. If you do not specify a value, the installation program assigns a value based on the next two variables. For more information about physical location codes, refer to the *Diagnostic Information for Multiple Bus Systems* guide.

HDISKNAME Specifies the path name of the target disk. The default value is blank. To name a target disk, use the *hdiskname* format, where *hdiskname* is the device name of your disk (for example, **hdisk0**).

locale Stanza

The locale stanza contains variables for the primary language the installed machine is to use. Refer to Understanding Locale Categories in *AIX 5L Version 5.2 National Language Support Guide and Reference*, which provides information about locales and the format to use when editing variables.

BOSINST_LANG Specifies the language that the installation program uses for prompts, menus, and error messages. The default value is blank.

CULTURAL_CONVENTION Specifies the primary locale to install. The default value is blank.

MESSAGES Specifies the locale for message catalogs to install. The default value is blank.

KEYBOARD Specifies the keyboard map to install. The default value is blank.

large_dumplv Stanza

The optional **large_dumplv** stanza specifies characteristics used if a dedicated dump device is to be created on the systems. A dedicated dump device is only created for systems with 4 GB or more of memory. The following characteristics are available for a dedicated large dump device:

DUMPDEVICE Specifies the name of the dedicated dump device.

SIZEGB Specifies the size of the dedicated dump device in gigabytes.

If the stanza is not present, the dedicated dump device is created when required. A dedicated dump device is created in machines with at least 4 Gigabytes of real memory during an overwrite install. By default, the name of the dedicated dump device is **lg_dumplv** and its size is determined by the following formula:

```
4>=  RAM < 12       size of dump device= 1 GB
12>= RAM < 24       size of dump device= 2 GB
24>= RAM < 48       size of dump device= 3 GB
     RAM >= 48      size of dump device= 4 GB
```

dump Stanza

The dump stanza specifies the following system dump characteristics:

PRIMARY Specifies the primary dump device to be set by **sysdumpdev -P -p *device*.**

SECONDARY Specifies the secondary dump device to be set by **sysdumpdev -P -s *device*.**

COPYDIR Specifies the directory to which the dump is copied at system boot.

FORCECOPY Specifies whether the system boots into menus that allow copy of the dump to external media if the copy fails.

ALWAYS_ALLOW Specifies whether the key mode switch can be ignored when a dump is requested.

If the stanza is not present in the **bosinst.data** file, no additional dump-device handling occurs beyond what is already in place. Checking on the values of the fields is limited; if the device specified for a dump device is not valid, any error processing comes from the **sysdumpdev** command and is sent to the console and stored in the BOS installation log.

- If **FORCECOPY** is specified and no **COPYDIR** is specified, the value field of the **autocopydump** attribute from **/etc/objrepos/SWservAt** is retrieved and used for the **sysdumpdev -[d|D]** ***copydir*** **operation.**
- If only the **COPYDIR** is specified without **FORCECOPY** being specified, **forcecopy** defaults to `yes`. The **sysdumpdev -d** (**FORCECOPY** = no) or **sysdumpdev -D** (**FORCECOPY** = yes) is used to set the copy directory.
- If **ALWAYS_ALLOW=yes**, run **sysdumpdev -K**. Otherwise, run **sysdumpdev -k**.
- If any values other than `yes` and `no` are specified for **FORCECOPY** or **ALWAYS_ALLOW**, the default actions occur, and processing continues.
- If no value is specified for a particular dump field, no analogous **sysdumpdev** operation is performed. This leaves the system values in the appropriate state, even for a migration or system backup image installation. If a **COPYDIR** is specified but **FORCECOPY** is not specified, the value of the **forcecopydump** attribute is retrieved from the **/etc/objrepos/SWservAt** file to determine the correct form of **sysdumpdev** to invoke.

bosinst.data File Example

The following example **bosinst.data** file shows you how customize a nonprompted network installation. The values in the **bosinst.data** file for this example are not specific to a network installation and can be applied for other types of installations, such as a **mksysb** installation.

> **Note:** The depicted values illustrate formatting only and do not apply to your installation.

For information about a **bosinst.data** variable or values, see “bosinst.data File Stanza Descriptions” on page 71.

To check the contents of your customized **bosinst.data** file, use the **bicheck** command, as follows:

`/usr/lpp/bosinst/bicheck` *`filename`*

Nonprompted Network Installation

The following is an example of a modified **bosinst.data** file that might be used in a nonprompted network installation:

```
control_flow:
   CONSOLE = Default
   INSTALL_METHOD = overwrite
   PROMPT = no
   EXISTING_SYSTEM_OVERWRITE = yes
   RUN_STARTUP = no
   RM_INST_ROOTS = yes
   ERROR_EXIT =
   CUSTOMIZATION_FILE =
   TCB = no
   BUNDLES =
   RECOVER_DEVICES = Default
   BOSINST_DEBUG = no
   ACCEPT_LICENSES = yes
   INSTALL_CONFIGURATION =
   DESKTOP = CDE
 INSTALL_DEVICES_AND_UPDATES = yes
 IMPORT_USER_VGS = yes
```

```
 ENABLE_64BIT_KERNEL = yes
 CREATE_JFS2_FS = yes
 ALL_DEVICES_KERNELS = yes
 GRAPHICS_BUNDLE = no
 DOC_SERVICES_BUNDLE = no
 NETSCAPE_BUNDLE = yes
 HTTP_SERVER_BUNDLE = yes
 KERBEROS_5_BUNDLE = yes
 SERVER_BUNDLE = yes
 ALT_DISK_INSTALL_BUNDLE = yes
 REMOVE_JAVA_118 = no

target_disk_data:
   PVID =
   CONNECTION =
   LOCATION =
   SIZE_MB =
   HDISKNAME = hdisk0

locale:
   BOSINST_LANG = en_US
   CULTURAL_CONVENTION = en_US
   MESSAGES = en_US
   KEYBOARD = en_US
```

Chapter 10. Optional Software Products and Service Updates

After the Base Operating System (BOS) is installed, you might want to install optional software or service updates. This chapter includes information on the following topics:

- "Optionally Installed Software"
- "Software Product Identification" on page 82
- "Software Licensing" on page 82
- "Installing Optional Software Products and Service Updates" on page 83
- "Maintaining Optional Software Products and Service Updates" on page 87
- "Cleaning Up Optional Software Products and Service Updates" on page 89
- "Software Service Management" on page 91
- "InstallShield MultiPlatform Packaged Installations" on page 93
- "Emergency Fix Management" on page 98

Note: AIX 5.2 provides the **cdromd** CD and DVD automount facility, which is included in the **bos.cdmount** fileset. To determine if the **cdromd** daemon is enabled on your system, run the following command:

```
# lssrc -s cdromd
```

The **cdromd** daemon can interfere with scripts, applications, or instructions that attempt to mount the CD or DVD device without first checking to see if the device is already enabled. A resource or device busy error occurs in such a condition. Use the **cdumount** or **cdeject** command to unmount the device. Then mount the device as specified in the program or instructions. Alternatively, use the **cdcheck -m** or **mount** command to determine the current mount point of the device. For further information, see the **cdromd** command documentation in the *AIX 5L Version 5.2 Commands Reference*.

The installation code allows for this automatic mounting. If **cdromd** is enabled and the **mkcd** command is run, the CD-R or DVD-RAM is ejected after the image is completed. If you do not want to have the media ejected, then the **cdromd** daemon must be put in the inoperative state with the following command:

```
# stopsrc -s cdromd
```

Optionally Installed Software

Optionally installable software includes:

- **Optional Software Products**: Software that is not automatically installed on your system when you install the BOS. Software products include those shipped with the operating system and those purchased separately. The BOS is divided into subsystems that can be individually updated, such as **bos.rte.install**. Any update that begins with **bos.rte** updates a BOS subsystem.
- **Service Updates**: Software that corrects a defect in the BOS or in an optional software product. Service updates are organized by filesets. This type of update always changes part of a fileset.

Software products can be divided into the following categories:

Licensed Program

A licensed program (LP) is also known as a *licensed program product* (LPP) or a *product*. An LP is a complete software product including all packages associated with that licensed program. For example, **bos** (the base operating system) is a licensed program.

Package

A group of separately installable units that provide a set of related functions. For example, **bos.net** is a package.

© Copyright IBM Corp. 2002, 2004

Fileset
An individually installable option. Filesets provide a specific function. An example of a fileset is **bos.net.nfs.client 5.1**. For more information on fileset packaging, see “Fileset Installation Packaging” on page 156.

Fileset Update
An individually installable update. Fileset updates either enhance or correct a defect in a previously installed fileset.

Bundle
A collection of packages, products, or individual filesets that suit a specific purpose, such as providing personal productivity software or software for a client machine in a network environment. A set of bundles is provided with BOS that contain a specific set of optional software. For more information on bundle packaging, see “Bundle Packaging” on page 157.

A product can be composed of several packages, which in turn can be composed of different filesets. A product might be installed in its entirety, or only certain packages or filesets for the product might be installed. Software products are subdivided in this way, because many software products are large and have many pieces that can be used independently. Dividing a product into separately installable filesets allows you to install only those filesets you need.

You can install all the filesets included in a package or the entire product, or you can install only *selected* filesets, especially if you have limited hard disk space on your system.

Software Product Identification

The product name and level number identify a software product. The format for a software product level in AIX 5.2 is as follows:

versionnumber.releasenumber.modificationlevel.fixlevel

Each field in the software product identification is defined as follows:

- The *versionnumber* field consists of 1 to 2 digits that identify the version number.
- The *releasenumber* field consists of 1 to 2 digits that identify the release number.
- The *modificationlevel* field consists of 1 to 4 digits that identify the modification level.
- The *fixlevel* field consists of 1 to 4 digits that identify the fix level.

For example, `05.01.0000.0000` is a software product level number, and `05.01.0000.0032` is a software product update level. It is not necessary to include the leading zeroes in the version, release, modification level, and fix level fields of the level. Level `05.01.0000.0000` can also be written as `5.1.0.0`.

Software Licensing

The types of software licensing that can be implemented in the software purchase are run-time licensing and acceptance of software license agreements. Normally, software requiring run-time licenses is only selected for installation when you have a license to use that software. Although the Web-based System Manager and System Management Interface Tool (SMIT) allow you to install licensed software even if you do not own a license, you might be prevented from using the newly installed software until you have obtained the appropriate license.

Accepting software license agreements requires that the license agreement be accepted as part of the installation process. If software installed as part of your BOS installation requires accepting a software license agreement, you cannot exit the Configuration Assistant (or the Installation Assistant for non-graphics consoles) until the license agreement has been accepted. You can view as well as accept

the license agreement. The BOS installation can be customized to automatically accept software licenses. For more information, refer to Chapter 8, “Customized BOS Installations,” on page 67.

For optional software installation, you can preview the license agreements on the installation media using the **smit license_on_media** fast path or the **installp -El** command. During the installation process, you can use the menu item to accept the software license, or you can use the **installp** command with the **-Y** flag. To view accepted license agreements on a system, you can use the SMIT **smit installed_license** fast path or the **lslpp -E** command. To view licenses using the Web-based System Manager, type `wsm` to start the application, and select **Software**. When a product is uninstalled, the license agreement acceptance is changed to the inactive state. If the product is reinstalled, you are not be prompted to reaccept the license agreement.

After completing the prerequisites in the next section, your next step is deciding whether to install software with the Web-based System Manager or with SMIT. Descriptions of both applications are included in this chapter.

Before you install optional software and service updates, refer to the specific instructions that accompany your installation media. If you ever need to reinstall your system, refer to the installation media instructions.

> **Note:** For information about developing software products that are installed using the **installp** command, refer to Packaging Software for Installation in *AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs*.

Installing Optional Software Products and Service Updates

If either of the following conditions apply to you, go to the referenced section. Otherwise, continue with the procedures in this chapter.

- If you need to commit updates or remove previously installed software, go to “Maintaining Optional Software Products and Service Updates” on page 87.
- If you are using a network installation server, refer to Part 3, “Network Installation,” on page 179.

Step 1. Complete the Prerequisites

Before installing optional software or service updates, complete the following prerequisites:

- You must be logged in to the system as the root user.
- AIX 5.2 BOS must be installed on your system. If the BOS is not yet installed on your system, go to Chapter 3, “Introduction to Base Operating System Installation,” on page 43, or if you are installing over a network, refer to Part 3, “Network Installation,” on page 179.
- Either insert the media that contains the optional software or service updates into the appropriate drive or know the local or routed path to the software.
- If you are installing service updates and do not have a current backup of your system, use the procedures in “Creating System Backups” on page 127. To create a system backup, you must have the backup fileset (**bos.sysmgt.sysbr**) installed on your system.
- If system files have been modified, back them up separately before updates are applied, because the update process might replace configuration files.
- If you are installing from CD-ROM or DVD-ROM and have a mounted *documentation CD* in the same CD/DVD-ROM drive that you want to install from, run the following commands in the sequence shown:

  ```
  # unlinkbasecd
  # umount /infocd
  ```

- To eject the documentation CD/DVD-ROM, press the eject button on the CD/DVD-ROM drive for at least two seconds.

Step 2. Install Optional Software Products or Service Updates

Optional software products and service updates can be installed using system management tools provided with the operating system. The following sections provide a brief description of the procedures:

- Web-based System Manager. See “Install Optional Software Products or Service Updates with Web-based System Manager.”
- System Management Interface Tool (SMIT). See “Install Optional Software and Service Updates Using SMIT” on page 85.

To view the files as they are being installed, do the following:

- In SMIT, you can set the **DETAILED Output** field to **yes** to list the files being restored during an installation.
- In Web-based System Manager, expand the **Software** container and select **Installed Software**. From the **Software** menu, select **New Software (Install/Update) —> Install Additional Software —> Advanced Method**. In the resulting screen, select the source drive and click on **Advanced**. In the resulting dialog, click beside **Show detailed messages**.
- You can also use the **installp** command with the verbose option **(-V2)** to show which files have been updated.

Install Optional Software Products or Service Updates with Web-based System Manager

The graphics interface provides access to Web-based System Manager options for installing the following:

- Optional software. See “Installing Optional Software.”
- Service updates. See “Installing Service Updates.”
- Software bundles. See “Installing Software Bundles.”

The Web-based System Manager allows you to install software, as well as to change the system’s default install settings and specify other options. By default, the Web-based System Manager *applies* and *commits* any software updates you are installing. However, you can change this default setting and have the software updates applied only.

> **Note:** Base software applications are always committed. If a previous version of the software is installed, it cannot be saved.

Installing Optional Software:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand **Software** in the Navigation area.
4. Select **Overview and Tasks**.
5. Select **Install Software**.

Installing Service Updates:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand **Software** in the Navigation area.
4. Select **Overview and Tasks**.
5. Select **Update Software to the Latest Level**.

Installing Software Bundles:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.

3. Expand **Software** in the Navigation area.
4. Select **Installed Software**.
5. From the **Software** menu, select **New Software (Install/Update) —> Install Bundles (Easy)**.

Install Optional Software and Service Updates Using SMIT

The following installation paths are available in SMIT:

Install Software
Install or update software from the latest levels of software available on the media. To shorten the list of software displayed, message and locale software are omitted from the list. To use this option, type `smit install_latest` on the command line.

Update Installed Software to Latest Level
Update all currently installed software to the latest level available on the installation media. To use this option, type `smit update_all` on the command line.

Beginning in AIX 5L Version 5.2 with the 5200-01 Recommended Maintenance package, if you select the option to install all devices and kernels during a BOS installation, then during subsequent **update_all** processing, any new **devices.*** filesets are installed from the installation media. This option can be turned off by setting the **ALL_DEVICES_KERNELS** variable in the **/var/adm/ras/bosinst.data** file to `no`.

Install Software Bundle
Install complete bundles of software simply by specifying the input device and which bundle you are installing. You can also preview a bundle installation to see what software will be installed and how much space is required in the file system to install the bundle. To use this option, type `smit install_bundle` on the command line.

Update Software by Fix
Install a specific fix for a problem. This menu allows you to list all service fixes on the media and select a fix to install. You can also preview the installation to see what software will be updated and how much space is required in the file system to apply the fix. To use this option, type `smit update_by_fix` on the command line.

Install and Update from ALL Available Software
Install or update software from all software available on the media. To use this option, type `smit install_all` on the command line.

Note: If a problem occurs during the installation of optional software that causes the installation process to halt abnormally, you might have to complete a *cleanup* procedure to remove the partially installed software from the system before attempting to reinstall it. If the system instructs you to do a cleanup, go to “Cleaning Up Optional Software Products and Service Updates” on page 89.

Completing the SMIT Installation and Reading the Status Messages

This section describes the system activity and actions that you must take after the installation process has begun.

1. When you press Enter to start the installation, the COMMAND STATUS screen displays. As the installation proceeds, a series of messages display. The amount of time that the installation takes varies depending on your system and the software you are installing and updating.

   Note: The system might prompt you to insert the volume of the installation media, with a message similar to the following:

   ```
   Mount volume 2 on /dev/cd0.
   Press the Enter key to continue.
   ```

   When this message displays, insert the specified media and press Enter.

When the installation finishes, the **Command: status** field on the COMMAND STATUS screen changes to **OK** or **failed**. **OK** indicates that the installation ran to completion, although some filesets may not have installed successfully. The **failed** status means that there was a problem with the installation. Although a preview installation always finishes with an **OK** status, always check the summaries.

For information about error messages, refer to “Handling System and Error Messages” on page 166.

2. When the installation halts or finishes, the screen returns to the top of the list of messages that display during installation. You can review the message list as described in the next step, or you can exit SMIT and review the **smit.log** file (**/smit.log** or **/home/***user_id***/smit.log**).
3. Review the message list for error messages on software products or service updates that may not have been successfully installed. Use the following procedure to correct any errors in the installation:
   a. Look at the pre- and post-installation summaries at the end of the message list to see whether any installation failure occurred.
   b. Use the message list to determine problems and which software products or service updates were involved. For example, space limits might have been exceeded or the requisites might not have been met for some software. The system lists how much extra space is needed and which requisite software products or service updates to install.
   c. Any product that is marked as *FAILED*, *BROKEN*, or *CANCELLED* can be reinstalled after the condition that caused the failure has been corrected. You do not need to reinstall any service update or software product that was marked as *SUCCESS* in the Installp Summary report. If you need to perform the installation again, change installation settings as appropriate. For example, if requisites were missing, set **AUTOMATICALLY install requisite software?** to **yes**. If there was not enough space to complete the installation, set **EXTEND file systems if space needed?** to **yes**.

      If you need to install again and you have AIX BOS multivolume media, insert volume 1 of the AIX product CDs. Press F3 to return to the previous screen, then restart the installation. See “Handling System and Error Messages” on page 166 for information about **bosboot** command errors that may occur while the installation program is running, and about recovery procedures for these errors.
   d. If the installation was interrupted (for example, a power failure), you might need to use the cleanup procedure before continuing. Press F10 (or Esc+0) to exit SMIT, and refer to “Cleaning Up Optional Software Products and Service Updates” on page 89.
   e. If the software has been installed successfully, and you have no other software to install, go to Step 4.

      If you have additional software to install from a different installation media, remove the media that is in that drive and insert the new media.

      Press F3 (or Esc+3) to return to the previous screen and continue installing the software product or service update.
4. Press F10 (or Esc+0) to exit SMIT.
5. Remove all installation media from the drives.
6. When you are directed, reboot your system by typing:

   ```
   # shutdown -Fr
   ```

Update Installed Software to Latest Level from the Command Line

The **install_all_updates** command updates installed system software to the latest level that is on the media and verifies the current recommended maintenance level.

Beginning in AIX 5L Version 5.2 with the 5200-01 Recommended Maintenance package, if you select the option to install all devices and kernels during a BOS installation, then during subsequent **update_all** processing, any new **devices.*** filesets are installed from the installation media. This option can be turned off by setting the **ALL_DEVICES_KERNELS** variable in the **/var/adm/ras/bosinst.data** file to `no`.

If the **ALL_DEVICES_KERNELS** variable is set to `no`, the **install_all_updates** command does *not* install any filesets that are present on the installation media but not installed on the system, unless these filesets are installed as requisites of other selected filesets.

For **installp** images, all **installp** requisites are enforced.

The following example shows how to install all **installp** updates on the **/dev/cd0** device and to verify the current recommended maintenance level:

```
# install_all_updates -d /dev/cd0
```

For more information about the **install_all_updates** command, refer to the *AIX 5L Version 5.2 Commands Reference*.

Maintaining Optional Software Products and Service Updates

During and after installation, the following major maintenance actions can be taken with optional software products and service updates. Optional software and service updates can be:

- Applied. See "Apply Action (Service Updates Only)."
- Committed. See "Commit Action (Service Updates Only)" on page 88.
- Rejected. See "Reject Action (Service Updates Only)" on page 88.
- Removed. See "Remove Action (Software Products Only)" on page 88.

Whether a particular action can be taken depends on whether the action is being applied to the entire software product, or only to a service update that has had a previous action taken on it.

You can perform these actions using either the Web-based System Manager or the System Management Interface Tool (SMIT) or by using commands directly from the command line. The following sections briefly describe how to do each action using Web-based System Manager, SMIT, or a command. Both Web-based System Manager and SMIT provide online help to guide you through each process.

Apply Action (Service Updates Only)

When installing a service update, it can be left in the *applied* state. In this state, the former version of that software product is saved in the **/usr/lpp/***PackageName* directory. Service Updates in the applied state allow you to restore the former version of the software without having to reinstall it.

Only service updates can be placed in the applied state. In contrast, after you install an entire software product, the product is left in the *committed* state. Software products in the committed state do not save the previous version of the software, because two versions of the same software product cannot be installed at the same time.

To apply a service update using Web-based System Manager:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand **Software**.
4. Select **Overview and Tasks**.
5. Select **Update Software to the Latest Level**.

To apply a service update using SMIT:

Type `smit update_by_fix` on the command line.

To apply a service update from the command line:

Use the **installp -a** command to only apply the update.

Commit Action (Service Updates Only)

Committing a service update removes the previous version of a product from the system, which conserves disk space. After a software product or update has been committed, it cannot be deleted from the system except by removing the entire software product (the base level product and all of its updates) or by force-installing it back to a previous level. To do a force-installation, you must have the base level of the fileset available on media.

Although applying and committing a service update are considered separate actions, both can be accomplished while installing the update. In fact, the default action under Web-based System Manager and SMIT is to both apply and commit the service update during installation. This default can be changed to just apply the update.

To commit a service update using Web-based System Manager:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand the **Software** container.
4. Select **Installed Software**.
5. From the **Software** menu, choose **Software Utilities -> Commit Applied Updates**.

   You can list all service updates in the applied state by selecting **List Installed Software -> Updates in Applied State** from the Software menu.

To commit a service update using SMIT:

Type `smit commit` on the command line.

You can list all service updates in the applied state by typing `smit list_installed` on the command line.

To commit a service update from the command line:

Use the `installp -c` command to commit applied updates.

You can list all service updates in the applied state by typing `installp -s` on the command line.

Reject Action (Service Updates Only)

When you reject an applied service update, the update files are removed from the system and the previous version of the software is restored. Only service updates in the applied state can be rejected. You can use the Web-based System Manager or SMIT to reject applied service updates.

To reject a service update using Web-based System Manager:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand the **Software** container.
4. Select **Installed Software**.
5. From the **Software** menu, choose **Software Utilities —> Reject Applied Updates**.

To reject a service update using SMIT:

Type `smit reject` on the command line.

To reject a service update from the command line:

Use the `installp -r` command to reject an applied update.

Remove Action (Software Products Only)

When you remove a software product, that product's files are removed from the system and the Software Vital Product Data information is changed to indicate that the product is removed. The remove process

also attempts to restore the system's configuration to its previous state, although this is dependent on the product and might not always be complete. After a product is removed, no version of that product remains running on the system.

Use the Web-based System Manager or SMIT to remove software products. In either application, if you set the **Remove dependent software?** field to **yes**, any requisite software (software that is dependent on the product you are removing) is also removed, unless it is required by other software on your system.

To remove a software product using Web-based System Manager:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand the **Software** container.
4. Select **Installed Software**.
5. Select the software product you want to remove.
6. From the **Selected** menu, choose **Remove Software**.

To remove a software product using SMIT:

Type `smit remove` on the command line.

To remove a software product from the command line:

Use the `geninstall -u` command to remove the product.

Copy Software Bundle to Hard Disk for Future Installation

The Copy Software Bundle to Hard Disk for Future Installation option allows you to copy a software bundle from a specified source to a location on your local system.

Installation software bundles include the following:

- Alt_Disk_Install
- CDE
- DocServices
- GNOME
- Graphics
- HTTP_Server
- KDE
- Kerberos_5

Cleaning Up Optional Software Products and Service Updates

> **Note:** This procedure applies only to the update or installation of optional software products. If your AIX 5.2 BOS installation was unsuccessful, see "Accessing a System That Does Not Boot" on page 163 for more information.

This section describes how to clean up software products and service updates after an interrupted installation. The cleanup procedure attempts to delete items that were partially installed or left in an incomplete state. For example, after an update is interrupted, the **lslpp -l** command might report the update status as **APPLYING** rather than **APPLIED**.

The cleanup procedure attempts to revert the update to its previous state. For example, when cleaning up an update that was interrupted in the **COMMITTING** state, the cleanup procedure attempts to return the update to its **APPLIED** state.

If the interruption occurs during the initial state of an installation, then the cleanup procedure attempts to delete the installation entirely and restore the previous version of the product (if there is one). When the

previous version is restored, it becomes the active version. When the previous version cannot be restored, the software is listed by the **lslpp -l** command as **BROKEN**.

When the product is deleted or **BROKEN**, you can attempt to reinstall the software. Any product in the **BROKEN** state cannot be cleaned up; it can only be reinstalled or removed.

The system automatically initiates a cleanup when an installation fails or is interrupted. Normally, you must initiate a cleanup procedure if the system shuts down or loses power during an installation or if the installation process terminates abnormally. Occasionally, you are prompted to reboot (restart) the system after running the cleanup procedure.

To initiate a cleanup procedure using Web-based System Manager:

1. Start the Web-based System Manager by typing `wsm` on the command line.
2. Expand the machine name.
3. Expand the **Software** container.
4. Select **Installed Software**.
5. From the **Software** menu, choose **Troubleshooting —> Cleanup Failed or Interrupted Installation**.

To initiate a cleanup procedure using SMIT:

1. Type `smit maintain_software` on the command line.
2. Select **Clean Up After Failed or Interrupted Installation**.

To initiate a cleanup procedure from the command line:

Type `installp -C` on the command line.

If you get a message indicating that no products were found that could be cleaned up, you may have run the cleanup procedure when it was not needed. Try your installation again.

If you get a message indicating that you need to clean up a failed installation, contact your point of sale for assistance.

Manage an Existing installp Image Source

The **lppmgr** command is used to manage an existing **installp** image source. The **lppmgr** command performs the following functions on an existing **installp** image source (also known as an **lpp_source** resource in the NIM environment):

- Remove duplicate updates (**-u** flag).
- Remove duplicate base levels (**-b** flag).
- Eliminate update images that are the same level as base images of the same fileset. Such update images can create conflicts that lead to installation failure (**-u** flag).
- Remove message and locale filesets other than the language you specify (**-k** flag).
- Remove superseded filesets (**-x** flag).
- Remove non-system images from a NIM **lpp_source** resource (**-X** flag).

By default, **lppmgr** lists all images filtered by the preceding routines. The **-r** flag can be used to remove the filtered images and the **-m** flag can be used to move the images to another location.

The **lppmgr** command does not replace the **bffcreate** command, perform installations, or work with installed filesets. Before using the **-X** flag, it is recommended that you have a good understanding of NIM, system images (known as SIMAGES in NIM), and the workings of a NIM **lpp_source** resource.

To list all duplicate and conflicting updates in the **/myimages** image source directory, type:

```
# lppmgr -d /myimages -u
```

To remove all duplicate and conflicting updates in the **/myimages** image source directory, type:

```
# lppmgr -d /myimages -u -r
```

For more information about the **lppmgr** command, refer to the *AIX 5L Version 5.2 Commands Reference*.

Software Service Management

The Software Service Management menu generates comparison reports to manage filesets installed on a system, filesets contained in a repository, and filesets available from the IBM eServer pSeries support Web site. The Software Service Management menu also provides a way for you to clean up and rename software images in a repository. The following options are available:

- "Comparison Reports"
- "Rename Software Images in Repository" on page 92
- "Clean Up Software Images in Repository" on page 92

You can perform these actions using either the SMIT **service_software** fast path or by using commands directly from the command line. The following sections briefly describe how to do each action using SMIT or a command.

Comparison Reports

The Comparison Reports menu allows you to generate several comparison reports to verify that the filesets for a particular fix or preventive maintenance package are installed by comparing filesets installed on a system to another source. This source could be a fix repository, such as an **lpp_source** or a directory of fixes, or a downloaded list from the IBM eServer pSeries support Web site.

If you want to verify that your **lpp_source** is up to date, you can also compare a fix repository to a downloaded list.

You can perform these actions in the SMIT **compare_report** fast path or using the **compare_report** command.

Compare Installed Software to Fix Repository

The Compare Installed Software to Fix Repository menu allows you to compare the filesets installed on a system to a fix repository. The following report lists are generated:

- Filesets on the system that are back-level (**lowerlevel.rpt**)
- Filesets on the system that are at a later level (**higherlevel.rpt**)
- Filesets in the fix repository that are not installed on the system (**notinstalled.rpt**)
- Filesets installed on the system that are not in the fix repository (**no_update_found.rpt**)

The Compare Installed Software to Fix Repository option is available using the SMIT **instofix_compare** fast path or the **compare_report** command with the following options:

```
compare_report -s -i FixDir {[ -l ] [ -h ] [ -m ] [ -n ]} \
     [ -t ReportDir -Z | -v ]
```

Compare Installed Software to List of Available Updates

The Compare Installed Software to List of Available Updates menu allows you to compare the filesets installed on a system to a downloaded list of available updates from the IBM eServer pSeries service Web site. The following report lists are generated:

- Filesets on the system that are back-level from the latest (**lowerthanlatest1.rpt**)
- Filesets on the system that are at a later level from the latest maintenance level (**higherthanmaint.rpt**)
- Filesets on the system that are back-level from the latest maintenance level (**lowerthanmaint.rpt**)

The Compare Installed Software to List of Available Updates option is available using the SMIT **instolist_compare** fast path or the **compare_report** command with the following options:

```
compare_report -s -r ServiceReport  {[ -l ] [ -h ]} [ -t ReportDir -Z | -v ]
```

Compare Fix Repository to List of Available Updates

The Compare Fix Repository to List of Available Updates menu allows you to compare the filesets in a fix repository, such as a fix directory or **lpp_source**, to a downloaded list of available updates from the IBM eServer pSeries service Web site. The report list that is generated contains information on filesets in the fix directory that are back-level from latest (**lowerthanlatest2.rpt**).

The Compare Fix Repository to List of Available Updates option is available using the SMIT **fixtolist_compare** fast path or the **compare_report** command with the following options:

```
compare_report -i FixDir -r ServiceReport [ -t ReportDir -Z | -v ]
```

Compare a List of Installed Software on a Base System to Another System

You can also compare a list of installed software on a base system to another system. This option allows you to compare the filesets installed on a system to another system. The **lslpp -Lc** output from one system is saved to a file and compared with the **lslpp -Lc** output from another system. The following report lists are generated:

- A list of base system installed software that is at a lower level (**baselower.rpt**)
- Filesets not installed on the base system, but installed on the other system (**otheronly.rpt**)
- A list of base system installed software that is at a higher level (**basehigher.rpt**)
- Filesets installed on the base system that are not installed on the other system (**baseonly.rpt**)

To compare a list of installed software on a base system to another system use the **compare_report** command with the following options:

```
compare_report -b BaseList -o OtherList {[ -l ] [ -h ] [ -m ] [ -n ]} [ -t ReportDir -Z | -v ]
```

Rename Software Images in Repository

The Rename Software Images in Repository option allows you to rename updates that have FIX ID numbers for names, to more meaningful fileset names like those generated when updates are copied to hard disk for future installation. This action renames all filesets in the indicated directory with the same format. This option is available using the SMIT **rename_software** fast path.

You can also use the **bffcreate** command to rename software images in a directory. To rename software images in a directory using the **bffcreate** command, use the **-c** flag and the **-d** flag for the directory containing the filesets. For example, to rename filesets in the **/usr/sys/inst.images** directory, type:

```
# /usr/sbin/bffcreate -cd /usr/sys/inst.images
```

You can also create a log file containing a mapping between the old names and new names, using the **-s** *logfile* option, as shown in the following example:

```
# /usr/sbin/bffcreate -cd /usr/sys/inst.images -s /usr/sys/inst.images/names.log
```

This example creates a **/usr/sys/inst.images/names.log** file that contains content formatted as follows:

```
old_fileset_name:new_fileset_name
```

This option is also available in SMIT Rename Software Images in Repository menu as the **LOG software name changes (location of log file)** option.

Clean Up Software Images in Repository

The Clean Up Software Images in Repository option allows you to remove unneeded or duplicate software images from a local software-image repository. You can remove duplicate software, superseded updates, and language software:

- The Remove Duplicate software option allows you to remove duplicate base and update images from the specified directory.
- The Remove Superseded updates option allows you to remove superseded filesets from the specified directory. This action applies only to update images.
- The Remove Language software option allows you to remove language and locale filesets that are not needed on your system. This option removes all language and locale filesets from the specified directory, except the language specified in the PRESERVE language field. By default, the value of the **LANG** environment variable for the system is used to determine the language to preserve.
- The Save Removed files option allows you to save all removed files to the location specified in the **DIRECTORY for storing saved files** field. Select true in this field if you want to move the images to another location instead of removing them from the hard drive.

This option is available using the SMIT **cleanup_software** fast path.

InstallShield MultiPlatform Packaged Installations

Some products that are distributed for installation on AIX are packaged and installed with InstallShield MultiPlatform (ISMP). Unlike **installp** or RPM Package Manager (RPM) installations which only provide nonprompted or silent installations of a product, ISMP-packaged products provide both interactive and silent interfaces for installing and uninstalling a product.

Similar to products packaged and installed with **installp** and RPM, ISMP-packaged products can be installed using the AIX system management tools, including SMIT and Web-based System Manager. These tools use the **geninstall** command to install or uninstall products that are packaged and installed with **installp**, RPM, or ISMP. As expected, the **geninstall** command can be used directly to install, list, or uninstall ISMP-packaged products.

For instructions for installing or uninstalling a specific product packaged and installed with ISMP, consult the product's documentation.

This section provides information about the following topics:

- "Installing an InstallShield MultiPlatform Product"
- "Uninstalling an InstallShield MultiPlatform Product" on page 94
- "Silent Installations and Using Response Files" on page 95
- "Using Response Files with NIM" on page 97

Installing an InstallShield MultiPlatform Product

You install an InstallShield MultiPlatform product using SMIT, Web-based System Manager, the **geninstall** command, or the files provided by the product.

- Use the SMIT **install_software** fast path to install ISMP products without knowledge of the exact location of the product installation files. For information on installing optional software using SMIT, see "Installing Optional Software Products and Service Updates" on page 83. Use the F4 key on the **SOFTWARE to install field** to select the product you want to install. ISMP products are displayed in the list similar to **installp** packages or RPM packages. Select the ISMP products, and press Enter to begin the installation.

  By default, ISMP product installations launched through SMIT is *silent* or *nonprompted* installations. To perform an interactive installation, use the Web-based System Manager, the **geninstall** command, or the instructions provided with the product documentation.

  Although SMIT has a preview option, this option is not available for ISMP installations. If you select the preview option, a message instructs you to launch an interactive installation using Web-based System Manager or the command line, which allows you to view the preinstallation summary panel before completing the product installation.

- Use the Software Application in the Web-based System Manager to launch ISMP-packaged product installations. For information on installing optional software using the Web-based System Manager, see “Installing Optional Software Products and Service Updates” on page 83. When you press the **Browse** button to list products on the media, ISMP-packaged products are displayed in the list.

  Note: If you select the preview option, but proceed through the entire installation wizard, the product is installed. Most ISMP products include a preinstallation summary panel that provides preview information about the installation, including space requirements and file system expansion. If you do not want to proceed with the installation after viewing this information, press the **CANCEL** button to exit the wizard.
- Use the **geninstall** command to install an ISMP-packaged product. To perform an interactive installation, specify the device or directory containing the product installation files with the **-d** flag and specify the product name. The product name is the same as the subdirectory name containing the product installation files. For example, if we have a product called `MyProduct`, and the product installation files are in the **/usr/sys/inst.images/ismp/ppc/MyProduct/** directory, use the following command for an interactive installation:

  ```
  /usr/sbin/geninstall -d /usr/sys/inst.images MyProduct
  ```

  OR

  ```
  /usr/sbin/geninstall -d /usr/sys/inst.images J:MyProduct
  ```

  Use the **J:** prefix to inform the **geninstall** command that the product is an ISMP package. The **geninstall** command recognizes the **ismp/ppc** subdirectory, just as it recognizes **RPMS/ppc** for RPM packages and **installp/ppc** for **installp** packages, so it is only necessary to pass the **/usr/sys/inst.images** base directory. You can also use the directory that contains the installation files. In this example, specify the directory as follows:

  ```
  /usr/sbin/geninstall -d /usr/sys/inst.images/ismp/ppc/MyProduct J:MyProduct
  ```

  If you want to launch a *silent* or *nonprompted* installation with **geninstall**, include the **-Z** flag:

  ```
  /usr/sbin/geninstall -d /usr/sys/inst.images -Z J:MyProduct
  ```

  For more information about silent installations, see “Silent Installations and Using Response Files” on page 95.
- You can use the installation files provided by the product developer to install an ISMP-packaged product. The product developer might provide a script or executable that can be used to launch an ISMP-packaged product installation. For more information, refer to the documentation provided with the product.

.

Uninstalling an InstallShield MultiPlatform Product

You uninstall an ISMP product using SMIT, Web-based System Manager, the **geninstall** command, or the files provided by the product’s developer.

- You can use the SMIT **remove** fast path to uninstall an ISMP-packaged product. If you use the F4 key to list the installed software for the **SOFTWARE to remove** field, the ISMP-packaged product is displayed in the list. You can also type the name of the product in the field.

  By default, uninstallation processes performed in SMIT are *silent* or *nonprompted*. To perform an interactive uninstallation, use Web-based System Manager, the **geninstall** command, or the instructions provided with the product documentation.

  In SMIT, the preview option is not available for the ISMP product uninstallation procedure. If you attempt to preview the uninstallation, a message instructs you to launch an interactive uninstallation using Web-based System Manager or the command line. This allows you to view the pre-unstallation summary panel before completing the product uninstallation.
- You can use the Software Application in Web-based System Manager to uninstall ISMP-packaged products.

Note: If you select the preview option, but proceed through the entire uninstallation wizard, the product is uninstalled. Most ISMP products include a pre-uninstallation summary panel that provides preview information about the uninstallation. If you do not want to proceed with the installation after viewing this information, press the **CANCEL** button to exit the wizard.

- You can use the **geninstall** command to perform an uninstallation for an ISMP-packaged product. To perform the uninstallation interactively, specify the **-u** flag for uninstallation, and the product name. For example, to uninstall the *MyProduct* product, type the following:

  ```
  /usr/sbin/geninstall -u MyProduct
  ```

  OR

  ```
  /usr/sbin/geninstall -u J:MyProduct
  ```

  To speed processing, use the **J:** prefix to inform the **geninstall** command that you are uninstalling an ISMP-packaged product.

  To perform a *silent* or *nonprompted* uninstallation with the **geninstall** command, use the **-Z** flag, as follows:

  ```
  /usr/sbin/geninstall -Zu J:MyProduct
  ```

- You can use installation files provided by the product developer to uninstall an ISMP-packaged product. The product developer might provide instructions for performing an ISMP-packaged product uninstallation. For more information, see the documentation provided with the ISMP product.

Silent Installations and Using Response Files

The section provides information on how to perform silent installations for ISMP-packaged products using response files.

A response file contains predetermined responses for an installation. By default, the **geninstall** command searches on the product media in the ISMP-product subdirectory for response files for each ISMP product. For example, the `MyProduct` ISMP product subdirectory is similar to the following:

`/basedir/ismp/ppc/MyProduct/`

The **geninstall** command searches in the ISMP-product subdirectory for each ISMP product specified in the install list or bundle for a ***.response** file. If multiple ***.response** files are found, the file named **Product.response** is used. If a ***.response** file does not exist, a warning message is displayed and the installation of the ISMP product is skipped.

The **-t** *ResponseFileLocation* option allows you to specify an alternate location for response files or response file templates. The *ResponseFileLocation* can either be a file or directory name. If the *ResponseFileLocation* is a directory, it must already exist. If the *ResponseFileLocation* is not an existing directory, it is assumed that a file name is specified.

To use response files with ISMP products, the following methods are available:

- Create a response file template. To create an ISMP response file template in the default location, use the **geninstall** command with the **-T** flag. The **-T** flag creates an ISMP response file template in the default location, which is the directory containing the product installation files. The resulting template can be used to create a response file for future installations of the same product with the desired options. Creation of the response file template does not result in installation of the ISMP product.

  To create an ISMP response file template for the `MyProduct` ISMP product using the product installation files in the `/usr/sys/inst.images/ismp/ppc/MyProduct/` default directory, do the following:

  ```
  /usr/sbin/geninstall -d /usr/sys/inst.images -T J:MyProduct
  ```

  The `MyProduct.template` response file template that is generated is similar to the following:

  ```
  ################################################################
  #
  # InstallShield Options File Template
  #
  ```

```
# Wizard name: Setup
# Wizard source: setup.jar
# Created on: Tue Jun 25 10:59:55 CDT 2002
# Created by: InstallShield Options File Generator
#
# This file can be used to create an options file (i.e., response file) for the
# wizard "Setup". Options files are used with "-options" on the command line to
# modify wizard settings.
#
# The settings that can be specified for the wizard are listed below. To use
# this template, follow these steps:
#
#    1. Enable a setting below by removing leading '###' characters from the
#    line (search for '###' to find settings you can change).
#
#    2. Specify a value for a setting by replacing the characters <value>.
#    Read each settings documentation for information on how to specify its
#    value.
#
#    3. Save the changes to the file.
#
#    4. To use the options file with the wizard, specify -options <filename>
#    as a command line argument to the wizard, where <filename>  is the name
#    of this options file.
#
############################################################################
############################################################################
#
# My Product Install Location
#
# The install location of the product. Specify a valid directory into which the
# product is  installed. If the directory contains spaces, enclose it in
# double-quotes. For example, to install the product to C:\Program Files\My
# Product, use
#
#    -P installLocation="C:\Program Files\My Product"
#
### -P installLocation=<value>
```

Although the preceding is a simple example, products often have many user-configurable options that might be set in the response file. Each of these options is presented in the template with an explanation of the expected value for that option.

- Create a response file recording. To create a response file recording, use the **geninstall** command with the **-E** flag. The **-E** flag creates an ISMP response file recording in the default location, which is the directory containing the product installation files. This option requires running the ISMP installation interactively and completely. The resulting response file can be used with the **-R** flag to select the same options on future installations of the same product. Creation of the response file recording will also result in installation of the ISMP product.

  To record the `MyProduct.response` response file with the `MyProduct` ISMP product and the product installation files in the `/usr/sys/inst.images/ismp/ppc/MyProduct/` default directory, do the following:

  ```
  /usr/sbin/geninstall -d /usr/sys/inst.images -E J:MyProduct
  ```

  This starts the interactive installation wizard. It is necessary to run the wizard to completion to successfully create the response file recording. When completed, a message similar to the following displays:

  ```
  Options record mode enabled - run the wizard to completion to create the options file response.file
  ```

  The resulting file `MyProduct.response` response file is similar to the following:

  ```
  ############################################################
  #
  # InstallShield Options File
  #
  # Wizard name: Setup
  # Wizard source: setup.jar
  # Created on: Tue Jun 25 11:05:34 CDT 2002
  ```

```
# Created by: InstallShield Options File Generator
#
# This file contains values that were specified during a recent execution of
# Setup. It can be used to configure Setup with the options specified below when
# the wizard is run with the "-options" command line option. Read each setting's
# documentation for information on how to change its value.
#
# A common use of an options file is to run the wizard in silent mode. This lets
# the options file author specify wizard settings without having to run the
# wizard in graphical or console mode. To use this options file for silent mode
# execution, use the following command line arguments when running the wizard:
#
#    -options "record.txt" -silent
#
################################################################################
################################################################################
#
# My Product Install Location
#
# The install location of the product. Specify a valid directory into which the
# product is  installed. If the directory contains spaces, enclose it in
# double-quotes. For example, to install the product to C:\Program Files\My
# Product, use
#
#    -P installLocation="C:\Program Files\My Product"
#
-P installLocation="/opt/MyProduct"
```

The **-P installLocation** value has been completed according to the response given while running the wizard. In the preceding example, the **/opt/MyProduct** directory was specified as the installation location in the wizard. The response file generated by this action can be used directly to launch a silent installation with the chosen installation location.

- Use a response file for a silent installation. You can use a response file generated by the two methods mentioned previously or one provided with the product to perform a silent installation with the desired options.

  To use a response file for a silent installation with the **geninstall** command, the `MyProduct` product, and the installation files and response file in the `/usr/sys/inst.images/ismp/ppc/MyProduct/` default directory, do the following:

  ```
  /usr/sbin/geninstall -Zd /usr/sys/inst.images -R J:MyProduct
  ```

  To use a response file for a silent installation with the **geninstall** command, `MyProduct` product, installation files in `/usr/sys/inst.images/ismp/ppc/MyProduct/` directory, and the `/tmp/MyProduct/MyProduct.response` response file, do the following:

  ```
  /usr/sbin/geninstall -Zd /usr/sys/inst.images -R \
        -t /tmp/MyProduct/MyProduct.response J:MyProduct
  ```

Using Response Files with NIM

If you are using NIM to install an ISMP-packaged product on one or more NIM clients, you can create and use a separate response file for each client. This is useful when there are properties of the installation operation that must be configured differently for each client. In order to install multiple clients, you must name each response file *CLIENT_NAME***.response**. These response files must be located in the default location (the same location as the product installer files).

For example, to install the **MyProduct** ISMP-packaged product located in an **lpp_source** resource in the **/export/lpp_source/lpp_source1/ismp/ppc/MyProduct** directory on the **CLIENT1** and **CLIENT2** clients, do the following:

1. Create a **CLIENT1.response** and **CLIENT2.response** response file.
2. Place the response files in the **/export/lpp_source/lpp_source1/ismp/ppc/MyProduct** directory.
3. Create the correct responses for each client in the corresponding response file.

4. When you run the NIM **cust** operation to install the **MyProduct** ISMP-packaged product on **CLIENT1** and **CLIENT2**, the response files are used automatically and appropriately for each client.

If you want to use the same response file for all clients, name the response file *PRODUCT_NAME***.response** and place in the same default location as the ISMP-packaged product (the product location in the **lpp_source** resource). For example, create a response file called **MyProduct.response** in the **/export/lpp_source/lpp_source1/ismp/ppc/MyProduct/** directory. If there are no client response files when you perform the NIM **cust** operation, the **MyProduct.response** file is used automatically.

Emergency Fix Management

The emergency fix (efix) management solution allows users to track and manage efix packages on a system. An efix package might be an emergency fix, debug code, or test code that contains commands, library archive files, or scripts that run when the efix package is installed.

The efix management solution consists of: the efix packager (**epkg**) command and the efix manager (**emgr**) command.

The **epkg** command creates efix packages that can be installed by the **emgr** command. The **emgr** command installs, removes, lists, and verifies system efixes.

Note: In this section when the term *package* is used, **installp**'s reference is the term *fileset*.

This section includes the following information:

- "Installing and Managing Emergency Fixes"
- "Additional Emergency Fix Information" on page 105
- "Packaging Emergency Fixes" on page 109

Installing and Managing Emergency Fixes

The **emgr** command installs and manages packages created with the **epkg** command, and maintains a database with efix information on the system. The **emgr** command performs the following operations:

- "Displaying efix Packages"
- "Installing efix Packages" on page 99
- "Removing an efix Package" on page 101
- "Listing Operation" on page 102
- "Checking Operation" on page 103
- "Mount Installation Operation" on page 104
- "Mounting and Unmounting Operations" on page 104
- "Display Package Locks Operation" on page 104
- "Force Removal Operation" on page 104

Displaying efix Packages

The **emgr** command's **-d** flag displays the contents and topology of the efix package. The **-d** flag works with the **-v** (verbosity) flag. The default verbosity level is 1, but you can the level to 1, 2, or 3. For example, to get a level 1 verbosity output on the efix package **test.102403.epkg.Z**, type the following command.

```
# emgr -d test.102403.epkg.Z
```

The verbosity levels include the following information:

LEVEL 1

Lists one efix per line with the following information:

- Label for the efix package
- efix files contained in the package
- Target location for each efix file

LEVEL 2

Lists the following information:

- All LEVEL 1 information
- Abstract
- Reboot requirement (`yes` or `no`)
- Prerequisite files needed
- Pre-install script
- Post-install script
- Pre-remove script
- Post-remove script
- File type for each efix file

LEVEL 3

Lists the following information:

- All LEVEL 2 information
- Packaging date for each efix file
- Virtually unique ID (VUID) for each package
- File size for each efix file
- Checksum for each efix file
- Package for each efix file
- Description of each efix file
- Contents of installation scripts and control files, if they are readable text
- Reboot scenario for each efix file
- Prerequisites of efix file on other efix files
- Packages that will be locked when the efix is installed
- efixes that will be superseded when the efix is installed

Note: Displaying is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

Installing efix Packages

The **emgr** command installs efix packages that are created with the **epkg** command. The following example shows the syntax for installing an efix package:

```
emgr -e efix pkg | -f input file [-w dir] [-bkpIqmFX]
```

The efix package installation operation consists of the following phases:

1. “Installation Preview Phase”
2. “Installation Phase” on page 100
3. “Summary and Cleanup Phase” on page 101

Installation Preview Phase: During the installation preview phase, the following steps occur:

1. The efix manager initializes all commands and libraries, and extracts the efix metadata from the efix package.
2. The efix attributes and description are listed.

3. The **emgr** command performs a lock-checking procedure by checking the installed efix data if the target files that are being delivered by this efix package already have existing efixes installed. If one or more efix files that are delivered by this efix package are locked, the **emgr** command does not allow the installation or installation preview to proceed.
4. The **emgr** command performs **installp** package prerequisite verification. If the user supplied an **installp** prerequisite file, the **emgr** command checks the prerequisites at this stage. If one or more of the prerequisites is not met, the **emgr** command does not allow the installation or installation preview operation to proceed.
5. The **emgr** command performs efix to efix prerequisite verification. The **emgr** command checks the following:
   - All efix prerequisites for the efix package being checked are installed.
   - All efix xrequisites for the efix package being checked are not installed.
   - There are no previously installed efixes that list the efix being checked as an xrequisite.

   Note: You can use the **epkg** command to specify the efix to efix prerequisites for an package. For more information on checking prerequisites, see "User-Specified efix Package Components" on page 110.
6. The **emgr** command checks for space requirements by checking whether the target file systems contain adequate space for the installation of the efix package. This includes space for unpacking the efix files, creating database entries, saving replaced files, installing efix files, creating efix mounts when using the **-m** flag, archiving library members, and other miscellaneous tasks. The **emgr** command also adds a small buffer to the various space calculations to account for file metadata and other factors.

   If the user specifies the auto-expand flag using the **-X** flag, then the **emgr** command attempts to expand the file system to the required size. If space requirements cannot be met, the **emgr** command stops the installation. If the user specifies a preview installation using the **-p** flag, then the **emgr** command only reports the space statistics without attempting expansion

   If the user specifies a preview installation using the **-p** flag, the **emgr** command does not perform the efix installation phase. Instead the **emgr** command skips to the summary and cleanup phase of installation.

Installation Phase: During the installation phase, the following steps occur:

1. During the efix installation setup step, the entire efix package is unpacked and the installation tools are initialized.
2. The **emgr** command checks whether the efix package supersedes any other currently installed efix files. If any of the installed efix files should be superseded, the **emgr** command removes them.

   Note: You can use the **epkg** command to specify that an efix package be superseded when you install another efix package. For more information on superseding, see "User-Specified efix Package Components" on page 110.
3. If a **pre_install** script is specified, it is run. If the **pre_install** script returns a failure, the **emgr** command halts the installation. If the **pre_install** script succeeds, the **emgr** command proceeds with the installation and sets the efix state to `INSTALLING`.
4. Any files that are replaced by efix files in the efix package are saved to a secured directory. If the efix package does not deliver any files, this step is skipped.

   From this point forward, any major failure causes the **emgr** command to run a failure-cleanup procedure, which attempts to clean up the failed installation. If this process fails, the efix is placed into the `BROKEN` state.
5. All efix files are installed to their target locations. If the installation is a mount installation operation using the **-m** flag, then the **emgr** command creates a unique mount file within the parent directory of the target file. The target file is then over-mounted by the efix mount point. For more information about the mount installation operation, see "Mount Installation Operation" on page 104.

6. Package locking occurs. The efix package locks are processed. If the installer for which the efix package is created supports efix package locking, the **emgr** command locks the package associated with the efix files installed in step 4. For example, the **installp** command supports efix locking, so an efix created for an **installp** package will support efix package locking.

   Note: In addition to implicit locking, you can use **epkg** to specify that a certain package be explicitly locked when you install another package. For more information on locking, see "User-Specified efix Package Components" on page 110.
7. If a **post_install** script is specified, it is run. If the **post_install** script returns a failure, the **emgr** command halts the installation.
8. Reboot processing occurs. If the efix package specifies that a reboot operation is required, the **emgr** command issues a message to the user and makes any necessary changes to the boot image. The **emgr** command does *not* reboot the system automatically.
9. At this point, all installation steps have succeeded and the **emgr** command changes the efix state to `STABLE` for a standard installation operation, or `MOUNTED` for a mount installation operation.

Summary and Cleanup Phase: During the summary and cleanup phase, the following steps occur:

1. The **emgr** command displays a summary of all operations and results. If more than one efix package was specified with an input file using the **-f** flag, the **emgr** command provides a report for each efix package.
2. The **emgr** command cleans up any temporary directories and files. It also unloads any memory modules that have been loaded into memory.

Removing an efix Package

The efix removal operation removes an installed efix. You can specify an individual efix by using one of the efix identification methods or specifying several individual efixes by using a list file. For more information about the efix identification methods, see "Referencing Emergency Fixes" on page 105.

The syntax for removing an installed efix is as follows:

```
emgr -r -L label | -n efix num | -u VUID | -f lfile [-w dir] [-bkpIqX]
```

The efix package removal operation consists of the following phases:

1. "Removal Preview Phase"
2. "Removal Phase"
3. "Summary and Cleanup Phase" on page 102

Removal Preview Phase: During the removal preview phase, the following steps occur:

1. The efix manager initializes all commands and libraries, and loads efix metadata from the efix database.
2. The efix attributes and descriptions are listed.
3. Space requirements are checked. The **emgr** command checks whether the target file systems contains adequate space to restore the saved files. This includes space-changing database entries, restoring saved files, archiving library members, and other miscellaneous tasks. The **emgr** command also adds a small buffer to the various space calculations to account for file metadata and other factors.

   If the user specifies to auto-expand the file system using the **-X** flag, the **emgr** command attempts to expand the file system to the required size. If space requirements cannot be met, the **emgr** command halts the remove operation. If the user specifies a preview installation operation using the **-p** flag, then the **emgr** command only reports the space statistics without attempting to expand the file system.

   If the user specifies a preview installation using the **-p** flag, the **emgr** command does not perform the efix removal and skips to the summary and cleanup phase.

Removal Phase: Any failure in the removal phase causes the efix state to change to `BROKEN`. During the removal phase, the following steps occur:

1. The **emgr** command initializes all remove utilities and changes the efix state to REMOVING.
2. Package unlocking occurs. All packages that are locked by the efix file being removed are unlocked. Because it is possible that a single package may be locked by multiple efixes, the **emgr** command only unlocks a package if this efix file is the last (or the only) efix file still holding a lock on the given package.
3. If a **pre_remove** script is specified, it is run. If the **pre_remove** script returns a failure, the **emgr** command halts the remove operation.
4. **emgr** checks that the efix being removed is not a prerequisite for another installed efix.
5. The efix is removed. If the efix was installed with a standard installation operation, the **emgr** command replaces the current efix files with the previously saved files. If the installation was a mount installation operation, the **emgr** command unmounts the efix files and removes them from the system.
6. If a **post_remove** script is specified, it is run. If the **post_remove** script returns a failure, the **emgr** command halts the installation.
7. Reboot processing occurs. If the efix package specified that a reboot is required, the **emgr** command issues a message to the user and make any necessary changes to the boot image. The **emgr** command does not reboot the system automatically.

   Note: You can use **epkg** to specify the reboot scenario you want when you install another package. For more information on reboot scenarios, see "User-Specified efix Package Components" on page 110.
8. At this point, all removal steps have succeeded and the **emgr** command removes the remaining efix data from the database and save directories.

Summary and Cleanup Phase: During the summary and cleanup phase, the following steps occur:

1. The **emgr** command issues a summary of all operations and results. If more than one efix package was specified with an input file using the **-f** flag, the **emgr** command reports for each efix package.
2. The **emgr** command cleans up any temporary directories and files. It also unloads any memory modules that have been loaded.

Listing Operation

The **emgr** command lists data on installed efixes with various levels of verbosity. The syntax for listing efixes is as follows:

```
emgr -l [ -L label | -n efix num | -u VUID ] [-v{1-3}X]
```

By default, the **emgr** command reports data on all installed efixes. You can specify an individual efix by using one of the efix identification methods. For information about the efix identification methods, refer to "Referencing Emergency Fixes" on page 105.

The default level of verbosity is 1. You can specify up to level 3 with the **-v** flag. The verbosity levels include the following information:

LEVEL 1

Lists one efix per line with the following information:

- efix ID
- efix state
- Install time
- efix abstract

LEVEL 2

Lists the following information:

- All LEVEL 1 information
- Virtually unique ID (VUID) for each efix file
- Number of efix files

- Location for each efix file
- Package for each efix file
- Installer for each efix File
- mount installation (yes or no) for each efix file

LEVEL 3

Lists the following information:

- All LEVEL 2 information
- Reboot requirement (yes or no)
- Prerequisite files needed
- Pre-install script
- Post-install script
- Pre-remove script
- Post-remove script
- File type for each efix file
- File size for each efix file
- Checksum for each efix file
- Access ownership and modes for each efix file
- Prerequisite information
- efix description
- Archive member name for each efix file
- If this is a mount installation operation, then display the mount status for each efix file
- Reboot scenario for each efix file
- efix to efix prerequisites for each efix file
- Packages that will be locked when the efix is installed
- efixes that will be superseded when the efix is installed

Checking Operation

The **emgr** command checks the status of installed efixes. The syntax for efix checking is as follows:

```
emgr -c [ -L label | -n efix num | -u VUID | -f lfile ]
         [ -w dir] [-v{1-3}X]
```

By default the **emgr** command verifies all installed efixes. You can specify an individual efix by using one of the efix identification methods or specify several individual efixes by using a list file. For information about the efix identification methods, refer to “Referencing Emergency Fixes” on page 105.

The default level of verification is 1. You can specify up to level 3 with the **-v** flag. The verification levels include the following checks:

LEVEL 1

Checks the following information:

- efix data and state
- If this is a mount installation operation, then check the efix mount status for all files

  Note: If the efix file is unmounted, the **emgr** command changes the efix state to UNMOUNTED
- efix checksum for all efix files or archive members

LEVEL 2

Checks the following information:

- All LEVEL 1 checks
- efix ownership and mode for all efix files or archive members

LEVEL 3

Checks the following information:

- All LEVEL 2 checks
- All prerequisites
- All efix to efix prerequisites, including the following:
  - All efix prerequisites for the efix package being checked are installed.
  - All efix xrequisites for the efix package being checked are not installed.
  - There are no installed efixes that list the efix being checked as an xrequisite.

Mount Installation Operation

If the **-m** flag is specified during efix installation, the **emgr** command performs a mount installation operation of the efix package. This means that the existing files that are being fixed are not removed from their present locations. Instead they are over-mounted by the efix files. This approach has both advantages and disadvantages. One advantage is that a system reboot unmounts all of the efixes. This means that any efix that created a serious problem is not mounted after a reboot. The disadvantages are that the administrator must monitor the mount status of efixes and some efixes may not be removed without a reboot.

The mount installation operation is not supported with efix packages that deliver new files.

Mounting and Unmounting Operations

The **emgr** command mounts or unmounts efixes that have been installed using the mount installation operation. The syntax for efix checking is as follows:

```
emgr -M | -U [ -L label | -n efix num | -u VUID | -f lfile]
     [ -w dir] [X]
```

By default, the **emgr** command applies the mount or unmount operation to all installed efixes. You can specify an individual efix by using one of the efix identification methods or specify several individual efixes by using a list file. For more information about the efix identification methods, see “Referencing Emergency Fixes” on page 105.

Using the mount operation with the **-M** flag, the **emgr** command attempts to mount all efix files that are unmounted. If all efix files are successfully mounted, and the previous efix state was `UNMOUNTED`, then the **emgr** command changes the efix state to `MOUNTED`.

Using the unmount operation with the **-U** flag, the **emgr** command attempts to unmount all efix files that are mounted. If at least one efix file is successfully unmounted, and the previous efix state was `MOUNTED`, then the **emgr** command changes the efix state to `UNMOUNTED`.

Display Package Locks Operation

The display package locks operation displays all packages that are locked by efix manager, their installer, and the locking label or labels. The syntax for the display package-locks operation is as follows:

```
emgr -P [ Package ] [X]
```

By default, the **emgr** command lists all locked packages. The user can specify an individual package as an argument to the **-P** flag.

Force Removal Operation

The force removal operation removes efix data. This operation also unlocks all efix packages associated with the efix label without removing the actual efix files, executing any removal scripts, or boot processing. The force removal operation can only be run on one efix at a time, and the efix label is required to identify the target efix. The syntax for performing a force removal operation is as follows:

```
emgr -R efix label [-w dir] [X]
```

Note: The force removal operation must be considered as an emergency procedure. It must *only* be run if all other methods to remove the efix have failed. This method can create inconsistencies on the target system.

Additional Emergency Fix Information

Additional information about emergency fixes includes the following topics:

- "Using MD5 Security Enhancements"
- "Referencing Emergency Fixes"
- "Using a List File" on page 106
- "Emergency Fix States" on page 106
- "Emergency Fix Logging" on page 107
- "Failure-Cleanup" on page 107
- "Considerations for Systems Using the Trusted Computing Base" on page 107
- "Emergency Fix Manager Command Paths" on page 107

Using MD5 Security Enhancements

At the beginning of any operation involving **epkg** formatted images, **emgr** looks on the system for a supported command that generates an MD5 checksum. If a command is located, **emgr** executes this command and displays the resulting MD5 checksum. You can then cross check this MD5 checksum with a secured source. If a command is not located, **emgr** takes no further action. You can force set an explicit path to a command that generates a checksum by exporting the **EMGR_MD5_CMD** shell variable. This variable should contain the absolute path to the command. **emgr** does not verify that the user set command in the **EMGR_MD5_CMD** variable is an actual command that generates an MD5 checksum. The syntax used by **emgr** to generate the MD5 checksum is as follows:

```
$EMGR_MD5_CMD epkg image file
```

The expected output is the MD5 checksum as the first word in the output.

Note: This feature is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

Referencing Emergency Fixes

Emergency fixes can be referenced as follows:

Reference by Label

Each efix that is installed on a given system has its a unique efix label. This label is called the *unique key* that binds the different database objects. To reference an efix by label, pass the label as an argument to the **-L** flag.

For example, to run a check operation on an efix with label ABC123, type the following:

```
# emgr -cL ABC123
```

Reference by efix ID

Each efix that is installed on a given system has a unique efix ID. This ID is the order number in which the efix is listed in the efix database. Using this option may be convenient if performing operations on efixes based on efix listings. The **emgr** command converts the efix ID into an efix label before performing the given operation. To reference an efix by ID, pass the ID as an argument to the **-n** flag.

For example, to run a check operation on the first efix with an ID equal to 1, type the following:

```
# emgr -cn1
```

Note: Emergency fix IDs are valid for short periods of time and change as efixes are removed and added. *Always* verify the current efix ID number by listing the efix using the **-l** flag.

Reference by VUID

The VUID is used to differentiate packages that have the same label. Unlike Authorized Program Analysis Reports (APARs), which are officially tracked, emergency fixes are not tracked by any organization, so it is possible to have two efix packages with the same label. However, the **emgr** command does not allow the installation of more than one efix with the same label. The **emgr** command converts the VUID into an efix label before performing the given operation. For example, to list an installed efix with VUID equal to `000775364C00020316020703`, type the following:

```
# emgr -l -u 000775364C00020316020703
```

The VUID is also displayed in the preview section of the efix installation and removal operations, and when using the listing operation with verbosity level 2 or greater. For more information on verbosity levels, see “Listing Operation” on page 102

Using a List File

You can perform operations on a set of efixes by specifying a list file. For the installation operation, the list file must contain one efix package location per line. For the removal operation and the mount and unmount operations, the list file must have one efix label name per line. The **emgr** command ignores any blank lines, or lines where the first non-white space character is the `#` character.

Emergency Fix States

The **emgr** command maintains a state for each installed efix, as follows:

STABLE

The efix was installed with a standard installation, and successfully completed the last installation operation. To verify the efix details, run a check operation on the given efix or efixes.

MOUNTED

The efix was installed with a mount installation operation, and successfully completed the last installation or mount operation. A state of `MOUNTED` does not mean all efixes are currently mounted. For example, the efixes might have been manually unmounted. This state represents the **emgr** command’s previous actions and determination of the mount status. To verify the efix details, including mount status, run a check operation on the given efix or efixes.

UNMOUNTED

The efix was installed with a mount installation operation and one or more efix files were unmounted in a previous **emgr** command operation. The state of `UNMOUNTED` does not mean that all efixes are currently unmounted. For example, the efixes might have been manually mounted or partially mounted. This state represents the **emgr** command’s previous actions and determination of the mount status. To verify the efix details, including mount status, run a check operation on the given efix or efixes.

BROKEN

An unrecoverable error occurred during an installation or removal operation. The status of the efix is unreliable. You can attempt to remove this efix and reinstall it from the efix package.

INSTALLING

The efix is in the process of installing. Normally, this state occurs only for a brief time during efix installation. However, if an efix installation is suddenly interrupted (such as in a sudden power loss or a system crash), and the **emgr** command is unable to clean up the failed installation, an efix might be left in the `INSTALLING` state. You can attempt to remove this efix and reinstall it from the efix package.

REBOOT REQUIRED

The efix was installed successfully and requires a reboot to fully integrate into the target system. After you reboot the target system, **emgr** changes the efix state to `STABLE`.

Note: This feature is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

REMOVING
The efix is in the process of being removed. Normally, this state occurs only for a brief time during efix removal. However, if an efix installation is suddenly interrupted (such as in a sudden power loss or a system crash), and the **emgr** command is unable to clean up the failed installation, an efix might be left in the `REMOVING` state. You can attempt to remove this efix and reinstall it from the efix package.

Note: **emgr** is called by [1] with the **-B** bootup flag during system start. **emgr** examines the state data for all efixes and changes the efix states as necessary. The **/etc/inittab** entry `rcemgr` is created the first time **emgr** installs an efix file. `rcemgr` is set to not block or respawn. All `rcemgr` activities and output are logged to the **emgr** log in **/var/adm/ras/emgr.log**. Do not directly execute **emgr** with the **-B** flag.

Emergency Fix Logging
The following operations are logged in the **/var/adm/ras/emgr.log emgr** log file:

- Installation
- Removal
- Checking
- Mounting
- Unmounting
- Force Removal

Failure-Cleanup
The failure-cleanup procedure is run when an efix installation operation fails after the installation preview (and **pre_install** script, if specified). The failure-cleanup procedure attempts to reverse any of the changes that have already been made by the installation process and is similar to the removal phase of the efix removal operation. This procedure sets the **EMGR_UNDO** global environment variable to `1` and allows packaging to take different paths in the **pre_remove** and **post_remove** scripts.

Considerations for Systems Using the Trusted Computing Base
The **emgr** command automatically detects if a system is enabled with the Trusted Computing Base (TCB). If TCB is enabled, the **emgr** command registers all of the installed efixes with the efix database. When the efixes are removed, the **emgr** command restores the original TCB data. Because mount installation operations can create variations in file attributes when efix files are mounted and unmounted, mount installation operations are not supported on a TCB-enabled system and are blocked by the **emgr** command.

If you do not want the **emgr** command to automatically manage TCB data, export the **EMGR_IGNORE_TCB** variable and set this variable to any value that is not null. When the **EMGR_IGNORE_TCB** variable is set, the **emgr** command behaves as if the system is not TCB-enabled. If the **EMGR_IGNORE_TCB** variable is set on a TCB-enabled system, you might be required to manually manage efix files within TCB.

To check if TCB is enabled on your system, run the **/usr/bin/tcbck** command. If a usage statement is returned, TCB is enabled. Otherwise, a message indicating that TCB is not enabled is returned.

Emergency Fix Manager Command Paths
The **emgr** command calls one or more of the following UNIX commands:

ar

1. init

awk
cat
chmod
chown
compress
cp
date
df
diff
du
egrep
fuser
id
ksh
ln
ls
mkdir
mount
mv
printf
ps
rm
rmdir
sed
sleep
sort
sum
tail
tar
tee
touch
umount
uname
vi
wc
zcat

The **emgr** command calls one or more of the following AIX commands:

aclget
aclput
bosboot
lslpp
odmchange
odmget
slibclean

tcbck

The **emgr** command looks for the UNIX and AIX commands previously listed in the following path order:

1. /usr/emgrdata/bin
2. /usr/bin
3. /usr/sbin
4. /bin
5. /sbin
6. /usr/local/bin
7. /usr/local/sbin

The **/usr/emgrdata/bin** directory is a secured directory that is created the first time the **emgr** command is run.

If you are attempting to install or remove an efix for one of the commands that the **emgr** command uses, you might not be able to successfully complete the operation. To solve this problem, do the following:

1. Manually install the efix file into the **/usr/emgrdata/bin** directory.
2. Perform the **emgr** operation.
3. Remove the manually installed efix file from the **/usr/emgrdata/bin** directory.

Using this method, the efix is registered and tracked with efix manager and all other **emgr** command processing takes place.

If the efix file is the **/usr/bin/ksh** file and the problem it fixes prevents the **emgr** command's operations from succeeding, then do the following:

1. Back up the original **/usr/bin/ksh** file.
2. Manually install the **/usr/bin/ksh** efix file to **/usr/bin/ksh**.
3. Perform the **emgr** command installation or remove operation.

Packaging Emergency Fixes

If you need to create your own efix and package it for distribution, use the **epkg** command to package the efix. The **epkg** command can be run in two modes: *interactive* and *template-based*. The interactive method prompts the user with several questions and constructs the efix package based on the answers. The template-based method uses an efix control file that is pre-filled with default answers that are then asked in interactive mode. The efix package can then be installed by the **emgr** command.

By using an efix control file as a template, efix packages can be created noninteractively. The following is an example of a completed efix control file:

```
# efix control file example
ABSTRACT=This is a test of epkg.
PRE_INSTALL=/tmp/pre_install
POST_INSTALL=.
PRE_REMOVE=/tmp/pre_remove
POST_REMOVE=.
REBOOT=yes
PREREQ=.
DESCRIPTION=/tmp/description
EFIX_FILES=2

EFIX_FILE:
        EFIX_FILE_NUM=1
        SHIP_FILE=/home/test/ls
        TARGET_FILE=/usr/bin/ls
        TYPE= 1
        INSTALLER= 1
        ACL= DEFAULT
```

```
        AR_MEM=.

EFIX_FILE:
        EFIX_FILE_NUM=2
        SHIP_FILE=/home/test/mystrcat.o
        TARGET_FILE=/usr/ccs/lib/libc.a
        TYPE= 2
        INSTALLER= 1
        ACL= root:system:555
        AR_MEM=strcat.o
```

User-Specified efix Package Components

The following efix control-file components are part of the overall efix package and are not related to specific files:

ABSTRACT
Describes the efix package. The abstract is limited to 38 bytes.

DESCRIPTION
Contains a detailed description of the efix package that is being installed.

E2E_PREREQ
Lists the efix label names of efixes that are prerequisites to the efix package being installed. Using this file causes **emgr** to check if the efix **PREREQ** label is installed. If the prerequisite is not installed, **emgr** aborts installation of the efix package. You can also use this file to specify an **XREQ** efix label. Specifiying **XREQ** efix labels causes **emgr** to not install the efix package if the specified efix is installed. The maximum number of supported efix labels is 32. You can specify the efix labels to check for in the following ways.

- Specify the file location with the **-g** flag. For example, to specify **efixprereq.epkg**, type the following:

  ```
  # epkg -g /tmp/efixprereq.epkg myefix
  ```

- Use the **-v** flag in interactive mode for extended options, and type the file location when prompted by **epkg**. For example, to specify **efixprereq.epkg**, when prompted, type the following:

  ```
  Enter the location for the supersede file or "." to skip.
     -> /tmp/efixprereq.epkg
  ```

- Set the **E2E_PREREQ** attribute in the efix control file to the local file location of the efix prerequisite file. For example, to specify **efixprereq.epkg**, set the attribute as follows:

  ```
  E2E_PREREQ=/tmp/efixprereq.epkg
  ```

The format of the efix prerequisite file entries is as follows (where **RequisiteType** is *PREREQ* or *XREQ*):

```
EfixLabel RequisiteType
```

Comments beginning with a ″#″ sign and leading white space are ignored. For example:

```
oldefix1 PREREQ  # Make sure oldefix1 is already installed
oldefix4 XREQ    # Make sure oldefix4 is NOT installed
```

Note: This feature is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

PKGLOCKS
Lists the packages that should be locked by **emgr** in addition to those that are automatically locked based on file ownership. You should specify the name of the package, the package lock action (either *ALWAYS* or *IFINST*), and the package file type. *ALWAYS* means always attempt to lock this package, and a failure to lock the package results in efix installation failure. *IFINST* means attempt to lock this package only if the package is installed, and failure to lock an *installed*

package results in efix installation failure. The maximum number of supported efix labels is 32. You can specify the packages to be locked in the following ways.

- Specify the file location with the **-l** flag. For example, to specify **pkglock.epkg**, type the following:

  ```
  # epkg -l /tmp/pkglock.epkg myefix
  ```

- Use the **-v** flag in interactive mode for extended options, and type the file location when prompted by **epkg**. For example, to specify **pkglock.epkg**, when prompted, type the following:

  ```
  Enter the location for the supersede file or "." to skip.
     -> /tmp/pkglock.epkg
  ```

- Set the **PKGLOCKS** attribute in the efix control file to the local file location of the package to be locked. For example, to specify **pkglock.epkg**, set the attribute as follows:

  ```
  PKGLOCKS=/tmp/pkglock.epkg
  ```

The format of the efix package locks file entries is as follows:

```
PackageName PackageAction PackageType
```

Comments beginning with a ″#″ sign and leading white space are ignored. In the following example, **emgr** will always attempt to lock **bos.rte.lvm** during installation and will unlock it on removal. **emgr** will lock **bos.games** if (and only if) it is installed and will unlock it on removal (if locked).

```
bos.rte.lvm ALWAYS installp
bos.games   IFINST installp
```

Note: This feature is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

PRE_INSTALL
Runs after an installation preview and before any efix files are installed. Failure in the **pre_install** script causes the efix package installation to be aborted. This script is useful for doing any preinstallation checking or work. Because the **emgr** command does not call a failure-cleanup procedure for preinstallation failures, this script performs failure cleanup (related to the script) before it exits. This component is optional.

POST_INSTALL
Runs after all efix files have been successfully installed. A failure in the **post_install** script causes the installation to fail and causes efix manager to run a failure-cleanup procedure. This component is optional. For more information about the **post_install** script, refer to “Installing and Managing Emergency Fixes” on page 98.

