



Sun Java™ System

Sun Java Enterprise System Upgrade Guide for HP-UX

2005Q4

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819-4460-10

Copyright © 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

THIS PRODUCT CONTAINS CONFIDENTIAL INFORMATION AND TRADE SECRETS OF SUN MICROSYSTEMS, INC. USE, DISCLOSURE OR REPRODUCTION IS PROHIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF SUN MICROSYSTEMS, INC.

U.S. Government Rights - Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, JDK, Java Naming and Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, the Duke logo, the Java Coffee Cup logo, the Solaris logo, the SunTone Certified logo and the Sun ONE logo are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon architecture developed by Sun Microsystems, Inc.

Legato and the Legato logo are registered trademarks, and Legato NetWorker, are trademarks or registered trademarks of Legato Systems, Inc. The Netscape Communications Corp logo is a trademark or registered trademark of Netscape Communications Corporation.

The OPEN LOOK and Sun(TM) Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this service manual are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright © 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuels relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs des brevets américains listés à l'adresse <http://www.sun.com/patents> et un ou des brevets supplémentaires ou des applications de brevet en attente aux Etats - Unis et dans les autres pays.

CE PRODUIT CONTIENT DES INFORMATIONS CONFIDENTIELLES ET DES SECRETS COMMERCIAUX DE SUN MICROSYSTEMS, INC. SON UTILISATION, SA DIVULGATION ET SA REPRODUCTION SONT INTERDITES SANS L'AUTORISATION EXPRESSE, ECRITE ET PREALABLE DE SUN MICROSYSTEMS, INC.

Cette distribution peut comprendre des composants développés par des tierces parties.

Des parties de ce produit peuvent être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Java, Solaris, JDK, Java Naming and Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, le logo Duke, le logo Java Coffee Cup, le logo Solaris, le logo SunTone Certified et le logo Sun[tm] ONE sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

Legato, le logo Legato, et Legato NetWorker sont des marques de fabrique ou des marques déposées de Legato Systems, Inc. Le logo Netscape Communications Corp est une marque de fabrique ou une marque déposée de Netscape Communications Corporation.

L'interface d'utilisation graphique OPEN LOOK et Sun(TM) a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de ce manuel d'entretien et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes biologiques et chimiques ou du nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.

Contents

List of Tables	11
Preface	15
Who Should Use This Book	15
How This Book Is Organized	16
Typographic Conventions	17
Shell Prompts	17
Related Documentation	18
Books in This Documentation Set	18
Accessing Sun Documentation	20
Third-Party Web Sites	20
Contacting Sun Technical Support	21
Sun Welcomes Your Comments	21
Chapter 1 Planning for Upgrades	23
Java ES 2005Q4 (Release 4) Components	23
Release 4 Product Components	24
Release 4 Shared Components	24
About Java ES Upgrades	26
Product Component Upgrades	26
Shared Component Upgrades	27
Upgrade Technologies	27
Operating System Issues	28
Required Operating System Patches	28
Upgrade Planning	28
What is an Upgrade Plan?	29
Upgrade Paths	30

Upgrade Dependencies	31
Upgrade All	31
Java ES Component Dependencies	32
Dependencies On Shared Components	32
Shared Component Upgrade Guidelines	35
Dependencies On Product Components	36
Multi-instance Upgrades	41
General Sequencing Guidelines	41
Chapter 2 Upgrading Java ES Shared Components	45
Shared Component Upgrade Overview	45
About Your Upgrade Plan	45
Technologies for Upgrading Shared Components	46
General Upgrade Procedure	47
Upgrading Shared Components from Java ES Release 3	48
Upgrading Shared Components by Applying Patches	49
Upgrading Components by Replacing Packages	51
Upgrade Strategy for Replacement of Packages	51
Upgrade Path	51
Package Versions	52
Package Locations	52
Procedures for Replacement of Packages	52
Special Instructions	52
Packages for HP-UX Platforms	52
Upgrading Packages for HP-UX Platforms	56
Upgrading J2SE for Java ES Release 4	58
Upgrading J2SE on HP-UX Platform	58
Chapter 3 Directory Server and Administration Server	61
Overview of Directory Server and Administration Server Upgrades	61
About Java ES Release 4	62
Java ES Release 4 Upgrade Roadmap	62
Directory Server and Administration Server Data	62
Compatibility Issues	63
Dependencies	63
Upgrading Directory Server and Administration Server from Java ES Release 3	64
Introduction	64
Directory and Administration Server Upgrade	64
Pre-Upgrade Tasks	65
Upgrading Directory Server and Administration Server	66
Upgrade Procedure	67

Verifying the Upgrade	68
Post-Upgrade Tasks	69
Rolling Back the Upgrade	69
Chapter 4 Directory Proxy Server	71
Overview of Directory Proxy Server Upgrades	71
About Java ES Release 4	72
Java ES Release 4 Upgrade Roadmap	72
Directory Proxy Server Data	72
Compatibility Issues	73
Dependencies	73
Upgrading Directory Proxy Server from Java ES Release 3	73
Introduction	73
Directory Proxy Server Upgrade	74
Pre-Upgrade Tasks	74
Upgrading Directory Proxy Server	76
Verifying the Upgrade	78
Post-Upgrade Tasks	78
Rolling Back the Upgrade	78
Multiple Instance Upgrades	79
Chapter 5 Web Server	81
Overview of Web Server Upgrades	81
About Java ES Release 4 Web Server	82
Web Server Upgrade Roadmap	82
Compatibility Issues	82
Web Server Dependencies	82
Upgrading Web Server from Java ES Release 3	82
Introduction	83
Web Server Upgrade	83
Chapter 6 Message Queue	87
Overview of Message Queue Upgrades	87
About Java ES Release 4 Message Queue	88
Message Queue Upgrade Roadmap	88
Message Queue Data	88
Compatibility Issues	89
Protocol Compatibility	90
Broker Compatibility	90
Administered Object Compatibility	90
Administration Tool Compatibility	90
Client Compatibility	90

Message Queue Dependencies	91
Upgrading Message Queue from Java ES Release 3	91
Introduction	91
Message Queue Upgrade	92
Pre-Upgrade Tasks	92
Upgrading Message Queue	93
Verifying the Message Queue Upgrade	94
Post-Upgrade Tasks	95
Rolling Back the Upgrade	95
Multiple Instance Upgrades	96
Chapter 7 High Availability Session Store	97
Overview of HADB Upgrades	98
About Java ES Release 4 HADB	98
HADB Upgrade Roadmap	98
HADB Data	98
Compatibility Issues	99
HADB Dependencies	99
Upgrading HADB from Java ES Release 3	99
Introduction	99
HADB Upgrade	100
Pre-Upgrade Tasks	100
Upgrading HADB	101
Verifying the Upgrade	104
Post-UpgradeTasks	105
Rolling Back the Upgrade	106
Chapter 8 Application Server	107
Overview of Application Server Upgrades	107
About Java ES Release 4 Application Server	108
Application Server Upgrade Roadmap	108
Application Server Data	108
Compatibility Issues	109
Application Server Dependencies	109
Upgrading Application Server from Java ES Release 3	110
Introduction	110
Application Server Upgrade	110
Pre-Upgrade Tasks	111
Upgrading Application Server	112
Verifying the Upgrade	113
Post-Upgrade Tasks	114
Rolling Back the Upgrade	114

Chapter 9 Access Manager	115
Overview of Access Manager Upgrades	115
About Java ES Release 4 Access Manager	116
Access Manager Upgrade Roadmap	117
Access Manager Data	117
Compatibility Issues	118
Access Manager Dependencies	119
Upgrading Access Manager from Java ES Release 3	120
Introduction	120
Full Access Manager Upgrade	121
Pre-Upgrade Tasks	121
Upgrading Access Manager	124
Verifying the Access Manager Upgrade	130
Post-Upgrade Tasks	131
Rolling Back the Upgrade	131
Multiple Instance Upgrades: Release 3 and Release 4 Co-existence	131
Access Manager SDK-only Upgrades	132
Pre-Upgrade Tasks	133
Upgrading Access Manager SDK	133
Verifying the Access Manager SDK Upgrade	134
Upgrade Rollback	134
Chapter 10 Directory Preparation Tool	135
Overview of Directory Preparation Tool Upgrades	135
About Java ES Release 4 Directory Preparation Tool	136
Directory Preparation Tool Upgrade Roadmap	136
Directory Preparation Tool Data	136
Compatibility Issues	137
Directory Preparation Tool Dependencies	137
Upgrading Directory Preparation Tool from Java ES Release 3	137
Introduction	137
Directory Preparation Tool Upgrade	138
Pre-Upgrade Tasks	138
Upgrade Procedure	140
Verifying the Upgrade	140
Post-Upgrade Tasks	140
Rolling Back the Upgrade	141
Chapter 11 Messaging Server	143
Overview of Messaging Server Upgrades	143
About Java ES Release 4 Messaging Server	144
Messaging Server Upgrade Roadmap	144
Messaging Server Data	144

Compatibility Issues	145
Messaging Server Dependencies	145
Upgrading Messaging Server from Java ES Release 3	146
Introduction	146
Messaging Server Upgrade	147
Pre-Upgrade Tasks	147
Upgrading Messaging Server	148
Chapter 12 Calendar Server	153
Overview of Calendar Server Upgrades	153
About Java ES Release 4 Calendar Server	154
Calendar Server Upgrade Roadmap	154
Calendar Server Data	154
Compatibility Issues	154
Calendar Server Dependencies	155
Upgrading Calendar Server from Java ES Release 3	155
Introduction	156
Calendar Server Upgrade	156
Pre-Upgrade Tasks	157
Upgrading Calendar Server	158
Verifying the Upgrade	159
Post-Upgrade Tasks	159
Rolling Back the Upgrade	159
Chapter 13 Communications Express	161
Overview of Communications Express Upgrades	161
About Java ES Release 4 Communications Express	162
Communications Express Upgrade Roadmap	162
Communications Express Data	162
Compatibility Issues	163
Communications Express Dependencies	163
Upgrading Communications Express from Java ES Release 3	164
Introduction	165
Communications Express Upgrade	166
Pre-Upgrade Tasks	166
Upgrading Communications Express	168
Verifying the Upgrade	170
Post-Upgrade Tasks	170
Rolling Back the Upgrade	170
Multiple Instance Upgrades	171

Chapter 14 Instant Messaging	173
Overview of Instant Messaging Upgrades	173
About Java ES Release 4 Instant Messaging	174
Instant Messaging Upgrade Roadmap	174
Instant Messaging Data	174
Compatibility Issues	175
Instant Messaging Dependencies	175
Upgrading Instant Messaging from Java ES Release 3	175
Introduction	176
Instant Messaging Upgrade	176
Pre-Upgrade Tasks	176
Upgrading Instant Messaging	178
Upgrade Procedure	179
Verifying the Upgrade	179
Post-Upgrade Tasks	179
Rolling Back the Upgrade	180
Multiple Instance Upgrades	180
Chapter 15 Portal Server	181
Overview of Portal Server Upgrades	181
About Java ES Release 4 Portal Server	182
Portal Server Upgrade Roadmap	182
Portal Server Data	182
Compatibility Issues	183
Portal Server Dependencies	184
Upgrading Portal Server from Java ES Release 3	184
Introduction	185
Portal Server Upgrade	185
Pre-Upgrade Tasks	185
Upgrading Portal Server	187
Verifying the Upgrade	189
Post-Upgrade Tasks	189
Rolling Back the Upgrade	189
Multiple Instance Upgrades	190
Chapter 16 Portal Server Remote Access	191
Overview of Portal Server Secure Remote Access Upgrades	191
About Java ES Release 4 Portal Server Secure Remote Access	192
Portal Server Secure Remote Access Upgrade Roadmap	192
Portal Server Secure Remote Access Data	192
Compatibility Issues	193
Portal Server Secure Remote Access Dependencies	193
Upgrading Portal Server Secure Remote Access from Java ES Release 3	194

Introduction	194
Portal Server Secure Remote Access Upgrade	195
Pre-Upgrade Tasks	195
Upgrading Portal Server Secure Remote Access	196
Verifying the Upgrade	198
Post-Upgrade Tasks	198
Rolling Back the Upgrade	198
Multiple Instance Upgrades	199
Chapter 17 Delegated Administrator	201
Overview of Delegated Administrator Upgrades	201
About Java ES Release 4 Delegated Administrator	202
Delegated Administrator Upgrade Roadmap	202
Delegated Administrator Data	202
Compatibility Issues	203
Delegated Administrator Dependencies	203
Upgrading Delegated Administrator from Java ES Release 3	204
Introduction	204
Delegated Administrator Upgrade	205
Pre-Upgrade Tasks	205
Upgrading Delegated Administrator	207
Verifying the Upgrade	209
Post-Upgrade Tasks	209
Rolling Back the Upgrade	209
Appendix A Java System Release Contents	211
Java ES 2005Q1 (Release 3)	211
Release 3 Installer Selectable Components	211
Release 3 Shared Components	213
Java ES 2005Q4 (Release 4)	214
Release 4 Installer-Selectable Components	214
Release 4 Shared Components	216
Index	219

List of Tables

Table 1	Books in the Java Enterprise System Documentation Set	19
Table 1-1	Java ES Release 4 Product Components	24
Table 1-2	Java ES Release 4 Shared Components	25
Table 1-3	Phases in the Upgrade Process	29
Table 1-4	Upgrade Paths to Java ES 2005Q4 (Release 4)	30
Table 1-5	Upgrade All	32
Table 1-6	Shared Component Dependencies of Java ES Release 4 Product Components	33
Table 1-7	Java ES Product Component Dependencies	37
Table 2-1	Upgrade Technologies to Upgrade Shared Components from Java ES Release 3	48
Table 2-2	Package Versions for Upgrading Shared Components on HP-UX	53
Table 3-1	Upgrade Paths to Java ES Release 4 Directory Server and Administration Server 2005Q4	62
Table 3-2	Directory Server and Administration Server Data Usage	63
Table 3-3	Directory Server Version Outputs	65
Table 3-4	Patches to Upgrade Directory Server and Administration Server	67
Table 4-1	Upgrade Paths to Java ES Release 4: Sun Java System Directory Proxy Server 5 2005Q4	72
Table 4-2	Directory Proxy Server Data Usage	73
Table 4-3	Directory Proxy Server Version Verification Outputs	75
Table 4-4	Patches to Upgrade Directory Proxy Server	76
Table 5-1	Upgrade Paths to Java ES Release 4 Web Server 6.1 2005Q4 SP5 Java ES	82
Table 5-2	Web Server Version Verification Outputs	84
Table 5-3	Patches to Upgrade Web Server	85
Table 6-1	Upgrade Paths to Java ES Release 4 Message Queue 3.6 SP3 2005Q4	88
Table 6-2	Message Queue Data Usage	89

Table 6-3	Message Queue Version Verification Outputs	93
Table 7-1	Upgrade Paths to Java ES Release 4: HADB 4.4.2 2005Q4	98
Table 7-2	HADB Data Usage	99
Table 7-3	HADB Version Verification Outputs	101
Table 7-4	Package Versions for Upgrading HADB on Solaris Platforms	102
Table 8-1	Upgrade Paths to Java ES Release 4: Sun Java System Application Server Enterprise Edition 8.1 2005Q4	108
Table 8-2	Application Server Data Usage	108
Table 8-3	Application Server Version Verification Outputs	111
Table 8-4	Patches to Upgrade Application Server	112
Table 9-1	Upgrade Paths to Java ES Release 4: Sun Java System Access Manager 7 2005Q4 ..	117
Table 9-2	Access Manager Data Usage	117
Table 9-3	Access Manager Version Verification Outputs	122
Table 9-4	Configuration Parameters: ampre70upgrade	125
Table 9-5	Patches to Upgrade Access Manager Mobile Access software	126
Table 9-6	Patches to Upgrade Access Manager Mobile Access software	128
Table 9-7	Access Manager Configuration Parameters: amupgrade	129
Table 10-1	Upgrade Paths to Java ES Release 4 Directory Preparation Tool 2005Q4	136
Table 10-2	Directory Preparation Tool Data Usage	136
Table 10-3	Patches to Upgrade Directory Preparation Tool	139
Table 11-1	Upgrade Paths to Java ES Release 4 Messaging Server 6 2005Q4	144
Table 11-2	Messaging Server Data Usage	144
Table 11-3	Messaging Server Version Verification Outputs	147
Table 11-4	Patches to Upgrade Messaging Server	149
Table 12-1	Upgrade Paths to Java ES Release 4 Calendar Server 6 2005Q4	154
Table 12-2	Calendar Server Data Usage	154
Table 12-3	Calendar Server Version Verification Outputs	157
Table 12-4	Patches Upgrade Calendar Server	158
Table 13-1	Upgrade Paths to Java ES Release 4: Sun Java System Communications Express 6 2005Q4	162
Table 13-2	Communications Express Data Usage	163
Table 13-3	Communications Express Version Verification Outputs	166
Table 13-4	Patches to Upgrade Communications Express	168
Table 14-1	Upgrade Paths to Java ES Release 4: Sun Java System Instant Messaging 7 2005Q4	174
Table 14-2	Instant Message Data Usage	174
Table 14-3	Instant Messaging Version Verification Outputs	177
Table 14-4	Patches to Upgrade Instant Messaging	178

Table 15-1	Upgrade Paths to Java ES Release 4 Portal Server 6 2005Q4	182
Table 15-2	Portal Server Data Usage	183
Table 15-3	Portal Server Version Verification Outputs	186
Table 15-4	Patches to Upgrade Portal Server	187
Table 16-1	Upgrade Paths to Java ES Release 4: Sun Java System Portal Server Secure Remote Access 6 2005Q4	192
Table 16-2	Portal Server Secure Remote Access Data Usage	193
Table 16-3	Portal Server Version Verification Outputs	195
Table 16-4	Patches to Upgrade Portal Server Secure Remote Access	197
Table 17-1	Upgrade Paths to Java ES Release 4: Sun Java System Communication Services Delegated Administrator 6 2005Q4	202
Table 17-2	Delegated Administrator Data Usage	202
Table 17-3	Delegated Administrator Version Verification Outputs	205
Table 17-4	Patches to Upgrade Delegated Administrator	207

Preface

The *Sun Java™ Enterprise System Upgrade Guide for HP-UX* contains the information you need to upgrade Sun Java™ Enterprise System (Java ES) software in a HP-UX Operating System. The Guide covers upgrade from Java ES 2005Q1 (Release 3) to Java ES 2005Q4 (Release 4).