PRE_REMOVE
Runs after the removal preview and before any efix files are removed during a remove operation and in the first stage of a failure-cleanup procedure. A failure in the **pre_remove** script causes the given operation to fail. In the case of a failure-cleanup procedure, the **emgr** command sets an **EMGR_UNDO** global environment variable to 1. If necessary, the **EMGR_UNDO** variable is used to take different actions for removal as opposed to a failure-cleanup. This component is optional.

POST_REMOVE
Runs after efix files are removed during a remove operation and a failure-cleanup procedure. A failure in the post-remove script causes the given operation to fail. In the case of a failure-cleanup procedure, the **emgr** command sets an **EMGR_UNDO** global environment variable to 1. The **EMGR_UNDO** variable is used to take different actions for removal as opposed to a failure-cleanup (if necessary). This component is optional.

REBOOT
Indicates whether a reboot operation is required for this efix. You can use this variable to specify one of the following reboot scenarios.

- Reboot is not required.
- Reboot is required, and the boot image will be rebuilt.
- Reboot is required, and the boot image will not be rebuilt.

You can specify the which of these reboot scenarios you want in the following ways.

- Specify the reboot scenario with the **-r** flag. Arguments for this flag are *n* (reboot is not required), *y* (reboot required and the boot image will be rebuilt), and *o* (reboot is required, but the boot image will not be rebuilt). For example, the following command specifies that a reboot is not required:

  ```
  # epkg -r n
  ```

- Use the **-v** flag in interactive mode for extended options, and select the reboot scenario you want when prompted by **epkg**. For example:

  ```
  Select reboot policy for this efix package:
     1) Reboot is NOT required.
     2) Reboot is required. The boot image will be rebuilt.
     3) Reboot is required. The boot image will NOT be rebuilt.
  ```

- Set the **REBOOT** and **BUILD_BOOTIMAGE** attributes in the efix control file to the appropriate values for the reboot scenario you want. For example, to specify that a reboot is not required, set the attributes as follows:

  ```
  REBOOT=no
  BUILD_BOOTIMAGE=no
  ```

  To specify that a reboot is required and the boot image will be rebuilt, set the attributes as follows:

  ```
  REBOOT=yes
  BUILD_BOOTIMAGE=yes
  ```

  To specify that a reboot is required and the boot image will not be rebuilt, set the attributes as follows:

  ```
  REBOOT=yes
  BUILD_BOOTIMAGE=no
  ```

  Notes:

  1. Setting **REBOOT** to *no* and **BUILD_BOOTIMAGE** to *yes*results in an error from **epkg**.
  2. This feature is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

PREREQ

Contains **installp** prerequisites. This component is optional.

- The file has one prerequisite per line.
- The format of the prerequisite entry is as follows:

  Fileset Min Level Max Level Type

  Fileset
  The name of the requisite **installp** fileset.

  Min Level
  The minimum level for the requisite fileset. The specification of NONE indicates no minimum level.

  Max Level
  The maximum level for the requisite fileset. The specification of NONE indicates no maximum level.

Type The following types are supported: `PREREQ` and `IFREQ`. `PREREQ` is the default type and requires that the requisite fileset meets all criteria. `IFREQ` requires that requisite fileset meet all criteria only if it is installed.

- Blank lines or lines that start with `#` are ignored, as shown in the following examples:

```
# Require that abc.xyz is installed at any level:
  abc.xyz NONE NONE
# Require that bos.rte.lvm is installed at level 5.1.0.10 or above:
  bos.rte.lvm 5.1.0.10 NONE
# Require bos.mp be between 5.1.0.0 and 5.1.0.40 if it is installed:
  bos.mp 5.1.0.0 5.1.0.40 IFREQ
```

SUPERSEDE

Specifies the efix label name of an efix or efixes that are to be superseded when an **epkg** is installed. Using this file causes **emgr** to remove any efix labels that are specified in this file (if they are installed) before installing the efix package. Failure to remove an installed superseded efix aborts the installation of the efix package. The maximum supported number of superseded labels is 32. You can specify the superseded file in the following ways.

- Specify the file location with the **-S** flag. For example, to specify **susperseded.epkg**, type the following:

```
# epkg -S /tmp/superseded.epkg myefix
```

- Use the **-v** flag in interactive mode for extended options, and type the file location when prompted by **epkg**. For example, to specify **susperseded.epkg**, when prompted, type the following:

```
Enter the location for the supersede file or "." to skip.
   -> /tmp/superseded.epkg
```

- Set the **SUPERSEDE** attribute in the efix control file to the local file location of the superseded file. For example, to specify **susperseded.epkg**, set the attribute as follows:

```
SUPERSEDE=/tmp/superseded.epkg
```

The format for the list of superseded files is one efix label to be superseded per line. Comments beginning with a ″#″ sign and leading white space are ignored. For example:

```
# Requisites for efix myefix3
myefix1
myefix2
```

Note: This feature is not supported in the original release of efix management. You should update to the latest level of efix management to enable this feature. To update efix management, update **bos.rte.install** to the latest level.

File Components

The following efix control-file components are related to specific files. The maximum number of efix files for each efix that the **epkg** and **emgr** commands support is 200.

EFIX_FILE_NUM

Number of the given file (1 - 200).

SHIP_FILE

Local file location that the **epkg** command is archiving into the efix package. Specify either an absolute or relative path to this file. The ship file is the efix that is delivered.

TARGET_FILE

Target file location where the **SHIP_FILE** is installed. This location is located on the system where the efix package is installed. Specify the absolute path to this file. If this file is part of a registered package, such as an **RPM** or **installp** package, you must specify the tracked location.

INSTALLER

This variable represents the type of installer that owns the efix package. Valid integer choices are as follows:

1 Tracked by **installp**

2 Tracked by **RPM**

3 Tracked by **ISMP**

4 Tracked by another installer

5 New file that will be tracked by **installp**

6 New file that will be tracked by **RPM**

7 New file that will be tracked by **ISMP**

8 New file that will be tracked by another installer

9 Not tracked by any installer

TYPE This is the type of file that is being installed. The valid choices are as follows:

1 Standard file or executable file

2 Library or archive member

An example of **TYPE 1** is the **/usr/bin/ls** file or the **/usr/bin/rm** file. An example of **TYPE 2** is the **shr.o** archive member as a member of the **libc.a** library.

ACL Specifies the access attributes (mode and ownership) for a given file. If this attribute is set to `DEFAULT`, the **emgr** command maintains the current permissions of the file to be replaced. However, if the target file is a new file or the user wants to specify permissions using the **-v** flag, the **ACL** attribute can be entered with the *Owner:Group:OctalModes* syntax, similar to the following:

```
ACL= root:system:555
```

AR_MEM

Specifies the name of the archive member. This option is only valid if `TYPE=2`. In this case, **SHIP_FILE** represents the local location of the archive member that is being shipped, **TARGET_FILE** represents the target archive, and **ACL** applies to the archive member. For example, the following attributes ship the **myshr.o** local file to the **shr.o** member in the **/usr/ccs/lib/libc.a** target archive:

```
TYPE=2
SHIP_FILE=/home/myshr.o
TARGET_FILE=/usr/ccs/lib/libc.a
AR_MEM=shr.o
```

Automatic Common Components

The following components are part of the overall efix package and are not related to specific files. These components are automatically determined by the **epkg** command. Typically, the user does not set the following components:

DATE Date and time that the backup was made.

INSTWORK

Amount of space (in 512 byte-blocks) required for unpacking the efix metadata.

VUID Virtually Unique ID. A combination of time and **cpuid**, this ID can be used to differentiate fixes that are otherwise identical.

QNEXT and QORDER

Internal trackers for interactive mode. If you are using an efix control file in nonprompted mode, make sure **QNEXT** and **QORDER** are not set, or set to `QEND`.

Automatic File Components

The following components are related to specific files. These components are automatically determined by the **epkg** command. Typically, the user does not set these components.

CKSUM
File checksum for the given file

SIZE Size for the given file

Interactive Mode

By default, the **epkg** command is run in interactive mode. The only required parameter is the efix label. If you interrupts a running **epkg** command session, the efix control file is saved. If you start a new session with the same efix label, you are asked whether youwant to keep working with the previous efix control file. You can use the **-u** flag to answer this question.

The **epkg** command maintains a record of the question order and allows the user to navigate between questions by using subcommands. Also, the **epkg** command remembers the previous answer the user provided and sets that answer as the default answer. The **epkg** subcommands are the following:

b! Returns to the previous question.

s! Shows the current efix control file.

q! Quits without saving efix control file (using the Ctrl-C key sequence will prompt this subcommand).

h! Displays help information for the current question.

The **epkg** command asks the following questions, one at a time:

```
1. Enter efix abstract [38 bytes maximum]:
   ** If "-s" flag is specified, go to question #3 ***

2. Does this efix deliver one or more files ? (yes/no):
   ** If "no", go to question #9 **

3. Enter the local ship file location for efix file number 1:

4. Enter target location for efix file number 1:

5. Select file type for efix file number 1:
   1) Standard (file or executable)
   2) Library member
   3) Other

6. Select the installer which tracks the file that is being fixed by efix
   file number 1:
   1) Currently tracked by installp.
   2) Currently tracked by RPM.
   3) Currently tracked by ISMP.
   4) Currently tracked by another installer.
   5) This is a NEW file that will be tracked by installp.
   6) This is a NEW file that will be tracked by RPM.
   7) This is a NEW file that will be tracked by ISMP.
   8) This is a NEW file that will be tracked by another installer.
   9) Not tracked by any installer.

   *** If "-m flag" and not new go to #7.1 ***
   *** If new, go to #7.2 ***
   *** Else, go to #8 ***

7.1 Enter the ACL for file 1 in the format of <owner>:<group>:<octal modes>.
For example to make the user="root", the group="system", and the modes "444",
you would enter root:system:444. Enter "." if you want to keep the default
(i.e. current) permissions on the existing target file.

7.2 Enter the ACL for file 1 in the format of <owner>:<group>:<octal modes>.
For example to make the user="root", the group="system", and the modes "444",
you would enter root:system:444.

8. Are there more efix files ? (yes/no):
```

```
   ** If "yes", increment file and go to question #3 **
   ** If "no" and "-s" flag, go to #14 **
   ** If "no" go to question #9 **

9.  Enter the local location for the pre-install script or "." to skip.

10. Enter the local location for the post-install script or "." to skip.

11. Enter the local location for the pre-remove script or "." to skip.

12. Enter the local location for the post-remove script or "." to skip.

14. Is a reboot required after installing this efix ? (yes/no):

15. Enter the local location for the installp prerequisite file or "." to skip.
    *** This question is skipped if -p flag ***

16. Enter the local location for the efix description file or "." to compose it
in an editor:
     *** This question is skipped if "-d" flag is specified           ***
     *** If the description file is not specified, the user will be   ***
     *** put into an editor to compose it. The user can specify       ***
     *** which editor to use by setting the EDITOR global environment ***
     *** variable. The default editor is /usr/bin/vi.                 ***
```

After all of the questions are answered, the **epkg** command verifies the efix control file and creates a compressed **tar** package that can be installed by using the **emgr** command.

Related Information

- For information about developing software products that are installed using the **installp** command, see Packaging Software for Installation in *AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs*.
- For information about creating a new backup of your system, see “Creating System Backups” on page 127.
- Read the readme files that were shipped with the operating system. Any additional software you installed also might have associated readme files with additional updates. For information about how to view readme files, see “Viewing Readme Files” on page ix.
- For information about installing the Documentation Library Service or the operating system’s online documentation, see “Using the Documentation Library Service” on page 118.
- For additional release information, see the *AIX 5.2 Release Notes*.

Chapter 11. AIX Documentation and the Documentation Library Service

The AIX online documentation library is available through the IBM eServer pSeries Information Center. The IBM eServer pSeries Information Center provides information about IBM eServer pSeries and the AIX operating system.

Even if you do not want to install the operating system documentation, you should install and configure the documentation library service, because other applications might depend on this service to provide their online documentation.

This chapter covers the first-time installation and configuration of the library service. If you already have the service configured and you want to change the configuration, see Changing the Configuration of the Documentation Library Service in the *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*.

This chapter provides information about the following topics:

- "Information Center"
- "Using the Documentation Library Service" on page 118
- "Installing the Online Documentation" on page 125

Information Center

The IBM eServer pSeries Information Center installs with the Base Operating System in the following filesets:

X11.Dt.bitmaps
AIX CDE Bitmaps

X11.Dt.rte
AIX Common Desktop Environment

bos.rte.install
LPP Installation Commands

Because the **bos.rte.install** fileset is always installed, the **infocenter** command will always be available. The **X11.Dt.bitmaps** and **X11.Dt.rte** filesets are installed only if the CDE (Common Desktop Environment) desktop is installed. If the CDE desktop is installed, an **Information Center** icon is available on the desktop.

The Information Center Web address is contained in the **/usr/lpp/bosinst/bos.vendor.profile** file. The Information Center command and icon starts the Information Center Web page in a browser only if a valid Web address is specified in this file.

To view the Information Center content fully, a Web browser that supports JavaScript is required.

The IBM eServer pSeries Information Center is available at the following Web address:

`http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base`

© Copyright IBM Corp. 2002, 2004

Using the Documentation Library Service

The Documentation Library Service allows you to navigate, read, and search registered HTML-formatted documents through your Web browser. The library service presents documents in a expandable-tree format, through which you can easily navigate by clicking on associated buttons. You can click to view a selected document. You can also search for specific information in one book, a selection of books, or the entire installed library.

The library service produces two types of graphical user interfaces (GUIs), a global GUI and an application GUI. The global GUI shows you all HTML documents on the document server that are registered with the global GUI. Global views may contain documents from many different applications. Access the global library application by typing **docsearch** on the command line or by clicking the **Documentation Library** icon in the Help subpanel under the CDE Desktop front panel.

The application GUI is launched when you click on a link inside a menu or document of an application. The resulting display shows library pages that contain online information for that application. For example, the **Search** link in the Web-based System Manager Help menu calls a library page that only displays the documentation for Web-based System Manager.

The components of the Documentation Library Service are installed along with the BOS. After installation, the service may need to be configured.

> **Note:** If you ordered a preinstalled system from the factory, the complete library service might have already been installed and configured at the factory.

If you are not sure whether the Documentation Library Service is installed and configured on your system, go to "Testing the Documentation Library Service." If you need to install or configure the library service, go to "Configuring the Documentation Library Service" on page 119.

Testing the Documentation Library Service

If you are not sure whether the library service is installed and configured, type `docsearch` on the command line.

One of the following occurs:

- **The library service appears, without error messages, and the search form appears at the top of the page.** This search form contains one or more fields for entering words that you want to search for. A listing of documents is visible in at least one of the views. When you open a document, you can read its contents. The search function completes successfully.
  - **The library service appears functional but you cannot read the documents properly.** Go to "Configuring the Documentation Library Service" on page 119.
  - **The library service appears functional but the search fails.** Go to Problem Determination in the *AIX 5L Version 5.2 System Management Guide: Operating System and Devices*.
- **You see the message `There are no documents installed for this view`**. The library service might be installed correctly, but the service cannot find any installed documents that are registered for the current view. Try clicking on the other views to see if they contain documents. When you find a view with documents, try reading and searching the documents.

  You cannot fully test the library service until all appropriate documentation has been installed and registered. The operating system and all applications that use the library service register themselves during document installation. To install the operating system online documentation, see "Installing the Online Documentation" on page 125. Ensure any application documentation is correctly installed. After you have done this, retest the library functions.
- **You see the message `Search is not enabled`**. The service has not been fully installed and configured. Go to "Configuring the Documentation Library Service" on page 119.

- **You see the message** `Search is not supported in this language`. The search engine cannot support searches in the language used by this installation of the operating system. This is not an error. The search function is not compatible with all languages supported by the operating system. This message always displays when you are using a language that the search function cannot support.

  If documents are visible in one of the Views, try reading the documents. If your documents display correctly and you can open them for reading, then no further configuration of the library service is necessary. If no documents are available in any view, the documentation has not been installed and registered. The operating system and all applications that use the library service register themselves during document installation. To install the operating system online documentation, see “Installing the Online Documentation” on page 125. Ensure any application documentation is correctly installed. After you have done this, retest the library functions.
- **You see a message similar to** `Cannot find or execute ds_form`. The exact wording of this message varies, but it means the library service is not installed and configured correctly. Go to “Configuring the Documentation Library Service.”
- **No browser appears.** The library service is not installed and configured correctly. Go to “Configuring the Documentation Library Service.”

Configuring the Documentation Library Service

You can set up a machine either as a documentation server or as a documentation client system. When users on a client request a search form or an HTML document, the request is sent to the Web server on a documentation server, which then sends back the requested object. When searches are performed, they are done on the server and the results are then sent back to the user on the client.

A documentation server has the following software installed:

- Documentation library service server and client software
- Library service search engine
- Documents
- Document search indexes - for searchable documents. If an application package is using the library service, these indexes are installed automatically when the documents are installed, or the system administrator may manually create indexes for locally written documents. Documents written in certain languages cannot be searched and do not have indexes.
- Web server software

A client needs only the Documentation Library Service client software and a Web browser installed.

If you have a standalone machine, both the server and client software are installed. Instead of going to a remote server, requests from users on a standalone machine go to the Web server software on that same computer. A documentation server on a network can also be made standalone in the sense that you can configure its Web server software to accept only requests from users logged in to the documentation server.

If you have a console that supports a graphical user interface and are running in AIXwindows, you can use the Configuration Assistant to install and configure the library service. Otherwise, you can use commands to install and configure manually. It is highly recommended that you use the Configuration Assistant because it automatically performs some steps for you and is easier to use.

To use the Configuration Assistant, see “Using Configuration Assistant for the Documentation Library Service” on page 120.

To use the manual method, see “Manually Installing and Configuring the Documentation Library Service” on page 120.

Using Configuration Assistant for the Documentation Library Service

The Configuration Assistant helps you configure this system as a documentation server or as a client that gets its documents from a remote documentation server.

> **Note:** HTTP Web server software must be installed on this computer before you can install documents and register them with the Documentation Library Service. The Configuration Assistant can automatically install the Lite NetQuestion Web server software or the IBM HTTP Server software. Both options are shipped with the operating system.
>
> If you want to use different Web server software, that software must have been installed and configured before you launch the Configuration Assistant. You must also know the full pathnames of the Web server's HTML documents home directory and the CGI-BIN directory.
>
> This machine does not need Web server software if you are configuring it as a client that obtains its documents from a remote documentation server.

To launch the Configuration Assistant, make sure you are logged in to the system as the root user, and enter the **configassist** command. When the Configuration Assistant opens, press **Next**. Then select **Configure Online Documentation and Search**. The Configuration Assistant guides you through installation and configuration.

After you have finished, use the same procedure as shown in step 3 on page 121 to add language support.

Manually Installing and Configuring the Documentation Library Service

This section describes how to install and configure the Documentation Library Service using the system management tools.

For information about installing the client, see "Installing the Client" on page 124.

Installing the Server

Use this procedure if you want to configure this machine as a documentation server, where you will install your online documentation. A server can be networked to serve remote clients, or it can be standalone to serve only its own users.

To create a documentation search server, complete the following steps:

1. **Install the Web server and browser software, if needed.**

   Web server software must be installed. You can use any Web server software that can run CGI programs. The Lite NetQuestion Web server is automatically installed with the base operating system, but it can serve only local users, not remote users.

   To serve both local and remote users, you must install server software. Any compatible HTTP server software can be used; however, the IBM HTTP Server software can be installed from a CD shipped with AIX 5.2.

   To verify what is already installed, type `smit list_installed` on the command line.

   To manually install the IBM HTTP Server software using SMIT, do the following:

   a. Use the SMIT **install_bundle** fast path.
   b. Select **HTTP_Server** from the bundle list.
   c. Verify with the product documentation if you need to select **yes** to accept new license agreements.
   d. Accept the default values for the remaining Install Software Bundle menu options.

   To manually install the IBM HTTP Server software from the command line, type:

   ```
   # geninstall -d. -IacYXg -f HTTP_Server.bnd
   ```

If the software product contains a license agreement, the **-Y** flag accepts the new license agreement.

A Web browser that can display forms must be installed. (The Netscape browser is on the *AIX 5.2 Expansion Pack* CD.)

2. **Configure and start Web server software.**

   Unless you are using the automatically installed Lite NetQuestion server, you must configure your Web server software to perform correctly with your system. Consult the documentation provided with your Web server software to configure and start your Web server software. Write down the full path names of the Web server directories where the server starts looking for HTML documents and CGI programs. If you are using one of the following Web servers, and you installed to the default location, you do not need to know the directory name:

   - IBM HTTP Server
   - IBM Internet Connection Server

   Other Web servers might not automatically create the HTML and CGI directories. If yours does not, you must create these directories before you continue.

   You must also configure your Web server software's permissions to allow access from the users and remote computers who should use this computer as their documentation search server.

3. **Configure the Documentation Library Service for additional languages, if needed.**

   You might want this documentation server to be able to serve documents that are written in a different language than the one used by the operating system installed on this machine. If this is the case, you must install additional language support filesets for the other languages you want to serve.

   For example, assume that you are using English when you are installing the base operating system and you want users to be able to access documents in both English and Spanish from the documentation server. The English messages will be automatically installed because that is the language in use during the operating system installation. You need only to manually install the Spanish language support package.

   There are two ways to install the library service package for another language:

   - After you install the base operating system and the Documentation Library Service, you can install the operating system locale (language environment) for the language you want to add. Installing a language's locale installs the available support for all applications that are currently installed on your system.

     To install or update an entire locale, type `smit mle_add_lang` on the command line. Use the online help if you need additional explanation.
   - If the locale for the desired language has been installed before the **bos.docsearch** package was installed, you can install only the Documentation Library Service messages.

   To install library service messages using Web-based System Manager:

   a. Start the Web-based System Manager by typing `wsm` on the command line.
   b. Expand the machine name.
   c. Expand **System Environment** in the Navigation area.
   d. Select **Settings**.
   e. Select **Cultural**, then **Available Resources**.
   f. When the dialog appears, use either the pull-down menu or select **Add** and install the language.

      Always install the Common messages. If you have the desktop installed on your system, also install the CDE messages. For example, assume that your server is configured for English when you are installing and you want to be able to serve both English and Spanish documents from the documentation server. The English messages are automatically installed. Therefore, you only need to manually select Spanish.

      The message filesets are not inside the docsearch package; instead, they are stored in the BOS messages packages (**bos.msg.*locale***, where locale=desired language) and are named:

      - **bos.msg.***locale***.docsearch.client.com** DocSearch CDE Action - *locale*
      - **bos.msg.***locale***.docsearch.client.Dt** DocSearch Common Messages - *locale*

g. After you have added all of the filesets you want to install into the window, click on the **OK** button.

h. A message dialog displays, showing the status of the installation. If the process completes with no problems, a **Success** message displays.

To install library service support using SMIT:

a. From the command line, type `smit install_all`.

b. Always install the Common messages. If you have the desktop installed on your system, also install the CDE messages. For example, assume that your server is configured for English when you are installing and you want to be able to serve both English and Spanish documents from the documentation server. The English messages are automatically installed. Therefore, you only need to manually select Spanish.

The message filesets are not inside the **bos.docsearch** package; they are stored in the BOS messages packages (**bos.msg.***locale*, where *locale* is the desired language) and are named:

- **bos.msg.***locale***.docsearch.client.com** DocSearch CDE Action - *locale*
- **bos.msg.***locale***.docsearch.client.Dt** DocSearch Common Messages - *locale*

In the dialog box that displays, type in the location of the filesets you want to install in the **INPUT device / directory for software** field, or click on the **List** button for a list of the available devices and directories.

c. When the next dialog displays, enter the fileset names you want to install. If you want to install more than one fileset at this time, separate each fileset name with a space. If you do not know the exact name of the fileset you want to install, or want to browse the available languages, click on the **List** button. Highlight the packages you want to install from the list.

d. Click on **OK**.

4. **Configure the Documentation Library Service.**

To configure the Documentation Library Service using Web-based System Manager:

a. As root user, start the Web-based System Manager by typing **wsm** on the command line.

b. Expand the machine name.

c. Expand **System Environment** in the Navigation area.

d. From the **System Environment** menu, choose **Settings —> Documentation Server**.

e. In the Browser command field, type the name of the command that launches the Web browser with a URL. This selection will be the default Web browser for all users of this computer.

- If you are using the Netscape browser, type `netscape` in the Browser field and continue with the next step.
- If you are using another browser, you must include any necessary flags that include a URL in the launch command. For example, if launching your Web browser to a specific site requires a **-u** flag, and your launch command looks similar to the following:

  YourWebBrowser `-u http://www.w3.org`

  Your entry in the Browser field would be:

  YourWebBrowser `-u`

  You do not include the URL itself in the Browser field entry.

  Note: Many browsers (for example, Netscape) do not require a flag.

f. Click **OK**.

g. Select **Settings —> Documentation Server** again.

h. Select **Local server**.

i. Under the **Location of document and CGI programs on local server**, select your Web server software. If the name of your Web server software is not listed, select **Other**, then type the full path names of the CGI directory and the Documents directory. If you selected your Web server from the listing, continue with the next step.

Note: If you installed any of the listed Web servers in any location other than the default, or if you have set up servers to use nonstandard locations for their CGI-BIN and HTML directories, you must select **Other**.

j. In the Server port field, type in the port number that the Web server software is using. The standard port is `80`. If you are not certain of the actual port number, set it to `80`. If you are using the Lite NetQuestion Web server, the port number must be set to `49213`.

k. Optionally, you can change the Default Documentation Language from this same screen. This is the language users see when they launch the library using the **docsearch** command or the library icon in the Help subpanel of the CDE desktop front panel.

l. Click **OK** to complete the configuration.

To configure the Documentation Library Service using SMIT:

a. On the server, change to root user. On a command line, type `smit web_configure`.

b. Select **Change/Show Default Browser**. In the ***Default browser LAUNCH COMMAND** field, type the command that sets the default browser for all users on this computer. Include any flags that are required when a URL is included in the command. If you want to have the default browser to open to a specific URL, also add the URL after the command to open the browser. For example, if you want your browser to open with the **http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base** page, type the following in the ***Default browser LAUNCH COMMAND** field:

```
YourWebBrowser flag http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base
```

Note: Many browsers (for example, Netscape) do not require a flag.

c. Return to the Web configuration main menu. Select **Change Documentation and Search Server**.

d. On the Documentation Search Server LOCATION screen, click **List** and select **local - this computer** for server location. Click **OK**.

e. On the Web Server SOFTWARE screen, click **List** and select the Web server software that you are using.

Note: If you installed any of the listed Web servers in any location other than the default, or if you have set up servers to use nonstandard locations for their CGI-BIN and HTML directories, you must select **Other**.

f. On the next screen, type in the full path names of the CGI directory and the Documents directory. If you selected a listed Web server that is installed in its default location, the correct path names should be already displayed.

If you set up your Web server to use a port other than the standard port 80, enter that port number. Leave it set to 80 if you are not sure of the port number. If you are using Lite NetQuestion, the port number must be set to `49213`.

g. Click **OK**. When the configuration is completed, a `Documentation server configuration completed!` message displays in the results panel.

h. Optionally, you can change the Default Documentation Language from this same screen. This is the language users see when they launch the library using the **docsearch** command or the library icon in the Help subpanel of the CDE desktop front panel.

i. Click **OK** to complete the configuration.

The documentation search functions on this server should now be ready to use. Any users logged in to this system before configuration finished must log off and then log back in to use the search functions.

5. **Install/register your documentation.**

Before any document can be searched using the Documentation Library Service, it must have an existing index that is registered with the library service. Some applications, such as Web-based System Manager, ship prebuilt documents inside their installation package. When the application is installed, the indexes are automatically registered. Indexes for all operating system documentation are registered during installation.

You can create indexes for your own HTML documents and register them with the library service so they can be searched online. For information about creating and registering indexes, see *AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs*.

Installing the Client

Use this procedure if you want to set up a client of a remote documentation library server. When users on this computer want to read or search online documentation, the request is sent to a remote documentation library server, where the request is handled and the results then sent back to a Web browser on this client.

Note: The search function is not supported in all languages.

To create a documentation library client, complete the following steps:

1. **Install the client software.**

   First, check the list of software that is installed on your system by typing **smit list_installed** on the command line. If the following software is not already installed on your client system, install it now:

   - A Web browser that can display HTML forms. (The Netscape browser is on the base operating system Netscape Products CD.)
   - The Documentation Library Service package (**bos.docsearch**). This package is installed by default with the base operating system. If it has been removed from your system, you need only install the following filesets of the package:
     - **Docsearch Client** fileset
     - **Docsearch Client - CDE Desktop Icons and Actions** fileset (if you are using the CDE Desktop).
     - **Docsearch Support** filesets. When you install the base operating system, it automatically installs support for the language used by this system. However, if you want to be able to use the library service in other languages, you must also install the docsearch support filesets for those languages. The support filesets are stored in the BOS messages packages (**bos.msg** *locale*) and are named **DocSearch CDE Action -** *language* **and DocSearch Common Messages -** *language***. You do not need to install the CDE message set if you are not using the CDE desktop.**

       Note: You must also have the BOS locale (language environment) installed for any language you want to use. If you install the locale *after* the Documentation Library Service is installed on your system, the messages for that language are also automatically installed. However, if the locale was installed *before* the library service (for example, you are updating the operating system), you must manually install the library service messages.

   For instructions on adding language support, use the procedure as shown in step 3 on page 121 of *Installing the Server*.

2. **Configure the Documentation Library Service**.

   To configure the Documentation Library Service using Web-based System Manager:

   a. As root user, start the Web-based System Manager by typing `wsm` on the command line.
   b. Expand the machine name.
   c. Expand **System Environment** in the Navigation area.
   d. From the **System Environment** menu, choose **Settings —> Documentation Server**.
   e. In the Browser command field, type the name of the command that launches the Web browser with a URL. This selection will be the default Web browser for all users of this computer.
      - If you are using the Netscape browser, type `netscape` in the Browser field and continue with the next step.
      - If you are using another browser, you must include any necessary flags that include a URL in the launch command. For example, if launching your Web browser to a specific site requires a **-u** flag, and your launch command looks similar to the following:

YourWebBrowser -u http://www.w3.org

Your entry in the Browser field would be:

YourWebBrowser -u

You do not include the URL itself in the Browser field entry.

Note: Many browsers (for example, Netscape) do not require a flag.

f. Click **OK**.

g. Select **Settings —> Documentation Server** again.

h. Select **Remote server**.

i. Type the name of the documentation server into the **Computer name** field. This documentation server contains the documents that you want this client to be able to search.

j. In the Server port field, type in the port number the Web server software is using. The standard port is 80.

k. Click **OK** to complete the configuration.

l. Close the Web-based System Manager.

To configure the Documentation Library Service using SMIT:

a. On the client, change to root user. On a command line, type `smit web_configure`.

b. Select **Change/Show Default Browser**. In the ***Default browser LAUNCH COMMAND** field, type the command that sets the default browser for all users on this computer. Include any flags that are required when a URL is included in the command. If you want to have the default browser to open to a specific URL, also add the URL after the command to open the browser. For example, if you want your browser to open with the **http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base** page, type the following in the ***Default browser LAUNCH COMMAND** field:

YourWebBrowser flag http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base

Note: Many browsers (for example, Netscape) do not require a flag.

c. Return to the Web configuration main menu. Select **Change Documentation and Search Server**.

d. On the Documentation Search Server LOCATION screen, click **List** and select **Remote computer** for server location. Click **OK**.

e. On the next screen, enter the name of the remote documentation search server you want the client to send its search requests to in the **NAME of remote documentation server** field. You can type a name or an IP address.

If the Web server on the remote server is set to use some port other than the standard port 80, enter that port number.

f. Click **OK** to configure your client system. When it is finished, a `Documentation server configuration completed!` message displays at the bottom of the results panel.

The documentation search functions on this client are now ready to use. Any users logged in to this client before configuration finished must log off and then log back in to use the search functions.

Installing the Online Documentation

Note: It is not necessary to install the operating system's online documentation onto your server. You can access all of the documentation through the IBM eServer pSeries Information Center on the Internet at the following URL:

http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base

However, if you use the Documentation Library Service to search any online documentation, it only searches the registered documentation that is installed. It does not search the documentation at the above Web site.

Documentation CD

The following types of documentation are located on the *documentation CD*:

- User guides
- System management guides
- Application programmer guides
- All commands reference volumes
- Files reference
- Technical reference volumes used by application programmers

The operating system and related products documentation on this CD is designed for use with an HTML 3.2 Web browser, such as the Netscape browser that is shipped with the operating system.

> **Note:** The *documentation CD* can be either installed or mounted. If you mount the CD, the entire CD is mounted. You cannot install part of the CD and mount the remainder.
>
> While mounting the CD saves on the amount of hard disk space used, it requires the CD be kept in the CD-ROM drive at all times. Also, searching the documentation from the CD-ROM drive can be significantly slower (in some cases, up to ten times slower) than searching the information if it is installed on a hard disk.

You can use either the Web-based System Manager or System Management Interface Tool (SMIT) to install the documentation. Both tools create a temporary mount point for the CD. Also, you can install the entire CD or only selected documentation. Some documentation might have been previously installed with the operating system or other licensed products.

Run the **lslpp** command to determine which packages and filesets are already installed on your system as follows:

```
lslpp -L | grep bos.html
```

Output similar to the following displays:

```
  bos.html.en_US.manage_gds.install
  bos.html.en_US.manage_gds.manage_bos
  bos.html.en_US.manage_gds.manage_commo
  bos.html.en_US.manage_gds.printers
  bos.html.en_US.topnav.navigate
```

Related Information

- Review readme files for late-breaking news. For information on how to view readme files, refer to “Viewing Readme Files” on page ix.
- Begin configuring and using the document search application. See “Using the Documentation Library Service” on page 118.

Chapter 12. Creating and Installing System Backups

This chapter provides information on creating and installing system backups. The chapter includes the following sections:

- “Creating System Backups”
- “Installing System Backups” on page 139

Notes:

1. References to CD in this chapter also apply to DVD.
2. AIX 5.2 provides the **cdromd** CD and DVD automount facility, which is included in the **bos.cdmount** fileset. To determine if the **cdromd** daemon is enabled on your system, run the following command:

   ```
   # lssrc -s cdromd
   ```

   The **cdromd** daemon can interfere with scripts, applications, or instructions that attempt to mount the CD or DVD device without first checking to see if the device is already enabled. A resource or device busy error occurs in such a condition. Use the **cdumount** or **cdeject** command to unmount the device. Then mount the device as specified in the program or instructions. Alternatively, use the **cdcheck -m** or **mount** command to determine the current mount point of the device. For further information, see the **cdromd** command documentation in the *AIX 5L Version 5.2 Commands Reference*.

   The installation code allows for this automatic mounting. If **cdromd** is enabled and the **mkcd** command is run, the CD-R or DVD-RAM is ejected after the image is completed. If you do not want to have the media ejected, then the **cdromd** daemon should be put in the inoperative state with the following command:

   ```
   # stopsrc -s cdromd
   ```

Creating System Backups

This section describes how to create and verify a bootable backup copy, or *mksysb image*, of your root volume group and how to make separate backup copies of user volume groups.

The *root volume group* is a hard disk or group of disks that contains:

- Startup files
- Base Operating System (BOS)
- System configuration information
- Optional software products

A *user volume group*, also called the *nonroot volume group*, typically contains data files and application software.

A system backup does the following:

- Contains a working copy of your system. In the event your system data becomes corrupted, you can use this information to restore your system to working order.
- Allows you to transfer installed and configured software from one system to others. You can use the Web-based System Manager or SMIT to make a backup image of the root volume group or user volume groups.

A backup transfers the following configurations from the source system to the target system:

- **rootvg** volume group information
- Paging space information
- Logical volume information

© Copyright IBM Corp. 2002, 2004

- Placement of logical partitions (if creating map files has been selected in the Web-based System Manager or SMIT).

  Note: The use of map files is not recommended if you plan to reinstall the backup to target systems other than the source system, or the disk configuration of the source system is to be changed before reinstalling the backup.

Using the Web-based System Manager or the SMIT backup menu lets you preserve configuration information, thus avoiding some of the configuring tasks normally required after restoring a system backup. A backup preserves the configuration if the following are true:

- The target system has the same hardware configuration as the source system.

  AND

- The target disk has enough space to hold the backup image.

The Web-based System Manager and SMIT both use the **mksysb** command to create a backup image, stored either on CD, DVD, tape, or in a file. If you choose CD, DVD, or tape, the backup program by default writes a *boot image*, which makes the medium suitable for installing. For more information, see "System Backup to CD-R, DVD-R, or DVD-RAM" on page 131.

Install All Device and Kernel Support Before the Backup is Created

In AIX 5.2, all devices and kernels are installed by default when performing a base operating system installation. This allows you to create a system backup that contains all devices and kernel types. Because the system backup contains all the devices and kernel support, the system backup can be used to install another system without the need for the AIX product media. This option is available in the Install Options menu in the BOS installation menus.If you change the default (**yes**) to **no**, only the devices and kernel type for your system configuration will be installed.

This value is read from the `ALL_DEVICES_KERNELS` field in the **/var/adm/ras/bosinst.data** file on the product media that you used to boot the system.

After the system is installed, you can check if all the devices and kernel types have been installed on the system as follows:

```
# grep ALL_DEVICES_KERNELS bosinst.data
```

Output similar to the following displays:

```
ALL_DEVICES_KERNELS = yes
```

For more information about the **bosinst.data** file, refer to Chapter 8, "Customized BOS Installations," on page 67.

Prerequisites for Creating Backups

Before creating system backups, complete the following prerequisites:

- Be sure you are logged in as root user.
- Consider altering passwords and network addresses if you use a backup to make master copies of a source system. Copying passwords from the source to a target system can create security problems. Also, if network addresses are copied to a target system, duplicate addresses can disrupt network communications.
- Mount all file systems you want to back up. The **mksysb** command backs up mounted JFS (journaled file systems) and JFS2 (enhanced journaled file systems) in the **rootvg**. Refer to the **mount** command for details.

  Note: The **mksysb** command does not back up file systems mounted across an NFS network.

- Unmount any local directories that are mounted over another local directory.

  This backup procedure backs up files twice if a local directory is mounted over another local directory in the same file system. For example, if you mount **/tmp** over **/usr/tmp**, the files in the **/tmp** directory are then backed up twice. This duplication might exceed the number of files a file system can hold, which can cause a future installation of the backup image to fail.
- Use the **/etc/exclude.rootvg** file to list files you do not want backed up.
- Make at least 8.8 MB of free disk space available in the **/tmp** directory. The **mksysb** command requires this working space for the duration of the backup.

  Use the **df** command, which reports in units of 512-byte blocks, to determine the free space in the **/tmp** directory. Use the **chfs** command to change the size of the file system, if necessary.

  For example, the following command adds 12 MB of disk space to the **/tmp** directory of a system with 4 MB partitions:

  ```
  chfs -a size=+24000 /tmp
  ```
- All hardware must already be installed, including external devices, such as tape and CD-ROM drives.
- The **bos.sysmgt.sysbr** fileset in the BOS System Management Tools and Applications software package must be installed. The **bos.sysmgt.sysbr** fileset is automatically installed in AIX 5.2. To determine if the **bos.sysmgt.sysbr** fileset is installed on your system, type:

  ```
  lslpp -l bos.sysmgt.sysbr
  ```

  If your system has the **bos.sysmgt.sysbr** fileset installed, continue with one of the following procedures:
  - "Creating a Root Volume Group Backup to Tape or File"
  - "System Backup to CD-R, DVD-R, or DVD-RAM" on page 131
  - "User Volume Group Backup" on page 135

  If the **lslpp** command does not list the **bos.sysmgt.sysbr** fileset, install it before continuing with the backup procedure. Refer to Chapter 10, "Optional Software Products and Service Updates," on page 81 for instructions, or enter the following command:

  ```
  installp -agqXd device bos.sysmgt.sysbr
  ```

  where *device* is the location of the software; for example, `/dev/cd0` for CD-ROM drive.

Creating a Root Volume Group Backup to Tape or File

You can use either the Web-based System Manager or SMIT to create a system backup to be stored to tape or in a file.

For instructions on how to back up to CD or DVD, see "System Backup to CD-R, DVD-R, or DVD-RAM" on page 131.

To create a root volume group backup:

- Use the Web-based System Manager **Backup and Restore** application and select **Back up the system**.

  OR
- Use the following SMIT procedure:
  1. Enter the **smit mksysb** fast path.
  2. In the Back Up the System menu, make the following selections:
     - Select which medium you want to use in the **Backup DEVICE or File** field. If you want to create a bootable backup, the medium must be tape or CD. See "System Backup to CD-R, DVD-R, or DVD-RAM" on page 131 for more information. Then, select the appropriate option below:

       TAPE Press the F4 key to list available devices and highlight the device name.

       FILE Enter a full path and file name in the entry field.

– If you want to create map files, select **yes** in the **Create Map Files?** field.

  Map files match the physical partitions on a drive to its logical partitions. When installing from a backup image, the BOS installation program uses map files to position the logical volumes on the target drive in the same partitions they were on in the source system. If you do not create map files, the installation program relies on the logical volume manager (LVM) to determine placement for the logical volumes. For more information, see Using Map Files for Precise Allocation in *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*.

  Note: If you plan to reinstall the backup to target systems other than the source system, or if the disk configuration of the source system might change before reinstalling the backup, do not create map files.

– To exclude certain files from the backup, select **yes** in the **Exclude Files** field, then create an **/etc/exclude.rootvg** file with an ASCII editor, and enter the file names that you do not want included in your system backup image. You can use patterns for the file names that conform to the pattern matching conventions of the **grep** command. For example, to exclude all the contents of the directory called `scratch`, put the following line in the exclude file:

  ```
  /scratch/
  ```

  For another example, exclude the contents of the directory called **/tmp** and avoid excluding any other directories that have **/tmp** in the pathname by adding the following line to the exclude file:

  ```
  ^./tmp/
  ```

  Note: All files are backed up relative to the current working directory. This directory is represented by a **.** (dot character). To exclude any file or directory for which it is important to have the search match the string at the beginning of the line, use a **^** (caret character) as the first character in the search string, followed by a **.** (dot character), and then followed by the file name or directory to be excluded.

  If the file name or directory being excluded is a substring of another file name or directory, use **^.** (caret character followed by dot character) to indicate that the search should begin at the beginning of the line and/or use **$** (dollar sign character) to indicate that the search should end at the end of the line.

– To list each file as it is backed up, select **yes** in the **List files as they are backed up?** field. Otherwise, you see a percentage-completed progress message while the backup is created.

– If you modified the **image.data** file and do not want a new one created, select **no** for **Generate new /image.data file?**. (The **image.data** file contains information about the sizes of all the file systems and logical volumes in your rootvg.)

– If you are creating a bootable tape and you want to expand the system **/tmp** file system (if required by the backup program), select **yes** for **EXPAND /tmp if needed?**.

– If the tape drive you are using provides packing (or compression), set the **Disable software packing of backup?** field to `yes`.

– If you chose tape as the backup medium, either leave the default in the **Number of BLOCKS to write in a single output** field or enter a different number.

– If you chose file as the backup medium, press Enter. If you chose tape as the backup medium, insert the first blank backup tape into the drive and press Enter.

3. The COMMAND STATUS screen displays, showing status messages while the system makes the backup image.

   If you chose tape as the backup medium, the system might prompt you to insert the next tape during the backup by displaying a message similar to the following:

   ```
   Mount next Volume on /dev/rmt0 and press Enter.
   ```

   If this message displays, remove the tape and label it, including the BOS version number. Then insert another tape and press Enter.

   When the backup process finishes, the **COMMAND:** field changes to **OK**.

4. When the backup completes, press F10 to exit SMIT.
5. If you selected tape as the backup medium, remove the last tape and label it. Write-protect the backup tapes.
6. Record any backed-up root and user passwords. Remember that these passwords become active if you use the backup to either restore this system or install another system.

You have created the backup of your root volume group (rootvg). If you created bootable tapes, you can use these tapes to start your system if for some reason you cannot boot from hard disks.

System Backup to CD-R, DVD-R, or DVD-RAM

Creating a backup on CD-R, DVD-R, or DVD-RAM media is similar to making a backup tape for your personal use, but with some noticeable differences. For DVD media, the following formats for creating backups are available:

- ISO9660 CD format, which is available for DVD-R/DVD-RAM media.
- Universal Disk Format (UDF), which is available for DVD-RAM media. For information about creating a backup to DVD-RAM using UDF, see “DVD-RAM and Universal Disk Format” on page 134.

Note: For information about CD-R, DVD-R, or DVD-RAM drives and CD-R, DVD-R, or DVD-RAM creation software, refer to the following readme file:

```
/usr/lpp/bos.sysmgt/mkcd.README.txt
```

Both Web-based System Manager and SMIT use the **mkcd** command, which calls the **mksysb** or **savevg** command, if needed.

For system backups, the CDs or DVDs can be created as:

- Non-bootable CDs or DVDs
- Bootable CDs or DVDs

In AIX 5.2, a bootable system backup contains the **chrp** boot image and all the device and kernel packages necessary to install a system. A backup CD or DVD can be used to install (clone) a large number of machines, which is convenient when each machine in the system environment needs to have the same image installed.

Note: It is possible that a backup CD or DVD would not boot all machines of the same type because not every machine has the same hardware configuration. Depending on what packages were made available during the creation of the backup, the backup might not have all the necessary packages to boot an individual system. Most required packages for systems are present on the BOS AIX 5.2 media.

Web-based System Manager and SMIT interfaces are available for the **mkcd** command. Online help can guide you through the required steps.

Hardware and Software Requirements

The **mkcd** command requires that you already have the software installed to create a CD/DVD file system in Rock Ridge format and to *burn* or write the CD/DVD. The GNU versions of the **cdrecord** and **mkisofs** commands are installed with a BOS installation. Hardware and software that has been tested with this command includes the following:

Software	Hardware
GNU and Free Software Foundation, Inc. readcd command version 1.9 mkisofs command version 1.13	Matsushita LF-D291 DVD-RAM

Using the mkcd Command

To run the **mkcd** command, you need extra working space. A separate file system or directory is required for each of the following:

- Storing a **mksysb** or **savevg** image
- Storing the CD or DVD file system contents
- Storing the CD or DVD images before they are recorded

The **mkcd** command creates the following file systems if they are not already present or if alternative file systems or directories have not been specified:

/mkcd/mksysb_image
: Space requirement depends on the size of the **mksysb** image that is to be created. The **mkcd** command attempts to calculate this space and verify that adequate space is available before starting to create the **mksysb** image.

  Note: When the **mkcd** command calculates the space requirements needed for the **/mkcd/mksysb_image** directory, it also adds the space used by the excluded files (**/etc/exclude.rootvg**). It is therefore possible that the **mkcd** command might not be able to create the **/mkcd/mksysb_image** directory.

/mkcd/cd_fs
: Requires 645 megabytes (up to 4.38 GB for DVD)

/mkcd/cd_images
: Requires at least 645 megabytes (up to 4.38 GB for DVD) of space. If the **-R** or **-S** flags are used to specify not removing the images and there are multiple volumes required, more space must be provided.

The space used in these file systems is only temporary (unless the **-R** or **-S** flag is specified to save the images). If the **mkcd** command creates the file systems, it also removes them. Each file system or directory might require over 645 megabytes (up to 4.38 GB for DVD).

If your machine does not have sufficient space, you can use NFS to mount some space from another server system; however, the file systems must be writable. You can create a **/mkcd** file system that is very large (1.5 GB for CD or 9 GB for DVDs). The **/mkcd** file system can then be mounted onto the clients when they want to create a backup CD or DVD for their systems. When creating very large backups (larger than 2 GB) with the **mkcd** command, the file system must be large-file enabled and the **ulimit** values must be set to `unlimited`.

The **mkcd** command with the **-L** flag allows the creation of ISO9660 DVD-sized images. The **mkcd** command with the **-U** flag allows the creation of UDF DVD images.

Creating a Root Volume Group Backup on CD or DVD with the ISO9660 Format

You can use Web-based System Manager or SMIT to create a root volume group backup on CD or DVD with the ISO9660 format, as follows:

- Use the Web-based System Manager **Backup and Restore** application and select **System backup wizard method**. This method lets you create bootable or non-bootable backups on CD-R, DVD-R, or DVD-RAM media.

  OR
- To create a backup to CD, use the **smit mkcd** fast path.
- To create a backup to DVD, use the **smit mkdvd** fast path and select **ISO9660 (CD format)**.

The following procedure shows you how to use SMIT to create a system backup to CD. (The SMIT procedure for creating a system backup to an ISO9660 DVD is similar to the CD procedure.)

1. Type the **smit mkcd** fast path. The system asks whether you are using an existing **mksysb** image.
2. Type the name of the CD-R device. (This can be left blank if the **Create the CD now?** field is set to no.)
3. If you are creating a **mksysb** image, select **yes** or **no** for the **mksysb** creation options, **Create map files?** and **Exclude files?**. Verify the selections, or change as appropriate.

   The **mkcd** command always calls the **mksysb** command with the flags to extend **/tmp**.

   You can specify an existing **image.data** file or supply a user-defined **image.data** file. See step 16 on page 134.
4. Enter the file system in which to store the **mksysb** image. This can be a file system that you created in the **rootvg**, in another volume group, or in NFS-mounted file systems with read-write access. If this field is left blank, the **mkcd** command creates the file system, if the file system does not exist, and removes it when the command completes.
5. Enter the file systems in which to store the CD or DVD file structure and final CD or DVD images. These can be file systems you created in the rootvg, in another volume group, or in NFS-mounted file systems. If these fields are left blank, the **mkcd** command creates these file systems, and removes them when the command completes, unless you specify differently in later steps in this procedure.
6. If you did not enter any information in the file systems' fields, you can select to have the **mkcd** command either create these file systems in the **rootvg**, or in another volume group. If the default of **rootvg** is chosen and a **mksysb** image is being created, the **mkcd** command adds the file systems to the exclude file and calls the **mksysb** command with the **-e** exclude files option.
7. In the **Do you want the CD or DVD to be bootable?** field, select **yes** to have a boot image created on the CD or DVD. If you select **no**, you must boot from a product CD at the same *version.release.maintenance* level, and then select to install the system backup from the system backup CD.
8. If you change the **Remove final images after creating CD?** field to **no**, the file system for the CD images (that you specified earlier in this procedure) remains after the CD has been recorded.
9. If you change the **Create the CD now?** field to **no**, the file system for the CD images (that you specified earlier in this procedure) remains. The settings that you selected in this procedure remain valid, but the CD is not created at this time.
10. If you intend to use an Install bundle file, type the full path name to the bundle file. The **mkcd** command copies the file into the CD file system. You must have the bundle file already specified in the **BUNDLES** field, either in the **bosinst.data** file of the **mksysb** image or in a user-specified **bosinst.data** file. When this option is used to have the bundle file placed on the CD, the location in the **BUNDLES** field of the **bosinst.data** file must be as follows:

    /../usr/sys/inst.data/user_bundles/*bundle_file_name*
11. To place additional packages on the CD or DVD, enter the name of the file that contains the packages list in the **File with list of packages to copy to CD** field. The format of this file is one package name per line.

    If you are planning to install one or more bundles after the **mksysb** image is restored, follow the directions in the previous step to specify the bundle file. You can then use this option to have packages listed in the bundle available on the CD. If this option is used, you must also specify the location of installation images in the next step.
12. Enter the location of installation images that are to be copied to the CD file system (if any) in the **Location of packages to copy to CD** field. This field is required if additional packages are to be placed on the CD (see the previous step). The location can be a directory or CD device.
13. You can specify the full path name to a customization script in the **Customization script** field. If given, the **mkcd** command copies the script to the CD file system. You must have the **CUSTOMIZATION_FILE** field already set in the **bosinst.data** file in the **mksysb** image or else use a user-specified **bosinst.data** file with the **CUSTOMIZATION_FILE** field set. The **mkcd** command copies this file to the **RAM** file system. Therefore, the path in the **CUSTOMIZATION_FILE** field must be as follows:

    /../*filename*

14. You can use your own **bosinst.data** file, rather than the one in the **mksysb** image, by typing the full path name of your **bosinst.data** file in the **User supplied bosinst.data file** field.
15. To turn on debugging for the **mkcd** command, set **Debug output?** to **yes**. The debug output goes to the **smit.log**.
16. You can use your own **image.data** file, rather than the **image.data** file in the **mksysb** image, by typing the full path name of your **image.data** file for the **User supplied image.data file** field.

DVD-RAM and Universal Disk Format

This section provides information on the Universal Disk Format (UDF) and how to create system backups using DVD-RAM media and UDF.

UDF allows you to manipulate files directly on the DVD-RAM media. The system backup image is an archived file composed of many files that cannot be manipulated. However, the installation packages and any files that are not contained in the backup image, can be directly manipulated on the DVD-RAM. After the DVD is mounted the files can be changed by using an editor or new files can be copied to the DVD using the various copy and restore commands such as the **cp**, **mv**, **restore** commands.

With UDF and DVD-RAM, system space is only needed for the backup image. A high-level description of the UDF backup process is as follows:

1. Create a backup of a volume group to a file (archive) on a hard disk containing enough space to hold the backup image.
2. Populate UDF with files needed to boot and install a system.
3. Copy backup to DVD-RAM media.

The **mkcd** command with the **-U** flag is used to create a UDF file system on the DVD-RAM.

UDF allows for the possibility of changing files directly on the DVD-RAM media, such as a **bosinst.data** file and **image.data** or **vgname.data** file. Without UDF for example, to add a user-defined **bosinst.data** file to a backup image, you must restore the backup image to a location, add the file, and then back up the files again.

Or, you had to create a supplemental diskette containing the changed **bosinst.data** file, and use the supplemental diskette in conjunction with the backup. However, some system configurations might not provide diskette drives, making this procedure more difficult.

Creating a Root Volume Group Backup on DVD-RAM with Universal Disk Format: To create a root volume group backup on DVD-RAM with UDF, do the following:

- Use the Web-based System Manager **Backup and Restore** application and select **System backup wizard method**. This method lets you create bootable or non-bootable backups on DVD-RAM media.

  OR
- Use SMIT to create a backup to DVD-RAM with UDF, as follows:
  1. Enter the **smit mkdvd** fast path. The system asks whether you are using an existing **mksysb** image.
  2. Select **UDF (Universal Disk Format)**.
  3. Enter the name of the DVD-RAM device.
  4. If you are creating a **mksysb** image, select **yes** or **no** for the mksysb creation options. The options are as follows:
     - **Create map files?**
     - **Exclude files?**

     The **mkcd** command always calls the **mksysb** command with the flags to extend **/tmp**.

     You can specify an existing **image.data** file or supply a user-defined **image.data**. See step 14 on page 135.

5. Enter the file system or directory in which to store the **mksysb** image. This can be a file system you created in the **rootvg**, in another volume group, or in NFS mounted file systems with read-write access. If left blank, the **mkcd** command creates the file system and removes it when the command completes.
6. If you did not enter information in the file system field, you can select to have the **mkcd** command either create these file systems in the **rootvg**, or in another volume group. If the default of **rootvg** is chosen and a **mksysb** image is being created, the **mkcd** command adds the file systems to the exclude file and calls the **mksysb** command with the exclude files option **-e**.
7. Do you want the DVD to be bootable? If you select **no**, you must boot from a product CD at the same *version.release.maintenance* level, and then select to install the system backup from the system backup DVD.
8. If you intend to use an Install bundle file, enter the full path name to the bundle file. The **mkcd** command copies the file into the DVD file system. You must have the bundle file already specified in the **BUNDLES** field, either in the **bosinst.data** file of the **mksysb** image or in a user-specified **bosinst.data** file. When this option is used to have the bundle file placed on the DVD, the location in the **BUNDLES** field of the **bosinst.data** file must be as follows:

   `/../usr/sys/inst.data/user_bundles/`*`bundle_file_name`*
9. Additional packages can be placed on the CD by entering the name of the file that contains the packages list in the **File with list of packages to copy to DVD** field. The format of this file is one package name per line.

   If you are planning to install one or more bundles after the **mksysb** image is restored, follow the directions in the previous step to specify the bundle file. You can then use this option to have packages listed in the bundle available on the DVD. If this option is used, you must also specify the location of installation images in the next step.
10. Enter the location of installation images that are to be copied to the CD file system (if any) in the **Location of packages to copy to DVD** field. This field is required if additional packages are to be placed on the DVD (see the previous step). The location can be a directory or DVD device.
11. You can specify the full path name to a customization script in the **Customization script** field. If given, the **mkcd** command copies the script to the CD file system. You must have the **CUSTOMIZATION_FILE** field already set in the **bosinst.data** file in the **mksysb** image or use a user-specified **bosinst.data** file with the **CUSTOMIZATION_FILE** field set. The **mkcd** command copies this file to the **RAM** file system. Therefore, the path in the **CUSTOMIZATION_FILE** field must be as follows:

    `/../`*`filename`*
12. You can use your own **bosinst.data** file, rather than the one in the **mksysb** image, by entering the full path name of your **bosinst.data** file in the **User supplied bosinst.data file** field.
13. To enable debugging for the **mkcd** command, set **Debug output?** to **yes**. The debug output goes to the **smit.log**.
14. You can use your own **image.data** file, rather than the **image.data** file in the **mksysb** image, by entering the full path name of your **image.data** file for the **User supplied image.data file** field.

User Volume Group Backup

The **savevg** command provides the ability to create a user-volume group backup to a CD, DVD, tape, or file. The **savevg** command finds and backs up all files belonging to a specified volume group. The volume group must be varied-on, and the file systems must be mounted.

This user backup contains a copy of a non-rootvg volume group, and is useful for volume groups that contain user data.

The **savevg** command uses a data file created by the **mkvgdata** command. The data file created is as follows:

`/tmp/vgdata/`*`vgname`*`/`*`vgname`*`.data`

The *vgname.data* file contains information about a user volume group. The **savevg** command uses this file to create a backup image that can be used by the **restvg** command to re-create the user volume group.

The **savevg** command with the **-r** flag is used to back up only a user-volume group's logical volume structure information. The data needed to list backup properties is also backed up. The **-r** flag runs the **mkvgdata** command for the volume group specified to create a **vgname.data** file. The **-r** flag backs up only the **vgname.data** file, any map files, and the **backup.data** file. The backup image that is created is used with the **restvg -r** command option to create only the volume group, logical volumes, and file system information contained in the file, without restoring any data. For example, to back up only the `paul` user volume group's structure information to the `/vg_backup/paul_vg_data` file, type the following:

```
savevg -r -f /vg_backup/paul_vg_data paul
```

You can also use the **mkcd** command to create a user volume group backup to CD or DVD. The **mkcd** command saves one volume group at a time to a CD or DVD.

The **mkcd** command with the **-L** flag allows the creation of ISO9660 DVD sized images. The **mkcd** command with the **-U** flag allows the creation of UDF DVD images.

If your **rootvg** image and **savevg** image are small enough to fit on one CD, you can save them both by using the **-I** (stacklist) and **-z** (customization_script) flags. The **-I** flag gives a list of images to copy to the CD. The **-z** flag lets you create a script to restore **savevg** backups. For example, if you make a copy of a non-rootvg volume group ahead of time, and then write a script that calls the **restvg** command, your non-rootvg volume group would be restored to `hdisk2` at the end of the installation of **rootvg**, as shown by the following command:

```
restvg -d /SPOT/installp/ppc/savevg_image hdisk2
```

This procedure is recommended *only* if you know you want to restore the non-rootvg volume group every time you install. Otherwise, you might just want to store it on the CD/DVD, then use **restvg** to restore it after reboot. The **restvg** command can restore from CD or DVD if the name of the image is *savevg_image*. If you save the non-rootvg backup on a CD or DVD with a different file name, you can insert that CD or DVD and use the full path to the file name as the device for the **restvg** command.

Use either Web-based System Manager or SMIT to back up user volume groups to CD or DVD.

Creating a User Volume Group Backup Using Web-based System Manager

Use Web-based System Manager and select **System backup wizard method**. You can create a non-bootable CD or DVD that contains only a volume group backup image of a user volume group.

Creating a User Volume Group Backup Using SMIT

Use SMIT to create a backup image of a user volume group, as follows:

1. To back up a user volume group to tape or file using SMIT, type `smit savevg` on the command line. Back up a user volume group to CD by typing `smit savevgcd` on the command line. Back up a user volume group to DVD by typing `smit savevgdvd` on the command line.
2. When the Save a Volume Group screen displays, use the steps for backing up the root volume group as a guide for backing up user volume groups. There is one exception to this procedure. If you want to exclude files in a user volume group from the backup image, create a file named **/etc/exclude.***volume_group_name*, where *volume_group_name* is the name of the volume group you want to backup.
3. If you exclude files, edit the **/etc/exclude.***volume_group_name* file and enter the patterns of file names that you do not want included in your backup image. The patterns in this file are input to the pattern-matching conventions of the **grep** command to determine which files are excluded from the backup.

Backup Options

After you have a system backup or a user volume group backup, you may want to verify the backup or list information about the backup image. This section provides information on the operations you can perform on a backup image. The commands used to perform these operations are the **lsmksysb** command for system backups, and the **lssavevg** command for user volume groups. Using the **lsmksysb** command or the **lssavevg** command, you can perform the following operations:

- "Preview Information About a Backup"
- "Verifying a System Backup"
- "View the Backup Log"
- "List Information About Filesets in a System Image" on page 138
- "List Files in a System Image" on page 138

Preview Information About a Backup

The preview option allows you to view volume group information, the date and time the backup was made, and the level of AIX.

You can use the **lsmksysb** command or the **lssavevg** command with the **-l** option to preview a backup image. For example, to preview a system backup file called **/tmp/mybackup**, type the following:

```
# lsmksysb -l -f /tmp/mybackup
```

Output similar to the following displays:

```
VOLUME GROUP:           rootvg
BACKUP DATE/TIME:       Mon Jul 29 22:03:27 CDT 2002
UNAME INFO:             AIX va08 2 5 000974AF4C00
BACKUP OSLEVEL:         5.2.0.0
none
MAINTENANCE LEVEL:      none
BACKUP SIZE (MB):       1408
SHRINK SIZE (MB):       1242

rootvg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
hd5                 boot       1     1     1    closed/syncd  N/A
hd6                 paging     16    16    1    open/syncd    N/A
hd8                 jfs2log    1     1     1    open/syncd    N/A
hd4                 jfs2       1     1     1    open/syncd    /
hd2                 jfs2       21    21    1    open/syncd    /usr
hd9var              jfs2       1     1     1    open/syncd    /var
hd3                 jfs2       1     1     1    open/syncd    /tmp
hd1                 jfs2       1     1     1    open/syncd    /home
hd10opt             jfs2       1     1     1    open/syncd    /opt
fslv00              jfs2       31    31    1    open/syncd    /export/nim
fslv01              jfs2       1     1     1    open/syncd    /tftpboot
```

To preview a backup image in the SMIT, use the **lsbackupinfo** fast path.

Verifying a System Backup

To list the contents of a **mksysb** image on tape or CD, you can use either Web-based System Manager (type `wsm` on the command line, then choose the Backup and Restore application) or SMIT (type `smit lsmksysb` on the command line). The listing verifies most of the information on the tape or CD, but does not verify that the backup media can be booted for installations. The only way to verify that the boot image(s) on a **mksysb** tape or CD function properly is by booting from the media.

View the Backup Log

You can view the backup log that is created each time a volume group is backed up. The log file contains information on previous volume group and system backups.

You can use the **lsmksysb** command or the **lssavevg** command with the **-B** option to view the backup log file. Type:

```
# lsmksysb -B
```

Output similar to the following displays:

```
#Device;Command;Date;Shrink Size;Full Size;Maintenance Level
/export/mksysb/generic_sysb;"mksysb -X -e /export/mksysb/generic_sysb";M
on Jul 29 22:11:17 CDT 2002;1242;1408;
/export/mksysb/generic_sysb;"mksysb -X -e /export/mksysb/generic_sys
b";Tue Jul 30 16:38:31 CDT 2002;2458;2720;
```

To view the backup log in the SMIT, select **View the Backup Log** in the System Backup Manager menu.

List Information About Filesets in a System Image

You can view the filesets installed in a system backup using the **lsmksysb** command with the **-L** option. For example, to view the filesets installed in a system backup, type the following:

```
# lsmksysb -L -f generic_sysb
```

Output similar to the following displays:

```
Fileset                      Level  State      Description
----------------------------------------------------------------------------
Path: /usr/lib/objrepos
IMNSearch.bld.DBCS         2.4.0.0  COMMITTED  NetQuestion DBCS Buildtime
Modules
.
.
.
bos.terminfo.wyse.data     5.2.0.0  COMMITTED  Wyse Terminal Definitions
bos.txt.spell.data         5.2.0.0  COMMITTED  Writer's Tools Data
bos.txt.tfs.data           5.2.0.0  COMMITTED  Text Formatting Services Data
```

To view the filesets installed in a system backup in SMIT, use the **lslppbackup** fast path.

List Files in a System Image

You can list all the files and file sizes in a backup using the **lsmksysb** command or the **lssavevg** command. For example, to view the files and file sizes in a system backup, type the following:

```
# lsmksysb -f generic_sysb
```

Output similar to the following displays:

```
.
.
.
      353218 ./smit.log
         252 ./smit.script
           0 ./tftpboot
           0 ./tmp
           5 ./u
          21 ./unix
The total size is 629313078 bytes.
```

To view the files and file sizes in a user-volume group backup that contains only the user-volume group's structure information, type the following:

```
# lssavevg -f /vg_backup/paul_vg_data -l
```

Output similar to the following displays:

```
VOLUME GROUP:           paul
BACKUP DATE/TIME:       Fri Feb 28 12:30:34 CST 2003
UNAME INFO:             AIX va06 2 5 000917184C00
```

```
BACKUP OSLEVEL:         5.2.0.10
MAINTENANCE LEVEL:      52010
BACKUP SIZE (MB):       0
SHRINK SIZE (MB):       0
VG DATA ONLY:           yes

paul:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
```

To view the files and file sizes in a system backup in SMIT, use the **lsmksysb** fast path.

Installing System Backups

This chapter describes how to install the Base Operating System (BOS) using a system backup image, also called a *mksysb image*. You can use a system backup to restore a corrupted operating system. But installing a system from a backup can also reduce (or even eliminate) repetitive installation and configuration tasks. For example, you can use a backup to transfer optional software installed on the *source* system (the machine from which you created the backup copy), in addition to the basic operating system. Also, the backup image can transfer many user configuration settings to the *target* system (a different machine on which you are installing the system backup).

The procedure you use to install from a system backup depends on whether you are installing on the source or target system and which interface you want to use:

- "Cloning a System Backup" on page 140 contains the procedure to install a system backup on a target machine to propagate a consistent operating system, optional software, and configuration settings.
- "Installing a System Backup on the Source Machine" on page 140 contains the Web-based System Manager and SMIT procedures to reinstall an operating system onto the same machine from which you created the backup.

You can install a system from a backup image that is stored on tape or CD, or in a file. If you want to install a backup stored in a directory on your network installation server, refer to "Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation)" on page 197.

The procedures to install from backup operate either in prompted or nonprompted mode, depending on conditions set in the **/bosinst.data** file and on compatibility between the backup image and the installed machine. See Chapter 8, "Customized BOS Installations," on page 67 for information on how to modify the **bosinst.data** file to preset installation parameters when you create a system backup.

When installing the backup image, the system checks whether the target system has enough disk space to create all the logical volumes stored on the backup. If there is enough space, the entire backup is recovered. Otherwise, the installation halts and the system prompts you to choose additional destination hard disks.

File systems are created on the target system at the same size as they were on the source system, unless the backup image was created with **SHRINK** set to **yes** in the **image.data** file, or you selected **yes** in the BOS Install menus. An exception is the **/tmp** directory, which can be increased to allocate enough space for the **bosboot** command. For more information about setting variables, see the **image.data** file in *AIX 5L Version 5.2 Files Reference*.

When you install a system backup on the source machine, the installation program restores the Object Data Manager (ODM) on that machine. Restoring the ODM allows devices to keep the same number they had on the original system. For example, if you install two ethernet cards in your source machine, first en0 in slot 3 and then en1 in slot 1, the cards are not renumbered if they are detected in reverse order when you install a system backup. When you clone a system backup, the installation program rebuilds the ODM on the target system after installing the image, so devices are renumbered. In both cases, the **rootvg** and all logical volumes have different IDs.

If you reinstall a system backup on the source machine, and the target system does not have exactly the same hardware configuration as the source system, the program might modify device attributes in the following target system files:

- All files in the **/etc/objrepos** directory beginning with ″Cu″
- All files in the **/dev** directory

The settings in the bootlist are not restored. After a system backup restore, the bootlist is reset to the primary boot device.

Cloning a System Backup

With a **mksysb** image, you can clone one system image onto multiple target systems. However, the target systems might not contain the same hardware devices or adapters, or require the same kernel (uniprocessor or microprocessor) as the source system. Beginning in AIX 5.2, all devices and kernels are automatically installed during a BOS installation. As a result, when you create a system backup, the **mksysb** image contains all the device and kernel support. For example, you can create a system backup from *System_A* and install *System_A*'s **mksysb** image onto *System_B* without having to use product media to boot *System_B*. For more information on installing all devices and kernels, see “Install All Device and Kernel Support Before the Backup is Created” on page 128.

Beginning in AIX 5.2, if you are performing a clone installation, device information will not be restored to the target system by default. During a clone installation, the BOS installation process verifies that the **mksysb** image is from the system you are trying to install. If the target system and the **mksysb** image are different, the device information is not recovered. This behavior is determined by the **RECOVER_DEVICES** variable in the **bosinst.data** file. This variable can be set to default, yes, or no. The following list shows the resulting behaviors for each value:

default
No recovery of devices

yes Attempted rebuild of ODM

no No recovery of devices

Note: You can override the default value of **RECOVER_DEVICES** by selecting **yes** or **no** in the Backup Restore menu or by editing the value of the attribute in the **bosinst.data** file.

If the source system does not have the correct passwords and network information, you can make modifications on the target system now. Also, some products (such as graPHIGS) ship device-specific files. If your graphics adapter is different on the target system, verify that the device-specific filesets for graphics-related LPPs are installed.

Installing a System Backup on the Source Machine

You can use Web-based System Manager or SMIT to restore an operating system onto the same machine from which you created the backup. For either interface, the following conditions must be met before beginning the procedure:

- All hardware must already be installed, including external devices, such as tape and CD/DVD-ROM drives.
- Obtain the system key for the lock (if present) on your system unit.
- Obtain your system backup image from one of the following sources:

CD or DVD BOS CDs, created in one of the following ways:

- Using the Web-based System Manager **Backup and Restore** application. Select **System backup to writable CD**.
- Using the SMIT Back Up This System to CD menu.
- From the command line, using the **mkcd** command.

Tape BOS tapes, created in one of the following ways:

- Using the Web-based System Manager **Backup and Restore** application. Select **Back up the system**.
- Using the SMIT Back Up the System to Tape/File menu.
- From the command line, using the **mksysb -i** *Target* command.

Note: If devices were removed from or replaced on the system after the backup was created, their information will be restored when you install a backup. The system shows these devices in a defined state because the ODM from the system at the time of backup is restored instead of rebuilt.

Network The path to your backup image file. For information about installing a backup across a network, refer to "Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation)" on page 197.

Note: Before you begin, select the tape or CD/DVD-ROM drive as the primary boot device. For additional information, refer to the section in your hardware documentation that discusses system management services.

Due to enhancements in the **mksysb** command, you can control how devices are recovered when you install a system backup on the source machine. This behavior is determined by the **RECOVER_DEVICES** variable in the **bosinst.data** file. This variable can be set to default, yes, or no. The following list shows the resulting behaviors for each value:

default
ODM is restored

yes ODM is restored

no No recovery of devices

Note: You can override the default value of **RECOVER_DEVICES** by selecting **yes** or **no** in the Backup Restore menu or by editing the value of the attribute in the **bosinst.data** file.

To use Web-based System Manager:

1. Start the Web-based System Manager by typing `wsm` on the command line as root user.
2. Expand **Software** in the Navigation Area, select **Overview and Tasks**, then select **Reinstall Operating System**.
3. Choose the installation device:
   - Network

     If you choose this option, your machine must either be a configured NIM client, or have access to a NIM environment. If your machine is not a NIM client, the Reinstall Base Operating System wizard leads you through the process. For more information on setting up a NIM environment, see "Using Installation Images to Install the Base Operating System (BOS) on a NIM Client" on page 196.
   - Tape or CD/DVD-ROM
4. Choose **Install a system backup image (mksysb)** as the installation type.
5. Follow the wizard prompts to complete the procedure.

To use SMIT:

1. Verify that your system is shut down. If your machine is currently running, you must power it off now by following these steps:

a. Log in as the root user.
b. Enter the following command:
```
shutdown -F
```
c. If your system does not automatically power off, place the power switch in the Off (0) position.

Attention: Do *not* turn on the system unit until Step 5 on page 143.

2. Turn on all attached external devices. These include:
   - Terminals
   - CD or DVD drives
   - Tape drives
   - Monitors
   - External disk drives

   Turning on the external devices first is necessary so that the system unit can identify them during the startup (boot) process.
3. Insert the installation media into the tape or CD or DVD drive.

   You might find that on certain tape drive units, the tape drive door does not open while the system is turned off. If you have this problem, use the following procedure:
   a. Turn on the system unit.
   b. Insert the boot installation tape (insert Volume 1 if you received more than one volume).
   c. Turn off the system unit and wait for 30 seconds.
4. If you are not using an ASCII terminal, skip to Step 6. If you are using an ASCII terminal, use the following criteria to set the communications, keyboard, and display options.

   Note: If your terminal is an IBM 3151, 3161, or 3164, press the Ctrl+Setup keys to display the Setup Menu and follow the onscreen instructions to set these options. If you are using some other ASCII terminal, refer to the appropriate documents for information about how to set these options. Some terminals have different option names and settings than those listed here.

Communication Options	
Option	**Setting**
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

Keyboard and Display Options	
Option	**Setting**
Screen	normal
Row and Column	24x80
Scroll	jump
Auto LF (line feed)	off
Line Wrap	on
Forcing Insert	line (or both)

Keyboard and Display Options	
Option	**Setting**
Tab	field
Operating Mode	echo
Turnaround Character	CR
Enter	return
Return	new line
New Line	CR
Send	page
Insert Character	space

5. Turn the system unit power switch from Off (0) to On (|). The system begins booting from the backup media. If your system is booting from tape, it is normal for the tape to move back and forth. If your system has an LED display, the three-digit LED should display c31.

   Note: You can boot from production media (tape or CD) if your backup media fails to boot. The initial Welcome screen includes an option to enter a maintenance mode in which you can continue the installation from your backup media. Refer to "Troubleshooting an Installation from a System Backup" on page 159 for more information.

   If you have more than one console, each terminal and directly attached display device (or console) might display a screen that directs you to press a key to identify your system console. A different key is specified for each terminal displaying this screen. If this screen is displayed, then press the specified key *only* on the device to be used as the system console. (The system console is the keyboard and display device used for installation and system administration.) Press a key on only one console.

   Note: If the **bosinst.data** file lists a valid display device for the **CONSOLE** variable, you do not manually choose a system console. Read Chapter 8, "Customized BOS Installations," on page 67 for more information about the **bosinst.data** file.

6. The type of installation that begins is determined by the settings of the **PROMPT** field in the control_flow stanza of the **bosinst.data** file. Use the following criteria to determine the type of installation you will be using:

   `PROMPT = no` Nonprompted Installation. This installation method is used if the backup image is configured to install automatically, without having to respond to the installation program. Go to step 8.

   `PROMPT = yes` Prompted Installation. This installation method is used if you need to use menu prompts to install the backup image. Also, use this installation method if a nonprompted installation halts and the Welcome to Base Operating System Installation and Maintenance screen displays. Go to step 9.

7. A successful nonprompted installation requires no further instructions because the installation is automatic.

   Note: If the backup image holds source system-configuration information that is incompatible with the target system, the nonprompted installation stops and a prompted installation begins.

   The Installing Base Operating System screen displays before the installation starts. The nonprompted installation pauses for approximately five seconds before beginning. After this time, the non-prompted installation continues to completion.

   However, if you decide to interrupt the automatic installation and start a prompted session, type 000 (three zeros) at the terminal and follow the remaining steps in this procedure.