This preface contains the following sections:

- “Who Should Use This Book” on page 15
- “How This Book Is Organized” on page 16
- “Typographic Conventions” on page 17
- “Related Documentation” on page 18
- “Accessing Sun Documentation” on page 20
- “Third-Party Web Sites” on page 20
- “Contacting Sun Technical Support” on page 21
- “Sun Welcomes Your Comments” on page 21

Who Should Use This Book

This book is intended for system administrators, or software technicians who wants to upgrade Java ES software.

This guide assumes you are familiar with the following:

- Installation of enterprise-level software products
- Java ES components currently deployed in your environment
- System administration and networking on your supported Java ES platform

How This Book Is Organized

This guide includes the following chapters:

- **Chapter 1, “Planning for Upgrades”**, provides information for planning the upgrade of the Sun Java™ Enterprise System (Java ES) software to Java ES 2005Q4 (Release 4).
- **Chapter 2, “Upgrading Java ES Shared Components”**, provides information on upgrading shared components for Sun Java™ Enterprise System (Java ES) 2005Q4 release.
- **Chapter 3, “Directory Server and Administration Server”**, describes the upgrading procedure for Directory Server and Administration Server.
- **Chapter 4, “Directory Proxy Server”**, describes the upgrading procedure for Directory Proxy Server.
- **Chapter 5, “Web Server”**, describes the upgrading procedure for Web Server.
- **Chapter 6, “Message Queue”**, describes the upgrading procedure for Message Queue.
- **Chapter 8, “Application Server”**, describes the upgrading procedure for Application Server.
- **Chapter 9, “Access Manager”**, describes the upgrading procedure for Access Manager.
- **Chapter 10, “Directory Preparation Tool”**, describes the upgrading procedure for Directory Preparation Tool.
- **Chapter 11, “Messaging Server”**, describes the upgrading procedure for Messaging Server.
- **Chapter 12, “Calendar Server”**, describes the upgrading procedure for Calendar Server.
- **Chapter 13, “Communications Express”**, describes the upgrade procedure for Communications Express.
- **Chapter 14, “Instant Messaging”**, describes the upgrading procedure for Instant Messaging.
- **Chapter 15, “Portal Server”**, describes the upgrading procedure for Portal Server.
- **Chapter 16, “Portal Server Remote Access”**, describes the upgrading procedure for Portal Server Remote Access.

- [Chapter 17, “Delegated Administrator”](#), describes the upgrading procedure for Delegated Administrator.
- [Appendix A](#) provides the contents of the various Java Enterprise System releases.
- [“Index,”](#) provides support for text search on a topic one has come across.

Typographic Conventions

Typeface*	Meaning	Examples
AaBbCc123	API and language elements, HTML tags, web site URLs, command names, file names, directory path names, onscreen computer output, and sample code.	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail.
AaBbCc123	What you type, when contrasted with on-screen computer output.	% su Password:
<i>AaBbCc123</i>	Book titles, new words or terms, words to be emphasized. A placeholder in a command or path name to be replaced with a real name or values.	Read Chapter 6 in the <i>User's Guide</i> . These are called <i>class</i> options. You <i>must</i> be superuser to do this. To delete a file, type <code>rm filename</code> .

*. The settings on your browser might differ from these settings.

Shell Prompts

The following table describes the shell prompts used in this book.

Shell	Prompt
C shell on UNIX or Linux	machine-name%
C shell superuser on UNIX or Linux	machine-name#

Shell	Prompt
Bourne shell and Korn shell on UNIX or Linux	\$
Bourne shell and Korn shell superuser on UNIX or Linux	#
Windows command line	C:\

Related Documentation

The <http://docs.sun.com> web site enables you to access the Sun technical documentation online. You can browse the archive or search for a specific book title or subject.

Books in This Documentation Set

The Sun Java Enterprise System manuals are available as online files in Portable Document Format (PDF) and Hypertext Markup Language (HTML) formats. Both formats are readable by assistive technologies for users with disabilities. The Sun™ documentation web site can be accessed here:

<http://docs.sun.com>.

The Sun Java Enterprise System documentation includes information about the system as a whole and information about its components. This documentation can be accessed here:

<http://docs.sun.com/prod/entsys.05q1>.

The following table lists the system-level manuals in the Sun Java Enterprise System documentation set. The left column provides the name and part number location of each document and the right column describes the general contents of the document.

Table 1 Books in the Java Enterprise System Documentation Set

Document or Book Title	Description
<i>Java Enterprise System Release Notes</i> http://docs.sun.com/doc/819-0057	Contains the latest information about the Java Enterprise System, including known problems. In addition, components have their own release notes.
<i>Java Enterprise System Documentation Roadmap</i> http://docs.sun.com/doc/819-0055	Provides descriptions of the documentation related to Java Enterprise System. Includes links to the documentation associated with the components.
<i>Java Enterprise System Technical Overview</i> http://docs.sun.com/doc/819-0061	Introduces technical concepts and terminology used in Java Enterprise System documentation. Describes the Java Enterprise System, its components, and role in supporting distributed enterprise applications. Also covers life-cycle concepts, including an introduction to system deployment.
<i>Java Enterprise System Deployment Planning Guide</i> http://docs.sun.com/doc/819-0058	Provides an introduction to planning large-scale deployments based on Java Enterprise System. Presents some basic concepts and principles of deployment planning and introduces a number of processes that you can use as a starting point when designing enterprise-wide deployments.
<i>Sun Java Enterprise System User Management Guide</i> http://docs.sun.com/doc/817-5761	Guides you through the process of optimizing user management.
<i>Java Enterprise System Deployment Example Series: Evaluation Scenario</i> http://docs.sun.com/doc/819-0059	Provides a series of examples for multiple evaluation scenarios.
<i>Java Enterprise System Deployment Example Series: Small Scale Communications Scenario</i> http://docs.sun.com/doc/819-0060	Provides a series of deployment scenarios for multiple system requirements.
<i>Java Enterprise System Installation Guide</i> http://docs.sun.com/doc/819-0056	Guides you through the process of installing your Java Enterprise System. Shows you how to select the components that you want to install, how to configure the components that you install, and how to verify that the software you install functions properly.
<i>Java Enterprise System Upgrade and Migration Guide</i> http://docs.sun.com/doc/819-0062	Guides you through the upgrade and migration process from the previous versions to the newer ones.

Table 1 Books in the Java Enterprise System Documentation Set

Document or Book Title	Description
<i>Java Enterprise System Glossary</i> http://docs.sun.com/doc/816-6873	Defines terms that are used in Sun Java Enterprise System documentation.

Accessing Sun Documentation

For product downloads, professional services, service packs and support, and additional developer information, refer to the following online resources:

- **Download Center**
<http://www.sun.com/software/download/>
- **Professional Services**
<http://www.sun.com/service/sunps/sunone/index.html>
- **Sun Enterprise Services, Windows Service Packs, and Support**
<http://sunsolve.sun.com/>
- **Developer Information**
<http://developers.sun.com/prodtech/index.html>

The following location contains information about Sun Java Enterprise System and its components:

<http://www.sun.com/software/learnabout/enterprisesystem/>

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at:

<http://www.sun.com/documentation>

Third-Party Web Sites

Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in this document, go to:

<http://www.sun.com/service/contacting>.

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. To share your comments, go to <http://docs.sun.com> and click Send Comments. In the online form, provide the document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the guide or at the top of the document.

Planning for Upgrades

This chapter provides information used for planning the upgrade of Sun Java™ Enterprise System (Java ES) 2005Q1 (Release 3) software to Java ES 2005Q4 (Release 4). It contains the following sections:

- [Java ES 2005Q4 \(Release 4\) Components](#)
- [About Java ES Upgrades](#)
- [Upgrade Planning](#)
- [About Java ES Upgrades](#)
- [General Sequencing Guidelines](#)

Java ES 2005Q4 (Release 4) Components

As an introduction to planning the upgrade of Java ES software, this section reviews the components included in Java ES Release 4. Depending on your upgrade scenario, you might need to upgrade one or more of these components to their Release 4 version.

Java ES components are grouped into different types, as described in the Java Enterprise System Technical Overview (<http://docs.sun.com/doc/819-0061>). Accordingly, system service components provide the main Java ES infrastructure services, while service quality components enhance those system services. These two types of Java ES components are together referred to here as product components, components that are selectable within the Java ES installer.

Each product component depends on one or more locally shared libraries known as Java ES shared components. Shared components are installed automatically by the Java ES installer during product component installation, depending on the product components that are being installed.

Release 4 Product Components

The Java ES Release 4 product components are shown in the following table, listed alphabetically. For the service quality components among them, the table includes the type of service enhancement they provide.

Table 1-1 Java ES Release 4 Product Components

Product Component	Type	Abbreviation
Access Manager	System service component	AM
Administration Server	Service quality: administrative component	ADS
Application Server	System service component	AS
Calendar Server	System service component	CS
Communications Express	Service quality: access component	CX
Delegated Admin	Service quality: administrative component	DA
Directory Preparation Tool	Service quality: administrative component	DPT
Directory Proxy Server	Service quality: access component	DPS
Directory Server	System service component	DS
Instant Messaging	System service component	IM
Message Queue	System service component	MQ
Messaging Server	System service component	MS
Portal Server	System service component	PS
Portal Server Secure Remote Access	Service quality: access component	PSRA
Service Registry	System service component	SR
Web Proxy Server	Service quality: access component	WPS
Web Server	System service component	WS

Release 4 Shared Components

Java ES shared components, upon which the product components installed on a single computer depend, cannot be selected or deselected within the Java ES installer. When installing Java ES product components, the Java ES installer automatically installs the shared components needed by the installed product components.

The Java ES Release 4 shared components are listed in the following table.

Table 1-2 Java ES Release 4 Shared Components

Shared Component	Abbreviation	Version
Jakarta ANT Java/XML-based build tool	ANT	1.6.2
Apache Common Logging	ACL	1.0.2
Berkeley Database	BDB	4.2.52
Common agent container	CACAO	1.1
Derby DataBase	DerbyDB	10.0.2.1
International Components for Unicode	ICU	3.2
Instant Messenger SDK	IM-SDK	6.2.8
JavaBeans™ Activation Framework	JAF	1.0.3
Java Studio Enterprise Web Application Framework	JATO	2.1.5
JavaHelp™ Runtime	JHELP	2.0
JavaMail™ Runtime	JMAIL	1.3.2
Java Architecture for XML Binding Runtime	JAXB	1.0.4
Java API for XML Processing	JAXP	1.2.6
Java API for XML Registries Runtime	JAXR	1.0.7
Java APIs for XML-based Remote Procedure Call) Runtime	JAX-RPC	1.1.2
Java Calendar API	JCAPI	1.2
Java Dynamic Management™ Kit Runtime Library	JDMK	5.1.1
Java Security Services	JSS	4.1
KT Search Engine	KTSE	1.3.2
LDAP C SDK	LDAP C SDK	5.12
LDAP Java SDK	LDAP J SDK	4.17
Mobile Access Core	MA Core	6.2
Netscape Portable Runtime	NSPR	4.5.2
Network Security Services	NSS	3.10
SOAP Runtime with Attachments API for Java	SAAJ	1.2.1
Simple Authentication and Security Layer	SASL	2.19

Table 1-2 Java ES Release 4 Shared Components

Shared Component	Abbreviation	Version
Sun Java Enterprise System Monitoring Framework	MFWK	2.0
Sun Java Web Console	SJWC	2.2.4
Web Service Common Library	WSCL	1.0
Zip Compression Library	Zlib	1.1.4

About Java ES Upgrades

The upgrade of Java ES software to Release 4 is not generally performed using the Java ES installer or any other system utility. It is performed component-by-component, computer-by-computer, using component-specific upgrade procedures.

The upgrade of a component can range from a major upgrade, which might not be compatible with the previous version of the component, to a fully-compatible upgrade that simply provides bug fixes. Because of dependencies between Java ES components, the nature of the upgrade can impact whether other components need to be upgraded as well.

Product Component Upgrades

Java ES product component upgrades involve two basic operations that mirror the initial installation and configuration of Java ES product components:

- **Installing upgraded software.** The new software can enhance or fix existing software, or replace existing software. In general, the new software is achieved through the application of patches to existing software packages, the replacement of existing packages, the installation of new packages, or a full re-installation of a component using the Java ES installer.
- **Re-configuration.** Re-configuration encompasses any change in configuration data, user data, or dynamic application data needed to support the upgraded software. A change in data can mean additional data, a change in data format (whether in property files or database schema), or a change in data location. Sometimes re-configuration requires that you perform an explicit procedure and sometimes it takes place automatically without your involvement.

These two aspects of component upgrades are described in this *Upgrade Guide* for each of the Java ES product components.

The Upgrade Guide also covers other important aspects of product component upgrades, including:

- dependencies that impact an upgrade
- operations you might need to perform before you upgrade a component
- operations you perform to verify successful upgrade
- operations you perform if you need to roll back an upgrade

Shared Component Upgrades

Java ES shared component upgrades are often a necessary part of upgrading the product components that depend on them.

The upgrading of shared components is typically more straightforward than the upgrading of product components. In general, the upgrade is achieved through the application of patches to existing packages or the replacement of existing packages. As compared to upgrading product components, there is normally no re-configuration required, nor pre or post upgrade procedures to be performed. While shared components can be upgraded one by one. For more information see, [Chapter 2, “Upgrading Java ES Shared Components” on page 45.](#)

Upgrade Technologies

The upgrade of both product components and shared components, as described in this *Upgrade Guide*, involves the modification or replacement of currently installed software packages and, in some cases, the installation of new packages. On HP-UX similar technologies are employed for managing installed software packages and tracking changes through a package registry.

The Java ES packages can be installed and removed through the HP-UX `swinstall` and `swremove` commands, using packages found on the Java ES software distribution. Package contents, once installed, can be modified using patches that are applied or removed through the `swinstall` and `swremove` commands.

Patches to HP-UX packages are distributed through the SunSolve website at: <http://sunsolve.sun.com>.

HP-UX patches can patch one or more packages. The `swinstall` command saves a backup of the package being patched to facilitate the removal of the patch using the `swremove` command. Patches are identified by a patch ID, which consists of a patch number followed by a revision number that is incremented as the patch is modified over time.

Operating System Issues

The operating system issues impacting the upgrading of Java ES software are as described below:

Required Operating System Patches

In some situations, the successful upgrade of a Java ES product component can require you to first patch the operating system or apply specific fixes. Rather than applying the specific operating system patch required in each case, it is generally preferable to simply bring the operating system up to date before performing Java ES upgrades.

HP-UX platform update releases are available at:

<http://www1.itrc.hp.com/service/patch/releaseIndexPage.do?BC=patch.breadcrumb.main%7C&admit=-682735245+1129885420506+28353475>

Select “Jun’04(11.11 Support Plus)”

Upgrade Planning

The approach you take for upgrading a deployed Java ES Release 3 software system to Java ES Release 4 can depend on many factors:

- Your upgrade objectives and priorities
- The scope and complexity of the deployment architecture
- Your upgrade path, which depends on the version of the currently deployed Java ES software
- The nature of the dependencies between deployed Java ES components

For example, your Java ES deployment architecture might consist of a single Java ES component running on a single computer, and your upgrade objective is to fix some bug in the previous software release. On the other hand, your Java ES deployment architecture might consist of a number of interdependent Java ES components deployed across a number of different computers, and your upgrade objective is to achieve some new functionality by upgrading the minimum number of components possible to achieve that end with minimal downtime.

These two examples represent upgrade scenarios of very different complexities, requiring substantially different upgrade plans.

In general, the greater the number of Java ES components and the greater the number of computers in your deployment architecture, the more complex will be your upgrade plan.

What is an Upgrade Plan?

An upgrade plan specifies how to approach each stage of the upgrade process. This process involves, at a minimum, the phases shown in the following table.

Table 1-3 Phases in the Upgrade Process

Upgrade Phase	Description
Preparation	You develop an upgrade plan. In it, you specify the Java ES components you need to upgrade and the sequence by which you need to upgrade those components on the various computers in your system. You also plan how to test upgrade procedures in a staging environment before executing them in your production environment. In this step, you also back up your current system and test your ability to restore it to its current configuration.
Execution	You obtain all the necessary packages, patches, and tools needed for the upgrade. You execute the upgrade and re-consideration of your Java ES deployed system in a staging environment. This involves the backup of configuration and application data, the upgrade of system software, and the re-configuration or migration of data to the upgraded system.
Verification	You start up the upgraded software components and perform verification tests as you proceed. If verification is not successful, and problems cannot be resolved within a reasonable time frame, you might be forced to roll back the upgrade and restore the system to its previous state.

Table 1-3 Phases in the Upgrade Process

Upgrade Phase	Description
Rollback/restoration	You restore the system to its previous state as specified in the preparation phase. You also perform tests to verify that the rollback is successful.

Upgrade Paths

Because of the different characteristics of the Release 3 to Release 4 on HP-UX, upgrade paths involves upgrade from JES3 to JES4, and because the upgrade procedures for product components often depend on the upgrade path, the separate chapters in this *Upgrade Guide* describing the upgrade of each product component is captured.

The upgrade path to Java ES Release 4 are described in the following table:

Table 1-4 Upgrade Paths to Java ES 2005Q4 (Release 4)

Product Number	JavaES Release	Upgrade Characteristics	Upgrade Strategies
2005Q1	Release 3	Java ES Release 4 supports a mixture of Release 3 and Release 4 components on a single computer. This includes both product components and shared components. Compatibilities between Release 3 and Release 4 components have been tested, and any known incompatibilities are noted in the Java Enterprise System Release Notes (http://docs.sun.com/app/docs/doc/819-2329).	The coexistence of Release 3 and Release 4 components provides for the possibility of upgrading within a deployment architecture consisting of multiple computers.

Upgrade Dependencies

One of the main issues in planning the upgrade of any given Java ES component is to understand that component's dependencies on other Java ES components, and whether such other components also need to be upgraded to support the upgrade of the dependent component.

In this respect, there are two types of upgrade dependencies:

- **Hard upgrade dependency.** A hard upgrade dependency is where an upgraded version of a component requires an upgraded version of some component upon which it has a dependency. You cannot successfully upgrade the component without first upgrading the component upon which it depends. In other words, the Release 4 version of the component is not compatible with the Release 3 version of the component upon which it has a dependency.
- **Soft upgrade dependency.** A soft upgrade dependency is where an upgraded version of a component does not require an upgraded version of some component upon which it has a dependency. You can successfully upgrade the component without upgrading the component upon which it depends. In other words, the Release 4 version of the component is compatible with the Release 3 version of the component upon which it has a dependency.

Upgrading a Java ES component requires you to upgrade all the components upon which it has hard upgrade dependencies, but allows you to not upgrade components upon which it has soft upgrade dependencies.

This general rule does not necessarily apply, however, when multiple interdependent components are involved in an upgrade. In such cases, you have to upgrade a component if only one of several other Java ES components has a hard upgrade dependency on that particular component.

Upgrade All

The difference between hard and soft upgrade dependencies allows for the possibility of selectively upgrading Java ES components within a deployed system. This possibility applies to upgrading from Release 3 to Release 4 on a single computer (see upgrade path characteristics in [“Upgrade Paths” on page 30](#)).

Upgrade All. You can upgrade all deployed Java ES components to Release 4. The complexity of this approach also depends on your deployment architecture. In some cases, it simply is not feasible for business reasons to upgrade an entire system at one time.

The two approaches to performing upgrades are compared in the following table.

Table 1-5 Upgrade All

Upgrade	Approach Advantages	Disadvantages
Upgrade All	A consistent version for all components in your deployed system	Maximizes number of components to upgrade

Java ES Component Dependencies

As mentioned in the previous section, an upgrade plan specifies the Java ES components you need to upgrade and the sequence by which you need to upgrade those components. One of the important considerations in an upgrade plan is the dependencies between the various Java ES components in your deployed system.

This section provides information about Java ES component dependencies. The following dependency factors impact your upgrade plan.

- [Dependencies On Shared Components](#)
- [Dependencies On Product Components](#)
- [Multi-instance Upgrades](#)

Each of these factors is discussed briefly in the following sections.

Dependencies On Shared Components

When upgrading Java ES product components, you have to take into account dependencies these Java ES components have on Java ES shared components.

When a product component has a hard upgrade dependency on a shared component, the shared component also must be upgraded.

Shared Component Dependency Matrix

[Table 1-6](#) shows the dependencies of Java ES 2005Q4 (Release 4) product components on Java ES shared components. The abbreviations for product components that head the columns of [Table 1-6](#) are taken from [Table 1-1 on page 24](#). The abbreviations for shared components are spelled out in [Table 1-2 on page 25](#).

Directory Preparation Tool (DPT) is not included in [Table 1-6](#) because it has no dependencies on shared components. Service Registry (SR) and Web Proxy Server are omitted because they are new product components for which to upgrade previous version from which to upgrade.

Within the matrix of [Table 1-6](#) hard upgrade dependencies for Release 3 to Release 4 upgrades are marked “H,” while soft upgrade dependencies are marked “S.”

Table 1-6 Shared Component Dependencies of Java ES Release 4 Product Components

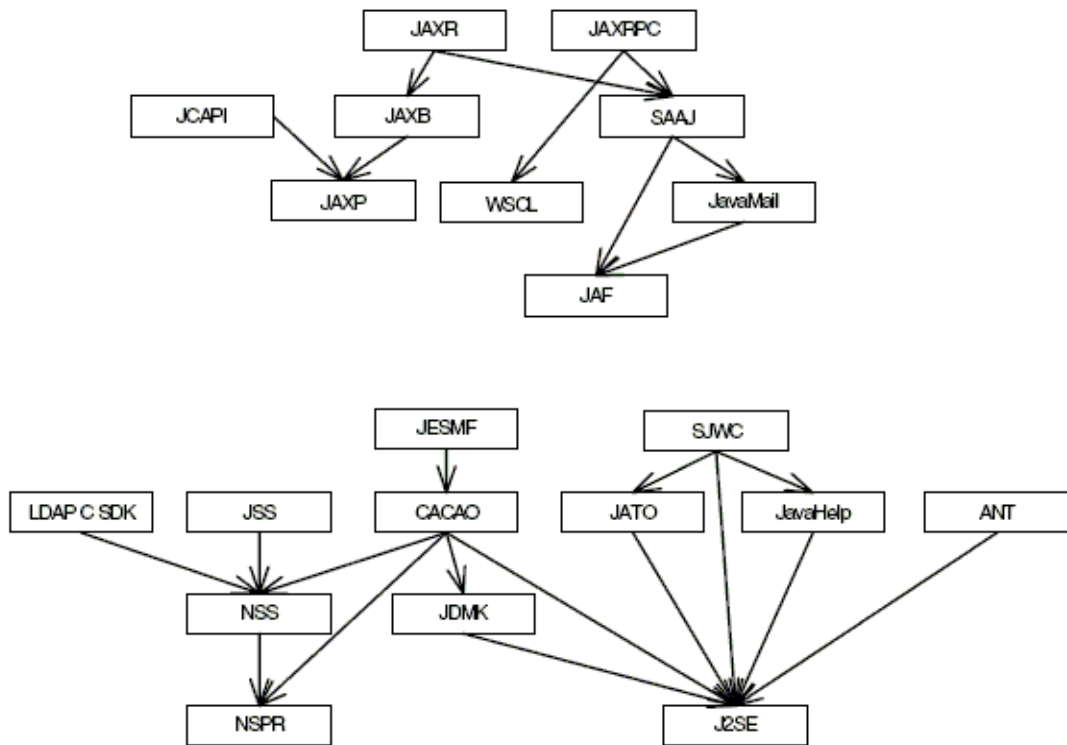
Shared Component	AM	ADS	AS	CS	DA	DPS	DS	IM	MQ	MS	PS	PSRA	WS
ANT			S										
ACL	S												
BDB	S												
CACAO								S					
ICU		S	S	S		S	S			H	S		S
IM-SDK								H			S		
jJAF	S		S					S	S		S	S	
JATO	S		S		S						S		
Java Help™	S		S						S				
Java Mail™	S		S					S	S		S	S	
JAXB	S		S										
JAXP	S		S					S	S		S	S	
JAXR	S		S										
JAX-RPC	S		S										
JCAPI								S					
JMDK			S					S					
JESMF								S					
JSS	S	S	S	S	S	S		S			S	S	
KTSE											S		S
LDAP C SDK		S		S	S	S			H				S
LDAP JSDK	S	S			S	S	S						
MA Core	S										H	H	

Table 1-6 Shared Component Dependencies of Java ES Release 4 Product Components

Shared Component	AM	ADS	AS	CS	DA	DPS	DS	IM	MQ	MS	PS	PSRA	WS
MFWK								H					
NSPR	S	S	S	S	S	S	S	S	S	H	S	S	H
NSS	S	S	S	S	S	S	S	S	S	H	S	S	H
SAAJ	S		S										
SASL	S	S											
SEDC													
SJWC	S		S										
WSCL	S		S										
ZLIB													H

The dependencies shown in [Table 1-6 on page 33](#) for any product component represent both direct and indirect shared component dependencies. In other words, a product component might depend on a specific shared component that, in turn, depends on one or more other shared components. The shared component dependencies shown in [Table 1-6 on page 33](#) include all such indirect dependencies. The following figure illustrates inter-dependencies among shared components.

Figure 1-1 Shared Components Inter-dependencies



Shared Component Upgrade Guidelines

[Table 1-6](#) lets you determine the shared components to upgrade when upgrading one or more product components on a given computer:

Release 3 to Release 4 Upgrades. If you are upgrading all product components from Release 3 to Release 4, all the shared components indicated in [Table 1-6](#) for the respective product components should be upgraded.

Even if you are selectively upgrading product components, however, it is the recommended practice to upgrade the shared components needed by all product components on the computer; Release 4 shared components are certified to support Release 3 product components.

If no hard upgrade dependencies are involved, you might not upgrade shared components at all. However, as a general rule, it is a good practice to upgrade your underlying Java ES shared component base to the most current versions.

NOTE The sequence of upgrading shared components can depend upon the shared component inter-dependencies shown in [Figure 1-1](#).

Also, if you plan to upgrade J2SE to J2SE 5.0, you should upgrade this shared component first. J2SE is the base component for many Java ES components.

For information on how to upgrade shared components, refer [Chapter 2, “Upgrading Java ES Shared Components,”](#).

Dependencies On Product Components

Dependencies of one product component on another are an important determinant of the Java ES components you need to upgrade and the sequence by which you need to upgrade them. Dependencies on product components fall into two general categories: runtime dependencies and configuration dependencies.

- **Runtime Dependencies.** The functioning of a software system is based on the interactions between its deployed components. The infrastructure dependencies between Java ES components are discussed in the *Java Enterprise System Technical Overview*. In upgrading any Java ES product component, you must take into account such dependencies. If an upgraded version of one component has a hard upgrade dependency on another component, that dependency implies that the dependent component should only be upgraded after the component upon which it depends is upgraded.
- **Configuration Dependencies.** In many cases a Java ES component must be installed, configured, and running for another component to be configured. For example, a Directory Server configuration directory must be running for you to configure Messaging Server components, or a Directory Server `user/group` directory must be running for an Access Manager service to be registered. Component upgrade procedures often involve re-configuration of upgraded components or migration of configuration data. In fact, some product components' main function is to provide configuration or administrative support to other components. As a result, configuration dependencies can have a strong impact on the sequence of upgrade procedures.

[Table 1-7](#) shows the dependencies between the Java ES product components listed in [Table 1-1 on page 24](#). Using [Table 1-7](#), you can diagram the chain of dependencies in your upgrade set. The left column lists each product component, the middle column shows its dependencies on other product components, the third characterizes each dependency, and the last column indicates whether or not the respective components must be local.

Table 1-7 Java ES Product Component Dependencies

Product Components	Dependencies	Nature of Dependencies	Must be Local
Access Manager	Directory Sever	To store configuration data and enable lookup of user data	No
	J2EE web container, one of: <ul style="list-style-type: none"> • Application Server • Web Server 	To provide web container runtime services	Yes
Access Manager SDK	Access Manger	To provide Access Manager services	No
	J2EE web container, one of: <ul style="list-style-type: none"> • Application Server • Web Server 	To provide web container runtime services	Yes
Administration Server	Directory Server	To provide configuration directory	No
Application Server	Message Queue	To provide reliable asynchronous messaging	Yes
	Web Server (optional)	To provide load balancing between instances	No

Table 1-7 Java ES Product Component Dependencies

Product Components	Dependencies	Nature of Dependencies	Must be Local
Calendar Server	Directory Server	To store and enable lookup of user data	No
	Directory Preparation Tool	To prepare directory for use by Calendar Server	No
	Access Manager (optional)	To provide a single sign-on	No
	Messaging Server (optional)	To provide email notifications	No
	Delegated Administrator (optional)	To provision users for calendar services	No
Communications Express	J2EE web container, one of: <ul style="list-style-type: none"> • Application Server • Web Server 	To provide web container runtime services	Yes
	Directory Server	To store and enable lookup of user data such as address books	No
	Directory Preparation Tool	To prepare directory for use by Communications Express	No
	Access Manager or Access Manager SDK	To provide authentication and authorization services, single sign-on	Yes
	Messaging Server	To provide web-based access to messaging	No
	Calendar Server	To provide web-based access to calendaring	No

Table 1-7 Java ES Product Component Dependencies

Product Components	Dependencies	Nature of Dependencies	Must be Local
Delegated Administrator	J2EE web container, one of:	To provide web container runtime services	Yes
	• Application Server		
	• Web Server		
	Directory Server	To store user data	No
Directory Preparation Tool	Directory Preparation Tool	To prepare directory for use by Delegated Administrator	No
	Access Manager or Access Manager SDK	To provide API needed for user provisioning	Yes
	Directory Server	To provide user/group directory that it is preparing for use by communications components	Yes
Directory Proxy Server	Administration Server	To configure Directory Proxy Server	No
	Directory Server	To provide access to Directory	No
Directory Server	Administration Server	To provide Directory Server	No
Instant Messaging	J2EE web container, one of:	To provide web container runtime services	Yes
	• Application Server		
	• Web Server		
	Directory Server	To provide access to a directory	No
Message Queue	Access Manager (optional)	To provide a sign on	No
	None		

Table 1-7 Java ES Product Component Dependencies

Product Components	Dependencies	Nature of Dependencies	Must be Local
Messaging Server Store	Directory Server	To store configuration data and enable lookup of user data	No
MTA			
MMP	Administration Server	To store configuration data in Directory Server configuration directory	Yes
MEM			
	Directory Preparation Tool	To prepare directory for use by Messaging Server	No
	Access Manager (optional)	To provide a sign-on	No
	Delegated Administrator	To provision users for messaging services	No
Portal Server	J2EE web container, one of: <ul style="list-style-type: none"> • Application Server • Web Server 	To provide web container runtime services	Yes
	Directory Server	To store configuration data and enable lookup of user profiles	No
	Access Manager or Access Manager SDK	To provide authentication and authorization services, single sign-on	Yes
	Communications Express	To provide messaging and calendar channels	No
Portal Server Secure Access	Portal Server	To provide access to portal	Yes
	Access Manager or Access Manager SDK	To provide authentication and authorization services, single sign-on	Yes
Web Server	None		
Web Proxy Server	None		
Service Registry	None		

Multi-instance Upgrades

The sequence of upgrade procedures can depend on whether and how redundancy is being used in a deployment architecture. Multiple instances of a Java ES component can be used to achieve high availability, scalability, serviceability, or some combination of these service qualities. There are three technologies that make use of redundant components in Java ES deployment architectures: load balancing, high availability techniques (High Availability Session Store), and multimaster replication (Directory Server).

In most cases where redundancy is involved, it is desirable to perform upgrades without incurring downtime. These rolling upgrades attempt to successively upgrade redundant instances of a component without compromising the service that they provide.

In most cases the redundant instances are deployed across multiple computers. From an upgrade planning perspective, this might imply isolating the upgrade of such replicated components from other component upgrades in order to achieve minimal downtime. In other words, you might perform all the pre-upgrade tasks for the component on each computer before performing a rolling upgrade of the replicated component.

Each replication technology has configuration or re-configuration procedures that might affect the overall sequence of Java ES component upgrades.

General Sequencing Guidelines

The factors discussed in the previous sections can all impact which Java ES components you plan to upgrade as well as the order in which you upgrade them. These factors also influence your approach to upgrading Java ES components that are deployed across multiple computers. The specific impact of all these factors depends on your deployment architecture.

Nevertheless a few general sequencing guidelines apply, though not in every case. The following list provides the order in which Java ES components can be successfully upgraded on a single computer or in a deployed system. When performing an upgrade, simply omit those components that are not part of your deployment architecture, or, if you are performing a selective upgrade, omit those components which are not part of your upgrade plan.

NOTE The chapters in this Upgrade Guide are arranged according to the order you would normally upgrade Java ES components, as indicated by these sequencing guidelines.

1. Shared Components (See [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.)

Shared components should generally be upgraded before the components which depend on them.

2. Directory Server and Administration Server (See [Chapter 3, “Directory Server and Administration Server”](#) on page 61.)

Many components store user data or configuration data in Directory Server, so upgrades to Directory Server should generally be performed before upgrading the components that have runtime or configuration dependencies on Directory Server.

3. Directory Proxy Server (See [Chapter 4, “Directory Proxy Server”](#) on page 71.)

Directory Proxy Server has a hard upgrade dependency on Directory Server and Administration Server and is therefore upgraded after Directory Server and Administration Server. Other components might access Directory Server through Directory Proxy Server.

4. Web Server (See [Chapter 5, “Web Server”](#) on page 81.)

A number of Java ES components require the support of a web container, which, if upgraded, should be upgraded before the components requiring web container services. Normally web container services are provided by Web Server, but if your architecture contains both, upgrade Web Server first.

5. Message Queue (See [Chapter 6, “Message Queue”](#) on page 87.)

Message Queue, if upgraded, is best upgraded before Application Server, which requires Message Queue to be Java 2 Enterprise Edition (J2EE) compliant.

6. High Availability Session Store (See [Chapter 7, “High Availability Session Store”](#) on page 97.)

High Availability Session Store, if upgraded, is best upgraded before Application Server, which requires High Availability Session Store for high availability.

7. Application Server (See [Chapter 8, “Application Server” on page 107.](#))

Application Server depends on Web Server for its load balancing plug in, so if you are using that capability, Application Server should be upgraded after Web Server.

8. Access Manager (See [Chapter 9, “Access Manager” on page 115.](#))

Access Manager plays a central role in authentication and authorization, including single sign-on, and, if upgraded, should be upgraded before the components that depend on it for those services. In addition, Access Manager requires specific Directory Server schema (Schema 2), which affects how other components use Directory Server.

9. Directory Preparation Tool (See [Chapter 10, “Directory Preparation Tool” on page 135.](#))

Directory Preparation Tool depends on the Directory Server schema and should therefore be run against Directory Server. If upgrading Directory Preparation Tool, it should be upgraded before you upgrade the communications components that depend on Directory Preparation Tool to make changes in the directory: Messaging Server and Calendar Server.

10. Messaging Server (See [Chapter 11, “Messaging Server” on page 143.](#))

Messaging Server, if upgraded, should be upgraded only after the preceding upgrades and should be upgraded before Communications Express, which has a dependency on Messaging Server components.

11. Calendar Server (See [Chapter 12, “Calendar Server” on page 153.](#))

Calendar Server, if upgraded, should be upgraded after Messaging Server since some of its functions require Messaging Server support. Calendar Server should be upgraded before Communications Express, which has a dependency on Calendar Server.

12. Communications Express (See [Chapter 13, “Communications Express” on page 161.](#))

Communications Express, if upgraded, depends on many of the preceding components (Calendar Server, Messaging Server, Directory Preparation Tool, Access Manager, Web Server, and Directory Server) and, if upgraded, should be upgraded after them.

13. Instant Messaging (See [Chapter 14, “Instant Messaging” on page 173.](#))

Instant Messaging, if upgraded, can be upgraded at almost any point after Access Manager has been upgraded.

14. Portal Server (See [Chapter 15, “Portal Server”](#) on page 181.)

Portal Server, like Communications Express, depends on many of the preceding components, but in particular, it depends on Communications Express to provide messaging and calendar channels, and, if upgraded, should therefore be upgraded after Communications Express.

15. Portal Server Remote Access (See [Chapter 15, “Portal Server”](#) on page 181.)

Portal Server Secure Remote Access, if upgraded, can be upgraded anytime after Portal Server has been upgraded.

16. Delegated Administrator (See [Chapter 16, “Portal Server Remote Access”](#) on page 191.)

Delegated Administrator, if upgraded, can be upgraded and used to provision users any time after Directory Preparation Tool has been upgraded and run against Directory Server. By convention, users are provisioned after other services have been upgraded and started, however, Delegated Administrator can be upgraded before upgrading the communications components that depend on Delegated Administrator for provisioning users.

NOTE On HP-UX, we do not have HADB server. Hence, for the load balancing feature we use HADB over Solaris platform.

Upgrading Java ES Shared Components

This provides information on upgrading shared components for Sun Java™ Enterprise System (Java ES) 2005Q4 release. This chapter contains the following sections:

- [Shared Component Upgrade Overview](#)
- [Upgrading Shared Components by Applying Patches](#)
- [Upgrading Components by Replacing Packages](#)

Shared Component Upgrade Overview

Upgrading shared components for Java ES 2005Q4 (Release 4) should be done as part of a larger upgrade plan, as discussed in [Chapter 1, “Planning for Upgrades”](#). To ensure that you have a successful upgrade, read Chapter 1 carefully and prepare an upgrade plan that meets your needs.