8. The Welcome to the Base Operating System Installation and Maintenance screen displays.

Note: You can view Help information at each screen of this installation process by typing 88.

Choose the **Change/Show Installation Settings and Install** option.

9. The System Backup Installation and Settings displays. This screen shows current settings for the system. An ellipsis follows the disk listed in the first line if there is more than one disk selected.
10. Either accept the settings or change them. For more information on using map files, see “Creating System Backups” on page 127.

    To accept the settings and begin the installation, skip to step 16.

    To change the settings, continue with step 11.
11. Type 1 in the System Backup Installation and Settings screen to specify disks where you want to install the backup image. The Change Disk(s) Where You Want to Install screen displays. This screen lists all available disks on which you can install the system backup image. Three greater-than signs (>>>) mark each selected disk.

    Type the number and press Enter for each disk you choose. Type the number of a selected disk to deselect it. You can select more than one disk.

    Note: You can also specify a supplemental disk by typing 66 and pressing the Enter key for the **Disks not known to Base Operating System Installation** option. This option opens a new menu that prompts for a device support media for the supplemental disk. BOS installation configures the system for the disk and then returns to the Change Disk(s) Where You Want to Install screen.
12. After you have finished selecting disks, press the Enter key.

    The screen that displays after you press the Enter key is dependent on the availability of map files for *all* of the selected disks. The criteria for this is as follows:

    - If one or more selected disks have no maps, BOS installation returns directly to the System Backup Installation and Settings screen. Skip to step 15.
    - If all selected disks have maps, the Change Use Maps Status screen displays, where you choose whether to use maps for installation. Continue with step 14.

    To preserve the placement of files during a future restoration of the backup, you can create map files before backing up a system. Map files, stored in the **/tmp/vgdata/rootvg** directory, match the physical partitions on a drive to its logical partitions. Create map files either with the SMIT Backup the System menu, using Web-based System Manager, or using the **-m** option when you run the **mksysb** command.

    For more information about map files, see Using Map Files for Precise Allocation in *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*.
13. Type either 1 or 2 in the Change Use Maps Status screen to specify whether the installation program is to use maps.

    When you complete this choice, BOS installation returns to the System Backup Installation and Settings screen.
14. Decide whether BOS installation is to shrink file systems on the disks where you install the system. When you choose this option, the logical volumes and file systems within a volume group are re-created to the minimum size required to contain the data. This reduces wasted free space in a file system.

    File systems on your backup image might be larger than required for the installed files. Press the 2 key to toggle the **Shrink File Systems** option between **Yes** and **No** in the System Backup Installation and Settings screen. The default setting is **No**.

    Note: Shrinking the file system disables the use of maps.
15. Type 0 to accept the settings in the System Backup Installation and Settings screen.

    The Installing Base Operating System screen displays the rate of completion and duration.

    If you specified a supplemental disk in step 12, an untitled screen temporarily replaces the Installing Base Operating System screen. When this screen displays, it prompts you to place the

device-support media in the drive and press the Enter key. BOS installation reconfigures the supplemental disk, then returns to the Installing Base Operating System screen.

The system reboots automatically when the installation completes.

Chapter 13. Alternate Disk Installation

Alternate disk installation lets you install the operating system while it is still up and running, which reduces installation or upgrade downtime considerably. It also allows large facilities to better manage an upgrade because systems can be installed over a longer period of time. While the systems are still running at the previous version, the switch to the newer version can happen at the same time.

Alternate disk installation can be used in the following ways:

- Installing a **mksysb** image on another disk. For further information, see "Alternate mksysb Disk Installation."
- Cloning the current running **rootvg** to an alternate disk. For further information, see "Alternate Disk rootvg Cloning" on page 148.
- Using the Network Installation Management (NIM) environment to perform a alternate disk migration installation of a NIM client. For further information, see "Alternate Disk Migration Installation" on page 149.

Filesets to Install

An alternate disk installation uses the following filesets:

bos.alt_disk_install.boot_images	Must be installed for alternate disk **mksysb** installations.
bos.alt_disk_install.rte	Must be installed for **rootvg** cloning and alternate disk **mksysb** installations.

Alternate mksysb Disk Installation

Alternate **mksysb** installation involves installing a **mksysb** image that has already been created from a system, onto an alternate disk of the target system. The alternate disk or disks cannot contain a volume group. The **mksysb** image is created on a system that either was the same hardware configuration as the target system, or had all the device and kernel support installed for a different machine type or platform, or different devices. The installed device and kernel support would be as follows:

- **devices.***
- **bos.mp**
- **bos.up**
- **bos.mp64**, if necessary

Note: Beginning in AIX 5.2, all device and kernel support is automatically installed during a base operating system installation.

When the **alt_disk_install** command is run, the **image.data** file from the **mksysb** image is used by default (unless a customized **image.data** is given) to create the logical volumes and file systems. The prefix **alt_** is added to the logical volume names, and the file systems are created with a prefix of **/alt_inst**. For example, `hd2` would be created as **alt_hd2**, and its file system, **/usr**, would be created as **/alt_inst/usr**. These names are changed back to their original names at the end of the alternate disk installation process.

The **mksysb** image is then restored into the alternate file system. A prepopulated boot image is then copied to the boot logical volume of the **altinst_rootvg**, and the boot record of the boot disk is modified to allow booting from the disk.

At this point, a script can be run to allow for any customization before the system is rebooted. The alternate file systems are still mounted as **/alt_inst/*real_file_system*** (for example: **/alt_inst/usr**,

© Copyright IBM Corp. 2002, 2004

/alt_inst/home). Files can be accessed at this point, but nothing can be installed into the alternate file system because the kernels and libraries of the **mksysb** image may not match those of the running system.

After the optional script is run, the file systems are unmounted, and the logical volume and file system names are changed to match the **image.data** file's names (for example, **alt_inst_hd6** is changed to **hd6** in the volume group descriptor area). The logical volumes are exported from the Object Data Manager (ODM), but the **altinst_rootvg** is only varied off. It is left in the ODM as a placeholder so the disk is not accidentally overwritten. The default action of the **alt_disk_install** command is to set the bootlist so that the next time the system boots, it boots from this newly installed volume group. This default action can be turned off. If specified, the system reboots at this point, and the system reboots from the new **rootvg**. The boot process proceeds to a certain point, with the new **rootvg**'s file systems mounted, and the **bosboot** command is called to rebuild a ″normal″ boot logical volume. The system then reboots.

After rebooting from the new alternate disk, the former **rootvg** volume group is contained in an **lspv** listing as `old_rootvg`, and includes all disk(s) in the original **rootvg**. This former **rootvg** volume group is set to `not varyon` at reboot and should *only* be removed with the **-X** flag. For example:

```
alt_disk_install -X old_rootvg
```

If a return to the original **rootvg** is necessary, the **bootlist** command is used to change the bootlist to reboot from the original **rootvg**.

If it is unclear which disk is the boot disk for a specific volume group, use the **-q** flag to determine the boot disk. This flag can be useful when a volume group comprises multiple disks and a change in the bootlist is necessary.

Alternate Disk rootvg Cloning

Cloning the **rootvg** to an alternate disk has many advantages. One advantage is having an online backup available, in case of a disk crash. Keeping an online backup requires an extra disk or disks to be available on the system. Another benefit of **rootvg** cloning occurs when applying new maintenance levels or updates. A copy of the **rootvg** is made to an alternate disk, then updates are applied to that copy. The system runs uninterrupted during this time. When it is rebooted, the system boots from the newly updated **rootvg** for testing. If updates cause problems, the **old_rootvg** can be retrieved by resetting the bootlist and then rebooting.

By default, calling the **alt_disk_install** command does the following:

1. Creates an **/image.data** file based on the current **rootvg**'s configuration. A customized **image.data** file can be used.
2. Creates an alternate **rootvg** (**altinst_rootvg**).
3. Creates logical volumes and file systems with the **alt_inst** prefix.
4. Generates a backup file list from the **rootvg**, and if an **exclude.list** file is given, those files are excluded from the list.
5. Copies the final list to the **altinst_rootvg**'s file systems.
6. If specified, the **installp** command installs updates, fixes, or new filesets into the alternate file system.
7. The **bosboot** command creates a boot logical volume on the alternate boot disk.
8. If a customization script is specified, it runs at this point.
9. The file systems are then unmounted, and the logical volumes and file systems are renamed.
10. The logical volume definitions are exported from the system to avoid confusion with identical ODM names, but the **altinst_rootvg** definition is left as an ODM placeholder.
11. By default, the bootlist is set to the new cloned **rootvg** for the next reboot.

Phased Alternate Disk Installation

For AIX 4.3.1 and later, alternate disk installation can be performed in stages. The installation itself is broken down into three phases. The default is to perform all three phases in the same invocation. The phases are as follows:

Phase 1	Creates the **altinst_rootvg** volume group, the **alt_** logical volumes, and the **/alt_inst** file systems. Also restores the **mksysb** or **rootvg** data.
Phase 2	Runs any specified customization script. For cloning only, installs updates, new filesets, fixes, or bundles. Also copies a **resolv.conf** file (if specified) and necessary files to remain a NIM client (if specified).
Phase 3	Unmounts the **/alt_inst** file systems, renames the file systems and logical volumes, removes the **alt_** logical volume names from ODM, and varies off the **altinst_rootvg**. It also sets the bootlist and reboots (if specified).

As an alternative to running all three phases, the phases can be completed by one of the following methods:

- Each phase separately
- Phases 1 and 2 together
- Phases 2 and 3 together (Phase 2 can be run multiple times before Phase 3 is run.)

You must run Phase 3 to obtain a usable **rootvg**. Running Phases 1 and 2 leave the **/alt_inst** file systems mounted. Any time during the phase process and before rebooting, the **altinst_rootvg** can be removed, and disk cleanup occurs using the following command:

```
alt_disk_install -X
```

Alternate Disk Migration Installation

Alternate disk migration installation allows the user to create a copy of **rootvg** to a free disk (or disks) and simultaneously migrate it through Network Installation Management (NIM) to a new release level. Using alternate disk migration installation over a conventional migration provides the following advantages:

- Reduced downtime (the migration is performed while the system is up normally and there is no need to boot from any media).
- Quick recovery in case of migration failure.
- High degree of flexibility and customization.

Reduced downtime. The migration is performed while the system is up and functioning. There is no requirement to boot from install media, and the majority of processing occurs on the NIM master.

Quick recovery in the event of migration failure. Because you are creating a copy of **rootvg**, all changes are performed to the copy (**altinst_rootvg**). In the event of serious migration installation failure, the failed migration is cleaned up, and there is no need for the administrator to take further action. In the event of a problem with the new (migrated) level of AIX, the system can be quickly returned to the premigration operating system by booting from the original disk.

High degree of flexibility and customization in the migration process. This is done with the use of optional NIM customization resources: **image_data**, **bosinst_data**, **exclude_files**, premigration script, **installp_bundle**, and post-migration script.

Requirements

Alternate disk migration installation has the following requirements:

- Configured NIM master running AIX 5.1 or later with AIX recommended maintenance level 5100-03 or later.

- The NIM master must have **bos.alt_disk_install.rte** installed in its **rootvg** and the **SPOT** which will be used.
- The level of the NIM master **rootvg**, **lpp_source**, and **SPOT** must be at the same level.
- The client (the system to be migrated) must be at AIX 4.3.3 or later.
- The client must have a disk (or disks) large enough to clone the **rootvg** and an additional 500 MB (approximately) of free space for the migration. The total amount of required space will depend on original system configuration and migration customization.
- The client must be a registered NIM client to the master.
- The nim master must be able to execute remote commands on the client using the **rshd** protocol.
- The client must have a minimum of 128 MBs of memory.
- A reliable network, which can facilitate large amounts of NFS traffic, must exists between the NIM master and the client.
- The client's hardware should support the level it is migrating to and meet all other conventional migration requirements.

Note: If you cannot meet the alternate disk migration installation requirements 1-10, perform a conventional migration. For information on the conventional migration installation method, see Chapter 6, "Migration Installation," on page 57. If you cannot meet requirement 11, no migration installation is possible.

Before performing an alternate disk migration installation, you are required to agree to all software license agreements for software to be installed. You can do this by specifying the **-Y** flag as an argument to the alternate disk migration command or setting the **ADM_ACCEPT_LICENSES** environment variable to **yes**.

Limitations

The following limitations apply to alternate disk migration installation:

- If the client's **rootvg** has the Trusted Computing Base option enabled, either disable it (permanently) or perform a conventional migration. TCB must access file metadata that is not visible over NFS.
- All NIM resources used must be local to the NIM master.
- During the migration, the client's active **rootvg** may experience a small performance decrease due to increased disk I/O, **nfsd** activity, and some CPU usage associated with **alt_disk_install** cloning.
- NFS tuning may be required to optimize performance.

Alternate Disk Migration Installation Usage

The syntax for the alternate disk migration installation command is the following:

```
nimadm -l lpp_source -c NIMclient -s SPOT -d Targetdisks
     [ -a PreMigrationScript ] [ -b installp_bundle] [ -z PostMigrationScript]
     [ -e exclude_files] [ -i image_data ] [ -m NFSMountOptions ] [ -o bosinst_data]
     [-P Phases] [-Y ] [ -F ] [ -D ] [ -E ] [ -V ] [ { -B | -r } ]
```

Use the **nimadm** command to target the *aix1* NIM client, using the *spot1* NIM **SPOT** resource, the *lpp1* NIM **lpp_source** resource, and *hdisk1* and *hdisk2* target disks, by typing the following:

```
nimadm -c aix1 -s spot1 -l lpp1 -d "hdisk1 hdisk2" -Y
```

Use the **-Y** flag to agree to all required software license agreements for the software being installed

Alternate Disk Migration Installation Process

The **nimadm** command performs a migration in 12 phases. Each phase can be executed individually using the **-P** flag. Before performing a migration in phases, the user should have a good understanding of the **nimadm** process. The **nimadm** phases are as follows:

1. The master issues the **alt_disk_install** command to the client, which makes a copy of the **rootvg** to the target disks (this is Phase 1 of the **alt_disk_install** process). In this phase, **altinst_rootvg** (alternate **rootvg**) is created.
2. The master runs remote client commands to export all of the **/alt_inst** file systems to the master. The file systems are exported as read/write with root access to the master.
3. The master NFS mounts the file systems exported in Phase 2.
4. If a premigration script resource has been specified, it is executed at this time.
5. System configuration files are saved. Initial migration space is calculated and appropriate file system expansions are made. The **bos** image is restored and the device database is merged (similar to a conventional migration). All of the migration merge methods are executed and some miscellaneous processing takes place.
6. All system filesets are migrated using **installp**. Any required RPM images are also installed during this phase.
7. If a **post-migration** script resource has been specified, it is executed at this time.
8. The **bosboot** command is run to create a client boot image, which is written to the client's boot logical volume (**hd5**).
9. All mounts made on the master in phase 3 are removed.
10. All client exports created in phase 2 are removed.
11. The **alt_disk_install** command is called again (phase 3 of **alt_disk_install**) to make final adjustments and put **altinst_rootvg** to sleep. The bootlist is set to the target disk (unless the **-B** flag is used).
12. Cleanup is executed to end the migration. The client is rebooted, if the **-r** flag is specified.

Note: The **nimadm** command supports migrating several clients at the same time.

For more information about the **nimadm** command, refer to the *AIX 5L Version 5.2 Commands Reference*.

Data Access Between the Original rootvg and the New Alternate Disk

You can initiate data access between the original rootvg and the new alternate disk. A volume group ″wake-up″ can be accomplished, on the non-booted volume group. The ″wake-up″ puts the volume group in a **post alt_disk_install** Phase 1 state. For example, the **/alt_inst** file system is then mounted.

The volume group that experiences the ″wake-up″ is renamed **altinst_rootvg**. When data access is no longer needed, the volume group can be ″put to sleep.″

Notes:

1. The running operating system's version must be greater than or equal to the version of the volume group that undergoes the ″wake-up.″ This might mean that it is necessary to boot from the **altinst_rootvg** and ″wake-up″ the **old_rootvg**. For example, an alternate disk is created from an **alt_disk_install** AIX 5.2 mksysb, on a AIX 4.3.0 system. It is then necessary to boot from the AIX 5.2 alternate disk and ″wake-up″ the AIX 4.3.0 **old_rootvg** volume group to access data between the two volume groups.

   This limitation is caused by a journaled file system (JFS) log entry incompatibility. It is possible to ″wake-up″ a volume group that contains a more recent version, but the volume group cannot have ever been the system **rootvg**. If this was true, the volume group would have made JFS log entries that could not be interpreted by an older version **rootvg**, when the volume group was experiencing a ″wake-up.″

   The **alt_disk_install** command does not allow a ″wake-up″ to occur on a volume group with a more recent version, unless the **FORCE** environment variable is set to **yes**.
2. The volume group that experiences a ″wake-up″ *must* be put to sleep before it can be booted and used as the **rootvg**.

Attention: If a FORCE ″wake-up″ is attempted on a volume group that contains a more recent version of the running operating system, and the ″waking″ volume group has been a system **rootvg**, errors occur.

Installing to an Alternate Disk using Web-based System Manager

The graphical interface provides access to Web-based System Manager options for installing a **mksysb** to an alternate disk and for cloning a rootvg to the alternate disk. At any time during the following procedures, you can view extended help by selecting **Contents** from the Help menu.

To install a mksysb to an alternate disk, do the following:

1. Start the Web-based System Manager by typing **wsm** on the command line.
2. Select the **Software** container.
3. From the pulldown, select **Alternate Disk Install—>Install Mksysb on an Alternate Disk**.

To clone the rootvg to an alternate disk, do the following:

1. Start the Web-based System Manager by typing **wsm** on the command line.
2. Select the **Software** container.
3. From the pulldown, select **Alternate Disk Install—>Clone the Rootvg to an Alternate Disk**.

Running Alternate Disk Installation Using SMIT

To run alternate disk **mksysb** installation, do the following:

1. At the system prompt, type the **smit alt_mksysb** fast path.
2. Type or select values in the entry fields. Press Enter after making all desired changes.

To run alternate disk **rootvg** cloning, do the following:

1. At the system prompt, type the **smit alt_clone** fast path.
2. Type or select values in the entry fields. Press Enter after making all desired changes.

Alternate Disk Installation and Dynamic Logical Partitioning

On a system that supports dynamic logical partitioning (DLPAR), you can dynamically add an adapter with disks to a running logical partition (LPAR). You can then install a new rootvg volume group to these newly added target disks using the **alt_disk_install** command with either the clone or **mksysb** option. If you are running the **alt_disk_install** command with dynamically added target disks on an LPAR system, the following flags might be used:

-O If the target disk will be used to boot an LPAR other than the one where the operation is being executed, use the **-O** flag to reset the device information.

-B This flag prevents the **bootlist** command from being run. A general limitation of dynamically added disks is that you can not specify them as a boot device (before an initial reboot operation). If you are attempting to boot an LPAR from dynamically added disks, set the boot list in the system management services (SMS) menus.

-g This flag causes the **alt_disk_install** command to run without checking if the disk is bootable. Dynamically added disks do not appear bootable to AIX until after a reboot operation. The user will need to verify that the newly added adapter and disks are bootable.

Examples

1. To clone the **rootvg** running 4.3.2.0 to `hdisk1` and update that clone with the latest maintenance level 4.3.3.0 that is on `cd0`, run the following command:

   ```
   alt_disk_install -C -b update_all -l /dev/cd0 hdisk1
   ```

In SMIT, use the **smit alt_clone** fast path and select **hdisk1** from the listing for Target Disk(s) to install, select the **update_all** bundle from the listings in the **Bundle to Install** field, and **/dev/cd0** from the listing in the **Directory or Device with images** field.

2. To clone the **rootvg** running 4.3.2 to `hdisk3`, then update to the latest fixes that are mounted from another system on `/433fixes`, and run a customized script named **/tmp/finish_alt_install**, run the following command:

   ```
   alt_disk_install -C -b update_all -l /433fixes \
   -s /tmp/finish_alt_install hdisk3
   ```

   In SMIT, use the **smit alt_clone** fast path and select **hdisk3** from the listing for Target Disk(s) to install, select the **update_all** bundle from the listings in the Bundle to Install field, type `/433fixes` in the **Directory or Device with images** field, and type `/tmp/finish_alt_install` in the **Customization script** field.

3. To install an AIX 5.2 **mksysb** tape that was created from a machine with the same hardware configuration as the target, to `hdisk1`, run the following command:

   ```
   alt_disk_install -d /dev/rmt0 hdisk1
   ```

   In SMIT, use the **smit alt_mksysb** fast path and select **hdisk1** from the listing for **Target Disk(s)** to install field and select **/dev/rmt0** from the listing for **Device** or image name field.

4. To install an AIX 5.2 **mksysb** image that is NFS mounted on file system **/mksysbs** to the alternate disk `hdisk2` using a customized **image.data** file and an exclude file containing **^./tmp/**, type the following command:

   ```
   alt_disk_install -d /mksysbs/my_52_mksysb -i /mksysbs/my_52_image.data \
   -e /mksysbs/my_exclude_file hdisk2
   ```

   Using the **^./tmp/** pattern does not backup files in the **/tmp** directory, but does backup files in the **/var/tmp** directory.

   > **Note:** All files are backed up relative to the current directory. This directory is represented by a **.** (dot character). If it is important that the search match the string at the beginning of the line when excluding a file or directory, it is necessary to use a **^.** (caret followed by a dot character) as the first part of the search string, followed by the filename or directory to be excluded. The form is as follows:
   >
   > ^./*filename*
   >
   > If the file name or directory being excluded is a substring of another file name or directory, use a **^.** (caret followed by a dot character) for the search to start at the beginning of the line and the **$** (dollar symbol) to have the search finish at the end of the line.

   In SMIT, use the **smit alt_mksysb** fast path and select **hdisk2** in the Target Disk(s) to install field. Next, type `/mksysbs/my_52_mksysb` in the **Device** or image name field, `/mksysbs/my_52_image.data` in the **image.data** file field, and **/mksysbs/my_exclude_file** in the **Exclude** list field.

5. To ″wake-up″ an original rootvg, after booting from the new alternate disk, run the following command:

   ```
   alt_disk_install -W hdisk0
   ```

   The following example illustrates the output that might display when running the command discussed above:

   ```
   # lspv
   hdisk0          000040445043d9f3    old_rootvg
   hdisk1          00076443210a72ea    rootvg

   # alt_disk_install -W hdisk0

   # lspv
   hdisk0          000040445043d9f3    altinst_rootvg
   hdisk1          00076443210a72ea    rootvg
   ```

   At this point, the **altinst_rootvg** volume group is varied-on and the **/alt_inst** file systems are mounted.

6. To ″put-to-sleep″ a volume group that had experienced a ″wake-up,″ type the following command:

   ```
   alt_disk_install -S
   ```

The following example illustrates the output that might display when running the command previously discussed:

```
# lspv
hdisk0          000040445043d9f3    altinst_rootvg
hdisk1          00076443210a72ea    rootvg

# alt_disk_install -S

# lspv
hdisk0          000040445043d9f3    altinst_rootvg
hdisk1          00076443210a72ea    rootvg
```

The **altinst_rootvg** is no longer varied on and the **/alt_inst** file systems are no longer mounted. If necessary for the **altinst_rootvg** volume group name to be changed back to **old_rootvg**, do this task with the **-v** flag.

Chapter 14. Software Product Packaging Concepts

This chapter discusses concepts regarding additional software product packaging. This chapter includes information on the following topics:

- "Software Package Formats"
- "Fileset Installation Packaging" on page 156
- "Creating Software Packages" on page 156
- "Bundle Packaging" on page 157

Software Package Formats

Beginning in AIX 5.1, you can install RPM Package Manager (**RPM**) and **InstallShield MutliPlatform** (**ISMP**) formatted packages in addition to **installp** formatted packages. Use the Web-based System Manager, SMIT, or the **geninstall** command to install and uninstall these types of packages. The **geninstall** command can detect the format type of a specified package and run the appropriate installation command.

Beginning in AIX 5.1, the AIX product media contains **installp** packages and **RPM** packages that are installed during a base operating system (BOS) installation. The **installp** packages are located in the following path:

/mount_point/installp/ppc

The **RPM** packages are located in the following path:

/mount_point/RPMS/ppc

If you have media that contains **ISMP** packages for AIX 5.1 and later, the **ISMP** packages are located in the following path:

/mount_point/ISMP/ppc

If you are using the **geninstall** command to install **RPM** or **ISMP** packages, use the prefix type to indicate to the **geninstall** command the type of package that you are installing. In AIX 5.1, the package prefix types are the following:

I: **installp** format

R: **RPM** format

J: **ISMP** format

For example, to install the **cdrecord RPM** package and the **bos.games installp** package, type the following:

```
# geninstall -d/dev/cd0 R:cdrecord I:bos.games
```

The **geninstall** command detects that the **cdrecord** package is an **RPM** package type and runs the **rpm** command to install the **cdrecord** package. The **geninstall** command then detects that **bos.games** is an **installp** package type and runs the **installp** command to install the **bos.games** package. The process for uninstallation is similar to the installation process.

In Web-based System Manager and SMIT, if you are selecting the packages from a software list, you need not specify the prefix type.

© Copyright IBM Corp. 2002, 2004

Fileset Installation Packaging

The installation packaging of each fileset in a product can be divided into three parts. These parts include the usr, root, and share parts. Although this can add further complexity to the understanding of the packaging, this parceling of a software product is necessary for the product to be used by diskless and dataless clients in AIX 5.1 and later. Because they are parceled, a product can be installed on one machine (called the *server*) and then be used remotely by other machines on a network (called the *clients*).

Note: The usr and root parts of a product are packaged in the same installable package.

usr part

The usr part of a software product contains the part of the product that can be shared by machines that have the same hardware architecture. Most of the software that is part of a product usually falls into this category.

In a standard system, the usr parts of products are stored in the **/usr** file tree. For example, the **ls** command would be in the **/usr/bin/ls** file.

root part

Every product has a usr part. The root part of a software product contains the part of the product that cannot be shared. The root part of a product is optional because many products may not have any files that need to be specific to each individual machine.

In a client/server environment, these are the files for which there must be a unique copy for each client of a server. Most of the root software is associated with the configuration of the machine or product.

In a standard system, the root parts of a product are stored in the root (**/**) file tree. The **/etc/objrepos** directory contains the root part of an installable software product's vital product data (VPD).

share part

The share part of a software product contains the part of the product that can be shared among machines, even if they have different hardware architectures, which can include nonexecutable text or data files. For example, the share part of a product might contain documentation written in ASCII text or data files containing special fonts.

The share part of a product is optional because many products might not have any files that can be shared among different hardware platforms. The share part of a product is always packaged in a separately installable package.

In a standard system, the share parts of products are usually stored in the **/usr/share** file tree. For example, a dictionary database might be stored in the **/usr/share/dict/words** file.

Creating Software Packages

The **mkinstallp** command is a tool that allows users to create their own software packages for AIX. Packages created with the **mkinstallp** command are in **installp** format and are installed or removed with the **installp** command.

Files to be packaged by the **mkinstallp** command must be in a directory structure such that the location of the file relative to the root build directory is the same as the destination of the file after installation. For example, if the **/usr/bin/***somecommand* command is to be installed by a **mkinstallp** package, the *somecommand* parameter must be in the *buildroot***/usr/bin** directory when the **mkinstallp** command is invoked.

When the contents of a package are in the correct directory structure, the **mkinstallp** command prompts for basic package data through the command line. This data includes the package name, requisites, descriptions of files to be packaged, and more. The **mkinstallp** command then generates a template file based on responses given by the user. To prevent command line prompting when using a template file, create and edit the template file directly and use the **mkinstallp** command with the **-T** flag.

For example, to package the **/usr/bin/foo** command using the **/tmp/packages** directory as the build root, make sure the following directory structure exists by typing the following at the command line:

```
mkdir /tmp/packages
touch /tmp/packages/usr/bin/foo
```

Then type the following:

```
mkinstallp -d /tmp/packages
```

For more examples, refer to the **/usr/lpp/bos/README.MKINSTALLP** file.

Bundle Packaging

The Web-based System Manager and the SMIT Install application look for bundles in **/usr/sys/inst.data/sys_bundles** and in **/usr/sys/inst.data/user_bundles**. The **sys_bundles** location is typically reserved for system-defined bundles (those which come with AIX). Users can create their own bundle files in the **user_bundles** directory.

The bundle definition file name must end in **.bnd**, because the AIX installation interfaces that process bundles recognize only bundle files that end in .bnd. Use any editor to create bundle files, which can contain comments and fileset names. Lines beginning with the pound sign (#) are recognized as comments and are ignored by the bundle processing code. When you have completed your list of filesets, save the file and make sure the file has the appropriate read permission. Invoking a bundle installation interface displays your bundle without the **.bnd** extension.

The following are examples of the predefined bundles:

- *Server Bundle*. A collection of software packages for machines running AIX in a multiuser standalone or networked environment. This bundle emphasizes functionality over disk utilization.
- *Graphics Bundle*. A collection of software packages that provides support of graphical environments. Graphical support may be automatically installed on some systems during BOS installation.
- *Migration Bundle*. This bundle is created when there was not enough disk space available to complete a migration installation during the BOS installation process. The bundle consists of a collection of software packages that must be installed to complete your migration. You must install this bundle to complete the migration installation. Install the bundle using the **smit update_all** fast path.

  You may also need to install the *Graphics Bundle*.

Some system bundles might refer to installation images that are spread across multiple media. If you see errors indicating that filesets could not be found on the media you are using, insert the media containing the missing filesets and retry the bundle installation.

The system bundles are located in the **/usr/sys/inst.data/sys_bundles** directory. To list the system bundles, type the following:

```
ls /usr/sys/inst.data/sys_bundles/*.bnd
```

You can also use the SMIT **list_bundle** fast path to list the system bundles.

Beginning in AIX 5L Version 5.2 with the 5200-01 Recommended Maintenance package, the **geninstall** and **gencopy** commands handle multiple software sources to be specified when a bundle file is used. This is accomplished by grouping software images together under **#MEDIA=** headings in the bundle file. Any images listed under such a heading must reside on the specified media. Media can be specified as the name of a CD (such as *Base Install Media Volume 1* or *AIX Linux Toolbox CD*) or as a local directory (such as the **/usr/sys/inst.images** directory).

The **#MEDIA=** heading is used to designate the location of the filesets or packages in the bundle. For example, the *BaseAndLinuxCD Bundle* might contain the following information:

```
# BaseAndLinuxCDBundle contains packages on volume 1 of base media and on the AIX
# Linux Toolbox CD

#MEDIA=Base Install Media Volume 1
I:bos.adt.prof

#MEDIA=AIX Linux Toolbox CD
R:mtools
R:vim-common
```

When the **geninstall** and **gencopy**commands prompt for the additional media, they use the words provided in the **#MEDIA=** line. In the previous examples, the **geninstall** and the **gencopy** commands display a message informing you that the **bos.adt.prof installp** package is located on *Base Install Media Volume 1*, and the **mtools** and **vim-common RPM** packages are located on the *AIX Linux Toolbox CD*.

The **#MEDIA=** heading can also be used to indicate a directory. For example, the *CD_Directory Bundle* might contain the following information:

```
# CD_DirectoryBundle contains packages on volume 1 of base install media
# and in /usr/sys/inst.images

#MEDIA=/usr/sys/inst.images
I:bos.games

#MEDIA=Base Install Media Volume 1
I:bos.adt.prof
R:cdrecord-1.9-4
```

This informs the **geninstall** and the **gencopy** commands that the **bos.games installp** package is located in the **/usr/sys/inst.images** directory, and the **bos.adt.prof installp** package and the **cdrecord-1.9-4 RPM** package are located on *Base Install Media Volume 1*.

Chapter 15. Troubleshooting Operating System and Optional Software Installation

This chapter provides problem-determination tactics and solutions for installation and configuration problems. Topics include:

- “Troubleshooting an Installation from a System Backup”
- “Troubleshooting a Migration Installation” on page 161
- “Troubleshooting an Alternate Disk Installation” on page 162
- “Troubleshooting After an Installation” on page 162
- “Accessing a System That Does Not Boot” on page 163
- “Troubleshooting a Full /usr File System” on page 165
- “Viewing BOS Installation Logs” on page 166
- “Handling System and Error Messages” on page 166.

Troubleshooting an Installation from a System Backup

This section describes solutions for common problems when installing from a system image created with the **mksysb** command.

Bootup Failure

If a backup tape fails to boot, you can still install by using a **mksysb** image stored on the tape.

Boot the machine from the product media (Volume 1 if there is more than one volume), then install the backup from Maintenance mode. For instructions on booting, refer to Chapter 3, “Introduction to Base Operating System Installation,” on page 43. Follow the instructions to the point when the Welcome to the Base Operating System Installation and Maintenance screen displays.

Booting from the Product CD-ROM

Complete the following steps when the Welcome screen is displayed:

1. Choose the **Start Maintenance Mode for System Recovery** option.
2. Choose the **Install from a System Backup** option.
3. Choose the drive containing the backup tape.

   The system reads the tape and begins the installation.
4. Do not remove the CD from the CD-ROM drive.

   The system installs the kernel and device support required on the target system from the CD.
5. Return to step 8 on page 143 in the Installing a System Backup on the Source Machine procedure and continue the instructions for installing the backup.

   Note: The **Use Maps** option is not supported in Maintenance Mode. For more information on the maps options in Maintenance Mode, refer to “Installing a System Backup on the Source Machine” on page 140.

Problems with mksysb Image Configuration on System Backup Tapes

Bootable **mksysb** tapes comprise the following images:

- Boot image
- BOS Installation/Maintenance image
- Table of contents image

© Copyright IBM Corp. 2002, 2004

- System backup image

The system backup image is the actual backup of the files in the rootvg in all JFS-mounted file systems.

The boot image, BOS Installation/Maintenance image, and the table of contents image must be created with a tape **block_size** value of 512. The **mksysb** command ensures that the block size is 512 when these images are created. There are no restrictions on the block size used for the fourth (system backup image) on the tape. The block size of the system, before it was temporarily set to 512, is used for the fourth image on the tape.

The value of the block size must be saved in the **/tapeblksz** file in the second image on the tape. The second and fourth images are stored in backup/restore format. Again, **mksysb** ensures the correctness of the tapes created by using the **mksysb** command.

If there are problems with the **bosinst.data** file, the **image.data** file, or the **tapeblksz** file, these files can be restored from the second image on the tape and checked. These files, as well as commands necessary for execution in the RAM file system (when running in maintenance mode after booting from the tape), are stored in the second image.

Restoring a File from the Second Image or Tape

To restore a file from the second image, follow these steps:

1. Be sure the tape block size is 512 by entering the following command:

   ```
   # lsattr -E -l rmt0
   ```

   If the block size is not correct, use the following command to set it to 512:

   ```
   # chdev -l rmt0 -a block_size=512
   ```

2. Make sure the tape is rewound. If the tape is not rewound, enter the following command:

   ```
   # tctl -f /dev/rmt0 rewind
   ```

3. Extract the necessary files by entering:

   ```
   # restore -xvq -s2 -f /dev/rmt0.1 ./filename
   ```

   Note: The filename should be the full path, and always preceded with a **.** (dot character), such as `./tapeblksz`.

4. Rewind the tape by entering:

   ```
   # tctl -f /dev/rmt0 rewind
   ```

5. Change the block size back to its original value, if necessary.

Troubleshooting Reported Problems with mksysb Backup Installations

The following troubleshooting tips apply to reported problems with installations from a **mksysb** image:

- Check that you have sufficient free blocks in the file systems to write temporary files.
- Check that each file system has at least 500 blocks free when the **mksysb** backup image is made. The system needs work space in each file system when installing from a **mksysb** backup image.
- Check that you are using the correct tape type for the density setting that you selected.
- Check that the tape is *not* write-protected.
- Clean the tape drive at the recommended intervals and use only approved data-grade tapes (not video tapes for 8 mm).
- Check that 7206 4-mm Digital Audio Tape (DAT) tape drives are using only DAT tapes marked with the Dataphone Digital Services (DDS) symbol. Any other DAT tapes (for example, voice grade) cannot be used.
- Check the **/smit.log** file for any errors from SMIT.

- Check that your **mksysb** backup image contains an **image.data** file. If you create the **mksysb** backup image through Web-based System Manager or SMIT, it is done automatically. If you run **mksysb** from the command line, you must either run the **mkszfile** command first, or use the **-i** flag with the **mksysb** command.

Troubleshooting a Migration Installation

The following sections offer solutions for problems that can occur during a migration installation.

Boot Logical Volume Not Large Enough

If you receive errors indicating the boot logical volume is not large enough, see "Handling System and Error Messages" on page 166.

Insufficient Disk Space for Migration

At the beginning of a migration installation, the system verifies that there will be enough space to attempt the migration. If there is not enough disk space, a message explains how much is needed. You must now reboot the machine from the media containing your current version of AIX, and make more space available in the **rootvg** volume group. After you do this, attempt the migration again.

You can use the following options for adding additional disk space for the migration installation:

- Add another disk to the **rootvg** volume group, using either the SMIT **smit extendvg** fast path or the **extendvg** command.
- Move any user-data logical volumes from the **rootvg** volume group to another volume group. You can use either the SMIT **smit cplv** fast path or the **cplv** command to move individual logical volumes to another volume group's disk. It is a good idea to have only system logical volumes in the **rootvg**, and have user-data logical volumes in other volume groups.

  After you use the **cplv** command, you must remove the original logical volumes with the **rmlv** command. If the moved logical volume contains a file system, you must modify its corresponding entries in the **/etc/filesystems** file to reflect the new logical volume name.

  For more detailed information about manipulating logical volumes and volume groups, refer to Logical Volumes in *AIX 5L Version 5.2 System Management Guide: Operating System and Devices*.
- Remove unneeded logical volumes (and file systems) from the **rootvg**. Run the **lsvg -l rootvg** command to see all the logical volumes in the **rootvg** volume group. The only logical volumes that must be in the **rootvg** are: hd2, hd3, hd4, hd5, hd6, hd8, and hd9var. The hd1 (**/home**) logical volume can be located in another volume group if necessary.

  Beginning in AIX 5.1, the hd7 (system dump) logical volume is not needed because the paging space logical volume (`hd6`) is used. The migration code automatically removes this logical volume if space is needed, but you can remove it ahead of time with the following commands:

  ```
  sysdumpdev -P -p /dev/hd6
  rmlv -f hd7
  ```
- If you cannot find extra space in your **rootvg**, you might have to do a *preservation* installation instead of a migration installation to AIX. A preservation installation saves all the ″non-system″ logical volumes and file systems (for example, **/home**), but removes and re-creates the following logical volumes: hd2, hd3, hd4, hd5 and hd9var.

  If you do a preservation installation, you must reinstall any applications that were installed in your **rootvg** after the preservation installation has completed. You must also reconfigure devices, as well as re-create users and groups. For more information about a preservation installation, see Chapter 3, "Introduction to Base Operating System Installation," on page 43.

After you have released enough space, reboot from your installation media, and try the migration installation again. You must have at least 8 MB of free disk space to complete the migration installation.

If there is insufficient space to complete the migration installation during the BOS installation process, a message similar to the following is displayed at the end of the installation:

```
An error occurred while migrating packages.

Some packages have not been installed.

Please see /var/adm/ras/devinst.log for details or perform an overwrite or
   preservation install.
```

If space limitations prevent the migration of all software that is usually automatically migrated, the installation program attempts to install the software that is usually installed for a Preservation or Overwrite installation. If there is still not enough disk space available, the minimum set of software required to support the use of the system is installed.

If there is not enough space to migrate all of the usually migrated software, a collection of software called a Migration Bundle will be available when you install additional software later. If the minimum set of software is installed, or if the installation is not performed from a graphics console, a Graphics_Startup Bundle is created. Before installing either of these bundles, create additional disk space on the machine you want to install. For more information about installing software bundles and migrating or installing optional software products, refer to Chapter 10, “Optional Software Products and Service Updates,” on page 81. “Maintaining Optional Software Products and Service Updates” on page 87 describes how to remove software from the system to release disk space.

Troubleshooting an Alternate Disk Installation

If you receive either of the following error messages, see “Handling System and Error Messages” on page 166.

- `0505-113 alt_disk_install: No target disk name provided.`
- `0505-117 alt_disk_install: Error restoring image.data file from mksysb image.`

Other Problems

Symptom: You have run the **alt_disk_install** command or used the SMIT menus to either clone or install a **mksysb** image on an alternate disk. However, you now want to remove the definition so you can use the disk to run the **alt_disk_install** command again or use the disk for another purpose.

> **Action:** *Do not run* the **exportvg** command. The **exportvg** examines the logical volumes on the disk (now called by their rootvg names: **hd1**, **hd2**, **hd3**, and so on) and tries to remove their corresponding entries from the **/etc/filesystems** file. This action removes the real file system stanzas from your running system and causes boot problems if you reboot with the missing stanzas.
>
> Use the **alt_disk_install -X** command to remove the **altinst_rootvg** name from the database. This removes only the ODM information from the CuDv database, so the **lspv** command shows the disk(s) as no longer belonging to **altinst_rootvg**. It also resets your bootlist to the boot disk on which the hd5 boot logical volume resides. You can still boot from the **altinst_rootvg**, because the volume group, logical volume, and file system information remain on the disk. However, you must set your bootlist to the **altinst_rootvg** boot disk.

Troubleshooting After an Installation

If your system was installed by a network installation server, Configuration Assistant or Installation Assistant will not display when the BOS installation program completes.

Configuration Assistant and Installation Assistant do not contain the tasks needed to configure your machine as a server. If you need to configure your system for a specific resource, refer to the documentation pertaining to that resource.

If your terminal type is not set, the first menu displayed by the ASCII Installation Assistant requires you to enter your terminal type (tty). If you enter a terminal type that is not valid, this menu redisplays until a valid type is entered.

If you enter a valid terminal type that does not match your terminal, the next screen displayed may be unreadable. In this case, press the break key sequence to return to the Set Terminal Type screen. For most terminal types, the break key sequence is Ctrl-C.

Accessing a System That Does Not Boot

This section describes how to access a system that will not boot from the hard disk. If a **mksysb** backup tape fails to boot, read “Troubleshooting an Installation from a System Backup” on page 159 for instructions.

This procedure enables you to get a system prompt so that you can attempt to recover data from the system or perform corrective action that will enable the system to boot from the hard disk.

Notes:

1. This procedure is intended only for experienced administrators who have knowledge of how to boot or recover data from a system that is unable to boot from the hard disk. Most administrators should not attempt this procedure but instead should follow local problem-reporting procedures.
2. This procedure is not intended for administrators who have just completed a New Installation, because the system will not contain data that needs to be recovered. If you are unable to boot from the hard disk after completing a New Installation, follow your local problem-reporting procedures.

The following steps summarize the procedure for accessing a system that will not boot.

1. Boot the system from Volume 1 of the BOS CD-ROM or a bootable tape.
2. Select **Maintenance Options**.
3. Recover data or perform corrective action using the system prompt.

Prerequisites

Before continuing with the procedure, confirm the following prerequisites have been met:

- Your system cannot be booted from the hard disk.
- All hardware is installed.
- AIX Base Operating System (BOS) is installed.
- Your system unit is set to Off.

Accessing the System

Use this procedure if you are unable to boot from the hard disk. The beginning of this procedure is similar to the one you used to install the Base Operating System. You will, however, use the maintenance screens instead of the installation screens to complete this procedure.

1. Turn on all attached external devices, such as terminals, CD-ROM drives, tape drives, monitors, and external disk drives *before* turning on the system unit. Do not turn on the system unit until step 5. Turning on the external devices first is necessary so that the system unit can identify them during the startup (boot) process.
   - If you are booting from a network device, refer to Appendix A, “Network Boot,” on page 343.
   - If you are not booting from a network device, go to step 3.
2. Insert Volume 1 of the installation media into the tape or CD-ROM drive. Some CD-ROM drives have a removable disc caddy, while others have a sliding drawer. If the CD-ROM drive on your system has

a sliding drawer, place the CD-ROM in the drawer and push the drawer in. If the CD-ROM drive on your system does not have a sliding drawer, insert the CD-ROM into the disc caddy and then insert the caddy into the CD-ROM drive.

Notes:

a. You may find that on specific hardware, the tape drive door will not open while the system unit is turned off. If you have trouble opening the tape drive door during installation, use the following procedure:
   1) Turn the system unit on.
   2) Insert the BOS tape (insert Volume 1 if you received more than one volume).
   3) Turn the system unit off and wait 30 seconds.

b. On some models that have a door to the tape drive, there may be a waiting period of up to three minutes before the tape drive door opens after you have pressed the button to open the tape drive. Some models also require that the button for the tape drive door be held in the pressed position for a few seconds before the tape drive door will open.

c. On some models, the eject button must be pressed for at least 2 seconds to eject a CD-ROM that is already in the disc caddy.

3. If you are not using an ASCII terminal, skip to step 5. If you are using an ASCII terminal, set the communications options 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

   Set the keyboard and display options as follows:
   - Screen = Normal
   - Row and Column = 24x80
   - Scroll = jump
   - Auto LF (line feed) = off
   - Line Wrap = on
   - Forcing Insert = line (or both)
   - Tab = field
   - Operating Mode = echo
   - Turnaround Character = CR
   - Enter = return
   - Return = new line
   - New Line = CR
   - Send = page
   - Insert Character = space

   Note: If your terminal is an IBM 3151, 3161, or 3164, press the Ctrl+Setup keys to display the Setup Menu and follow the onscreen instructions to set these options. If you are using some other ASCII terminal, refer to the appropriate documentation for information about how to set these options. Some terminals have different option names and settings than those listed here.

4. Turn the system unit power switch to the On position. The system begins booting from the installation media. If your system is booting from tape, it is normal for the tape to move back and forth. After several minutes, c31 is displayed in the LED.

   If you have more than one console, each terminal and directly attached display device (or console) might display a screen that directs you to press a key to identify your system console. A different key is specified for each terminal displaying this screen. If this screen is displayed, then press the specified key on the device to be used as the system console. The system console is the keyboard and display device used for installation and system administration. Press a key on only one console.
5. Type 3 to select **Start Maintenance Mode for System Recovery** from the Welcome to the Base Operating System Installation and Maintenance screen when it displays.

   Note: If you customized the **bosinst.data** file in your installation media to specify a nonprompted installation, the installation and maintenance screens are not displayed. The system instead reboots from the installation media using the settings already defined in the **bosinst.data** file. To access the installation and maintenance screens, override the nonprompted mode. You can do this when three zeros are displayed on the screen. When you observe the three zeros, type 000 (zeros) and press Enter at the terminal.

   You can select 88 to display help on this or any subsequent screen.

   After you have selected the **Start Maintenance Mode for System Recovery** option, the Maintenance screen displays.
6. Select option 1, **Access a Root Volume Group**, from the Maintenance screen. The Warning screen displays.
7. Read the information displayed on the Warning screen. When you are ready to continue, type 0 and press Enter. The Access a Root Volume Group screen displays.
8. Select the option for the root volume group whose logical volume information you want to display. The Access a Root Volume Group screen lists all of the volume groups (root and otherwise) on your system. After entering your selection, the Volume Group Information screen displays.

   Note: Reviewing the disk and location code information on the Volume Group Information screen enables you to determine whether the volume group you selected was the root volume group. You can return to the Access a Root Volume Group screen if the choice you made was not the root volume group. If you have not chosen a root volume group, you cannot continue beyond the Volume Group Information screen.
9. Select one of the options from the Volume Group Information screen and press Enter. Each option does the following:

Choice 1	**Access this volume group and start a shell.** Selecting this choice imports and activates the volume group and mounts the file systems for this root volume group before providing you with a shell and a system prompt.
Choice 2	**Access this volume group and start a shell before mounting file systems.** Selecting this choice imports and activates the volume group and provides you with a shell and system prompt before mounting the file systems for this root volume group.
Choice 99	Typing 99 returns you to the Access a Root Volume Group screen.

   After you select either choice 1 or 2, a shell and system prompt display.
10. Take appropriate measures to recover data or take action (such as using the **bosboot** command) to enable the system to boot normally.

Troubleshooting a Full /usr File System

To release space in a full **/usr** file system, complete one or more of the following tasks:

- Type **installp -c all** to commit all updates and release space in the **/usr** file system.
- If the system is not a Network Installation Management (NIM) system serving a Shared Product Object Tree (SPOT), enter /usr/lib/instl/inurid -r to remove client information for **root** file system

installations. For information about NIM and SPOTs, see “SPOT (Shared Product Object Tree) Resource” on page 278 in the NIM Resources section.

- Remove software that you do not need. See “Maintaining Optional Software Products and Service Updates” on page 87.

Viewing BOS Installation Logs

Information saved in BOS installation log files may help you determine the cause of installation problems. To view BOS installation log files, type `cd /var/adm/ras` and view the files in this directory. One example is the **devinst.log**, which is a text file that can be viewed with any text editor or paged.

Viewing Logs with SMIT

To view some logs in the **/var/adm/ras** directory, you can use the following SMIT fast path:

```
smit alog_show
```

The resulting list contains all logs that are viewable with the **alog** command. Select from the list by pressing the F4 key.

Viewing Logs with the alog Command

To view some logs in the **/var/adm/ras** directory, type:

```
alog -o -f bosinstlog
```

Handling System and Error Messages

This section lists messages that can appear during the installation of AIX 5.1 and later. Information about most messages is provided in the following format:

System Message	The system message is displayed in **bold** type.
Explanation	Describes what is likely to have caused the system message to be displayed.
System Action	Describes what the system does after the message is displayed.
User Action	Suggests a possible resolution to the problem suggested by the system message.

Note: Multiple messages can have the same explanation, system action, and user action.

0516-404 allocp: Not enough resources available to fulfill allocation. Either not enough free partitions or not enough physical volumes to keep strictness. Try again with different allocation characteristics.

0516-788: extendlv: Unable to extend logical volume

0503-008 installp: There is not enough free disk space in file system /usr (506935 more 512-byte blocks are required.) An attempt to extend this file system was unsuccessful. Make more space available, then retry this operation.

Explanation	There is not enough space to complete the installation.
System Action	The installation cannot begin until the problem is resolved.

User Action	You have several options: • Select fewer filesets than the number originally selected for installation. OR • Extend the root volume group to another disk. Type: `extendvg rootvg hdisk`*`Number`*, where *Number* is the number of the specified disk. OR • Remove user-defined file systems to release space in the **rootvg** file system. OR • Follow the instructions in "Troubleshooting a Full /usr File System" on page 165.

BOS Install: After saving all the data from the previous system into /tmp, it was discovered that there will not be enough free space in /tmp to make the boot image. Please reboot in normal mode and increase the size of /tmp or reduce the number of files to save as listed in the /etc/preserve.list file.

Explanation	During a preservation installation, files listed in the**/etc/preserve.list** file were copied to the **/tmp** file. After doing so, there was not enough room in **/tmp** to create the boot image.
System Action	Installation cannot continue.
User Action	Reboot in normal mode and increase the size of **/tmp** or reduce the number of files to be saved.

BOS Install: You chose to create logical volumes mapped exactly as they were on the previous disks, but there are no map files specified in the image.data file.

Explanation	On system backup restore, **EXACT_FIT = yes** was specified in the **image.data** file, but no map files were specified in the **image.data** file.
System Action	Nonprompted mode is terminated. The user is prompted.
User Action	Run the **mkszfile** command with the **-m** option before creating the system backup tape. OR Do not specify **EXACT_FIT = yes** in the **image.data** file.

The boot logical volume (`hd5`) must be at least 12 MB. The system you are installing has a boot logical volume smaller than this, and the system does not have enough free contiguous physical partitions on *diskname* to increase the size of the boot logical volume. Please reboot in normal mode and correct this problem, or restart the installation and choose an overwrite install. Use the lspv -M *diskname* command to see the current allocation map of the disk.

OR

Error: No space available to create a larger boot logical volume. In order to proceed with this installation the size of the boot logical volume (`hd5`) must be increased to 12 MB. At this time there are not *N* contiguous physical partitions available on the boot disk (*diskname*) for recreating the larger boot logical volume. You must free up this space by removing or relocating one or more logical volumes or file systems from *diskname*. Use lspv -M *diskname* to see its current partition allocation map.

Explanation	Starting with AIX 5.2, the boot logical volume (**blv**), logical volume hd5, must be greater than 12 megabytes. If your system had disks less than 4 gigabytes in size in the root volume group, or was originally installed with a version of AIX earlier than AIX 4.3.2 your boot logical volume may only be 4 megabytes. You might experience this failure during preservation or migration installations. Overwrite installations create the boot logical volume with a minimum size of 12 megabytes. If free partitions contiguous to hd5 are available or if another location on the disk contains hd5 is identified, the installation process increases the size of hd5 and continues. Only the disk that currently contains the boot logical volume is checked for additional partitions in order to increase the size of the boot logical volume. Other disks in the rootvg are not checked.
System Action	You will be prompted to reboot in normal mode from the existing rootvg and increase the boot logical volume, or restart the installation and choose an overwrite install.

User Action	Only a system administrator with root authority should attempt to increase the boot logical volume. To increase the boot logical volume, follow the process described below:
	
	If you received this error, then your partition size is less than 8 megabytes, and you must increase the number of partitions in hd5 (boot logical volume). You can check your partition size as follows:
	1. Type the following:
	`# lsvg rootvg`
	2. Look for the field: PP SIZE:
	3. Obtain the current number of partitions in hd5, as follows:
	`# lslv hd5`
	4. Look for the field: LPs:
	5. Your boot logical volume must contain enough partitions such that:
	• PP SIZE multiplied by LPs is greater than or equal to 8.
	• The partitions for the boot logical volume must be contiguous.
	
	If there were free partitions available next to hd5 or at some other location on the disk that contains hd5, the installation process would have increased the size of hd5, and continued.
	
	To view the current allocation map (free and used partitions) of a disk, use the command:
	`# lspv -M diskname`

User Action, continued	If there are not enough contiguous free partitions, you must increase the size of the boot logical volume (hd5) using one of the options described below, and rerun the installation. The options for increasing the boot logical volume size are as follows:
	• If a user-created logical volume or file system follows hd5 on the disk (check the allocation map), and has free partitions, you can back up, remove, re-create, and restore the logical volume.
	• If there is another disk in the rootvg, that has enough contiguous free partitions, then you could move hd5 to the other disk with the following steps:
	1. Verify that the disk you plan to move hd5 to is bootable by using the command:
	`bosinfo -B diskname`
	– If 1 is returned, the disk is bootable.
	– If 0 is returned, the disk is not bootable.
	2. Find the free contiguous partitions you need on the other disk by viewing the allocation map with the command:
	`lspv -M diskname`
	3. Create a map file to use when re-creating hd5. For example, if you want to re-create hd5 on hdisk2, on partitions 88 and 89, use the command:
	`echo "hdisk2:88-89" > your_MAP_file`
	4. Remove the existing hd5:
	`rmlv -f hd5`
	5. Create the new hd5:
	`mklv -y hd5 -t boot -m your_MAP_file rootvg 2`
	The 2 represents the number of partitions and can vary as needed.
	Note: If the **mklv** command moves hd5 to a new location, you must run the following command:
	`echo ":C:C:C" | /usr/lpp/bosinst/blvset -d /dev/hdiskN`
	Where *C* is the message, locale, and keyboard (respectively) and *hdiskN* is the disk that contains hd5.
	6. Run the **mkboot** command to clear the boot record from the disk that previously contained hd5 (boot logical volume). For example, if hd5 was previously on hdisk0, use the command:
	`mkboot -d /dev/hdisk0 -c`
	7. Use the **bosboot** command to re-create the boot image and boot record on the new disk. For example, if hd5 was re-created on hdisk2, use the command:
	`bosboot -a -d /dev/hdisk2`

User Action, continued	8. Change the bootlist of your system to boot from the new disk. To see the current bootlist, use the command:
	`bootlist -m normal -o`
	
	OR
	If your previous `hd5` was on `hdisk0`, the output might be:
	`hdisk0`
	To change the bootlist to use `hdisk2`, use the command:
	`bootlist -m normal hdisk2`
	If there were additional items in your bootlist, add them after `hdisk2`, with spaces separating each item.
	9. If there were no errors, reboot your system.
	10. If you encountered this error when installing a **mksysb** on a system other than the system it was created on (cloning), then you might be able to use a customized **image.data** file to increase the size of `hd5`.
	The **vg_data** stanza contains the size of the physical partitions in the **PPSIZE** field. Use this information to determine how many partitions are needed for `hd5`. The **lv_data** stanza for `hd5` contains the fields for the number of logical partitions (**LPs**), the number of physical partitions (**PP**), and the minimum number of logical partitions required for the logical volume (**LV_MIN_LPS**). These fields must be set to the number of partitions needed.
	See "Create and Use a Supplementary Diskette" on page 68 for information on putting an **image.data** file on diskette and a **bosinst.data** file.
	If the source machine had no free partitions, and the target machine has the same disk size, then you might need to install using the shrink option, as well as the customized **image.data** file.

BOS Install: Could not create boot image.

Explanation	The **bosboot** command failed.
System Action	The boot image was not created.
User Action	Check the **/var/adm/ras/bosinst.log** file for errors (**alog -o -f bosinst.log** I **pg**). This log is updated by appending, so make sure you check the last entry.

The bosinst.data file does not specify any bootable disks.

Explanation	The **bosinst.data** file does not specify any bootable disks.
System Action	Nonprompted mode is terminated. The user is prompted.
User Action	When the system prompts, select bootable disks to install on. OR Add a bootable disk to the **bosinst.data** file target_disk_data stanzas.

The bosinst.data file specified doing a migration install, but there is no existing root volume group of level 4.2, 4.3, or 5.1.

Explanation	A BOS installation method of **migration** was specified in the **bosinst.data** file, but the existing volume group is at level 3.1 or 5.2.
System Action	This error only occurs during a nonprompted BOS installation. The installation menus are displayed.

User Action	Respond to the menu prompts to complete the installation.

The bosinst.data file specified doing either a migration or a preservation install, but there is no existing root volume group.

Explanation	A BOS installation method of **migrate** or **preserve** was specified in the **bosinst.data** file, but no root volume group was found.
System Action	This error only occurs during a nonprompted BOS installation. The installation menus are displayed.
User Action	Respond to the menu prompts to complete the installation.

The data file did not specify enough disk space to contain the operating system.

Explanation	Nonprompted mode was specified, and there were not enough disks specified in the **bosinst.data** file to hold the operating system.
System Action	Nonprompted mode is terminated. The user is prompted.
User Action	When the system prompts, select disks to install on. OR Add more **target_disk_data** stanzas to **bosinst.data** file.

Duplicate lv_data stanzas specified in the image.data file. The installation cannot continue because data may be lost.

Explanation	An **lv_data** stanza was duplicated in the **image.data** file.
System Action	Installation cannot continue.
User Action	Correct the problem and try the installation again.

Duplicate fs_data stanzas specified in the image.data file. The installation cannot continue because data may be lost.

Explanation	An **fs_data** stanza was duplicated in the **image.data** file.
System Action	Installation cannot continue.
User Action	Correct the problem and try the installation again.

The following disks failed the preliminary diagnostic tests: <disk name>

bosset: No hard disks can be accessed.

Explanation	The listed disks failed pretest.
System Action	The system initiated a diagnostic pretest on the specified disk.
User Action	Run full diagnostics on the specified disks.

Disks specified in bosinst.data do not define a root volume group.

Explanation	Nonprompted mode was specified. The install method was set to **preserve** or **migrate**, and the disks specified in **bosinst.data** do not define a root volume group.
System Action	Nonprompted mode is terminated. The user is prompted.

User Action	When the system prompts, select a root volume group to install on. OR Specify disks in the **bosinst.data** file that define a root volume group.

Encountered an unrecoverable error.

Explanation	The menus subsystem encountered an unrecoverable error.
System Action	The menu is restarted.
User Action	None

The image.data file contains no vg_data stanza for rootvg. The installation cannot continue.

Explanation	The **image.data** file is incomplete.
System Action	Installation cannot continue.
User Action	Use the default **image.data** file supplied with product media.

image.data has invalid logical volume data. Cannot continue.

Explanation	The system could not parse the logical volume data stanzas in the **image.data** file.
System Action	Installation cannot continue.
User Action	Use the default **image.data** file supplied with product media.

image.data has invalid file system data. Cannot continue.

Explanation	The system detected invalid file system data stanzas in the **image.data** file.
System Action	Installation cannot continue.
User Action	Use the default **image.data** file supplied with product media.

0516-366 putlvodm: Volume group rootvg is locked. Try again.

0516-788: extendlv: Unable to extend logical volume.

Explanation	You interrupted the installation of your optional software.
System Action	When an installation is interrupted, the system sometimes locks the root volume group.
User Action	Unlock the root volume group. Then attempt the installation procedure again. To unlock a root volume group: 1. Log in with root authority. 2. Type `chvg -u rootvg` 3. Type `smit_install` and attempt to install your optional software products again.

installp: An error occurred during bosboot processing.

Please correct the problem and rerun.

0301-52 bosboot: not enough file space to create: /tmp/disk.image.

OR

0301-152 bosboot: not enough file space to create: /tmp/unix.

Explanation	The **bosboot** command was unable to finish processing because of insufficient space in **/tmp**.
System Action	The bosboot process is interrupted. The error message, the amount of disk space required, and the available disk space are displayed. The disk space displayed indicates the number of 1024 KB blocks required.
User Action	Release space in the **/tmp** file system or extend the **/tmp** file system. Continue or restart the installation process. To resize the **/tmp** file system and complete the installation, do the following: 1. Note the error message preceding this one. Either the message `bosboot verification starting` or `bosboot process starting` will precede this message. 2. Change directories to **/tmp**. List the files and determine which files can be deleted. If there is sufficient space available, go to step 6. If you need to expand the **/tmp** file system, continue with this procedure. 3. Type `smit chfs` 4. Select the **/tmp** file system from the displayed list. 5. Add the additional block space required. The **smit chfs** command requires disk space to be defined in 512-KB blocks. Double the required disk space displayed in the system message. 6. If the message `installp: An error occurred during bosboot processing` was displayed after the message `bosboot verification starting`, rerun the installation procedure. OR If the message `installp: An error occurred during bosboot processing` was displayed after the message `bosboot process starting`, enter `installp -C`. 7. Continue the installation process.

installp: An error occurred during bosboot processing.

Please correct the problem and rerun.

301-155 bosboot: Invalid or no boot device specified.

Explanation	A device specified with the **bosboot -d** command is not valid. The **bosboot** command was unable to finish processing because it could not locate the required boot device. The **installp** command calls the **bosboot** command with **/dev/ipldevice**. If this error does occur, it is probably because **/dev/ipldevice** does not exist. **/dev/ipldevice** is a link to the boot disk.
System Action	The bosboot process is interrupted.

User Action	Determine if the link to the boot device is missing or incorrect, correct the error and complete the installation process. To identify the boot device and complete the installation: 1. To identify the boot disk, enter `lslv -m hd5`. The boot disk name displays. 2. Create a link between the boot device indicated and the **/dev/ipldevice** file. Enter: `ln /dev/`*`boot_device_name`*` /dev/ipldevice` (An example of *boot_device_name* is **rhdisk0**.) 3. If the message `installp: An error occurred during bosboot processing` was displayed after the message `bosboot verification starting`, rerun the installation procedure. OR If the message `installp: An error occurred during bosboot processing` was displayed after the message `bosboot process starting`, enter `installp -C`. Continue the installation process.

Missing image.data file. The tape does not contain a valid install image.

Explanation	The system could not find an **image.data** file.
System Action	Installation cannot continue.
User Action	The most likely cause of this error is the tape is bad. Try a different tape.

0512-0016 mksysb: Attempt to create a bootable tape failed: bosboot -d /dev/device -a failed with return code xxx.

OR

0512-0016 mksysb: Attempt to create a bootable tape failed: mkinsttape /dev/device failed with return code xxx.

Explanation	The *xxx* return code indicates the error: **5 OR 1** Not enough space in one or more of three file systems: • **/** must have at least 500 1KB blocks. • **/tmp** must have at least 7400 1KB blocks. • **/usr** must have at least 4000 1KB blocks. **11** Defective tape. **42 OR 45** Either the **/usr/lib/boot/unix** file is corrupted (may be 0 length) or the link to **/unix** is missing. **48** Cannot write to the tape drive or cannot read **/dev/blv**. This is probably caused by an incorrect density setting for the tape drive. It could also be caused by either a hardware problem with the tape drive or by dirty heads on the drive.
System Action	The **mksysb** command failed to make a bootable tape.

User Action	The return code *xxx* indicates the action required: **5 OR 1** Check the **/**, **/tmp**, and **/usr** file systems and create more space as required. **11** Replace the defective tape. **42 OR 45** Either restore the **/usr/lib/boot/unix** file from the original tape or create the missing link. **48** Check the tape drive settings and clean the heads.

There are no disks available on this system.

Explanation	No hard disks are configured on the system. Consequently, the only functioning menu option is the maintenance option.
System Action	Installation cannot begin until the problem is resolved.

User Action	You have several options:
	• Select **Maintenance** (option 3) from the Welcome to Base Operating System Install Menu, and select the **Limited Function Maintenance Shell**. Verify that no disks were configured by entering the following command:
	`lsdev -Cc disk`
	To determine if there were configuration errors, enter the command:
	`cfgmgr -v 2>1 | tee /tmp/cfgmgr.out`
	You can use the **cat** command to view the **/tmp/cfgmgr.out** file, and look specifically for errors in configuration of disks. The file can be copied to diskette media using either the **dd** or **pax** commands, and moved to a running system for ease of viewing.
	OR
	• Turn off the system and check the following on systems with SCSI devices:
	– Check all SCSI devices to ensure that all SCSI addresses are unique.
	– Make sure the SCSI cards are properly terminated.
	– If external SCSI devices are in use, make sure that the SCSI chain is terminated and that the devices are turned on.
	– Check the SCSI cabling and connections.
	– Reboot and attempt the installation again.
	OR
	• Turn off the system and check the following on systems with IDE devices:
	– Check all IDE devices to ensure that all IDE master and slave settings are unique per controller. If only one IDE device is connected to a controller, it must be set to master. If an ATA device (disk) and an ATAPI device (CD-ROM or tape) are connected to the same controller, the ATA device must be set to the master device and the ATAPI device must be set as the slave device.
	– Check the IDE cabling and connections.
	– Reboot and attempt the installation again.
	OR
	• Boot from the diagnostics and check the hard disks.
	OR
	• Follow your local problem-reporting procedures.

There are no disks on this system which can be booted.

Explanation	The system could not find any bootable disks on the system.
System Action	Installation cannot continue.

User Action	Some third-party disks are not bootable. If a disk should be bootable but is not, run diagnostics.

You chose to install only onto disks in the existing root volume group and those not in any volume group. There are not enough of those disks to contain the mksysb image.

Explanation	The EXISTING_SYSTEM_OVERWRITE field in **bosinst.data** was set to **yes**, and prompt was set to **no**, and there were not enough disks on the system that contained the root volume group or contained no volume group.
System Action	Nonprompted mode is terminated. The user is prompted.
User Action	Use **target_disk_data** stanzas to specify the disks to install on, set SHRINK to yes in the image.data file, or at the BOS Install prompt set the EXISTING_SYSTEM_OVERWRITE in the **bosinst.data** file to **any**. This allows any disks to be used for the installation. **Attention:** If EXISTING_SYSTEM_OVERWRITE is set to **any**, user volume groups might be overwritten. OR When the system prompts, select disks on which to install or select to shrink the file systems.

You chose to install only onto disks which are not contained in a volume group, but there are not enough of those disks to contain the mksysb image.

Explanation	The EXISTING_SYSTEM_OVERWRITE field in **bosinst.data** was set to **no**, and prompt was set to **no**, and there were not enough disks on the system that contained a volume group.
System Action	Nonprompted mode is terminated. The user is prompted.
User Action	If you want the system to select the disk to install on, use the **target_disk_data** stanzas to specify the target disks and set the appropriate setting for EXISTING_SYSTEM_OVERWRITE, leave EXISTING_SYSTEM_OVERWRITE blank in the **bosinst.data** file, or set SHRINK to yes in the **image.data** file and retry the installation. OR When the system prompts, select disks on which to install.

0505-113 alt_disk_install: No target disk name provided.

Explanation	This message is displayed in the following situations: • You did not enter a target disk. • The disk that was specified as the target disk has a volume group already associated with it. Running the **lspv** command should show the word `None` by disks that do not have a volume group associated with them, which is what the **alt_disk_install** command checks. • The target disk (or disks) specified are not bootable. The **alt_disk_install** command runs **bootinfo -B** *disk_name* on each disk specified in the target disk list. If any one **bootinfo -B** command returns a **0**, then the disk is not bootable, and it cannot be used as a target disk for the **alt_disk_install** operation.

0505-117 alt_disk_install: Error restoring image.data file from mksysb image.

Explanation	This message is displayed when you are trying to install a **mksysb** image from tape. The **alt_disk_install** command first checks the second image on the tape for a **./tapeblksz** file, which contains the block size in which the **mksysb** image was created. The **mksysb** command creates this file and puts it in the second image on the tape. The first three images of a **mksysb** tape are always created at a 512-byte block size. The **mksysb** image (the fourth image on the tape) can be created at another block size. If the **alt_disk_install** command cannot restore the **./tapeblksz** file from the second image, the block size will remain what it was when the **alt_disk_install** command was started. It will attempt to restore the **./image.data** file from the **mksysb** image. If this block size does not match the block size in which the **mksysb** image was created, the restore fails, and the **alt_disk_install** command produces this error.

Part 3. Network Installation

Part 3 provides information about installing and configuring AIX in a networked environment using the Network Installation Management (NIM) environment.

© Copyright IBM Corp. 2002, 2004

Chapter 16. What's New in Network Installation Management (NIM)?

This chapter provides an overview of the new features in the Network Installation Management (NIM) interface for AIX 5.2.

Working with EZNIM

The SMIT EZNIM menu helps the system administrator by organizing the commonly used NIM operations and simplifies frequently used advanced NIM operations.

Features of SMIT EZNIM include:

- Task-oriented menus
- Automatic resource naming that includes the level of the software used to create NIM resources.
- The user can review what steps will take place *before* executing a task, whenever possible.

For information about EZNIM, see Chapter 19, "EZNIM," on page 187.

Creating Resources Simultaneously

Previously, when NIM ran a process that calculated and consumed file system space, such as creating a **SPOT**, **lpp_source** and **mksysb** resource, it would lock a server for the duration of the process. Because these operations calculate free space and enlarge the size of a file system, NIM limited servers to one of these operations at a time. Other operations could occur on the locked server that did not interfere with the operation that was calculating free space and enlarging a file system.

Beginning in AIX 5.2, you can use NIM to simultaneously create multiple **lpp_source** and **mksysb** resources in separate file systems on the same server. You cannot simultaneously create multiple **SPOT** resources, but you can simultaneously create a **SPOT**, **lpp_source**, and **mksysb** resource. The locking mechanism is set for each file system instead of for each server.

However, if you know that you have enough space in a file system to create multiple resources simultaneously, you can use the force option (**-F** flag) to prevent the locking mechanism from being used. This allows you to create multiple resources simultaneously in the same file system.

LPP Source Enhancements

Enhancements to the management of **lpp_source** resources include the following:

- The **lpp_source** resource is no longer a *required* for mksysb installations.
- The **update** operation, which allows you to update an **lpp_source** resource by adding and removing packages. Previously, you could copy packages into an **lpp_source** directory or remove packages from an **lpp_source** directory and run **nim -o check** to update the **lpp_source** attributes. Previously, SMIT allowed you to add packages to a **lpp_source** through the **smit nim_bffcreate** fast path. However, this SMIT function does not check to see if the **lpp_source** is allocated or locked, nor does it update the **simages** attribute when finished.

  The **update** operation has been created to address this situation. For more information, see "update" on page 300.
- The **lppmgr** operation is available to help you manage your **lpp_source** resources. The **lppmgr** operation is available through the command line and through SMIT. For more information, see "lppmgr" on page 295.

© Copyright IBM Corp. 2002, 2004

Resource Groups

Resource groups allow you to create a group of resources and specify clients (or a group of clients) as defaults. Previously, every time a NIM operation occurred, the NIM resources had to be specified. A default resource group can be created containing the required resources, so when a NIM operation is performed, the resources are already associasted to the client by the default resource group.

NIM Commands

The following NIM commands have been added to AIX:

- The **nim_master_setup** command. For further information, see “Network Installation Management Commands Reference” on page 212.
- The **nim_clients_setup** command. For further information, see “Network Installation Management Commands Reference” on page 212.

Chapter 17. Network Installation Management Introduction

This chapter provides an introduction to AIX Network Installation Management (NIM) and the operations you can perform to manage the installation of the Base Operating System (BOS) and optional software on one or more machines.

The types of machines you can manage are *standalone*, *diskless*, and *dataless*. A *standalone* machine is one that can boot (start up) by itself. Diskless and dataless systems cannot boot by themselves. They must use remote resources to boot. *Diskless* systems have no disk drive. *Dataless* systems have a local disk drive but they cannot boot from it. This section provides concepts and procedures for setting up the NIM environment, initiating the installation of standalone machines, and initializing resources for diskless and dataless machines.

Using NIM, you can install a group of machines with a common configuration or customize an installation for the specific needs of a given machine. The number of machines you can install simultaneously depends on the throughput of your network, the disk access throughput of the installation servers, and the platform type of your servers.

The NIM environment comprises client and server machines. A *server* provides resources (for example, files and programs required for installation) to another machine. A machine that is dependent on a server to provide resources is known as a *client*. In this guide and reference, any machine that receives NIM resources is a client, although the same machine can also be a server in the overall network environment.

All operations on clients in the NIM environment require one or more resources. NIM resource objects represent files and directories that are used to support some type of NIM operation. Because NIM resources are ordinary file system objects in the AIX operating system, most of them are provided to clients with standard Network File System (NFS) software. This means that many resources must reside locally on the servers providing these resources, because NFS can only export file system objects that are stored on local media in the machines from which they are exported.

Most installation tasks in the NIM environment are performed from one server, called the *master*. A set of installation tasks can also be performed from NIM clients. Once the network installation setup is complete, users of standalone clients can, from the client, install software that is available on NIM servers.

The machines you want to manage in the NIM environment, their resources, and the networks through which the machines communicate are all represented as *objects* within a central database that resides on the master. Network objects and their attributes reflect the physical characteristics of the network environment. This information does not affect the running of a physical network but is used internally by NIM for configuration information.

Each object in the NIM environment has a unique name that you specify when the object is defined. The NIM name is independent of any of the physical characteristics of the object it identifies and is only used for NIM operations. The benefit of unique names is that an operation can be performed using the NIM name without having to specify which physical attribute should be used. NIM determines which object attributes to use. For example, to easily identify NIM clients, the host name of the system can be used as the NIM object name, but these names are independent of each other. When an operation is performed on a machine, the NIM name is used, and all other data for the machine (including the host name) is retrieved from the NIM database.

For more information about NIM concepts, see Chapter 24, “Network Installation Management Concepts,” on page 257. For information on a particular NIM task, refer to Chapter 18, “NIM Task Roadmap,” on page 185.

© Copyright IBM Corp. 2002, 2004

Chapter 18. NIM Task Roadmap

The NIM Task Roadmap provides an overview of NIM configuration tasks and installation tasks and where they can be found in this guide. Also provided is a brief description of the task. Where appropriate, the SMIT fast path is provided.

Table 5. NIM Task Roadmap

NIM Task	SMIT Fast Path	Description
Chapter 19, “EZNIM,” on page 187	smit eznim	Configure the NIM enviroment using EZNIM. Allows you to configure your system as a NIM master or a NIM client. If you configure your system as a NIM master, EZNIM also creates the minimum basic installation resources.
“Configuring the NIM Master and Creating Basic Installation Resources” on page 190	smit nim_config_env	Configure the NIM master, create the minimum basic installation resources required to install NIM client machines, and manage the resources for diskless and dataless clients.
“Adding a Standalone NIM Client to the NIM Environment” on page 192	smit nim_mkmac	Describes how to add standalone clients to the NIM environment.
“Using Installation Images to Install the Base Operating System (BOS) on a NIM Client” on page 196	smit nim_bosinst	Describes how to perform a BOS installation on a NIM client.
“Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation)” on page 197	smit nim_bosinst	Describes how to restore a **mksysb** image and additional software to a target NIM client from a **mksysb** resource in the NIM environment.
“Performing a Nonprompted BOS Installation” on page 200	• smit nim_mkres • smit nim_bosinst	Provides information about how to perform a nonprompted NIM BOS installation using a **bosinst_data** resource.
“Installing to Clients on ATM Networks” on page 201		Provides information about how to configure NIM to work with ATM adapters.
“Customizing NIM Clients and SPOT Resources” on page 203	smit nim_task_inst	Describes how to use NIM to install software packages, updates, and maintenance levels on running, configured NIM clients and **SPOT** resources.
Remove Installed Software from a NIM Client or a SPOT Resource.	smit nim_remove	From the NIM master, you can uninstall software that resides on client machines. NIM also gives you the capability of committing and rejecting updates over the network.
Perform Maintenance Operations on NIM Client Machines	smit nim_mac_op	Hardware diagnostics can be performed on NIM clients using a diagnostic boot image from a NIM server, rather than booting from a diagnostic tape or CD-ROM. Not only does this eliminate the need for diagnostic boot media, it eliminates the need to have diagnostics installed on the local disks of machines. For maintenance operations, you can boot a NIM client into maintenance mode from the boot image on a NIM server instead of using a bootable tape or CD-ROM.
Perform Maintenance Operations on NIM Resources	smit nim_res	Allows you to list NIM resources; change characteristics of a resource; show the contents of a resource; and verify resources.

© Copyright IBM Corp. 2002, 2004

Table 5. NIM Task Roadmap (continued)

Software Maintenance and Utilities	smit nim_task_maint	Provides information about how to commit, reject, remove, copy, verify, and clean up software.
"Adding a Diskless or Dataless Client to the NIM Environment" on page 207	smit nim_task_dd	Provides information about how to add diskless and dataless systems to your NIM environment. You can also manage resources for diskless and dataless clients from the NIM master.
"Installing to an Alternate Disk on a NIM Client (cloning or mksysb)" on page 210	smit nim_alt_install	NIM can be used to clone the running of **rootvg** (root volume group) to an alternate disk, or install a **mksysb** image to an alternate disk.
"Alternate Disk Migration Installation" on page 149	smit nimadm	NIM can be used to perform an alternate disk migration installation to a NIM client.

Chapter 19. EZNIM

The SMIT EZNIM feature helps the system administrator by organizing the commonly used NIM operations and simplifies frequently used advanced NIM operations.

Features of SMIT EZNIM include:

- Task-oriented menus
- Automatic resource naming that includes the level of the software used to create NIM resources.
- The user can review what steps will take place *before* executing a task, whenever possible.

Use the SMIT **eznim** fast path to open the EZNIM main menu. If the NIM environment has not been set up on your system, the EZNIM main menu displays the following options:

- Configure as a NIM Master
- Configure as a NIM Client

EZNIM Master Operations

If you select **Configure as a NIM Master**, the following options display:

```
Setup the NIM Master environment
Add fixes to the NIM Master environment
Add client to the NIM environment

Update clients
Backup a client
Reinstall clients
Reset clients

Show the NIM environment
Verify the NIM environment
Remove NIM environment
```

- To configure your current system as a NIM master, select **Setup the NIM Master environment**. You can select the software source to configure from, select the volume group to use for the NIM resources, and select the file system to use for the NIM resources. When the NIM master environment is configured, the basic NIM resources are created. To view the NIM resources created by EZNIM, select **Show the NIM environment**, or run the **lsnim** command on the NIM master.
- To install updates and maintenance level packages to the NIM master, select **Add fixes to the NIM Master environment**. This option performs an update installation of a specified set of fixes onto the default **SPOT** resource. A second **SPOT** resource containing the newly installed fixes is created by this operation. You can optionally select to update all your NIM clients during this operation.
- To update a client using EZNIM, select **Update clients**. The Update clients option allows you to perform an **update_all** operation on a selected client (or clients) using an **lpp_source** resource.
- To back up a client using EZNIM, select **Backup a client**. The Backup a client option allows you to create a system backup image of a selected client and store the backup image on the NIM master.
- To reinstall a client using EZNIM, select **Reinstall clients**. The Reinstall clients option performs a **mksysb** restore on a selected client (or clients). You must select a system backup image to restore, and decide whether to reboot and install the client now.
- To reset a NIM client to the *ready* state, select **Reset clients**. This option resets the state of a client or clients in the NIM environment. Use this option after a NIM operation has failed, and you want to return the client to the *ready* state.

© Copyright IBM Corp. 2002, 2004

EZNIM Client Operations

EZNIM also allows you to manage a NIM client. On a client system, use the SMIT **eznim** fast path. Select **Configure as a NIM Client**, and the following options display:

```
Add this system to a NIM environment
Update this system
Reinstall this system
Reset this system
```

- To define your client in the NIM environment, select **Add this system to a NIM environment**.
- To update your client, select **Update this system**. This option allows you to perform an **update_all** operation on your client using an **lpp_source** resource.
- To reinstall your client, select **Reinstall this system**. This option performs a **mksysb** restore on the client. You must select a system backup image to restore, and decide whether to reboot and install the client now.
- To reset your client in the NIM environment, select **Reset this system**. This option resets the state of the client in the NIM environment. Use this option after a NIM operation has failed, and you want to return the client to the *ready* state.

Chapter 20. Basic NIM Operations and Configuration

Note: AIX 5.2 provides the **cdromd** CD and DVD automount facility, which is included in the **bos.cdmount** fileset. To determine if the **cdromd** daemon is enabled on your system, run the following command:

```
# lssrc -s cdromd
```

The **cdromd** daemon can interfere with scripts, applications, or instructions that attempt to mount the CD or DVD device without first checking to see if the device is already enabled. A resource or device busy error occurs in such a condition. Use the **cdumount** or **cdeject** command to unmount the device. Then mount the device as specified in the program or instructions. Alternatively, use the **cdcheck -m** or **mount** command to determine the current mount point of the device. For further information, see the **cdromd** command documentation in the *AIX 5L Version 5.2 Commands Reference*.

The installation code allows for this automatic mounting. If **cdromd** is enabled and the **mkcd** command is run, the CD-R or DVD-RAM is ejected after the image is completed. If you do not want to have the media ejected, then the **cdromd** daemon should be put in the inoperative state with the following command:

```
# stopsrc -s cdromd
```

This chapter describes the following procedures for performing basic NIM operations and simple configuration tasks using the Web-based System Manager NIM interface or the System Management Interface Tool (SMIT), as well as from the command line. The procedures in this chapter identify prerequisite tasks or conditions that must be met before performing the task.

- "Configuring the NIM Master and Creating Basic Installation Resources" on page 190
- "Adding a Standalone NIM Client to the NIM Environment" on page 192
- "Using Installation Images to Install the Base Operating System (BOS) on a NIM Client" on page 196
- "Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation)" on page 197
- "Performing a Nonprompted BOS Installation" on page 200
- "Installing to Clients on ATM Networks" on page 201
- "Customizing NIM Clients and SPOT Resources" on page 203
- "Configuring the NIM Master and Creating Resources to Support Diskless and Dataless Clients Only" on page 204
- "Adding a Diskless or Dataless Client to the NIM Environment" on page 207
- "Initializing and Booting a Diskless or Dataless Machine" on page 209
- "Uninitializing Diskless and Dataless Machines" on page 210
- "Installing to an Alternate Disk on a NIM Client (cloning or mksysb)" on page 210
- "Network Installation Management Commands Reference" on page 212

To start the Web-based System Manager NIM interface from an X-Windows session on the NIM master, type:

```
wsm
```

Notes:

1. For additional information about NIM operations and the required and optional attributes to customize operations, see Chapter 24, "Network Installation Management Concepts," on page 257.

© Copyright IBM Corp. 2002, 2004

2. Using an AIX Version 5 **lpp_source** to install filesets on an AIX Version 4 client through NIM, is not supported. If installing Version 5 filesets on a Version 4 system is necessary, the user can NFS export the **lpp_source**, mount it on the client, and then use the **installp** command or **geninstall** command to perform the installation procedures.
3. For assistance, use the online contextual help available for both the Web-based System Manager and SMIT interfaces.

   Extended help is available in the Web-based System Manager interface, offering guidance for tasks you may need to do in NIM. To view extended help, select **Contents** from the Help menu in the NIM container.
4. For tasks performed at the command line, the root user must be using **ksh**. Unexpected results can occur if the root user is set to another shell, such as **csh**.
5. If errors occur, it may be necessary to reset the machine before retrying the operation. For information about testing a NIM client, see “Resetting Machines” on page 216.

Configuring the NIM Master and Creating Basic Installation Resources

Using this procedure, you can configure the NIM master, create the minimum basic installation resources required to install NIM client machines, and manage the resources for diskless and dataless clients.

Note: This procedure produces a large amount of output, especially when creating the **SPOT** resource. Be sure to scan through the output to look for nonfatal errors and warnings that may not be evident from a successful return code.

Prerequisites

The NIM master must have at least 1 GB of available disk space. If such space is not available, see “Using Client Machines as Resource Servers” on page 217, and “Defining an lpp_source on CD-ROM versus Disk” on page 235.

From Web-based System Manager

1. Insert the *AIX 5.2 Volume 1* CD into the appropriate drive of the designated master machine.
2. Start the Web-based System Manager Software application by entering `wsm`.
3. In the navigation area, expand and select the NIM container, then follow the directions in the wizard.

From SMIT

1. Insert the *AIX 5.2 Volume 1* CD into the appropriate drive of the designated master machine.
2. To install the **bos.sysmgt.nim.master** fileset, enter the **smit install_latest** fast path.
3. Using the LIST option, select **/dev/cd0** for the INPUT device / directory for software.
4. Specify **bos.sysmgt.nim.master** as the SOFTWARE to install.
5. Accept the default values for all other fields on this screen. After successful completion of this installation, exit SMIT.
6. To configure the NIM master, enter the **smit nim_config_env** fast path.
7. Using the LIST option, select the Primary Network Interface for the NIM Master.
8. Using the LIST option, select **/dev/cd0** or **/dev/rmt0** for the **Input device for installation / images** field.
9. If you will be supporting diskless and dataless clients, select **yes** at the **Create Diskless/Dataless Machine Resources?** field, and supply names for the resources to be created.
10. Select **yes** at the **Remove all newly added NIM definitions and file systems if any part of this operation fails?** field. This will make it easier to restart this procedure if failures occur.
11. Accept the default values for all other fields on this screen.

Notes:

1. Depending on the speed of your machine, creating the basic NIM resources could be a lengthy process.
2. This procedure provides the capability for much more than just configuring the NIM master and creating the **lpp_source** and **SPOT** resources. However, for this simple configuration, only a subset of the available functions will be used. Advanced NIM administrators can use the SMIT screens accessed through this procedure to create a more complex environment.
3. As you develop a better understanding of configuration tasks, you may prefer to not automatically undo all configuration when failures occur (as in step 10 in the previous procedure). Continuing from the last point of failure results in faster configuration for experienced administrators.

From the Command Line

1. Insert the *AIX 5.2 Volume 1* CD into the appropriate drive of the designated master machine.
2. To install the **bos.sysmgt.nim.master** fileset from the CD, enter:

   ```
   # installp -agXd /dev/cd0 bos.sysmgt.nim.master
   ```

   To install the **bos.sysmgt.nim.master** fileset from a tape, enter:

   ```
   # installp -agXd /dev/rmt0 bos.sysmgt.nim.master
   ```

3. To configure the NIM master using the **nimconfig** command, enter:

   ```
   # nimconfig -a attr1=value1 \
   -a attr2=value2 \
   ...
   ```

   For example, to configure a NIM master with the following configuration:

   ```
   master host name = master1
   primary network interface = tr0
   ring speed = 16
   platform = chrp
   kernel type = mp
   ```

   Enter the following command sequence:

   ```
   # nimconfig -a netname=network1 -a pif_name=tr0 \
   -a ring_speed1=16 -a platform=chrp -a netboot_kernel=mp
   ```

   Note: For additional attribute information, see the **nimconfig** command.
4. To create a file system in the rootvg volume group with 400 MB of space with a mount point of **/export/lpp_source**, enter:

   ```
   # crfs -v jfs -g rootvg -a size=$((2000*400)) \
   -m /export/lpp_source -A yes -p rw -t no \
   -a frag=4096 -a nbpi=4096 -a compress=no
   ```

5. To mount the file system, enter:

   ```
   # mount /export/lpp_source
   ```

6. The **lpp_source** contains the installation images copied from the source device (in this example, the CD-ROM). The server of the **lpp_source** will be the NIM master. The images will be stored in the **/export/lpp_source/lpp_source1** directory. To create the **lpp_source** resource named **lpp_source1**, enter:

   ```
   # nim -o define -t lpp_source -a source=/dev/cd0 \
   -a server=master -a location=/export/lpp_source/lpp_source1 \
   lpp_source1
   ```

7. To create a file system in the rootvg volume group with 200 MB of space with a mount point of **/export/spot**, enter:

   ```
   # crfs -v jfs -g rootvg -a size=$((2000*200)) \
   -m /export/spot -A yes -p rw -t no \
   -a frag=4096 -a nbpi=4096 -a compress=no
   ```

8. To mount the file system, enter:

   ```
   # mount /export/spot
   ```

9. The **SPOT** resource will be installed from images in the image source (in this case, the **lpp_source** that was created in step 6). The server of the resource will be the NIM master, and the **SPOT** will be stored in the **/export/spot/spot1** directory. To create the **SPOT** resource named `spot1`, enter:

   ```
   # nim -o define -t spot -a source=lpp_source1 \
   -a server=master -a location=/export/spot spot1
   ```

10. If you are not supporting diskless and dataless clients, you do not need to continue with this procedure. If you are supporting diskless and dataless clients, create and mount a file system for their resources.

    To create a file system in the rootvg volume group with 150 MB of space and a mount point of **/export/dd_resource**, enter:

    ```
    # crfs -v jfs -g rootvg -a size=$((2000*150)) \
    -m /export/dd_resource -A yes -p rw -t no \
    -a frag=4096 -a nbpi=4096 -a compress=no
    ```

11. To mount the file system, enter:

    ```
    # mount /export/dd_resource
    ```

12. Create the diskless and dataless client resources in subdirectories of the **/export/dd_resource** directory. Not all resources are required. Create only the resources to be used in your environment.

    To create the **root** resource named **root1** (required for diskless and dataless clients), enter:

    ```
    # nim -o define -t root -a server=master \
    -a location=/export/dd_resource/root1 root1
    ```

    To create the **dump** resource named **dump1** (required for diskless and dataless clients), enter:

    ```
    # nim -o define -t dump -a server=master \
    -a location=/export/dd_resource/dump1 dump1
    ```

    To create the **paging** resource named **paging1** (required for diskless clients), enter:

    ```
    # nim -o define -t paging -a server=master \
    -a location=/export/dd_resource/paging1 paging1
    ```

    To create the **home** resource named **home1** (optional), enter:

    ```
    # nim -o define -t home -a server=master \
    -a location=/export/dd_resource/home1 home1
    ```

    To create the **shared_home** resource named **shared_home1** (optional), enter:

    ```
    # nim -o define -t shared-home -a server=master \
    -a location=/export/dd_resource/shared_home1 shared_home1
    ```

    To create the **tmp** resource named **tmp1** (optional), enter:

    ```
    # nim -o define -t tmp -a server=master \
    -a location=/export/dd_resource/tmp1 tmp1
    ```

Notes:

1. The file systems created for the NIM resources are not required, but they can be beneficial for storage management.
2. A **SPOT** resource is required for supporting diskless and dataless clients. The **SPOT** created in step 13 can be used for this purpose, so there is no need to create another **SPOT** specifically for diskless and dataless clients.
3. For more information about NIM resources, see “NIM Resources” on page 267.

Adding a Standalone NIM Client to the NIM Environment

This procedure describes how to add standalone clients to the NIM environment. Standalone clients are machines that, once installed, can obtain a boot image and mount all file systems from the local hard disk, unlike diskless and dataless clients which depend on remote servers.

From one of the following interfaces, use Method A if the client machine is not running or if the client does not have AIX installed. Method A can also be used if BOS is to be installed on the client and the client is to be network-booted manually or to initiate the installation from a **force-push** operation. This procedure automatically adds NIM networks when needed.

From one of the following interfaces, use Method B if the client machine has AIX already installed.

If the NIM client being defined is on a network that is not currently defined in the NIM environment, the **niminit** command will fail. If this is the case, use "Method A:" of this procedure to define the client on the NIM master, and then follow the steps in "Method B:" to complete the configuration.

Prerequisites

1. The NIM master must be configured. For more information, see "Configuring the NIM Master and Creating Basic Installation Resources" on page 190.
2. You must know the subnet mask, the default gateway for the client machine, and the default gateway for the NIM master.

From Web-based System Manager

1. To start the Web-based System Manager Software application, type:

   ```
   wsm
   ```
2. In the navigation area, expand and select the NIM container.
3. Select the Machines container.
4. From the Machines menu, select **New** —> **Machine**.
5. Use the wizard to complete the task.

From SMIT

To add a standalone NIM client to the NIM environment using SMIT, choose between the following methods:

Method A:

1. On the NIM master, add a standalone client to the NIM environment by typing the **smit nim_mkmac** fast path.
2. Specify the host name of the client.
3. The next SMIT screen displayed depends on whether NIM already has information about the client's network. Supply the values for the required fields or accept the defaults. Use the help information and the LIST option to help you specify the correct values to add the client machine.

Method B:

1. On a system that you have chosen to be a NIM client, verify that if the **bos.sysmgt.nim.client** fileset is installed by typing the following:

   ```
   # lslpp -L bos.sysmgt.nim.client
   ```
2. If the **bos.sysmgt.nim.client** fileset is not installed, then install the fileset from the **AIX Volume 1** CD by typing the following:

   ```
   # installp -acXd /dev/cd0 bos.sysmgt.nim.client
   ```
3. Enter the **smit niminit** fast path.
4. Supply the values for the required fields or accept the defaults. Use the help information and the LIST option to help you specify the correct values for defining your client machine.

Note: In AIX 5.2, if the LIST option is used to display valid platforms for the client definition, only **chrp** is displayed as a selectable platform.

From the Command Line

To add a standalone NIM client to the NIM environment from the command line, choose between the following methods:

Method A:

On the NIM master, type:

```
# nim -o define -t standalone -a platform=PlatformType \
-a netboot_kernel=NetbootKernelType \
-a if1=InterfaceDescription \
-a net_definition=DefinitionName -a ring_speed1=SpeedValue \
-a cable_type1=TypeValue -a iplrom_emu=DeviceName MachineName
```

Example 1:

To add the machine with host name `machine1` with the following configuration:

```
host name=machine1
platform=chrp
kernel=up
network type=ethernet
subnet mask=255.255.240.0
default gateway=gw1
default gateway used by NIM master=gw_master
cable type=bnc
network boot capability=yes (no emulation needed)
```

enter the following command sequence:

```
# nim -o define -t standalone -a platform="chrp" \
-a netboot_kernel="up" -a if1="find_net machine1 0" \
-a cable_type1="bnc" \
-a net_definition="ent 255.255.240.0 gw1 gw_master" machine1
```

Example 2:

To add the machine with host name `machine2` with the following configuration:

```
host name=machine2
platform=chrp
netboot_kernel=up
network type=token ring
subnet mask=255.255.225.0
default gateway=gw2
default gateway used by NIM master=gw_master
ring speed=16
```

enter the following command sequence:

```
# nim -o define -t standalone -a platform="chrp" \
-a netboot_kernel="up" -a if1="find_net machine2 0" \
-a ring_speed1="16" \
-a net_definition="tok 255.255.225.0 gw2 gw_master" machine2
```

Notes:

1. If the **find_net** keyword in the **if** attribute causes NIM to successfully match a network definition to the client definition, the **net_definition** attribute is ignored.
2. For more information about the attributes you can specify when defining NIM clients, see “Defining NIM Clients” on page 258.

Method B:

1. Install the **bos.sysmgt.nim.client** fileset on the client machine.
2. From the machine being defined as a client, enter:

```
# niminit -a name=ClientDefinitionName -a master=MasterName \
-a pif_name=Interface -a platform=PlatformType \
-a netboot_kernel=NetbootKernelType -a ring_speed1=SpeedValue \
-a cable_type1=TypeValue -a iplrom_emu=DeviceName
```

Note: For detailed attribute information, see the **niminit** command.

Example 1:

To add the machine with host name `machine1` with the following configuration:

```
host name=machine1
NIM master's host name=master_mac
primary interface adapter=en0
platform=chrp
kernel=up
cable type=bnc
network boot capability=yes (no emulation needed)
```

enter the following command sequence:

```
# niminit -a name=machine1 -a master=master_mac \
-a pif_name=en0 -a platform=chrp -a netboot_kernel=up \
-a cable_type1=bnc
```

Example 2:

To add the machine with host name `machine2` with the following configuration:

```
host name=machine2
NIM master's host name=master_mac
primary interface adapter=tr0
platform=chrp
netboot_kernel=up
ring speed1=16
```

enter the following command sequence:

```
# niminit -a name=machine2 -a master=master_mac \
-a pif_name=tr0 -a platform=chrp -a netboot_kernel=up \
-a ring_speed1=16
```

Verifying the Status of Your Client Machine

To verify that the **niminit** command completed successfully, enter the following command at the NIM client:

```
# nimclient -l -l MachineObjectName
```

Note: There is neither a Web-based System Manager application nor a SMIT menu to do this task.

The system returns output similar to the following:

```
Standalone2:
   class          =  machines
   type           =  standalone
   Cstate         =  ready for a NIM operation
   platform       =  chrp
   netboot_kernel =  up
   if1            =  Network2 standalone2 08005acd536d
   cable_type1    =  bnc
   iplrom_emu     =  /dev/fd0
   prev_state     =  customization is being performed
   cpuid          =  000247903100
   Mstate         =  currently running
   Cstate_result  =  success
```

If the system output to this query indicates any errors, you must validate all of your data, checking for accurate spelling, nonduplication of NIM names, and so forth, and redo the **niminit** operation.

Be sure to coordinate this operation with the system administrator of the NIM master, and ensure that *all* NIM object names are unique in the entire NIM environment.

Using Installation Images to Install the Base Operating System (BOS) on a NIM Client

Using installation images to install BOS on a NIM client is similar to the traditional BOS installation from a tape or CD-ROM device, because the BOS image is installed from the installation images in the **lpp_source** resource.

Prerequisites

- The NIM master must be configured, and **lpp_source** and **SPOT** resources must be defined. See "Configuring the NIM Master and Creating Basic Installation Resources" on page 190.
- The NIM client to be installed must already exist in the NIM environment. To add the client to the NIM environment, see "Adding a Standalone NIM Client to the NIM Environment" on page 192.

From Web-based System Manager

1. From the NIM container, select the Machines container.
2. In the contents area, select a target standalone machine for the installation.
3. From the Selected menu, choose **Install Operating System**.
4. If the client machine being installed is not already a running, configured NIM client, NIM will not automatically reboot the machine over the network for installation. If the client was not rebooted automatically, initiate a network boot from the client to install it. For information about performing a network boot, see "Booting a Machine Over the Network" on page 343.
5. After the machine boots over the network, the display on the client machine will begin prompting for information about how to configure the machine during installation. Specify the requested information to continue with the installation.

   Note: To perform a nonprompted installation, follow the instructions in "Performing a Nonprompted BOS Installation" on page 200 to complete the prerequisite tasks.

From SMIT

1. To install BOS on a NIM client using an **rte** installation, type `smit nim_bosinst` from the NIM master.
2. Select the TARGET for the operation.
3. Select **rte** as the installation TYPE.
4. Select the SPOT to use for the installation.
5. Select the LPP_SOURCE to use for the installation.
6. In the displayed dialog fields, supply the correct values for the installation options or accept the default values. Use the help information and the LIST option to help you.
7. If the client machine being installed is not already a running, configured NIM client, NIM will not automatically reboot the machine over the network for installation. If the client was not rebooted automatically from SMIT, initiate a network boot from the client to install it. Use the procedure for "Booting a Machine Over the Network" on page 343 to initiate the network boot.
8. After the machine boots over the network, the display on the client machine will begin prompting for information about how the machine should be configured during installation. Specify the requested information to continue with the installation.

Note: To perform a nonprompted installation, follow the instructions in “Performing a Nonprompted BOS Installation” on page 200 to complete the prerequisite tasks.

From the Command Line

1. To initiate the **bos_inst** operation, type:

   ```
   # nim -o bos_inst -a source=rte -a lpp_source=Lpp_Source \
   -a spot=SPOTName -a accept_licenses=yes -a boot_client=yes/no ClientName
   ```

   Specify the resources to be used to support the installation and any additional options for customizing the installation. To perform a simple **rte** installation, specify the **lpp_source** and **SPOT** resources.

   If the client machine being installed is not already a running, configured NIM client, NIM will not automatically reboot the machine over the network for installation. A network boot must be performed manually on the machine. If that is the case, supply the **boot_client=no** attribute to the **bos_inst** command. If the **boot_client** attribute value is not specified, it defaults to **boot_client=yes**.
2. If the client was not rebooted automatically, initiate a network boot from the client to install it. Follow the “Booting a Machine Over the Network” on page 343 procedure to perform the network boot.
3. After the machine boots over the network, the display on the client machine will begin prompting for information about how to configure the machine during installation. Specify the requested information to continue with the installation.

   Example

   The client machine, `machine1`, is not a running, configured NIM client. You should specify **boot_client=no**. To install the client using the **lpp_source** named `lpp_source1` and the **SPOT** named `spot1`, enter:

   ```
   # nim -o bos_inst -a source=rte -a lpp_source=lpp_source1 \
   -a spot=spot1 -a accept_licenses=yes -a boot_client=no machine1
   ```

   Notes:

   a. The steps to perform an **rte** installation are almost identical to the steps to perform other types of BOS installations. The main difference is that **rte** must be specified in the **source** attribute of the **nim bos_inst** command.

   b. To perform a nonprompted installation, follow the instructions in “Performing a Nonprompted BOS Installation” on page 200 to complete the prerequisite tasks.

   c. For a complete description of the different ways that a BOS installation can be customized by NIM, see “bos_inst” on page 285.

Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation)

A **mksysb** installation restores BOS and additional software to a target from a **mksysb** image in the NIM environment.

For a complete description of different ways to customize a BOS installation using NIM, see “bos_inst” on page 285.

Prerequisites

1. The NIM master must be configured, and **SPOT** and **mksysb** resources must be defined. See “Configuring the NIM Master and Creating Basic Installation Resources” on page 190.
2. The NIM client to be installed must already exist in the NIM environment. To add the client to the NIM environment, see “Adding a Standalone NIM Client to the NIM Environment” on page 192.
3. The **mksysb** must be available on the hard disk of the NIM master or a running NIM client, or the **mksysb** image is created during this procedure from either the NIM master or a running NIM client.

4. The **SPOT** and **mksysb** resources should be at the same level of AIX when used for NIM BOS installations.
5. Many applications, particularly databases, maintain data in *sparse files*. A sparse file is one with empty space, or gaps, left open for future addition of data. If the empty spaces are filled with the ASCII null character and the spaces are large enough, the file will be sparse, and disk blocks will not be allocated to it.

   This situation creates an exposure in that a large file will be created, but the disk blocks will not be allocated. As data is then added to the file, the disk blocks will be allocated, but there may not be enough free disk blocks in the file system. The file system can become full, and writes to any file in the file system will fail.

   It is recommended that you either have no sparse files on your system or that you ensure you have enough free space in the file system for future allocation of the blocks.

Cloning Considerations

The **mksysb** images enable you to clone one system image onto multiple target systems. The target systems might not contain the same hardware devices or adapters, require the same kernel (uniprocessor or multiprocessor).

Because NIM configures TCPIP at the end of an installation, it is recommended that a **bosinst_data** resource be allocated for cloning **mksysb** installations with the **RECOVER_DEVICES** field set to `no`. This will prevent the BOS installation process from attempting to configure the devices as they were on the source machine of the **mksysb** image.

Beginning in AIX 5.2, devices are not recovered if the **mksysb** image that is being installed was not created on the same system.

Attention: If the system you have cloned is using OpenGL or graPHIGS, there may be some device filesets from these LPPs that must be installed after a clone. OpenGL and graPHIGS have graphics adapter-specific filesets, so if you cloned onto a system with a different graphics adapter, you will need to create a bundle as follows:

```
echo OpenGL.OpenGL_X.dev > /usr/sys/inst.data/user_bundles/graphic_dev.bnd
echo PEX_PHIGS.dev >> /usr/sys/inst.data/user_bundles/graphic_dev.bnd
```

You can allocate this bundle when you install the **mksysb**, and the device filesets will be installed automatically if OpenGL and graPHIGS are in your **lpp_source**.

From Web-based System Manager

1. In the NIM container, select the Resources container.
2. From the Resources menu, select **New —> Resource —> New Resources**.
3. Use the wizard to complete the task.

Note: To perform a nonprompted installation, follow the instructions in “Performing a Nonprompted BOS Installation” on page 200 to complete the prerequisite tasks.

From SMIT

1. If the **mksysb** resource has already been created, skip to step 6. Otherwise, to create the **mksysb** resource, enter the **smit nim_mkres** fast path.
2. Select **mksysb** from the list of resource types that can be defined.
3. In the displayed dialogs, supply the values for the required fields. Use the help information and the **LIST** option to help you specify the correct values for defining your **mksysb** resource.
4. If the **mksysb** image does not exist, create it by supplying the values for the fields under **System Backup Image Creation Options**.

Note: If the **mksysb** image already exists as a file on the hard disk of the NIM master or client, no additional information is needed to define your **mksysb** resource.

5. Upon successful completion of this task, exit SMIT.
6. To use the **mksysb** resource to install a NIM client, enter the **smit nim_bosinst** fast path.
7. Select a TARGET for the operation.
8. Select **mksysb** as the installation TYPE.
9. Select the MKSYSB to use for the installation.
10. Select the SPOT to use for the installation.
11. In the displayed dialog fields, supply the correct values for the installation options or accept the default values. Use the help information or the LIST option to help you.
12. Run the SMIT dialog to install the NIM client.
13. If the client machine being installed is not already a running, configured NIM client, NIM will not automatically reboot the machine over the network for installation. If the client was not rebooted automatically from SMIT, initiate a network boot from the client to install it. For information about to initiating a network boot, see “Booting a Machine Over the Network” on page 343.
14. After the machine boots over the network, the display on the client machine will begin prompting for information about how to configure the machine during installation. Specify the requested information to continue with the installation.

    Note: To perform a nonprompted installation, follow the instructions in “Performing a Nonprompted BOS Installation” on page 200 to complete the prerequisite tasks.

From the Command Line

1. If the **mksysb** resource has already been created, skip to step 2. To create the **mksysb** resource, enter:

   ```
   nim -o define -t mksysb -a server=ServerName \
   -a location=LocationName -a mk_image=yes \
   -a source=SourceMachine ResourceName
   ```

   Specify the server name and location of the **mksysb** image. The **mk_image** and **source** attributes are used to create the **mksysb** image if it does not already exist.

   For a complete description of all the options that can be specified when creating a **mksysb** resource, see “mksysb Resource” on page 274.

   Example 1:

   To define a **mksysb** resource, **mksysb_res1**, from an existing **mksysb** image located in **/export/backups/client_mksysb** on the master, enter:

   ```
   nim -o define -t mksysb -a server=master \
   -a location=/export/backups/client_mksysb mksysb_res1
   ```

   Example 2:

   To create a **mksysb** image of the client machine, **client1**, in **/export/resources/new_mksysb** on the master, and to define a **mksysb** resource, **mksysb_res2**, enter:

   ```
   nim -o define -t mksysb -a server=master \
   -a location=export/resources/new_mksysb -a mk_image=yes \
   -a source=client1 mksysb_res2
   ```

2. To initiate the **bos_inst** operation, enter:

   ```
   nim -o bos_inst -a source=mksysb -a mksysb=mksysb \
   -a spot=SPOTName -a boot_client=yes/no ClientName
   ```

   Specify the resources to be used to support the installation and any additional options for customizing the installation. To perform a simple **mksysb** installation, specify the **mksysb** and **SPOT** resources.

   If the client machine being installed is not already a running, configured NIM client, NIM will not automatically reboot the machine over the network for installation. A network boot must be performed

manually on the machine. If that is the case, supply the **boot_client=no** attribute to the **bos_inst** command. If the **boot_client** attribute value is not specified, it defaults to **boot_client=yes**.

3. If the client was not rebooted automatically, initiate a network boot from the client to install it. For information about performing a network boot, see “Booting a Machine Over the Network” on page 343.
4. After the machine boots over the network, the display on the client machine will begin prompting for information about how to configure the machine during installation. Specify the requested information to continue with the installation.

   Example 3:

   To perform a **mksysb** installation using the **mksysb**, `mksysb1`, an optional **lpp_source**, `lpp_source1`, and the **SPOT**, `spot1`, on client machine, `machine1`, which is not a running, configured NIM client, enter:

   ```
   nim -o bos_inst -a source=mksysb -a mksysb=mksysb1 \
   -a lpp_source=lpp_source1 -a spot=spot1 -a boot_client=no machine1
   ```

Notes:

1. The steps to perform a **mksysb** installation are almost identical to the steps to perform other types of BOS installations. The main differences are that **mksysb** must be specified in the **source** attribute of the **nim bos_inst** command, and a **mksysb** resource must be allocated for the operation.
2. To perform a nonprompted installation, follow the instructions in “Performing a Nonprompted BOS Installation” to complete the prerequisite tasks.

Performing a Nonprompted BOS Installation

This procedure provides information about how to create a **bosinst_data** resource to use for a nonprompted BOS installation. After you have created the **bosinst_data** resource, refer to the following procedures to perform the nonprompted installation:

- “Using Installation Images to Install the Base Operating System (BOS) on a NIM Client” on page 196
- “Using a mksysb Image to Install the Base Operating System (BOS) on a NIM Client (mksysb Installation)” on page 197

Prerequisites

1. The NIM master must be configured, and **lpp_source** and **SPOT** resources must be defined. See “Configuring the NIM Master and Creating Basic Installation Resources” on page 190.
2. The NIM client to be installed must already exist in the NIM environment. To add the client to the NIM environment, use the procedure “Adding a Standalone NIM Client to the NIM Environment” on page 192.
3. If any of the software to be installed during the BOS installation requires acceptance of a license agreement, determine whether to accept the license agreement during BOS installation or defer acceptance until after the client has booted. Note that license acceptance takes place at the client. For a sample **bosinst.data** file that specifies the syntax to control license acceptance, see “bosinst.data File Example” on page 78.

From Web-based System Manager

You can use the Install Base OS wizard to create a **bosinst_data** resource. The **bosinst_data** resource can be used for a **rte** installation or a **mksysb** installation.

If you want to create the **bosinst_data** resource, continue with the following steps:

1. In the Resources container, from the Resources menu, select **New —> Resource**. The Add New Resource wizard displays.
2. Follow the wizard instructions to create a **bosinst_data** resource. The wizard creates a basic **bosinst.data** file, which can be used ″as is″ or can be enhanced according to sample files. For a

sample **bosinst.data** file, see “Nonprompted Network Installation” on page 78. To do a nonprompted installation, the **bosinst_data** resource must be created first.

From SMIT

1. On the NIM master or any running NIM client, create a **bosinst.data** file that describes how a machine should be configured during a BOS installation. For a sample **bosinst.data** file, see “bosinst.data File Example” on page 78.
2. To define the **bosinst.data** file as a **bosinst_data** resource in the NIM environment, enter the **smit nim_mkres** fast path.
3. Select **bosinst_data** from the list of resource types displayed on your screen.
4. Supply the values for the required fields. Use the help information and the LIST option to help you specify the correct values for defining your **bosinst_data** resource.
5. After the **bosinst_data** resource has been defined, follow the procedures for performing an **rte** or **mksysb** installation on a standalone machine. Be sure to specify the **bosinst_data** resource to use during the installation.

From the Command Line

1. On the NIM master or any running NIM client, create a **bosinst.data** file that describes how a machine should be configured during a BOS installation.

   Note: To accept license agreements for software to be installed during the BOS installation, specify `-a accept_licenses=yes` on the `nim -o bos_inst` command.
2. To define the **bosinst.data** file as a **bosinst_data** resource, enter:

   ```
   # nim -o define -t bosinst_data -a server=ServerName \
   -a location=LocationName NameValue
   ```

   Using the **server** attribute, specify the name of the machine where the **bosinst.data** file is located.

   Using the **location** attribute, specify the full path name of the **bosinst.data** file that is to be used as a resource.
3. After the **bosinst_data** resource has been defined, follow the normal procedure for performing an **rte** or **mksysb** installation on standalone machines. Be sure to specify that the **bosinst_data** resource be used for the installation.

   For example, to perform a nonprompted **rte** installation of `machine1` using the `lpp_source1`, `spot1`, and `bosinst_data1` resources, enter:

   ```
   # nim -o bos_inst -a source=rte -a lpp_source=lpp_source1 \
   -a spot=spot1 -a accept_licenses=yes -a bosinst_data=bosinst_data1 \
    machine1
   ```

Installing to Clients on ATM Networks

Unlike other network adapters, ATM adapters cannot be used to boot a machine. Installing a machine over an ATM network requires special processing.

BOS Installation over Non-ATM Adapters

Normally when a machine performs a network boot over a specified adapter, the adapter is configured by IPL-ROM or firmware. Then a boot image is transferred from the boot server to the client using **tftp**. This boot image performs further configuration and mounts network installation resources before starting the BOS installation.

BOS Installation over ATM Adapters

Because an ATM adapter cannot be configured by IPL-ROM or firmware, a boot image cannot be obtained over the network to perform a BOS installation. The NIM **bos_inst** operation must copy a boot image to

the hard disk of the client before the machine is rebooted. Some Object Data Manager (ODM) information is also saved on the client machine so that when the machine is rebooted, the ATM adapter can be configured properly.

NIM clients may not have the programs installed to support the special processing required for installation over ATM, so the **/usr/lib/boot/bin** and **/usr/lpp/bos.sysmgt/nim/methods** directories are mounted at the client from the NIM master. These directories contain the programs that run during the setup performed by the NIM **bos_inst** operation.

After the initial setup completes, an **at** job is issued to reboot the machine after one minute has elapsed. When the machine reboots, the boot image that was copied to the hard disk configures the ATM adapter and mounts network installation resources for the BOS installation. The installation then proceeds as normal until the customization phase. During NIM customization, the ATM adapter is not reconfigured with a **mktcpip** command because the ODM already contains information carried over from before the machine was reinstalled. All other aspects of NIM customization are the same as for non-ATM clients.

NIM Configuration Requirements for ATM Networks

- Machines that will have BOS installed over ATM must be running and configured NIM clients.

  Note: Configured NIM clients have the **bos.sysmgt.nim.client** fileset installed, are registered in the NIM master database, and have a valid **/etc/niminfo** file.
- BOS installations over ATM adapters will always use the **at0** interface on the client.
- The NIM master fileset must be installed at AIX 4.3 or later with the update for ATM installation or any superseding level.
- The SPOT that will be used to install the clients must be at version AIX 4.3 or later with the update for ATM installation or any superseding level.

Converting Generic Networks Into ATM Networks

Prior to the support of BOS installations over ATM, it was necessary to define ATM networks as ″generic″ networks for performing other types of NIM operations. To convert generic networks into ATM networks, enter the following command:

```
nim -o change -a new_type=atm (network)
```

The adapter names for the client interfaces on the ATM network will automatically be set to **at0** in the NIM database.

To change the name of the network, type the following:

```
nim -o change -a new_name=new_network_name current_network_name
```

System Recovery After Boot Failure

Because BOS installation over ATM requires a special boot image to be written to the hard disk of the client, the original boot image on the machine will be lost. If the installation is stopped or fails before BOS is reinstalled, it will not be possible to perform a normal reboot of the client unless system maintenance is performed. By performing system maintenance, a new boot image can be created on the hard disk to allow the machine to be booted for normal use. Use the following procedure:

1. Boot the client from a CD.
2. When the installation options are displayed, select the option to perform system maintenance.
3. Make the necessary selections to access the machine's root volume group.
4. In the maintenance shell, run the following sequence of commands:
   a. `bosboot -ad /dev/ipldevice`
   b. `BLVDISK='lslv -l hd5 | grep hdisk | head -1 |cut -d' ' -f1'`

c. `bootlist -m normal $BLVDISK`

d. `sync`

e. `sync`

f. `sync`

g. `reboot -q`

If errors are detected during the NIM **bos_inst** operation and the client machine has not rebooted, it is possible to stop the machine from rebooting, and then execute the sequence of commands in the above step 4 on the running system. To stop the reboot, use the following procedure:

1. List the **at** jobs on the machine by entering the command: `at -l`

   The first name in the output field will be the name of the job. For example:

   ```
   $ at -l
   root.884205595.a Wed Jan   7 14:39:55 1998
   ```

2. To remove the **at** job, enter the following command: `at -r` *name of job*

   For example:

   ```
   $ at -r root.884205595.a
   at file:  root.884205595.a deleted
   ```

Note: The reboot can also be prevented by removing the shutdown script that the **at** job was instructed to run by typing:

`rm/tmp/_NIM_shutdown`

Customizing NIM Clients and SPOT Resources

This procedure describes how to use NIM to install software on running, configured NIM clients and **SPOT** resources.

Prerequisites

1. If the software is to be installed on a machine, the machine must be a running, configured NIM client with push permissions enabled for the NIM master. Push permissions are enabled by default when a client is configured or installed by NIM.
2. If the software is to be installed on a **SPOT** resource, the server of the **SPOT** must be running.
3. The installation image to be installed on the target is available in an **lpp_source** resource, and a **check** operation was performed on the **lpp_source** at some point after the image was first copied there. (The **check** operation updates the **.toc** file with information about the images present in the **lpp_source**.)

From Web-based System Manager

1. From the NIM container, select the Machines container.
2. In the contents area, select a target machine (master or standalone), or in the Resources container, select a target **SPOT**.
3. From the Selected menu, choose **Install/Update Software** —> **Install Additional Software (Custom)** to display the Install Software dialog.
4. Use the dialog to complete the task.

From SMIT

The SMIT screens follow the same structure as those used for local installation operations performed on a system. When performing NIM customization operations, select the SMIT screen that most closely describes the installation you want to perform.

1. From the command line, enter the **smit nim_task_inst** fast path.
2. Select the SMIT menu item that matches the type of installation you want to perform.
3. Select a TARGET for the operation.
4. Select the **lpp_source** that contains the installation images to be used.
5. Select any other required resources.
6. In the final SMIT dialog, supply the values for the required fields or accept the defaults. Use the help information and the LIST option to help you specify the correct values.

From the Command Line

To perform the installation operation, enter:

```
nim -o cust -a lpp_source=Lpp_Source -a filesets=FilesetsList \
-a installp_bundle=InstallpBundle \
-a installp_flags=InstallpFlags TargetName
```

You will specify the resources to use to support the installation and any additional attributes for customization.

The software to be installed on the client can be specified on the command line using either the **filesets** attribute or by specifying an **installp_bundle** resource that lists the software.

The default **installp** flags to be used to install the software are **-a**, **-g**, **-Q**, and **-X**. To specify a different set of **installp** flags, you can list them in the **installp_flags** attribute.

Example 1:

To install the **bos.diag** and **bos.dosutil** filesets on the client, `machine1`, using the **lpp_source** resource named `lpp_source1`, enter:

```
nim -o cust -a lpp_source=lpp_source1 \
-a filesets="bos.diag bos.dosutil" machine1
```

Example 2:

To install software into the **SPOT** resource, `spot1`, using the **lpp_source** resource, `lpp_source1`, and the list of filesets specified in the **installp_bundle** resource, `installp_bundle1`, enter:

```
nim -o cust -a lpp_source=lpp_source1 \
-a installp_bundle=installp_bundle1 spot1
```

Note: Several other resources and attributes can be specified on the command line with the **cust** operation. For a complete description of the **cust** operation, see “NIM Operations” on page 282.

Configuring the NIM Master and Creating Resources to Support Diskless and Dataless Clients Only

Use this procedure only if the NIM environment is to be used exclusively for diskless and dataless client management. If the NIM environment is also to be used for installing and maintaining software on standalone machines, follow the procedure for “Configuring the NIM Master and Creating Basic Installation Resources” on page 190.

Note: This procedure produces a large amount of output, especially when creating the **SPOT** resource. Be sure to scan through the output to look for nonfatal errors and warnings that may not be evident from a successful return code.

Prerequisites

The NIM master must have at least 300 MB of available disk space. If such space is not available, see “Using Client Machines as Resource Servers” on page 217, and “Defining an lpp_source on CD-ROM versus Disk” on page 235.

From Web-based System Manager

1. Insert the AIX media into the appropriate drive of the designated master machine.
2. Start the Web-based System Manager application by typing `wsm`.
3. In the navigation area, select and expand the Software container.
4. While still in the navigation area, select the Installed Software container.
5. From the Software menu, choose **New Software (Install/Update)** —> **Install Additional Software**.
6. In the Install Software dialog, select `/dev/cd0` as the software source.
7. Specify `bos.sysmgt.nim` as the software to install.
8. In the navigation area, select the NIM container.
9. From the NIM menu, select **Configure Environment**.
10. Follow the wizard instructions to guide you through the configuration.

From SMIT

1. Insert the AIX media into the CD-ROM or tape drive of the designated master machine.
2. To install the **bos.sysmgt.nim** fileset, enter the **smit install_latest** fast path.
3. Using the LIST option, select **/dev/cd0** or **/dev/rmt0** for the INPUT device / directory for software.
4. Specify **bos.sysmgt.nim** as the SOFTWARE to install.
5. Accept the default values for all other fields on this screen. After completion of this installation, exit SMIT.
6. To configure the NIM master, enter the **smit nimconfig** fast path.
7. Specify a name in the Network Name field to be assigned to the NIM master’s network.
8. Using the LIST option, select the Primary Network Interface for the NIM Master.
9. Accept the default values for all other fields on this screen.
10. After the master is configured, exit SMIT.
11. Restart SMIT using the **smit nim_mkres_dd_name_server** fast path.
12. When prompted, select the NIM master as the server of the client resources.
13. Select **yes** in the `Create a new SPOT?` field, because there is not a **SPOT** currently defined in your environment.
14. Using the LIST option, select **/dev/cd0** or **/dev/rmt0** as the input device for installation images.
15. Specify a name in the SPOT Name field.
16. Specify names for the other resources to be created in the NIM environment. If a name is not specified, the resource will not be created.
17. Select **yes** at the `Remove all newly added NIM definitions and file systems if any part of this operation fails?` field. This will make it easier to restart this procedure if failures occur.
18. Accept the default values for all other fields on this screen.

    Note: In most NIM environments, the **SPOT** will already exist to support base operating system installation operations on standalone machines. In such environments, it is not necessary to create a new **SPOT**.

From the Command Line

1. Insert the AIX media into the CD-ROM or tape drive of the designated master machine.
2. If installing from a tape, skip to step 5. To create a mount point for the CD, type:

   ```
   mkdir /cdfs
   ```
3. To create a **cdrom** file system, type:

   ```
   crfs -v cdrfs -p ro -d'cd0' -m'/cdfs'
   ```
4. To mount the CD, type:

   ```
   mount /cdfs
   ```
5. To install the **bos.sysmgt.nim** fileset from the CD, type:

   ```
   installp -agX -d /cdfs/usr/sys/inst.images bos.sysmgt.nim
   ```

   or to install the **bos.sysmgt.nim** fileset from a tape, type:

   ```
   installp -agX -d /dev/rmt0 bos.sysmgt.nim
   ```
6. If installing from CD, to unmount the **cdrom** file system, type:

   ```
   unmount /cdfs
   ```
7. To configure the NIM master using the **nimconfig** command, type:

   ```
   nimconfig -a attr1=value1 \
             -a attr2=value2 \
             ...
   ```

   For example, to configure a NIM master with the following configuration:

   ```
   master host name = master1
   primary network interface = tr0
   ring speed = 16
   platform = chrp
   kernel type = mp
   ```

   enter the following command sequence:

   ```
   nimconfig -a netname=network1 -a pif_name=tr0 -a ring_speed=16 \
   -a platform=chrp -a netboot_kernel=mp
   ```

   Note: For additional attribute information, see the **nimconfig** command.
8. To create a file system in the `rootvg` volume group with 200 MB of space and a mount point of `/export/spot`, enter:

   ```
   crfs -v jfs -g rootvg -a size=$((2000*200)) \
   -m /export/spot -A yes -p rw -t no \
   -a frag=4096 -a nbpi=4096 -a compress=no
   ```
9. To mount the file system, enter:

   ```
   mount /export/spot
   ```
10. The **SPOT** resource will be installed from images in the image source (in this example, the CD). The server of the resource will be the NIM master, and the **SPOT** will be stored in the `/export/spot/spot1` directory. To create the **SPOT** resource, enter:

    ```
    nim -o define -t spot -a source=/dev/cd0 -a server=master \
    -a location=/export/spot spot1
    ```
11. To create a file system in the `rootvg` volume group with 150 MB of space and a mount point of `/export/dd_resource`, enter:

    ```
    crfs -v jfs -g rootvg -a size=$((2000*150)) \
    -m /export/dd_resource -A yes -p rw -t no \
    -a frag=4096 -a nbpi=4096 -a compress=no
    ```
12. To mount the file system, enter:

    ```
    mount /export/dd_resource
    ```

13. Create the diskless and dataless client resources in subdirectories of the **/export/dd_resource** directory. Not all resources are required. Create only the resources to be used in your environment.

    To create the `root` resource named `root1` (required for diskless and dataless clients), enter:

    ```
    nim -o define -t root -a server=master \
    -a location=/export/dd_resource/root1 root1
    ```

    To create the `dump` resource named `dump1` (required for diskless and dataless clients), enter:

    ```
    nim -o define -t dump -a server=master \
    -a location=/export/dd_resource/dump1 dump1
    ```

    To create the `paging` resource named `paging1` (required for diskless clients), enter:

    ```
    nim -o define -t paging -a server=master \
    -a location=/export/dd_resource/paging1 paging1
    ```

    To create the `home` resource named `home1` (optional), enter:

    ```
    nim -o define -t home -a server=master \
    -a location=/export/dd_resource/home1 home1
    ```

    To create the `shared_home` resource named `shared_home1` (optional), enter:

    ```
    nim -o define -t shared_home -a server=master \
    -a location=/export/dd_resource/shared_home1 shared_home1
    ```

    To create the `tmp` resource named `tmp1` (optional), enter:

    ```
    nim -o define -t tmp -a server=master \
    -a location=/export/dd_resource/tmp1 tmp1
    ```

Notes:

1. The file systems created for the NIM resources are not required, but they can be beneficial for storage management.
2. For more information about NIM resources, see “NIM Resources” on page 267.

Adding a Diskless or Dataless Client to the NIM Environment

This procedure describes how to add diskless and dataless clients to the NIM environment by adding an entry for the client to the NIM database on the master. This provides NIM with the information required to satisfy boot requests from the client. However, resources for the diskless or dataless client machine must be initialized before the client will be able to successfully boot and configure. See “Initializing and Booting a Diskless or Dataless Machine” on page 209 for more information. Diskless clients must mount all file systems from remote servers. Dataless clients can have paging space, as well as the **/tmp** and **/home** file systems on the local disk. Neither diskless nor dataless clients have a boot image on the local disk. Therefore, they must boot over the network.

Prerequisites

1. The NIM master must be configured, and the resources for diskless or dataless clients must be defined. For more information, see “Configuring the NIM Master and Creating Resources to Support Diskless and Dataless Clients Only” on page 204.
2. You must know the subnet mask, the default gateway for the client machine, and the default gateway for the NIM master.

From Web-based System Manager

1. In the Machines container, from the Machines menu, select **New —> OK**. The Add New Machine wizard displays.
2. Follow the wizard instructions to add a diskless or dataless client to the NIM environment.

From SMIT

1. To define a diskless or dataless client, enter the **smit nim_mkmac** fast path.
2. Specify the host name of the machine.
3. The SMIT screen displayed next depends on whether NIM already has information about the client's network. Supply the values for the required fields or accept the defaults. Use the help information and the LIST option to help you specify the correct values to define the client machine.

From the Command Line

To define a diskless or dataless client, enter:

```
nim -o define -t Diskless/Dataless \
-a platform=PlatformType -a netboot_kernel=NetbootKernelType \
-a if1=InterfaceDescription -a net_definition=DefinitionName \
-a ring_speed1=Speedvalue -a cable_type1=TypeValue \
-a iplrom_emu=DeviceName MachineName
```

Note: For detailed attribute information, see the descriptions of diskless and dataless clients in "NIM Machines" on page 257.

Example 1:

To add the diskless client with the host name `diskless1` to the NIM environment with the following configuration:

```
host name=diskless1
platform=rspc
kernel=up
network type=ethernet
subnet mask=255.255.240.0
default gateway=gw1
default gateway used by NIM master=gw_master
cable type=bnc
network boot capability=yes (no emulation needed)
```

enter the following command sequence:

```
nim -o define -t diskless -a platform="rspc" \
-a netboot_kernel="up" -a if1="find_net diskless1 0" \
-a cable_type1="bnc" \
-a net_definition="ent 255.255.240.0 gw1 gw_master" \
diskless1
```

Example 2:

To add the dataless client with the host name `dataless1` to the NIM environment with the following configuration:

```
host name=dataless1
platform=rs6k
netboot_kernel=up
network type=token ring
subnet mask=255.255.225.0
default gateway=gw2
default gateway used by NIM master=gw_master
ring speed=16
network boot capability=no (use emulation on a diskette)
```

enter the following command sequence:

```
nim -o define -t dataless -a platform="rs6k" \
-a netboot_kernel="up"  -a if1="find_net dataless1 0" \
-a ring_speed1="16" \
-a net_definition="tok 255.255.225.0 gw2 gw_master" \
-a iplrom_emu="/dev/fd0" dataless1
```

Note: If the **find_net** keyword in the **if** attribute causes NIM to successfully match a network definition to the client definition, the **net_definition** attribute is ignored.

Initializing and Booting a Diskless or Dataless Machine

This procedure describes how to use NIM to configure and boot a machine as a diskless or dataless client in the NIM environment.

Prerequisites

1. The NIM master must be configured, and the resources for diskless and dataless clients must be defined. See “Configuring the NIM Master and Creating Resources to Support Diskless and Dataless Clients Only” on page 204.
2. The NIM client must already exist in the NIM environment. To add the client to the NIM environment, use the “Adding a Diskless or Dataless Client to the NIM Environment” on page 207 procedure.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select the diskless or dataless machine you want to initialize.
3. From the Selected menu, choose **Initialize Machine Resources**.
4. Use the dialog to specify or select the resources to use for initialization. You will specify either the Home resource or Shared Home resource for the machine, but not both.
5. After completion of the initialization operation, use the “Booting a Machine Over the Network” on page 343 procedure to boot the client machine over the network.

   Note: On older model **rspc** systems, it may be necessary to permanently set the bootlist from the firmware menus to make the client always boot over the network. For other systems, the bootlist is automatically set the first time the machine is booted as a diskless/dataless client.
6. After the client boots over the network and performs some initialization, the client will display instructions for you to select the console for the machine.

From SMIT

1. On the NIM master, enter the **smit nim_dd_init** fast path.
2. Select the client to be initialized from the list of clients displayed on your screen.
3. Supply the values for the required fields. Use the help information and the LIST option to help you specify the correct values for the initialization options.
4. After completion of the initialization operation, use the “Booting a Machine Over the Network” on page 343 procedure to boot the client machine over the network.

   Note: On older model **rspc** systems, it may be necessary to permanently set the bootlist from the firmware menus to make the client always boot over the network. For other systems, the bootlist is automatically set the first time the machine is booted as a diskless/dataless client.
5. After the client boots over the network and performs some initialization, the client will display instructions for you to select the console for the machine.

From the Command Line

1. To initialize the client resources for diskless clients, enter the following on the NIM master:

   ```
   nim -o dkls_init -a spot=SPOTName -a root=RootName \
   -a dump=DumpName -a paging=PagingName ClientName
   ```
2. To initialize the client resources for dataless clients, enter the following on the NIM master:

```
nim -o dtls_init -a spot=SPOTName -a root=RootName \
-a dump=DumpName ClientName
```

Note: For detailed information about other attributes you can specify for the **dkls_init** and **dtls_init** operations, see “dkls_init” on page 293 and “dtls_init” on page 293.

3. After completion of the initialization operation, use the “Booting a Machine Over the Network” on page 343 procedure to boot the client machine over the network.

   Note: On older model **rspc** systems, it may be necessary to permanently set the bootlist from the firmware menus to make the client always boot over the network. For other systems, the bootlist is automatically set the first time the machine is booted as a diskless/dataless client.

4. After the client boots over the network and performs some initialization, the client will display instructions for you to select the console for the machine.

Uninitializing Diskless and Dataless Machines

Diskless and dataless machines are uninitialized by performing the **reset** operation. This action also provides the option to deallocate all resources for the machine. Deallocating all resources from the diskless or dataless machine removes all root data for the machine. Without deallocating resources, the uninitialize operation deallocates just the network boot image.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select the diskless or dataless machine you want to initialize.
3. From the Selected menu, choose **Uninitialize Machine Resources**.
4. Use the dialog to uninitialize and, if desired, deallocate all resources from the client.

From SMIT

1. To uninitialize diskless and dataless machines, enter the **smit nim_dd_uninit** fast path.
2. Select the Target.
3. If you want to remove all root data, change the DEALLOCATE Resources field to **yes**.

From the Command Line

1. To uninitialize the client machine, enter the following on the NIM master:

   ```
   nim -F -o reset ClientName
   ```

2. To deallocate all resources and remove root data, enter the following on the NIM master:

   ```
   nim -o deallocate -a subclass=all ClientName
   ```

Installing to an Alternate Disk on a NIM Client (cloning or mksysb)

NIM allows you to install an AIX 4.3 or later **mksysb** image (mksysb resource) on a NIM client’s alternate disk or to clone a NIM client’s current disk onto an alternate disk and apply updates. Because the client system is running during installation, less time is required than for a normal installation.

Note: For information about the different ways NIM can customize an alternate disk installation, see “alt_disk_install” on page 283.

Prerequisites

1. The NIM master must be configured. To install a **mksysb** image onto the alternate disk, the **mksysb** resource must be defined. See “Configuring the NIM Master and Creating Basic Installation Resources” on page 190.
2. The NIM client must already exist in the NIM environment and must be running. To add the client to the NIM environment, see “Adding a Standalone NIM Client to the NIM Environment” on page 192.
3. The **bos.alt_disk_install.rte** fileset must be installed on the NIM client. To install a new fileset on a NIM Client, see “Customizing NIM Clients and SPOT Resources” on page 203.

From Web-based System Manager

1. Select the Machines Container.
2. In the contents area, select the standalone machine for the alternate disk installation.
3. From the Selected menu, choose **Alternate Disk Installation —> Clone the Rootvg to an Alternate Disk** or **Install Mksysb on an Alternate Disk**.
4. Use the dialog to finish the installation.

From SMIT

1. Enter the **smit nim_alt_mksysb** fast path from the NIM master.
2. Select the Target Machine or Target Group to Install.
3. Enter the Target Disk or Disks on the Target machine.
4. Accept the default installation options, or supply different ones in the displayed dialog fields. Use the help information and the LIST option for guidance.
5. The alternate disk installation will be initiated on the client, and progress can be seen with the **lsnim** command (**smit lsnim**). If the ″Reboot when complete?″ option is set to **yes** and the ″Phase to execute″ is **all** or includes Phase 3, the client will reboot from the newly installed disk when the **alt_disk_install** command is complete.
6. To clone a disk onto a NIM client’s alternate disk, enter the **smit nim_alt_clone** fast path from the NIM master.

From the Command Line

The **alt_disk_install** command is initiated on the target system, and progress is shown with the **lsnim** command. In addition, a log kept on the target system, **/var/adm/ras/alt_disk_inst.log**, contains progress messages and any error or warning messages that might occur. The **/var/adm/ras/nim.alt_disk_install** log will contain debug information, if requested.

Installing mksysb on an Alternate Disk

Initiate the **alt_disk_install** operation by entering:

```
nim -o alt_disk_install -a source=mksysb -a mksysb=Mksysb \
-a disk='diskname(s)' ClientName
```

Specify the **mksysb** resource to be used and any additional options for customizing the installation. To perform a simple alternate disk **mksysb** install, specify the **source**, **mksysb**, and **disk** resources.

Note: For detailed information about the mksysb resources, see “mksysb Resource” on page 274.

Cloning the rootvg to an Alternate Disk

To clone a disk onto a NIM client’s alternate disk, enter:

```
nim -o alt_disk_install -a source=rootvg -a disk=diskname(s) ClientName
```

Specify any additional options for customizing the installation.

Examples

The client machine `machine1` is a running system with a disk, `hdisk2`, that is not currently occupied by a volume group.

- To install this disk with a **mksysb** resource named `51mksysb` enter:

```
nim -o alt_disk_install -a source=mksysb -a mksysb=51mksysb \
-a disk=hdisk2 machine1
```

- To clone the rootvg to `hdisk2` enter:

```
nim -o alt_disk_install -a source=rootvg -a disk=hdisk2 machine1
```

Network Installation Management Commands Reference

This section provides information on NIM commands and where to find more information on the NIM commands.

The nim_master_setup Command

The **nim_master_setup** command installs the **bos.sysmgt.nim.master** fileset, configures the NIM master, and creates the required resources for installation, including a **mksysb** system backup.

The **nim_master_setup** command uses the **rootvg** volume group and creates an **/export/nim** file system, by default. You can change these defaults using the **volume_group** and **file_system** options. The **nim_master_setup** command also allows you to optionally not create a system backup, if you plan to use a **mksysb** image from another system. The **nim_master_setup** usage is as follows:

```
Usage nim_master_setup: Setup and configure NIM master.
       nim_master_setup [-a mk_resource={yes|no}]
 [-a file_system=fs_name]
 [-a volume_group=vg_name]
 [-a disk=disk_name]
 [-a device=device]
 [-B] [-v]

-B    Do not create mksysb resource.
-v    Enable debug output.

Default values:
mk_resource = yes
file_system = /export/nim
volume_group = rootvg
device = /dev/cd0
```

The nim_clients_setup Command

The **nim_clients_setup** command is used to define your NIM clients, allocate the installation resources, and initiate a NIM BOS installation on the clients. The **nim_clients_setup** command uses the definitions in the **basic_res_grp** resource to allocate the necessary NIM resources to perform a **mksysb** restore operation on the selected clients. The usage for **nim_clients_setup** is as follows:

```
Usage nim_clients_setup: Setup and Initialize BOS install for NIM clients.
       nim_clients_setup [-m mksysb_resource]
 [-c] [-r] [-v] client_objects
-m    specify mksysb resource object name -OR- absolute file path.
-c    define client objects from client.defs file.
-r    reboot client objects for BOS install.
-v    Enables debug output.
```

Note: If no client object names are given, all clients in the NIM environment are enabled for BOS installation; unless clients are defined using the **-c** option.

Other NIM Commands Reference

The *AIX 5L Version 5.2 Commands Reference* provides reference information about the NIM commands, AIX operating system commands, and commands for other licensed programs for end users, system administrators, and programmers. This set of books contains examples and descriptions of the commands and their available flags. The command entries are arranged in alphabetic order:

- *AIX 5L Version 5.2 Commands Reference, Volume 1* contains commands ac through cx
- *AIX 5L Version 5.2 Commands Reference, Volume 2* contains commands da through hy
- *AIX 5L Version 5.2 Commands Reference, Volume 3* contains commands ib through mw
- *AIX 5L Version 5.2 Commands Reference, Volume 4* contains commands na through rw
- *AIX 5L Version 5.2 Commands Reference, Volume 5* contains commands sa through uu
- *AIX 5L Version 5.2 Commands Reference, Volume 6* contains commands va through yp

For example, *AIX 5L Version 5.2 Commands Reference, Volume 3* contains reference information for the NIM **lsnim** command. The *AIX 5L Version 5.2 Commands Reference, Volume 4* contains reference information for the NIM **nim**, **nimclient**, **nimconfig**, **nimdef**, and **niminit** commands.

You can access all of the documentation through the IBM eServer pSeries Information Center on the Internet at the following Web address:

`http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base`

Chapter 21. Advanced NIM Installation Tasks

This chapter describes the following procedures for performing advanced NIM installation tasks using the Web-based System Manager application or the System Management Interface Tool (SMIT), as well as from the command line:

- "Controlling the Master or Client"
- "Resetting Machines" on page 216
- "Using Client Machines as Resource Servers" on page 217
- "Defining a Machine Group" on page 218
- "Adding New Members to Machine Groups" on page 219
- "Removing Members from Machine Groups" on page 219
- "Including and Excluding a Group Member from Operations on the Group" on page 220
- "Using Resource Groups to Allocate Related Resources to NIM Machines" on page 221
- "Managing Software on Standalone Clients and SPOT Resources" on page 222
- "Rebuilding Network Boot Images for a SPOT" on page 225
- "Maintaining Software in an lpp_source" on page 225
- "Viewing Installation, Configuration, and Boot Logs" on page 227
- "Verifying Installation with the lppchk Operation" on page 227
- "Using NIM to Install Clients Configured with Kerberos Authentication" on page 228
- "Concurrency Control" on page 229
- "Alternate Disk Migration Installation" on page 149

Controlling the Master or Client

In the NIM environment, control is held by the NIM master or the standalone client. The system allocating the resources has control. The allocation of resources is the act of making resources available to clients for NIM operations. Normally, resources are allocated automatically as part of an operation, but they may also be allocated prior to the initiation of an operation. The control status acts like a locking mechanism and remains with the client or the master until the resources are deallocated. Using NIM, if the installation of a standalone client completes successfully, the resources are automatically deallocated.

When there are no resources allocated to the standalone client by the NIM master, the standalone client takes control by allocating resources or disabling the NIM master's push permissions. The **control** attribute is managed by the master and indicates whether the master or the standalone client has permission to perform operations on the standalone client.

The **control** attribute indicates four control states. You can display the **control** attribute from a NIM client by entering:

```
nimclient -l -l StandAloneClientName
```

The **control** attribute can be displayed from the NIM master by entering:

```
lsnim -l StandaloneClientName
```

The control states are as follows:

control attribute is not set	If the **control** attribute is not displayed when listing the machine object attributes, then neither the master nor the standalone client has control.
control = master	The master has allocated resources to the client and is ready to initiate an operation (or has already initiated an operation).

© Copyright IBM Corp. 2002, 2004

control = *StandaloneClientName*	The standalone client has allocated resources and can now initiate NIM operations on itself.
control = *StandaloneClientName* **push_off**	The standalone client has prohibited the NIM master from allocating resources or initiating operations on the client. The client itself can still control the allocation of NIM resources and the initiation of NIM operations.

Disabling Master Push Permissions

The NIM master must have push permissions to perform push operations on the NIM clients. You can disable the NIM master's push permissions using Web-based System Manager, SMIT, or command line as follows:

From Web-based System Manager

1. From the main Web-based System Manager container, select the Software icon.
2. From the Software menu, select **NIM Client** —> **Permissions**.
3. Select whether to grant or deny permission for the NIM master to initiate push installations.

From SMIT

To disable the master's push permissions, enter the **smit nim_perms** fast path from the client machine.

From the Command Line

To set **control** on the client to **push_off**, enter the following on the client machine:

```
nimclient -P
```

To re-enable push permission on the client, enter the following on the client machine:

```
nimclient -p
```

Resetting Machines

The operations performed using NIM can be very complex. To help ensure that the operations can be completed successfully, NIM requires that a machine be in the **ready** state before operations can be run on it. While an operation is being performed, the state of the machine will reflect the current operation. After the operation completes, the machine returns to the **ready** state.

If an operation on a machine is interrupted, the machine state may continue to reflect the operation. If this occurs, the machine must be reset to the **ready** state before performing any further operations. To return a machine to the **ready** state, use the NIM **reset** operation.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a target standalone, diskless, or dataless machine to reset.
3. From the Selected menu, choose **Administration** —> **Reset NIM State**.
4. Use the dialog to reset the state of the machine.

You can also do this task from Troubleshooting. From the Selected menu, choose **Troubleshooting** —> **Clean Up Failed or Interrupted Installation**.

From SMIT

1. To return a machine to the **ready** state, enter the **smit nim_mac_op** fast path.
2. Select the target machine for the operation.
3. Select **reset** as the Operation to Perform.

4. To deallocate resources, change the `Deallocate All Resources?` field to **yes**.
5. Change the Force field to **yes**.

From the Command Line

1. To return a machine to the **ready** state, enter:

   ```
   nim -Fo reset MachineName
   ```

2. To deallocate resources, enter:

   ```
   nim -o deallocate -a ResourceType=ResourceName MachineName
   ```

   where *ResourceType* is the type of the resource being deallocated (for example, **lpp_source**, **SPOT**, **Script**, etc.), *ResourceName* is the name of the resource being deallocated, and *MachineName* is the name of the machine that has been allocated the resources.

   Note: Resetting a machine will not automatically deallocate all the resources that were allocated for the operation. To deallocate resources, use the NIM **deallocate** operation.

Using Client Machines as Resource Servers

Any machine in the NIM environment can be a resource server. In simple environments, the NIM master is usually used to serve all the NIM resources.

Defining resources on client machines can be beneficial for the following reasons:

- Disk space limitations on the NIM master may prohibit the storage of all the resources on a single machine.
- Resource usage may be heavy, and communications and data access bottlenecks could occur if all the resources were served by a single machine.

  For example, if you use NIM to install 200 machines on 5 different subnets, you could have a set of resources created and available on each subnet. Each set of resources would be used to install the machines on the same subnet. In addition to distributing the workload among several resource servers, this would also reduce the network traffic across the gateways between the different subnets.

From Web-based System Manager

1. Select the Resources container.
2. From the Resources menu, select **New Resources**.
3. Follow the wizard instructions to create the resource.

From SMIT

1. To create a resource on a NIM client, enter the **smit nim_mkres** fast path.
2. Select the Resource Type.
3. In the displayed dialog fields, supply the correct values for the resource options. Be sure to specify the name of the client machine for the Server of the Resource field. Use the help information or the LIST option to help you. All attributes specified when the resource is defined (such as **location** and **source**) must be local to the server machine.

From the Command Line

To create a resource on a NIM client, specify the client's NIM name for the **server** attribute when defining the resource.

Example:

To create an **lpp_source** resource named `images2` from a CD on the NIM client machine, `client_mac1`, in the `/resources/images` directory, enter:

```
nim -o define -t lpp_source -a server=client_mac1 \
-a location=/resources/images -a source=/dev/cd0 images2
```

Defining a Machine Group

Machine groups can be defined to collect multiple clients in a common target for NIM operations. Groups can be defined for standalone, diskless, or dataless clients; but a group can only contain clients of a single type.

Web-based System Manager supports the following types of machine groups:

- A temporary machine group is created when multiple machines are selected in the NIM container and an action from the Selected menu is performed. The temporary group is removed when the action is completed.
- If you want a more permanent machine group, you can create it using the **New Machine Group** menu option in the NIM menu.

In the Web-based System Manager NIM application, machine groups are not explicitly created and managed, but ad hoc groupings are supported by multi-selecting the icons representing machines in the NIM container. After they are selected, a group of machines can be administered by selecting an action from the Selected menu.

Note: You can perform most operations only on multi-selected machines of the same type.

From Web-based System Manager

1. Select the Groups container.
2. From the Groups menu, select **New —> Group**.
3. Select the machine type.
4. Select a machine from the list on the right, and click the **<** button to add the machine to the new group. Repeat this step until all the desired members of the machine group are in the **Members** list on the left.
5. Click **OK**.

From SMIT

1. To define a machine group, enter the **smit nim_mkgrp** fast path.
2. Select the type of group you want to define.
3. Enter the name of the group and member information.

From the Command Line

To define a machine group, enter:

```
nim -o define -t mac_group -a add_member=MemberName GroupName
```

For example, to create a machine group named `MacGrp1` containing previously defined machines `Standalone1`, `Standalone2`, and `Standalone3`, enter:

```
nim -o define -t mac_group -a add_member=Standalone1 \
-a add_member=Standalone2 -a add_member=Standalone3 \
-a comments="Machines for Department d03" MacGrp1
```

Adding New Members to Machine Groups

New members can be added to machine groups, however, the new member must be of the same machine type as existing members. Members can be added to machine groups using the Web-based System Manager NIM application.

From Web-based System Manager

1. Select the Groups container.
2. In the contents area, select a group.
3. From the Selected menu, choose **Add/Remove Members...**.
4. Select a machine from the list on the right, and click on the **<** button to add the machine to the new group. Continue with this step until all the desired members of the machine group are in the list on the left.
5. Click on **OK**.

From SMIT

1. To add members to a machine group, enter the **smit nim_chgrp** fast path.
2. Select the machine group to modify.
3. Specify members to add to the group. Use the LIST option to select members to add.

From the Command Line

To add a member to a machine group, enter:

```
nim -o change -a add_member=MachineName GroupName
```

For example, to add the diskless client, `diskless5`, to the machine group, `diskless_grp`, enter the following command:

```
nim -o change -a add_member=diskless5 diskless_grp
```

Alternatively, you could have specified group members in both the **define** and **change** operations by using sequenced member attributes, such as `-a member1=Standalone1 -a member2=Standalone2` and so forth.

Removing Members from Machine Groups

Members can be removed from machine groups. Whenever the last member of a machine group is removed, the group definition is also removed.

The Web-based System Manager NIM application can be used to remove members from machine groups.

From Web-based System Manager

1. Select the Groups container.
2. From the Selected menu, choose **Add/Remove Members**.
3. Select a machine from the list on the left and click on the **>** button to add the machine to the list on the right. Continue with this step until all the desired members of the machine group have been removed.
4. Click on **OK**.

From SMIT

1. To remove members from a machine group, enter the **smit nim_chgrp** fast path.
2. Select the machine group to modify.
3. Specify members to remove from the group. Use the LIST option to select members to remove.

From the Command Line

To remove a member from a machine group, enter the following command:

```
nim -o change -a rm_member=MachineName GroupName
```

For example, to remove machine, Standalone2, and add machine, Standalone4, to the group, MacGrp1, enter:

```
nim -o change -a rm_member=Standalone2 \
-a add_member=Standalone4 MacGrp1
```

Including and Excluding a Group Member from Operations on the Group

Group members may be included or excluded by using the Web-based System Manager NIM application, SMIT, or from the command line. Use the **select** operation from the command line to indicate that specific members of a machine group should be included or excluded from operations on that group. This capability is useful if an operation needs to be tried again on specific group members that failed during an initial operation on the group. When a group member is marked as being excluded, it remains so until it is included again.

From Web-based System Manager

1. Select the Groups container.
2. In the contents area, expand a group container to view the members included in that group.
3. Select a machine from those listed in the container.
4. From the Selected menu, choose **Properties**.

From SMIT

1. To include or exclude a group member from operations on the group, enter the **smit nim_grp_select** fast path.
2. Select the name of the group from which you want to include or exclude members.
3. Select the members to include or exclude.

From the Command Line

To include or exclude a group member, enter the following:

```
nim -o select -a include_all=Value -a exclude_all=Value \
-a include=MemberName -a exclude=MemberName GroupName
```

As an example, to exclude the machine, Standalone2, from further operations on machine group, MacGrp1 and to include a previously excluded machine, Standalone3, enter:

```
nim -o select -a exclude=Standalone2 -a include=Standalone3  MacGrp1
```

The special attributes **include_all** and **exclude_all**, when assigned a value of **yes**, can be used respectively to include or exclude all members in a group. The **select** operation evaluates command line attributes from left to right. The following example shows how to exclude all members except Standalone2 from subsequent operations on the MacGrp1 machine group:

```
nim -o select -a exclude_all=yes -a include=Standalone2 MacGrp1
```

Using the special **-g** option shows the excluded status of the group's members:

```
lsnim -g MacGrp1
```

Group member information similar to the following is displayed:

```
MacGrp1:
type   = mac_group
member1=Standalone1;ready for a NIM operation,not running;EXCLUDED
member2=Standalone2;ready for a NIM operation; currently running;
member3=Standalone3;ready for a NIM operation,not running;EXCLUDED
```

Using Resource Groups to Allocate Related Resources to NIM Machines

NIM resource groups allow association of resources so they can be allocated as a logical unit to machines prior to other NIM operations. Resource groups can only contain one of each resource type, except for **script** and **installp_bundle** resources, which may occur multiple times in a given resource group.