About Your Upgrade Plan

Your upgrade plan should cover the following areas:

- **Operating System Issues.** Perform any operating system upgrades, as described in [“Operating System Issues” on page 28](#) before you upgrade shared components.

- **Upgrade Path.** Determine which version of Java Enterprise System you currently have and make sure you understand the supported path to upgrade to Java ES Release 4. Procedures that depend on a specific upgrade path are noted in this chapter. For more information on upgrade path, refer to [“Upgrade Paths” on page 30](#).
- **Upgrade Dependencies.** Understand the interdependencies of the components you are upgrading. Typically, you sequence the upgrade of product components according to their dependencies. For example, before you upgrade a component you upgrade any component upon which it depends. There are various other factors to consider, such as hard and soft dependencies, as explained in [“Upgrade Dependencies” on page 31](#).

Use [Table 1-6 on page 33](#) to determine which shared components need to be upgraded before you upgrade product components.

- **Upgrade All.** If you are upgrading all product components on a computer, then you should upgrade all shared components upon which the product components depend. Best practice is to upgrade all shared components upon which all product components on the computer depend. Release 4 shared components are certified to support Release 3 product components. For more information, refer to [“Shared Component Upgrade Guidelines” on page 35](#).
- **Sequencing Guidelines.** Review the sequencing guidelines listed in [“General Sequencing Guidelines” on page 41](#). Typically, shared components are upgraded first. However, you should understand the entire sequence of your upgrade to Java ES Release 4 before beginning your upgrade process.

Technologies for Upgrading Shared Components

There are three technologies for upgrading shared components to Java ES Release 4. The technologies you use depend on the number and type of shared components you are upgrading, according to your upgrade plan.

- **Patches:** Most shared components on HP-UX platforms can be upgraded to Java ES Release 4 through the application of patches. Patches typically upgrade a single component or a group of related components.

If your upgrade plan calls for upgrading a few shared components, then you might consider applying individual patches to those components for which upgrade patches are available.

[Table 2-1 on page 48](#) shows the upgrade patches that are available for each shared component. [“Upgrading Shared Components by Applying Patches” on page 49](#) provides instructions for downloading and applying patches.

- **Patch Clusters.** A patch cluster bundles all the upgrade patches available for shared components. This simplifies the upgrade process because you can upgrade all the corresponding shared components by executing a single upgrade script provided with the patch cluster.

There is a separate patch cluster for each Solaris platform. A patch cluster is not available for the HP-UX platform.

- **Replacement of Packages.** Some shared components can only be upgraded by replacing existing packages on your system with newer versions of the packages. The newer versions of shared component packages are available with your Java ES Release 4 distribution.

[Table 2-1 on page 48](#) shows the shared components that use replacement of packages when upgrading to Java ES Release 4. [“Upgrading Components by Replacing Packages” on page 51](#) provides details on replacement of packages.

General Upgrade Procedure

The general steps you take to upgrade shared components are listed below:

1. From your upgrade plan, determine your upgrade path, the shared components you wish to upgrade, and the sequence for upgrading them. Review the earlier sections in this overview for information on developing an upgrade plan. You can also refer to [“Upgrade Planning” on page 28](#) for additional information.
2. Determine the upgrade technologies available to upgrade the shared components listed in your upgrade plan.

You can find this information in the following section:

[“Upgrading Shared Components from Java ES Release 3” on page 48](#)

3. Depending on your specific needs, follow the procedures in the following section:
 - [“Upgrading Shared Components by Applying Patches” on page 49](#)
 - [“Upgrading Components by Replacing Packages” on page 51](#)

Upgrading Shared Components from Java ES Release 3

[Table 2-1](#) below shows the upgrade technologies to use when upgrading shared components from Java ES (Release 3). Please note the following different upgrade options that are available:

- In many cases, you apply specific patches to upgrade shared components to Java ES Release 4.
- In some cases, you replace packages Release 3 packages with Release 4 packages provided with the Java ES distribution.
- For shared components that have not changed since Release 3, no upgrade is necessary.

NOTE In [Table 2-1](#), the trailing two digits in the patch ID specify the revision number for the patch. A higher revision number indicates a newer version.

“[Release 4 Shared Components](#)” specifies the full name of shared components listed in [Table 2-1](#).

Table 2-1 Upgrade Technologies to Upgrade Shared Components from Java ES Release 3

Shared Component	Upgrade Method	Patch-ID
ANT	Apply specific patch	121508-01
ACL	Unchanged since Release 3	
BDB	Apply specific patch	121496-01
CACAO	Unchanged since Release 3	
Derby	Unchanged since Release 3	
ICU	Apply specific patch	121497-01
IM-SDK	Apply specific patch	121892-01
JAF	Unchanged since Release 3	
JATO	Apply specific patch	121491-01
Java Help™	Unchanged since Release 3	
Java Mail™	Unchanged since Release 3	

Table 2-1 Upgrade Technologies to Upgrade Shared Components from Java ES Release 3

Shared Component	Upgrade Method	Patch-ID
JAXB	Apply specific patch	121498-01.
JAXP	Apply specific patch	121499-01
JAXR	Apply specific patch	121501-01
JAX-RPC	Apply specific patch	121500-01
JCAPI	Apply specific patch	121492-01
JDMK	Apply specific patch	121490-01
JSS	Apply specific patch	121506-01
KTSE	Unchanged since Release 3	
LDAP C SDK	Unchanged since Release 3	
LDAP J SDK	Apply specific patch	121494-01
MFWK	Unchanged since Release 3	121489-01
NSPR	Apply specific patch	121502-01
NSS	Apply specific patch	121503-01
SAAJ	Apply specific patch	121507-01
SASL	Apply specific patch	121493-01
SJWC	Replace packages	121495-01
WSCL	Apply specific patch	121509-01
ZLIB	Unchanged since Release 3	

Upgrading Shared Components by Applying Patches

On HP-UX platforms, many packages for shared components can be upgraded from Java ES 2005Q1 (Release 3) by application of patches that modify the installed packages.

The advantage of patch technology over replacement of packages is that revisions applied by a patch can later be backed out, if needed. Typically, the size of a patch is smaller than the size of an updated package, so it is easier to download and install. Patches are generally more current than the latest available package. The most current revision of a patch is readily available for download from SunSolve, as described in this section.

[Table 2-1 on page 48](#) shows the patch IDs for all shared component patches used to upgrade to Java ES 2005Q4 (Release 4) from Java ES Release 3.

If your upgrade plan calls for upgrading only a few shared components, you probably want to apply individual patches as described in the following procedure. You can later back out patches you apply, as described following the procedure.

NOTE Before making any changes to your system, it is advisable to first backup your system

➤ **To apply a patch to upgrade to Java ES 2005Q4**

1. Retrieve the patch from the following location:

<http://sunsolve.sun.com>.

[Table 2-1 on page 48](#) lists the available patch IDs to upgrade each shared component.

2. Log in as superuser or become superuser by using the `su -` command.

NOTE Become superuser with `su -`, not `su`, to make sure you obtain the complete superuser environment

3. Before applying the patch, read any special instructions in the README supplied with the patch.
4. Apply the patch using the `swinstall` command, as indicated in the following example, which applies the patch to the NSPR shared component:

```
swinstall 121502-01
```

For information on the `swinstall` command, refer to the `swinstall` man page.

➤ **To back out a patch you previously applied**

1. Log in as superuser or become superuser by using the `su -` command.

NOTE Become superuser with `su -`, not `su`, to make sure you obtain the complete superuser environment

2. Back out the patch using the `swremove` command, as indicated in the following example, which backs out the patch to the NSPR shared component:

```
swremove 121502-01
```

3. For information on the `swremove` command, refer to the `swremove` man page.

Upgrading Components by Replacing Packages

Many shared components do not use patch technology to upgrade to Java ES 2005Q4 (Release 4). Instead, you upgrade these components by installing new packages or replacing existing packages with newer versions of the packages. In a few cases, you run special procedures to properly upgrade a shared component.

- For upgrades from Java ES 2005Q1 (Release 3), [Table 2-1 on page 48](#) shows which shared components use package replacement to upgrade to Java ES Release 4.
- [Table 2-2 on page 53](#) provides any special procedures that might be required for HP-UX platforms.

Upgrade Strategy for Replacement of Packages

The number of shared components you upgrade depends on your upgrade plan and the Java ES components installed on your computer.

For upgrades from Java ES Release 3, you might be either upgrading all components to Release 4. While you can choose to upgrade only those Java ES shared components needed to support the product components you select to upgrade, it is advisable to upgrade all shared components on your computer. For more information, refer to [“Shared Component Upgrade Guidelines” on page 35](#).

NOTE The sequence of upgrading shared components is determined by component interdependencies, which should be reflected in your upgrade plan.

Upgrade Path

The procedures for upgrading shared components by replacement of packages generally do not depend on your upgrade path.

However, the section [“Packages for HP-UX Platforms” on page 52](#) contain tables that specify the few instances where the upgrade path determines the procedures you use.

Package Versions

When replacing packages, you should only replace packages with newer versions of those packages. The section [“Packages for HP-UX Platforms” on page 52](#) provides information on how to compare package versions before upgrading.

Package Locations

The packages for upgrading most shared components are provided with your Java ES distribution under one of the following directories, depending on your platform:

```
HPUX_PA-RISC/Product/shared_components/Packages
```

Localized versions of shared component packages can be found in the following directory:

```
HPUX_PA-RISC/Product/shared_components/Packages/locale/
```

Procedures for Replacement of Packages

With few exceptions, packages can be replaced following general procedures. Refer to [“Upgrading Packages for HP-UX Platforms” on page 56](#) for procedure.

Special Instructions

Some packages have special instructions for preserving configuration information or other data. Links to special instructions are contained in the package tables listed in the section [“Packages for HP-UX Platforms” on page 52](#).

Packages for HP-UX Platforms

[Table 2-2](#) below lists the HP-UX depot names for shared components that can be upgraded by replacement of packages. The table also lists the versions of the packages available in Java ES Release 4 and special instructions for upgrading the component.

For each shared component, the packages are listed in the sequence you would install them.

You should only replace packages with newer versions. Before you replace a package, compare the version of the package on your system with the version of the package you intend to use to replace it.

To determine the version of an installed package use the `swlist` command. The output of this command provides the package version.

```
swlist -a revision <depot name>
```

NOTE The versioning system for different packages varies, but generally a higher number indicates a newer version of the package.

[“Release 4 Shared Components” on page 24](#) specifies the full name of shared components listed in [Table 2-2](#).

Table 2-2 Package Versions for Upgrading Shared Components on HP-UX

Shared Component	Depot Names	Version No	Special Instructions
ANT	sun-ant	1.6.2	None
ACL	sun-aclg	Unchanged since Release 3	
BDB	sun-berkeleydatabase, sun-berkeleydatabase-java	4.2.52	None
CACAO	sun-cacao sun-cacao-man sun-cacao-cfg	Unchanged since Release 3	
Derby	sun-derby-core sun-derby-javadoc	Unchanged since Release 3	
ICU	sun-icu sun-icux	3.2	None
IMSDK	sun-im-dev	7.0	None
JAF	sun-jaf	Unchanged since Release 3	
JATO	sun-jato, sun-jatodmo, sun-jatodoc	2.1.5	None
Java Help™	sun-javahelpruntime	Unchanged since Release 3	
Java Mail™	sun-javamail	Unchanged since Release 3	

Table 2-2 Package Versions for Upgrading Shared Components on HP-UX

Shared Component	Depot Names	Version No	Special Instructions
JAX-Related Components			
JAXB	sun-jaxb	1.0.4	<ol style="list-style-type: none"> 1. Before installation of this patch, ensure that the patches for shared components of JAXP and WSCL are installed. 2. Do not uninstall the dependent packages/patches before uninstalling this patch.
JAXP	sun-jaxp	1.2.6_01	None
JAXR	sun-jaxr	1.0.7	<ol style="list-style-type: none"> 1. Before installation of this patch, ensure that the patches SAAJ and JAX-B are installed. 2. Do not uninstall the dependent packages/patches before uninstalling this patch.
JAX-RPC	sun-jaxrpc	1.1.2	<ol style="list-style-type: none"> 1. Before installation of this patch, ensure that the patches SAAJ and WSCL are installed. 2. Do not uninstall the dependent packages/patches before uninstalling this patch.
JCAPI	sun-jcapi	1.2	None
JDMK	sun-jdmk-runtime and sun-jdmk-runtime-jmx	5.1	None

Table 2-2 Package Versions for Upgrading Shared Components on HP-UX

Shared Component	Depot Names	Version No	Special Instructions
JSS	sun-jss sun-jssx	4.1	<p>1. Before installation of this patch, ensure that the patch NSS is installed.</p> <p>2. Do not uninstall the dependent packages/patches before uninstalling this patch.</p>
KTSE	sun-ktsearch	Unchanged since Release 3	
LDAP C SDK	sun-ldapcsdk sun-ldapcsdkx	Unchanged since Release 3	
LDAP J SDK	sun-ljdk	1	None
MFWK	sun-mfwk-man sun-mfwk-agent sun-mfwk-cfg	2.0	
NSPR	sun-nspr sun-nsprx sun-nspr-devel	4.5.2	None
NSS	sun-nss sun-nss-devel sun-nssx sun-nssu sun-nssux	3.10	<p>1. Before installation of this patch, ensure that the patch NSPR is installed.</p> <p>2. Do not uninstall the dependent packages/patches before uninstalling this patch.</p>
SAAJ	sun-saaj	1.2.1	<p>1. Before installation of this patch, ensure that the patch JAXP is installed.</p> <p>2. Do not uninstall the dependent packages/patches before uninstalling this patch.</p>
SASL	sun-sasl, sun-saslx	2.18	None

Table 2-2 Package Versions for Upgrading Shared Components on HP-UX

Shared Component	Depot Names	Version No	Special Instructions
SJWC	sun-mcon	2.2.4	
	sun-mconr		
	sun-mcosx		
	sun-mcos		
	sun-mctag		
WSCL	sun-wscl	1.0	None
ZLIB	sun-zlib	Unchanged since Release 3	

Upgrading Packages for HP-UX Platforms

The following procedure shows the general instructions for upgrading packages on HP-UX platforms by replacement of depots.

A few shared components require special instructions in addition to these general instructions. [Table 2-2 on page 53](#) provides special instructions for components that require them.

NOTE Before making any changes to your system, it is advisable to first backup your system.

➤ **To upgrade packages for shared components on HP-UX platforms by replacement of depots**

1. Log in as superuser or become superuser by using the `su -` command.

NOTE Become superuser with `su -`, not `su`, to make sure you obtain the complete superuser environment.

2. Check [Table 2-2 on page 53](#) for special instructions that might apply to the component you are upgrading.

Follow any special instructions before upgrading the package. If no special instructions are indicated, proceed to the next step.

3. Navigate to the location of the packages in your Java ES Release 4 distribution. The packages are found under one of the following directories:

```
HPUX_PA-RISC/Product/shared_components/Packages
```

```
HPUX_PA-RISC/Product/shared_components/Packages/locale
```

NOTE If you have a subset distribution of Java Enterprise System, this subset distribution contains all the necessary shared components at the location indicated above.

4. Remove the current versions of the packages that you are upgrading by using the `swremove` command.

For example, to remove packages for webconsole:

```
swremove sun-mcon sun-mconr sun-mcos sun-mcosx sun-mctag
```

For detailed information on removing packages, refer to the `swremove man` page.

5. Install the packages from your distribution using the `swinstall` command.

```
swinstall sun-mcon sun-mconr sun-mcos sun-mcosx sun-mctag
```

For detailed information on installing packages, refer to the `swinstall man` page.

6. Verify the version of an installed package use the `swlist` command. The output of this command provides the package version.

```
swlist -a revision <depot name>
```

For detailed information on these commands refer to the `swlist man` pages.

Upgrading J2SE for Java ES Release 4

Java ES 2005Q4 (Release 4) is certified for Java 2 Platform, Standard Edition (J2SE) Version 5.0.01, identified here as J2SE 5.0. (J2SE 5.0 is sometimes referred to as developer version 1.5.0.01).

Upgrading J2SE on HP-UX Platform

Before proceeding with upgrading J2SE on HP-UX platform, you should determine the version of J2SE currently used by your Java ES installation. For Java ES 2005Q1 (Release 3), the default is J2SE Version 1.5.0. For various reasons, you might have upgraded J2SE from the default versions.

Use the following procedure to determine which version of J2SE your Java ES installation is using.

► To display the J2SE version used by your Java ES installation

1. Run the following command:

```
/usr/jdk/entsys-j2se/bin/java -version
```

2. Here is an example of the version string displayed for J2SE 1.4.2.03:

```
java version "1.4.2.03"
```

You can upgrade J2SE to 5.0.01 by downloading J2SE 5.0 from the HP-UX Java at:

http://www.hp.com/products1/unix/java/java2/jdkjre5_0/index.html.

Java Enterprise System does not use the default J2SE installed on your computer, but instead maintains a symbolic link to the supported version of J2SE. After installing J2SE you might need to update the Java ES symbolic link so it points to the newly installed J2SE.

The procedures you use to upgrade J2SE depends on whether you are upgrading from J2SE 1.4 or an earlier version of J2SE 5.0. The general steps for HP-UX platforms follow. Details of these steps are in procedures that appear later in this section.

► Follow these steps if you are upgrading from J2SE 1.4.2_03

1. Install J2SE 5.0.01 or later.
2. Set the Java ES symbolic link to point to the newly installed J2SE.

NOTE You do not need to remove J2SE 1.4. Both versions can be installed on the same computer.

You may elect to continue running some Java ES services on the previous version of J2SE. Consult the appropriate component product administration guides to do so. For example, you can change the J2SE symlink used by Application Server to point to the previous J2SE version.

➤ **Follow these steps if you are upgrading from an earlier version of J2SE 5.0**

1. Remove the earlier version of J2SE 5.0.
2. Install J2SE 5.0.01 or later.
3. Set the Java ES symbolic link to point to the newly installed J2SE.

Starting and Stopping Services

When upgrading J2SE you might want to shut down any services that depend on the currently installed J2SE before proceeding. This is to avoid any problems that might arise with services that are using the current J2SE. If you do not shut down the services that depend on J2SE, after installing J2SE and setting the Java ES symbolic link to it, you should reboot your system.

Installing J2SE 5.0 on HP-UX Platform

You can install J2SE 5.0.01 on HP-UX platform by using the bits available from the HP website.

➤ **To install J2SE 5.0 on HP-UX platform using the HP packages**

1. Navigate to the following location in the HP website to retrieve the current version of JDK 5.0 and the installation instructions:

http://www.hp.com/products1/unix/java/java2/jdkjre5_0/index.html.

2. Follow the instructions for installation available with the download. Before installing J2SE, you might want to stop services that depend on J2SE, as described in “[Starting and Stopping Services](#)” on page 59.
3. After installation is complete, proceed to the section “[Setting the J2SE Symlink for Java Enterprise System on HP-UX Platform](#)” on page 60.

Setting the J2SE Symlink for Java Enterprise System on HP-UX Platform

Java Enterprise System maintains a symbolic link that points to the supported version of J2SE platform. Java Enterprise System maintains this link to ensure that Java ES services can find the correct J2SE runtime to use.

If you upgraded to J2SE 5.0.01 from J2SE 1.4.2_03, then you need to set the symbolic link so it points to the newly installed J2SE 5.0.01. If you upgraded to J2SE 5.0.01 from an earlier version of J2SE 5.0, then you just need to verify that your Java Enterprise System installation is using the updated version.

The following procedures show how to set the Java Enterprise System symbolic link and verify the version of your upgraded J2SE installation.

► **To Set the J2SE symbolic link for Java Enterprise System on HP-UX**

1. Reset the `/usr/jdk/entsys-j2se` symbolic link to point to the newly installed or updated J2SE installation as indicated below:

If you installed J2SE 5.0.01 in the default location, reset the symbolic link as follows:

```
rm /usr/jdk/entsys-j2se
```

```
ln -s /opt/java1.5 /usr/jdk/entsys-j2se
```

If you installed J2SE 5.0.01 in a non-default location, replace the default path (path `/opt/java1.5`) with the path to your non-default location.

2. If you previously stopped services prior to upgrading or installing J2SE 5.0 Update 4, restart the services.

If you did not stop services prior to upgrading or installing J2SE 5.0 you might want to reboot your system so services that depend on J2SE 5.0 use the new symbolic link.

Chapter 3

Directory Server and Administration Server

This chapter describes how to upgrade Directory Server and Administration Server components to Java ES Release 4 (2005Q4). These upgrades are documented together because the two components work closely together.

The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Directory Server and Administration Server Upgrades](#)
- [Upgrading Directory Server and Administration Server from Java ES Release 3](#)

Overview of Directory Server and Administration Server Upgrades

This section describes the following general aspects of Directory Server and its associated Administration Server components that impact upgrading to Java ES Release 4:

- [About Java ES Release 4](#)
- [Java ES Release 4 Upgrade Roadmap](#)
- [Directory Server and Administration Server Data](#)
- [Compatibility Issues](#)
- [Dependencies](#)

About Java ES Release 4

Java ES Release 4 versions of Directory Server and Administration Server represent only minor bug fixes and improvements. There are no new functional capabilities.

Java ES Release 4 Upgrade Roadmap

Table 3-1 shows the supported Directory Server and Administration Server, upgrade paths to Java ES Release 4.

Table 3-1 Upgrade Paths to Java ES Release 4 Directory Server and Administration Server 2005Q4

Java ES Release	Components	General Approach	Re-configuration Required
Release 3	Sun Java System Directory Server 5 Q1 Sun Java System Administration Server 5 Q1	Direct Upgrade: Apply patches and re-configure configuration directory	Automatic re-configuration of data in configuration directory

Directory Server and Administration Server Data

Directory Server and Administration Server make use of Directory Server itself for storing configuration data. The data is stored in a specific tree structure within the directory. The Directory Server instance hosting the configuration is referred to as the configuration directory.

The configuration directory can be a dedicated Directory Server instance, which is a recommended security practice, or it can also host user identity data or service configuration data. The configuration directory can reside on the same computer as other Directory Server instances and the Administration Server; however in most deployment architectures, the configuration directory is remote from the other components that use it to store configuration information.

The following table shows the type of data that could be impacted by an upgrade of Directory Server and Administration Server software.

Table 3-2 Directory Server and Administration Server Data Usage

Type of Data	Location	Usage
Directory Server configuration data	Configuration directory	Configuration of Directory Server
Administration Server configuration data	Configuration directory	Configuration of Administration Server

Compatibility Issues

Release 4 Directory Server and Administration Server do not introduce any interface changes. These components are, as a group, backwardly compatible with earlier versions. However, any one of these components is not backwardly compatible with earlier versions of the others; all need to be upgraded as a unit.

Dependencies

Dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Directory Server and Administration Server software. Each of these components has dependencies on Java ES components as follows:

- **Directory Server.** Directory Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)). Directory Server has a dependency on Administration Server, which is used to configure Directory Server replication and other aspects of Directory Server functions.
- **Administration Server.** Administration Server (and the Administration Console user interface) has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)). Administration Server has a dependency on Directory Server (specifically a configuration directory) where it stores configuration data.

Upgrading Directory Server and Administration Server from Java ES Release 3

This section includes information about upgrading Directory Server and Administration Server from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Directory and Administration Server Upgrade](#)

Introduction

When upgrading Java ES Release 3 Directory Server and Administration Server, to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Re-configuration of Directory Server and Administration Server are achieved by synchronizing the configuration directory with the upgraded software.
- **Upgrade Dependencies.** While Directory Server and Administration Server have dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Directory Server and Administration Server are compatible with the Release 3 versions of these shared components. Upgrade of these shared components is therefore optional with respect to upgrade of Directory Server and Administration Server to Release 4.

Directory Server has a hard upgrade dependency on Administration Server. Both components should therefore be upgraded together to Release 4.

- **Backward Compatibility.** Release 4 Directory Server and Administration Server, backwardly compatible with their Release 3 versions.
- **Upgrade Rollback.** A rollback of the Release 4 upgrade is achieved on HP-UX by removing the Release 4 upgrade patches and re-synchronizing the configuration directory with the previous software state.

Directory and Administration Server Upgrade

This section describes how to perform an upgrade of Directory Server and Administration Server from Java ES Release 3 to Java ES Release 4 on the HP-UX. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Directory Server and Administration Server](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Directory Server and Administration Server, you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Directory Server and associated components by restarting the Directory Server daemon using the `-v` option:

```
# cd serverroot/bin/slapd/server # ./ns-slapd -v
```

You can see the result of the command as mentioned below:

```
Sun Java(TM) System Directory Server/5.2_Patch_3
```

and checking the startup messages in the Directory Server error log:

```
/opt/sun/mps/serverroot/slapd- <server-instance>/logs/errors
```

Table 3-3 Directory Server Version Outputs

Java ES Release	Directory Server Version Number
Release 3	Sun Java(TM) System Directory Server/5.2_Patch_3
Release 4	Sun Java(TM) System Directory Server/5.2_Patch_4

Upgrade Directory Server and Administration Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, because Directory Server and Administration Server do not require upgrading Release 3 shared components, this task is optional.

Back Up Directory Server Data

The Directory Server and Administration Server upgrade process modifies configuration directory data. Therefore, before you upgrade, it is recommended that you back up your configuration directory data using the Directory Server Console or a command-line utility such as `db2bak`.

Obtain Required Configuration Information and Passwords

You should know the Directory Server administrator user ID and password for your currently installed version.

In addition, Directory Server, and its associated Administration Server must run as the same user and group. That is, they must run with the same UID and GID.

Upgrading Directory Server and Administration Server

This section discusses considerations that impact the upgrade procedure for Directory Server and Administration Server followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Directory Server and Administration Server software to Java ES Release 4 takes into account the following considerations:

- Any Java ES components using a Directory Server instance (such as Messaging Server) should be shut down before you upgrade that instance. However, most deployment architectures use multiple instances of Directory Server to provide high availability or scalability. In such cases, you can perform a rolling upgrade of Directory Server and the Directory Server clients need not be shut down.
- The upgrade of the two components must be done in the following sequence: Administration Server and Directory Server. This sequence is because the re-configuration data must take place in a particular order.
- The component being upgraded must be shut down when patches are being applied, however the associated configuration directory must subsequently be running to re-configure the component being upgraded.
- In a deployment architecture in which there are multiple instances of Directory Server running on a single computer (all corresponding to the same installed Directory Server image), upgrading the Directory Server image will upgrade all the instances. In such architectures, there is only one Administration Server instance per installed Directory Server image.

- In many deployment architectures the configuration directory is a separate Directory Server instance. It might be local or on a different computer system from where the upgrade is being performed. Similarly, the Administration Server might be local or on a different computer system from where the upgrade of Directory Server is being performed.
- In some deployment architectures Directory Server has been installed standalone by deselecting Administration Server at installation time. In that case, however, the Administration Server upgrade procedure must still be performed, in addition to the Directory Server upgrade procedure, as described in the instructions that follow.

The Release 4 upgrade patches for HP-UX are as shown in the following table:

Table 3-4 Patches* to Upgrade Directory Server and Administration Server

Component	Patch ID
Directory Server	121393-01
Directory Server locale	121931-01
Administration Server	121515-01
Administration Server locale	121933-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to Directory Server and Administration Server instances residing locally on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 2-1 on page 48](#) for the shared components.
2. Obtain the required patches, based on [Table 3-4](#) for the Directory and Administration Server.

Patches can be downloaded from:

<http://sunsolve.sun.com>.

3. Login as super user.

4. Upgrade the Shared Components.

5. Upgrade the Administration Server.

```
swinstall -s <absolute-location>/<patch_ID> <patch_ID>
```

- a. /opt/sun/sbin/mpsadminserver sync-cds 5.2

You will be prompted for the admin username and password.

- b. /opt/sun/mps/serverroot/startadmin

6. Upgrade the Directory Server.

a. swinstall -s <absolute-location>/<patch_ID> <patch_ID>

b. cd /opt/sun/mps/serverroot/shared/bin; ./sync-product-cds -r /opt/sun/mps/serverroot -v 5.2_Patch_4 -j ds524.jar -g ds524.jar -n "Sun Java(TM) System Directory Server" -b "2005.286.1827" -i "cn=Sun ONE Directory Server, cn=Server Group, cn=<server-instance>.red.iplanet.com, ou=red.iplanet.com, o=NetscapeRoot"

You will be prompted for the admin username and password

Verifying the Upgrade

1. You can verify successful upgrade of Directory Server and associated components by restarting the Directory Server daemon using the `-v` option:

```
# cd <server_root>/bin/slapd/server
# ./ns-slapd -v
```

You can verify the result as mentioned below:

```
Sun Java(TM) System Directory Server/5.2_Patch_4
```

and then checking the startup messages in the Directory Server error log:

```
/opt/sun/mps/serverroot/slapd-< server-instance>/logs/errors
```

2. You can verify successful upgrade of Administration Server using the command as mentioned below:

```
# cd <server_root>/
# ./startconsole
```

- a. You have to enter the appropriate *user-id* and *password* in the System Server Console Login panel.

- b. Click Administration Instance in System Server Console panel.

You can see the version number as mentioned below:

```
version :5.2.4
```

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [“Upgrade Procedure” on page 67](#).

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Directory Server and Administration Server followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Directory Server and Administration Server is pretty much the reverse of the procedure for upgrading to Release 4. The patches are removed and the configuration directory is re-synchronized.

Rollback Procedure

1. Login as super user.
2. Rollback the Directory Server using the following commands:
 - a. `swremove <patch_ID>`
 - b. `cd /opt/sun/mps/serverroot/shared/bin; ./sync-product-cds -r /opt/sun/mps/serverroot -v 5.2_Patch_3 -j ds523.jar -g ds523.jar -n "Sun Java(TM) System Directory Server" -b "2005.179.2117" -i "cn=Sun ONE Directory Server, cn=Server Group, cn=<server-instance>.red.iplanet.com, ou=red.iplanet.com, o=NetscapeRoot"`

You will be prompted for the admin username and password.

3. Verify the rollback of Directory Server:

```
# cd <server_root>/bin/slapd/server
```

```
# ./ns-slapd -v
```

You can verify the result as mentioned below:

```
Sun Java(TM) System Directory Server/5.2_Patch_3
```

4. Roll back the Shared Components.
5. Roll back the Administration Server by using the following commands:

- a. `swremove <patch_ID>`
- b. `/opt/sun/sbin/mpsadminserver sync-cds 5.2`

You will be prompted for the admin username and password.

6. Verify the rollback of Administration Server.

```
/opt/sun/mps/serverroot/startconsole
```

- a. You have to enter the appropriate *user-id* and *password* in the System Server Console Login panel
- b. Click Administration Instance in System Server Console panel

You can see the version number as mentioned below:

```
version :5.2.3
```

NOTE When you try to authenticate using the 'admin_pwd', if you notice the Directory Server is not running, start the Directory Server using the following command:

```
/opt/sun/mps/serverroot/slaped-<server-instance>/start-slaped
```

Directory Proxy Server

This chapter describes how to upgrade Directory Proxy Server to Java ES 2005Q4 (Release 4) Sun Java System Directory Proxy Server 5 2005Q4.

The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [“Overview of Directory Proxy Server Upgrades” on page 71](#)
- [“Upgrading Directory Proxy Server from Java ES Release 3” on page 73](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *serverRoot*. At least part of this path might have been specified as an installation directory when Directory Proxy Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *serverRoot* is `/opt/sun/mps/serverroot`.

Overview of Directory Proxy Server Upgrades

This section describes the following general aspects of Directory Proxy Server that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4](#)
- [Java ES Release 4 Upgrade Roadmap](#)
- [Directory Proxy Server Data](#)
- [Compatibility Issues](#)

- [Dependencies](#)

About Java ES Release 4

Java ES Release 4 Directory Proxy Server represents only minor bug fixes and improvements. There are no new functional capabilities.

Java ES Release 4 Upgrade Roadmap

[Table 4-1](#) shows the supported Directory Proxy Server upgrade paths to Java ES Release 4.

Table 4-1 Upgrade Paths to Java ES Release 4:
Sun Java System Directory Proxy Server 5 2005Q4

Java ES Release	Directory Proxy Server Version	General Approach	Re-configuration Required
Release 3	Sun Java System Directory Proxy Server 5 2005Q1	Direct upgrade: Apply patches and re-configure configuration directory.	Automatic re-configuration of data in configuration directory

Directory Proxy Server Data

Directory Proxy Server makes use of Directory Server for storing configuration data. The data is stored in a specific tree structure within the directory. The Directory Server instance hosting the configuration is referred to as the configuration directory.

In most deployment architectures, the configuration directory is remote from the other components that use it to store configuration information.

The following table shows the type of data that could be impacted by an upgrade of Directory Proxy Server software.

Table 4-2 Directory Proxy Server Data Usage

Type of Data	Location	Usage
Directory Proxy Server configuration data	Configuration directory	Configuration of Directory Proxy Server

Compatibility Issues

Java ES Release 4 Directory Proxy Server does not introduce any interface changes and is backwardly compatible with earlier versions.

Dependencies

Dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Directory Proxy Server software. Directory Proxy Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)). Directory Proxy Server provides front-end access to Directory Server and uses Administration Server for configuration purposes. Directory Proxy Server therefore has dependencies on both Directory Server and Administration Server.

Upgrading Directory Proxy Server from Java ES Release 3

This section includes information about upgrading Directory Proxy Server from Java ES 2005 Q1 (Release 3) to Java ES 2005Q4 (Release 4). The section covers the following topics:

- [Introduction](#)
- [Directory Proxy Server Upgrade](#)
- [Multiple Instance Upgrades](#)

Introduction

When upgrading Java ES Release 3 Directory Proxy Server to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Java ES Release 3 version. Re-configuration of Directory Proxy Server is achieved by automatically synchronizing the configuration directory with the upgraded software.
- **Upgrade Dependencies.** While Directory Proxy Server has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Java ES Release 4 Directory Proxy Server is compatible with the Release 3 versions of these shared components. Upgrade of these shared components is therefore optional with respect to upgrade of Directory Proxy Server to Release 4.

Directory Proxy Server has a hard upgrade dependency on both Directory Server and Administration Server. All three components should therefore be upgraded together to Release 4.

- **Backward Compatibility.** Release 4 Directory Proxy Server is backwardly compatible with its Release 3 version.
- **Upgrade Rollback.** A rollback of the Release 4 upgrade is achieved Release 4 upgrade patches.

Directory Proxy Server Upgrade

This section describes how to perform an upgrade of Directory Proxy Server from Java ES Release 3 to Java ES Release 4 on HP-UX platform. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Directory Proxy Server](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Directory Proxy Server, you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Directory Proxy Server using the following commands:

```
cd serverRoot/bin/dps/server/bin
./ldapfwd -v
```

The output is shown in the following table:

Table 4-3 Directory Proxy Server Version Verification Outputs

Java ES Release	Directory Proxy Server Version Number
Release 3	Sun ONE Directory Proxy Server Version 5.2_Patch_3
Release 4	Sun ONE Directory Proxy Server Version 5.2_Patch_4

Upgrade Directory Proxy Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4.

Directory Proxy Server has hard upgrade dependencies on Directory Server and Administration Server, even when they run on remote computers, so these components should be upgraded before upgrading Directory Proxy Server.

Upgrading of Java ES Release 3 shared components upon which Directory Proxy Server depends is optional, but recommended.

You can upgrade Directory Proxy Server dependencies in the following order, all before you upgrade Directory Proxy Server. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server”](#) on page 61.

Back Up Directory Server Data

The Directory Proxy Server upgrade process modifies configuration directory data. Therefore, before you upgrade, it is recommended that you back up your configuration directory data using the Directory Server Console or a command-line utility such as `db2bak`.

For more information about backing up Directory Server, see the *Sun Java System Directory Server Administration Guide* (<http://docs.sun.com/doc/817-7613>).

Obtain Required Configuration Information and Passwords

Directory Proxy Server must run as the same user and group as Directory Server and Administration Server. That is, they must all run with the same UID and GID.

Upgrading Directory Proxy Server

This section discusses considerations that impact the upgrade procedure for Directory Proxy Server followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Directory Proxy Server software to Java ES Release 4 takes into account the following considerations:

- Any Java ES components using a Directory Proxy Server instance (such as Access Manager, Communications Express, Messaging Server, Portal Server, and so forth) should be shut down before you upgrade that instance. However, many deployment architectures use multiple instances of Directory Proxy Server to provide high availability or scalability. In such cases, you can perform a rolling upgrade of Directory Proxy Server and the Directory Proxy Server clients need not be shut down.
- The upgrade of Directory Proxy Server should only be performed after the upgrade of Administration Server and Directory Server because re-configuration must take place in a particular order.
- Directory Proxy Server must be shut down when patches are being applied, however the associated configuration directory must be running to perform re-configuration.
- In a deployment architecture in which there are multiple instances of Directory Proxy Server running on a single computer (all corresponding to the same installed Directory Proxy Server image), upgrading the Directory Proxy Server image will upgrade all the instances. In such architectures, there is only one Administration Server instance per installed Directory Proxy Server image.
- The Release 4 Directory Proxy Server upgrade patches for HP-UX are shown in the following table:

Table 4-4 Patches* to Upgrade Directory Proxy Server

Component	Patch-ID
Directory Proxy Server	121521-01
Directory Proxy Server locale	121932-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to Directory Proxy Server instances residing locally on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 4-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>.