Defining a Resource Group

From SMIT

1. To define a resource group, enter the **smit nim_mkgrp_resource** fast path.
2. Enter the name of the group with member information.

From the Command Line

To define a resource group, enter:

```
nim -o define -t res_group -a ResourceType=ResourceName GroupName
```

As an example, to create a resource group named `ResGrp1` containing previously defined resources, `images1`, `spot1`, `bosinst_data1`, and `bundle1`, enter:

```
nim -o define -t res_group -a lpp_source=images1 -a spot=spot1 \
-a bosinst_data=bosinst_data1 -a installp_bundle=bundle1 \
-a comments="BOS Install Resources" ResGrp1
```

Allocating a Resource Group

From SMIT

1. To allocate a resource group, enter the **smit nim_alloc** fast path.
2. Select the machine or machine group from the list of defined machines (for example, `Standalone1`).
3. A list of resource groups is displayed. Select the resource group you want to allocate.

From the Command Line

To allocate a resource group, enter:

```
nim -o allocate -a group=ResGroupName TargetName
```

For example, to allocate a resource group named `ResGrp1` to a machine named `Standalone1`, enter:

```
nim -o allocate -a group=ResGrp1 Standalone1
```

Alternatively, the group resource can be specified on the command line to the operation. For example, to allocate the resource group, `ddResGrp`, while performing the **dkls_init** operation on a group of diskless machines named `DklsMacs`, enter:

```
nim -o dkls_init -a group=ddResGrp DklsMacs
```

Defining Default Resource Groups

After a resource group is defined, you may want to specify the group as the set of defaults for all operations that require resources. Set the **default_res** attribute on the master to the name of the resource group that you want to be the default.

From SMIT

1. To define default resource groups, enter the **smit nim_grp** fast path.
2. Choose Select/Unselect a Default Resource Group.
3. Fill in the name of the group that is to act as the default.

From the Command Line

To define default resource groups, enter:

```
nim -o change -a default_res=ResGroupName master
```

For example, if the `ResGrp1` resource group should be the set of default resources for all NIM operations, enter:

```
nim -o change -a default_res=ResGrp1 master
```

Note: All applicable resources are allocated from the group specified as the default for all operations, except for **installp_bundle** for a **maint** operation.

A resource from the default group will only be allocated if a resource of the same type is not already allocated and if a resource of that type is not specified on the command line for automatic allocation. The exceptions are the **script** and **installp_bundle** resources, of which all occurrences in the resource group and specified on the command line will be allocated.

Default members can be overridden by specifying a null value in the attribute assignment for that resource.

The following **bos_inst** operation allocates all applicable **bos_inst** resources from the resource group specified as the default, except for the **bosinst_data** resource:

```
nim -o bos_inst -a bosinst_data=Standalone1
```

Managing Software on Standalone Clients and SPOT Resources

The commands for managing software on standalone clients and **SPOT** resources are generally the same. Specify the name of the machine, group, or **SPOT** as the target of the option.

Note: If the **SPOT** is currently allocated to a NIM client, NIM prevents the change to the **SPOT**. Use the **Force** (**-F**) option to force the operation.

Software updates to a **SPOT** cause the **SPOT**'s network boot images to be rebuilt when necessary. If you think the boot images are bad, you can force them to be rebuilt using the NIM **check** operation.

Software updates to a **SPOT** may also cause software updates to occur in the root parts of diskless and dataless clients of the **SPOT**. This will occur automatically. You can force a synchronization of the client root parts using the NIM **sync_roots** operation on the **SPOT**.

For information on how to install additional software on standalone clients and SPOT resources, see "Customizing NIM Clients and SPOT Resources" on page 203.

Listing Software Installed on a Standalone Client or SPOT

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a target machine (master or standalone), or in the Resources container, select a target **SPOT** resource.
3. From the Selected menu, choose **List Installed Software** —> **All Installed**.

From SMIT

1. To list software installed on a standalone client or **SPOT**, enter the **smit nim_list_installed** fast path.
2. Select the menu item that describes the list operation you want to perform.
3. Select a target for the operation.
4. In the displayed dialog fields, supply the required values. Use the help information or the LIST option to help you.

From the Command Line

Enter the following command:

```
nim -o lslpp [-a lslpp_flags=LslppFlags] TargetName
```

where *LslppFlags* are the flags to be passed to the **lslpp** command, and *TargetName* is the name of the client or **SPOT** object.

For example:

```
nim -o lslpp -a lslpp_flags=La spot1
```

Listing Software Updates, Installed on a Standalone Client or SPOT, by Keyword

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a target machine (master or standalone), or in the Resources container, select a target **SPOT** resource.
3. From the Selected menu, choose **List Installed Software** —> **Fix (APAR) Status**.
4. Use the dialog to list the installation status of specific installed fixes.

From SMIT

1. To list fixes installed on a standalone client or **SPOT** by APAR number or keyword, enter the **smit nim_mac_op** fast path for standalone clients, or enter the **smit nim_res_op** fast path for **SPOTs**.
2. Select the standalone client or **SPOT** resource object.
3. Select the **fix_query** operation.
4. Select the desired **fix_query** flags or accept the default settings. Specify the **fix_bundle** object name; or to check the installation status of an APAR, specify the fix APAR numbers. If you leave both blank, all known fixes are displayed.

From the Command Line

Enter the following command:

```
nim -o fix_query [ -afixes="FixKeywords" ] \
[-afix_bundle=FixBundleName ] [ -afix_query_flags=FixQueryFlags ] \
TargetName
```

where *FixKeywords* are APAR numbers; *FixBundleName* is the object name of the **fix_bundle** resource; *FixQueryFlags* are optional flags to the **fix_query** operation, and *TargetName* is the client, group, or **SPOT** for which to display fix information.

Valid *FixQueryFlags* are as follows:

-a Displays symptom text.
-c Displays output in colon-separated format.
-F Returns failure unless all filesets associated with a fix are installed.
-q Quiet option; if **-q** is specified, no heading is displayed.
-v Verbose option; gives information about each fileset associated with a fix (keyword).

For example:

- To query the `fix` database on `standalone1` to determine if all fileset updates for fix `IX12345` are installed, enter:

  ```
  nim -o fix_query -afixes=IX12345 standalone1
  ```

- To list fix information for all known fixes installed on `spot1`, with symptom text, enter:

  ```
  nim -o fix_query -afix_query_flags=a spot1
  ```

Maintaining Software on Standalone Clients and SPOT Resources

NIM uses the **installp** command to construct a **SPOT** by installing in the **SPOT** the software products that each **SPOT** needs to support the NIM environment. Because the **installp** command also supports software maintenance tasks, you can perform these tasks on **SPOTs** as well. For example, you can remove previously installed optional software from a **SPOT** when they are no longer being used. This kind of task is accomplished by performing the NIM **maint** operation on a **SPOT** using the Web-based System Manager NIM application, SMIT, or command line interface. You interact with the **installp** command by supplying the **installp_flags,** and either **filesets** or **installp_bundle** attributes.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a target standalone machine, or in the Resources container, select a target **SPOT** resource.
3. From the Selected menu, choose **Software Utilities** —> **Commit Applied Updates**, **Reject Applied Updates**, or **Remove Software**, depending upon the task you want to perform.

From SMIT

1. To perform software maintenance, enter the **smit nim_task_maint** fast path.
2. Select the menu item that describes the maintenance that you want to perform.
3. Select the target for the operation.
4. In the displayed dialog fields, supply the required values. Use the help information or the LIST option to help you.

From the Command Line

Enter the following command:

```
nim -o maint -a installp_flags="InstallpFlags"  \
[-a filesets="FileSetNames" | \
-a installp_bundle=BundleResourceName ] [-F] TargetName
```

where *InstallpFlags* are the flags you want to pass to the **installp** command; *FileSetNames* are the names of the filesets or packages you want to maintain; *BundleResourceName* is the object name of the **installp_bundle** resource; and *TargetName* is the object name of the standalone client, group, or **SPOT**.

For example:

- To remove the `bos.adt` and `bos.INed` software packages from `standalone1`, enter:

  ```
  nim -o maint -a filesets="bos.adt bos.INed" -a \
  installp_flags="-u" standalone1
  ```

- To remove the `bos.INed` software package from `spot1`, which is allocated to diskless or dataless clients, without deallocating `spot1` first, enter:

  ```
  nim -o maint -F -a filesets=bos.INed -a installp_flags="-u" \
  spot1
  ```

- To remove the packages from `spot1` which are listed in the bundle pointed to by the **installp_bundle** resource object, `bundle1`, enter:

  ```
  nim -o maint -a installp_flags="-u" -a installp_bundle=bundle1 \
  spot1
  ```

- To clean up from an interrupted software installation on `spot1`, enter:
  ```
  nim -o maint -a installp_flags="-C" spot1
  ```

Rebuilding Network Boot Images for a SPOT

From Web-based System Manager

1. Select the Resources container.
2. In the contents area, select a target **SPOT**.
3. From the Selected menu, choose **Check SPOT**.
4. Use the dialog to select the Build debug network boot images and/or the force option, if needed.

You can also perform this task from Troubleshooting. From the Selected menu, choose **Troubleshooting —> Build Non-Debug Network Boot Images**.

From SMIT

1. To rebuild network boot images for a **SPOT**, enter the **smit nim_res_op** fast path.
2. Select the **SPOT**.
3. Select the **check** operation.
4. In the displayed dialog fields, set the Force option to **yes**.

From the Command Line

To force the rebuild of the boot images, enter:

```
nim -Fo check SPOTName
```

For information on how to install additional software on standalone clients and SPOT resources, see “Customizing NIM Clients and SPOT Resources” on page 203.

Maintaining Software in an lpp_source

To add or remove software in an **lpp_source**, add or remove the installation image from the **lpp_source** directory, and then initiate the NIM **check** operation on the **lpp_source**.

Copying Software to an lpp_source

From Web-based System Manager

1. Select the Resources container.
2. In the contents area, select an **lpp_source**.
3. From the Selected menu, choose **Properties**. The General page of the properties notebook displays.
4. From the General page, identify the location of the resource. Close the notebook.
5. From the Resources menu, select **Copy Software to Directory**, and specify as the destination directory, the location of the resource identified in the notebook.
6. After the copy is completed, select the **lpp_source** and from the Selected menu, choose **Check NIM State**. This action updates the table of contents (**.toc**) file for the **lpp_source**.

From SMIT

1. To copy software from installation media to an **lpp_source**, insert the installation media in the appropriate drive of the **lpp_source** server.
2. To copy the software to the **lpp_source** directory, enter **smit bffcreate** from the resource server.
3. Enter the INPUT device / directory for software.

4. In the displayed dialog fields, supply the correct values or accept the default values. Be sure to specify the **lpp_source** location for the directory to store the installation images. Use the help information and the LIST option to help you.

From the Command Line

1. Copy the software from the media to the **lpp_source** directory.
2. Perform the NIM check operation on the **lpp_source** by entering the following command:

   ```
   nim -o check Lpp_sourceName
   ```

Removing Software from an lpp_source

To remove software from an **lpp_source**, delete the installation image from the **lpp_source** directory.

Note: This function is only available from the command line interface.

From the Command Line

1. Remove the installation image from the **lpp_source** directory.
2. Perform the NIM check operation on the **lpp_source** by entering the following command:

   ```
   nim -o check Lpp_sourceName
   ```

Running the NIM check Operation

After adding or removing software, you must run the NIM **check** operation on the **lpp_source** to update the installation table-of-contents file for the resource.

In addition to updating the table-of-contents for the **lpp_source**, the **check** operation also updates the **simages** attribute for the **lpp_source**, which indicates whether the **lpp_source** contains the images necessary to install the Base Operating System images on a machine.

From Web-based System Manager

1. Select the Resources container.
2. In the contents area, select a target **lpp_source** resource.
3. From the Selected menu, choose **Check NIM State**.

From SMIT

1. To run the NIM **check** operation, enter the **smit nim_res_op** fast path.
2. Select the **lpp_source** for the operation.
3. Select **check** for the operation to be performed.

From the Command Line

To initiate the NIM **check** operation on the **lpp_source**, enter:

```
nim -o check Lpp_sourceName
```

If the **lpp_source** is currently allocated to a client, use the **Force** option as follows:

```
nim -F -o check Lpp_sourceName
```

Viewing Installation, Configuration, and Boot Logs

After installing a standalone machine, use the **showlog** operation to check the installation results by viewing the installation, boot, and configuration logs. One of several log types can be viewed by specifying one of the following as the value of the **log_type** attribute to the **showlog** operation:

devinst	Output from the installation of key system and device-driver software
niminst	Output from the installation of user-specified software (including installation of NIM client software during a **bos_inst** operation)
bosinst	Output from the BOS installation program
boot	The machine's boot log
lppchk	A log of the output from the **lppchk** operation executed on a standalone NIM client
script	Output from any configuration script resources allocated for a **bos_inst** operation
nimerr	Errors encountered during execution of the **nim** command.

By default, the **showlog** operation applied to a standalone machine displays the **niminst** log and shows the output logged when software was last installed on the machine using NIM. The last entry is also shown by default for the **script** and **lppchk** logs. The entire contents of the **niminst**, **script**, and **lppchk** logs can be displayed by assigning the **full_log** attribute a value of `yes` when executing the **showlog** operation. The entire log is shown for all other log types.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a target machine (master, standalone, diskless, or dataless), or in the Resources container, select a target **SPOT**.
3. From the Selected menu, choose **Troubleshooting** —> **Show NIM Logs**.
4. Use the dialog to select the log you want to examine.

From SMIT

1. Enter the **smit nim_mac_op** fast path to view a machine's log, or enter **smit nim_res_op** to view a **SPOT**'s log.
2. Select the object name of the machine or **SPOT** whose log you want to view.
3. Select **showlog** from the list of operations.
4. Select the log type to be viewed.
5. Specify if the full log should be viewed (only applicable to **script**, **lppchk**, and **niminst** logs).

From the Command Line

To view a log on a standalone machine or **SPOT**, enter:

```
nim -o showlog -a LogType=value ObjectName
```

where *LogType* represents the log you want to view, and *ObjectName* is the name of the machine or **SPOT** whose log will be viewed.

Verifying Installation with the lppchk Operation

When investigating functional problems in software, you can use the **lppchk** operation to check the integrity of installed software.

From Web-based System Manager

1. Select the Machines container.

2. In the contents area, select a target standalone machine, or in the Resources container, select a target **SPOT**.
3. From the Selected menu, choose **Troubleshooting** —> **Verify Installed Software**.
4. Use the dialog to select whether to verify all or some installed software on the selected machine or **SPOT**.

From SMIT

1. Enter the **smit nim_mac_op** fast path to check software on a machine, or enter **smit nim_res_op** to check software on a **SPOT**.
2. Select the target of the **lppchk** operation.
3. Select the desired verification mode.

From the Command Line

Enter the following command:

```
nim -o lppchk -a filesets=FilesetName \
-a lppchk_flags="lppchkFlags" ObjectName
```

where *FilesetName* is the name of a single fileset (or a name with the * wildcard character), and *ObjectName* is the name of the machine or **SPOT** which is the target of the **lppchk** operation. Valid **lppchk_flags** are defined as follows:

-f	Fast check (file existence, file length)
-c	Checksum verification
-v	Fileset version consistency check (default)
-l	File link verification **Note:** Only one of the flags **-f**, **-c**, **-v**, or **-l** may be specified.
-u	Update inventory (only valid with **-c** or **-l**)
-m*n*	Controls detail of messages. *n* equals 1 to 3, where 3 is the most verbose.

For example, to perform the **lppchk** operation while verifying checksums for all filesets on the machine named `Standalone1`, enter the following:

```
nim -o lppchk -a lppchk_flags="-c" Standalone1
```

Using NIM to Install Clients Configured with Kerberos Authentication

Normally, NIM relies on Standard AIX authentication to allow the NIM master to remotely execute commands. Standard AIX authentication uses the **.rhosts** file to provide this capability. While NIM functionality depends on its ability to remotely execute commands, some system environments require stricter authentication controls. Kerberos authentication provides a higher level of authentication for executing remote commands on the system without disabling NIM's capabilities.

Using NIM to Install Clients Configured with Kerberos 4 Authentication

In AIX 4.3.3 and later, NIM can be used to install machines in an RS/6000 SP environment configured for Kerberos 4 authentication. Clients configured for Kerberos 4 authentication will contain a **$HOME/.klogin** file for the root user. This file will determine what ticket is required to allow remote command execution. The user must obtain the required ticket before attempting to execute remote commands through NIM.

The NIM master and all secure clients must have the IBM Parallel System Support Program for AIX 3.1 (or later) installed and configured.

If secure clients will be reinstalled with BOS (Base Operating System), the authentication methods on the NIM master should be set for both Kerberos 4 and Standard UNIX. Because NIM will not have configured

Kerberos 4 on the client after the BOS is installed. NIM will therefore have to rely on a **.rhosts** file to guarantee that it can remotely execute commands on the client until the client can be configured with Kerberos 4 and made into a secure client.

If only software customization and maintenance will be performed, the NIM master must have its authentication methods set to match those of the clients. To manage secure clients, the master will need authentication methods set to include Standard UNIX.

For more information on installing and configuring Kerberos 4, see the *SP Administration Guide* (GC23–3897).

Using NIM to Install Clients Configured with Kerberos 5 Authentication

In AIX 4.3.2 and later, NIM can be used to install machines in an environment configured for Kerberos 5 authentication. Clients configured for Kerberos 5 authentication will contain a **$HOME/.k5login** file for the root user. This file will contain an entry that specifies what host token is required to allow remote command execution. This entry uses the following form:

```
hosts/hostname/self@cell
```

The NIM master and all secure clients must have DCE installed and configured at a level greater than or equal to 2.2.1.

If secure clients will be reinstalled with BOS, the authentication methods on the NIM master should be set for both Kerberos 5 and Standard UNIX. Because the client will not have DCE or Kerberos 5 configured and running after the BOS is installed. NIM will therefore have to rely on standard **rhosts** to remotely execute commands on the client until it can be configured with Kerberos 5 and made into a secure client.

If only software customization and maintenance will be performed, the NIM master must have its authentication methods set to match those of the clients. To manage secure clients, the master will need authentication methods set to include Standard UNIX.

Concurrency Control

NIM installations can become overburdened when they are being performed on a large number of clients at the same time. This can be caused by network bandwidth or workload on the NIM servers. Users can ease the severity of this situation by controlling the number of clients installing at the same time.

The **concurrent** and **time_limit** attributes can be used in conjunction with the **bos_inst**, **cust**, and **alt_disk_install** operations to control the number of client machines being operated on simultaneously from a client group. The **concurrent** attribute controls the number of clients in a group that are processing a particular operation at one time. After a client finishes the operation, another client will initiate the operation one at a time. The **time_limit** attribute prohibits NIM from initiating an operation on any more clients of the group, after the specified time (in hours) has elapsed.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select multiple targets.
3. From the Selected menu, choose any of the following options:
   - **Install Operating System**
   - **Install/Update Software**
   - **Alternate Disk Install**
4. From any of those dialogs, select the **NIM settings** or **Advanced** button.

5. In those dialogs, a section containing the Concurrency Controls can be specified.

Note: Web-based System Manager does not provide support for continuing after a failure or if the group of machines were individually selected and the time limit expired. The user must reselect the clients that failed or were not attempted and then reissue the command.

From SMIT

The Concurrency Control attributes can be accessed from all SMIT panels under the Install and Update Software menu and the Alternate Disk Installation menu.

From the Command Line

The **concurrent** and **time_limit** attributes can be used in conjunction with the **bos_inst**, **cust** and **alt_disk_install** operations.

For example, to have the **bos.games** fileset installed on only five machines from the client group **tmp_grp** at one time, enter the following command:

```
nim -o cust -a lpp_source=lpp_source1 -a filesets=bos.games \
-a concurrent=5 tmp_grp
```

In this example, to BOS install only 10 clients from **tmp_grp**, using **lpp_source**, `lpp_source1`, and SPOT, `spot1`, with no other installs permitted after three hours have elapsed, enter the following command:

```
nim -o bos_inst -a lpp_source=lpp_source1 -a spot=spot1 \
-a concurrent=10 -a time_limit=3 tmp_grp
```

Note: The Concurrency Controlled operation can complete and leave the group in one of the following states:

- All machines install successfully.
- Some machines may fail the installation.
- If the **time_limit** attribute was used, time may have expired before the installation operation was complete.

In the first situation, the group will revert to the state prior to the operation. In the second and third situations, the group will be left in a state that indicates some machines have completed and some have not. Problems with failing machines should be investigated. At this point, the user can continue with the machines that did not complete by rerunning the command on the group. Alternatively, the user can ″reset″ the group, which will set the group back to its state prior to the Concurrency Controlled operation.

Chapter 22. Advanced NIM Configuration Tasks

This chapter describes the following procedures for performing advanced Network Installation Management (NIM) configuration tasks using the Web-based System Manager NIM application or the System Management Interface Tool (SMIT), as well as from the command line:

- "Removing Machines from the NIM Environment"
- "Creating Additional Interface Attributes" on page 232
- "Defining /usr versus non-/usr SPOTs" on page 233
- "Re-Creating SPOT Resources from Existing Directories" on page 234
- "Defining an lpp_source on CD-ROM versus Disk" on page 235
- "Establishing a Default NIM Route Between Networks" on page 235
- "Establishing a Static NIM Route Between Networks" on page 236
- "Recovering the /etc/niminfo File" on page 237
- "Backing Up the NIM Database" on page 238
- "Restoring the NIM Database and Activating the NIM Master" on page 238
- "Unconfiguring the NIM Master" on page 239
- "Booting Diagnostics" on page 239
- "Booting in Maintenance Mode" on page 240
- "Secondary Adapter Support" on page 242

Removing Machines from the NIM Environment

Removing a machine from the NIM environment can be done by removing the client information from the NIM database.

Note: When a client is removed from the NIM environment, NIM attempts to remove the **/etc/niminfo** file from the client machine. However, the client fileset and rhost permission for the NIM master must be removed manually from the client system if such additional cleanup is desired.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a machine to remove from the NIM environment.
3. From the Selected menu, choose **Delete** to remove the machine.

From SMIT

1. To remove a machine from the NIM environment, enter the **smit nim_rmmac** fast path.
2. Select the machine to remove.
3. In the displayed dialog fields, accept the defaults.

From the Command Line

To remove a machine from the NIM environment, enter:

```
nim -o remove MachineName
```

where *MachineName* is the name of the machine to be removed.

© Copyright IBM Corp. 2002, 2004

Creating Additional Interface Attributes

The primary interface or the first interface (**if1**) is created when the master is activated, and a sequence number is used to identify the additional interfaces (**if2**, **if3,** ...) in the machine object definition. To create an additional **if** attribute for the master object, use either Web-based System Manager, SMIT, or the **nim -o change** command operation.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select any machine (master, standalone, diskless, or dataless).
3. From the Selected menu, choose **Properties**. The General page of the Machine Properties notebook for the selected machine displays.
4. Select the NIM Interfaces tab.
5. Follow the dialog instructions.

From SMIT

1. To create an additional **if** attribute, enter the **smit nim_mac_if** fast path.
2. Select the Define a Network Install Interface option.
3. Select the machine object name. In the example, this is `master`.
4. Enter the host name for the interface.
5. Complete the network-specific information in the entry fields on the Define a Network Install Interface screen.

   Note: If a NIM network does not already exist corresponding to the IP address of the host name specified for the interface, additional network information will be requested so the network can be defined.

From the Command Line

To create an additional **if** attribute for the master object, enter:

For Token-Ring:

```
nim -o change -a ifseq_no='NetworkObjectName AdapterHostName \
AdapterHardwareAddress' -a ring_speedseq_no=Speed master
```

For Ethernet:

```
nim -o change -a ifseq_no='NetworkObjectName AdapterHostName \
AdapterHardwareAddress' -a cable_typeseq_no=Type master
```

For FDDI:

```
nim -o change -a ifseq_no='NetworkObjectName AdapterHostName \
AdapterHardwareAddress' master
```

For other networks:

```
nim -o change -a ifseq_no='NetworkObjectName AdapterHostName \
AdapterHardwareAddress' master
```

Note: If you do not know the name of the NIM network to which the interface is attached or if a network corresponding to the interface has not been defined, use the **find_net** keyword and **net_definition** attribute as described in “Defining NIM Clients” on page 258.

In the example, the following command is run:

```
nim -o change -a if2='Network2 srv1_ent 0' -a \
cable_type2=bnc master
```

With this syntax, another **if** attribute is created for the master, which tells NIM that the master has an Ethernet interface that uses a host name of `srv1_ent`, that the Ethernet adapter's hardware address is `0` (not used), and that the master connects to the `Network2` network object.

To display detailed information about the master which will now show the **if2** attribute, enter:

```
lsnim -l master
```

The command produces output similar to the following:

```
master:
  class             =  machines
  type              =  master
  Cstate            =  ready for a NIM operation
  reserved          =  yes
  platform          =  rs6k
  serves            =  boot
  serves            =  nim_script
  comments          =  machine which controls the NIM environment
  Mstate            =  currently running
  prev_state        =  ready for a NIM operation
  if1               =  Network1 server1 10005AA88399
  master_port       =  1058
  registration_port =  1059
  ring_speed1       =  16
  if2               =  Network2 Srv1_ent 02608c2e222c
  cable_type2       =  bnc
```

Defining /usr versus non-/usr SPOTs

A **SPOT** resource contains operating system files that are normally installed in the **/usr** file system of a machine. If disk space is limited on a machine or a **SPOT** must be created quickly, it may be helpful to convert the machine's **/usr** file system to a **SPOT** instead of creating an entirely separate **SPOT** at a different location.

If the **/usr** file system of a machine is converted to a **SPOT**, additional software will be installed on the machine to provide support for machines with different hardware configurations. Most of the operating system files will already be installed on the system and will not be reinstalled when the **SPOT** is created.

After a **/usr** file system is converted to a **SPOT**, all software installation and maintenance operations on the machine should be performed using NIM on the **/usr SPOT** resource that was created. This will ensure that all necessary **SPOT** operations are performed in addition to software installation or maintenance on the machine.

From Web-based System Manager

1. Select the Resources container.
2. From the Resources menu, select **New Resource**.
3. Follow the wizard instructions to create the **SPOT** resource.

From SMIT

1. To create a **/usr SPOT**, enter the **smit nim_mkres** fast path.
2. Select the Resource Type.
3. Type **/usr** in the Location of Resource field.
4. Supply the values or accept the defaults for all other fields on this screen.

From the Command Line

To create a **/usr-SPOT**, enter:

```
nim -o define -t spot -a server=ServerName \
-a location=/usr -a source=SourceName ResourceName
```

Example:

To convert the **/usr** file system on the machine, `client1`, to a **SPOT** named `usrspot` using `lppsource1` as the source for additional installation images, enter:

```
nim -o define -t spot -a server=client1 -a location=/usr \
-a source=lpp_source1 usrspot
```

Using the installp Command

After you convert a **/usr** file system to a **SPOT**, it is not recommended that you use the **installp** command to install or maintain software on the machine serving the **SPOT**. The diskless and dataless clients and network boot images associated with the **SPOT** will not be updated by the **installp** command unless it is invoked using NIM's **cust** or **maint** operations. If you need to use the **installp** command to install or maintain software on a **/usr SPOT** server, use the following steps:

1. Ensure that all NIM operations on the server and any clients associated with the **SPOT** are complete.
2. Deallocate the **SPOT** from all standalone clients.
3. Run the **installp** command.
4. Run the **check** operation on the **SPOT** after the **installp** command has completed:

   ```
   nim -o check -F usrSPOTName
   ```

   Note: The **-F** flag is required for rebuilding the boot images.
5. If this **SPOT** is being used to serve diskless or dataless clients, resynchronize all diskless and dataless clients with the **SPOT** after the **installp** command completes by issuing the **nim** command with the **sync_roots** operation for the **/usr SPOT**:

   ```
   nim -o sync_roots usrSPOTName

   nim -o check -F usrSPOTName
   ```

   The **cust** and **maint** operations must be used to manage software installed on non-**/usr SPOT**s.

Re-Creating SPOT Resources from Existing Directories

Defining NIM resources from existing files and directories can usually be done by specifying the **server** and **location** attributes to the **nim - o define** command. SPOT resources take longer to define because software must be installed from installation images into the SPOT location. The **nim -o** command line interface always builds a SPOT from installation images. However, if a directory structure for a SPOT already exists from a prior creation, it is possible to call a NIM method directly to redefine the SPOT without reinstalling all the software.

The need to define a SPOT from an existing SPOT directory typically arises only when it is necessary to rebuild the NIM database during system recovery.

To define a SPOT from a directory that previously had a SPOT installed in it, use the following command:

```
/usr/lpp/bos.sysmgt/nim/methods/m_mkspot -o -a server=server \
-a location=location -a source=no  spotname
```

Example:

A SPOT named **spot1** was created on the NIM master in the **/export/spot directory**. Later, the NIM database became corrupted and has to be rebuilt. The SPOT files are still on the machine, but the SPOT must be redefined to NIM using the following command:

```
/usr/lpp/bos.sysmgt/nim/methods/m_mkspot -o -a server=master \
 -a location=/export/spot -a source=no spot1
```

Defining an lpp_source on CD-ROM versus Disk

Normally an **lpp_source** resource is created by copying installation images from installation media to the hard disk of the **lpp_source** server. If disk space is limited on the server or if an **lpp_source** is needed quickly, you can use a directory mounted from CD-ROM installation media as the **lpp_source**.

From Web-based System Manager

1. Select the Resources container.
2. From the Resources menu, select **New Resource**.
3. Follow the wizard instructions to create the **lpp_source** resource.

Note: You can also define an **lpp_source** resource through the Configure NIM wizard, both when you are configuring your environment, and after configuration.

From SMIT

1. Mount the CD as a **CDROM** file system. The installation images can be found in the **/usr/sys/inst.images** directory under the mount point of the **CDROM** file system.
2. To define the **lpp_source** using the directory of install images, enter the **smit nim_mkres** fast path.
3. Specify the name of the machine with the CD-ROM as the Server.
4. Specify *CD_MountPoint*/**usr/sys/inst.images** as the location of the **lpp_source**, and leave the Source field blank.

From the Command Line

1. Mount the CD as a **CDROM** file system. The installation images can be found in the **/usr/sys/inst.images** directory under the mount point of the **CDROM** file system.
2. Define the **lpp_source** using the directory of install images for the **location** attribute. Do not specify a value for the **source** attribute, since an existing set of images will be used. With the CD mounted at **/cdfs** on the NIM master, to define an **lpp_source** named `cd_images`, enter:

   ```
   nim -o define -t lpp_source -a server=master \
   -a location=/cdfs/usr/sys/inst.images cd_images
   ```

Establishing a Default NIM Route Between Networks

This procedure describes how to create default NIM routes for two Networks (for example, `Network1` and `Network3`).

From Web-based System Manager

1. Select the Networks container.
2. In the contents area, select any network.
3. From the Selected menu, choose **Properties**. The General page of the Properties notebook for the selected network displays.
4. Select the NIM Routes tab. The NIM Routes page of the Properties notebook displays.
5. Use the NIM Routes page to add the default route.

From SMIT

1. To create default NIM routes, enter the **smit nim_mkdroute** fast path.
2. In the displayed dialog fields, supply the values or accept the defaults. Use the help information and the LIST option to help you.

From the Command Line

To create a default NIM route for a network, enter:

```
nim -o change -a routingseq_no='default Gateway' NetworkObject
```

where `default` is the reserved keyword used by NIM to indicate a default route, and *Gateway* is the host name (or IP address) of the interface that clients on *NetworkObject* use to contact other networks in the NIM environment.

For example, to establish default NIM routes for `Network1` and `Network3`, enter:

```
nim -o change -a routing1='default gw1_tok' Network1
nim -o change -a routing1='default gw1_fddi' Network3
```

where `gw1_tok` is the host name of the default gateway for machines on `Network1`, and `gw1_fddi` is the host name of the default gateway for machines on `Network3`.

The detailed information for the network objects now shows the added default routes. To display the detailed information for the two networks, enter:

```
lsnim -l Network1 Network3
```

which produces output similar to the following:

```
Network1:
        class           = networks
        type            = tok
        net_addr        = 9.101.1.0
        snm             = 255.255.255.0
        Nstate          = ready for use
        prev_state      = ready for use
        routing1        = default gw1_tok

Network3:
        class           = networks
        type            = fddi
        net_addr        = 9.101.3.0
        snm             = 255.255.255.0
        Nstate          = ready for use
        prev_state      = information is missing from this
                          object's definition
        routing1        = default gw1_fddi
```

Establishing a Static NIM Route Between Networks

This procedure describes how to create a static NIM route between two networks (for example, `Network1` and `Network3`).

From Web-based System Manager

1. Select the Networks container.
2. In the contents area, select any network.
3. From the Selected menu, choose **Properties**. The General page of the Properties notebook for the selected network displays.
4. Select the NIM Routes tab. The NIM Routes page of the Properties notebook displays.

5. Use the NIM Routes page to add the static route.

From SMIT

1. To create a static NIM route, enter the **smit nim_mkroute** fast path.
2. In the displayed dialog fields, supply the values or accept the defaults. Use the help information and the LIST option to help you.

From the Command Line

To create a static NIM route between two networks, enter:

```
nim -o change -a routingseq_no='DestinationNetworkObject \
Gateway1 Gateway2' NetworkObject
```

where *Gateway1* is the host name of the interface that clients on *NetworkObject* use to get to *DestinationNetworkObject*, and *Gateway2* is the host name that clients on *DestinationNetworkObject* use to get back to *NetworkObject*.

For example, to establish a NIM route between Network1 and Network3, enter:

```
nim -o change -a routing1='Network3 gw1_tok gw1_fddi' Network1
```

where gw1_tok is the host name of the gateway that machines on Network1 use to communicate with machines on Network3, and gw1_fddi is the host name of the gateway that machines on Network3 use to communicate with machines on Network1.

The detailed information for the network objects now shows the added routing attributes.

To display the detailed information about the two networks, enter:

```
lsnim -l Network1 Network3
```

The command produces output similar to the following:

```
   Network1:
      class       =  networks
      type        =  tok
      net_addr    =  9.101.1.0
      snm         =  255.255.255.0
      Nstate      =  ready for use
      prev_state  =  ready for use
      routing1    =  Network3 gw1_tok

   Network3:
      class       =  networks
      type        =  fddi
      net_addr    =  9.101.3.0
      snm         =  255.255.255.0
      Nstate      =  ready for use
      prev_state  =  information is missing from this object's
                     definition
      routing1    =  Network1 gw1_fddi
```

Recovering the /etc/niminfo File

The **/etc/niminfo** file, which resides on the master and running NIM clients, is required to run NIM commands and perform NIM operations. If the **/etc/niminfo** file is accidentally deleted, you can rebuild the file.

From Web-based System Manager

1. Select the NIM container.
2. From the NIM menu, select **Advanced Configuration —> Rebuild Master Configuration File**.

Note: The Configure NIM wizard will detect when you do not have a **niminfo** file, yet do have NIM database entries. The wizard queries whether it should rebuild the master configuration file.

From the Command Line

Enter the following command from the master to rebuild the file:

```
nimconfig -r
```

To rebuild the **/etc/niminfo** file from a running NIM client, enter:

```
niminit -a master_port=PortNumber -a master=MasterHostName \
-a name=ClientMachineObjectName
```

Backing Up the NIM Database

To back up the NIM database, you will be prompted for the name of a device or a file to which the NIM database and the **/etc/niminfo** file will be backed up. The level of the installed NIM master fileset will also be written to a file called **/etc/NIM.level** and saved in the backup. A backup of a NIM database should only be restored to a system with a NIM master fileset which is at the same level or a higher level than the level from which the backup was created.

From Web-based System Manager

1. From the NIM container, from the NIM menu, select **Back Up Database**.
2. Use the dialog to specify the backup device or file.

From SMIT

To back up the NIM database, enter the **smit nim_backup_db** fast path.

From the Command Line

Save the following NIM files:

/etc/niminfo

/etc/objrepos/nim_attr

/etc/objrepos/nim_attr.vc

/etc/objrepos/nim_object

/etc/objrepos/nim_object.vc

Restoring the NIM Database and Activating the NIM Master

Note: A NIM database should only be restored to the same or later level of NIM that was used for the backup.

From Web-based System Manager

1. From the NIM container, from the NIM menu, select **Restore Database**.
2. Use the dialog to specify the restore device or file.

From SMIT

To configure a NIM master from a NIM database backup, enter the **smit nim_restore_db** fast path.

From the Command Line

Restore the files saved in “Backing Up the NIM Database” on page 238.

Unconfiguring the NIM Master

This operation removes the NIM daemons from the system and removes all configuration from the NIM database. The NIM master should only be unconfigured if the NIM environment is to be completely redefined or if the NIM master fileset is to be removed from the system.

From Web-based System Manager

1. From the NIM Container, from the NIM menu, select **Unconfigure Environment**.
2. You have the option to back up the NIM database before starting the unconfigure action.

From SMIT

To unconfigure a NIM master, enter the **smit nim_unconfig** fast path.

The SMIT screen will prompt you to first back up your NIM database before unconfiguring the NIM master.

From the Command Line

To unconfigure a NIM master, enter **nim -o unconfig master**.

Booting Diagnostics

Hardware diagnostics can be performed on all NIM clients using a diagnostic boot image from a NIM server, rather than booting from a diagnostic tape or CD-ROM. This is useful for standalone clients, because the diagnostics do not have to be installed on the local disk. Diagnostic support comes from a **SPOT** resource.

From Web-based System Manager

1. Select the NIM container.
2. In the contents area, select the standalone, diskless, or dataless machine you want to enable for diagnostics boot.
3. From the Selected menu, choose **Troubleshooting** —> **Enable Diagnostic Boot**.
4. Use the dialog to select a **SPOT** resource from which to boot.

From SMIT

Initiating the diag Operation from the Client

1. Enter the **smit nim_client_op** fast path.
2. Select the **diag** operation from the displayed list of operations.

Initiating the diag Operation from the Master

1. Enter the **smit nim_mac_op** fast path.
2. Select the machine object.
3. Select the **diag** operation from the list of operations.

From the Command Line

To perform the **diag** operation from the client, enter:

```
nimclient -o diag -a spot=SPOTName
```

To perform the **diag** operation from the master, enter:

```
nim -o diag -a spot=SPOTName MachineObjectName
```

Verifying the diag Operation

After you have enabled the client to perform a diagnostic boot, you can verify the success of the operation by querying the client's *control state* (**Cstate**).

On the client, enter:

```
nimclient -l -l ClientMachineObjectName
```

On the master, enter:

```
lsnim -l ClientMachineObjectName
```

If the operation is successful, output similar to the following is displayed:

```
Cstate = Diagnostic boot has been enabled
```

For the client to boot the diagnostics, you need to reboot the client. If it is a diskless or a dataless client, you have already defined a network adapter as the default boot device (BOOTP request), so no additional action is required. For a standalone machine, the boot list for normal boot lists the hard disk as the primary boot device, so you must follow the procedure described in "Booting a Machine Over the Network" on page 343.

Loading Diagnostics without the diag Operation

In addition to using the procedure in the previous section, diskless and dataless clients have another way of loading diagnostics from the network. You can boot a diskless or dataless client from the network the same way you do for normal use, but with the machine's key mode switch in the Service position. If the client's key mode switch is in the Service position at the end of the boot process, hardware diagnostics from the server's **SPOT** are loaded. If a standalone client boots with the key mode switch in the Service position, the diagnostics (if installed) are loaded from the hard disk.

Booting in Maintenance Mode

If you need to perform maintenance on a standalone machine that is not part of the NIM environment, the system must be booted from a bootable tape or CD-ROM. This may require connecting an external device. If the machine is part of a NIM environment, you can enter maintenance mode directly by enabling the **maint_boot** operation for a NIM standalone machine.

From Web-based System Manager

1. Select the Machines container.
2. In the contents area, select a target standalone machine you want to enable for maintenance boot.
3. From the Selected menu, choose **Troubleshooting** —> **Enable Maintenance Boot**.
4. Use the dialog to select a **SPOT** resource from which to boot.

From SMIT

Initiating the maint_boot Operation from the Client

1. Enter the **smit nim_client_op** fast path.
2. Select the **maint_boot** operation.
3. Select the **SPOT** to be used for the operation.
4. Press Enter to enable the client for maintenance boot.

Initiating the maint_boot Operation from the Master

1. Enter the **smit nim_mac_op** fast path.
2. Select the client's machine object.
3. Select the **maint_boot** operation.
4. Select the **SPOT** to be used for the operation.
5. Press Enter to enable the client for maintenance boot.

From the Command Line

To issue the **maint_boot** operation from the client, enter:

```
nimclient -o maint_boot -a spot=SPOTNAME
```

To issue the **maint_boot** operation from the master, enter:

```
nim -o maint_boot -a spot=SPOTNAME CLIENT
```

To verify that the maintenance boot operation worked:

1. On the client, enter:

   ```
   nimclient -l -l ClientMachineObjectName
   ```

2. On the master, enter:

   ```
   lsnim -l  ClientMachineObjectName
   ```

If the operation was successful, the client's **Cstate** output will look similar to the following:

```
Cstate = maintenance boot has been enabled
```

For the machine to boot into maintenance mode, follow the procedure for issuing the BOOTP request from the client. See "Booting a Machine Over the Network" on page 343 for more information about initiating a BOOTP request.

Using Maintenance Mode

After successfully booting and defining the console, the System Maintenance menu is displayed. The maintenance menu options and their descriptions are described below. For more information about maintenance mode, see "Accessing a System That Does Not Boot" on page 163.

Access a Root Volume Group	This option allows you to activate the root volume group and start the maintenance shell with a full set of commands.
Copy a System Dump to Removable Media	This option allows you to copy a previous system dump to external media.
Access Advanced Maintenance Function	This option allows you to start a maintenance shell with a limited set of commands.

Secondary Adapter Support

Previously, during a NIM **rte** BOS installation operation, only the network adapter and interface used during BOS installation were configured. Using NIM secondary adapter definitions, you can have additional network adapters and interfaces configured during a BOS installation or customized installation.

The **nimadapters** command parses a secondary adapter stanza file to build the files required to add NIM secondary adapter definitions to the NIM environment as part of an **adapter_def** resource. The **nimadapters** command does not configure secondary adapters. The configuration takes place during a **nim -o bos_inst** operation or a **nim -o cust** operation that references the **adapter_def** resource.

Secondary adapter support is available for AIX 5.2 or later. Before you enable a secondary adapter, you must verify the AIX version the client is running. Secondary adapter support is not available for clients at AIX 5.1 or earlier. The secondary adapters will fail to configure, because NIM is unable to find the **/usr/lpp/bos.sysmgt/nim/methods/c_cfgadptrs** client method. The following example shows the outcome if you attempt to enable this support on your NIM master.

```
nim -o cust -a adapter_def=adapter_def1 rspc10
trigger.austin.xyz.com. 0042-001 nim: processing error encountered on "master":
0042-001 m_cust: processing error encountered on "rspc10":
0042-175 c_script: An unexpected result was returned by the
"trigger.austin.xyz.com:/export/nim/scripts/rspc10.script" command:
/tmp/_nim_dir_4714/script[10]: /usr/lpp/bos.sysmgt/nim/methods/c_cfgadptrs:  not found.
```

The secondary adapter stanza file is processed by the **nimadapters** command and turned into a file that contains one stanza for each secondary adapter or interface on the NIM client. During a BOS installation, NIM processes this information and configures the secondary adapters.. If a secondary adapter is already configured in the requested manner, NIM does not reconfigure the secondary adapter.

Note: Before using the **nimadapters** command, you must configure the NIM master. For information on configuring the NIM master, see “Configuring the NIM Master and Creating Basic Installation Resources” on page 190.

Secondary Adapter File Rules

The format of the secondary adapter file must comply with the following rules:

- After the stanza header, follow attribute lines of the form: *Attribute* = *Value*
- If you define the value of an attribute multiple times within the same stanza, only the last definition is used.
- If you use an invalid attribute keyword, that attribute definition is ignored.
- Each line of the file can have only one header or attribute definition.
- More than one stanza can exist in a definition file for each machine host name.
- Each stanza for a machine host name represents a secondary adapter definition on that NIM client. No two secondary adapter definitions for the same machine host name can have the same location or **interface_name**. There should be only one definition per adapter or interface on a given NIM client.
- If the stanza header entry is the **default** keyword, this specifies to use that stanza for the purpose of defining default values.
- You can specify a default value for any secondary adapter attribute. However, the **netaddr** and **secondary_hostname** attributes must be unique. Also, the **location** and **interface_name** attributes must be unique on a NIM client.
- If you do not specify an attribute for a secondary adapter but define a default value, the default value is used.
- You can specify and change default values at any location in the definition file. After a default value is set, it applies to all definitions that follow.

- To turn off a default value for all following machine definitions, do not set the attribute value in a default stanza.
- To turn off a default value for a single machine definition, do not set the attribute value in the machine stanza.
- You can include comments in a client definition file. Comments begin with the number sign (**#**).
- When parsing the definition file for header and attribute keywords and values, tab characters and spaces are ignored.

Note: During a **nim -o bos_inst** or **nim -o cust** operation, if NIM examines the configuration data on the client and determines that a secondary adapter is already configured with precisely the attributes requested in the **adapter_def** resource, this secondary adapter is not reconfigured.

Secondary Adapter File Keywords

The secondary adapter file uses the following keywords to specify machine attributes:

Required Attributes

machine_type = secondary
Specifying the **machine_type** attribute as **secondary** clearly distinguishes the **nimadapters** input from **nimdef** input. If a secondary adapter's file is mistakenly passed to the **nimdef** command, the error can be detected.

netaddr
Specifies the network address for the secondary adapter.

network_type = en | et | sn | ml
Specifies the type of network interface, which can be one of **en**, **et**, **sn**, or **ml**.

subnet_mask
Specifies the subnet mask used by the secondary adapter.

Note: Configuring a secondary adapter on the same subnet as another adapter does not provide failover. Packets alternate between adapters when they are configured on the same subnet. If one of the adapters fails, the other adapter will not take over the failed adapter's workload, and the subnet will have connectivity problems. Commands, such as **mount**, might fail if this occurs.

Optional Attributes

attributes
Blank-separated list of interface attributes and values. For example, *Attribute1=Value1 Attribute2=Value2*. To see the list of attributes that can be set for the requested interface, run the command **lsattr -E -l** *InterfaceName*.

cable_type
Specifies the cable type (optional if **network_type** is **en** or **et**).

comments
Specifies a comment to include in the secondary adapter definition. Enclose the comment string in quotation marks.

interface_name
Specifies the name of the network interface for the secondary adapter (for example, **en1**, **sn0**, **ml0**). Do not specify both **location** and **interface_name**.

Note: The value of the **interface_name** attribute must be consistent with the value of the **network_type** attribute.

location
Specifies the physical location of the adapter corresponding to this network interface. Do not specify both the **location** and the **interface_name** attributes.

Note: Except for the multilink pseudo-device, use of the **location** attribute is highly recommended. If the **location** attribute is not specified and the user adds multiple adapters or adds an adapter at the same time that the operating system is reinstalled, the adapter and network interface names might be reassigned by the operating system in unexpected ways.

media_speed
Specifies the media speed (optional if the **network_type** attribute's value is either **en** or **et**).

secondary_hostname
Host name to save in the **/etc/hosts** file with the **netaddr** attribute. This host name is not set using the **hostname** command or the **uname -S** command.

Secondary Adapter File Stanza Errors

A secondary adapter stanza causes an error under any of the following conditions:

- The host name that was used in the stanza header for the definition cannot be resolved.
- A required attribute is missing.
- An invalid value was specified for an attribute.
- An attribute mismatch occurs. For example, if the **network_type** attribute's value is not set to either **en** or **et**, you cannot specify **cable_type**=**bnc** or **media_speed**=**1000_Full_Duplex**.
- The stanza contains both a **location** attribute and an **interface_name** attribute.
- Secondary adapter definitions occur multiple times for the same adapter location and the same host name.
- Secondary adapter definitions occur multiple times for the same **interface_name** and the same host name.

If a secondary adapter stanza is incorrect, the errors are reported, the stanza is ignored, and the following input is processed without regard to the incorrect stanza.

Example Secondary Adapter File

The following is an example of a secondary adapter file:

```
# Set default values.
default:
    machine_type  = secondary
    subnet_mask   = 255.255.240.0
    network_type  = en
    media_speed   = 100_Full_Duplex

# Define the machine "lab1"
# Take all defaults and specify 2 additional attributes.
# Unlike the case of the client definitions that are input to the
# nimdef command, the secondary adapter definition includes at least
# one required field that cannot be defaulted.
lab1:
    netaddr = 9.53.153.233
    location = P2-I1/E1

# Change the default "media_speed" attribute.

default:
    media_speed   = 100_Half_Duplex

# define the machine "test1"
# Take all defaults and include a comment.
test1:
    comments      = "This machine is a test machine."
```

Examples

1. To preview the **secondary_adapters.defs** client definition file, type:

   ```
   nimadapters -p -f secondary_adapters.defs adapter_def
   ```

2. To add the NIM secondary adapters described in the **secondary_adapters.defs** secondary adapters definition file, type:

   ```
   nimadapters -d -f secondary_adapters.defs adapter_def
   ```

3. To define the NIM secondary adapters for the **pilsner** client, type:

   ```
   nimadapters -d \
   -a info="en,P2-I1/E1,N/A,1000_Full_Duplex,9.53.153.233,255.255.254.0" \
   -a client=pilsner adapter_def
   ```

Chapter 23. Additional NIM Topics

This chapter describes Network Installation Management (NIM) topics that are not part of the usual installation procedures.

The following topics are included:

- "NIM Master Management Tasks"
- "NIM Name Resolution" on page 249
- "Booting a FDDI Interface Over a Router" on page 249
- "Default Paging Space During BOS Installation Through NIM" on page 250
- "Migrating Diskless and Dataless Clients and NIM SPOTS" on page 250
- "Defining the NIM Environment Using the nimdef Command" on page 251
- "Name Requirements for NIM Object Definitions" on page 251
- "Interacting with the Dynamic Host Configuration Protocol" on page 251
- "Creating File Resources in the root Directory" on page 251
- "Restricting NIM Client Resource Allocation" on page 252
- "Preventing Machines from Adding Themselves as Clients" on page 252
- "Disabling Client CPU ID Validation" on page 253
- "Exporting NIM Resources Globally" on page 253
- "Creating Network Boot Images to Support Only the Defined Clients and Networks" on page 254
- "Updating a Spot with New Device Support for a New Level of AIX" on page 255
- "Tuning Client-Request Processing" on page 255

NIM Master Management Tasks

The following tasks can be performed on the NIM master:

- "Deactivating the NIM Master and Removing the NIM Master Fileset"
- "Increasing the Number of Hosts to Which NIM Can NFS-Export a Resource" on page 248
- "Controlling the Asynchronous Behavior of NIM Operations" on page 248
- "Suppressing Output from NIM Operations" on page 248
- "Reducing Space Requirements for NIM Resources" on page 249

Deactivating the NIM Master and Removing the NIM Master Fileset

After the NIM master fileset has been installed, the master activated, and the master object defined in the NIM database, this object, and hence the master fileset itself, cannot be removed. The master must be deactivated before the NIM master fileset can be removed.

To deactivate the master using Web-based System Manager, see "Unconfiguring the NIM Master" on page 239.

To use the command line to deactivate the master and remove the NIM master fileset, enter:

```
nim -o unconfig master
installp -u bos.sysmgt.nim.master
```

© Copyright IBM Corp. 2002, 2004

Increasing the Number of Hosts to Which NIM Can NFS-Export a Resource

By default, when NIM exports a file or directory through NFS during resource allocation, it creates an entry in the **/etc/exports** file granting the target host both client mount access and root access for root users. As a result, when exporting to numerous clients, the limit on the length of a line in the exports file (32767 characters) may be exceeded, resulting in failure.

NIM provides an option to decrease the line length of an allocation entry in an NFS exports file by approximately one-half, effectively permitting files to be allocated to a greater number of hosts. This action has the side effect of increasing the number of machines permitted in a NIM machine group. NIM achieves this by only granting root access to allocation target hosts. The client mount access list is not created, which allows any machine to mount the resource, but still restricts root access to NIM clients only. NFS permits no more than 256 host names in a root exports file entry.

To enable this mode of operation, set the **restrict_nfs_exports** attribute to `no` on the master's NIM object. Use the **change** operation as follows:

```
nim -o change -a restrict_nfs_exports=no master
```

To restore client mount access restrictions, set **restrict_nfs_exports** to `yes` with the **change** operation.

For information about how to export NIM resources globally, see "Exporting NIM Resources Globally" on page 253.

Controlling the Asynchronous Behavior of NIM Operations

Certain NIM operations are asynchronous, meaning that NIM master might initiate the operation on the client, but does not wait for the operation to finish. The reason for this asynchronous behaviour is because the NIM operation running on the client is typically time-consuming. An example of an asynchronous operation is the **bos_inst** operation. Examples of synchronous operations are the **cust**, **maint**, and **lppchk** operations on a single machine target. However, these operations, when applied to members of a machine group, are asynchronous. The **nim** command initiates these operations on each member of the group without waiting for the operation to finish.

If desired, the asynchronous behavior of the **cust**, **maint**, and **lppchk** operations can be controlled by setting the **async** attribute on the command line. For example, to ensure that the execution of a customization script identified by the NIM resource `script1` is executed completely on a given member of the group `MacGrp1` before initiating execution of the script on the next member of the group, enter the following:

```
nim -o cust -a script=script1 -a async=no MacGrp1
```

To force the master to not wait for the customization operation to finish when running the script on machine `Standalone1` that is not part of a machine group, enter:

```
nim -o cust -a script=script1 -a async=yes Standalone1
```

Suppressing Output from NIM Operations

By default, progress messages are displayed by the **nim** command operating on machine groups to inform the user of how much processing remains. Similarly, the output from the installation and customization programs invoked by the **cust** and **maint** operations on **SPOTs** and machines is also displayed. This output can be suppressed by setting the **show_progress** attribute to `no` on the command line. For example, to indicate to NIM not to display output from the **installp** command when updating the machine `Standalone1` with software from the **lpp_source** named `images1`, enter the following command:

```
nim -o cust -a show_progress=no -a lpp_source=images1 \
-a fixes=update_all Standalone1
```

Reducing Space Requirements for NIM Resources

It is not unusual for resources such as the **SPOT** and **lpp_source** to take several hundred megabytes of storage space on a NIM server. By creating **/usr SPOTs** and defining CD-ROM file-system directories as **lpp_sources**, space consumption can be reduced significantly on resource servers.

A **/usr SPOT** can be created from the **/usr** file system of the NIM master or any NIM client. The AIX system files for the BOS are already installed, so only software for additional device support will be added to the system. The resulting system ultimately has more software installed on it than it needs to run, but far less disk space is used than otherwise would have been, had a **non-/usr SPOT** been created on the same system. For more information on creating **/usr SPOT** resources, see “SPOT (Shared Product Object Tree) Resource” on page 278 and “Defining /usr versus non-/usr SPOTs” on page 233.

A directory on the AIX product CD can be mounted and defined as an **lpp_source**, eliminating the need to copy installation images to the hard disk of a resource server. The defined **lpp_source** contains all the images available on the CD, but the CD must remain mounted at the server for the **lpp_source** to be usable in NIM operations. For more information about using a CD-ROM file system as an **lpp_source**, see “Defining an lpp_source on CD-ROM versus Disk” on page 235.

NIM Name Resolution

NIM relies on standard AIX library routines to perform name resolution. If a network environment uses multiple sources for name resolution, NIM will resolve host names by querying the sources in whatever order is specified for the system. For example, if a system is configured to resolve host names by first querying NIS, then BIND/DNS, then a local **/etc/hosts** file, NIM will also follow that order when resolving client host names.

Problems may result if the NIM master and the NIM clients use different orders when querying sources for name resolution. Problems may also arise if a name service is available to one machine but not to another, causing different name resolution sources to be used.

Note: Mixing BIND/DNS, which is not case-sensitive, with NIS, which is case-sensitive, may result in problems.

It is possible to override the default system-wide order that AIX and NIM use when querying sources for host name resolution. This can be done by setting the **NSORDER** environment variable in the environment where NIM commands are being run. For example, to configure the environment to query NIS first, then BIND/DNS, then a local **/etc/hosts** file, type the following on the command line where NIM operations are being run:

```
export NSORDER=nis,bind,local
```

For more information on TCP/IP name resolution, refer to *AIX 5L Version 5.2 System Management Guide: Communications and Networks*.

Booting a FDDI Interface Over a Router

Boot over a router on a FDDI interface only if the router supports all-route broadcast. Booting over a router that does not support all-route broadcast on a FDDI interface may fail due to known limitations of these router types.

Default Paging Space During BOS Installation Through NIM

In AIX 4.3 or later, default paging space is set by the BOS installation process when installing through NIM. Default paging space is set by the BOS installation process, if the following conditions are met:

- The method of installation is **overwrite**.
- Neither an **image_data** resource nor an **image.data** file on the diskette is specified for the installation.
- The source of the BOS image is not a **mksysb** image.
- The source of the BOS image is a **SPOT**, and the default **image.data** file contains more than one entry for paging. This file is located at:

  `(spot_location)/lpp/bosinst/image_template`
- The source of the BOS image is a **SPOT**, and the LPs value for the single paging entry is set to the default value of **16**.

The default paging size is calculated from the smaller value of **optimal_ps** and **recommended_ps** where:

RAM = amount of memory on the target system measured in megabytes (MB).

optimal_ps = maximum between **RAM** and (0.2 size of rootvg)

IF CDE (Common Desktop Environment) is installed, **recommmended_ps** =

– amount of **RAM** is less than 32 MB, then **recommended_ps** = 3 * **RAM**

– amount of **RAM** is 32 MB or more, then **recommended_ps** = **RAM** + 64 MB

IF CDE (Common Desktop Environment) is not installed, **recommmended_ps** =

– amount of **RAM** is less than 32 MB, then **recommended_ps** = 2 * **RAM**

– amount of **RAM** is 32 MB or more, then **recommended_ps** = **RAM** + 32 MB

The default paging space set by this process is never greater than 512 MB.

Migrating Diskless and Dataless Clients and NIM SPOTS

Migration to a new release of AIX is not supported for diskless and dataless clients. Also, migration of a **SPOT** that is not a converted **/usr** file system is not supported.

After migrating a machine that is a **SPOT** server to a new release of AIX, you must remove and redefine the **SPOT** in order to also bring it to the new AIX level.

To remove and redefine the **SPOT**, enter:

```
nim -o remove SPOT_name
nim -o define -t spot -a location=SPOTDirectory \
-a server=SPOTServer -a source=SPOTSource SPOTName
```

A **/usr SPOT** served by a client in the NIM environment can be reinstalled with a new level of AIX using the migration procedure, but the **SPOT** object must be removed and then redefined after the migration completes. Any diskless or dataless clients served by that **SPOT** must be reinitialized. To reinitialize diskless and dataless clients after migrating a **/usr SPOT** server, deallocate, then reallocate the root resources, and then perform the **dtls_init** or **dkls_init** operation accordingly.

To reinitialize diskless and dataless clients, enter:

```
nim -o reset -F ClientName
nim -o deallocate -a root=RootResourceName ClientName
nim -o allocate -a root=RootResourceName ClientName
nim -o dkls_init ClientName
```

Attention: Any customization that was done previously will be erased, because deallocating the root resource will delete all the files in the root directory.

Defining the NIM Environment Using the nimdef Command

The **nimdef** command assists administrators when defining complex NIM environments and adding large numbers of client machines.

The **nimdef** command solves a common usability problem when defining large NIM environments.

Regardless of how well a NIM environment is understood, it can be a very time-consuming process to execute all the commands necessary to define it. If NIM could process a simple definition file for configuration of the NIM environment, a great deal of time could be saved that would otherwise be spent defining each network and machine manually.

The **nimdef** command reads a definition file for input. The definition file is in a structured stanza format. Each stanza describes a machine that will be added to the NIM environment. Included in the stanza is information about the machine's network adapter and routing configuration. Based on the supplied information, the **nimdef** command can determine the remaining information needed to define both networks and machines in the NIM environment.

For more information, see the **nimdef** command. For a sample definition file for the **nimdef** command, see Chapter 25, "Sample Files," on page 305.

Name Requirements for NIM Object Definitions

The name that you give a NIM object will be used in all future operations involving that object. This name must be unique among NIM objects, and it must adhere to certain restrictions:

- It must have between 1 and 39 characters.
- Valid NIM name characters include the uppercase and lowercase letters of the alphabet, the numbers 0-9, and the underscore character.
- Invalid NIM name characters include the dot character, all shell metacharacters, all file system metacharacters, and all regular expression metacharacters.

Interacting with the Dynamic Host Configuration Protocol

Select your NIM master to be the same system as the Dynamic Host Configuration Protocol (DHCP) server when using NIM in an environment that uses DHCP. Use host names whenever possible when defining NIM machine objects.

For more information, refer to DHCP and Network Installation Management (NIM) Interactions and Suggestions in *AIX 5L Version 5.2 System Management Guide: Communications and Networks*.

Creating File Resources in the root Directory

Due to a limitation in NFS, file resources, such as **bosinst_data** and **script** resources cannot be created in the root directory ("**/**") of a resource server.

Restricting NIM Client Resource Allocation

NIM provides client machines with the capability of allocating and using any resource in the NIM environment. In some tightly controlled NIM environments, administrators may not want clients to be able to access all resources at all times. To control client-resource allocation, a NIM administrator can use the **client_alloc** attribute. The restrictions placed by the **client_alloc** attribute will prevent clients from allocating and using resources, but the NIM master will continue to have the full capability of performing operations on clients.

Note: This task is not supported by Web-based System Manager.

From SMIT

NIM client-allocation restrictions can be changed from the SMIT interface by typing the SMIT fast path:

```
smit nim_control_alloc
```

From the Command Line

To restrict all clients from being able to use any resources, set the attribute **client_alloc=no** on the NIM master:

```
nim -o change -a client_alloc=no master
```

To restrict a particular client from being able to use any resources, set the attribute **client_alloc=no** on the client:

```
nim -o change -a client_alloc=no clientname
```

To restrict all clients from being able to use a particular resource, set the attribute **client_alloc=no** on the resource:

```
nim -o change -a client_alloc=no resourcename
```

To lift the restrictions on client-resource allocation, remove the **client_alloc** attribute by setting it to **yes** for the applicable object:

```
nim -o change -a client_alloc=yes master
nim -o change -a client_alloc=yes clientname
nim -o change -a client_alloc=yes resourcename
```

Preventing Machines from Adding Themselves as Clients

Machines may add themselves as clients in NIM environments by using the **niminit** command and specifying the hostname of a NIM master. In some environments, administrators may want total control over which machines are added as clients of their masters. To prevent clients from adding themselves to a NIM environment, an administrator can use the **client_reg** attribute.

Note: This task is not supported by Web-based System Manager.

From SMIT

The option to allow clients to add themselves to a NIM environment can be changed from the SMIT interface by typing the SMIT fast path:

```
smit nim_client_reg
```

From the Command Line

To prevent machines from adding themselves as clients in a NIM environment, set the attribute **client_reg=no** on the NIM master:

```
nim -o change -a client_reg=no master
```

To allow machines to add themselves as clients of a NIM master, remove the **client_reg** attribute by setting it to **yes** on the master:

```
nim -o change -a client_reg=yes master
```

Disabling Client CPU ID Validation

The CPU ID of a NIM client is stored in the NIM database so that the master can perform verification that NIM client commands are coming from the machines that were originally registered as clients. A NIM administrator would not want this CPU ID validation to be performed in the following situations:

- When the hardware of a client machine is changed, giving the client a new CPU ID.
- When a single client definition is used to install different machines, as on a preinstall assembly line.

From Web-based System Manager

To enable or disable NIM client CPU ID validation from the NIM application:

1. From the NIM menu, select **Advanced Configuration** —> **Control Client CPU ID Validation**.
2. Use the dialog to complete the task.

From SMIT

Manage the client CPU ID validation from the SMIT interface by typing the SMIT fast path:

```
smit nim_cpuid_validate
```

From the Command Line

Client CPU ID validation can be managed on the NIM master by using the **validate_cpuid** attribute.

To disable client CPU ID validation, set the attribute **validate_cpuid=no** on the NIM master:

```
nim -o change -a validate_cpuid=no master
```

To perform client CPU ID validation, remove the **validate_cpuid** attribute from the master by setting it to ″yes″:

```
nim -o change -a validate_cpuid=yes master
```

Attention: The value of the **validate_cpuid** attribute should not be changed while operations are being performed on NIM clients because it could potentially disrupt client communications for active machines.

Exporting NIM Resources Globally

When resources are allocated for use during NIM operations, they are NFS-exported to the client machines where the operations will be performed. If operations are performed simultaneously on many different clients, the **/etc/exports** and **/etc/xtab** files may become very large on the resource servers. This may cause size limits to be exceeded in the files, and it may also negatively affect NIM performance as the files are locked and modified for each resource allocation or deallocation.

In environments where administrators are not concerned about who has access to the NIM resources, they may set an option to globally export the resources and thereby eliminate the repeated updates to the **/etc/exports** and **/etc/xtab** files. The only resources that may not be globally exported are those that are used exclusively by diskless and dataless clients. The global export of a NIM resource will make it readable by any machine in the network, not just those in the NIM environment. The resource will be globally exported as long as it is allocated to any client. When the resource is deallocated from all clients, it is unexported.

From Web-based System Manager

To enable or disable global export of NIM resources from the NIM application:

1. From the NIM menu, select **Advanced Configuration** —> **Export NIM Resources Globally**.
2. Use the dialog to complete the task.

From SMIT

To manage global exporting of NIM resources from the SMIT interface, type the SMIT fast path:

```
smit nim_global_export
```

From the Command Line

Global exporting of NIM resources for use by clients can be managed with the **global_export** attribute.

To enable global exporting of NIM resources, set the attribute **global_export=yes** on the NIM master:

```
nim -o change -a global_export=yes master
```

To disable global exporting of NIM resources, remove the **global_export** attribute from the master by setting it to **no**:

```
nim -o change -a global_export=no master
```

Do not change the enablement and disablement of global exports when there are resources allocated to clients because this could lead to situations where resources are exported with incorrect permissions. All NIM operations should be completed and resources deallocated before any attempts are made to change the **global_export** value. If resources are currently allocated to clients, the **nim** command will fail to change the **global_export** value.

Creating Network Boot Images to Support Only the Defined Clients and Networks

When a SPOT resource is created, network boot images are created in the **/tftpboot** directory to support certain NIM operations.

NIM only creates network boot images to support clients and networks that are defined. If a new client is defined and there is no network boot image already created for it in the environment, then the boot image will not be created until either the SPOT is allocated to the client or a check operation is performed on the SPOT to rebuild the boot images.

When clients are removed from the NIM environment, boot images are not automatically removed. To remove boot images that are no longer necessary for a NIM environment, the list of required machine-network combinations in the environment must be rebuilt. The boot images must then be rebuilt for each SPOT.

From Web-based System Manager

To limit or enable boot image creation according to whether the interface is defined from the Web-based System Manager application:

1. From the NIM menu, select **Advanced Configuration** —> **Control Network Boot Image Creation**.
2. Use the dialog to complete the task.

From SMIT

To manage the creation of boot images from the SMIT interface, type the SMIT fast path:

```
smit nim_control_boot
```

From the Command Line

To rebuild the list of machine types and networks that must be supported by network boot images in the NIM environment, perform a **change** operation on the NIM master with the **if_discover=yes** attribute:

```
nim -o change -a if_discover=yes master
```

To rebuild network boot images from a SPOT, perform a **check** operation on the SPOT with the **force** option:

```
nim -Fo check spot_name
```

If an administrator prefers to have NIM always create all possible boot images from the SPOT resources, the **if_prebuild=yes** attribute can be specified on the master:

```
nim -o change -a if_prebuild=yes master
```

To return NIM to the behavior of creating only the boot images that are required for the environment, remove the **if_prebuild** attribute from the master by setting it to ″no″:

```
nim -o change -a if_prebuild=no master
```

Updating a Spot with New Device Support for a New Level of AIX

A NIM SPOT may be updated from one level of AIX to another using the **update_all** option of the NIM **cust** operation. This process will update all current SPOTs with the latest level of code on the installation media. However, this process will not automatically install new software packages or device drivers from the installation media.

Machines in the NIM environment that are being upgraded to a new level of AIX require that new applicable device support be updated for any existing NIM SPOTs intended to support network boot and installation. This must be done after the SPOT is updated to the new level of AIX.

The new device support can be installed in the SPOT using NIM's **cust** operation, specifying the desired device-specific filesets in an **installp_bundle** resource or by using the **filesets** attribute. Alternatively, a fileset name of **devices** can be specified as the value of the filesets attribute to install all devices on the installation media. For further details about the **cust** operation, see “cust” on page 291.

Tuning Client-Request Processing

For large installation environments, NIM can be scaled to support anywhere from 20 to 150 client requests simultaneously. NIM scaling is done by enabling the multithreaded option on the **nimesis** daemon. The multithreaded option provides better handling of the volume of client information change requests and client state changes. Without the use of the multithreaded option, the NIM master can become overloaded by activity on the NIM database and the number of active processes, resulting in simultaneous failures during the installation of a large number of client machines.

The multithreaded **nimesis** daemon will serialize and buffer NIM client requests to protect the NIM master from process overload, without causing significant performance degradation. The user must understand that many of the client information changes will not be reflected in the NIM database. The most recent information changes for any client, however, are eventually processed. Debugging of failed or hung clients will not be adversely affected.

The number of threads assigned to this daemon determines how many simultaneous NIM client requests can be handled in the NIM environment. Because most of the NIM client requests are processed rapidly, it is not necessary to have one thread for every client installing. The number of threads needed to support the activities in a NIM environment is dependent upon several items. The following should be considered when determining the number of threads:

- Number of clients that will be operated on at the same time
- Processing capacity of the NIM master machine
- What type of operations are planned

In general, one thread can support two to four clients that are installing BOS at the same time. For example, when installing 150 machines, 50 to 75 threads is sufficient. The number of threads is highly dependent on the processing power of the NIM master machine, and slower master machines may require more threads.

For smaller NIM environments, enabling the multithreaded daemon can monopolize system resources on the master that will not be used. For example, when installing 50 machines simultaneously, 20 to 25 threads or even the single-threaded daemon would suffice.

Note: The multithreaded option alone will not allow more machines to be installed simultaneously. The multithreaded option should be used in conjunction with global export of NIM resources, distribution of NIM resources throughout the NIM environment, and a network environment capable of handling a large volume of throughput.

From SMIT

To tune client-request processing from the SMIT interface, type the SMIT fast path:

```
smit nim_tune_nimesis
```

From Web-based System Manager

To tune client-request processing from the NIM application:

1. From the NIM menu, select **Advanced Configuration—>Tune Client Request Processing.**
2. Use the dialog to complete the task.

From the Command Line

The **max_nimesis_threads** attribute can be used to tune client-request processing. To enable the multithreaded **nimesis** daemon, set a value to the **max_nimesis_threads** attribute on the NIM master using the following command:

```
nim -o change -a max_nimesis_threads=value master
```

Note: The range for the *value* attribute above is 20 to 150.

To disable the multithreaded **nimesis** daemon, set a null value to the **max_nimesis_threads** attribute on the NIM master:

```
nim -o change -a max_nimesis_threads="" master
```

Chapter 24. Network Installation Management Concepts

This chapter discusses the concepts required to understand the operation of Network Installation Management (NIM). To use all the available features in NIM, you should understand various components of AIX installation. The details discussed in this chapter focus on command-line operations, but the information is applicable to the other NIM interfaces as well. Use this chapter as reference material to supplement the online help available in the other interfaces.

This chapter contains the following topics:

- "NIM Machines"
- "NIM Networks" on page 263
- "NIM Resources" on page 267
- "NIM Operations" on page 282
- "NIM Groups" on page 301

NIM Machines

The types of machines that can be managed in the NIM environment are *standalone*, *diskless*, and *dataless* clients. This section describes the differences between the machines, the attributes required to define the machines, and the operations that can be performed on them. The NIM environment is composed of two basic machine roles: *master* and *client*. The NIM master manages the installation of the rest of the machines in the NIM environment. The master is the only machine that can remotely run NIM commands on the clients. All other machines participating in the NIM environment are clients to the master, including machines that may also serve resources.

NIM Operations on Client Machines

There are unique operations to initialize the different client configurations. NIM checks that the operation is a valid operation for a specific client configuration. The following table shows the operations that can be performed on the different client configuration types.

NIM Operation	Machine Configuration		
	Standalone	Diskless	Dataless
bos_inst	x		
dkls_init		x	
dtls_init			x
diag	x	x	x
cust	x		
fix_query	x		
lppchk	x		
maint	x		
maint_boot	x		
reset	x	x	x
check	x	x	x
showlog	x	x	x
reboot	x	x	x

For more information about NIM operations, see "NIM Operations" on page 282.

© Copyright IBM Corp. 2002, 2004

Defining NIM Clients

Standalone, diskless, and dataless clients are defined in the NIM environment using the NIM **define** operation. The command line syntax is as follows:

```
nim -o define -t MachineType -a Attribute=Value ... MachineName
```

where the following attributes are required:

-t *MachineType*
: Specifies the type of machine being defined. Valid values are **standalone**, **diskless**, and **dataless**.

-a if=*Value ...*
: Stores network interface information for a NIM client, and requires a sequence number when specified. The value for this attribute consists of three required values and a fourth, optional value:

  Value 1
  : Specifies the name of the NIM network to which this interface connects. If the name of the NIM network is unknown, then the **find_net** keyword can be used to match the client's IP address to a defined NIM network. If the **find_net** keyword is used, but NIM does not find a matching network, the optional **net_definition** attribute should be used to define the network, as well.

  Value 2
  : Specifies the host name associated with this interface.

  Value 3
  : Specifies the network adapter hardware address of this interface. A value of **0** can be specified unless broadcasting is used for network boot of the client.

  Value 4
  : Specifies the logical device name of the network adapter used for this interface. If this value is not specified, NIM uses a default based on the type of network interface defined. This field is required when the client is defined on a heterogeneous network.

    This attribute requires a sequence number for NIM to distinguish between multiple network interfaces. Because machines can be multihomed, NIM allows more than one **if** attribute per machine.

The following attributes are optional:

-a ring_speed=*Value*
: Specifies the ring speed of the client's token-ring adapter. This value is required if the client's NIM network is token-ring. This attribute requires a sequence number for NIM to distinguish between ring speeds for multiple interfaces on the machine.

-a cable_type=*Value*
: Specifies the cable type of the client's ethernet adapter. This value is required if the client's NIM network is Ethernet. This attribute requires a sequence number for NIM to distinguish between cable types for multiple interfaces on the machine.

-a platform=*Value*
: Specifies the platform of the machine being defined. The default value is **chrp**. Run the **bootinfo -p** command on a running machine to determine its platform.

-a netboot_kernel=*Value*
: Specifies the kernel type of the client. Valid values are **up** for uniprocessor machines and **mp** for multiprocessor machines. The default value is **up**.

-a iplrom_emu=*Value*
: Specifies the device that contains the IPL ROM emulation software. IPL ROM emulation is required for machines that do not have bootp-enabled IPL ROM.

-a net_definition=*Value ...* Defines a NIM network to be associated with the client being defined. The value for this attribute consists of required values and optional values:

Value 1 = NetworkType **(required)**
Specified values are **tok**, **ent**, **fddi**, and **generic**.

Value 2 = SubnetMask **(required)**
Specifies the dotted decimal mask for the network.

Value 3 = ClientGateway **(optional)**
Specifies the IP address or host name of the default gateway used by the machine being defined to communicate with the NIM master.

Value 4 = MasterGateway **(optional)**
Specifies the IP address or host name of the default gateway used by the NIM master to communicate with clients on other subnets.