2. Log in as superuser.

```
su root
```

3. Stop the Administration Console if it is running locally.
4. Shut down all Java ES components dependent on the Directory Proxy Server instances that are to be upgraded. This step might depend on how Directory Proxy Server is replicated within your deployment architecture.

For information about how to shut down a Java ES component, see its respective administration guide.

5. Make sure you have upgraded any Java ES components upon which Directory Proxy Server has hard upgrade dependencies (see “[Upgrading Directory Proxy Server](#)” on page 76).
6. Upgrade Directory Proxy Server.

- a. Ensure that the configuration directory is running.

If it is local you might have to start it up. If it is remote, check to make sure it is running.

- b. Apply the Directory Proxy Server patches in [Table 4-4](#).

```
swinstall -s $<path_to_DirectoryProxyServer_patch>/<patch_ID>  
<patch_ID>
```

7. Restart Directory Proxy Server and all Java ES components dependent on Directory Proxy Server.

To restart Directory Proxy Server:

```
serverRoot/dps-hostName/restart-dps
```

Verifying the Upgrade

You can verify successful upgrade of Directory Proxy Server using the following commands:

```
cd serverRoot/bin/dps/server/bin  
./ldapfwd -v
```

See [Table 4-3 on page 75](#) for output values.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in “[Upgrade Procedure](#)” on page 77.

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Directory Proxy Server, followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Directory Proxy Server is pretty much the reverse of the procedure for upgrading to Release 4. The patches are removed and the configuration directory is re-synchronized.

One special consideration is that when you apply patches, you upgrade the SSL certificate database to a cert8 format. The patch backs up the cert7 data, and then converts it to cert8 format. If you subsequently decide to roll back the upgrade and have added new certificates to the certificate database, you should manually extract these certificates, back out the patches, and then add the certificates back to the previous cert7 format certificate database.

When you roll back an upgrade after having changed the SSL certificate database, you cannot start in SSL mode. To work around this problem, turn off SSL mode, restart Administration Server and Directory Proxy Server, reinstall the certificate, and then enable SSL mode.

Rollback Procedure

1. Log in as superuser.

```
su root
```

2. Stop the Administration Console if it is running locally.

3. Shut down all Java ES components dependent on the Directory Proxy Server instances that are to be upgraded. This step might depend on how Directory Proxy Server is replicated within your deployment architecture.

For information about how to shut down a Java ES component, see its respective administration guide.

4. Roll back the Directory Proxy Server upgrade.
 - a. Ensure that the configuration directory is running.
If it is local you might have to start it up. If it is remote, check to make sure it is running.
 - b. Remove the Directory Proxy Server patches in [Table 4-4 on page 76](#).

```
swremove <patch_ID>
```

5. Roll back upgrades to any Java ES components upon which Directory Proxy Server has hard upgrade dependencies, in particular Directory Server and Administration Server.
6. Restart Directory Proxy Server and all Java ES components dependent on Directory Proxy Server.

Multiple Instance Upgrades

In some deployment architectures Directory Proxy Server is deployed on multiple computer systems to provide for scalability and to improve availability. For example, you might have Directory Proxy Server components running on multiple computers with a load balancer to distribute the load.

In the case of load-balanced instances of Directory Proxy Server, you can perform a rolling upgrade in which you upgrade the Directory Proxy Server instances sequentially without interrupting service. You upgrade each instance of Directory Proxy Server while the others remain running. You perform the upgrade of each instance as described in [“Directory Proxy Server Upgrade” on page 74](#).

Web Server

This chapter describes how to upgrade Web Server to Java ES Release 4 Web Server 6.1 2005Q4 SP5. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Web Server Upgrades](#)
- [Upgrading Web Server from Java ES Release 3](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *web_svr_base*. At least part of this path might have been specified as an installation directory when Web Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *web_svr_base* is `/opt/sun/webserver`.

Overview of Web Server Upgrades

This section describes the following general aspects of Web Server that impact upgrading to Java ES Release 4:

- [About Java ES Release 4 Web Server](#)
- [Web Server Upgrade Roadmap](#)
- [Compatibility Issues](#)
- [Web Server Dependencies](#)

About Java ES Release 4 Web Server

Java ES Release 4 versions of Web Server represents a number of bug fixes, including security fixes that depend upon the NSS shared component. For details, see the appropriate release notes.

Web Server Upgrade Roadmap

[Table 5-1](#) shows the supported Web Server upgrade paths to Java ES Release 4.

Table 5-1 Upgrade Paths to Java ES Release 4 Web Server 6.1 2005Q4 SP5 Java ES

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Web Server 6.1 2005Q1 Update 1 SP 4	Direct Upgrade: Apply patches.	None

Compatibility Issues

Release 4 Web Server does not introduce any interface changes and is backwardly compatible with Release 3.

Web Server Dependencies

Web Server has no dependencies on other Java ES components other than on Java ES shared components (see [Table 1-6](#)).

Upgrading Web Server from Java ES Release 3

This section includes information about upgrading Web Server from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Web Server Upgrade](#)

Introduction

When upgrading Java ES Release 3 Web Server to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. No re-configuration of Web Server is required in upgrading from Java ES Release 3 Web Server to Release 4.
- **Upgrade Dependencies.** Web Server has dependencies on a number of Java ES shared components [Table 1-6 on page 33](#). Before upgrading the Web Server upgrade the following shared Java ES shared components.
 - NSPR
 - NSS
 - SASL
 - LDAP J SDK
 - ICU
- **Backward Compatibility.** Release 4 Web Server is backwardly compatible with the Release 3 version.
- **Upgrade Rollback.** Rollback of the Release 4 upgrade to Release 3 is achieved by removing the patches applied during the upgrade.

Web Server Upgrade

This section describes how to perform an upgrade of Web Server from Java ES Release 3 to Java ES Release 4 on HP-UX. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Web Server](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Web Server you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Web Server by starting the Web Server instance server with the `-version` option:

```
# web_svr_base/https-hostName.domainName/start -version
```

Table 5-2 Web Server Version Verification Outputs

Java ES Release	Web Server Version Number
Release 3	6.1SP4
Release 4	6.1SP5

Upgrade Web Server Dependencies

The Java ES shared components listed in [page 83](#) should be upgraded to Java ES Release 4. Instructions for upgrading Java ES shared components to Release 4, are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on [page 45](#).

Back Up Web Server Data

The Web Server upgrade from Release 3 to Release 4 does not modify configuration data. There is no need to back up current data.

Obtain Required Configuration Information and Passwords

No special information about your currently installed version is needed. However you will have to log in as superuser to perform the upgrade.

Upgrading Web Server

This section discusses considerations that impact the upgrade procedure for Web Server followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Web Server software to Java ES Release 4 takes into account the following considerations:

- Any J2EE components running in an Web Server instance should be shut down before you upgrade that instance.
- All Web Server instances corresponding to the same installed Web Server image are upgraded at the same time. All such instances should be shut down when patches are being applied to the installed image.
- The Release 4 upgrade patches for HP-UX are shown in the following table:

Table 5-3 Patches* to Upgrade Web Server

Component	Patch ID
Web Server	121510-01
Web Server Locale	121935-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the above table.

Upgrade Procedure

The procedure documented below applies to all Web Server instances corresponding to the same installed Web Server image on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 5-3](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>

2. Log in as superuser.
3. Stop all running instances of Web Server and the Administration Server.

```
# web_svr_base/https-instanceName/stop
```

```
# web_svr_base/https-admserv/stop
```

4. If not already done, upgrade the shared components listed on [page 83](#) to Release 4.
5. Apply the appropriate Web Server patch in [Table 5-3](#).

```
swinstall -s <patch_location> -x patch_match_target = true
```

.

NOTE If you have Java1.5.0 installed on your computer, uninstall Java 1.5.0 and clean up the installed directory.

6. Install Java1.5.0.01.
7. Restart the Web Server instances that were stopped in [Step 3](#).

Verifying the Upgrade

You can verify the upgrade of Web Server to Release 4 by starting the Web Server instance server with the `-version` option:

```
# web_svr_base/start -version
```

See [Table 5-2](#) for output values.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in the above “[Upgrade Procedure](#)” on page 85.

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Web Server followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Web Server is pretty much the reverse of the procedure for upgrading to Release 4. The patches are removed.

Rollback Procedure

1. Log in as superuser.
2. Stop all running instances of Web Server and the Administration Server.

```
# web_svr_base/https-instancename/stop
```

```
# web_svr_base/https-admserv/stop
```

3. Remove the Web Server patch in [Table 5-3](#).

```
# swremove <patch_ID>
```

4. Remove the shared component patches.

```
# swremove <patch_ID of Shared Component>
```

5. Restart the Web Server instances that were stopped in [Step 2](#).

Message Queue

This chapter describes how to upgrade Message Queue software from previous Java ES versions to Java ES 2005 (Release 4): Sun Java System Message Queue 3 Enterprise Edition 2005Q4.

The chapter provides a general overview of Message Queue upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Message Queue Upgrades](#)
- [Upgrading Message Queue from Java ES Release 3](#)

NOTE Message Queue commands used in this chapter are run with respect to the directory location of executable files, which depends on operating system platform:

The default value is `/opt/sun/mq/bin`.

Overview of Message Queue Upgrades

This section describes the following general aspects of Message Queue that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Message Queue](#)
- [Message Queue Upgrade Roadmap](#)
- [Message Queue Data](#)
- [Compatibility Issues](#)
- [Message Queue Dependencies](#)

About Java ES Release 4 Message Queue

Java ES Release 4 Message Queue represents minor code fixes with no new features or enhancements. As such, Release 4 does not introduce any new compatibility issues (see “[Compatibility Issues](#)” on page 89).

Message Queue software includes two editions, a Platform Edition and an Enterprise Edition, each corresponding to a different feature set and licensed capacity. Enterprise edition is for deploying and running messaging applications in an enterprise production environment. Platform Edition is mainly for developing, debugging, and load testing messaging applications and components.

Message Queue Upgrade Roadmap

[Table 6-1](#) shows the supported Message Queue upgrade paths to Java ES Release 4.

Table 6-1 Upgrade Paths to Java ES Release 4 Message Queue 3.6 SP3 2005Q4

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Message Queue 2005Q2 (3.6) Enterprise Edition only	Direct upgrade: Performed using the mqupgrade script.	None

Message Queue Data

Message Queue, like other Java ES components, makes use of various kinds of data that for any specific upgrade might need to be migrated to an upgraded version. The following table shows the type of data that could be impacted by an upgrade of Message Queue software.

[Table 6-2](#) shows the location of data on HP-UX system. In [Table 6-2](#) instanceName identifies the name of the Message Queue broker instance with which the data is associated.

Table 6-2 Message Queue Data Usage

Data Category	Location	Usage
Broker instance configuration properties	<code>/var/opt/sun/mq/instances/instanceName/props/config.properties</code>	Broker and related services configurations
Persistent store for dynamic application data	<code>/var/opt/sun/mq/instances/instanceName/fs350/</code> or a JDBC-accessible data store	Stores messages, destinations, durable subscriptions, transactions, and other dynamic data
Administered objects (object store)	local directory of your choice or an LDAP Directory Server or an LDAP Directory Server	Objects used to configure client/broker connections
Security: user repository	<code>/var/opt/sun/mq/instances/instanceName/etc/passwd</code> or an LDAP Directory Server	Stores user data used for authentication and authorization
Security: access control file (default location)	<code>/var/opt/sun/mq/instances/instanceName/etc/accesscontrol.properties</code>	Sets the rules that authorize user access to destinations and related capabilities
Security: passfile directory (default location)	<code>/var/opt/sun/mq/instances/instanceName/etc/</code>	Stores encrypted password information.
Security: broker's keystore file location	<code>/var/opt/sun/mq</code>	Stores encrypted certificate information for secure messaging.

Compatibility Issues

Release 4 Message Queue introduces no new incompatibilities over Release 3. The following general Message Queue compatibility issues relate to versions earlier than Release 3.

Protocol Compatibility

Message Queue has a dependency on a web container to provide HTTP protocol support between Message Queue clients and broker. Due to a protocol change, when using Sun Java System Web Server to provide a web container for the Message Queue `imqhttp.war` application, you cannot upgrade the Web Server component without also upgrading Message Queue (see [“Post-Upgrade Tasks” on page 95](#)).

Broker Compatibility

A Release 4 Message Queue broker will inter-operate with a Release 3 broker.

Release 4 Message Queue can use Release 3 and Release 2 data, except that on Linux systems, Release 2 data must be first migrated to Release 4.

When updating to Release 4 Message Queue, consider the following:

- You can use earlier Message Queue `config.properties` files. You can also copy them to another location and consult the property settings they contain when you configure Release 4 Message Queue brokers.
- Any persistent Message Queue data—messages, destinations, durable subscriptions—is automatically converted, if necessary, to Release 4 Message Queue data when starting up a broker for the first time. For example, any existing destinations will be converted, if necessary, to Release 4 Message Queue destinations, preserving existing attributes and using default values of new attributes.

Administered Object Compatibility

Release 4 Message Queue administered objects are identical to Release 3 administered objects. However, some Release 3 administered objects were renamed or enhanced with new attributes with respect to earlier versions.

Administration Tool Compatibility

Because of the addition of new commands and new administrative capabilities in Release 3, the Release 4 administration tools (the Administration Console and command line utilities) only work with Release 3 and Release 4 brokers.

Client Compatibility

Release 3 clients are completely compatible with Release 4 Message Queue.

Message Queue Dependencies

Message Queue dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Message Queue software. Changes in Message Queue interfaces or functions, for example, could require upgraded version of components upon which Message Queue depends. The need to upgrade such components depends upon the specific upgrade path.

Message Queue has dependencies on the following Java ES components:

- **Shared components.** Message Queue has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Directory Server (optional).** If you want to configure Message Queue to store administered objects and/or user data in an LDAP directory rather than locally, then you can use Directory Server for that purpose.
- **Web Container (optional).** If you need HTTP messaging between client and broker, then Message Queue requires web container support from Java ES Web Server, Java ES Application Server, or third-party web containers.

Upgrading Message Queue from Java ES Release 3

This section includes information about upgrading Message Queue from Java ES 2005Q1 (Release 3) to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Message Queue Upgrade](#)
- [Multiple Instance Upgrades](#)

Introduction

When upgrading Java ES Release 3 Message Queue to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed using an `mqupgrade` script that replaces previous software packages with new ones and retains configuration data from Release 3 automatically.

- **Upgrade Dependencies.** While Message Queue has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Message Queue is compatible with the Release 3 versions of all these components. Upgrade of these shared components is therefore optional with respect to upgrade of Message Queue to Release 4.

In addition, Release 4 Message Queue is optionally dependent on Directory Server and Web Server (or Application Server), as described in “[Message Queue Dependencies](#)” on page 91. However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Message Queue to Release 4.

- **Backward Compatibility.** Release 4 Message Queue is fully compatible with Release 3 (see “[Compatibility Issues](#)” on page 89).
- **Upgrade Rollback.** There is no utility for rolling back the Message Queue upgrade to Release 3. You have to remove the upgraded components and manually restore the previous version and configuration data.

Message Queue Upgrade

This section describes how to perform a Message Queue upgrade from Java ES Release 3 to Java ES Release 4. This section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Message Queue](#)
- [Verifying the Message Queue Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Message Queue, perform the procedures described in the following sections.

Verify Current Version Information

You can determine the version and edition of Message Queue installed on your system by starting the Message Queue broker with the `-version` option:

```
/opt/sun/mq/bin/imqbrokerd -version
```

Table 6-3 Message Queue Version Verification Outputs

Java ES Release	Message Queue Version Number
Release 3	Sun Java System (tm) Message Queue 3 2005Q1 Version: 3.6
Release 4	Sun Java System (tm) Message Queue 3 2005Q1 Version: 3.6 SP3

Upgrade Message Queue Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, because Message Queue does not require upgrading the Java ES Release 3 components upon which it depends, this task is optional.

However, if you choose to upgrade all Message Queue dependencies, they should be upgraded in the following order, all before you upgrade Message Queue. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.
2. **Directory Server (optional).** Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server”](#) on page 61.
3. **Web Container Software (optional).** Instructions for upgrading Web Server or Application Server are provided in [Chapter 5, “Web Server”](#) on page 81 and [Chapter 8, “Application Server”](#) on page 107, respectively.

Back Up Message Queue

There is no script for rolling back Message Queue to its previous state. Because Release 4 data is compatible with Release 3 data, there is no reason to back up configuration data. In addition, there is no reason to back up the installed image because you can use the Release 3 installer if you need to roll back Release 4 Message Queue to Release 3.

Upgrading Message Queue

The upgrade of Message Queue software to Java ES Release 4 makes use of the `mqupgrade` script, which installs freshbitted packages.

The upgrade procedure consists of the following steps:

1. Stop any running Message Queue client applications.

If Message Queue is being used in an Application Server environment, shut down Application Server, as well.

2. Stop any running brokers. You will be prompted for the admin user name and password.

```
imqcmd shutdown bkr [-b hostName:port]
```

3. If you do not want to preserve dynamic data, the Message Queue flat-file user repository, and the Message Queue access control file associated with each broker instance, remove this data using the following command.

```
imqbrokerd -name instanceName -remove instance
```

4. Log in as Root.

```
su root
```

5. Change directories to the location of the `Tools` directory of the Java ES distribution.

```
cd HPUX_PA-RISC/Product/message_queue/Tools
```

6. Run the `mqupgrade` script.

- a. Start the script:

```
./mqupgrade
```

The `mqupgrade` script lists installed Message Queue components.

- b. Enter `y` (yes) to upgrade Message Queue components.

The `mqupgrade` script detects and lists installed locale files.

If you do not want to upgrade Message Queue components, enter `n` (no). The `mqupgrade` script will exit without upgrading Message Queue components.

- c. If prompted, enter `y` (yes) to upgrade locale files.

The `mqupgrade` script sends output to a log file in the following location:

```
/var/opt/sun/install/logs/Message_Queue_upgrade_'date'.log
```

Verifying the Message Queue Upgrade

After you finish the upgrade procedure, verify that it was successful by starting the Message Queue broker with the `-version` option.

```
imqbrokerd -version
```

The command returns the Java ES version number as well as the Message Queue-specific version number.

Post-Upgrade Tasks

If you have upgraded the web container and are using the Message Queue HTTP tunneling servlet, you may need to re-deploy it in the new web container. Otherwise, there has been no change to the HTTP tunneling servlet between Release 3 and Release 4, and you do not need to re-deploy it after upgrading Message Queue to Release 4. See the *Message Queue Administration Guide*, (<http://docs.sun.com/doc/819-2571>) for more information on HTTP support.

Rolling Back the Upgrade

No scripts are provided for rolling back Message Queue to its pre-upgrade state. The process must be performed manually using the following steps:

1. Stop any running Message Queue client applications.

If Message Queue is being used in an Application Server environment, shut down Application Server, as well.

2. Stop any running brokers. You will be prompted for the admin user name and password.

```
imqcmd shutdown bkr [-b hostName:port]
```

3. If you want to delete dynamic data, the Message Queue flat-file user repository, and the Message Queue access control file associated with each broker instance, remove this data using the following command.

```
imqbrokerd -name instanceName -remove instance
```

4. Log in as superuser.

```
su root
```

5. Retrieve the list of installed Message Queue packages with the following command:

```
swlist -l product sun-mq*
```

6. Remove the Message Queue packages, using the following command:

```
swremove depotName
```

where `depotName` is any of the Message Queue packages. To remove multiple packages, separate the package names by a space.

Because other products might be using Message Queue packages, be careful about removing them.

7. Type “`q`” to quit.
8. Exit the root shell.
9. Re-install Release 3 Message Queue, using the Java ES Release 3 Installer
Release 4 Message Queue data will work fine.

Multiple Instance Upgrades

To upgrade a Message Queue cluster, in which multiple brokers interact to provide a scalable message service, you can do a rolling upgrade in which the cluster remains online as each Message Queue instance is upgraded from Release 3 to Release 4. The two conditions to keep in mind when performing a cluster upgrade are:

- While a broker is shut down for upgrade, the persistent messages it is storing are not available until the broker is restarted.
- The Master broker should be upgraded last.

Otherwise the procedure is straightforward: you shut down, upgrade, and restart the brokers one at a time until all have been upgraded.

High Availability Session Store

This chapter describes how to upgrade High Availability Session Store to Java ES 2005Q4 (Release 4): High Availability Session Store (HADB) 4.4.2.

The chapter provides a general overview of upgrade issues before covering the different upgrade paths supported by Java ES Release 4 on Solaris.

- [“Overview of HADB Upgrades” on page 98](#)
- [“Upgrading HADB from Java ES Release 3” on page 99](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *HADB-base*. At least part of this path might have been specified as an installation directory when HADB was initially installed. If not, the Java ES installer assigned a default value.

The value of *HADB-base* is in relation to the Application Server directory structure, as follows:

ApplicationServer-base/hadb/*version_number*

The default value of *HADB-base* depends on the default value of *ApplicationServer-base*, which depends on operating system platform:

- Solaris: /opt/SUNWappserver/appserver/hadb/*version_number*
 - Linux: /opt/sun/appserver/hadb/*version_number*
-

Overview of HADB Upgrades

This section describes the following general aspects of HADB that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 HADB](#)
- [HADB Upgrade Roadmap](#)
- [HADB Data](#)
- [Compatibility Issues](#)
- [HADB Dependencies](#)

About Java ES Release 4 HADB

Java ES Release 4 versions of HADB represents bug fixes to the Java ES 2005Q1 (Release 3) version.

HADB Upgrade Roadmap

[Table 7-1](#) shows the supported HADB upgrade paths to Java ES Release 4. The table applies to both Solaris and Linux operating systems.

Table 7-1 Upgrade Paths to Java ES Release 4:
HADB 4.4.2 2005Q4

Java ES Release	HADB Version	General Approach	Re-configuration Required
Release 3	HADB 4.4.1 (2005Q1)	Direct upgrade: Online and Offline upgrades are both available.	None

HADB Data

The following table shows the type of data that could be impacted by an upgrade of HADB software.

Table 7-2 HADB Data Usage

Type of Data	Location	Usage
Dynamic application data	/var/opt/SUNWhadb and /etc/opt/SUNWhadb	High availability session store and configuration information.

Compatibility Issues

HADB provided with Java ES Release 4 is backwardly compatible with HADB provided with Java ES Release 3.

HADB Dependencies

HADB provided with Java ES Release 4 requires Java™ 2 Platform, Standard Edition (J2SE™) Version 5.0 or later.

Upgrading HADB from Java ES Release 3

This section includes information about upgrading HADB from Java ES 2005Q1 (Release 3) to Java ES 2005Q4 (Release 4). The section covers the following topics:

- [Introduction](#)
- [HADB Upgrade](#)

Introduction

When upgrading Java ES Release 3 HADB to Java ES Release 4, consider the following aspects of the upgrade process:

General Upgrade Approach. Upgrades consist of removing the Java ES Release 3 HADB packages and adding the Java ES Release 4 packages. There are two upgrades approaches available:

- **Online upgrade.** Use online upgrade to avoid interruption of HADB services.

- **Offline upgrade.** Use offline upgrade if you can interrupt HADB services when replacing HADB packages with newer versions.
- **Upgrade Dependencies.** HADB requires J2SE Version 5.0 or later.
- **Backward Compatibility.** HADB provided with Java ES Release 4 is backwardly compatible with HADB provided with Java ES Release 3.
- **Upgrade Rollback.** Rollback from the Java ES Release 4 upgrade to Java ES Release 3 is achieved by restoring the Release 3 version, which is left undisturbed in a separate directory by the upgrade to Release 4.
- **Platform Issues.** The general approach for upgrading HADB is the same on both Solaris and Linux operating systems.

HADB Upgrade

This section describes how to perform an upgrade of HADB from Java ES Release 3 to Java ES Release 4 on both the Solaris and Linux platform. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading HADB](#)
- [Verifying the Upgrade](#)
- [Post-UpgradeTasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade HADB you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of HADB using standard version checking utilities. For example:

Solaris:

```
pkgparam -v SUNWhadba
```

Linux:

```
rpm -qi sun-hadb-a-4.4.2-7.i386.rpm
```

Table 7-3 HADB Version Verification Outputs

Java ES Release	HADB Version Number
Release 2	VERSION=4.4.0,REV=14 SUNW_PRODVERS=4.4.0
Release 3	VERSION=4.4.1,REV=7 SUNW_PRODVERS=4.4.1
Release 4	VERSION=4.4.2,REV=7 SUNW_PRODVERS=4.4.2

Upgrade HADB Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. The upgrade of HADB to Release 4 depends on J2SE 5.0 or later.

Back Up Directory Data

The HADB upgrade from Java ES Release 3 to Java ES Release 4 does not in itself modify HADB dynamic data. However, you can back up the Java ES Release 3 packages in case you need to roll back the upgrade.

Obtain Required Configuration Information and Passwords

HADB upgrade requires you to know the superuser password.

Upgrading HADB

This section discusses considerations that impact the upgrade procedure for HADB followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of HADB software to Java ES Release 4 takes into account the following considerations:

- Based on your production requirements, you need to determine whether an online or offline upgrade is more appropriate.
- The Java ES Release 4 upgrade packages for Solaris and Linux platforms are shown in the following table. Solaris packages are listed in their installation sequence.

Table 7-4 Package Versions for Upgrading HADB on Solaris Platforms

Solaris Packages	Linux Packages
SUNWhadba	sun-hadb-a-4.4.2-7.i386.rpm
SUNWhadbc	sun-hadb-c-4.4.2-7.i386.rpm
SUNWhadbe	sun-hadb-e-4.4.2-7.i386.rpm
SUNWhadbi	sun-hadb-i-4.4.2-7.i386.rpm
SUNWhadbj	sun-hadb-j-4.4.2-7.i386.rpm
SUNWhadbm	sun-hadb-m-4.4.2-7.i386.rpm
SUNWhadbo	sun-hadb-o-4.4.2-7.i386.rpm
SUNWhadbs	sun-hadb-s-4.4.2-7.i386.rpm
SUNWhadbv	sun-hadb-v-4.4.2-7.i386.rpm
SUNWhadbx	sun-hadb-x-4.4.2-7.i386.rpm

Online Upgrades of HADB

Online upgrades of HADB are only available when upgrading from Java ES Release 3.

When you perform an online upgrade of HADB, you first install HADB on each server in the cluster being upgraded. Each server first unregisters from an earlier installation of HADB and then registers with the newly installed version of HADB.

For details on performing an online upgrade, refer to the following section in *Sun Java System Application Server Enterprise Edition 8.1 2005Q2 High Availability Administration Guide* (<http://docs.sun.com/doc/819-2555/6n4r9qo7n?a=view>).

Offline Upgrades of HADB

An offline upgrade of HADB is available when upgrading from either Java ES Release 3.

To perform an offline upgrade, shut down your HADB services and replace the existing HADB packages with the newer versions available from your Java Enterprise System Release 4 distribution, shown in [Table 7-4 on page 102](#).

Use the below procedures.

► **To upgrade packages for shared components on Solaris platforms**

1. Log in as superuser or become superuser by using the `su -` command.

NOTE Become superuser with `su -`, not `su`, to make sure you obtain the complete superuser environment.

2. Navigate to the location of the packages in your Java ES Release 4 distribution.

The packages are found under one of the following directories, depending on your platform:

`Solaris_sparc/Product/shared_components/Packages/`

`Solaris_x86/Product/shared_components/Packages/`

NOTE If you have a subset distribution of Java Enterprise System, this subset distribution contains all the necessary shared components at the location indicated above.

3. Remove the current versions of the packages that you are upgrading by using the `pkgrm` command.

For example, to remove packages for HADB:

```
pkgrm SUNWhadba SUNWhadbc SUNWhadbe SUNWhadbi SUNWhadbj SUNWhadbm
SUNWhadbo SUNWhadbs SUNWhadbv SUNWhadbx
```

For detailed information on removing packages, refer to the `pkgrm(1m)` man page.

4. Install the packages from your distribution using the `pkgadd` command.

For example, to install packages for HADB:

```
pkgadd SUNWhadba SUNWhadbc SUNWhadbe SUNWhadbi SUNWhadbj SUNWhadbm
SUNWhadbo SUNWhadbs SUNWhadbv SUNWhadbx
```

For detailed information on removing packages, refer to the `pkgadd(1m)` man page.

5. Verify that the package is correctly installed using the `pkgparam` and `pkginfo` commands.

Use `pkgparam` with the `-v` option to verify the version. `pkginfo` provides additional information about the package.

For detailed information on these commands refer to the `pkgparam(1)` and `pkginfo(1)` man pages.

Verifying the Upgrade

After completing the online upgrade, verify the upgrade by using the following procedure. After verifying that the upgrade is successful, the old installation packages can be deleted.

To verify that running processes are using the upgraded HADB services, you can perform the following steps.

1. For all HADB services running, issue either of the following commands:

```
HADB-base/bin/ma -V
```

```
HADB-base/bin/hadbm -V
```

For example, on the Solaris 8 platform:

```
HADB-base/bin/ma -V
```

```
Sun Java System High Availability Database 4.4 Database Management  
Agent Version : 4.4.2.7 [V4-4-2-7 2005-05-26 13:49:01 server@domain]  
\(SunOS_5.8_sparc)
```


2. Check whether the database is running by issuing the commands in the following example for a database named ExampleDB:

```
HADB-base/bin/hadbm status -n databaseName
```

```
HADB-base/bin/hadbm list
```

```
Database
```

```
ExampleDB
```

```
HADB-base/bin/hadbm status ExampleDB
```

```
Database Status
```

```
ExampleDB FaultTolerant
```

```
HADB-base/bin/hadbm status -n ExampleDB
```

```
NodeNo HostName Port NodeRole NodeState MirrorNode
```

```
0 sungod012 15000 active running 1
```

```
1 sungod012 15020 active running 0
```

All HADB services for listed nodes should in the “running” state.

3. Verify that all products using HADB are using the new HADB path by issuing the command in following example for a database named ExampleDB:

```
HADB-base/bin/hadbm get PackageName ExampleDB
```

```
Attribute Value
```

```
PackageName V4.4.2.7
```

The above command displays the current version of HADB. For a detailed listing, issue the following command:

```
hadbm get --all ExampleDB
```

Post-UpgradeTasks

There are no post-upgrade tasks beyond the steps described in [“Upgrading HADB from Java ES Release 3” on page 99](#).

Rolling Back the Upgrade

To roll back the upgrade to HADB, replace the newer versions of the HADB packages you installed with the versions you previously had installed. Use the same procedure described in [“Rolling Back the Upgrade” on page 106](#). There is no data or configuration files that need to be changed.

Application Server

This chapter describes how to upgrade Application Server to Java ES 2005Q4 (Release 4): Sun Java System Application Server Enterprise Edition 8.1 2005Q4.

The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Application Server Upgrades](#)
- [Upgrading Application Server from Java ES Release 3](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *ApplicationServer-base*. At least part of this path might have been specified as an installation directory when Application Server was initially installed. If not, the Java ES installer assigned a default value.

The default value is *ApplicationServer-base* depends on Application Server version and HP-UX:

- Release 3 and Release 4 (Application Server 8.1)
ApplicationServer-base: /opt/sun/appserver .
-

Overview of Application Server Upgrades

This section describes the following general aspects of Application Server that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Application Server](#)
- [Application Server Upgrade Roadmap](#)

- [Application Server Data](#)
- [Compatibility Issues](#)
- [Application Server Dependencies](#)

About Java ES Release 4 Application Server

Java ES Release 4 Application Server represents selected bug fixes to the Release 3 version. Functionally Release 4 Application Server is the same as Release 3.

Application Server Upgrade Roadmap

[Table 8-1](#) shows the supported Application Server upgrade paths to Java ES Release 4.

Table 8-1 Upgrade Paths to Java ES Release 4: Sun Java System Application Server Enterprise Edition 8.1 2005Q4

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Application Server Enterprise Edition 8.1 2005Q1	Direct upgrade: Performed by applying patches.	None

Application Server Data

The following table shows the type of data that could be impacted by an upgrade of Application Server software.

Table 8-2 Application Server Data Usage

Type of Data	Location	Usage
Environment variables	<i>ApplicationServer-base</i> /config/asenv	Global Variables
Configuration data	Release 3 & 4: <i>ApplicationServer-base</i> /domains/ <i>domainName</i> /config	Configuration of Application Server instances

Table 8-2 Application Server Data Usage

Type of Data	Location	Usage
Deployment data	Release 3 & 4: <i>ApplicationServer-base/domains/domainName/ applications</i>	Configuration of J2EE container for specific J2EE components and applications.

Compatibility Issues

Release 4 Application Server does not introduce any interface changes with respect to Release 3.

Application Server Dependencies

Application Server dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Application Server software. Changes in Application Server interfaces or functions, for example, could require upgraded versions of components upon which Application Server depends. The need to upgrade such components depends upon the specific upgrade path.

Application Server has dependencies on the following Java ES components:

- **Shared components.** Application Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Message Queue.** Application Server depends on Message Queue to provide J2EE Java Message Service-compliant asynchronous messaging support.
- **Web Container (optional).** Application Server depends upon web container services for its optional load balancing plugin. This support can be provided either by Java ES Web Server or third-party web containers (such as Apache Web Server, and Microsoft IIS).
- **High Availability Session Store (optional).** Application Server depends upon High Availability Session Store to maintain session state information needed to support failover between instances.

Upgrading Application Server from Java ES Release 3

This section includes information about upgrading Application Server from Java ES 2005Q1 (Release 3) to Java ES 2005Q4 (Release 4). The section covers the following topics:

- [Introduction](#)
- [Application Server Upgrade](#)

Introduction

When upgrading Java ES Release 3 Application Server to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. No re-configuration or migration of J2EE components is required to upgrade from Release 3 Application Server to Release 4.
- **Upgrade Dependencies.** While Application Server has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Application Server is compatible with the Release 3 versions of all these components. Upgrade of these shared components is therefore optional with respect to upgrade of Application Server to Release 4.

In addition, Release 4 Application Server is dependent upon Release 4 Message Queue and optionally dependent on Java ES Web Server or third-party web containers, as described in [“Application Server Dependencies” on page 109](#). However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Application Server to Release 4.

Release 4 Application Server is also optionally dependent upon on High Availability Session Store. If being used by Application Server, High Availability Session Store should be upgraded to Release 4. Note that upgrade of High Availability Session Store automatically upgrades the J2SE shared component to Release 4.

Application Server Upgrade

This section describes how to perform an upgrade of Application Server from Java ES Release 3 to Java ES Release 4. This section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Application Server](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Application Server you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Application Server by entering the following command:

```
ApplicationServer-base/bin/asadmin version --verbose
```

Table 8-3 Application Server Version Verification Outputs

Java ES Release	Application Server Version Number
Release 3	Sun Java Enterprise System Application Server Enterprise Edition 8.0.0_01
Release 4	Sun Java Enterprise System Application Server Enterprise Edition

Upgrade Application Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, because the upgrade of Application Server to Release 4 does not require upgrading other Release 3 components, this task is optional. If you choose to upgrade components upon which Application Server depends, those components would generally be upgraded in the following order:

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.
2. **Message Queue.** Instructions for upgrading Message Queue to Release 4 are provided in [Chapter 6, “Message Queue”](#) on page 87.

- 3. Web Container Software.** Instructions for upgrading Web Server or Application Server are provided in [Chapter 5, “Web Server” on page 81](#) and [Chapter 8, “Application Server” on page 107](#), respectively.
- 4. High Availability Session Store.** Instructions for upgrading High Availability Session Store on Solaris are provided in [Chapter 7, “High Availability Session Store” on page 97](#) as the High Availability Session Store instructions are not available on HP-UX.

Back Up Application Server Data

The Application Server upgrade from Release 3 to Release 4 does not modify configuration data. There is no need to back up current data.

Obtain Required Configuration Information and Passwords

You should know the Application Server administrator user ID and password for your currently installed version.

Upgrading Application Server

This section discusses considerations that impact the upgrade procedure for Application Server followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Application Server software to Java ES Release 4 takes into account the following considerations:

- Any J2EE components running in an Application Server instance should be shut down before you upgrade that instance. However, if load balancing provides for high availability or scalability, this requirement can be relaxed.
- All instances of Application Server running on a single computer (all corresponding to the same installed Application Server image) must be shut down while the patch is being applied to the installed image.
- In multiple node deployments, perform the upgrade procedure on each node or computer that hosts Application Server instances.

Table 8-4 Patches* to Upgrade Application Server

Component	Patch-ID
Application Server	121514-01
Application Server locale	121934-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies Application Server instances residing locally on the computer where the upgrade is taking place.