Value 5 = NetworkName **(optional)**
Specifies a name to be given to the NIM definition created for the network. (Otherwise, a unique default value is assigned.)

When specifying the **net_definition** attribute to create or change a machine definition, the **find_net** keyword must be specified as the first component of the **if** attribute for the machine. The **net_definition** attribute may also be specified when defining additional NIM interfaces (**if** attributes) for machine definitions.

-a cpuid=*Value* Specifies the CPU ID of the machine being defined. This attribute can be used for client verification during NIM operations. To display the CPU ID on a running machine, use the **uname -m** command. This field is optional and will be automatically set the first time a client communicates with the NIM master.

-a master_port=*Value* Specifies the port number used by the NIM master for socket communication with the clients. The default master port number is **1058**.

-a registration_port=*Value* Specifies the port number used by clients to register themselves with the NIM master. The default registration port number is **1059**.

-a group=*Value* Specifies a machine group to which the client should be added. The group will be defined if it does not exist.

-a comments=*Value* Provides comments about the client being defined.

-a verbose=*Value* Displays information for debugging. Use **verbose=5** to show maximum detail.

-a net_settings=*Value1 Value2* Specifies the speed and duplex settings to use for the client's ethernet adapter during a network install, and requires a sequence number when specified. When initiating an install and reboot of a client, NIM will set these parameters in the bootlist. This attribute is only valid for clients running AIX 5.2 or higher.

Value1= auto, 10, 100, or 1000
The default is value is *100*.

Value2 = auto, half, or full
The default value is *full*.

For example:

```
nim -o change -a net_settings1="10 half" jellyfish
```

Standalone Clients

Standalone NIM clients are clients with the capability of booting and running from local resources. Standalone clients mount all file systems from local disks and have a local boot image. Standalone clients are not dependent upon network servers for operation.

Network Booting a Standalone Client

Although an installed standalone client is capable of booting from the local disk, it may be necessary to perform a network boot of the client for certain NIM operations. Clients must boot over the network in

order for NIM to perform a BOS installation (**bos_inst**) of the client or to boot into maintenance mode (**maint_boot**) and diagnostics (**diag**). For instructions on booting a client over the network, see “Booting a Machine Over the Network” on page 343.

Managing Software on Standalone Clients

The AIX Base Operating System can be installed directly on standalone clients using the NIM **bos_inst** operation. Additional software and updates can be installed and managed on standalone clients using the NIM **cust** and **maint** operations. For more information about these and other operations, see “NIM Operations” on page 282.

Diskless and Dataless Clients

Diskless and dataless clients are machines that are not capable of booting and running without the assistance of servers on a network. As their names imply, diskless clients have no hard disk, and dataless clients have disks that are unable to hold all the data that may be required for operation. Diskless machines must mount paging space and all file systems from remote servers. Dataless machines can only use a local disk for paging space and the **/tmp** and **/home** file systems. Neither diskless nor dataless clients have a local boot image, and they must boot from servers on the network.

Defining a machine as diskless or dataless has the following advantages:

- **Cost savings**

  No hard disk is required for diskless clients. Only a small hard disk is needed for dataless clients.
- **Manage software configurations on machines**

  On diskless and dataless clients, the file system containing the BOS is mounted from a server. All client systems that mount the same file system for BOS run from identical software.
- **Manage storage of user data**

  User data for diskless and dataless clients are stored on remote servers. A system administrator can manage storage allocation and data backups for the client machines by managing the data on the server, rather than on each machine separately.

Required and Optional Resources for Diskless and Dataless Clients

The file systems that are mounted by the diskless and dataless client machines are treated as resources in the NIM environment. Like other resources, they exist on a server in the NIM environment, and they are NFS-exported to the clients that use them.

The following resources are managed by NIM to support diskless and dataless clients:

boot
: Defined as a network boot image for NIM clients. The **boot** resource is managed automatically by NIM and is never explicitly allocated or deallocated by users.

SPOT
: Defined as a directory structure that contains the AIX run-time files common to all machines. These files are referred to as the **usr** parts of the fileset. The **SPOT** resource is mounted as the **/usr** file system on diskless and dataless clients.

  Contains the **root** parts of filesets. The **root** part of a fileset is the set of files that may be used to configure the software for a particular machine. These **root** files are stored in special directories in the **SPOT**, and they are used to populate the root directories of diskless and dataless clients during initialization.

  The network boot images used to boot clients are constructed from software installed in the **SPOT**.

  A **SPOT** resource is required for both diskless and dataless clients.

root Defined as a parent directory for client ″**/**″ (**root**) directories. The client root directory in the **root** resource is mounted as the ″**/**″ (**root**) file system on the client.

When the resources for a client are initialized, the client **root** directory is populated with configuration files. These configuration files are copied from the **SPOT** resource that has been allocated to the same machine.

A **root** resource is required for both diskless and dataless clients.

dump Defined as a parent directory for client dump files. The client dump file in the **dump** resource is mounted as the dump device for the client.

A **dump** resource is required for both diskless and dataless clients.

paging Defined as a parent directory for client paging files. The client paging file in the **paging** resource is mounted as the paging device for the client.

A **paging** resource is required for diskless clients and optional for dataless clients.

home Defined as a parent directory for client **/home** directories. The client directory in the **home** resource is mounted as the **/home** file system on the client.

A **home** resource is optional for both diskless and dataless clients.

shared_home Defined as a **/home** directory shared by clients. All clients that use a **shared_home** resource will mount the same directory as the **/home** file system.

A **shared_home** resource is optional for both diskless and dataless clients.

tmp Defined as a parent directory for client **/tmp** directories. The client directory in the **tmp** resource is mounted as the **/tmp** file system on the client.

A **tmp** resource is optional for both diskless and dataless clients.

resolv_conf Contains nameserver IP addresses and a network domain name.

Unlike the other resources used by diskless/dataless clients, the **resolv_conf** resource does not remain mounted by the client. Instead, it is copied to the **/etc/resolv.conf** file in the client's root directory.

A **resolv_conf** resource is optional for both diskless and dataless clients.

Initialization of Diskless and Dataless Clients

Diskless and dataless clients are not installed in the same way as standalone machines. Instead, they are initialized. Initialization of diskless and dataless clients involves several phases of operation:

Resource Allocation The resources required to support a diskless/dataless client must be allocated to the client before or during the initialization operation.

If the resource is a parent directory of client directories, the allocation will create an empty subdirectory for the client. The client subdirectory is then NFS-exported to the client. The client subdirectories are not populated until the initialization is actually performed.

Client Initialization

The **dkls_init** and **dtls_init** operations are used in NIM to initialize the resources for client use.

Among the operations performed during client initialization are the following:

- The boot image is made available to the client for performing a network boot.
- The root files, which are used for machine-specific customization, are copied into the client's subdirectory in the **root** resource. The files that are copied into the client root directories come from the **SPOT** resource that has been allocated to the client.
- The **/tftpboot/***Client***.info** file is created on the boot server (which is the **SPOT** server). This file contains information that will be needed by the client during the start-up configuration processing to successfully configure as a diskless or dataless client.

  The following are some of the variables defined in the *Client***.info** file:

```
export NIM_CONFIGURATION=diskless
export RC_CONFIG=rc.dd_boot
export ROOT=Host:Client_Root_Directory
export DUMP=Host:Client_Dump_Directory
export SPOT=Host:SPOT_Location
```

  The paging location is set in the client's root directory in the **/etc/swapspaces** file.

Network Boot of the Client

The client machine is booted over the network using standard **bootp** procedures for the machine type. The client obtains the boot image and begins running a mini-kernel in a file system in RAM.

The client tftp's the *Client***.info** file from the **/tftpboot** directory on the **SPOT** server. The information in the *Client***.info** file is used to properly configure the client as a diskless or dataless machine.

The remote file systems are mounted from the resource servers.

If the client is a dataless client, and no **paging**, **tmp**, **home**, or **shared_home** resource is allocated, then the client will create the missing file system on the local hard disk.

Managing Software on Diskless and Dataless Clients

The **/usr** and **root** file systems of diskless and dataless clients are resources that have been mounted from a server. Therefore, in order to install or uninstall software on a diskless or dataless client, the processing must actually occur on the resources that the clients use.

The **SPOT** contains the directory structure for an installed **/usr** file system. It also contains subdirectories for the ″root″ parts of installed filesets. Because the **SPOT** contains both **usr** and **root** files, software maintenance must be performed on the **SPOT** in order to update the software that is running on the clients. Such actions must be performed using the NIM **cust** and **maint** operations. For more information about the **cust** and **maint** operations, see "NIM Operations" on page 282.

If the **SPOT** is currently allocated for client use, NIM will prevent software customization operations from being performed on it. This is to safeguard the **SPOT** from changes that may adversely affect running client machines. However, this restriction can be overridden by specifying the **force** option when performing the operation.

When NIM is used to install software in a **SPOT**, the following operations are performed to manage the software for diskless and dataless clients:

1. The **/usr** files are installed in the **SPOT**. These files are automatically seen by all the clients that mount the **SPOT** as their **/usr** file systems.
2. The root files are installed in special subdirectories in the **SPOT**.
3. After all the filesets have been installed in the **SPOT**, the **root** files are copied to the **root** directories of any diskless or dataless clients that have been initialized with the **SPOT**.

When NIM is used to uninstall software in a **SPOT**, the following operations are performed to manage the software for diskless and dataless clients:

1. The **/usr** files are removed from the **SPOT**. This also automatically ″removes″ the files from the client systems.
2. The **root** files of the software are removed from the client **root** directories.

NIM also provides a **sync_roots** operation to perform consistency verification and correction to ensure the client **root** directories match the **root** parts stored in the **SPOT**.

NIM Networks

In order to perform certain NIM operations, the NIM master must be able to supply information necessary to configure client network interfaces. The NIM master must also be able to verify that client machines can access all the resources required to support operations. To avoid the overhead of repeatedly specifying network information for each individual client, NIM networks are used to represent the networks in a NIM environment. When NIM clients are defined, the associated network for the client must be specified. During NIM operations, the NIM master is able to use information from the client's network definition when necessary.

When the NIM master is configured, the network associated with the master is automatically defined in the NIM environment. It is necessary only to define additional NIM networks if clients reside on other local area networks or subnets. The procedures described in this guide and reference are designed to automatically define NIM networks, if necessary, when clients are added. However, this section is included to describe NIM networks in detail in case manual definition of networks and routes is required.

Supported NIM Network Types

The supported network types are as follows:

- Ethernet
- Standard Ethernet
- IEEE 802.3 Ethernet
- Token-Ring
- FDDI
- ATM
- Generic

Network boot support is provided for Ethernet, Token-Ring, and FDDI. Unlike other network adapters, ATM adapters cannot be used to boot a machine. Therefore, installing a machine over an ATM network requires special processing. See “Installing to Clients on ATM Networks” on page 201. The Generic network type is used to represent all other network types where network boot support is not available. For clients on Generic networks, NIM operations that require a network boot, such as **bos_inst** and **diag**, are not supported. However, nonbooting operations, such as **cust** and **maint**, are allowed. Diskless and dataless clients cannot be associated with Generic networks, because they inherently rely on network boot capability.

Defining NIM Networks

Networks are defined in the NIM environment using the NIM **define** operation. The command line syntax is as follows:

```
nim -o define -t NetworkType -a Attribute=Value ... MachineName
```

where the following attributes are required:

-a net_addr=*Value* Specifies the IP address of the network being defined. If the network address is not known, see "Determining a Network's IP Address."

-a snm=*Value* Specifies the subnet mask for the network.

-t *NetworkType* Specifies the type of network being defined. Valid values are **atm**, **tok**, **ent**, **fddi**, and **generic**.

The following attributes are optional:

-a comments=*Value* Provides comments about this network.

-a ieee_ent=*Value* Specifies IEEE 802.3 Ethernet configuration. This is only valid for networks that are defined with the **ent** type or those that have an **other_net_type** attribute set to **ent**.

-a other_net_type=*Value* Specifies another network type that applies to this logical network. Each NIM network is used to represent one logical network that exists in the NIM environment. When the network is defined, the type of network interface used in the network must be supplied. Usually, a network is composed of only one type. However, a bridge can be used to connect different network types together to form one logical network. In that situation, NIM needs to know what the other network interface types are, and this attribute is used to specify that information. For more information on how to use the **other_net_type** attribute, see "Defining a Heterogeneous Network" on page 265.

-a routing=*Value* ... Stores NIM routing information for a network. This attribute requires a sequence number when specified. When a new NIM route is specified, the **routing** attribute consists of three values:

Value 1
Specifies the NIM name of the destination network for this route.

Value 2
Specifies the host name of the gateway to use in order to communicate with the destination network.

Value 3
Specifies the host name of the gateway used by the destination network to get back to this network.

This attribute can be used to add a default route or static route. To add a default route, specify **default** for *Value 1*. Then, specify the default gateway for the network in *Value 2*. Leave *Value 3* blank.

For more information on adding and changing routes, see "NIM Routes" on page 265, "Establishing a Default NIM Route Between Networks" on page 235, and "Establishing a Static NIM Route Between Networks" on page 236.

-a verbose=*Value* Displays information for debugging. Use **verbose=5** to show maximum detail.

It is also possible to define NIM networks automatically when client machines are defined. To do this, use the **find_net** and **net_definition** attributes when defining the client. For more information, see "NIM Machines" on page 257.

Determining a Network's IP Address

NIM determines a network's IP address by performing a bitwise ″AND″ on the binary representations of the network's subnet mask and the address of any machine's IP address on the same network. For example:

```
    subnet mask = 255.255.254.0
 client address = 129.35.58.207

In binary:
```

```
    subnet mask = 11111111.11111111.11111110.00000000
 client address = 10000001.00100011.00111010.11001111
network address = 10000001.00100011.00111010.00000000

In decimal:

network address = 129.35.58.0
```

NIM Routes

Routing information is used internally by NIM to ensure that a client on one network can communicate with a server on another network. It defines the gateway to use to go from one network to the other network.

NIM provides the ability to define default or static routes. Default NIM routes provide the following advantages over static routes:

- They more closely model the network configuration of common network environments.
- They permit resources that are distributed throughout a NIM environment to be more easily accessed by any client in the NIM environment.

To determine the gateway used by machines on a given network, run **netstat -rn** on a running machine on the network to see if a default gateway is listed. You can also issue **traceroute** *Host_Name* from a running machine on the network in question, where *Host_Name* is the name of the master's primary network interface if determining the gateway for a client, or the name of a target client if determining the gateway used by the master. The first gateway listed is the gateway used by machines on the specified network.

Note that NIM routes are not required if the only networks defined in a NIM environment are associated with interfaces (**if** attributes) defined on the NIM master and if all resources will be defined on the master. If resources are served by machines other than the master to clients that do not reside on the same network as the server, NIM routes are required between those networks even if all networks are attached to interfaces belonging to the master. In this case, the master must act as a gateway (with IP-forwarding switched on), and the host name of the interface on the master should be used as a gateway.

Networks with default routes may be created automatically when NIM machines are being defined.

Communications between networks go through several gateways. However, it is important to remember that when defining NIM routes for networks, the only gateways of interest are the first ones used by the networks to reach their destinations. Intermediate gateways between the originating and destination networks are irrelevant for NIM routing purposes.

Defining a Heterogeneous Network

This section describes the NIM feature that enables NIM to model networks consisting of different data-link protocol segments. These kinds of networks use bridges to connect two segments that have different data link protocols. A network consisting of a Token-Ring and an Ethernet segment can be connected to form a single logical network, as shown in the following figure.

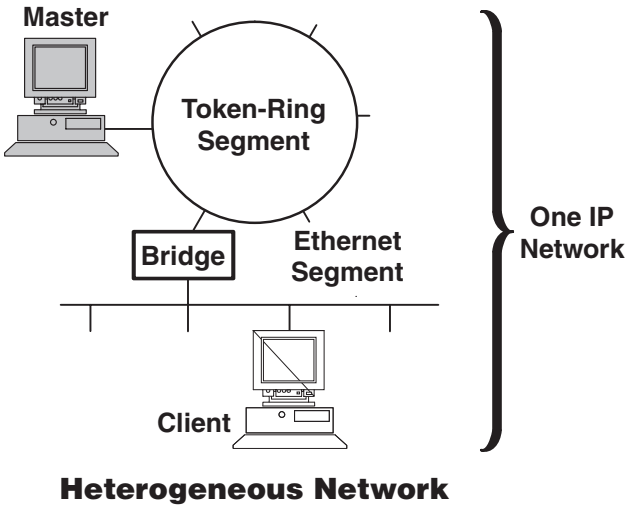


Figure 1. Heterogeneous Network. This illustration shows a single IP network in which the master server uses its token-ring connection and a bridge to communicate with its client on an Ethernet segment.

Because a single NIM network object is used to represent one network, the **other_net_type** attribute is reserved for a different type of interface that can exist in a network. The **other_net_type** attribute can be added to the definition of a network object. When present in a network definition, the **other_net_type** attribute tells NIM that this logical network uses a bridge to connect the other network type to the network type that was specified when the object was defined.

When you define a machine object to be connected to a network object, NIM checks to see if the network has any **other_net_type** attributes. If so, NIM requires that the fourth field, which is usually optional, in the **if** attribute, be specified. This field specifies the logical name of the client's network adapter. The following example defines a network object that has a bridge joining a Token-Ring and an Ethernet segment:

```
nim -o define -t tok -a net_addr=129.35.129.0 \
    -a snm=255.255.240.0 -a other_net_type1=ent b905net

lsnim -l b905net

class           =  network
type            =  tok
net_addr        =  129.35.128.0
snm             =  255.255.240.0
other_net_type1 =  ent
Nstate          =  ready for use
prev_state      =  information is missing from this object's def>
```

The **other_net_type** attribute requires a sequence number because a network could be composed of all three types of interfaces linked by bridges.

When you define a client's interface that is physically connected to an Ethernet segment joined with a Token-Ring network using a bridge (with master being on the Token-Ring side), you must supply the fourth field:

```
nim -o define -t standalone -a if1='find_net mymac 08005ac9430c \
ent' -a cable_type1=bnc mymac
```

Adding Another Network Type to a NIM Network

From Web-based System Manager

1. In the NIM Network container, double-click on a network. The General page of the properties notebook displays.

2. Use the General page to add a network type to the network.

From SMIT

1. To add another network type, enter the **smit nim_chnet** fast path.
2. Select the network to change.
3. Specify the additional network type to be supported.

From the Command Line

To define a NIM network, enter:

```
nim -o change -a other_net_type<SequenceNumber=NetworkType NetworkName
```

For example, to change a Token-Ring network called `network1` to also support Ethernet and FDDI, enter:

```
nim -o change -a other_net_type1=ent -a other_net_type2=fddi  network1
```

NIM Resources

A large number of resources (files and directories) are needed to support NIM software installation and maintenance operations. Resources can be added through the Web-based System Manager **New Resources** wizard.

To obtain detailed information about any resource, enter the following from the NIM master:

```
lsnim -Pa ResourceType
```

The Web-based System Manager and SMIT interfaces are designed to hide much of the detail required for the command line interface. Therefore, these sections only document the resource task procedures for the command line. The following information applies to the other interfaces as well, but discussion of those interfaces is deferred to the online contextual help available for those applications. The following sections describe detailed information about each NIM resource:

- “adapter_def Resource” on page 268
- “boot Resource” on page 268
- “bosinst_data Resource” on page 268
- “dump Resource” on page 269
- “exclude_files Resource” on page 269
- “fb_script Resource” on page 270
- “fix_bundle Resource” on page 271
- “home Resource” on page 271
- “image_data Resource” on page 272
- “installp_bundle Resource” on page 272
- “lpp_source Resource” on page 273
- “mksysb Resource” on page 274
- “nim_script Resource” on page 275
- “paging Resource” on page 275
- “resolv_conf Resource” on page 276
- “root Resource” on page 277
- “script Resource” on page 277
- “shared_home Resource” on page 278
- “SPOT (Shared Product Object Tree) Resource” on page 278
- “tmp Resource” on page 281

adapter_def Resource

The **adapter_def** resource represents a directory that contains secondary adapter configuration files that are used during **bos_inst** and **cust** operations. The **adapter_def** resource directory is populated with secondary-adapter configuration files by the **nimadapters** command.

Defining an adapter_def Resource

The command line syntax for defining an **adapter_def** resource is:

```
nim -o define -t adapter_def -a Attribute=Value ... adapter_defName
```

The following attributes are required for the **adapter_def** resource:

-a location=*Value*	Specifies the full path name of the **adapter_def** resource directory.
-a server=*Value*	Specifies the name of the machine where the **adapter_def** resource directory resides. Only the master can serve an **adapter_def** resource.

The following attributes are optional for the **adapter_def** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of `5`.

Secondary adapter support is available for AIX 5.2 or later. Before you enable a secondary adapter, you must verify the AIX version the client is on. Secondary adapter support is not available for clients at AIX 5.1 or earlier. The secondary adapters will fail to configure because NIM is unable to find the **/usr/lpp/bos.sysmgt/nim/methods/c_cfgadptrs** client method. The following example shows the outcome if you attempt to enable this support on your NIM master.

```
nim -o cust -a adapter_def=adapter_def1 rspc10
trigger.austin.xyz.com. 0042-001 nim: processing error encountered on "master":
0042-001 m_cust: processing error encountered on "rspc10":
0042-175 c_script: An unexpected result was returned by the
"trigger.austin.xyz.com:/export/nim/scripts/rspc10.script" command:
/tmp/_nim_dir_4714/script[10]: /usr/lpp/bos.sysmgt/nim/methods/c_cfgadptrs:  not found.
```

boot Resource

The **boot** resource is an internally managed NIM resource used to indicate that a boot image has been allocated to a client. The **boot** resource is automatically allocated to clients to support NIM operations requiring a network boot. The **boot** resource will be automatically deallocated when the operation completes.

bosinst_data Resource

A **bosinst_data** resource represents a file that contains information for the BOS installation program. Normally, the BOS installation program looks for this information in the **/bosinst.data** file in the BOS installation image. If this file does not exist or if it does not contain all the information that the BOS installation program requires, the program prompts for information by using a console that is local to the target. Information must then be specified manually for the BOS installation to proceed. With a **bosinst_data** resource, the data can be specified in a NIM resource prior to the installation to prevent the need for prompting at the console.

A sample **bosinst.data** file (*SPOT_Offset* **/usr/lpp/bosinst/bosinst.template**) is located on the **SPOT** resource server. Also, see “Performing a Nonprompted BOS Installation” on page 200 for a sample **bosinst_data** file.

For instructions on how to create and use a **bosinst_data** file, see “Performing a Nonprompted BOS Installation” on page 200.

Defining a bosinst_data Resource

The command line syntax for defining a **bosinst_data** resource is:

```
nim -o define -t bosinst_data -a Attribute=Value ... bosinst_dataName
```

The following attributes are required for the **bosinst_data** resource:

-a location=*Value*	Specifies the full path name of the **bosinst_data** resource file.
-a server=*Value*	Specifies the name of the machine where the **bosinst_data** resource file resides.

The following attributes are optional for the **bosinst_data** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **bosinst_data** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

dump Resource

A **dump** resource represents a directory in which client dump directories are maintained. When this type of resource is allocated to a client, NIM creates a subdirectory for the client’s exclusive use. This allocated subdirectory is initialized by the **dkls_init** or **dtls_init** operation, which creates an empty file in this subdirectory. After initialization, the client uses this file to store any **dump** images it creates.

Note: If you subsequently deallocate this resource, NIM removes the **dump** file and the subdirectory that NIM created for the client’s use.

Defining a dump Resource

The command line syntax for defining a **dump** resource is:

```
nim -o define -t dump -a Attribute=Value ... DumpName
```

The following attributes are required for the **dump** resource:

-a location=*Value*	Specifies the full path name of the parent directory for the client **dump** files.
-a server=*Value*	Specifies the name of the machine where the directory for the **dump** resource will be created.

The following attributes are optional for the **dump** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

exclude_files Resource

An **exclude_files** resource represents a file that contains a list of files and directories that should be excluded when creating a system backup image. This resource may be used when a **mksysb** resource is being created from a running NIM client.

Defining an exclude_files Resource

The command line syntax for defining an **exclude_files** resource is:

```
nim -o define -t exclude_files -a Attribute=Value ... exclude_filesName
```

The following attributes are required for the **exclude_files** resource:

-a location=*Value*	Specifies the full path name of the file containing the list of files and directories to exclude from the **mksysb**.
-a server=*Value*	Specifies the name of the machine where the file for the **exclude_files** resource resides.

The following attributes are optional for the **exclude_files** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **exclude_files** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

fb_script Resource

An **fb_script** resource represents a file that is used to configure devices when a NIM client is booting for the first time after the BOS installation process is completed. During BOS installation, certain customization operations (such as device configuration) cannot be performed because they require certain daemons to be running. However, at this point in the BOS installation process, daemons are not available. As a result, certain devices may not be configured during system reboot, and have to be manually configured after the system has booted.

You can use an **fb_script** resource to provide device-configuration information. The BOS installation process adds the content of the **fb_script** resource to the **/etc/firstboot** file, which is run the first time that a client is booted. The **/etc/firstboot** file then performs the device configuration.

Defining an fb_script Resource

The command line syntax for defining an **fb_script** resource is as follows:

```
nim -o define -t fb_script -a server=server_name \
 -a location=fbscript_file_location fbscript_object_name
```

After the **fb_script** resource is defined, you can allocate the resource and initiate a BOS installation operation using the **fb_script** resource, as follows:

```
nim -o bos_inst -a fb_script=fbscript_object_name client_name
```

The following attributes are required for the **fb_script** resource:

-a location=*Value*	Specifies the full path name of the file being defined as the **fb_script** resource.
-a server=*Value*	Specifies the name of the machine where the file for the **fb_script** resource resides.

The following attributes are optional for the **fb_script** resource:

-a comments=*Value*	Describes the resource.
-a source=*Value*	Specifies an existing **fb_script** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

fix_bundle Resource

A **fix_bundle** resource represents a file containing **fix** keywords to be used by the **instfix** command, which is called by the NIM **cust** and **fix_query** operations. NIM mounts the **fix_bundle** resource on the client so it can be used by the local **instfix** command. NIM automatically unmounts the resource when the operation has completed.

A fix can include either a single fileset update or multiple fileset updates that are related in some way; fixes are identified by unique keywords. When a fix is identified with an Authorized Program Analysis Report (APAR) number, it includes all the fileset updates that are necessary to fix the reported software problem identified by that number.

Defining a fix_bundle Resource

The command line syntax for defining a **fix_bundle** resource is:

```
nim -o define -t fix_bundle -a Attribute=Value ... fix_bundleName
```

The following attributes are required for the **fix_bundle** resource:

-a location=*Value*	Specifies the full path name of the file containing the list of fixes to manage.
-a server=*Value*	Specifies the name of the machine where the **fix_bundle** resource file resides.

The following attributes are optional for the **fix_bundle** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **fix_bundle** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

home Resource

A **home** resource represents a directory in which client **/home** directories are maintained. When this type of resource is allocated to a client, NIM creates a subdirectory for the client's exclusive use. This allocated subdirectory is subsequently initialized when you perform the **dkls_init** or **dtls_init** operation. After initialization, any time the client performs a network boot, the client NFS mounts this subdirectory over **/home** to gain access to the **home** directory that has been set up for its use. This subdirectory remains mounted over **/home** on the client as long as the client is running.

Note: Whenever this resource is deallocated, NIM removes the subdirectory that was created for the client's use. Therefore, back up any files you want to save in the client's subdirectory before you deallocate a resource of this type.

Defining a home Resource

The command line syntax for defining a **home** resource is:

```
nim -o define -t home -a Attribute=Value ... HomeName
```

The following attributes are required for the **home** resource:

-a location=*Value*	Specifies the full path name of the parent directory for the client **/home** directories.
-a server=*Value*	Specifies the name of the machine where the directory for the **home** resource will be created.

The following attributes are optional for the **home** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

image_data Resource

An **image_data** resource represents a file that contains information for the BOS installation program. This information describes how physical disks and file systems should be configured in the root volume group during installation. Normally, the BOS installation program determines default values that should be used, or uses an **image.data** file from a **mksysb** being restored. Use a customized **image_data** resource only in special cases.

A sample **image.data** file (*SPOT_Offset***/usr/lpp/bosinst/image.template**) is located on the **SPOT** resource server. For more information about the **image.data** file, see the *AIX 5L Version 5.2 Files Reference*.

Defining an image_data Resource

The command line syntax for defining an **image_data** resource is:

```
nim -o define -t image_data -a Attribute=Value ... image_dataName
```

The following attributes are required for the **image.data** resource:

-a location=*Value*	Specifies the full path name of the **image_data** resource file.
-a server=*Value*	Specifies the name of the machine where the **image_data** resource file resides.

The following attributes are optional for the **image.data** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **image_data** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

installp_bundle Resource

An **installp_bundle** resource represents a file that contains the names of filesets that should be managed by NIM. During an installation or maintenance operation, NIM mounts the **installp_bundle** file on the client machine so it can be used by the local **installp** command. NIM automatically unmounts the resource from the client when the operation has completed.

Defining an installp_bundle Resource

The command line syntax for defining an **installp_bundle** resource is:

```
nim -o define -t installp_bundle -a Attribute=Value ... installp_bundleName
```

The following attributes are required for the **installp_bundle** resource:

-a location=*Value*	Specifies the full path name of the file containing the list of software to manage.
-a server=*Value*	Specifies the name of the machine where the **installp_bundle** resource file resides.

The following attributes are optional for the **installp_bundle** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **installp_bundle** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

lpp_source Resource

An **lpp_source** resource represents a directory in which software installation images are stored. If the **lpp_source** contains the minimum set of support images required to install a machine, it is given the **simages** attribute and can be used for BOS installation (**bos_inst**) operations. If an **lpp_source** does not contain enough software to be an **simages lpp_source**, then it can only be used in NIM **cust** operations to install software on running machines and **SPOTs**.

NIM uses an **lpp_source** for an installation operation by first mounting the **lpp_source** on the client machine. The **installp** commands are then started on the client using the mounted **lpp_source** as the source for installation images. When the installation operation has completed, NIM automatically unmounts the resource.

In addition to providing images to install machines, **lpp_source** resources can also be used to create and update **SPOT** resources.

Note: When copying device images to a directory that you plan to define as an **lpp_source**, be sure to copy all the device images for a given type of device. For example:

```
cp /cdfs/usr/sys/inst.images/devices.pci.*  lpp_source_directory
```

You can define an **lpp_source** in several ways:

- If a directory containing installation images already exists, it can be directly defined as an **lpp_source** resource.
- If a directory should be created and populated by NIM with the default set of support images for a BOS install, use the **source** attribute when defining the resource. This attribute specifies the name of the device that contains the installation images. NIM copies the software images from this device into the location specified for the **lpp_source**. The images copied will include those from the **simages** list, all available device support, and some additional software that is typically installed as well (for example, X11).
- If an **lpp_source** should be created from a source device using a list of software other than the default set of images, specify the **packages** attribute when defining the **lpp_source**. Use the **packages** attribute to list the alternative set of software images to copy.

The size of an **lpp_source** may vary greatly with the amount of software it includes. A minimum **lpp_source** with just enough software to qualify for the **simages** attribute may be under 100 MB, but a default **lpp_source** created from a CD-ROM may be over 350 MB. It is recommended that a separate file system be created to contain an **lpp_source** so the space can be more easily managed. By default, NIM automatically expands a file system as needed when creating an **lpp_source** and copying images from a source device.

Beginning in AIX 5.2, the **simages** message displays only if the user is creating an **lpp_source** resource with the default installation packages. The **simages** message will display if the **simages** attribute could not be set for the **lpp_source**. If a user creates an **lpp_source** and specifies a list of packages, the **simages** message is not printed. The **simages** attribute is set correctly, whether or not a **simages** message is printed or not.

If a user attempts to do an **rte** BOS installation with an **lpp_source** that does not have the **simages** attribute, the user receives an error. The error message instructs the user to run **nim -o check** on the **lpp_source** to determine the missing packages needed for an **rte** BOS installation. Whenever a user runs **nim -o check** on an **lpp_source** after it has been created, the **simages** message is printed if the **lpp_source** does not contain all the images needed for a **rte** BOS installation.

Defining an lpp_source Resource

The command line syntax for defining an **lpp_source** resource is:

```
nim -o define -t lpp_source -a Attribute=Value ... lpp_sourceName
```

The following attributes are required for the **lpp_source** resource:

-a location=*Value*	Specifies the directory that will contain the installation images.
-a server=*Value*	Specifies the name of the machine where the **lpp_source** is to be created.

The following attributes are optional for the **lpp_source** resource:

-a comments=*Value*	Describes the **lpp_source**.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a packages=*Value*	Specifies a list of filesets to copy into the **lpp_source** if the default list of images is not desired.
-a source=*Value*	Identifies the source device for copying installation images when defining the **lpp_source**. This attribute is not required if the location of the **lpp_source** already contains installation images.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

If a migration installation will be performed on NIM client machines, the **lpp_source** used in the operation must contain all the required software to migrate the machine.

If the directory specified in the **location** attribute does not exist, NIM will create the directory. NIM will also remove the directory and its contents if the **lpp_source** is later removed.

mksysb Resource

A **mksysb** resource represents a file that is a system backup image created using the **mksysb** command. This type of resource can be used as the source for the installation of a client. The **mksysb** image must reside on the hard disk of a machine in the NIM environment in order to be defined as a resource. It cannot be located on a tape or other external media.

A **mksysb** resource can be defined from an image that already exists on the hard disk of the NIM master or any NIM client. If such an image does not exist, it can be created when the resource is defined. To create the image when the resource is defined, specify the name of the NIM client that will be the **source** for the backup, and set the **mk_image** attribute to **yes** in the command to define the **mksysb** resource. Use an **exclude_files** resource to list any files and directories that should not be included in the backup image.

Defining a mksysb Resource

The command line syntax for defining a **mksysb** resource is:

```
nim -o define -t mksysb -a Attribute=Value ... mksysbName
```

The following attributes are required for the **mksysb** resource:

-a location=*Value*	Specifies the full path name of the **mksysb** image.

-a server=*Value*	Specifies the name of the machine where the **mksysb** image resides or is to be created.

The following attributes are optional for the **mksysb** resource:

-a comments=*Value*	Describes the **mksysb**.
-a exclude_files=*Value*	Specifies an **exclude_files** resource to use to exclude files and directories from the system backup.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a mk_image=*Value*	Specifies the flag to use to create a **mksysb** image from a machine in the NIM environment.
-a mksysb_flags=*Value*	Specifies the flags to use to tell the command how to create the backup.
-a size_preview=*Value*	Specifies the flag to verify that space is available before creating a **mksysb** image.
-a source=*Value*	Specifies the name of the machine to be backed up in the **mksysb** image.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies the name of the machine to be backed up in the **mksysb** image if the **mk_image** attribute is specified. If the **mk_image** attribute is not specified, this value specifies an existing **mksysb** resource to be replicated when defining a new **mksysb** resource. The file pointed to by the source resource will be copied to the new location.

nim_script Resource

The **nim_script** resource is an internally managed NIM resource used to indicate that a script should be run by NIM as part of a NIM operation. The **nim_script** resource is automatically allocated to support some NIM operations, and it is automatically deallocated when the operations complete.

Depending on the operation, NIM will use the following rules to determine which NIM server to place the **nim_script resource** on:

- For a **bos_inst** operation, the **nim_script** resource will be placed on the **SPOT** server.
- For **cust** operation with an **lpp_source**, the **nim_script** resource will be placed on the **lpp_source** server.
- For a **cust** operation without an **lpp_source**, the **nim_script** resource will be placed on the script server.
- Otherwise, the **nim_script** resource will be placed on the NIM master.

paging Resource

A **paging** resource represents a directory where client paging files are maintained. When this type of resource is allocated to a client, NIM creates a subdirectory for the client's exclusive use. This allocated subdirectory is initialized by the **dkls_init** or **dtls_init** operation, which creates a file in this subdirectory that the client configures as a paging device when it performs a network boot. By default, 32 MB are reserved for this file. A different value can be specified using the **size** flag when the **dkls_init** or **dtls_init** operation is performed.

After this resource has been initialized for a client, it is configured as a paging device by the client each time the client performs a network boot.

Note: If you subsequently deallocate this resource, NIM removes the **paging** file and the subdirectory it created for the client's use.

Defining a paging Resource

The command line syntax for defining a **paging** resource is:

```
nim -o define -t paging -a Attribute=Value ... PagingName
```

The following attributes are required for the **paging** resource:

-a location=*Value*	Specifies the full path name of the parent directory for the client **paging** files.
-a server=*Value*	Specifies the name of the machine where the directory for the **paging** resource will be created.

The following attributes are optional for the **paging** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

resolv_conf Resource

A **resolv_conf** resource represents a file containing valid **/etc/resolv.conf** entries that define Domain Name Protocol name-server information for local resolver routines. A **resolv_conf** resource can be allocated to a standalone machine as part of a **bos_inst** operation or to a diskless or dataless machine as part of a **dkls_init** or **dtls_init** operation. Upon successful installation and reboot, the machine will be configured to use the domain name services defined by the resource.

The following are sample entries in a **resolv_conf** resource file:

```
nameserver      129.35.143.253
domain          test.ibm.com
```

Defining a resolv_conf Resource

The command line syntax for defining a **resolv_conf** resource is:

```
nim -o define -t resolv_conf -a Attribute=Value ... resolv_confName
```

The following attributes are required for the **resolv_conf** resource:

-a location=*Value*	Specifies the full path name of the file containing the information for domain name server (DNS) name resolution.
-a server=*Value*	Specifies the name of the machine where the **resolv_conf** resource file resides.

The following attributes are optional for the **resolv_conf** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **resolv_conf** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

root Resource

A **root** resource represents a directory in which client **root** directories are maintained. When this type of resource is allocated to a diskless or a dataless client, NIM creates a subdirectory for the client's exclusive use. This allocated subdirectory is subsequently initialized when you perform the **dkls_init** or **dtls_init** operation.

After initialization, anytime the client performs a network boot, the client NFS mounts this subdirectory over "**/**" to gain access to the **root** directory that has been set up for its use. This subdirectory remains mounted over **/** on the client as long as the client is running.

Note: Whenever this resource is deallocated, NIM removes the subdirectory that was created for the client's use. Therefore, any files you want to save in the client's subdirectory should be backed up before you deallocate a resource of this type.

Defining a root Resource

The command line syntax for defining a **root** resource is:

```
nim -o define -t root -a Attribute=Value ... RootName
```

The following attributes are required for the **root** resource:

-a location=*Value*	Specifies the full path name of the directory under which client **root** directories will be created.
-a server=*Value*	Specifies the name of the machine where the directory for the **root** resource will be created.

The following attributes are optional for the **root** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

script Resource

A **script** resource represents a file that is a user-defined shell script. After it is defined, this type of resource can be used to perform processing on a client as part of a NIM **cust** or **bos_inst** operation.

The **script** resources are always run by NIM after software installation is performed in **cust** or **bos_inst** operations. This allows the scripts to perform configuration processing on the client after all the software is installed. Multiple **script** resources can be allocated for client use, but the order in which the scripts will be run is not predictable.

Note: The **script** resources must not point to files that reside in the **/export/nim/scripts** directory. This directory is used for the **nim_script** resource that is managed by NIM. NFS restrictions prevent defining multiple resources in the same location.

Defining a script Resource

The command line syntax for defining a **script** resource is:

```
nim -o define -t script -a Attribute=Value ... ScriptName
```

The following attributes are required for the **script** resource:

-a location=*Value*	Specifies the full path name of the **script** resource file.
-a server=*Value*	Specifies the name of the machine where the **script** resource file resides.

The following attributes are optional for the **script** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.
-a source=*Value*	Specifies an existing **script** resource to be replicated when defining a new resource. The file pointed to by the source resource will be copied to the new location.

shared_home Resource

A **shared_home** resource represents a directory that can be used as a common **/home** directory by one or more clients. When this type of resource is allocated to a client, and when the **dkls_init** or **dtls_init** operation is performed, NIM configures the client's configuration to use this common directory. After initialization, anytime the client performs a network boot, the client NFS mounts this common directory over its **/home** directory. This common directory remains mounted as long as the client is running.

Note: Whenever this resource is deallocated, NIM only changes the client's configuration so that this directory is no longer used by the client. NIM does not remove the common directory.

Defining a shared_home Resource

The command line syntax for defining a **shared_home** resource is:

```
nim -o define -t shared_home -a Attribute=Value ... shared_homeName
```

The following attributes are required for the **shared_home** resource:

-a location=*Value*	Specifies the full path name of the directory to be used as a common **/home** directory among clients.
-a server=*Value*	Specifies the name of the machine where the directory for the **shared_home** resource will be created.

The following attributes are optional for the **shared_home** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

SPOT (Shared Product Object Tree) Resource

The **SPOT (Shared Product Object Tree)** is a fundamental resource in the NIM environment. It is required to install or initialize all machine configuration types. A **SPOT** provides a **/usr** file system for diskless and dataless clients, as well as the network boot support for all clients.

Everything that a machine requires in a **/usr** file system, such as the AIX kernel, executable commands, libraries, and applications are included in the **SPOT**. Machine-unique information or user data is usually stored in the other file systems. A **SPOT** can be located on any standalone machine within the NIM environment, including the master. The **SPOT** is created, controlled, and maintained from the master, even though the **SPOT** can be located on another system.

You can create a **SPOT** by converting the **/usr** file system (**/usr SPOT**, or you can locate the **SPOT** elsewhere within the file system (non-**/usr SPOT**) on the server.

The **/usr SPOT** inherits all the optional software that is already installed on the server. All the clients using the **/usr SPOT** have access to the optional software installed on the server. The non-**/usr SPOT** can be used to manage a different group of optional software than those that are installed and licensed for the server.

Creating a **SPOT** by converting the **/usr** file system has the advantage of being fast and using much less disk space. However, this method does not give you the flexibility to choose which software packages will be included in the **SPOT**, because all the packages and filesets installed in the **/usr** file system of the machine serving the **SPOT** will be included in the **SPOT**. The second method, creating a non-**/usr SPOT**, uses more disk space, but it is more flexible. Initially, only the minimum set of software packages required to support NIM clients is installed in the **SPOT**, but additional packages and filesets can be installed. Also, it is possible to have multiple **SPOT**s, all with different additional packages and filesets installed, serving different clients.

Note: Do not create a non-**/usr SPOT** in a subdirectory of the **/usr** file system.

A **SPOT** varies in size from 100 MB up to, and sometimes in excess of, 300 MB depending on the software that is installed. Because all device support is installed in the **SPOT** and the number of device filesets typically increases, the size is not easily predictable from one release of AIX to another.

SPOTs are used to support all NIM operations that require a machine to boot over the network. These operations are as follows:

- **bos_inst**
- **maint_boot**
- **diag**
- **dkls_init**
- **dtls_init**

When a **SPOT** is created, network boot images are constructed in the **/tftpboot** directory of the **SPOT** server, using code from the newly created **SPOT**. When a client performs a network boot, it uses **tftp** to obtain a boot image from the server. After the boot image is loaded into memory at the client, the **SPOT** is mounted in the client's RAM file system to provide all additional software support required to complete the operation.

Each boot image created is up to 4 MB in size. Before creating a **SPOT**, ensure there is sufficient space in the root (**/**) file system, or create a separate file system for **/tftpboot** to manage the space required for the network boot images.

A single network boot image can be accessed by multiple clients; therefore, the network boot image cannot contain any client-specific configuration information. The platform type is specified when the machine object is defined, while the network type is determined from the primary interface definition. Two files are created in the **/tftpboot** directory on the **SPOT** server for each client to be network-booted: *ClientHostName* and *ClientHostName***.info**. The *ClientHostName* file is a link to the correct network boot image, while the *ClientHostName***.info** file contains the client configuration information.

When the **SPOT** is defined (and created), the following occurs:

- The BOS image is retrieved from archive or, for **/usr** conversion, just the **root** directory is retrieved from archive (**/usr/lpp/bos/inst_root**).
- The device support required to support NIM operations is installed.
- Network boot images are created in the **/tftpboot** directory.

To list the software installed in a **SPOT**, enter the following command:

```
nim -o lslpp SPOTName
```

If you want to change your **/usr SPOT** back to a normal **/usr** file system, you must remove the **SPOT** from the NIM database.

For information about software installation and maintenance tasks you can perform on a **SPOT**, see “Customizing NIM Clients and SPOT Resources” on page 203.

Network Boot Images for AIX 4.2 SPOTs

Each network boot image supports a single network, platform, and kernel type. The network boot image files are named *SPOTName.Platform.Kernel.Network*. The network types are Token-Ring, Ethernet, and FDDI. The platform types are:

rs6k (AIX 5.1 and earlier)	Used for POWER family/POWER family2/P2SC/POWER-based MCA bus-based machines.
rspc (AIX 5.1 and earlier)	Used for POWER-based Reference Platform (PREP) Architecture-based machines.
chrp	Used for POWER-based Common Hardware Reference Platform (CHRP) Architecture-based machines.

The **rs6ksmp** platform for AIX 4.2 (and later) **SPOTs** is represented by the boot image with a platform type of **rs6k** and a kernel type of **mp**.

The kernel types are:

up	Used for single processor machines.
mp	Used for multiple processor machines.

Both **up** and **mp** boot images are created for each platform and network type. The network boot images located in **/tftpboot** for a **SPOT** named `42spot` look similar to the following:

```
42spot.rs6k.mp.ent

42spot.rs6k.mp.fddi

42spot.rs6k.mp.tok

42spot.rs6k.up.ent

42spot.rs6k.up.fddi

42spot.rs6k.up.tok

42spot.rspc.mp.ent

42spot.rspc.mp.tok

42spot.rspc.up.ent

42spot.rspc.up.tok
```

The amount of space used in the **/tftpboot** directory for boot images may become very large. An AIX 4.2.1 (or later) **SPOT** that supports network boot for all possible combinations of platforms, kernel types, and network adapters may require as much as 60 MB in **/tftpboot**. If the same server serves multiple **SPOTs**, the space required in **/tftpboot** will be even more because each **SPOT** creates its own set of boot images.

Network Boot Images for AIX 4.3 or later SPOTs

In AIX 4.3 or later, NIM creates by default only the boot images required to support the machines and network types that are defined in the environment. This situation should significantly reduce the amount of disk space used and the time required to create boot images from SPOT resources.

Defining a SPOT Resource

The command line syntax for defining a **SPOT** resource is:

```
nim -o define -t spot -a Attribute=Value ... SPOTName
```

The following attributes are required for the **SPOT** resource:

-a location=*Value*	Specifies the parent directory under which the **SPOT** is to be created.
-a server=*Value*	Specifies the name of the machine where the **SPOT** is to be created.
-a source=*Value*	Identifies the source device for installation images to create and install the **SPOT**.

The following attributes are optional for the **SPOT** resource:

-a auto_expand=*Value*	Expands the file system as needed when installing the **SPOT**. The default value is **yes**.
-a comments=*Value*	Describes the **SPOT**.
-a debug=*Value*	Builds debug-enabled network boot images. The default value is **no**.
-a installp_flags=*Value*	Specifies the flags that describe how **installp** should install software into the **SPOT**. The default value is **agQX**.
-a show_progress=*Value*	Shows **installp** output as **SPOT** is installed. The default value is **yes**.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

Note: The creation of a **SPOT**, by default, produces a large amount of output. Be sure to scan the output to look for nonfatal errors and warnings that may not be evident from a successful return code.

tmp Resource

A **tmp** resource represents a directory where client **/tmp** files are maintained. When this type of resource is allocated to a client, NIM creates a subdirectory for the client's exclusive use. This allocated subdirectory is subsequently initialized when you perform the **dkls_init** or **dtls_init** operation. After initialization, anytime the client performs a network boot, the client NFS mounts this subdirectory over **/tmp** to gain access to the **/tmp** directory that has been set up for its use. This subdirectory remains mounted over **/tmp** on the client as long as the client is running.

Note: Whenever this resource is deallocated, NIM removes the subdirectory that was created for the client's use. Therefore, back up any files you want to save in the client's subdirectory before you deallocate a resource of this type.

Defining a tmp Resource

The command line syntax for defining a **tmp** resource is:

```
nim -o define -t tmp -a Attribute=Value ... TmpName
```

The following attributes are required for the **tmp** resource:

-a location=*Value*	Specifies the full path name of the directory where client**/tmp** directories will be created.
-a server=*Value*	Specifies the name of the machine where the directory for the **tmp** resource will be created.

The following attributes are optional for the **tmp** resource:

-a comments=*Value*	Describes the resource.
-a group=*Value*	Specifies the name of a resource group to which this resource should be added.
-a verbose=*Value*	Displays information for debugging. To show maximum detail, specify a value of **5**.

Distributed NIM Resources

Usually, a NIM administrator will use the NIM master as the server for all resources. This strategy keeps all resources together on one machine. However, there are several reasons to distribute resources onto client machines:

- If the NIM environment requires several large resources to be defined, it may not be possible to put them all on the same server because of disk space limitations. Creating resources on different machines allows the burden of disk consumption to be distributed over several machines.
- Serving resources from different machines helps avoid bottlenecks when performing NIM operations on large numbers of clients. Bottlenecks can occur on server machines or on network gateways, so it may be beneficial to distribute resources across servers running in different subnets.
- Multiple resources of the same type can be created on different machines to increase the availability of resources when servers are taken offline for scheduled maintenance.
- Some **SPOT** resources at certain levels cannot be served by some machines at certain levels. Specifically, **SPOT** creation is not supported when the level of AIX installed in the **SPOT** is higher than the level of AIX running on the server. When you are creating **SPOTs** at multiple levels, it may be necessary to distribute the **SPOTs** on different servers.

Distributing resources on different machines in the NIM environment is simply a matter of specifying the correct server information when the resource is defined. After the resources are created, they are used no differently than resources defined on the master.

NIM Operations

A large number of operations can be performed to manage a NIM environment and perform software installation and maintenance. The Web-based System Manager and SMIT interfaces are designed to hide much of the detail required for the command line interface. Therefore, this section only documents the operations for the command line. All of this information applies to the other interfaces as well, but discussion of those interfaces is deferred to the online contextual help available for those applications.

Most NIM operations are performed by running the **nim** command with various attributes for each possible operation. The command line syntax is as follows:

```
nim -o OperationName -a Attribute=Value ... TargetName|TargetNames
```

The NIM operations you can perform are:

- “allocate” on page 283
- “alt_disk_install” on page 283
- “bos_inst” on page 285
- “change” on page 290
- “check” on page 290
- “cust” on page 291
- “deallocate” on page 292
- “define” on page 292
- “diag” on page 292
- “dkls_init” on page 293

- “dtls_init” on page 293
- “fix_query” on page 294
- “lppchk” on page 295
- “lppmgr” on page 295
- “maint” on page 295
- “maint_boot” on page 296
- “reboot” on page 296
- “remove” on page 297
- “reset” on page 297
- “select” on page 297
- “showlog” on page 298
- “showres” on page 298
- “sync_roots” on page 299
- “unconfig” on page 300
- “update” on page 300

allocate

The **allocate** operation is used to make resources available to NIM clients for subsequent operations.

The command line syntax for the **allocate** operation is as follows:

```
nim -o allocate -a ResourceType=ResourceName ... TargetName|TargetNames
```

The target of an **allocate** operation may be a NIM client or group of NIM clients.

The following attribute can be specified for the **allocate** operation:

-a *ResourceType=ResourceName* (required)	Specifies the resource to allocate to the client, for example, `lpp_source=42_images`.

When a resource is allocated to a client, an entry is added to the **/etc/exports** file on the resource server to NFS export the resource to the client. The allocation count for the resource is also incremented. When the allocation count is greater than 0, the resource cannot be modified. During NIM operations, a client mounts and uses the resources that have been allocated to it.

alt_disk_install

The **alt_disk_install** operation (available in AIX 4.3 or later) can be used to install a **mksysb** image on a client system’s alternate disk or disks, or it can be used to clone a client running **rootvg** to an alternate disk.

The command line syntax for the **alt_disk_install mksysb** operation is as follows:

```
nim -o alt_disk_install -a source=mksysb -a mksysb=mksysb_resource \
 -a disk=target_disk(s) -a attribute=Value.... TargetName|TargetNames
```

The command line syntax for the **alt_disk_install rootvg** clone operation is as follows:

```
nim -o alt_disk_install -a source=rootvg -a disk=target_disk(s) \
 -a attribute=Value.... TargetName|TargetNames
```

The target of an **alt_disk_install** operation can be a standalone NIM client or a group of standalone NIM clients. The clients must also have the **bos.alt_disk_install.rte** fileset installed.

To display the alternate disk installation status while the installation is progressing, enter the following command on the master:

```
lsnim -a info -a Cstate ClientName
```

OR

```
lsnim -l ClientName
```

The following are required attributes for **alt_disk_install mksysb** operation:

-a source=mksysb	Specifies the type of **alt_disk_install** to perform.
-a disk=*target_disk(s)*	Specifies the disks on the client system that the **mksysb** image will be restored. This disk or these disks must not currently contain any volume group definition. The **lspv** command should show these disks as belonging to volume group **None**. If you are specifying more than one disk, the disk names must be enclosed in a set of single quotes; for example, `'hdisk2 hdisk3'`.
-a mksysb=*mksysb_resource*	Specifies the **mksysb** resource to use.

The following are required attributes for the **alt_disk_install rootvg** clone operation:

-a source=rootvg	Specifies the type of **alt_disk_install** to perform.
-a disk=*target_disk(s)*	Specifies the disks on the client system that the **mksysb** image will be restored. This disk or these disks must not currently contain any volume group definition. The **lspv** command shows these disks as belonging to volume group **None**. If you are specifying more than one disk, the disk names must be enclosed in a set of single quotes; for example, `'hdisk2 hdisk3'`.

The following are optional attributes that can be specified for both **alt_disk_install mksysb** and the **alt_disk_install rootvg** clone operation:

-a concurrent=Value	Specifies the maximum number of machines from the selected group that should be installing at any given time. This attribute is only valid when the target of the operation is a machine group. If specified, NIM will monitor the progress of all machines in the group and attempt to keep no more or less than the number specified installing until all machines in the group are installed.
-a set_bootlist=*Value*	Specifies whether to set the bootlist to point to the new **rootvg** when the install is complete. *Value* can be `yes` or `no`, where `yes` is the default value. The next time the system is rebooted, it will boot from the newly installed alternate disk if *Value* is set to `yes`.
-a boot_client=*Value*	Specifies whether to reboot the client when the **alt_disk_install** operation is completed. *Value* can be `yes` or `no`, where `no` is the default value. This attribute would normally be set only if the **set_bootlist** attribute was also set to `yes`.
-a debug=*Value*	Specifies whether to print debug **(set -x)** output from the **alt_disk_install** script. *Value* can be `yes` or `no`, where `no` is the default value. This output does not go to the screen, but is saved to the NIM log, **/var/adm/ras/nim.alt_disk_install**, on the client system. This file can be checked after the **alt_disk_install** has completed.
-a image_data=*Value*	Specifies the image_data resource to use when creating the new alternate **rootvg** and its logical volumes and file systems. The new volume group created must be large enough to restore the **mksysb** image or a copy of the running **rootvg**. An **exclude_files** attribute can also be used with an **alt_disk_install rootvg** clone to specify files or directories that should not be backed up.
-a resolv_conf=*Value*	Specifies the **resolv_conf** resource to use for configuring the domain and name resolution on the client system when the system is rebooted. This is the **/etc/resolv_conf** file that will be copied into the alternate disk's file system. This may be useful if the **mksysb** image you are using has a different **/etc/resolv_conf** file than the one you want the client to retain.

-a script=*Value* Specifies the script resource to call at the end of the **alt_disk_install** operation. This script is called on the running system before the **/alt_inst** file systems are unmounted, so files can be copied from the running system to the **/alt_inst** file systems before the reboot. This is the only opportunity to copy or modify files in the alternate file system because the logical volume names will be changed to match those of **rootvg**, and they will not be accessible until the system is rebooted with the new alternate **rootvg**.

-a time_limit=Value, Specifies the maximum number of hours that should elapse before ceasing to initiate installation of additional members of the selected group of machines. This value can only be specified when limiting the number of concurrent operations on a group.

-a verbose=*Value* Specifies whether to show files as they are being backed up for a **rootvg** clone, or to show files as they are being restored for a **mksysb** install. *Value* can be `yes` or `no`, where `no` is the default value. The output goes to the **alt_disk_install** log on the client, **/var/adm/ras/alt_disk_inst.log**.

The following are optional attributes that can be specified only for the **alt_disk_install rootvg** clone operation:

-a exclude_files=*Value* Specifies an **exclude_files** resource to use to exclude files and directories from the **rootvg**. Files and directories specified in this file will not be copied to the new cloned **rootvg**.

-a filesets=*Value* Specifies the list of filesets to install into the alternate **rootvg** after the clone of the **rootvg** is complete.

-a fixes=*Value* Specifies the APARs to install into the alternate **rootvg** after the clone of the running **rootvg**. The fixes are in the format `"IX123456"` or `"update_all"`.

-a fix_bundle=*Value* Specifies the **fix_bundle** resource that lists the APARs to install into the alternate **rootvg** after the clone of the running **rootvg**.

-a installp_bundle=*Value* Specifies an **installp_bundle** resource that lists filesets to install into the alternate **rootvg** after the clone of the running **rootvg**.

-a installp_flags=*Value* Tells **installp** how to apply the filesets, **installp_bundle**, fixes, or **fix_bundles** attributes. The default value is `installp_flags=-acgX`.

bos_inst

The **bos_inst** operation is used to install the AIX BOS on standalone clients.

The command line syntax for the **bos_inst** operation is as follows:

```
nim -o bos_inst -a source=Value -a Attribute=Value ... TargetName|TargetNames
```

The target of a **bos_inst** operation can be a standalone NIM client or a group of standalone NIM clients.

The following NIM resources are required attributes that can be specified for the **bos_inst** operation to install and customize a machine:

-a lpp_source=*Value*

Identifies the **lpp_source** resource to be used. The **lpp_source** resource is only required for an **rte** installation. The **lpp_source** resource specified must have the **simages** attribute set. However, if you are performing a **bos_inst** operation using a **mksysb** resource and an **lpp_source** resource, then the **simages** attribute is optional. The **lpp_source** provides software for machine customization. It also provides the BOS image for installation if the **source** attribute is **rte**.

-a source=*Value*

Identifies the source for BOS run-time files. Valid values are:

rte Installs from a BOS image in the **lpp_source**.

mksysb Installs the machine from a **mksysb** image.

-a spot=*Value*
Identifies the **SPOT** resource to be used. The **SPOT** provides support for network boot and operations in the boot environment.

The following NIM resources are optional attributes that can be specified for the **bos_inst** operation:

-a accept_licenses=*Value*
Specifies whether license agreements should be accepted during BOS installation. Before the installation process can complete, this attribute must be set to **yes**. The default value is **accept_licenses=no**. If the **bosinst_data** resource resides on the NIM master, the **ACCEPT_LICENSES** field in the **bosinst_data** resource can also be set to **yes**. You can also set the **NIM_LICENSE_ACCEPT** global environment variable to yes on the NIM master.

-a adapter_def=*Value*
Specifies the directory containing secondary adapter definition files. The **nimadapters** command parses a secondary-adapters stanza file to build the files required to add NIM secondary adapter definitions to the NIM environment as part of the **adapter_def** resource. The **nimadapters** command does not configure secondary adapters. The actual configuration takes place during a **nim -o bos_inst** or **nim -o cust** operation that references the **adapter_def** resource.

-a async=*Value*
Specifies whether NIM should perform operations on group members asynchronously and not wait for the operation to complete on one member before beginning the operation on the next. The default value is **async=yes**.

-a auto_expand=*Value*
Indicates whether to expand file systems when setting up a client for a **force_push** installation. The default value is **auto_expand=yes**.

-a boot_client=*Value*
Indicates whether NIM should attempt to reboot the client immediately for BOS installation. The **boot_client** attribute is the converse of the **no_client_boot** attribute. The default value is **boot_client=yes**, indicating that NIM should attempt to reboot the client.

-a bosinst_data=*Value*
Specifies the **bosinst_data** resource to use for nonprompted installation.

-a concurrent=Value
Specifies the maximum number of machines from the selected group that should be installing at any given time. This attribute is only valid when the target of the operation is a machine group. If specified, NIM will monitor the progress of all machines in the group and attempt to keep no more or less than the number specified installing until all machines in the group are installed.

-a filesets=*Value*
Specifies a list of filesets to install on the target after BOS installation.

-a force_push=*Value*
Indicates whether or not a **force_push** installation should occur. A **force_push** should be used for installing machines that are running, but are not configured with the NIM client fileset. See the “force_push Attribute” on page 289 for more information.

-a group=*Value*
Specifies the name of a resource group to use for installation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes. If a resource group is specified, and it contains a **SPOT** and **lpp_source**, the **spot** and **lpp_source** attributes are no longer required.

-a image_data=*Value*
Specifies an **image_data** resource to describe how physical and logical data is organized on the client.

-a installp_bundle=*Value*
Specifies an **installp_bundle** resource that lists filesets to install on the target after BOS installation.

-a installp_flags=*Value*
Tells **installp** how to apply the filesets specified by the **filesets** or **installp_bundle** attributes. The default value is **installp_flags=-agQX**.

-a mksysb=*Value*
Provides the run-time files for BOS and other filesets if the **source** attribute is **mksysb**. The level of BOS run-time files in the **mksysb** must be equal to the level of the **SPOT** resource used for the installation.

If the level of the **SPOT** resource is greater than the level of the **mksysb** resource, then an **lpp_source** resource must be used and match the level of the **SPOT** resource. When this situation occurs, an update operation is performed by default.

-a no_client_boot=*Value*
Indicates whether the target should remain in the NIM environment after installation completes. The default value is **no**, indicating that the target system should remain in the NIM environment.

-a physical_loc=*Value*
Specifies the physical location code or AIX location code of the installation disk to the BOS installation process. This attribute allows you to specify the location code for the installation disk or disks on the command line, and allows you to have a *generic* **bosinst.data** file that does not contain location code information.

To determine a disk's physical location code, type the following:

```
lsdev -Cc disk -l hdisk0 -F "name physloc"
```

-a preserve_res=*Value*
Indicates whether resources in non-rootvg file systems should be preserved on the client system being installed. The default value is **preserve_res=no**.

-a resolv_conf=*Value*
Specifies the **resolv_conf** resource to use for configuring domain and name resolution on a client.

-a script=*Value*
Specifies the **script** resource to be run on the target system after all software has been installed.

-a set_bootlist=*Value*
Indicates whether NIM should set the bootlist of the client so that the client boots over the network on the next reboot. Usually, **set_bootlist** would be **yes** if the client is not going to be rebooted immediately for installation (**no_client_boot=yes** or **boot_client=no**). The default value is **set_bootlist=no**.

-a show_progress=*Value*
Indicates whether status should be displayed for each group member when the installation target is a group of machines. The default value is **show_progress=yes**.

-a time_limit=*Value*
Specifies the maximum number of hours that should elapse before ceasing to initiate installation of additional members of the selected group of machines. This value can only be specified when limiting the number of concurrent operations on a group.

-a verbose=*Value*
Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

When a **bos_inst** operation is performed on a client, the following occurs:

On the **SPOT** server:

1. A link is created in **/tftpboot** to a boot image matching the platform type, kernel type, and network adapter of the client.
2. The **/etc/bootptab** file is updated with client information to allow the boot image to be used.
3. A *ClientName***.info** file is created in **/tftpboot** to provide client-specific installation and configuration information in the boot environment.
4. The **/etc/tftpaccess.ctl** file is modified, if necessary, to allow access to the **/tftpboot** directory.

On the target system:

1. The bootlist is modified so the network adapter is the default boot device for normal mode boot, unless **no_client_boot=yes**, **set_bootlist=no**, and **force_push=no** are specified.
2. The client is rebooted to begin the installation, unless **no_client_boot=yes**, **boot_client=no**, and **force_push=no** are specified.

When the client boots over the network adapter, it obtains the boot image from the **SPOT** server. The boot image configures devices and sets up the machine for the BOS installation. The *Client*.info file is transferred to the client machine; and based on its contents, the network adapter is configured, routes are added, and NIM resources are mounted in the boot environment. Processing control is then passed to the BOS installation program.

NIM BOS Installation Details

The BOS installation program requires access to an image that contains the BOS run-time files. This image is used by the BOS installation program to populate the target's **/usr** file system. In the NIM environment, this image can come from one of the following resources:

- A BOS run-time image that is part of the **lpp_source** resource that has been allocated to the target
- A **SPOT** resource that has been allocated to the target
- A **mksysb** image that has been allocated to the target

A **spot** and **lpp_source** are always required to support the **bos_inst rte** operation. A **bos_inst mksysb** operation only requires a **spot** resource be used.

To indicate which BOS image to use, specify the **source** attribute when performing the **bos_inst** operation. The **source** attribute may have one of the following values:

rte When an **rte** value (the default) is used for the **source** attribute, NIM directs the BOS installation program to use the BOS run-time image that is in the **lpp_source** directory. This image contains only the BOS run-time files; it does not contain any optional software packages. Selecting an **rte** source may increase the BOS installation time, because the BOS installation program installs the appropriate device support after populating the target's **/usr** file system to make the target viable. The installation time may also be increased due to additional **installp** activity during the NIM customization phase.

Note: The **rte** source must be used when performing BOS migration installation.

mksysb
Using **mksysb** as the source results in a target machine that has the same configuration as the machine from which the **mksysb** image was created. This may save installation and configuration time. The **mksysb** images could be very large, and the installation will fail if the target does not have enough disk space to accommodate the image.

After the installation is initiated from the master, the NIM master attempts to contact the target and execute a script that will force the system to reboot. The target system issues a BOOTP request to the server after it has shut down. The **bos_inst** operation is considered complete even if the target does not immediately issue a BOOTP request. The target must issue a BOOTP request to load a network boot image from the server to start the installation.

If the master is unable to contact the target system for any reason (for example, the system is turned off, it is not a running NIM client, or there is a network problem), a message is displayed and user intervention is then required at the target to issue the BOOTP request using the IPL ROM. See “Booting a Machine Over the Network” on page 343.

By default (`no_nim_client=no`), NIM also includes the customization required for the target to remain a NIM client after the install. This customization includes the installation and configuration of the **bos.sysmgt.nim.client** fileset and its requisite filesets, **bos.net.tcp.client** and **bos.net.nfs.client**, so that the NIM master can communicate with and control the client after installation. The **installp_flags** are passed to the **installp** command for installing the software on the standalone client. The **filesets** attribute can be used to install a list of additional filesets or software packages from the allocated **lpp_source**.

To display BOS installation status information while the installation is progressing, enter the following command on the master:

```
lsnim -a info -a Cstate ClientName
```

OR

```
lsnim -l ClientName
```

Errors in the allocation of a **nim_script** or **boot** resource type are fatal errors because the network BOS installation process cannot proceed without these resources. On the other hand, any error encountered during the attempt to cause the target to issue a BOOTP request is a nonfatal error to NIM because, at that point, NIM has successfully initialized the environment to perform a network installation. As soon as the target has successfully loaded its allocated network boot image, the BOS installation process begins.

force_push Attribute

When assigned a value of **yes**, the **force_push** attribute tells NIM that the target of the **bos_inst** operation does not necessarily have the **bos.sysmgt.nim.client** fileset installed and configured. NIM will attempt to NFS mount or copy the minimal client support to the target system to perform an unattended installation or migration of the base operating system. If client support is copied to the target machine, NIM will automatically expand the necessary file systems on the target unless the **auto_expand** attribute to **bos_inst** is set to **no**.

The **force_push** attribute requires that the client grant root **rsh** permissions to the master and that the key on the client be in the normal position. The **force_push** attribute also requires that a **bosinst_data** file be allocated to the target machine to indicate that a no-prompt installation should occur. The **force_push** attribute is set to **yes** by setting the Force Unattended Installation Enablement? option to **yes** when using SMIT to perform the **bos_inst** operation.

boot_client Attribute

When assigned a value of **no**, the **boot_client** attribute is used to instruct NIM not to attempt to initiate the BOS installation on the target machine after setting up the installation with the **bos_inst** operation. This allows a BOS installation to be set up while deferring the actual installation until the client is rebooted at a later time. Also, if the client is not a running machine, this attribute will avoid waiting for the reboot attempt to time-out or fail. If the installation of the client system is going to be initiated later from the server, the normal mode boot device list on the client must be set so that a network boot is attempted when the client is rebooted. No attempt is made to modify the boot list when **boot_client** is set to **no** unless the **force_push** or **set_bootlist** attributes are specified and set to a value of **yes**. The **boot_client** attribute is set to **no** by setting Initiate Boot Operation on Client to **no** when using SMIT to perform the **bos_inst** operation.

set_bootlist Attribute

The **set_bootlist** attribute can be used with the **boot_client** attribute to modify the boot device list on the client for normal mode so a network boot is attempted when the client is rebooted. It is not necessary to specify the **set_bootlist** attribute if the **force_push** attribute is set to **yes** or if **boot_client** is unspecified or set to **yes**. In both instances, the boot list will be modified as the default. The only valid values for **set_bootlist** are **yes** and **no**. The **set_bootlist** attribute is set to **yes** by setting Set Boot List if Boot not Initiated on Client? when using SMIT to perform the **bos_inst** operation.

preserve_res Attribute

The **preserve_res** attribute can be used to preserve the NIM database definitions for resources residing on a NIM client that is being reinstalled. When set to **yes**, any resources that reside in file systems which are being preserved by the BOS installation process will also be preserved.

change

The **change** operation is used to modify attributes of NIM objects. The command line syntax is as follows:

```
nim -F -o change -a Attribute=Value ... TargetName|TargetNames
```

-F (optional) Tells NIM to **force** the operation if the target is currently in use.

The target of a **change** operation can be any network, machine, resource, or group in the NIM environment. Not all attributes can be modified on targets. Usually, the attributes are changed automatically as parts of other operations, so there is little need for you to use the **change** operation explicitly.

check

The **check** operation is used to verify the usability of a machine or resource in the NIM environment.

The command line syntax for the **check** operation is as follows:

```
nim -F -o check -a debug=Value TargetName|TargetNames
```

The target of a **check** operation can be any NIM client, a group of NIM clients, a **SPOT** resource, or an **lpp_source** resource.

The flags and attributes that can be specified for the **check** operation are as follows:

-F (optional) Tells NIM to ″force″ the operation if the target is currently in use. If the **-F** flag is specified when the target is a **SPOT** resource, it will force the **SPOT**'s network boot images to be rebuilt. The **-F** flag is typically not required when performing the **check** operation on client machines.

-a debug=*Value* (optional) Builds a **SPOT**'s network boot images in debug mode if **debug=yes** is specified. This attribute is only valid if the target is a **SPOT** resource. The default value is **debug=no**. For more information on the **debug** attribute, refer to “Producing Debug Output from a Network Boot Image” on page 338.

When applied to NIM clients, the **check** operation updates the machine state (**Mstate**) of the client. A ping test is performed to check whether the client is reachable. After the **check** operation is performed, the client's **Mstate** is set to either **running** or **not running**.

When applied to **SPOT** resources, the **check** operation performs root synchronization for diskless and dataless clients and rebuilds the **SPOT**'s network boot images, if necessary.

When applied to **lpp_source** resources, the **check** operation rebuilds the table of contents (**.toc**) file in the **lpp_source** directory. It also determines whether all filesets are included in the resources to qualify the **lpp_source** for the **simages** attribute.

cust

The **cust** operation is used to install software filesets and updates on standalone clients and **SPOT** resources.

See "Customizing NIM Clients and SPOT Resources" on page 203 for information on performing a software customization of standalone NIM clients.

The command line syntax for the **cust** operation is as follows:

```
nim -o cust -a Attribute=Value ... TargetName|TargetNames
```

The target of a **cust** operation can be a standalone NIM client, a group of standalone NIM clients, or a **SPOT** resource.

The following are required attributes that can be specified for the **cust** operation:

-a filesets=*Value* Specifies a list of filesets to install on the target. This attribute is required unless an **installp_bundle** is used for the operation.

-a installp_bundle=*Value* Specifies an **installp_bundle** resource that lists filesets to install on the target. This attribute is required unless the **filesets** attribute is specified.

-a lpp_source=*Value* Identifies the **lpp_source** resource that will provide the installation images for the **cust** operation.

The following are optional attributes that can be specified for the **cust** operation:

-a accept_licenses=*Value* Specifies whether software licenses should be automatically accepted during installation. If `accept_licenses=yes`, the `-Y` flag is passed on the **installp** command and licenses are automatically accepted. If `accept_licenses=no`, license processing is controlled by the **installp_flags** attribute. The default value is **accept_licenses=no**.

-a async=*Value* Specifies whether NIM should perform operations on group members asynchronously and not wait for the operation to complete on one member before beginning the operation on the next. The default value is **async=yes**.

-a concurrent=Value Specifies the maximum number of machines from the selected group that should be installing at any given time. This attribute is only valid when the target of the operation is a machine group. If specified, NIM will monitor the progress of all machines in the group and attempt to keep no more or less than the number specified installing until all machines in the group are installed.

-a fix_bundle=*Value* Contains a list of fixes to install on the target. Fixes should be listed in the **fix_bundle** resource by APAR number with one number per line.

-a fixes=*Value* Identifies a list of fixes to install on the target. Fixes should be listed by APAR number. For example, `fixes="IX12345 IX54321"`.

-a group=*Value* Specifies the name of a resource group to use for the installation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes. If a resource group is specified, and it contains an **lpp_source**, the **lpp_source** attribute is no longer required.

-a installp_flags=*Value* Identifies the flags that tell **installp** how to apply the filesets specified by the **filesets**, **installp_bundle**, **fixes**, and **fix_bundle** attributes. The default value is **installp_flags=agQX**.

-a resolv_conf=*Value* Specifies a **resolv_conf** resource for configuring domain and name resolution on a client.

-a script=*Value* Specifies a **script** resource to be run on the target system after all software has been installed.

-a show_progress=*Value* Indicates whether status should be displayed while software is installed. The default value is **show_progress=yes**.

-a time_limit=Value, Specifies the maximum number of hours that should elapse before ceasing to initiate installation of additional members of the selected group of machines. This value can only be specified when limiting the number of concurrent operations on a group.

deallocate

The **deallocate** operation is used to unlock and unexport resources when they are no longer needed by NIM clients. It is generally unnecessary to perform explicit deallocations after NIM operations, because upon successful completion, operations will automatically deallocate resources from the clients.

The command line syntax for the **deallocate** operation is as follows:

```
nim -o deallocate -a ResourceType=ResourceName ... -a subclass=all TargetName|TargetNames
```

The target of a **deallocate** operation may be a NIM client or group of NIM clients.

The following list includes all the attributes that can be specified for the **deallocate** operation:

-a *ResourceType=ResourceName* Specifies the resource to deallocate from the client, for example, `lpp_source=42_images`. This attribute is required.

-a subclass=all Specifies that all resources should be deallocated from the target. This attribute is optional.

When a resource is deallocated from a client, the **/etc/exports** file on the resource server is modified to unexport the resource from the client. The allocation count for the resource is also decremented.

define

Creates objects in the NIM environment. Networks, machines, and resources can be created using the **define** operation.

The command line syntax for the **define** operation is as follows:

```
nim -o define -t ObjectType -a Attribute=Value ... ObjectName
```

The attributes for the **define** operation vary for the different object types. For a complete description of the attributes required to define the various NIM objects, see “NIM Networks” on page 263, “NIM Machines” on page 257, “NIM Resources” on page 267, and “NIM Groups” on page 301.

diag

The **diag** operation is used to prepare resources for a client to be network-booted into diagnostics mode.

The command line syntax for the **diag** operation is as follows:

```
nim -o diag -a Attribute=Value ... TargetName|TargetNames
```

The target of a **diag** operation can be any standalone NIM client or group of standalone NIM clients.

The following are required attributes that can be specified for the **diag** operation:

-a spot=*Value* Specifies the **SPOT** resource to be used to provide network boot and diagnostics support.

The following are optional attributes that can be specified for the **diag** operation:

-a group=*Value*	Specifies the name of a resource group to use for the operation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes.
-a verbose=*Value*	Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

dkls_init

The **dkls_init** operation is used to prepare resources for use by a diskless client.

The command line syntax for the **dkls_init** operation is as follows:

```
nim -o dkls_init -a Attribute=Value ... TargetName|TargetNames
```

The target of a **dkls_init** operation can be any diskless NIM client or group of diskless NIM clients.

The following are required attributes that can be specified for the **dkls_init** operation:

-a dump=*Value*	Specifies the **dump** resource that contains client dump files.
-a paging=*Value*	Specifies the **paging** resource that contains client paging files.
-a root=*Value*	Specifies the **root** resource that contains the client root (**/**) directories. The **root** resource must be served by the same machine that serves the **SPOT**.
-a spot=*Value*	Specifies the **SPOT** resource to be used to provide network boot support and the **/usr** file system for clients.

The following are optional attributes that can be specified for the **dkls_init** operation:

-a group=*Value*	Specifies the name of a resource group to use for the installation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes.
-a home=*Value*	Specifies the **home** resource that contains client **/home** directories.
-a resolv_conf=*Value*	Specifies the **resolv_conf** resource to configure Domain Name Protocol name server information on the client.
-a shared_home=*Value*	Specifies the **shared_home** resource that contains a common **/home** directory for multiple clients.
-a size=*Value*	Specifies the size in megabytes for client paging files.
-a tmp=*Value*	Specifies the **tmp** resource that contains client **/tmp** directories.
-a verbose=*Value*	Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

The **dkls_init** operation populates client directories and creates client paging and dump files. A network boot image is also allocated to the client. When the client boots over the network, it obtains the boot image and is configured to mount the remaining resources.

dtls_init

The **dtls_init** operation is used to prepare resources for use by a dataless client.

The command line syntax for the **dtls_init** operation is as follows:

```
nim -o dtls_init -a Attribute=Value ... TargetName|TargetNames
```

The target of a **dtls_init** operation can be any dataless NIM client or group of dataless NIM clients.

The following are required attributes that can be specified for the **dtls_init** operation:

-a dump=*Value*	Specifies the **dump** resource that contains client dump files.
-a spot=*Value*	Specifies the **SPOT** resource to be used to provide network boot support and the **/usr** file system for clients.
-a root=*Value*	Specifies the **root** resource that contains the client root (**/**) directories. The **root** resource must be served by the same machine that serves the **SPOT**.

The following are optional attributes that can be specified for the **dtls_init** operation:

-a paging=*Value*	Specifies the paging resource containing client paging files.
-a group=*Value*	Specifies the name of a resource group to use for the installation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes.
-a home=*Value*	Specifies the **home** resource that contains client **/home** directories.
-a resolv_conf=*Value*	Specifies the **resolv_conf** resource to configure Domain Name Protocol name server information on the client.
-a shared_home=*Value*	Specifies the **shared_home** resource that contains a common **/home** directory for multiple clients.
-a size=*Value*	Specifies the size in megabytes for client paging files.
-a tmp=*Value*	Specifies the **tmp** resource that contains client **/tmp** directories.
-a verbose=*Value*	Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

The **dtls_init** operation populates client directories and creates client paging and dump files. A network boot image is also allocated to the client. When the client boots over the network, it obtains the boot image and is configured to mount the remaining resources.

fix_query

The **fix_query** operation is used to display whether specified fixes are installed on a client machine or a **SPOT** resource.

The command line syntax for the **fix_query** operation is as follows:

```
nim -o fix_query -a Attribute=Value ... TargetName|TargetNames
```

The target of a **fix_query** operation can be any standalone NIM client, group of standalone NIM clients, or **SPOT** resource.

The following are optional attributes that can be specified for the **fix_query** operation:

-a fix_bundle=*Value*	Specifies a **fix_bundle** resource containing a list of fix keywords. This attribute is required unless the **fixes** attribute is specified for the operation.
-a fixes=*Value*	Specifies a list of keywords for the **fix_query** operation. Fix keywords are APAR numbers used to identify software updates that can span multiple filesets. This attribute is required unless a **fix_bundle** is used for the operation.
-a group=*Value*	Specifies the name of a resource group to use for the operation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes.
-a fix_query_flags=*Value*	Tells the **fix_query** operation how to display information. Valid flags are those used by the **instfix** command.
-a show_progress=*Value*	Indicates whether status should be displayed as the operation is performed. The default value is **show_progress=yes**.

Note: There are no required attributes for the **fix_query** operation.

lppchk

The **lppchk** operation is used to verify that software was installed successfully by running the **lppchk** command on a NIM client or **SPOT** resource.

The command line syntax for the **lppchk** operation is as follows:

```
nim -o lppchk -a Attribute=Value ... TargetName|TargetNames
```

The target of a **lppchk** operation can be any standalone NIM client, a group of standalone NIM clients, or a **SPOT** resource.

The following are optional attributes that can be specified for the **lppchk** operation:

-a async=*Value*	Specifies whether NIM should perform operations on group members asynchronously and not wait for the operation to complete on one member before beginning the operation on the next. The default value is **async=yes**.
-a filesets=*Value*	Specifies a list of filesets on the target on which the **lppchk** operation will be performed.
-a lppchk_flags=*Value*	Tells the **lppchk** command how to perform software verification.
-a show_progress=*Value*	Indicates whether status should be displayed as the operation is performed. The default value is **show_progress=yes**.
-a verbose=*Value*	Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

Note: There are no required attributes for the **lppchk** operation.

lppmgr

The NIM **lppmgr** operation helps to manage base installation images and update images in an **lpp_source**. Although the **lppmgr** command is a separate command, it does use NIM objects as parameters. By having the **lppmgr** operation in NIM, the **lppmgr** command is called by NIM to be executed on **lpp_source** resources on other servers, and allows NIM to do sufficient checking of the **lpp_source** before and after **lppmgr** is executed. The format of the operation will be as follows:

The format of the command is as follows:

```
nim -o lppmgr -a lppmgr_flags=<flags> <lpp_source_object>
```

The NIM **lppmgr** operation is also available in SMIT using the **nim_lppmgr** fast path.

The **lppmgr** operation does not check the **lppmgr_flags** attribute for conflicts.

Note: Do not use the **-p** flag for prompting to move or remove installation images.

To list the names of duplicate filesets which should be removed with space usage information, type the following:

```
nim -o lppmgr -a lppmgr_flags="-lsb" lpp_source1
```

For more information on the **lppmgr** command, see the *AIX 5L Version 5.2 Commands Reference*.

maint

The **maint** operation is used to uninstall software filesets and commit and reject updates on standalone clients and **SPOT** resources.

The command line syntax for the **maint** operation is as follows:

```
nim -o maint -a Attribute=Value ... TargetName|TargetNames
```

The target of a **maint** operation can be a standalone NIM client, a group of standalone NIM clients, or a **SPOT** resource.

The following are required attributes that can be specified for the **maint** operation:

-a installp_flags=*Value* Identifies the flags that tell **installp** what to do with the installed software.

The following are optional attributes that can be specified for the **maint** operation:

-a async=*Value* Specifies whether NIM should perform operations on group members asynchronously and not wait for the operation to complete on one member before beginning the operation on the next. The default value is **async=yes**.

-a filesets=*Value* Specifies a list of filesets to be maintained on the target.

-a group=*Value* Specifies the name of a resource group to use for the operation. A resource group can be specified as an alternative to specifying multiple resources as separate attributes.

-a installp_bundle=*Value* Specifies an **installp_bundle** resource that contains a list of filesets to be maintained on the target.

-a show_progress=*Value* Indicates whether status should be displayed as maintenance is performed. The default value is **show_progress=yes**.

maint_boot

The **maint_boot** operation is used to prepare resources for a client to be network-booted into maintenance mode.

The command line syntax for the **maint_boot** operation is as follows:

```
nim -o maint_boot -a Attribute=Value ... TargetName|TargetNames
```

The target of a **maint_boot** operation can be any standalone NIM client or group of standalone NIM clients.

The following are required attributes that can be specified for the **maint_boot** operation:

-a spot=*Value* Specifies the **SPOT** resource to be used to provide network boot and maintenance mode support.

The following are optional attributes that can be specified for the **maint_boot** operation:

-a group=*Value* Specifies the name of a resource group to use for the operation.

-a verbose=*Value* Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

After the **maint_boot** operation is performed, the client must be rebooted over the network to load the network boot image and enter maintenance mode.

reboot

The **reboot** operation is used to reboot a NIM client machine.

The command line syntax for the **reboot** operation is as follows:

```
nim -o reboot -a Attribute=Value ... TargetName|TargetNames
```

The target of a **reboot** operation can be any standalone NIM client or group of standalone NIM clients.

The following are optional attributes that can be specified for the **reboot** operation:

-a inst_warning=*Value* Indicates whether a warning should be displayed to warn users that the machine will be rebooted. The default value is **inst_warning=yes**.

Note: There are no required attributes for the **reboot** operation.

remove

The **remove** operation is used to remove objects from the NIM environment. The command line syntax is as follows:

```
nim -o remove TargetName|TargetNames
```

The **remove** operation does not take any attributes. The target of this operation can be any network, machine, resource, or group in the NIM environment.

reset

The **reset** operation is used to change the state of a NIM client or resource, so NIM operations can be performed with it. A **reset** may be required on a machine or resource if an operation was stopped before it completed successfully.

The command line syntax for the **reset** operation is as follows:

```
nim -F -o reset TargetName|TargetNames
```

The target of a **reset** operation can be any NIM client, a group of NIM clients, or a **SPOT** resource.

The following list includes all the flags and attributes that can be specified for the **reset** operation:

-F (optional) Tells NIM to ″force″ the operation if the target is currently in use.

When applied to NIM clients, the **reset** operation updates the control state (**Cstate**) of the client. After the **reset** operation is performed, the client's **Cstate** is set to **ready**, and it is possible to perform NIM operations on the client. Although the **Cstate** of the client is reset by the operation, resources are not deallocated automatically. For information on deallocating resources, see "deallocate" on page 292.

When applied to **SPOT** resources, the **reset** operation updates the resource state (**Rstate**) of the **SPOT**. After the **reset** operation is performed, the **SPOT**'s **Rstate** is set to **ready**, and you can use the **SPOT** in NIM operations.

select

The **select** operation is used to include and exclude group members from operations performed on the group.

The command line syntax for the **select** operation is as follows:

```
nim -o select -a Attribute=Value ... TargetName|TargetNames
```

The target of a **select** operation must be a group of NIM clients.

The following are optional attributes that can be specified for the **select** operation:

-a exclude=*Value*	Specifies the name of the group member to exclude from operations on the group.
-a exclude_all=*Value*	Indicates that all members of the group should be excluded from operations on the group. Valid values are **yes** and **no**.
-a include=*Value*	Specifies the name of the group member to include in operations on the group.
-a include_all=*Value*	Indicates that all members of the group should be included in operations on the group. Valid values are **yes** and **no**.
-a verbose=*Value*	Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

To display the group members that are included and excluded from operations, use the **lsnim -g** *GroupName* command syntax.

showlog

The **showlog** operation is used to list software installed on a NIM client or **SPOT** resource.

The command line syntax for the **showlog** operation is as follows:

```
nim -o showlog -a Attribute=Value ... TargetName|TargetNames
```

The target of a **showlog** operation can be any standalone NIM client, a group of standalone NIM clients, or a **SPOT** resource.

The following are optional attributes that can be specified for the **lppchk** operation:

-a full_log=*Value*	Indicates whether the entire log is displayed or only the last entry. The default value is **full_log=no**.
-a log_type=*Value*	Specifies the type of log to display. The log types supported for both standalone clients and SPOT resources are:

boot	Machine's boot log
bosinst	Output from the BOS installation program
devinst	Output from the installation of key system and device-driver software
lppchk	Log of the output from the **lppchk** operation executed on a standalone NIM client
nimerr	Errors encountered during execution of the **nim** command
niminst	Output from the installation of user-specified software (including installation of NIM client software during a **bos_inst** operation)
script	Output from any configuration script resources allocated for a **bos_inst** operation.

-a show_progress=*Value*	Indicates whether status should be displayed as the operation is performed. The default value is **show_progress=yes**.
-a verbose=*Value*	Displays information for debugging. Valid values are 1-5. Use **verbose=5** to show maximum detail. The default is to show no debugging output.

Note: There are no required attributes for the **showlog** operation.

showres

The **showres** operation is used to display the contents of a resource. The contents displayed will be appropriate for the type of resource on which the operation is run.

The command line syntax for the **showres** operation is as follows:

```
nim -o showres -a Attribute=Value ... TargetName
```

The target of a **showres** operation may be a **SPOT**, **lpp_source**, **script**, **bosinst_data**, **image_data**, **installp_bundle**, **fix_bundle**, **resolv_conf**, **exclude_files**, or an **adapter_def** resource.

The following are optional attributes that can be specified for the **showres** operation:

-a client=*Value*
: Specifies which client's secondary adapter configuration file is displayed from an **adapter_def** resource. This attribute is only applicable when the target of the operation is an **adapter_def** resource.

-a filesets=*Value*
: Specifies a list of filesets for which information should be displayed. This attribute is only applicable to **lpp_source** and **SPOT** targets.

-a installp_flags=*Value*
: Specifies flags that tell the **installp** command how to format the display of filesets. This attribute is only applicable to **lpp_source** and **SPOT** targets.

-a instfix_flags=*Value*
: Specifies flags that tell the **instfix** command how to format the display of fixes. This attribute is only applicable to **lpp_source** targets.

-a lslpp_flags=*Value*
: Specifies flags that tell the **lslpp** command how to format the display of installed software. This attribute is only applicable to **SPOT** targets.

-a reference=*Value*
: Specifies a reference machine or **SPOT** resource for fileset comparison. This attribute is only applicable when the target of the operation is an **lpp_source**. Available filesets in the **lpp_source** are compared against installed filesets in the reference machine or **SPOT**. If the **showres** operation is performed from a NIM client, the **reference** attribute is automatically set to the name of the client.

-a resource=*Value*
: Specifies the name of the resource whose contents should be displayed. This attribute is only necessary when the **showres** operation is performed from a NIM client.

-a sm_inst_flags=*Value*
: Specifies flags that tell the **sm_inst** command how to format the display of filesets. This attribute is only applicable to **lpp_source** and **SPOT** targets. This attribute must be used in conjunction with the **reference** attribute and is normally used only within the SMIT application.

Note: There are no required attributes for the **showres** operation.

- When the target of the **showres** operation is a **SPOT**, the list of filesets installed in the **SPOT** is displayed.
- When the target of the **showres** operation is an **lpp_source**, the list of filesets contained in the **lpp_source** is displayed.
- For all other resources that are valid targets for the **showres** operation, the character contents of the files are displayed.

sync_roots

The **sync_roots** operation is used to verify that diskless and dataless clients have the correct root files for the **SPOT** resource they use.

The command line syntax for the **sync_roots** operation is as follows:

```
nim -F -o sync_roots -a num_parallel_syncs=Value TargetName
```

The target of a **sync_roots** operation must be a **SPOT** resource.

The following are optional flags and attributes that can be specified for the **sync_roots** operation:

-a num_parallel_syncs=*Value*	Specifies the number of client root directories to simultaneously synchronize with the **SPOT**'s root files. Valid values are numeric. The default value is **num_parallel_syncs=5**.
-F	Specifies that NIM should **force** the operation.

A **sync_roots** operation can be performed automatically when the **check** operation is performed on a **SPOT**.

unconfig

The **unconfig** operation is used to unconfigure the NIM master. The **unconfig** operation must be performed before the NIM master can be reconfigured or the NIM master fileset can be uninstalled.

Attention: Performing the **unconfig** operation removes all information from the NIM database and should be used with caution.

The command line syntax for the **unconfig** operation is as follows:

```
nim -o unconfig master
```

The target of the **unconfig** operation must be the NIM master.

No attributes can be specified for the **unconfig** operation.

The **unconfig** operation completely unconfigures the NIM master by performing the following:

- Removes the **nimesis** and **nimd** daemon entries from the System Resource Controller (SRC)
- Removes all data from the **nim_attr** and **nim_object** databases

update

The NIM **update** operation updates **lpp_source** resources by adding and removing packages. The format of the command is as follows:

```
nim -o update -a packages=<all | list of packages with levels optional> \
   [-a gencopy_flags=<flags>] ] [-a installp_bundle=<bundle_file>] \
  [-a smit_bundle=<bundle_file>] [-a rm_images=<yes>] \
  [-a source=<dir | device | object>] [-a show_progress=<yes | no>]
  <lpp_source_object>\
```

The NIM **update** operation is also available in SMIT using the **nim_update** fast path.

The source attribute must be a directory or device that is local to the server of the target **lpp_source** resource or an existing NIM **lpp_source** resource. The default operation is to add packages to the target **lpp_source**. If the **rm_images** attribute is present, the operation will remove packages from the **lpp_source**. A user must specify either the source or **rm_images** attribute and must specify the **packages**, **installp_bundle**, or **smit_bundle** attribute, but not more than one.

Generally, the **all** keyword means to perform a multi-volume installation when the source is a CD-ROM. However, the **update** operation will only do a single volume copy (equivalent to passing the **gencopy -S** flag).

The default behavior for this command is to display output. To turn off the output, pass **show_progress=no**.

To add packages to an **lpp_source** resource, run the following:

```
# nim -o update -a packages=all -a source=/tmp/inst.images lpp_source1
```

To remove packages from an **lpp_source** resource, run the following:

```
# nim -o update -a packages="bos.games 5.1.0.25 bos.sysmgt.nim" -a rm_images=yes lpp_source2
```

NIM Groups

NIM groups are essentially lists of machines and resources. Groups can be used in NIM operations to simplify repetitive administrative tasks.

Machine Groups

Machine groups are used to represent collections of machines of similar types. The machine types in a group must all be the same (for example, standalone, diskless, or dataless) and of the same architecture, because some NIM operations are restricted to certain target types. The first member added to a group determines the architecture and type of machine the group can contain. By having multiple machines defined as members of a single group, you can perform a single operation on all machines by specifying the group as the target. NIM iterates through the list of machines in the group, and performs the operation on each member, in turn.

Group members can be excluded from NIM group operations by marking them for exclusion before performing the operation. Excluding a member marks the member list in the group representation, so NIM skips the member when it iterates through the list. Excluding a member does not change the definition of the client in the NIM database. For information on marking group members for inclusion and exclusion, see “Including and Excluding a Group Member from Operations on the Group” on page 220.

Machines can be added or removed from groups, as well as be members of multiple groups. When all members of a group are removed, the group definition in the NIM database is automatically deleted. For information on adding and removing group members, see “Adding New Members to Machine Groups” on page 219 and “Removing Members from Machine Groups” on page 219.

The command line syntax for defining a machine group is:

```
nim -o define -t mac_group -a Attribute=Value ... MachineGroupName
```

where the following attributes are optional:

add_member=*Value*	Specifies the name of a NIM client to add to the machine group. NIM automatically converts this attribute to a **member** attribute with an appropriate sequence number.
member=*Value*	Specifies the name of a NIM client to add to the machine group. This attribute requires a sequence number.

Operations performed on machine groups are, by default, performed asynchronously on the non-excluded members of the group. NIM does not wait for an operation to complete on one group member before initiating the operation on the next member. When performing operations asynchronously, it is not possible for NIM to display all the output as it occurs on each client. Therefore, you should use the **lsnim** command to check the states of the group members to determine how far, and how successfully, the operations have executed. If errors do occur, the log files on client machines can be viewed using the NIM **showlog** operation.

To change the behavior of NIM group operations from asynchronous to synchronous, use the **async=no** attribute when running the **nim** command.

The number of machines permitted in a machine group is not explicitly limited by NIM. However, the following factors limit the number for practical reasons:

Operation being Performed	Operations that are not resource-intensive (such as the **maint** or **showlog** operations) may be performed on a group containing any number of machines. Operations that are resource-intensive (such as **cust** or **bos_inst**) are limited by the throughput of the network, the disk access throughput of the installation servers, and the platform type of servers.
NFS Export Limitations	The maximum number of hosts to which a file or directory may be exported with root permissions is limited by NFS to 256. Also, the length of a line in an exports file has an upper limit which could determine the maximum number of machines permitted in a group. For information on how to increase the number of machines to which a resource can be allocated, refer to “Exporting NIM Resources Globally” on page 253.

Resource Groups

Resource groups are used to represent collections of NIM resources. A resource group can contain multiple **installp_bundle** and **script** resources, and one resource from each of the other types. If a resource group is allocated or specified for use in a NIM operation, all applicable resources in the group are allocated to the target. The use of resource groups can save NIM administrators from having to repeatedly specify the same set of resources individually, when the allocation of a single resource group would suffice.

The command line syntax for defining a resource group is:

```
nim -o define -t res_group -a default=Value \
-a ResourceType=ResourceName ... ResourceGroupName
```

where the following attributes are optional:

default=*Value*	Specifies whether a resource group should be made the default. The default value is **default=no**.
ResourceType	Specifies the type (for example, **spot**, **lpp_source**, **script**, etc.) and name of the resource to add to the group. One resource of each type may be specified, except for **script** and **installp_bundle** resources, which may have multiple resources participate in an operation.

The allocation of individual resource group members can be overridden by specifying additional resource attributes for the members to be changed.

For example, the resource group, `res_grp1`, contains the `spot1`, `lpp_source1`, `bosinst_data1`, `script1`, and `resolv_conf1` resources. To use the resource group to perform an **rte bos_inst** operation on `client1`, but using no **bosinst_data** resource, and using `resolv_conf2` instead of `resolv_conf1`, use the following command:

```
nim -o bos_inst -a source=rte -a group=res_group1 \
-a bosinst_data= -a resolve_conf=resolv_conf2 client1
```

A resource group can be specified as the default set of resources to use for all NIM operations. This is done by setting the master's **default_res** attribute to the name of the resource group that will be the default. When a default resource group is defined, the applicable member resources will always be automatically allocated during NIM operations, unless they are specifically overridden.

To set the default resource group to `res_group1`, enter:

```
nim -o change -a default_res=res_group1 master
```

or enter:

```
nim -o change -a default=yes res_group1
```

To stop using a default resource group, enter:

```
nim -o change -a default_res=master
```

or enter:

```
nim -o change -a default=no res_group1
```

Chapter 25. Sample Files

This chapter contains information about the following sample files:

- “script Resource File”
- “Definition File for the nimdef Command”

script Resource File

The following is an example of a customizing script that configures the target's TCP/IP domain name resolution and routing.

The **resolv_conf** resource should be used when installing clients running the latest version of AIX.

```
#!/bin/ksh  CUSTOMIZING SCRIPT to set the hostname,
#             establish the nameserver and DNS domain name,
#             and configure the routing table for the
#             target standalone client

# Truncate the host name
# if the host name is set to the fully qualified host name
#
#NOTE: This procedure will NOT result in a truncated host name if
#the bos installation operation is installing a mksysb image
#(ie. -a source=mksysb) unless the bos_inst operation is
#instructed not to configure the target as a NIM client upon
#completion (ie. unless -a no_nim_client=yes is specified)
#
chdev -l inet0 -a hostname =$(/usr/bin/hostname | cut -d. -f1)
# Set Name server and Domain Name

if [[ -f /etc/resolv.conf ]]
then
   /usr/sbin/namerslv -E '/etc/resolv.conf.sv'
fi
/usr/sbin/namerslv -a -i '9.101.1.70'
/usr/sbin/namerslv -c 'enterprise.ca'

# Flush routing table and add default route

/etc/route -n -f
odmdelete -o CuAt -q "name=inet0 and attribute=route"
chdev -l inet0 -a route=net,,'0','9.101.1.70'
```

Definition File for the nimdef Command

The following is an example of a definition file for the **nimdef** command:

```
# Set default values.

default:
    machine_type  = standalone
    subnet_mask   = 255.255.240.0
    gateway       = gateway1
    network_type  = tok
    ring_speed    = 16
    platform      = rs6k
    machine_group = all_machines

# Define the machine "lab1"
# Take all defaults.

lab1:
```

© Copyright IBM Corp. 2002, 2004

```
# Define the machine "lab2"
# Take all defaults and specify 2 additional attributes.
# The machine "lab2" uses IPL ROM emulation, and will be added to
# the machine groups "all_machines" (by default) and "lab_machines".

lab2:
    ipl_rom_emulation  = /dev/fd0
    machine_group      = lab_machines

# Define the machine "lab3"
# Take all defaults, but do not add the machine to the
# default group.

lab3:
    machine_group=

# Define the machine "lab4"
# Take all defaults, but do not add "lab4" to the default group
# "all_machines".
# Instead add it to the groups "lab_machines" and "new_machines".

lab4:
    machine_group =
    machine_group = lab_machines
    machine_group = new_machines

# Change the default "platform" attribute.

default:
    platform = rspc

# define the machine "test1"
# Take all defaults and include a comment.

test1:
    comments = "This machine is a test machine."
```

Chapter 26. NIM Error and Warning Messages

Note: You can access NIM error messages information (message number and message text only) through the Message Center located on the IBM eServer pSeries Information Center on the Internet at the following Web address:

`http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base`

This chapter contains information about Network Installation Management (NIM) error and warning messages, with suggestions for resolving specific problems.

If an error condition is detected when a NIM command is executed, the command returns an error message. If a NIM command returns a warning message, this indicates that either a less severe problem was encountered by NIM, or a problem was encountered in a command called by NIM, and the severity of the problem cannot be readily determined by NIM. In the latter case, additional messages or output from the command often reveal the nature of the problem.

All NIM error messages begin with `0042` and are followed by a three-digit error code.

Notes:

1. If you require usage information for a NIM command, type the command without any parameters or with a question mark as a parameter (for example, `nim -?`). Additional information can be obtained from the **lsnim** command, which provides several options to display NIM help and usage information. For more information, refer to the **-q**, **-O**, and **-P** options of the **lsnim** command. You can also use the **lsnim -p -a** command to display information for all NIM classes, subclasses, types, and attributes. For example, to determine the list of valid values for an attribute, enter:

   `lsnim -p -a` *`AttributeName`*
2. In some cases, a **nim** or **nimclient** operation that is being blocked because an object is in a particular state may be permitted with the use of the **force** option (the **-F** flag). However, by using the **force** option, you may adversely affect part of the NIM environment by forcing an operation that should only proceed after other actions are complete. Use error messages that are displayed without using the **force** option to determine if the **force** operation is a reasonable action.
3. If you believe that your problem is the result of a software defect, or if the User Actions provided here do not provide adequate resolution to a problem, contact your point of sale.

Information about each message listed in this chapter is organized in the following manner:

Message	Indicates the warning or error message ID number returned by the command
Explanation	Describes what is likely to have caused the message to be displayed
User Action	Suggests a possible resolution to the problem

Note: If a User Action for a given error or warning specifies using the **lsnim** command for recovery hints, and if you are operating from a NIM client, use **nimclient -l** *lsnimOperations*, substituting the suggested **lsnim** options as appropriate.

Message	0042-001
Explanation	An error was detected by an underlying NIM method (a subcommand). This message describes where the error occurred with respect to the NIM client or master and may be useful in troubleshooting the problem. The messages that are displayed subsequent to this error are normally the true source of the problem.
User Action	Read the additional information and error messages, and refer to their explanation and recovery hints as appropriate.

Message	0042-002

© Copyright IBM Corp. 2002, 2004

Explanation An interrupt signal has been received, perhaps because the user entered Ctrl-C or used the **kill** command on a NIM process.

User Action The NIM operation that was active has been interrupted. Perform the operation again.
Note: This error is expected if it occurs after the **nimclient -o bos_inst** operation is performed on a client.

Message 0042-003 and 0042-004

Explanation An error has been returned from a system call.

User Action Fix the condition that caused the system call to fail and perform the operation again.

Message 0042-005

Explanation The Object Data Manager (ODM) has returned an error.

User Action Refer to the Message Database located on the Information Center Web page for specific details of the error. Fix the ODM problem and perform the NIM operation again.

Message 0042-006

Explanation Generic error message used for rarely occurring NIM errors.

User Action Phrases contained in this error message are constructed from debug information and from messages returned by commands called by NIM. If the content of the message does not give insight into the true cause of failure, contact your point of sale.

Message 0042-007

Explanation An internal NIM error has occurred.

User Action Try the operation again.

Message 0042-008

Explanation NIM has attempted to establish socket communications with a remote machine, and it has refused the connection.

User Action If the failing operation occurred on the master, verify that the master has **rsh** permissions on the client and that **inetd** is active on the client; otherwise, verify that the **nimesis** daemon is active on the master. If the failing operation was the **niminit** command on the client, a possible cause of failure is that the master does not have a network object that corresponds to the client's network. A network object that represents the client's network needs to be added to the database on the master; then a route needs to be added from the master's network to the client's network.

If the failure occurs during operations initiated from a client, using the **nimclient** command, or during a NIM installation of the base operating system, the **cpuid** attribute on the client's machine definition may be obsolete (for example, if the machine's system planar was recently replaced). To guarantee that this is not the case, erase the **cpuid** from the machine definition by issuing the following from the master:

```
nim -Fo change -a cpuid= ClientName
```

Message 0042-011

Explanation The **/etc/niminfo** file is not accessible.

User Action The **niminfo** file is required by all NIM commands and methods. This file is created when the **bos.sysmgt.nim.master** and **bos.sysmgt.nim.client** packages are configured. If this file is not available, this indicates that the NIM package has not been initialized or that this file has been deleted. To create the **niminfo** file, execute the **nimconfig** command on the master or the **niminit** command on the client. To recreate a deleted or corrupted **niminfo** file, enter from the master:

```
nimconfig -r
```

OR enter from the client:

```
niminit -aname=ClientName -amaster=MasterHostName -amaster_port=MasterPortValue
```

Message 0042-012
Explanation The specified command may only be executed on the master.
User Action Execute the desired operation on the NIM master.

Message 0042-013
Explanation The global lock used for synchronized access to the NIM database could not be obtained.
User Action Try the operation again. If the same error is returned, verify that there are no active NIM commands. If this is true, remove the **/var/adm/nim/glock** file and try the operation again. If the file does not exist and the error persists, contact your point of sale.

Message 0042-014
Explanation An internal NIM error has occurred.
User Action Perform the **remove** operation on the NIM object followed by the appropriate **define** operation.

Message 0042-015
Explanation A syntax error has been detected.
User Action Refer to the appropriate man page for the NIM command and try again using valid syntax.

Message 0042-016
Explanation An invalid option has been specified.
User Action Refer to the appropriate man page for the NIM command and try again using valid syntax.

Message 0042-017
Explanation An invalid value was specified for an option argument.
User Action Refer to the appropriate man page for the NIM command and try again using valid syntax.

Message 0042-018
Explanation A required option was not supplied.
User Action Refer to the appropriate man page for the NIM command and try again using valid syntax.

Message 0042-019
Explanation An option that requires an argument was specified without its argument.
User Action Refer to the appropriate man page for the NIM command and try again, specifying the missing argument.

Message 0042-20
Explanation An operand was required but not supplied. Usually, the operand is the NIM object to which a given operation is being applied (that is, a NIM name for a network, machine or resource object that is the target of the NIM operation).
User Action Refer to the appropriate man page for the NIM command and try again using valid syntax. If you do not know the name of an operand, and if the failing operation was targeted toward an existing NIM object, enter:

```
lsnim -l -t ObjectType
```

OR

```
lsnim -l
```

to determine the operand name.

Message 0042-021
Explanation A NIM attribute was required for the operation.

User Action Specify the missing attribute. If the failing command is the **nim** or **nimclient** command, to obtain a list of attributes, enter from the master:

```
lsnim -q ObjectName
```

OR

```
lsnim -q -t ObjectType
```

OR enter from the clients:

```
nimclient -l lsnimOptions
```

For the other NIM commands, see the appropriate NIM man page.

Message 0042-022
Explanation A value was specified that exceeds the bounds of acceptable values.
User Action Supply a value within the acceptable bounds.

Message 0042-023
Explanation The specified value is not valid.
User Action Try the command again with a valid value. To determine the valid values for classes of objects and operations as they pertain to those objects, enter:

```
lsnim -Pc ObjectClass
```

AND

```
lsnim -POc ObjectClass
```

where *ObjectClass* is one of machines, networks, or resources.

Message 0042-024
Explanation An invalid NIM object type was specified.
User Action Specify a valid NIM object type. See user actions for error 023 for **lsnim** options to determine a valid object type.

Message 0042-025
Explanation The specified operation cannot be supplied to the specified NIM object.
User Action Specify an operation that can be applied to the object. Enter `lsnim -O ObjectName` for a list of valid operations that can be applied to the object.

Message 0042-027
Explanation The specified object is missing an attribute that is required to complete the specified operation.
User Action Redefine the object that is missing an attribute by performing the **remove** operation followed by the **define** operation.

Message 0042-028 and 0042-029
Explanation The specified information cannot be supplied in the current context.
User Action Try the operation again without supplying the offending attribute.

Message 0042-030
Explanation A sequence number was opened to an attribute that doesn't allow sequence numbers.
User Action Try the operation again without a sequence number on the offending attribute.

Message 0042-031
Explanation An internal NIM error has occurred. NIM is unable to generate a unique object ID.
User Action Try the operation again.

Message 0042-032
Explanation The specified value for the attribute is not unique and it must be.
User Action Supply a unique value for the attribute.

Message 0042-033
Explanation The specified value is not unique and it must be. An attribute with a sequence number requires a unique value.
User Action Supply a unique value.

Message 0042-034
Explanation The specified value is not unique and it must be.
User Action Supply a unique value.

Message 0042-035
Explanation NIM was attempting to access an attribute that had the specified characteristics, but the attribute doesn't exist.
User Action Make sure the attribute exists and retry the operation.

Message 0042-036
Explanation The **define** operation failed for a resource because the specified server does not have a standalone configuration.
User Action Try the operation again using a NIM client that is a standalone machine.

Message 0042-037
Explanation The NIM state of the specified object prevents the operation from succeeding.
User Action NIM states are used to synchronize activity among NIM objects. To perform the desired operation, the state of the specified object must be changed. If the specified object is in an unexpected state, check the system to make sure another user or process is not manipulating the object. Use the **reset** operation to set the object to a known state and try the operation again.

Message 0042-038
Explanation An object that NIM would operate on is already locked and thus cannot be operated on.
User Action NIM object locks are used to synchronize activity among NIM objects. These locks are temporary, so try the operation again after some delay. The value of the lock is the process ID of a NIM process that is using the lock. If the lock persists and no NIM commands are active, reset all NIM locks by stopping the **nimesis** daemon, then restarting it.

Message 0042-039
Explanation The operating system version or release level of the specified object is unacceptable.
User Action Perform the desired operation on objects that have the appropriate operating system version and release levels.

Message 0042-040
Explanation A NIM object could not be removed because it is being used by some other NIM object.
User Action Remove all references to the object to be removed before the **remove** operation is specified. If NIM states are such that you cannot remove references to the object and you want to remove the object anyway, provide the **-F** flag to the **remove** operation.

Message 0042-041
Explanation A specified value has already been defined to NIM.
User Action Specify a value that isn't already known to NIM.
Note: If **/etc/niminfo** is the value and the NIM command producing this error is **niminit**, this means that **niminit** has already been performed. If you want to reinitialize your NIM master or client, deinstall the appropriate fileset, and then reinstall and reconfigure the NIM master or client fileset.

Message 0042-042
Explanation The specified machine could not be reached with the **ping** command from the master.

User Action If the operation you were attempting to perform requires that the target machine be running and that it can be reached, then verify that the machine is currently running. If not, turn it on; otherwise, perform network diagnostic procedures to determine why the master could not reach the target machine.

Message 0042-043
Explanation The remove operation cannot be performed, because the target machine currently serves a NIM resource that has been allocated for use. Performing the operation at this time could lead to processing failures on clients that are attempting to use the served resources.
User Action You need to deallocate all resources that the target serves before you can remove the machine.

Message 0042-044
Explanation You have specified a NIM attribute without an accompanying value. Most NIM attributes can only be specified with a value assigned to them in the form of *attr=value*.
User Action Retry the operation with a value assigned to the specified attribute.

Message 0042-045
Explanation Some NIM attributes can be added to an object's definition more than once. In these cases, a sequence number is used to uniquely identify each attribute of that type. In this case, you have specified an attribute of this type without its required sequence number and, therefore, NIM is unable to determine which attribute you are attempting to specify.
User Action Verify the sequence number and try the operation again.

Message 0042-046
Explanation NIM was unable to perform an operation on the specified file. This may be due to the permissions on the file. The file usually needs read, write, and, in some cases, execute permissions for root.
User Action Change the permissions of the specified file and try the operation again.

Message 0042-047
Explanation Some types of NIM resources may only be used by specific machine types. In this case, you attempted to allocate a NIM resource to a type of machine that is not allowed to use that type of resource.
User Action Specify a resource type that the machine is allowed to use when performing allocation for the target machine.

To determine the valid resource types, enter:

`lsnim -p -s` *`ResourceSubclassForMachineType`*

To view the subclasses that are available, enter: `lsnim -p -S`

Message 0042-048
Explanation When resource allocation is requested, NIM verifies that the designated client has the potential to communicate with the server of the resource. NIM does this by checking the NIM routing between the network that the client's primary interface connects to and all the networks that the server connects to. In this case, a NIM route is missing between the client and the server.
User Action Either establish a NIM route between the client and the server or choose a different resource to allocate.

Message 0042-049
Explanation Only one resource of this type may be allocated to the client and one has already been allocated.
User Action Choose the resource that you want to use and deallocate the currently allocated resource of this type if you want to use the new one.

Message 0042-051
Explanation NIM was unable to resolve a host name to an IP address or the other way around.

User Action All host names that are used in the NIM environment must be resolvable. Perform the appropriate network administration tasks to ensure that the specified host name is resolvable and try the operation again.

Message 0042-052

Explanation One or more NIM resources are still allocated to the machine that you have requested to be removed from the NIM environment. To remove a machine, it cannot have any resources allocated to it.

User Action Deallocate all resources that have been allocated to the target machine and try the operation again.

Message 0042-053

Explanation You have specified the name of a NIM object that does not currently exist in the NIM environment. NIM can only operate on objects that have been defined to NIM.

User Action Verify that you have spelled the name of the object correctly and that it has already been defined. The name of a target machine for a NIM operation must be the NIM name, not the host name. Enter:

```
lsnim -l -t ObjectType
```

OR

```
lsnim -l
```

to obtain listings of currently defined objects in the NIM environment. If you need to define the object, use the **define** operation.

Message 0042-055

Explanation Many NIM operations require a source for installable images. You have specified a source that cannot be used for this operation. Examples of valid sources for NIM operations are:

- **/dev/rmt0**, **/dev/cd1** for **lpp_source** definition
- **rte**, **spot**, **mksysb** for **bos_inst** operation

User Action Try the operation again using a source that the operation can use.

Message 0042-056

Explanation You have specified the same attribute assignment more than once.

User Action Try the operation again using only one instance of the attribute assignment.

Message 0042-058

Explanation You have attempted to allocate a **SPOT** to a client whose primary network interface type or platform is not supported by the **SPOT**. For a client to use a **SPOT**, the **SPOT** must support the network interface type and platform of the client's primary interface.

User Action Install the appropriate device support into the **SPOT**, which will allow the **SPOT** to support the client's primary interface type and platform, or choose a different **SPOT** that supports the client's primary interface type and platform.

Message 0042-059

Explanation In an attribute assignment (in the form of *attr=value*), the *value* you have specified represents a NIM object whose type conflicts with the object type of the specified *attr*.

User Action Try the operation again using the *attr* that corresponds to the type of object that *value* represents.

Message 0042-060

Explanation You have specified multiple attribute assignments for an attribute that may only be specified once.

User Action Try the operation again, using only one instance of the attribute.

Message 0042-061

Explanation You have requested an operation to be performed on a NIM resource object that is currently allocated for client use. NIM is not allowing this operation to be performed because it may interrupt the client's use of the resource.

User Action Try the operation again when the resource is not allocated for client use. If necessary, try the **force** option (**-F** flag) to disregard the preventive check by NIM. In some cases, NIM will allow the operation to be performed.

Message 0042-062

Explanation The NIM object that was operated on is missing something that is required for its definition to be complete.

User Action List information about the object using the **lsnim** command. Each item that is missing from the object's definition will be represented by a missing attribute. Perform the appropriate NIM operation that will add the missing item to the object's definition. For a **SPOT**, if network boot images are missing, apply the **check** operation to the **SPOT**. If software filesets are missing from a **SPOT**, allocate an **lpp_source** that contains the required filesets and apply the **cust** operation to the **SPOT**.

Message 0042-063

Explanation Some NIM operations require access to one or more NIM resources to complete successfully. This access is granted through the **allocate** operation. In this case, you have not allocated all the resources that are required for this operation.

User Action Allocate all the required resources and try the operation again. For a list of required and optional resources for a given operation, enter:

```
lsnim -q Operation ObjectName
```

OR

```
lsnim -q Operation -t ObjectType
```

Message 0042-064

Explanation The machine that is the target of the requested operation currently serves a NIM resource that is allocated for client use. The requested operation cannot be performed until all resources that the target serves have been deallocated for use.

User Action Deallocate all resources that the target serves and try the operation again.

Message 0042-065

Explanation You have specified a name that is reserved for NIM internal use only.

User Action Try the operation again using a different name. To determine what names are reserved, enter:

```
lsnim -a reserved
```

Message 0042-066

Explanation You have specified one or more characters that are not allowed in NIM object names. NIM uses regular expressions to perform many of its operations, so any character that has special meaning for regular expressions cannot be used (for example, ^). Also, any character that has special meaning to the shell cannot be used (for example, /).

User Action Try the operation again using valid characters.

Message 0042-067

Explanation You have requested an operation to be performed on a NIM object that has been reserved for NIM internal use only.

User Action Try the operation again, using a NIM object that is not reserved. To determine what objects are reserved, enter:

```
lsnim -a reserved
```

Message 0042-069

Explanation The requested operation cannot be performed at this time because it conflicts with the current NIM state of the target. NIM uses states to synchronize NIM activity so that operations don't interfere with each other.

User Action Try the operation again when the state changes or, if necessary, try using the **force** option (**-F** flag). In some cases, NIM will allow you to override this state checking.

If you encounter this error as a result of trying to remove, using the **reset** operation, the **boot** resource from a client that incorrectly has a state of ″ready for a NIM operation″, you can remove the **boot** resource from the NIM master by entering:

```
/usr/lpp/bos.sysmgt/nim/methods/m_dealloc_boot client_name
```

where *client_name* is the name of the NIM object for the client.

Message 0042-073

Explanation To perform customization on a machine, NIM constructs a shell script that is executed on the target. To construct this script, some type of resource that can be used for customization must be used. In this case, NIM could not create the customization script because no resources have been allocated to the target that could be used for customization purposes.

User Action Allocate one or more resources that can be used for customization and try the operation again. To display the subclass of resources that can be used for customization, enter:

```
lsnim -p -s cust_res
```

Message 0042-074

Explanation You have specified an attribute assignment in which the **value** represents a relative path name. NIM only allows absolute path names (that is, path names that begin with /) to be used.

User Action Try the operation again, using an absolute path name.

Message 0042-075

Explanation The requested operation requires that a NIM resource be exported for a machine's use. In this case, NIM attempted to export the resource but an error was returned by an NFS utility.

User Action Fix the error condition that the NFS utility reported and try the operation again.

Message 0042-076

Explanation You have specified a port number that is already in use.

User Action Try the operation again, using a port number that is currently not being used. Check the **/etc/services** file.
Note: NIM uses both the specified port number and its successor. Therefore, ensure that the port number after the specified port number is also free.

Message 0042-077

Explanation The **niminit** command is used to join the NIM environment. When executed, this command attempts to add routing information that the NIM master has determined the client needs to participate in the NIM environment. In this case, one or more of the required routes could not be added.

User Action Perform the appropriate network diagnostic task to determine why the route could not be added.

Message 0042-078

Explanation You have specified a change to a NIM routing attribute in which the destination network is different from its current value. This is not allowed because only the gateway field of the routing attribute may be changed.

User Action If you are trying to change the connectivity between NIM networks, then you must remove the current NIM route by supplying a NULL value for the appropriate routing attribute. Otherwise, specify the same destination network when attempting to change the gateway field of the routing attribute.

Message 0042-079

Explanation In the NIM environment, one resource may depend on another for information. In this case, an allocated resource has a dependency on the resource you have specified for deallocation.

User Action Deallocate the resource that is dependent on the resource causing the error.

Message 0042-081

Explanation NIM uses NFS to make remote resources available for client use. To avoid NFS export errors, NIM enforces some restrictions on where a resource can be defined. In general, a NIM resource cannot be defined within a directory that is already a NIM resource. Conversely, a NIM resource cannot be defined for a directory that already contains an existing NIM resource.

User Action Move the resource to a location that adheres to NIM export rules and try the operation again.

Message 0042-083

Explanation Each network communications adapter has an associated network hardware address that is unique. In this case, you attempted to define a NIM network interface using a network hardware address already being used by a NIM machine object.

User Action Only one NIM interface attribute may be defined for each network communications adapter a client might have. If you are attempting to add another interface definition, then verify that the hardware address is correct. If so, then you must first change the interface attribute that is currently using that address. If not, try the operation again with the correct hardware address.

Message 0042-084

Explanation The machine has already been configured to be a NIM master.

User Action If you want to reconfigure the machine as a NIM master, enter `nim -o unconfig master`, then deinstall and reinstall the master fileset. You may then run the **nimconfig** command.

Message 0042-086

Explanation You have attempted to add a NIM route between two NIM networks that already have a NIM route between them. Only one NIM route can be specified between any two NIM networks.

User Action If you are attempting to change NIM routing, delete the existing NIM route and try the operation again.

Message 0042-093

Explanation NIM attempted to create a directory, and the **mkdir** command returned an error.

User Action Fix the error reported by the **mkdir** command and try the operation again.

Message 0042-109

Explanation To complete the requested operation, NIM requires information about one or more file systems about which it was unable to obtain information.

User Action Verify that the file systems exist. If not, either specify the correct path name when performing the NIM operation or redefine the NIM environment so that all the participating file systems exist.

Message 0042-111

Explanation When a sequence number is specified for a NIM attribute, it must be within a specific range. You have specified a sequence number that falls outside of the acceptable bounds.

User Action Try the operation again using a sequence number that is within the acceptable bounds.

Message 0042-113

Explanation To complete the requested operation, NIM requires information about the size of one or more objects, which NIM was unable to determine.

User Action If the object is a file or directory that does not exist, then create the file or directory and try the operation again.

Message 0042-118

Explanation You have requested to change characteristics of a NIM network on which there is currently one or more active NIM operations. NIM is not allowing the change because changing the network characteristics at this time could result in failures in the active operations.

User Action Wait until the NIM resources allocated to machines that use the network being changed have been deallocated and try the operation again.

Message 0042-121
Explanation An invalid value has been specified.
User Action Try the operation again, using a valid value.

Message 0042-124
Explanation An NFS option was specified that is not supported.
User Action Try the operation again using valid NFS options. Refer to NFS Problem Determination in *AIX 5L Version 5.2 System Management Guide: Communications and Networks*.

Message 0042-129
Explanation An invalid resource type was specified for this operation.
User Action Use the **lsnim -q** *Operation* **-t** *TargetType* command to view a list of required and optional resources for *Operation* when applied to *TargetType*.

Message 0042-130
Explanation You have specified an attribute that cannot be used for this requested operation.
User Action Try the operation again, without using the attribute you specified.

Message 0042-131
Explanation You have specified two or more attributes that conflict with each other.
User Action Try the operation again, using only one of the attributes.

Message 0042-132
Explanation You have specified a **value** for an attribute assignment that is invalid in the context in which the operation is being performed.
User Action Try the operation again, using a different **value** for the attribute assignment.

Message 0042-133
Explanation The physical entity that is represented by the NIM resource object you have requested to be deleted could not be deleted.
User Action Delete the file or directory, using the **rm** command.

Message 0042-134
Explanation The operation you requested requires the designated target to reboot using a network boot image. NIM has automatically initialized the environment to enable the target to do this; however, NIM was unable to force the target to load the network boot image.
User Action Intervention at the target is required. Follow the procedures for initiating a BOOTP request.

Message 0042-135
Explanation To synchronize NIM operations that can be initiated from a client or on the master, NIM keeps track of which machine (the client or the master) is used to allocate the first resource to the client; this machine is said to be in control. For example, if the first resource allocation occurs from the client, then the client is in control. Once a machine has control, it remains in control until all resources for that client have been deallocated. You have requested an operation to be performed from a machine that is currently not in control of the specified target.
User Action Perform the desired operation from the machine that is in control of the target, or from the controlling machine deallocate the resources (to remove the control), or override this behavior by using the **force** (**-F** flag) option when performing the operation from the master.

Message 0042-136
Explanation The requested operation cannot be performed because a NIM route does not exist between two machines that participate in this operation.
User Action Establish a NIM route between the networks.

Message 0042-137

Explanation The **/etc/niminfo** file contains information about the NIM environment that all NIM commands require. In this case, the **/etc/niminfo** file is missing some information that is required to continue, which indicates that the file has been corrupted.

User Action Reinitialize the NIM environment.

Message 0042-138

Explanation Unable to update the **rhost** file.

User Action Edit the **$HOME/.rhosts** file for root on the client machine to add an entry for the host name of the NIM master.

Message 0042-139

Explanation The process of installing a machine prevents any attached disks from being used as the source for installable images. You have allocated a resource to the target of the install operation that is served by the target itself.

User Action Deallocate the resource, allocate another resource of this type that is served by another machine, and try the operation again.

Message 0042-140

Explanation You have requested that a machine object be removed from the NIM environment and this has been done; however, NIM was unable to remove the **/etc/niminfo** file on the machine that has been removed from the NIM environment.

User Action Remove the **/etc/niminfo** file from the machine that was removed from the NIM environment. **Note:** Verify that the **.rhost** permissions for the master have been removed from the client.

Message 0042-141

Explanation By specifying an attribute assignment with a NULL value, you have requested NIM to remove the specified *attr*. However, in this case, the specified *attr* is not currently part of the specified object's definition.

User Action Try the operation again, using an attribute that is part of the object's definition.

Message 0042-142

Explanation All attribute values must be unique. You have specified a *value* in an attribute assignment that already exists.

User Action Try the operation again, using a unique *value* for the attribute.

Message 0042-143

Explanation Some NIM attributes can only be added to an object's definition once. In this case, you have specified an attribute of this type when one already exists for the specified object.

User Action Only one attribute of this type can be used in the object's definition. Perform the change operation on the object if you want to replace the current value with a new one.

Message 0042-144

Explanation Some NIM attributes require a unique sequence number so that NIM can distinguish between multiple attributes of that type. In this case, you have specified a sequence number that is already being used.

User Action Try the operation again, using a sequence number that is not currently being used. To display the sequence number that are being used, enter:

```
lsnim -a AttributeName ObjectName
```

Message 0042-145

Explanation You have specified an attribute that does not exist in the NIM environment.

User Action Try the operation again, using a valid NIM attribute. To display a list of valid attribute names, enter:

```
lsnim -p -s info_subclass
```

Message 0042-146

Explanation You have specified an object type that does not exist in the NIM environment.

User Action Try the operation again, using a valid NIM object type. On the NIM master, the **lsnim** command can be used to display the valid NIM object types.

Message 0042-147
Explanation You have attempted to execute a NIM command on the NIM master that can only be executed on NIM clients.
User Action Execute the command on a NIM client.

Message 0042-148
Explanation The information contained in the specified attribute is no longer valid.
User Action Change the information in the attribute to reflect valid information and try the operation again.

Message 0042-150
Explanation Any directory used to store NIM resources must be local to the machine that serves those resources. This is required because NIM can only NFS export local directories. In this case, you have specified a directory that is not local to the designated server of the directory. NIM has obtained this information from the file system of the designated server and the **vfstype** listed corresponds to values in the **/usr/include/sys/vmount.h** file.
User Action Either copy the desired resources onto the designated server and perform the operation again, or specify the correct server when performing the operation.

Message 0042-151
Explanation For NIM to use a file, it must be of a specific type. In this case, you have specified a file whose type cannot be used by NIM. NIM has obtained this information from the file system of the designated server of the file and the file type corresponds to values in the **/usr/include/sys/mode.h** file.
User Action Change the file type of the file and try the operation again.

Message 0042-152
Explanation When an **installp** operation is performed on a **SPOT**, the root directories of all diskless and dataless clients that use that **SPOT** must be synchronized with the changes made within the **SPOT**. In this case, one or more errors occurred when performing the **root sync** operation on a root directory.
User Action Investigate why some of the root syncs failed and perform the operation again. The **nim.installp** log for the client root is located in *RootResrcParentDir*/*ClientName*/**var/adm/ras**.

Message 0042-153
Explanation For NIM to use a file, it must have specific file permissions. In this case, you have specified a file whose permissions conflict with those required by NIM. NIM has obtained this information from the file system of the designated server of the file, and the value of the file permissions comes from the **/usr/include/sys/mode.h** file.
User Action Change the file permissions of the file and try the operation again.

Message 0042-154
Explanation For NIM to use a file, it must exist. You have specified a file that does not exist.
User Action Create the file and try the operation again.

Message 0042-155
Explanation For NIM to keep diskless and dataless root directories in sync with their corresponding **SPOTs**, NIM requires that the client's root directory be served from the same machine as its **SPOT**. In this case, you have requested a resource to be allocated that violates that requirement.
User Action Try the operation again using resources that do not violate the NIM requirement.

Message 0042-156
Explanation You have requested an operation to be performed that involves a directory that does not exist.
User Action Create the missing directory and try the operation again.

Message 0042-157

Explanation The operation you have requested could not be performed because a required file could not be accessed.

User Action Create the missing file and try the operation again. For example:

- If the missing file is a boot image with a name whose format is *SpotName.NetworkInterface.Platform* (for example, `myspot.tok.up`), recreate the boot image by performing the check operation on the **SPOT**.
- If the missing files are directories with which **root** or **paging** resources are associated, delete the resource definition with the **remove** operation, create the directories, and then redefine the resource.
- If a **SPOT**'s **image.template** file is missing, this indicates that the **SPOT** has been corrupted or was not constructed successfully. To recover, you may need to remove and rebuild the **SPOT** with the **remove** and **define** operations.

Message 0042-158

Explanation The operation you have requested requires NIM to modify a file that it was unable to modify successfully.

User Action Check the file permissions on the file and try the operation again.

Message 0042-159

Explanation Required software is missing which prevents the target machine from acting as a **SPOT** server.

User Action Install the missing software and retry the operation.

Message 0042-160

Explanation The operation you requested requires the construction of network boot images and NIM was unable to do that.

User Action Fix the problem that prevented the network boot images from being constructed and try the operation again.

Message 0042-161

Explanation There is insufficient free disk space to complete the requested operation.

User Action Increase the amount of available space, as detailed in the error message.

Message 0042-162

Explanation To perform the requested operation, NIM requires an **lpp_source** type resource object that has the **simages** attribute as part of its definition. This attribute is used to designate that an **lpp_source** contains the total set of optional packages that are required to support NIM install operations. In this case, you have not supplied an **lpp_source** that fulfills this requirement.

User Action Try the operation again using an **lpp_source** that has the **simages** attribute in its definition.

Message 0042-163

Explanation NIM coordinates access between a client and the server of the resource. To do this, NIM must identify a network interface that can be used by the client. This becomes a complex problem when the server has more than one network interface. NIM uses a connectivity algorithm to establish which network interface to use. This error message occurred because the connectivity algorithm detected a problem with the client's routing and the interface the algorithm has selected to use. NIM does not allow the interface on the server that the client uses as a gateway to be used to serve resources because the operation requiring the resource could fail.

User Action If the server has other network interfaces that are not known to NIM, change the server machine object to add the interfaces.

Define a NIM route between the client's primary network and one of the other networks to which the server connects.

Message 0042-164

Explanation Some NIM operations do not allow the source of installable images to be a CD-ROM. NIM is not always able to construct an environment that supports the use of a CD-ROM for the operation being performed. This is true for the operation you tried to perform.

User Action Try the operation again using a different source for installable images.

Message 0042-165

Explanation Some attributes can only be specified together; others are mutually exclusive. In this case, you specified one or more attributes that conflict.

User Action Try the operation again, omitting the attribute that was in conflict. For example, the **ring_speed** and **cable_type** attributes cannot be used with the same **if** attribute; the one you should use depends on the type of network interface referenced by the corresponding **if** attribute.

Message 0042-166

Explanation The **if** attribute specifies network interface information, which includes a reference to the network object that the interface connects to. In this case, you have omitted a required attribute which is associated with the **if** attribute.

User Action Try the operation again, including the required attribute. For example, the **ring_speed** attribute corresponds with the Token-Ring network interface, and the **cable_type** attribute corresponds with the Ethernet network interface.

Message 0042-167

Explanation The device which you have specified as the source for the IPL ROM emulation, does not contain a valid, bootable image of the IPL ROM emulation.

User Action If the specified device has media in it, this media either does not contain the IPL ROM emulation, or the media has been corrupted. Remake the IPL ROM emulation, and try the operation again. If the specified device has no media in it, make the IPL ROM emulation, put it in the device, and try the operation again.

Message 0042-168

Explanation You have specified that the originating and destination network are the same. Machines that are on the same network do not need routing to communicate; therefore, adding a route from a network to itself is not allowed.

User Action Specify a different originating and destination network when adding a NIM route.

Message 0042-169

Explanation You have allocated an **lpp_source**, but you have not specified which filesets are to be installed using that **lpp_source**.

User Action Specify the filesets to install using the **filesets** attribute in the command, or by allocating an **installp_bundle** that contains a list of the filesets to install.

Message 0042-170

Explanation You entered a platform type that is not known to NIM.

User Action The valid platform types are **rs6k**, **rs6ksmp,** and **rspc**. Correct the platform type attribute and try the operation again.

Message 0042-171

Explanation Not all platform types are supported on all configuration types. For example, the diskless configuration type is not supported on the platform type **rs6ksmp**.

User Action Use the correct platform type and try the operation again.

Message 0042-172

Explanation You have specified the incorrect name of the machine object for the NIM client machine. When the **niminit** command is used to rebuild the **niminfo** file, the master registration process checks the CPU ID of the machine with the value stored in the NIM database for the named machine. If the stored value does not match the value passed by **niminit**, this message is issued.

User Action Use the correct name and try the command again.

Message 0042-173

Explanation You specified that the **installp** command should expand file systems (using the **-X** flag) while specifying that NIM should not auto expand (using the **auto_expand** attribute). This is not an allowable combination for the command invoked.

User Action Use either the **-X** flag or the **auto_expand** attribute, but not both.

Message 0042-174

Explanation You specified an invalid value for an attribute whose only valid values are **yes** and **no**.

User Action Retry the operation with a value of **yes** or **no** for the attribute indicated.

Message 0042-175

Explanation An unexpected result has been returned from a command that NIM tried to execute.

User Action Fix the problem that caused the executed command to fail and try the operation again.

If the command failed due to a shortage of space, its error messages indicating this should be displayed. Expand the indicated file system, (for most NIM operations use the **auto_expand** attribute) and retry the operation. If a space failure occurred during **SPOT** creation, and if the **bosboot** command failed to make boot images as a result, increase the free space and run the **check** operation.

If the command listed by NIM in this message is the **installp** command, check the **nim.installp** log for failure and recovery information. (For standalone client operations, this is located in the **/var/adm/ras** directory of the client. For **SPOT cust** and **maint**operations, this is located in *SPOTParentDir*/SPOTName**/usr/lpp/bos/inst_root/var/adm/ras** on the **SPOT**. For diskless and dataless clients, this is located in *RootResrcParentDir*/*ClientName***/var/adm/ras**.

Message 0042-176

Explanation The resource cannot serve as a support image (**simages**)**lpp_source**. When an **lpp_source** serves as a support image resource, it contains a minimal set of software packages for facilitating installation and the use of the base operating system.

User Action No action is necessary if this resource does not need to serve as a support images **lpp_source**. If the resource needs to be a support images **lpp_source**, add the missing software to the **lpp_source**. If the **lpp_source** is a directory, you can do this by copying the missing packages to the location of the **lpp_source** and running the **check** operation.

Message 0042-177

Explanation The operation you requested could not be completed due to insufficient free space in one or more file systems.

User Action Make more space available if possible, by extending the file system displayed. For most NIM operations, the **auto_expand** attribute is available to automatically expand file systems.

Message 0042-178

Explanation The **if** attribute is made up of four fields. The fourth field is optional in most cases. In this case. the network object that you specified (in field *one*) has more than one type of network. In this case, NIM requires that the fourth field has a value that specifies the logical device name of the network adapter.

User Action Add the appropriate value to the **if** attribute, and try the operation again.

Message 0042-179

Explanation You are attempting to remove an **if** or **other_net_type** attribute on which one or more NIM clients have a dependency.

User Action If this is not a mistake, remove the NIM clients that are dependent on the network, or remove the **if** attribute from the NIM client object definition.

Message 0042-180

Explanation The address of the machine that is being defined is not connected to the network that is represented by the specified network object.

User Action Define a network object that represents the physical network to which the machine is connected. Use this network object when defining the machine.

Message 0042-181
Explanation The **fix_query_flags** attribute has an illegal value. Use **lsnim -Pa fix_query_flags** for a list of legal values.
User Action Determine the correct flags and retry the operation.

Message 0042-182
Explanation A resource of one type cannot be allocated for the current operation at the same time as a resource of another type. Allocate one or the other, but not both.
User Action The resources specified are mutually exclusive. Determine which one is needed for the operation, and omit the other.

Message 0042-183
Explanation An attribute cannot be specified for the current operation when a type of resource is allocated. Use one or the other, but not both.
User Action The attribute and the resource specified are mutually exclusive. Determine which one is needed for the operation, and omit the other.

Message 0042-184
Explanation The network address (**net_addr**) or subnet mask (**snm**) cannot be changed for the network, because NIM clients are currently defined as being connected to that network. Remove the client definitions before changing the network.
User Action The **nimdef** command can be used to quickly redefine NIM clients after they have been removed to update the network definition.

Message 0042-185
Explanation Failed to link or copy files. Check permissions and file system space.
User Action Verify that space and inodes are available for the files and links specified in the error message.

Message 0042-186
Explanation Failed to copy setup programs. Either start NFS on the client or free 1000 512-byte blocks in the file system.
User Action Programs required to set up the operation could not be copied to the client system. Either start NFS on the client, or increase space in the file system specified in the error message.

Message 0042-187
Explanation Failed to expand file system.
User Action Attempt to manually expand the file system specified in the error message, then retry the operation.

Message 0042-188
Explanation Failed to NFS mount.
User Action Verify that NFS is running on both the resource server and the client specified in the error message. Retry the operation when the NFS problems have been resolved.

Message 0042-189
Explanation Failed saving existing boot image. Check space in the file system.
User Action Increase space in the file system specified by the error message, and retry the operation.

Message 0042-190
Explanation The key is *not* in the NORMAL position. Unattended installation cannot complete unless the key is in the NORMAL position.
User Action Turn the key on the client machine to the NORMAL position and retry the operation.

Message 0042-191

Explanation Unable to write the IPLROM emulation.

User Action The **mkboot** command failed to write the IPLROM emulation on the client. Boot the client manually over the network to begin the BOS installation.

Message 0042-192

Explanation Unable to find boot logical volume.

User Action Verify that a boot logical volume is defined for the machine. NIM attempts to use the **lslv -l hd5** command to determine the boot logical volume.

Message 0042-193

Explanation The client does not have an **.rhosts** entry for the master, or the client host ID is not resolvable.

User Action Verify that the client host name is resolvable by the master. Then verify that an entry exists for the master in the **$HOME/.rhosts** file for root on the client machine.

Message 0042-194

Explanation The client does not allow NIM **push** operations. Remove **/etc/nimstop** on %s if **push** operation is necessary.

User Action On the client machine, run the **nimclient -p** command to re-enable master push permissions.

Message 0042-195

Explanation Unable to order boot device list.

User Action An error was returned by the **bootlist** command on the client. If a network boot must be performed for a **bos_inst**, **diag**, or **maint_boot** operation, manually set the boot list and reboot the client, or follow the normal procedure to boot the client over the network.

Message 0042-196

Explanation The **set_bootlist** attribute is only valid when used in combination with the **no_client_boot** or **boot_client** attributes.

User Action Only specify the **set_bootlist** attribute to the **nim** command when changing the default behavior with the **no_client_boot** or **boot_client** attributes.

Message 0042-197

Explanation If the target machine has more than one interface for a given network type, the network adapter's logical device name must be specified in the **if1** attribute of the target machine's NIM definition when using the **force_push** attribute.

User Action Modify the client's **if1** attribute using the NIM **change** operation. Change the **if1** attribute to include one of the client's network adapter logical device names listed in the error message.

Message 0042-198

Explanation When converting a machine's **/usr** file system to a **SPOT**, the bos image on the media (**lpp_source**) being used to create the **SPOT** must match the bos image that was used to install the machine.

User Action When defining the **/usr SPOT**, use the same installation media that was used to install the machine originally. For example, if a machine was originally installed with AIX 4.3.2 and then updates were applied to bring the machine to AIX 4.3.3, the installation media that should be used when defining the **/usr SPOT** on the machine would still need to be the AIX 4.3.2 product media.

Message 0042-199

Explanation The **no_client_boot** and **boot_client** attributes may not be specified together.

User Action To avoid the possibility of giving conflicting instructions to the NIM command, do not supply both the **no_client_boot** and **boot_client** attributes in the same NIM operation.

Message 0042-204

Explanation The **mk_image** and **source** attributes are only valid when specified together.

User Action When creating a **mksysb** resource from a running client machine, use the **mk_image=yes** attribute to indicate that a **mksysb** should be created, and use the **source=***ClientName* attribute to specify the name of the client that is to be backed up.

Message 0042-205
Explanation The **bos.sysmgt.sysbr** fileset must be installed on the client to perform the system backup. You may install this fileset with the NIM **cust** operation.
User Action Install the **bos.sysmgt.sysbr** fileset on the client machine before retrying the operation.

Message 0042-206
Explanation There is already a resource allocated.
User Action Only one resource of the type specified can be allocated to the client. Deallocate the first resource before attempting to allocate the other.

Message 0042-207
Explanation Unable to allocate a resource to a client.
User Action Look for other NIM error messages that may accompany this error and which may provide more information about the problem. Verify that the resource specified is NFS-exportable to the client.

Message 0042-208
Explanation Unable to lock a client. This could mean that the client is already locked, or the name given does not refer to a valid NIM client.
User Action If another NIM operation is being performed on the same client, wait for the process to complete before retrying the operation. If no other NIM operations are being performed, stop and restart the **nimesis** daemon to remove locks.

Message 0042-209
Explanation The **mksysb_flags** attribute contains an illegal value. Use the **lsnim -Pa mksysb_flags** command to get a list of legal values.
User Action Specify the correct values for the **mksysb_flags** attribute, and retry the operation.

Message 0042-210
Explanation The maximum space required for the backup is greater than the amount of free space in the target file system. To ignore space requirements, use the **-F** flag when defining the **mksysb** resource.
User Action Either increase the space of the target file system where the **mksysb** is to be created, or use the **-F** flag as specified in the error message.

Message 0042-211
Explanation The member already exists in group.
User Action No additional action is required, since the member is already added to the group.

Message 0042-212
Explanation The member was not added to the group, because it is not a valid NIM name.
User Action The name of a member to add to a group was invalid. Verify that the member was specified correctly.

Message 0042-213
Explanation The group was not created, because it did not contain any valid members.
User Action A group must contain at least one member. Redefine the group with valid members to add it to the NIM environment.

Message 0042-214
Explanation Unable to add a member to a group.
User Action Look for other NIM error messages that may accompany this error and which may provide more information about the problem.

Message 0042-215
Explanation An invalid log type for the **showlog** operation was specified.
User Action Specify one of the valid log types listed in the error message.

Message 0042-216
Explanation An invalid log type for the **showlog** operation was specified for a **SPOT**.
User Action Specify one of the valid log types listed in the error message.

Message 0042-217
Explanation An invalid log type for the **showlog** operation was specified for a diskless or dataless machine.
User Action Specify one of the valid log types listed in the error message.

Message 0042-218
Explanation The log file is either empty or does not exist.
User Action No information is available in the log file for the machine or **SPOT** specified.

Message 0042-219
Explanation The object is incompatible with the group.
User Action The object cannot be added to the group, because its type is not allowed in the group. Machine groups can only contain one type of NIM client, and that type is determined by the first member added. Resource groups can only contain members whose types are resources.

Message 0042-220
Explanation You cannot have more than one resource of the specified type in a resource group.
User Action You must remove the current member with the specified type from the resource group before the new member with the same type can be added.

Message 0042-221
Explanation The group *GroupName* is being removed, because its single remaining member was removed during this operation.
User Action A group cannot be empty. Redefine the group with at least one member if it should remain in the NIM environment.

Message 0042-222
Explanation An unknown error occurred allocating resources to the machine.
User Action Look for other NIM error messages that may accompany this error and which may provide more information about the problem. Verify that the resource specified is NFS-exportable to the client.

Message 0042-223
Explanation Invalid input file. The file either cannot be read, is empty, or contains no valid entries.
User Action Verify that the file specified in the error message is the correct file for the operation.

Message 0042-224
Explanation The limit on the length of a line in an NFS exports file was exceeded. The **export** operation cannot be performed.
User Action Manually edit the **/etc/exports** and **/etc/xtab files** to remove any obsolete entries. The number of hosts to which NIM can NFS-export a resource can also be increased by setting the **restrict_nfs_exports** attribute to **no** on the master by running the **nim -o change -a restrict_nfs_exports=no master** command.

Message 0042-225
Explanation An error occurred while updating the exports file. Check for corruption in the file.

User Action Manually edit the **/etc/exports** and **/etc/xtab** files to fix any file corruption problems. Attempt to determine why NIM was unable to successfully update the files. Check file and directory permissions, and verify that file systems are not full.

Message 0042-226

Explanation A timeout occurred while attempting to initiate the operation on the client. The operation may not have started successfully.

User Action If the operation that was performed was **bos_inst**, the client only needs to be rebooted manually over the network to begin the installation. For all other operations, the problem is most likely due to network communication problems between the master and the client. Verify that the client is reachable by the master and that **rsh** permission is still granted by the client to the master.

Message 0042-227

Explanation The state of the machine indicates that it may not be ready for certain NIM operations.

User Action Check to see if any NIM operations are still being performed on the machine. If not, reset the state of the machine with the **nim -Fo reset** *MachineName* command. This returns the machine to the **ready** state so NIM operations can be performed on it. The **reset** operation does not deallocate resources, so deallocate resources if necessary using the **nim deallocate** operation.

Message 0042-228

Explanation Invalid release level.

User Action The release level of the resource is incomplete, or incorrectly specified. The level of the resource can be obtained by running the **lsnim -l** *ResourceName* command and viewing the **version**, **release**, and **mod** attributes. To correct the problem, either recreate the resource, or modify the NIM database to contain the correct level using the command on the NIM master:**/usr/lpp/bos.sysmgt/nim/methods/m_chattr -a** *Attribute*=*Value ResourceName*, where *Attribute* is **version**, **release**, or **mod**; *Value* is the correct value; and *ResourceName* is the name of the resource with the incorrect level specification.

Message 0042-229

Explanation When installing a system using a **mksysb** as the source for the installation, the level of the **SPOT** used for the installation must match the level of the **mksysb** image being installed. The release levels of the **SPOT** and the **mksysb** do not match.

User Action Create a **SPOT** that matches the level of the **mksysb** being installed, and use that **SPOT** when performing a **mksysb** BOS installation. The level of **mksysb** and **SPOT** resources can be obtained by running the **lsnim -l** *ResourceName* command and viewing the **version**, **release**, and **mod** attributes.

Message 0042-230

Explanation When installing a system using a **mksysb** as the source for the installation, the level of the **SPOT** used for the installation should match the level of the **mksysb** image being installed. If this convention is not followed, the installation may not complete successfully.

User Action Create a **SPOT** that matches the level of the **mksysb** being installed, and use that **SPOT** when performing a **mksysb** BOS installation. The level of **mksysb** and **SPOT** resources can be obtained by running the **lsnim -l** *ResourceName* command and viewing the **version**, **release**, and **mod** attributes.

Message 0042-231

Explanation A temporary list of software that should be installed is created and used for this operation. The list could not be created.

User Action Check previous error messages to understand why the error occurred. Correct the problem and try the operation again.

Message 0042-232

Explanation A temporary **installp_bundle** resource is created and used for this operation. The temporary resource could not be created.

User Action Check previous error messages to understand why the creation of the resource failed. Correct the problem and try the operation again.

Message 0042-233
Explanation The operation cannot be performed because the NIM Master is already initialized.
User Action Unconfigure the NIM Master and try the operation again.

Message 0042-234
Explanation You cannot restore a NIM database backup onto a machine that has an earlier level of the NIM master fileset installed. For example, a NIM database backup of a system with level 4.2.0.0 of the NIM master cannot be restored to a system that has a level of the NIM master lower than 4.2.0.0.
User Action Install a level of the NIM master fileset that is at the same level or a later level than that from which the backup was created. Then attempt to restore the NIM database backup.

Message 0042-235
Explanation An image source was not specified for creating the **SPOT**.
User Action Specify a device containing installation images or specify an **lpp_source** with the **simages** attribute for creating the **SPOT**.

Message 0042-236
Explanation A name for the **lpp_source** and/or a directory to contain the **lpp_source** was not specified for the **lpp_source** that will be created.
User Action Specify a name and a directory for the **lpp_source** and try the operation again.

Message 0042-237
Explanation A name for the **SPOT** and/or a directory to contain the **SPOT** was not specified for the **SPOT** that will be created.
User Action Specify a name and a directory for the **SPOT** and try the operation again.

Message 0042-238
Explanation A parent directory was not specified for the diskless and dataless machine resources that will be created.
User Action Specify a directory for the diskless/dataless machine resources and try the operation again.

Message 0042-239
Explanation A name for the resource and/or directory to contain the resource was not specified for the resource that will be created.
User Action Specify a name and a directory for the resource and try the operation again.

Message 0042-240
Explanation A parent directory was not specified for the diskless and dataless machine resources that will be created.
User Action Specify a directory for the diskless/dataless machine resources and try the operation again.

Message 0042-241
Explanation The size and/or volume group was not specified for the creation of a new file system to contain a NIM resource.
User Action Specify both the size and volume group for the file system and try the operation again.

Message 0042-242
Explanation The size and/or volume group was not specified for the creation of a new file system to contain diskless and dataless machine resources.
User Action Specify both the size and volume group for the file system and try the operation again.

Message 0042-243

Explanation An attempt was made to create the same file system twice: once for an **lpp_source** and once for a **SPOT**.

User Action Specify a different directory for either the **lpp_source** or the **SPOT**. This will cause different file systems to be created for the resources. If a new file system really should be created to contain both resources, then only specify that the file system should be created for one of the resources, but specify the same directory for both resources.

Message 0042-244

Explanation An attempt was made to create the same file system twice: once for an **lpp_source** and once for diskless/dataless machine resources.

User Action Specify a different directory for either the **lpp_source** or the diskless/dataless resources. This will cause different file systems to be created for the resources. If a new file system really should be created to contain both sets of resources, then only specify that the file system should be created for one of the resources, but specify the same directory for both resources.

Message 0042-245

Explanation An attempt was made to create the same file system twice: once for a **SPOT** and once for diskless/dataless machine resources.

User Action Specify a different directory for either the **SPOT** or the diskless/dataless resources. This will cause different file systems to be created for the resources. If a new file system really should be created to contain both sets of resources, then only specify that the file system should be created for one of the resources, but specify the same directory for both resources.

Message 0042-246

Explanation Not enough space on the volume group to create the specified file system.

User Action Specify a different volume group for the file system to be created and try the operation again.

Message 0042-247

Explanation Creation of the file system failed.

User Action Check the previous output for error messages to understand what caused the file system creation to fail. Correct the error and try the operation again.

Message 0042-248

Explanation An error occurred during file system creation.

User Action Check the previous output for error messages to understand what caused the file system creation to fail. Correct the error and try the operation again.

Message 0042-249

Explanation NIM master initialization failed.

User Action Check the previous output for error messages to understand what caused the configuration of the NIM master to fail. Correct the error and attempt to reinitialize the master. The most frequent cause of this failure is that the master is already initialized. The master can be unconfigured with the **nim -o unconfig master** command and reinitialized. However, this should be done with extreme caution, since unconfiguring the master will remove all definitions from the NIM database.

Message 0042-250

Explanation Unable to continue with configuration.

User Action Check the previous output for error messages to understand what caused the configuration to fail. Correct the error and attempt to configure the system again from the point of failure.

Message 0042-251

Explanation A route cannot be added to the network, because a required default route is missing. Add a default route to the network, and try this operation again.

User Action Add a default route to the network specified in the error message, and retry the operation.

Message 0042-252

Explanation Unable to locate a matching network.

User Action The **find_net** keyword was used in the **if** attribute of the machine. However, no matching network was found. Either define the network prior to defining the machine interface, or use the **net_definition** attribute in conjunction with the **find_net** keyword to define the network while the interface is being defined.

Message 0042-253

Explanation You cannot use the **net_definition** attribute when the **find_net** keyword is not specified as the first field of the **if** attribute.

User Action The **net_definition** attribute is invalid when using a known network in the **if** attribute. Specify the **find_net** keyword in the **if** attribute, or omit the **net_definition** attribute, and retry the operation.

Message 0042-254

Explanation Invalid format for the specified value of **net_definition**. The value of the attribute should be as follows:

NetType
Network type (for example, `tok`, `ent`, `fddi`, etc.).

snmName
Dotted decimal subnet mask for the network.

Client_gwName
Optional default gateway IP address or host name used by the machine being defined to communicate with the master.

Master_gwName
Optional default gateway IP address or host name used by the master to communicate with clients on other subnets.

NetName
Optional name given to the NIM definition created for the network. (Otherwise, a unique default name is used.)

If you want to specify *NetName* and if *Client_gwName* or *Master_gwName* are not applicable, specify **0** in their place. If *Client_gwName* is **0**, *Master_gwName* cannot be nonzero.

User Action Correct the syntax error, and retry the operation.

Message 0042-255

Explanation The master already has a default route, and the gateway you specified as being the default for the master is different from that which is already defined. Use the **change** operation if you want to modify the master's default gateway.

User Action To change the default gateway for a network, use the following command:

```
nim -o change -a routingX="default GtName" NetName
```

where *X* is the sequence number for the **routing** attribute; *GtName* is the default gateway to use; and *NetName* is the name of the master's network.

Message 0042-256

Explanation A default route already exists for the network. You can modify the default gateway, but you cannot define more than one default route.

User Action To change the default gateway for a network, use the following command:

```
nim -o change -a routingX="default GtName" NetName
```

where *X* is the sequence number for the **routing** attribute; *GtName* is the default gateway to use; and *NetName* is the name of the network to modify.

Message 0042-257

Explanation You cannot specify the **net_definition** attribute without specifying the **if** attribute when changing a machine definition.

User Action The **net_definition** must reference a machine interface, so specify an **if** attribute when using the **net_definition** attribute.

Message 0042-258
Explanation You cannot specify the **net_definition** attribute when creating or modifying more than one if attribute in the same **change** operation. Use two separate operations.
User Action To avoid ambiguity, manipulate only one machine interface (**if** attribute) at a time when using the **net_definition** attribute.

Message 0042-259
Explanation The value of **default_res** specified on the master's database definition is not a valid NIM resource group.
User Action Specify a valid NIM resource group as the default resource. Obtain a list of resource groups by running the **lsnim -t res_group** command.

Message 0042-260
Explanation The **default** attribute is only applicable when manipulating a resource group.
User Action Setting the **default=yes/no** attribute on a resource group makes it the default set of resources to use in NIM operations. The **default** attribute is invalid when used as an attribute in other NIM operations.

Message 0042-261
Explanation Illegal use of the **async** attribute. This attribute can only be specified for the **lppchk** operation when the target is a standalone machine or a group of standalone machines.
User Action Omit the **async** attribute when performing the **lppchk** operation, unless the target is a standalone machine or a group of standalone machines.

Message 0042-262
Explanation The file name of the client definition file is missing for this operation.
User Action Specify the client definition file that should be used to add machines to the NIM environment. For more information, see "Network Installation Management Commands Reference" on page 212.

Message 0042-263
Explanation The **netboot_kernel** attribute can only be assigned a value of **up** or **mp**.
User Action Correct the value specified for the **netboot_kernel** attribute.

Message 0042-264
Explanation The image source that was used to define the **lpp_source** is missing one or more requested packages.
User Action Installation images were not copied into the **lpp_source** directory. The source for installation images may not contain all of the filesets specified to populate the **lpp_source**. Copy the missing installation images to the **lpp_source** directory, and then perform the NIM **check** operation on the **lpp_source**.

Message 0042-265
Explanation The image source that was used to define the **lpp_source** is missing one or more items from the list of default packages.
User Action Installation images were not copied into the **lpp_source** directory. The source for installation images may not contain all of the default filesets used to populate the **lpp_source**. Copy the missing installation images to the **lpp_source** directory, and then perform the NIM **check** operation on the **lpp_source**.

Message 0042-266
Explanation Requested packages are missing from the defined **lpp_source**.

User Action Installation images were not copied into the **lpp_source** directory. The fileset names may have been specified incorrectly, or the source for installation images may not contain all of the specified filesets. Copy the missing installation images to the **lpp_source** directory, and then perform the NIM **check** operation on the **lpp_source**.

Message 0042-267
Explanation The defined **lpp_source** does not have the **simages** attribute, because one or more packages are missing.
User Action Copy the missing installation images to the **lpp_source** directory, and perform the NIM **check** operation on the **lpp_source** to add the **simages** attribute.

Message 0042-268
Explanation The operation cannot be performed, because all members of the target group specified are currently excluded from operations on the group. You must unmark (or include) excluded group members before proceeding.
User Action Perform the NIM **select** operation on the group to include members in further operations.

Message 0042-269
Explanation Only one type of verification can be performed at a time when verifying installed filesets on a NIM client.
User Action Disable or deselect all but one verification option and try the operation again.

Message 0042-270
Explanation The operation is only supported on **SPOTs** and NIM clients installed with a version and release level of AIX 4.2 or greater.
User Action The NIM client fileset on the target is at an earlier level and does not support the attempted operation. The client software on the target must be upgraded before the operation can be performed.

Message 0042-271
Explanation A resource matching the type is already allocated. You cannot allocate more than one resource of this type to a machine.
User Action Deallocate the first resource before attempting to allocate the second. It may be necessary to reset the machine before the resource can be deallocated.

Message 0042-272
Explanation A value specified is not a valid value for **default_re** because it is not a valid NIM resource group.
User Action Specify a different resource group for the **default_res** attribute, or correct the resource group in question.

Message 0042-273
Explanation A value specified cannot be used as the location for the **mksysb** image because it is a directory. You must specify the filename where the **mksysb** image currently resides or will reside after creation.
User Action Specify a file name instead of a directory for the location of the **mksysb** resource.

Message 0042-274
Explanation The **-e** flag in the **mksysb_flags** attribute and the **exclude_files** attribute cannot be specified together. Specify the **-e** flag with the **mksysb_flags** attribute to exclude the files in **/etc/exclude.rootvg** from the backup, or specify an **exclude_files** attribute.
User Action Do not specify both the **-e mksysb** flag and an **exclude_files** resource when performing this operation.

Message 0042-275
Explanation Unable to obtain possession of a lock file. If no NIM operations are currently in progress, remove the file and repeat the operation.

User Action Use the **ps -ef | grep nim** command to list the running NIM processes on the system. If any NIM processes other than the **nimesis** daemon are running, wait for them to finish and then remove the file specified by the error message.

Message 0042-276

Explanation A fileset must be installed before this operation can be performed.

User Action Install the fileset listed in the error message before retrying the operation. Generally, the fileset needs to be installed on the client system. However, depending on the operation being performed, the NIM master may also need to have the fileset installed before the operation will succeed.

Message 0042-277

Explanation Diskless and dataless machines cannot be defined with a primary network install interface residing on a generic NIM network. It is presumed that a network adapter defined on a generic NIM network does not support network boot.

User Action To define the systems as diskless or dataless clients, they must first be connected to a NIM network that is known to support network boot, such as ethernet, token-ring, or FDDI.

Message 0042-278

Explanation The interface specified does not correspond to a network adapter that is known to support network boot. As a result, the NIM master has been defined on a generic NIM network. Network boot-dependent operations, such as base operating system installation, will not be possible on any NIM client whose primary network install interface is defined on the same network as the NIM master.

User Action Operations that rely on network boot capability cannot be performed on clients on generic NIM networks. Such operations must be performed using local media on the system.

Message 0042-279

Explanation The interface specified maps to a subnet which has been defined as a generic NIM network. It will not be possible to perform network boot-dependent operations, such as base operating system installation, on the machine definition created by this operation.

User Action Operations that rely on network boot capability cannot be performed on clients on generic NIM networks. Such operations must be performed using local media on the system.

Message 0042-280

Explanation Specify a complete date and time for the scheduled operation in the form: `YYMMDDhhmm`.

User Action Use the format described in the error message to correctly schedule a date and time for the operation.

Message 0042-281

Explanation The **/usr** file system on the specified server cannot be converted to a NIM **SPOT**. Either the **RM_INST_ROOTS** variable was set to **yes** in a **bosinst.data** file during initial installation of the machine or **inurid -r** was subsequently invoked. The only way to create a **SPOT** on this machine is to specify the location to be something other than **/usr** or reinstall the machine and then create a **SPOT** in **/usr**.

User Action The system is unable to support the creation of a **/usr SPOT**. A **non-/usr SPOT** may be created on the system by specifying a different value for the **location** attribute.

Message 0042-282

Explanation The BOS installation has been enabled but could not be initiated, because the following file was not found on the target. To start the installation, do one of the following:

1. Initiate a network boot operation from the target.
2. Correct the state of the target with NIM's **reset** operation and invoke the **bos_inst** operation again using one of the following:
   a. The Force Push option (**-a force_push=yes**)
   b. After installing and configuring the **bos.sysmgt.nim.client** fileset on the target.

User Action The NIM client fileset is not properly installed and configured on the target system. Follow the directions specified in the error message to correct the problem.

Message 0042-283

Explanation The existence of a file on the server indicates that a NIM **SPOT** may still be mounted in a subdirectory which will be removed by this operation. Before attempting the operation again, unmount the **SPOT**'s directory along with any other directories that may be mounted beneath the directory being removed.

Failure to do so will result in loss of data on the **SPOT** server.

User Action A **SPOT** operation failed, and NIM was unable to unmount all the directories mounted into the **SPOT**. Manually unmount the directories specified in the error message before retrying the operation. The **mount** command can be used to list the directories mounted on the system, and the **unmount** command can be used to unmount directories. Use the **-f** option with the **unmount** command if necessary to force the unmount.

Chapter 27. NIM Troubleshooting

This chapter suggests solutions for network boot problems and describes procedures for producing debug output for NIM BOS installations. Refer to Chapter 26, “NIM Error and Warning Messages,” on page 307 for information about error messages.

Debugging a Network Boot Problem

If a client machine is unable to network boot from its boot server, there may be a problem in one or more of the network boot stages. The network boot stages are listed in the following tasks:

- “Verifying Network Communication Between the Client and Server”
- “Obtaining the Boot Image from the Server”
- “Running the Boot Image on the Client” on page 336.

Verifying Network Communication Between the Client and Server

Before initiating the network boot on the client, do the following to verify network communication between the client and the server:

1. Perform a ping test from the client **bootp** menus.
2. If the ping test fails, verify that the client, server, and gateway addresses are specified correctly.
3. If the addresses are correct, try to ping the server from a different machine in the client’s subnet.

   If the server can be pinged from another machine, the network adapter on the boot client may be faulty.
4. If the server cannot be pinged from another machine in the client’s subnet, there may be routing problems between the client and the server, or network communications on the server may be faulty. For information on network-debugging procedures, refer to TCP/IP Problem Determination in the *AIX 5L Version 5.2 System Management Guide: Communications and Networks*.

Obtaining the Boot Image from the Server

1. If the ping test is successful, perform a network boot of the client. When a network boot is initiated on a client, a **bootp** request packet is sent from the client to the server. The server then replies with a packet to the client. The client machine displays the number of packets sent and received for the **bootp** request. If a packet is sent from the client, but none is received, another packet will be sent.

   If **bootp** packets continue to be sent but not received, the boot server may not be responding to the request.
2. From the **bootp** server, view the **/etc/bootptab** file on the server. It should contain an entry for the client machine with the following information:

   ```
   hostname_of_client
   bf=boot_file
   ip=client_ip_address
   ht=network_type
   sa=boot_server_address
   sm=client_subnet_mask
   ha=network_adapter_hardware_address (required only if bootp requests are sent by broadcasting)
   ```

   If an entry does not exist, either the NIM command used to set up the current operation failed, or the machine was reset before the boot operation could occur. Rerun the NIM **bos_inst**, **diag**, **or maint_boot** operation to prepare the server for the client boot request.

   If the entry exists in **/etc/bootptab**, verify that the specified data is correct. If a field contains incorrect data, the information that was used to define the machine or network in the NIM database was probably incorrect. Correct this problem by resetting the client machine, correcting the invalid data in the client or network definition, retrying the NIM operation, and rebooting the client.

© Copyright IBM Corp. 2002, 2004

3. If the **/etc/bootptab** file is correct, verify that the **inetd** daemon is running. If it is not running, start it and retry the network boot from the client. If the **inetd** daemon is running, it should automatically start the **bootpd** daemon when the **bootp** request is received at the server.
4. If the **bootpd** daemon is not started, verify that the **bootps** entry in the **/etc/inetd.conf** file is not commented out. If it is commented out, uncomment it and restart **inetd** with the **refresh -s inetd** command. Retry the network boot from the client.
5. If a **bootp** reply is still not received at the client, manually start the **bootpd** daemon in debug mode:
   a. Comment out the **bootps** entry from the **/etc/inetd.conf** file on the server.
   b. Stop all running **bootpd** processes.
   c. Restart **inetd** using the **refresh -s inetd** command.
   d. Start **bootpd** from the command line, using the **/usr/sbin/bootpd -s -d -d -d** command.
6. Retry the network boot from the client. If no output is displayed from the running **bootpd** command, the client **bootp** request is not reaching the server. Verify that the addresses specified in the **bootp** menus are correct. If they are correct, perform network debugging procedures to determine why the packet is not reaching the server.

   If the server receives the client **bootp** request, the running **bootpd** command displays output matching the client data in the **/etc/bootptab** file. Verify that the specified addresses are correct. This information is sent back to the client in the **bootp** reply.
7. If the client is still not receiving the **bootp** reply, perform network-debugging procedures to determine why the reply packet is not reaching the client.

   After the client receives the **bootp** reply, it will **tftp** the boot image from the server.

   The number of **tftp** packets transferred to the client will be displayed at the client machine.

   The boot image has been successfully retrieved at the client machine when the LED shows 299 on **rs6k**-platform machines or when the bottom third of the screen turns gray on other platform machines.
8. If the **tftp** of the boot image does not complete successfully, the client may be trying to get the wrong boot image. Verify that the client definition in the NIM database shows the correct platform and kernel type. If the data is incorrect, correct it, reset the client machine, rerun the NIM operation, and reboot the client over the network.
9. Verify that the **/tftpboot** directory on the boot server contains a link with the client name to the correct boot image. If the link does not exist, reset the client machine, rerun the NIM operation, and reboot the client over the network.
10. If the link with the client name is pointing to the correct boot image and the **tftp** of the boot image does not complete successfully, the boot image may be corrupted. Re-create the boot image by performing a NIM **check** operation with the **force** flag on the **SPOT**. If the client is not an **rs6k**-platform machine, also make sure the client has the latest version of the firmware installed.

Running the Boot Image on the Client

After the client machine has successfully received the boot image from the server, the most common errors encountered are hangs with the LED showing 608, 611, or 613. Some machines may not have LED displays. Debugging such problems on these machines will require using debug-enabled boot images. For information on building debug boot images, see “Producing Debug Output from the BOS Installation Program” on page 338.

608
Explanation **tftp** retrieve of client info file failure.

Action If a 608 hang is encountered, verify that the *ClientName*.info file exists in the **/tftpboot** directory. If it does not exist, retry the NIM operation to create it. If it does exist, verify that **tftp** access to the **/tftpboot** directory is not restricted in the **/etc/tftpaccess.ctl** file. It is also possible that the network adapter was not configured properly in the boot environment. Use debug-enabled network boot images to look for errors in the boot environment. If the client is not an **rs6k**-platform machine, make sure that it has the latest version of firmware installed.

611

Explanation Remote mount of NFS file system failure.

Action 611 hangs occur when the client machine is unable to mount a resource from a server. Ensure that NFS is running on the resource server. Verify that the resources specified for the operation are exported properly by checking the **/etc/exports** and **/etc/xtab** files on the server. Also, confirm that the resources have permissions set correctly for reading. Debug-enabled network boot images can also be used to determine exactly which **mount** command is failing on the client.

613

Explanation Failure setting up route tables.

Action 613 hangs usually occur because a route is incorrectly defined for a network in the NIM database. Verify that the correct gateways are specified between networks, and all gateways are functional. Use debug-enabled network boot images to determine which routes could not be defined.

Producing Debug Output for NIM BOS Installations

Due to problems in the network or in the NIM configuration, clients may fail to boot or install properly. When this happens, it may be necessary to produce debug information in order to determine the cause of the problem. If a client machine fails to configure properly from the network boot image, debug output from the boot image can be obtained by building the debug-enabled image and attaching a tty to the client system. This will display the commands and output that are run while the client is configured before further processing is done by AIX.

If the system has been booted from the network boot image, but failures are still occurring during a BOS installation, it may be necessary to collect debug information from the BOS installation program. The commands and output from the BOS installation program will automatically be displayed on the tty if the boot image was built debug-enabled. If the boot image was not built for debugging, output can be obtained by either setting a value in a **bosinst.data** file or by entering special codes at the installation menus.

When problems arise during a NIM BOS installation, you will most likely get system hangs. Viewing the debug output can be useful, because you will be able to see the commands that failed. The problem may be a misconfiguration of the network adapter or an inability to perform an operation from the client to the server. By examining the debug output, you can determine what failed and make corrections to avoid the error in the future.

You will see the **showled** command running in the debug output. This command displays status values on the LEDs on the front of the machine. Frequently, known problems and solutions are referenced by the LED value that is displayed when a problem occurs. Some machines do not have LEDs for displaying such information. Therefore, when debugging problems on such machines, give special attention to observing the values that the **showled** commands are displaying.

Obtaining debug information from a network installation can save you time in determining the root cause of a problem. Usually, the problem will be an incorrect definition in the NIM environment that can be found without the debug information. However, with the debug information, you can significantly reduce the scope of the investigation.

Producing Debug Output from a Network Boot Image

To create debug versions of the network boot images, do the following:

1. Use the Web-based System Manager or SMIT interfaces or run the following command:

   ```
   nim -Fo check -a debug=yes SPOTName
   ```

   where *SPOTName* is the name of your **SPOT**.
2. Obtain the address for entering the debugger by doing the following:

   From Web-based System Manager:

   a. From the NIM Resources container, double-click the **SPOT** resource to open the properties notebook.

   b. Click the Boot Image Information tab in the properties notebook to obtain the address.

   Alternatively, you can use the following command to get the address:

   ```
   lsnim -a enter_dbg SPOTName
   ```

   where *SPOTName* is the name of your **SPOT**. The displayed output will be similar to the following:

   ```
   spot1:
      enter_dbg = "chrp.mp 0x001840d4"
      enter_dbg = "chrp.up 0x00160b7c"
      enter_dbg = "rs6k.mp 0x001840d4"
      enter_dbg = "rs6k.up 0x00160b7c"
      enter_dbg = "rspc.mp 0x001840d4"
      enter_dbg = "rspc.up 0x00160b7c"
   ```

   Write down the **enter_dbg** address for the client you are going to boot. For example, if your client is an **chrp**-uniprocessor machine, you would write down the address `160b7c`.
3. Attach a tty device to your client system (port 1).
4. Set up and perform the NIM operation that will require the client to boot over the network. Boot the client over the network.
5. After the client gets the boot image from the **SPOT** server, the debug screen will appear on the tty. At the > prompt, enter:

   ```
   st Enter_dbg_Value 2
   ```

   where *Enter_dbg_Value* is the number you wrote down in step 2 as your machine type's **enter_dbg** value. Specifying a 2 at the address of the **enter_dbg** value prints the output to your tty.
6. Type `g` (for `go`) and press Enter to start the boot process.
7. Use Ctrl-s to temporarily stop the process as you watch the output on the tty. Use Ctrl-q to resume the process.
8. To rebuild your boot images in non-debug mode, use the following command:

   ```
   nim - Fo check SPOTName
   ```

   where *SPOTName* is the name of your **SPOT**.

   If the boot image is left in debug mode, every time a client is booted from these boot images, the machine will stop and wait for a command at the debugger ″>″ prompt. If you attempt to use these debug-enabled boot images and there is not a tty attached to the client, the machine will appear to be hanging for no reason.

Producing Debug Output from the BOS Installation Program

There are two ways to obtain debug output from the BOS installation program. Method A involves entering a special value at one of the installation menus and Method B uses a **bosinst_data** resource to tell the installation program to display debug output. Both methods are described as follows:

Method A: To Produce Debug Output Without Using a bosinst_data Resource

1. To enable debugging for the BOS installation program, start by performing all the processing you would normally do to install a client.

   Because you are not using a **bosinst_data** resource, you will be prompted to supply information about the installation to the BOS installation program.
2. Select your console.
3. Select your language.
4. The **Welcome to Base Operating System Installation and Maintenance** menu is displayed. Instead of selecting one of the options, type `911` at the prompt and press Enter.
5. Continue the normal procedure for selecting options and specifying data until the installation begins. Debug output will be sent to the client's display while the installation proceeds.

Method B: To Produce Debug Output When Using a bosinst_data Resource

1. To enable debugging for the BOS installation program, set the value `BOSINST_DEBUG = yes` in the `control_flow` stanza of the **bosinst.data** file that you are using for your **bosinst_data** resource.

   A minimum **bosinst.data** file for debugging purposes would contain the following lines:

   ```
   control_flow:
        BOSINST_DEBUG = yes
   ```
2. In addition to the processing you would normally do to install a client, include the modified **bosinst_data** resource as a resource for the operation.

   After the client boots over the network, it will use the **bosinst_data** resource to obtain settings for the installation. If the only data specified in your **bosinst.data** file is `BOSINST_DEBUG = yes`, you will be prompted for the remaining required information before the installation will continue. Debug output will be sent to the client's display while the installation continues.

Port Number Conflicts with NIM and Other Applications

When the NIM Master is configured, two port numbers are selected to be used by the **nimesis** daemon for client communications. The default port numbers are 1058 and 1059. If either port is taken by another application, the **nimesis** daemon will not run and nimclient commands will fail with an error similar to the following:

0042-006 nimclient: (To master) rcmd connection refused

If the **nimesis** daemon cannot be started, it may be necessary to stop the other applicants on the system to free the port.

Rebooting the system will usually eliminate the problem, because when a machine is booted, the **nimesis** daemon is started very early by **init** and the likelihood that the ports are taken will be very small.

Part 4. Appendixes

© Copyright IBM Corp. 2002, 2004

Appendix A. Network Boot

Note that in the examples, `tok0` and `ent0` are adapter names. Do not use network names, such as `tr0`, `en0`, or `et0`.

If an **rs6k** machine is not running, it is possible to determine whether IPL ROM emulation is required by booting the machine with the key turned to Secure. If the LEDs on the front of the machine eventually stop at 200, no emulation is needed.

To create IPL ROM emulation, see “Creating IPL ROM Emulation Media” on page 346. If you are using NIM on machines that must use IPL ROM emulation to boot from a network adapter, you should always initiate the boot once from the IPL menus from a system console on the client. This writes the necessary addresses to NVRAM, avoiding a problem with seeing alternating LEDs 227 and 229 during the network boot on older hardware. After initiating the network boot/install from the client via this method once, the problem is automatically corrected and subsequent network boot/installation may be initiated from the NIM master.

The platform and kernel type of a client determines the procedure required to boot the machine over the network.

To determine the platform of a running machine, use the **bootinfo -p** command if the machine is running AIX Version 4.2 or later. If the machine is running AIX 4.1, use the **bootinfo -T** command.

To determine the kernel type of a running machine, use the **bootinfo -z** command.

Booting a Machine Over the Network

If you are booting an rs6k machine with an up kernel, use Method A. If you are booting an rs6k machine with an mp kernel, use Method B. For some models of rspc machines, you may use Method C. For all other platforms and kernel types, follow the procedures in your hardware documentation to perform the network boot.

Method A

1. Begin with your machine turned off.
2. If your client requires IPL ROM emulation, insert the emulation disk into the diskette drive of the client, and turn on the machine with the hardware key in the Service position. When the **bootp** menus display, continue with step 3.

   If your client does not require emulation, turn the key to the Secure position and turn on the machine. Watch the LEDs on the front of the machine. They will eventually stop changing and display **200**. Then, change the key position to Service and quickly press the reset (yellow) button. When the **bootp** menus display, continue with step 3.

   Note: On model numbers 570, 580, 58H, 59H, 591, and 595, the system powers on in the Secure mode, the power-on light does not come on, and the LED remains blank. The system performs no further operations until the key is set to the Normal or Service position. Refer to the documentation that came with your particular hardware model for more information about performing a network boot of the machine.
3. From the **bootp** main menu, choose the Select BOOT (Startup) Device option.
4. In the next menu, select the boot device.

   Select the network adapter to be used. Choose the adapter with the correct network type (Ethernet, Token-Ring, etc.) and adapter characteristics (thick cable, twisted pair, 16 Mb data rate, 4 MB data rate, etc.).

© Copyright IBM Corp. 2002, 2004

5. Set or change the network addresses.

   Note: You do not need to type the '.' characters in the IP addresses, but you must specify any leading `0' characters that make up parts of the addresses.

   Specify the IP address of:

   - The client machine you are booting in the `client address` field.
   - Your **SPOT** server in the `bootp server address` field.
   - Your client's gateway in the `gateway address` field.

   After you specify the addresses, enter **99** to save the addresses and return to the main menu. If no gateway is used by the client to communicate with the boot server, you can leave this field empty.

   If broadcasting will be used to obtain a boot image, leave these IP addresses empty.
6. From the main menu, select the Send Test Transmission (PING) option.
7. Verify that the displayed addresses are the same as the addresses you specified for your boot device.

   If the addresses are incorrect, enter **99** to return to the main menu. Then, go back to step 3.

   If the addresses are correct, select the START THE PING TEST option.

   If the ping test fails, verify that the addresses are correct, and perform network problem determination if necessary. If the ping test completes successfully, enter **99** to return to the main menu.
8. From the main menu, select the Exit Main Menu and Start System (BOOT) option.
9. Turn the hardware key to the Normal position, and press Enter to boot your client over the network.

Method B (Booting Micro Channel-Based, Symmetric Multiprocessor Systems)

1. Turn the key mode switch to the Secure position.
2. Turn the power switch on the system unit to the On position.
3. When the LED displays `200`, turn the key mode switch to the Service position.
4. Press the Reset button once.

   The LED persistently displays `260`, `261`, or `262`, and The Maintenance Menu screen appears.
5. Select the **System Boot** option on the Maintenance Menu screen.
6. Select the **Boot from Network** option from the sub-menu. The MAIN MENU is displayed.
7. Select the **Select BOOT (Startup) Device** option on the MAIN MENU screen.
8. Select the network adapter from which the machine will boot. If there are multiple network adapters installed, type `88` and press the Enter key to view the other entries. Type a number from the list and press the Enter key.

   Note: If you are using a Token-Ring network, select the number that corresponds to the correct ring speed for your environment.
9. If a network adapter is selected, the SET OR CHANGE NETWORK ADDRESSES screen is displayed next. The hardware address for the network adapter is displayed in the hardware address field. Record the hardware address for defining the NIM machine object.

   If this client and the BOOTP server are on the same LAN, leave the IP address fields as zeros for the BOOTP request to be broadcasted over the LAN. If there are multiple BOOTP servers on the LAN or the client is on a different network to the server, enter the client and server IP addresses. Type in the IP addresses using leading zeros to pad the network address fields, for example, `009.101.002.050`. If this machine must use a gateway to reach the server, enter the IP address for the gateway.

   Type `99` and press the Enter key to save the address information and return to the MAIN MENU.
10. (This step is optional.) Select the **Send Test Transmission (PING)** option on the MAIN MENU to test the network connection between the client and the server systems. Type `3` and press the Enter key to

start the ping test. If the ping test was not successful, check that the IP addresses are correct and that the physical network connections are sound. If the ping test was successful, type `99` and press the Enter key to return to the MAIN MENU.

11. Select the **Exit Main Menu and Start System (BOOT)** option.
12. Follow the instructions on the screen to turn the key mode switch to the Normal position and press the Enter key.

The BOOTP request will be issued, followed by a TFTP transfer of the network boot image.

Method C (Booting an rspc Platform Machine)

1. Begin with your machine turned off.
2. If your system requires a System Management Services (SMS) diskette, insert it into the diskette drive of the client and turn on the machine. If you do not insert an SMS diskette at this time and one is required, you will be prompted to insert one later.
3. A graphics image is displayed on your screen. Press the **F4** key as icons begin to display from left to right on the bottom of your display.

   Note: If the last icon is displayed prior to pressing the **F4** key, the normal mode boot list is used instead of the System Management Services diskette.
4. The System Management Services menu displays on your screen. Select the **Utilities** option.
5. From the System Management Services Utilities menu, select the **Remote Initial Program Load Setup** option.
6. From the Network Parameters screen, select the **IP Parameters option**.
7. Set or change the values displayed so they are correct for your client system.

   Note: You do not need to specify any leading `0' characters, but you must specify the `.' characters in the IP addresses.
8. Specify the IP address of:
   - The client machine you are booting in the `client address` field.
   - Your **SPOT** server in the `bootp server address` field.
   - Your client's gateway in the `gateway address` field.

   Note: If broadcasting will be used to obtain a boot image, leave these IP address fields empty.
9. Specify the subnet mask for your client machine if you are prompted for one in the `subnet mask` field. All machines in your subnet have the same subnet mask.
10. After you specify the addresses, press Enter to save the addresses and continue.
11. The Network Parameters screen is displayed. Select the **Ping** option.
12. Select the network adapter to be used as the client's boot device.
13. Verify that the displayed addresses are the same as the addresses you specified for your boot device.
14. If the addresses are incorrect, press Esc until you return to the main menu. Then, go back to step 5.
15. If the addresses are correct, press Enter to perform the ping test. The ping test may take several seconds to complete.
16. If the ping test fails, verify that the addresses are correct, and perform network problem determination if necessary. If the ping test completes successfully, press Enter to acknowledge the *success* message. Then, press Esc until you return to the System Management Services menu.
17. From the System Management Services menu, choose the **Select Boot Devices** option.

18. Select the network adapter to be used for the network boot from the list of displayed bootable devices. Be sure to select the correct network type (Ethernet, Token-Ring, etc.) and adapter characteristics (thick cable, twisted pair, 16 Mb data rate, 4 Mb data rate, etc.). After making your selection, the machine will boot over the network.

Note: When performing a BOS installation on a NIM client with a ″rspc″ platform, the machine may fail to boot from the network adapter if network traffic is heavy.

If the network boot was initiated from the NIM Master, the machine will eventually boot from the disk. If the network boot was initiated from the SMS (System Management Services) menus on the NIM client, the machine will return control to the SMS menus.

Contact your service representative to receive a firmware update to correct this problem.

Creating IPL ROM Emulation Media

Use this procedure to create the IPL ROM emulation media on the NIM master for machines that do not have a BOOTP-enabled IPL ROM.

1. Insert a formatted diskette or a tape into the appropriate drive on the NIM master.
2. Enter:

```
bosboot -T rs6k \
-r /usr/lpp/bos.sysmgt/nim/methods/IPLROM.emulation \
-d DeviceName -M both
```

where *DeviceName* can be `fd0`, `/dev/fd0`, `rmt0`, or `/dev/rmt0`. This operation requires that the **devices.base.rte** fileset be installed on the machine upon which the emulation media is being created.

3. Insert the IPL ROM emulation media in the appropriate drive on the target machine.

From Web-based System Manager

1. From the NIM container, from the NIM menu, select **Create IPL ROM Emulation Media**.
2. Use the dialog to complete the task; all fields are required.

From SMIT

IPL ROM emulation can also be created using the **smit iplrom** fast path.

Appendix B. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Dept. LRAS/Bldg. 003
11400 Burnet Road
Austin, TX 78758-3498
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

© Copyright IBM Corp. 2002, 2004

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include 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.

Trademarks

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

AIX
AIX 5L
eServer
IBM
pSeries
RS/6000
SP
Xstation Manager

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Java and all Java-based trademarks and logos are registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Glossary

/usr file system. Contains files and programs necessary for operating the machine.

/tmp file system. A shared storage location for files.

/var file system. Contains files that are variable on a per-client basis, such as spool and mail files.

/ file system. The root file system; contains files that contain machine-specific configuration data.

APAR. Authorized program analysis report. A report of a problem caused by a suspected defect in a current, unaltered release of a program.

applet. A program, intended for delivery over the Internet, which can be included in an HTML page, just as an image can be included.

apply. When a service update is installed or *applied*, it enters the applied state and becomes the currently active version of the software. When an update is in the applied state, the previous version of the update is stored in a special save directory. This allows you to restore the previous version, if necessary, without having to reinstall it. Software that has been applied to the system can be either *committed* or *rejected*. The **installp -s** command can be used to get a list of applied products and updates that are available to be either committed or rejected. See also *commit* on page 349 and *reject* on page 352.

Base Operating System (BOS). The collection of programs that controls the resources and the operations of the computer system.

boot device. The device that assigns the fixed disk within the root volume group (rootvg) that contains the startup (boot) image.

bosinst.data. The file that controls the actions of the BOS installation program.

bundle. A collection of software products available for installation.

CD-ROM. High-capacity, read-only memory in the form of an optically read compact disc.

clean up. The clean-up procedure instructs the system to attempt to remove software products that were partially installed. The system also attempts to revert to the previous version of the removed product. If the system successfully reverts to the previous version, it becomes the currently active version. If this cannot be done, then the software product is marked as broken. After the clean up procedure is complete, you can attempt to install the software again.

client. In a distributed file system environment, a system that is dependent on a server to provide it with programs or access to programs.

commit. When you commit software updates, you are making a commitment to that version of the software product. When you commit a product update, the saved files from all previous versions of the software product are removed from the system, thereby making it impossible to return to a previous version of the software product. Software updates can be committed at the time of installation by using either the Web-based System Manager or SMIT interface (or by using the **-ac** flags with the **installp** command). Note that committing already applied software does not change the currently active version of the software product. It merely removes saved files for the previous version of the software product. Once a new version of a product update is committed, you cannot reject it if you want to return to the previous version of the code. To return to the previous version of the code, you must reinstall the base level of the software product and reapply any associated updates. You should not commit software updates until you are sure the update does not cause other problems or regressions. Compare to *apply* on page 349 and contrast with *reject* on page 352 and *remove* on page 353.

complete overwrite installation. An installation method that completely overwrites an existing version of the Base Operating System that is installed on your system. This procedure might impair recovery of data or destroy all existing data on your hard drives. Be sure to back up your system before doing a complete overwrite installation.

Configuration Assistant. A graphical interface application used to perform post-installation system configuration tasks.

© Copyright IBM Corp. 2002, 2004

configure. To describe to a system the devices, optional features, and program products installed on a system.

console device. During the installation of the Base Operating System (BOS), the system console is the display device at the system on which you are installing the software.

corequisite. A product or update that must be installed concurrently with another specified product or update.

daemon. A program that runs unattended in the background to perform a standard service. Some daemons trigger automatically to perform their task and others operate on a timed or periodic basis.

dataless. A workstation without local file systems or local boot images that accesses some of its resources remotely. Dataless clients use a local disk used for paging and dump devices.

dependent. A software product that requires another product or update to be installed *before* or *at the same time* it is installed. Contrast with *prerequisite* (page 352).

destination disk. The disk to which you are installing.

directory. A type of file containing the names and controlling information for other files or other directories.

diskless. A workstation without local file systems or local boot images that accesses some of its resources remotely. Diskless clients boot remotely from a diskless server and use the server for remote paging.

display. A computer output screen on which visual information is displayed.

display device. See *display* (page 350).

environment. (1.) The settings for shell variables and paths that are set when the user logs in. These variables can be modified later by the user. (2.) A named collection of logical and physical resources used to support the performance of a function.

environment variable. (1.) A variable that describes the operating environment of the process. Common environment variables describe the home directory, command search path, the terminal in use, and the current time zone (the **HOME**, **PATH**, **TERM**, and **TZ** variables, respectively). (2.) A variable that is included in the current software environment and is therefore available to any called program that requests it.

file. The collection of related data that is stored and retrieved by an assigned name. Contrast with *special file* (page 353).

file system. The collection of files and file management structures on a physical or logical mass storage device, such as a diskette or minidisk.

file tree. The complete directory and file structure of a particular node, starting at the root directory. A file tree contains all local and remote mounts performed on directories and files.

fileset. An individually installable option or update. Options provide specific function and updates correct an error in, or enhance, a previously installed option.

fixed disk. (1.) A flat, circular, nonremovable plate with a magnetizable surface layer on which data can be stored by magnetic recording. A rigid magnetic disk used in a fixed-disk drive. (2.) The term fixed disk is also used loosely in the industry for boards and cartridges containing microchips or bubble memory that simulate the operations of a fixed-disk drive.

full path name. The name of any directory or file expressed as a string of directories and files beginning with the root directory. See also *path name* (page 352).

graphical user interface. A type of computer interface consisting of a visual metaphor of a real-world scene, often a desktop. Within that scene are icons, representing actual objects, that the user can access and manipulate with a pointing device.

hard disk. See *fixed disk* (page 350).

hardware. The physical equipment of computing and computer-directed activities. The physical components of a computer system. Contrast with *software* (page 353).

host. (1.) The primary or controlling computer in a communications network. (2.) A computer attached to a network.

host name. The Internet address of a machine in the network. Also known as the host ID.

HTML. HyperText Markup Language is the tagging language that a web browser uses to interpret and display documents.

hypertext. A way of presenting information online with connections between one piece of information and another. These connections are called hypertext links. Thousands of these hypertext links enable you to explore additional or related information throughout the online documentation. See also *hypertext link* (page 351).

hypertext link. A connection between one piece of information and another. See also *hypertext* (page 351).

icon. A picture or graphical representation of an object on a display screen to which a user can point to with a device, such as a mouse, to select a particular operation or perform a certain action.

initial program load (IPL). (1.) The initialization procedure that causes an operating system to commence operation. (2.) The process by which a configuration image is loaded into storage at the beginning of a work day or after a system malfunction. (3.) The process of loading system programs and preparing a system to run jobs.

input device. The device that is the source of the software you are installing. The input device can be a tape drive, CD-ROM drive, diskette drive, or a directory.

Installation Assistant. An application used to perform system configuration tasks.

installation image. An installation image contains a copy of the software you are installing in backup format, as well as copies of other files the system needs to install the software product.

Internet address. The numbering system used in TCP/IP internetwork communications to specify a particular network or a particular host on that network with which to communicate. Internet addresses are commonly denoted in dotted decimal form.

IPL. See *initial program load* (page 351).

license password. The key that allows a software product to be used. A string encoded with license information for a software product.

locale. A subset of a user's environment that defines conventions for a specified culture, such as time formatting, numeric formatting, monetary formatting, and character classification, conversion,and collation.

logical partition (LP). (1.) One to three physical partitions (copies). The number of logical partitions within a logical volume is variable. (2.) A fixed-size portion of a logical volume. A logical partition is the same size as the physical partitions in its volume group. Unless the logical volume of which it is a part is mirrored, each logical partition corresponds to, and its contents are stored on, a single physical partition. See also *logical volume* (page 351).

logical volume (LV). A collection of physical partitions organized into logical partitions all contained in a single volume group. Logical volumes are expandable and can span several physical volumes in a volume group. See also *logical partition* (page 351), *volume group* (page 353), and *migration installation.* (page 351)

maintenance level update. The service updates that are necessary to upgrade the Base Operating System (BOS) or an optional software product to the current release level. See also *service update* (page 353).

migration installation. An installation method for upgrading AIX 3.2 or later to the current release while preserving the existing root volume group. This method preserves the /usr, /tmp, /var, and / (root) file systems, as well as the root volume group, logical volumes, and system configuration files. Migration is the default installation method for any machine that is running AIX 3.2 or later. See also *root volume group* (page 353) and *logical volume* (page 351).

monitor. (1.) A device that observes and verifies operations of a data processing system. (2.) Synonym for *display*.

mount. To make a file system accessible.

name server. A host that provides name resolution for a network. Name servers translate

symbolic names assigned to networks and hosts into the efficient Internet addresses used by machines.

Network File System (NFS). A distributed file system that enables users to access files and directories located on remote computers and treat those files and directories as if they were local. NFS is independent of machine types, operating systems, and network architectures through the use of remote procedure calls (RPC).

Network Installation Management (NIM). An environment that provides installation and configuration of software within a network interface.

new installation. An installation method used when the fixed disk or disks you are installing BOS onto are empty. A fixed disk is considered empty if it does not contain any data or if it contains data not in a volume group.

NIM. See *Network Installation Management* (page 352).

Object Data Manager (ODM). A data manager intended for the storage of system data. The ODM is used for many system management functions. Information used in many commands and SMIT functions is stored and maintained in the ODM as objects with associated characteristics.

option. An installable unit of a software package. Software product options are separately installable units that can operate independently from other options of that software package.

optional software. Also referred to as *optional software products*. Software that is *not* automatically installed on your system when you install the Base Operating System (BOS). Optional software can be products packaged and sold with BOS. Optional software can also be separately purchased software products that are specially ordered and not sold as part of BOS. In either case, BOS must be installed on your system before you can install optional software.

package. An installable unit of a software product. Software product packages are separately installable units that can operate independently from other packages of that software product.

paging. (1.) The action of transferring instructions, data, or both between real storage and external page storage. (2.) Moving data between memory and a mass storage device as the data is needed.

path name. A file name specifying all directories leading to the file. See also *full path name* (page 350).

physical volume. The portion of a single unit of storage accessible to a single read/write mechanism; for example, a drum, a disk pack, or part of a disk storage module.

preinstalled. Software that is installed by the manufacturer and ready to use.

prerequisite. A software product or a service update that must be installed *before* another software product or service update is installed. If you attempt to install software products or service updates without the required prerequisite software, a system message displays the names of required prerequisite software. Contrast with *dependent* (page 350).

preservation installation. An installation method used when a previous version of the Base Operating System (BOS) is installed on your system and you want to preserve the user data in the root volume group. However, this method overwrites the **/usr**, **/tmp**, **/var**, and **/** (root) file systems, so any user data in these directories is lost. System configuration must be done after doing a preservation installation.

Preventive Maintenance Package (PMP). A maintenance level update for your system. A PMP includes updates for the Base Operating System (BOS) and for each optional software product that is installed on your system.

primary language. The primary locale you want your system to use for screen information.

Problem Management Record (PMR). A number assigned by a support center to a reported problem.

product. A software product is made up of software packages that are separately installable.

reboot. To reinitialize the execution of a program by repeating the initial program load (IPL) operation.

reject. To cause portions of applied updates from becoming permanent parts of the product, based

on the results of a test period. When you reject an applied service update, the update's files are deleted and the software vital product data (SWVPD) information is changed to indicate that the update is no longer on the system. The previous version of the software, if there is one, is restored and becomes the active version of the software. Contrast with *apply* (page 349) and *commit* (page 349).

remove. For a software option, the deletion of the option and all of its applied or committed updates from the system. The software vital product data (SWVPD) information is changed to indicate that the option has been removed from the system. Depending on the option, system configuration information is also cleaned up, although this is not always complete. If a previous version, release, or level of the option is on the system, the system does not restore the previous version. Only an option with its updates can be removed. Updates cannot be removed by themselves. See also *commit* (page 349).

requisite. A software product or a service update that must be installed with another software product or service update. If you attempt to install software products or service updates without the required software, a system message displays the names of required software.

root user authority. The unrestricted ability to access and modify any part of the operating system, usually associated with the user who manages the system.

root volume group (rootvg). A volume group containing the Base Operating System (BOS). See also *migration installation.* (page 351)

server. On a network, the computer that contains the data or provides the facilities to be accessed by other computers on the network.

service update. Software that corrects a defect in or adds new function to the Base Operating System (BOS) or to an optional software product. See also *maintenance level update* (page 351).

SMIT. See *System Management Interface Tool* (page 353).

software. Programs, procedures, rules, and any associated documentation pertaining to the operation of a system. Contrast with *hardware* (page 350).

source. A system, a program within a system, or a device that makes a request to a target. Contrast with *target* (page 353).

special file. Used in the operating system to provide an interface to input/output devices. There is at least one special file for each device connected to the computer. Contrast with *directory* (page 350) and *file* (page 350).

stacked tape. A bootable tape with multiple software images.

System Management Interface Tool (SMIT). A set of menu-driven services that facilitates the performance of such system tasks as software installation and configuration, device configuration and management, problem determination, and storage management. SMIT is provided in both a character-based curses interface and an AIXwindows-based graphical user interface.

target. A system, a program within a system, or a device that interprets, rejects, or satisfies, and replies to requests received from a source. Contrast with *source* (page 353).

Transmission Control Protocol/Internet Protocol (TCP/IP). A communications subsystem that allows you to set up local area and wide area networks.

Universal Coordinated Time (UCT). The standard term for worldwide time-telling that has the same meaning as Greenwich Mean Time.

update. See *service update* (page 353).

upgrade. Software that fixes a defect in a previously released software product.

verify. The verify procedure instructs the system to verify the software you are installing. The system confirms that your software files are the correct length and contain the correct number of digits and characters. If any errors are reported, it might be necessary to install the software product again. The verification process can add a significant amount of time to the installation process.

volume group (VG). A set of one or more physical volumes from which space can be allocated to one or more logical volumes. A collection of 1 to 32 physical volumes (read-write

fixed-disk drives) of varying size and type. See also *logical volume* (page 351).

Web-based System Manager. A graphical user interface (GUI) tool for managing systems. Based on the OO (Object Oriented) model, Web-based System Manager enables users to perform administration tasks by manipulating icons representing objects in the system, as an alternative to learning and remembering complex commands.

Index

Special characters

/ file system
  messages 174
  root part 156
/dev directory 140
/dev/ipldevice file 173, 174
/etc/exclude.rootvg file 130
/etc/niminfo file
  recovering 237
/etc/objrepos directory
  after installing from system backup 140
/tmp file system
  free space in 32, 129
  messages 173, 174
  size during installation from backup 139
/tmp/disk.image file 172
/tmp/unix file 173
/tmp/vgdata/rootvg directory 144
/usr file system 278
  messages 172, 174
  troubleshooting when full 165
  user part 156
/usr/share file system 156

A

accessing an unbootable system 163
adapter_def resource 268
  defining 268
additional topics
  dynamic host configuration protocol
    interacting with 251
advanced configuration 231
  adding another network type 266
  backing up the NIM database 238
  booting diagnostics 239
  booting in maintenance mode 240
  creating additional interface attributes 232
  defining /usr vs. non-/usr SPOTs 233
  defining a heterogeneous network 265
  defining an lpp_source on CD-ROM vs. Disk 235
  establishing a default route 235
  establishing a static route 236
  recovering the /etc/niminfo file 237
  removing machines 231
  restoring the database and activating the
    master 238
  unconfiguring the master 239
advanced installation
  controlling the master or client 215
  group member
    excluding 220
    including 220
  logs
    viewing boot 227
    viewing configuration 227
    viewing installation 227
advanced installation *(continued)*
  lpp_source
    copying software 225
    maintaining software 225
    removing software 226
    running the check operation 226
  lppchk operation
    verifying installation 227
  machine groups
    adding new members 219
    defining 218
    removing members 219
  machines
    resetting 216
  resource group
    allocating 221
    defining 221
  resource groups
    defining default 221
  resource servers
    using clients 217
  SPOT
    listing software 222
    listing software updates by keyword 223
    maint operation 224
    managing software 222
    rebuilding network boot images 225
  standalone client
    listing software 222
    listing software updates by keyword 223
  standalone clients
    maint operation 224
    managing software 222
advanced tasks
  installation 215
all devices and kernels 75
allocate
  resource group 221
allocate operation 283
allocating resources
  resource groups 221
alt_disk_install 210
alternate disk installation 147
  cloning 148
  data access 151
  dynamic logical partitioning 152
  examples 152
  mksysb installation 147
  phased installation 149
  using SMIT 152
Alternate Disk Migration Installation 149
American Standard Code for Information
  Interchange 63
applying optional software
  definition of 87
  description 81
ASCII Installation Assistant 63
  introduction to tasks 63

© Copyright IBM Corp. 2002, 2004

ASCII procedures 81
ASCII terminals
  setting communications options 142
  setting display and keyboard options 52, 142
  setting options 52
ATM networks
  installing to clients 201
ATM Networks 263
  converting generic networks into 202
attributes
  if 232
  if1 232
  if2 232
  other_net_type 265

B

backup
  disk space 139
backup image 139
backup installation
  changing installation modes 143
backup, of system 127
  exclude files 130
  introduction 131
  mounting and unmounting file systems 32, 128
  procedure
    list files in a system image 138
    list information about filesets in a system image 138
    lsmksysb command 137
    lssavevg command 137
    prerequisites 32, 128
    preview information about a backup 137
    root volume group 32, 129
    verifying backup 135
    verifying system backups 137
    view the backup log 137
basic configuration
  creating basic installation resources 190
basic operations and configuration 189, 210
  adding a diskless or dataless client 207
  adding a standalone client 192
  clients and SPOT resources
    customizing 203
  configuring the master 190
  diskless and dataless clients
    configuring the master and creating resources 204
  diskless or dataless machine
    initializing and booting 209
    uninitializing 210
  mksysb install 197
  non-prompted install 200
  rte install 196
binary compatibility 57
boot logs
  view 227
boot resource 268
booting
  diagnostics 239
booting *(continued)*
  over router 249
booting the system
  problems with 163
  procedure 51, 61, 142
BOS installation
  cloning 34, 140
  system backup, from 139
BOS maintenance mode
  accessing 163
BOS menus
  installation and setting screen 45
  installation summary screen 44
  welcome screen 44
BOS run-time image
  source for 288
bos_inst operation 285
  ATM adapters 201
  paging space 250
  using the boot_client attribute 289
  using the force_push attribute 289
  using the preserve_res attribute 290
  using the set_bootlist attribute 290
bos.sysmgt.nim.master
  removing 247
bosboot
  troubleshooting 172, 174
    device problems 173
    space problems 172, 173
bosinst_data 268
  defined 268
  defining 269
  overview 268
bosinst.data file 71
  ACCEPT_LICENSES variable 74
  ALL_DEVICES_KERNELS variable 75
  ALT_DISK_INSTALL_BUNDLE variable 75
  ALWAYS_ALLOW variable 77
  BOSINST_DEBUG variable 74
  BOSINST_LANG variable 77
  BUNDLES variable 73
  CONNECTION variable 77
  CONSOLE variable 71
  COPYDIR variable 77
  CREATE_JFS2_FS variable 74
  CULTURAL_CONVENTION variable 77
  CUSTOMIZATION_FILE variable 72
  description 67
  DESKTOP variable 74
  DOC_SERVICES_BUNDLE variable 75
  DUMPDEVICE variable 77
  ENABLE_64BIT_KERNEL variable 74
  ERROR_EXIT variable 72
  EXISTING_SYSTEM_OVERWRITE variable 72
  FORCECOPY variable 77
  GRAPHICS_BUNDLE variable 75
  HARDWARE_DUMP variable 75
  HDISKNAME variable 77
  HTTP_SERVER_BUNDLE variable 75
  IMPORT_USER_VGS variable 74
  INSTALL_DEVICES_AND_UPDATES variable 74

bosinst.data file *(continued)*
INSTALL_METHOD variable 71
INSTALL_TYPE variable 73
INSTALL_X_IF_ADAPTER variable 72
KERBEROS_5S_BUNDLE variable 75
KEYBOARD variable 77
LOCATION variable 77
MESSAGES variable 77
NETSCAPE_BUNDLE variable 75
nonprompted BOS installation 71
PHYSICAL_LOCATION variable 76
PRIMARY variable 77
procedure for using 68
PROMPT variable 71
prompted mode, with 165
PVID variable 76
RECOVER_DEVICES variable 74
REMOVE_JAVA_118 variable 75
RM_INST_ROOTS variable 72
RUN_STARTUP variable 72
SAN_DISKID variable 76
SECONDARY variable 77
SERVER_BUNDLE variable 75
SIZE_MB variable 77
SIZEGB variable 77
TCB variable 73
bosinst.data sample file 78
bundles 81, 157
definition of 81
examples of 157
types of 157

C

CD or DVD
creating backups CD or DVD 132
ISO9660 format 132
CD/DVD
bootable 131
non-bootable 131
cdrecord 131
change operation 290
check operation 290
cleaning up failed software installation 35, 89
client
determining control 215
diskless or dataless
adding 207
operations 257
tasks performed from 231
client machine
verifying status of 195
client operations 231
clients
dataless 260
initializing 261
optional resources 260
required resources 260
defining 258
diskless 260
initializing 261
clients *(continued)*
diskless *(continued)*
optional resources 260
required resources 260
standalone 259
managing software 260
network booting 259
standalone, adding 192
cloning
backup, of system 34, 140
cloning using Alternate Disk Installation
Alternate Disk Installation
Alternate Disk Installation 17
commands
nim_clients_setup 212
nim_master_setup 212
nimclient 231
committing service updates
definition of 88
introduction 81
Common Criteria 48
communications
ASCII terminals options 52
setting options for ASCII terminals 142
compatibility 57
concepts
NIM 257
concurrency control 229
configuration assistant
configuring online documentation 63
documentation 63
web browser, installing 63
web server, installing 63
configuration logs
view 227
configuration tasks
basic 189
configuration, system 63
access remote resources 63
add license passwords 63
back up the system 63
change language environment 63
configure printer 63
create user accounts 63
date and time 63
exit and log in 63
install optional software 63
introduction to 63
list of tasks 63
set root password 63
control operations
definition of 282
control status
master or client 215
Controlled Access Protection Profile and Evaluation Assurance Level 4+ 48
Create and Install a Software Bundle
Create and Install a Software Bundle
software bundle 12, 21
creating software packages 156
cust operation 291

cust operation *(continued)*
asynchronous behavior 248
customizing BOS installation
bosinst.data file 67
procedure 68

D

data recovery
introduction 163
procedure 163
when system will not boot 163
dataless clients 260
initializing 261
managing software 262
resource
boot 260
dump 261
home 261
paging 261
resolv_conf 261
root 261
shared_home 261
SPOT 260
tmp 261
resources
optional 260
required 260
deallocate operation 292
debug
BOS installation 74
debug mode
installing BOS 44
NIM 337
NIM BOS installation 338
using a bosinst.data file for NIM BOS installation 339
default routes
creating 235
define
machine group 218
resource group 221
define operation 292
diag operation 239, 292
diagnostics
booting 239
loading from network
for diskless and dataless machines 240
directories
/dev 140
/etc/objrepos 140
/tmp/vgdata/rootvg 144
disk
specifying for BOS installation
CD-ROM, DVD-ROM or tape 53
system backup 144
disk space
messages 172
planning for master machine 190
diskless and dataless
managing software 262
diskless and dataless clients
creating resources to support 204
migrating 250
diskless and dataless tasks
booting diagnostics 239
diskless clients 260
initializing 261
managing software 262
resource
boot 260
dump 261
home 261
paging 261
resolv_conf 261
root 261
shared_home 261
SPOT 260
tmp 261
resources
optional 260
required 260
diskless or dataless client
adding 207
initializing and booting 209
uninitializing 210
disks, hard (fixed) 53
displays
setting options for ASCII terminals 52, 142
distributed resources 282
dkls_init operation 293
documentation
documentation library service 117
information center 117
online documentation 117
dtls_init operation 293
dump resource 269
defined 269
defining 269
overview 269
DVD-RAM
creating backups DVD-RAM and UDF 134
Universal Disk Format 134
dynamic host configuration protocol
interacting with 251

E

efix management 98
efix control file 109
emgr command 98
epkg command 109
installing and managing efixes 98
packaging efixes 109
electronic license agreements 45
emergency fix management
see also efix management 98
environment
changing language 54
epkg command 109
error conditions 159, 335

error messages
attempt to create bootable tape failed 174
check available disk space 172
error occurred during bosboot 172, 173
format 166
hard disks not accessed 171
hard disks not configured 171
invalid or no boot device specified 173
NIM 307
no disks are available 175
not enough file space to create: /tmp/disk.image 172
not enough file space to create: /tmp/unix 173
unable to expand file system /usr 172
error recovery 159, 335
exclude files from system backup 130
exclude_files 269
defined 269
defining 270
overview 269
exporting resources
number of hosts 248
eznim 187

F

fb_script 270
defined 270
defining 270
overview 270
FDDI
router 249
file systems
/
messages 174
root part 156
/tmp 139
free space in 32, 129
messages 173, 174
/usr
messages 172, 174
troubleshooting when full 165
usr part 156
/usr/share 156
mounting and unmounting 32, 128
files
/dev/ipldevice 173, 174
/etc/exclude.rootvg 130
/etc/niminfo 237
/tmp/disk.image 172
/tmp/unix 173
/usr 278
bosinst.data
examples of 78
explanation of 67
introduction 67
procedure for using 68
prompted mode, with 165
examples
bosinst.data 78
definition file for nimdef command 305
files *(continued)*
examples *(continued)*
script resource 305
image.data 68, 139
map 144
preserve.list 68
readme, viewing ix
sample 305
filesets 81, 156
fix_bundle 271
defined 271
defining 271
overview 271
fix_query operation 294
fixed disks 53
fixes
listing for SPOT 223
listing for standalone client 223

G

geninstall command 93, 155
graphical user interface
Easy Install
prerequisites 83
procedural overview 81
graphical user interfaces
Installation Assistant
introduction to tasks 63
group
resource
allocate 221
define 221
groups
establishing 301
machine 301
defining 218
resource 302
GUIs 63

H

hard disks 53
location codes of 53
specifying for CD/DVD-ROM or tape installation 53
specifying for system backup installation 144
unaccessible 171
unconfigured 171, 175
hardware
diagnostics
diskless and dataless 239
heterogeneous networks
defining 265
home resource 271
defined 271
defining 271
overview 271

I

if attribute
  creating additional 232
if1 attribute 232
if2 attribute 232
image_data 272
  defined 272
  defining 272
  overview 272
image.data file 68, 139
information center 117
initiating
  BOOTP request on diskless and dataless systems 343
  BOOTP request on standalone machine 343
install_all
  SMIT fast path 85
install_all_updates command 86
installation
  alternate disk 147
  change method of 53
  verifying with lppchk operation 227
installation (BOS)
  customizing with the bosinst.data file 67
Installation Assistant 64
  introduction to tasks 63
installation images
  mksysb configuration on backup tape 159
installation logs
  view 227
installation methods
  definition of 53
  specifying 53
installation resources
  creating 190
installation screens (BOS)
  Change Disk(s) Where You Want to Install 54, 144
  Installing Base Operating System 54, 62
  specifying installation language 53, 61
  specifying system console 53, 61, 143
  System Backup Installation and Settings (BOS) 144
installation tasks
  advanced 215
installation, nonprompted
  specifying with the bosinst.data file 67
installing AIX
  nonprompted new and complete overwrite installation 10
  using the CD to manually install a standalone system
    new and complete overwrite installation 3
installing BOS
  debug mode 44
  new and complete overwrite 51
  preservation 51
installing BOS from CD or DVD-ROM
  procedure
    specifying destination disk 53
installing BOS from CD or tape
  procedure
    booting (starting) the system 142
    initiating the installation 142
installing BOS from CD-ROM
  procedure
    changing language environment 54
    prerequisites for 51
installing BOS from CD-ROM or tape
  troubleshooting 166
installing BOS from CD, DVD or tape
  customizing the installation 67
installing BOS from CD/DVD-ROM
  procedure
    booting (starting) the system 51
    initiating the installation 51
    introduction to 51
installing BOS from CD/DVD-ROM or tape
  procedure
    booting (starting) the system 61
    initiating the installation 61
    introduction to 59, 60
installing BOS from DVD-ROM
  procedure
    prerequisites for 51
installing BOS from system backup
  introduction to 139
  procedure 140
  resolving reported problems 160
  source system 139
  target system 139
  troubleshooting 159
installing BOS using Web-based System Manager
  network installation 140
  system backup 140
installing optional software
  applying 81, 87
  cleaning up failed installation 35, 89
  committing 81, 88
  introduction to 81
  prerequisites 83
  procedural overview 81
  procedure (graphical interface) 84
  procedure (SMIT) 85
    accessing SMIT 85
    status messages 85
  rejecting 81, 88
  removing 81, 88
  selection criteria 82
  software licenses 82
  software packaging 82
  status messages (SMIT) 85
  troubleshooting 35, 89, 172
installp format
  creating software packages 156
installp_bundle 272
  defined 272
  defining 272
  overview 272
InstallShield MultiPlatform 93
  installing a package 93
  response files 95
  response files and NIM 97
  silent installation 95, 97
  uninstalling a package 94

interface attribute 232
creating additional 232
introduction to NIM
dataless
definition of 183
diskless
definition of 183
network objects
definition of 183
resources
definition of 183
standalone
definition of 183
IP address
determining 264
IPL ROM
emulation
creating media 346
determining if needed 343
ISMP
see InstallShield MultiPlatform 93

K

keyboards
setting options for ASCII terminals 52, 142

L

language environment
changing 54
setting 54
license acceptance 74
license agreements 45
licensed programs 81
packaging of 82
selection criteria for installation 82
licenses, software
function of 82
listing fixes
installed on a SPOT 223
installed on a standalone client 223
listing information
software in SPOT 222
software in standalone client 222
listing software updates
installed on a SPOT 223
installed on a standalone client 223
locale 54
location codes, of the hard disk 53
logical volumes
accessing 163
introduction 163
procedure 163
logs
boot
viewing 227
configuration
viewing 227
installation
view 227
lpp_source 273
copying software 225
defined 273
defining 274
maintaining software 225
overview 273
removing software 226
running the check operation 226
lppchk operation 295
lppmgr command 90
lppmgr operation 295
lsmksysb command 137
lssavevg command 137

M

machine
diskless or dataless
initializing and booting 209
uninitializing 210
machine groups
defining 218, 301
operations 219, 220
machine operations
defined 282
list of 282
machines 257
maint operation 224, 295
asynchronous behavior 248
maintain software in SPOT 224
maintain software on standalone clients 224
maint_boot operation 296
maintaining optional software
applying 87
committing 88
concepts defined 87, 91
rejecting 88
removing 88
maintenance mode
accessing (BOS) 163
using 241
managing software
clients
standalone 260
SPOT 222
standalone clients 222
map files 144
master
activating 238
backing up NIM database 238
configuring 190
disabling push permissions 216
managing
activate the master 238
back up NIM database 238
restore NIM database 238
removing master fileset 247
unconfiguring 239
master fileset
removing 247

messages
NIM error 307
NIM warning 307
system and error 166
migrating
clients
diskless and dataless 250
NIM SPOTs 250
migrating installing AIX
migrating a system to the AIX 5.2
Migration 7
migrating to new version
procedure
prerequisites for 60
migration installation
definition of 43
mkcd command 132
mkinstallp command 156
mkisofs 131
mksysb 274
alternate disk installation 210
backup images on CD/DVD 131
cloning 34, 140
defined 274
defining 274
installation from 139
overview 274
resolving reported problems 160
source for BOS run-time image 288
system backup tapes 159
troubleshooting installation from 159
mksysb install
performing 197
monitors
setting options for ASCII terminals 52, 142

N

name resolution 249
network
heterogeneous
defining 265
network boot
procedures for 343
network booting
clients
standalone 259
network objects
managing
creating interface attributes 232
establishing a route between networks 236
network types
supported 263
networks
ATM 201
defining 263
defining heterogeneous 265
NIM 263
new and complete overwrite installation 51
definition of 43
NIM
alternate disk installation 210
client requests, tuning 255
configuration
basic 189
configure using EZNIM 20
error messages 307
introduction 183
Kerberos 5 229
Kerberos authentication 228
machines 257
networks 263
nimesis daemon 255
operations
basic 189
overview 183
response files and InstallShield MultiPlatform products 97
SPOTs
migrating 250
warning messages 307
NIM attributes 265
NIM concepts 257
NIM database
backing up 238
restoring 238
NIM environment 257
defining
using the nimdef command 251
NIM eznim 187
NIM groups 301
NIM networks
defining 263
IP address
determining 264
routes 265
types
supported 263
NIM object definitions
name requirements 251
NIM objects
definitions
name requirements 251
NIM operations 239
allocate 283
alternate disk installation 186
alternate disk migration installation 186
bos_inst 285
change 290
check 290
cust 291
deallocate 292
define 292
diag 292
diagnostics, booting 185
diskless and dataless clients, adding 186
dkls_init 293
dtls_init 293
fix_query 294
lppchk 295
lppmgr 295

NIM operations *(continued)*
maint 295
maint_boot 296
maintenance mode, booting 185
reboot 296
remove 297
reset 297
resources 185
select 297
showlog 298
showres 298
software, customizing 185
software, removing 185
sync_roots 299
unconfig 300
update 300
NIM output
suppressing 248
NIM resources 267
NIM routes 265
nim_script resource 275
nimadapters command 242
nimclient command 231
nimdef command
NIM environment
defining 251
sample definition file 305
non-prompted install
performing a 200
nonprompted installation
changing to prompted 143
nonprompted installation, customizing for 67
nonprompted mode
overriding 165
nonroot volume group
definition of 127

O

online documentation
installing 125
operations
allocate 283
basic 189
BOS run-time image
selecting source 288
bos_inst 285
change 290
check 290
cust 291
deallocate 292
define 292
diag 292
dkls_init 293
dtls_init 293
excluding a group member 220
fix_query 294
including a group member 220
list of 282
lppchk 295
lppmgr 295
operations *(continued)*
maint 295
maint_boot 296
NIM
diag 239
machine 282
performed from client 231
on client machines 257
performing 282
reboot 296
remove 297
reset 297
select 297
showlog 298
showres 298
sync_roots 299
types 282
unconfig 300
update 300
optional software
cleaning up failed installation of
introduction 35, 89
definition of 81
options
language environment 54
nonprompted installation
specifying with the bosinst.data file 67
setting communications (ASCII) 52, 142
setting display (ASCII) 52, 142
setting monitor (ASCII) 52, 142
specifying installation disk (BOS)
CD-ROM, DVD-ROM or tape 53
system backup 144
specifying installation language 53, 61
specifying installation method (BOS) 53
specifying system console 53, 61, 143
system configuration 63
other_net_type attribute 265
output, NIM
suppressing 248
overview
NIM 183, 257

P

package
definition of 81
package formats
installp 155
software filesets 156
ISMP 155
RPM 155
paging
defined 275
overview 275
paging resource 275
defining 276
port conflicts
nimesis daemon 339
post_migration command 58
pre_migration command 58

preservation installation 51
definition of 43
preserve_res 290
preserve.list file 68
problems, recovering from 159, 335
procedures
accessing BOS maintenance 163
advanced configuration 231
adding another network type 266
backup up the NIM database 238
booting diagnostics 239
booting in maintenance mode 240
creating additional interface attributes 232
defining a heterogeneous network 265
defining an lpp_source on CD-ROM vs. Disk 235
establishing a default route 235
establishing a static route 236
recovering the /etc/niminfo file 237
removing machines from the NIM environment 231
restoring the database and activating the master 238
unconfiguring the master 239
advanced installation
adding new members 219
allocating a resource group 221
controlling the master or client 215
copying software to an lpp_source 225
defining a machine group 218
defining a resource group 221
defining default resource groups 221
excluding a group member 220
including a group member 220
listing software in a SPOT 222
listing software on a standalone client 222
listing software updates by keyword 223
maintaining software in an lpp_source 225
maintaining software on a SPOT 224
removing members 219
removing software from an lpp_source 226
resetting machines 216
running the NIM check operation 226
standalone clients and SPOT resources, managing software 222
using clients as resource servers 217
verifying installation with lppchk operation 227
viewing logs 227
advanced installation, managing software
rebuilding network boot images for a SPOT 225
basic configuration
creating basic installation resources 190
basic operations and configuration
adding a diskless or dataless client 207
adding a standalone client 192
clients and SPOT resources, customizing 203
configuring the master 190
creating resources to support diskless/dataless clients 204
diskless or dataless machine, initializing and booting 209
diskless or dataless machine, uninitializing 210
procedures *(continued)*
basic operations and configuration *(continued)*
mksysb install, performing 197
non-prompted install 200
rte install, performing an 196
customizing BOS installation 68
defining /usr vs. non-/usr SPOTs 233
identifying boot device 174
installing BOS from CD/DVD-ROM 51
installing BOS from CD/DVD-ROM or tape 59, 60
installing BOS from system backup 140
installing optional software 81
resizing /tmp 173
root volume group, backing up 32, 129
troubleshooting a mksysb installation 159, 160
troubleshooting full /usr file system 165
unlocking the root volume group 172
user volume group, backing up 135
verifying system backup 137
viewing readme files ix
product identification, optional software 82
prompted installation 144
help information 144
prompted mode
changing to (BOS) 165
push permissions
master
disabling 216

R

readme files, viewing ix
reboot operation 296
recover devices 74
recovery 159, 335
rejecting optional software
definition of 88
introduction 81
remove operation 297
removing optional software
definition of 88
introduction 81
reset operation 297
resolv_conf
defined 276
defining 276
overview 276
resolv_conf resource 276
resource
boot
diskless/dataless 260
dump
diskless/dataless 261
home
diskless/dataless 261
paging
diskless/dataless 261
resolv_conf
diskless/dataless 261
root
diskless/dataless 261

resource *(continued)*
shared_home
diskless/dataless 261
SPOT
diskless/dataless 260
tmp
diskless/dataless 261
resource group
allocate 221
define 221
resource groups
allocating resources 221
defining 302
defining default 221
resources 267
adapter_def 268
defining 268
AIX Version 4.2 spots
network boot images 280
AIX Version 4.3 or later spots 281
boot 268
bosinst_data 268
defining 269
clients and SPOT
customizing 203
distributed 282
dump 269
defining 269
exclude_files 269
defining 270
exporting
number of hosts 248
fb_script 270
defining 270
fix_bundle 271
defining 271
home 271
defining 271
image_data 272
defining 272
installp_bundle 272
defining 272
list of 267
lpp_source 273
defining 274
mksysb 274
defining 274
nim_script 275
operations
list of 267
paging 275
defining 276
resolv_conf 276
defining 276
root 277
defining 277
script 277
defining 277
shared product object tree 278
shared_home 278
defining 278
resources *(continued)*
SPOT 278
defining 281
tmp 281
defining 281
resources, NIM
reducing space requirements 249
root directory
creating file resources 251
root resource 277
defined 277
defining 277
root volume group (rootvg)
backing up 32, 129
definition of 127
unlocking 172
router
booting over 249
routes 265
creating a default 235
creating a static 236
rte
source for BOS run-time image 288
rte install
performing 196
run-time image 288

S

sample files 305
bosinst.data 78
nimdef command
definition file 305
script resource 305
savevg command 135
screens
Access a Root Volume Group (BOS) 165
Change Disk(s) Where You Want to Install (BOS) 54, 144
Installing Base Operating System (BOS) 54, 62
Maintenance (BOS) 165
System Backup Installation and Settings 144
Volume Group Information (BOS) 165
Welcome to Base Operating System Installation and Maintenance (BOS) 165
script 277
defined 277
defining 277
overview 277
script resource
sample file 305
secondary adapter support 242
security evaluation technology
see Controlled Access Protection Profile and Evaluation Assurance Level 4+ 48
select operation 297
service updates
committing 88
rejecting 88
removing 88
shared product object tree 278

shared_home
  defining 278
  overview 278
shared_home resource 278
  defined 278
showlog operation 298
showres operation 298
SMIT fast path
  alt_clone 152
  alt_mksysb 152
  assist 64
  cleanup_software 93
  commit 88
  compare_report 91
  eznim 187
  fixtolist_compare 92
  install_all 85
  install_latest 85
  install_update 64
  installed_license 82
  instofix_compare 91
  instolist_compare 92
  license_on_media 82
  maintain_software 90
  reject 88
  remove 89
  rename_software 92
  service_software 91
  update_all 85
  update_by_fix 85
SMIT interfaces
  Custom Install path
    description of 85
  Easy Install path
    description of 85
    prerequisites 83
    procedural overview 81
  Installation Assistant 63
SMIT procedures
  installing optional software 81, 85
software
  listing for SPOT 222
  listing for standalone client 222
software bundles 81, 157
  definition of 81, 157
  examples of 157
software filesets
  definition of 81
software licenses
  function of 82
software packages
  definition of 81
software packaging
  root part 156
  share part 156
  user part 156
software products 82
  applying 87
  bundle, definition of 81
  bundle, examples of 157
  committing 88
software products *(continued)*
  fileset, definition of 81
  identification of 82
  licensed program, definition of 82
  package, definition of 81
  packaging of 82
  rejecting 88
  removing 88
  selection criteria for installation 82
software service management 91
  clean up software images 92
  comparison reports 91
  rename software images 92
software updates
  listing for SPOT 223
  listing for standalone client 223
software, optional
  definition of 81
source system 139
SPOT
  defined 278
  defining 281
  maintaining software 224
  managing software 222
  overview 278
  resources 234
  source for BOS run-time image 288
  updating 255
standalone client operations 231
standalone clients 231, 259
  adding 192
  maintaining software 224
  managing software 222, 260
  network booting 259
standalone machines
  determining control 215
starting the system 51, 61, 142
static routes
  creating 236
suppressing
  NIM output 248
sync_roots operation 299
system
  booting (starting) 51, 61, 142
  configuring 63
system backup 127
system backup to tape 32
system backup, BOS installation from 139
system bundles 157
system configuration 63
System Management Interface Tool 63
system messages 166
system settings
  changing during BOS installation 53

T

tape
  creating bootable, troubleshooting 174
  creating bootable,troubleshooting 174
tapeblksz 159

target disk
specifying for CD/DVD-ROM or tape installation 53
specifying for system backup installation 144
target system 139
tasks
bos, installing 185
configuring NIM environment 185
installation
advanced 215
terminals (ASCII)
setting communications options 52, 142
tmp resource 281
defined 281
defining 281
overview 281
troubleshooting 166
boot problems 163
introduction 163
procedure 163
cleaning up failed optional software installation
introduction 35, 89
full /usr file system 165
installation from system backup (mksysb) 159
resolving reported problems 160
introduction 159, 335
network boot problem 335
client and server, establishing network communication 335
obtaining the boot image from the server 335
running the boot image on the client 336
nonprompted mode, overriding 165
producing debug output 337
producing debug output from
a network boot image 338
BOS install program 338
prompted mode, changing to 165
troubleshooting procedures
recovering /etc/niminfo file 237

U

unconfig operation 300
unconfiguring
master 239
update operation 300
update_all
install_all_updates command 86
SMIT fast path 85
updates, service
explanation of 81
user volume group
backing up 135
definition of 127

V

volume groups
accessing 163
introduction 163
procedure 163
backing up 32, 129, 135
volume groups *(continued)*
nonroot 127
root 127, 172
user 127

W

warning messages
NIM 307
Web-based System Manager
BOS installation 140

Readers' Comments — We'd Like to Hear from You

AIX 5L Version 5.2
Installation Guide and Reference

Publication No. SC23-4389-04

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC23-4389-04

IBM

Cut or Fold
Along Line

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

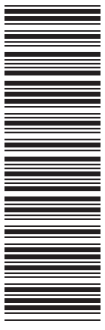
IBM Corporation
Information Development
Department H6DS-905-6C006
11501 Burnet Road
Austin, TX 78758-3493

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Part Number: 16R1762

Printed in U.S.A.



(1P) P/N: 16R1762

SC23-4389-04

Spine information:

AIX 5L Version 5.2

AIX 5L Version 5.2: Installation Guide and Reference

SC23-4389-04