1. Obtain the required patches using the patch numbers and depot names from [Table 8-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>

2. Shut down all J2EE components running in the Application Server instances that are to be upgraded.
3. Shut down all Application Server instances on the computer that is to be upgraded.

```
ApplicationServer-base/bin/asadmin stop-domain domainName
```

4. Install java 1.5.0.01.
5. Apply the appropriate Application Server patch in [Table 8-4](#).

```
swinstall -s <absolute_path_of_depot>/<patch_ID>
```

6. Confirm that the patch upgrade was successful:

```
swlist | grep <patch_ID>
```

7. Restart the Application Server instance.

```
ApplicationServer-base/bin/asadmin start-domain domainName
```

Verifying the Upgrade

You can verify that the patch has been properly applied using the following command:

```
ApplicationServer-base/bin/asadmin version --verbose
```

See [Table 8-3 on page 111](#) for output values.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in “[Upgrade Procedure](#)” on page 113.

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Application Server followed by the procedure itself.

Access Manager

This chapter describes how to upgrade Access Manager software from previous Java ES versions to Java ES 2005Q4 (Release 4): Sun Java System Access Manager 7 2005Q4.

The chapter provides a general overview of Access Manager upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [“Overview of Access Manager Upgrades” on page 115](#)
- [“Upgrading Access Manager from Java ES Release 3” on page 120](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *AccessManager-base*. At least part of this path might have been specified as an installation directory when Access Manager was initially installed. If not, the Java ES installer assigned a default value.

The default value of *AccessManager-base* for HP-UX platform:

HP-UX: `/opt/sun/identity`.

Overview of Access Manager Upgrades

This section describes the following general aspects of Access Manager that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Access Manager](#)
- [Access Manager Upgrade Roadmap](#)
- [Access Manager Data](#)

- [Compatibility Issues](#)
- [Access Manager Dependencies](#)

NOTE Versions of Access Manager that predated Java ES Release 3 were named Identity Server. Hence references to Identity Server in this chapter are to earlier versions of the Java ES Access Manager component.

About Java ES Release 4 Access Manager

Java ES Release 4 Access Manager has been enhanced in major ways. On the back end, the product has been re-architected to support multiple identity repositories, or user data stores. Thus Release 4 Access Manager supports not only an LDAP directory such as Directory Server, but other data storage protocols and formats as well. Release 4 Access Manager includes new interfaces and new services to support the integration of multiple identity repositories.

On the front end, a new Access Manager Console is used to configure the new Access Manager services and identity repositories.

The new functional capabilities and interfaces make Release 4 Access Manager a major new release. In order to provide backward compatibility, Release 4 can be run in legacy mode, which supports the Java ES components that depend on Release 3 Access Manager services (for more information, see “[Compatibility Issues](#)” on page 118).

Access Manager Upgrade Roadmap

Table 9-1 shows the supported Access Manager upgrade paths to Java ES Release 4.

Table 9-1 Upgrade Paths to Java ES Release 4: Sun Java System Access Manager 7 2005Q4

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Access Manager 6.3 2005Q1	Direct upgrade: Performed by removing the Release 3 version and then doing a full installation and re-configuration of Release 4.	Configuration data Customized JSPs for Access Manager console and authentication UI Directory schema

Access Manager Data

Access Manager, like other Java ES components, makes use of various kinds of data that for any specific upgrade might need to be migrated to an upgraded version. The following table shows the type of data that could be impacted by an upgrade of Access Manager software.

Table 9-2 Access Manager Data Usage

Data Category	Location	Usage
Configuration data	<code>/etc/AccessManager-base/config/AMConfig.properties</code>	Broker and related services configurations
	<code>/etc/AccessManager-base/config/serverconfig.xml</code>	
	JAR files for authentication and customized modules <code>AccessManager-base/lib</code>	

Table 9-2 Access Manager Data Usage

Data Category	Location	Usage
Web container configuration	<p>Web Server: server.policy and server.xml files in /opt/sun/webserver/https-hostname/config</p> <p>Application Server: server.policy and domain.xml files in /var/opt/appserver/domain/domain1/config</p> <p>WebSphere and WebLogic: Respective policy and configuration files are modified when Access Manager is configured for these web containers</p>	Configuration of Access Manager web container instance.
Customization data (Web container customized JSP files)	<p>Admin Console: <i>AccessManager-base</i>/web-src/applications</p> <p>Authentication UI: <i>AccessManager-base</i>/web-src/services</p>	Configuration of Access Manager administration interfaces.
Directory schema Services configuration User data	Directory Server	Access Manager provides authentication and authorization services for end users, based on services configuration, user, and policy data that is stored in a directory.
Dynamic application data	None	Access Manager does not persistently store application data such as session state.

Compatibility Issues

The new functional capabilities of Release 4 Access Manager involve the following new interfaces:

- Plug-ins for multiple back-end identity repositories

- New directory information tree structure for storing service configuration information so that authentication properties and authorization policies can be grouped into access control realms that can be associated with a user or group of users.
- New API for Access Manager clients
- New Access Manager Console user interface

Access Manager support for these new interfaces is enabled by configuring Access Manager to run in enhanced (Realm) mode. However, Realm mode is not compatible with the earlier Java ES Release 3 or Release 2 Access Manager. For example, directory data has to be migrated to support Realm mode operation. The enhanced Access Manager Console is needed to support enhanced Access Manager services.

In addition, Realm mode does not support other Java ES components, such as Portal Server, Communications Express, Messaging Server, and others.

To support backward compatibility, Release 4 Access Manager can be configured to run in Legacy mode. With some minor exceptions (see Sun Java System Access Manager 7 2005Q4 Release Notes (<http://docs.sun.com/doc/819-2134>), Legacy mode is backwardly compatible with Release 3 Access Manager.

Legacy mode is necessary to support other Java ES components, as well as older versions of Access Manager policy agents, which cannot interoperate with Access Manager in Realm mode. This incompatibility is an important upgrade consideration, and means in most Java ES deployments, that Access Manager should be upgraded to Release 4 Legacy mode.

Even when configured to run in Legacy mode, however, Release 4 Access Manager is incompatible with Release 3 Delegated Administrator. If Access Manager is upgraded to Release 4, then Delegated Administrator also must be upgraded to Release 4 to provision users for Messaging Server and Calendar Server. However, Messaging Server and Calendar Server do not, themselves, have to be upgraded to Release 4.

Access Manager Dependencies

Access Manager dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Access Manager software. Changes in Access Manager interfaces or functions, for example, could require upgraded version of components upon which Access Manager depends. The need to upgrade such components depends upon the specific upgrade path.

Access Manager has dependencies on the following Java ES components:

- **Shared components.** Access Manager has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)). Access Manager upgrades might depend upon upgraded versions of these shared components.
- **Web Container.** Access Manager depends upon web container services, which can be provided either by Java ES Web Server, Java ES Application Server, or third-party web containers (from Weblogic and WebSphere). Access Manager upgrades must therefore be re-configured for a web container instance. In addition, any customized JSPs for the Access Manager console or for the authentication UI need to be migrated to the upgraded Access Manager environment.
- **Directory Server.** Access Manager stores configuration data and also accesses user data stored in Directory Server. As a result, Access Manager upgrades might require extensions of directory schema.

Upgrading Access Manager from Java ES Release 3

This section includes information about upgrading Access Manager from Java ES 2005Q1 (Release 3) to Java ES 2005Q4 (Release 4). The section covers the following topics:

- [Introduction](#)
- [Full Access Manager Upgrade](#)
- [Multiple Instance Upgrades: Release 3 and Release 4 Co-existence](#)
- [Access Manager SDK-only Upgrades](#)

Introduction

When upgrading Java ES Release 3 Access Manager to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by removing previous versions and newly installing Release 4. An `ampre70upgrade` script is provided for removing the Release 3 version and the Java ES installer is then used to install Release 4. Re-configuration of Access Manager is subsequently performed using the `amconfig` script, and directory schema is migrated using the `amupgrade` script.
- **Upgrade Dependencies.** While Access Manager has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Access Manager is compatible with the Release 3 versions of all these components. Upgrade of these components is therefore optional with respect to upgrade of Access Manager to Release 4.

In addition, Release 4 Access Manager is dependent upon Directory Server and Web Server (or Application Server or third-party web containers), as described in [“Access Manager Dependencies” on page 119](#). However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Access Manager to Release 4.

- **Backward Compatibility.** Release 4 Access Manager is not compatible with Release 3, however it does support a compatible legacy mode (see [“Compatibility Issues” on page 118](#)).
- **Upgrade Rollback.** There is no utility for rolling back the Access Manager upgrade. In fact, the number of re-configurations required to roll back Access Manager to its original state make such a rollback impractical.

Full Access Manager Upgrade

This section describes how to perform a full Access Manager upgrade from Java ES Release 3 to Java ES Release 4:

- [Pre-Upgrade Tasks](#)
- [Upgrading Access Manager](#)
- [Verifying the Access Manager Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Access Manager, perform the procedures described in the following sections.

Verify Current Version Information

You can verify the current version of Access Manager using the following command:

```
AccessManager-base/bin/amadmin --version
```

Table 9-3 Access Manager Version Verification Outputs

Java ES Release	Access Manager Version Number
Release 3	6 2005Q1
Release 4	7 2005Q4

Upgrade Access Manager Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, because Access Manager does not require upgrading the Java ES Release 3 components upon which it depends, this task is optional.

However, if you choose to upgrade all Access Manager dependencies, they should be upgraded in the following order, all before you upgrade Access Manager. You can skip any that might already have been upgraded.

1. Shared Components. Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components” on page 45](#).
2. Directory Server. Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server” on page 61](#).
3. Web Container Software. Instructions for upgrading Web Server or Application Server are provided in [Chapter 5, “Web Server” on page 81](#) and [Chapter 8, “Application Server” on page 107](#), respectively.

If web container software is not upgraded before Access Manager, the upgrade procedure (using the `amconfig` script) will configure and re-deploy Access Manager to the existing web container.

Back Up Directory Server Data

The Access Manager upgrade process uses scripts that modify Directory Server schema. Therefore, before you upgrade Access Manager, back up your Directory Server data using the Directory Server Console or a command-line utility such as `db2bak`.

For more information about backing up Directory Server, see the Sun Java System Directory Server Administration Guide (<http://docs.sun.com/doc/817-7613>).

Back Up Release 3 Access Manager Configuration Information

Because the re-configuration of Release 4 Access Manager software requires the re-configuration of the Release 3 version, it is important to back up configuration files to a known location. The following files should be backed up:

- The `AMConfig.properties` file
`/etc/AccessManager-base/config/AMConfig.properties`
- The `serverconfig.xml` file
`/etc/AccessManager-base/config/serverconfig.xml`
- Web container configuration files:
 - For Web Server: `server.policy` and `server.xml` files located in `WebServer-base/https-hostname/config`
 - For Application Server: `server.policy` and `domain.xml` files located in `/var/ApplicationServer-base/domain/domain1/config`
 - For third-party web containers: the appropriate configuration files
- JAR files for authentication and customized modules.
`AccessManager-base/lib`

Back Up Release 3 Access Manager Log and Debug Files

For the purpose of analyzing system state information, it is a good idea to back up log and debug files so they are not lost. These files are at the following locations:

- Debug files
`/var/AccessManager-base/debug`
- Log files
`/var/AccessManager-base/logs`

Obtain Required Configuration Information and Passwords

To upgrade Access Manager, you must provide specific configuration information, including:

- Access Manager administrator user ID and password
- LDAP user ID and password
- Directory Manager name and password for the Directory Server instance that Access Manager is using

Upgrading Access Manager

The upgrade of Access Manager software to Java ES Release 4 includes procedures for re-configuring Access Manager and for migrating Access Manager data.

Upgrade Summary

The procedure for upgrading Access Manager consists of the following steps:

1. **Remove the Java ES Release 3 Version of Access Manager.** Use the `ampre70upgrade` script.
2. **Use the Java ES Release 4 installer with the Configure Later option.** Use the Java ES Release 4 installer with the Configure Later option.
3. **Upgrade mobile access software.**
4. **Re-customize JSPs for Access Manager.**
5. **Undeploy Access Manager, re-configure, and re-deploy into a Web Container..** Use the `amconfig` script.
6. **Update the directory structure and schema.** Use the `amupgrade` script.

These steps are each documented in the following procedures.

Upgrade Procedures

1. Remove the Java ES Release 3 Version of Access Manager.
 - a. Log in as root to the computer hosting Release 3 Access Manager.
 - b. Change directory to the `platform/Product/identity_svr/Tools` directory in the Java ES Release 4 distribution.

- c. Obtain the values of the following parameters to be requested by the `ampre70upgrade` script:

Table 9-4 Configuration Parameters*: `ampre70upgrade`

Parameter	Value
Directory Server Host	Set the fully qualified name: <i>hostname.domian</i>
Directory Server Port	Specify a non-SSL port number1 Default: 389
Access Manager Administrator DN	Default: <code>uid=amadmin,ou=People,dc=iplanet,dc=com</code>
Access Manager Administrator Password	

*. The pre-upgrade process will not complete successfully if you specify a Directory Server SSL port such as the default SSL value of 636.

- d. Make sure that Directory Server is running or start it if it is not.
- e. Run the `ampre70upgrade` script.

```
./ampre70upgrade
```

The script backs up Access Manager configuration files and removes Release 3 packages.

2. Install the Java ES Release 4 Version of Access Manager.
 - a. Run the Java ES installer on the computer hosting Release 3 Access Manager.
 - b. Select Access Manager from the selection panel.

If a “Conflict” message appears on the screen, it means the Installer has found Access Manager configuration information from the previous version, which is expected. Re-configuration will be performed in subsequent steps. You can ignore this “Conflict” message and proceed.
 - c. Specify the same installation directory in which Release 3 was installed.
 - d. Select the Configure Later option.

- e. Quit the Java ES installer when installation is complete.

NOTE If you are using the Java ES Installer command line interface to install Access Manager, it will automatically install Directory Server software as well. If you are using a remote Directory Server you can uninstall the local Directory Server software using the procedures in the *Java Enterprise System Installation Guide for UNIX*.

3. Upgrade mobile access software.

Access Manager mobile access software needs to be upgraded by patching the Release 3 version. The patches needed are shown in the following table:

Table 9-5 Patches* to Upgrade Access Manager Mobile Access software

Description	Patch id
Mobile Access software	121724-01
Access Manager locale	121929-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

- a. Obtain the required patches using the patch numbers from [Table 9-5](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>.

- b. Log in as superuser.

```
su root
```

- c. Apply the patches.

```
swinstall -f patchlocation <patch_ID>
```

4. Re-customize JSPs for Access Manager.

Re-apply the Release 3 customizations to JSPs for the Access Manager Console and authentication user interface (UI) that you saved under “[Back Up Release 3 Access Manager Configuration Information](#)” on page 123.

Then, copy the customized JSP files to the correct directories. For example, on HP-UX systems:

- o Console: *AccessManager-base*/web-src/applications/console
- o Authentication UI:
 AccessManager-base/web-src/services/config/auth/default or
 AccessManager-base/web-src/services/config/auth/default_*locale*
 (where *Locale* is a locale indicator like ja)

For more information, see the Sun Java System Access Manager Developer’s Guide (<http://docs.sun.com/doc/819-2139>).

5. Undeploy Access Manager, re-configure, and re-deploy into a Web Container.

Configure Access Manager for your specific web container by running the `amconfig` script. The `amconfig` script (and the associated `amsamplesilent` template input file) resides in the following directory:

```
AccessManager-base/bin
```

For information about the `amconfig` script and the `amsamplesilent` template file, see the Sun Java System Access Manager Administration Guide (<http://docs.sun.com/doc/817-7647>).

Perform the following steps to re-configure and re-deploy Access Manager to the web container:

- a. If you choose to upgrade your web container software, as described in “[Upgrade Access Manager Dependencies](#)” on page 122, make sure the upgrade is complete.
- b. Check that Directory Server and the appropriate web container are running.
- c. Create an `amconfig` input file based on the `amsamplesilent` template input file:

```
cp amsamplesilent config-file
```

- d. Set the configuration parameters in `config-file`.

All the parameters need to be set correctly. Some of the values can be migrated from the `AMConfig.properties` file and others are more specific to the upgrade procedure, as shown in the following table.

Table 9-6 Patches to Upgrade Access Manager Mobile Access software

Parameter	Value
Upgrade Parameters	
DEPLOY_LEVEL	26 (for undeploy) or 1 (for re-configure and deploy)
DIRECTORY_MODE	5 (Existing Upgrade)
AM_REALM	set to disabled (Realm Mode is disabled, Legacy Mode is therefore enabled) (Default = enabled)
JAVA_HOME	set to JDK Release 4 directory: <code>/usr/java/jdk1.5.0_04/</code>
WEB_CONTAINER	set to the value appropriate to the web container type you are using and fill out only the corresponding section of <code>config-file</code> .
WS61_INSTANCE (If using Web Server as the web container)	= <code>https-<hostname>.<domain></code> where the value above matches the instance name in <code>/WebServer-base/SUNWsbvsvr/</code> The values is case-sensitive.
Migrated from <code>AMConfig.properties</code>	
SERVER_PROTOCOL	<code>com.iplanet.am.server.protocol</code>
SERVER_PORT	<code>com.iplanet.am.server.port</code>
SERVER_HOST	<code>com.iplanet.am.server.host</code>
DS_HOST	<code>com.iplanet.am.directory.host</code>
DS_PORT	<code>com.iplanet.am.directory.port</code>
ROOT_SUFFIX	<code>com.iplanet.am.defaultOrg</code>
CONSOLE_DEPLOY_URI	<code>com.iplanet.am.console.deploymentDescriptor</code>
SERVER_DEPLOY_URI	<code>com.iplanet.am.services.deploymentDescriptor</code>
PASSWORD_DEPLOY_URI	<code>com.sun.identity.password.deploymentDescriptor</code> or
AM_ENC_PWD	<code>am.encrypted.pwd</code>

For other parameters, provide the same values that were used in the Release 3 configuration that you are upgrading, unless you are changing web container or passwords.

- e. Run `amconfig` to undeploy Access Manager

Set the value of `DEPLOY_LEVEL` in `config-file` to 26.

```
cd /AccessManager-base/bin
./amconfig -s AccessManager-base/bin/config-file
```

- f. Run `amconfig` to reconfigure Access Manager and deploy into web container.

Set the value of `DEPLOY_LEVEL` in `config-file` to 1.

```
cd /AccessManager-base/bin
./amconfig -s AccessManager-base/bin/config-file
```

6. Update the directory structure and schema.

Release 4 Access Manager co-exists with the Release 3 directory structure, but the structure must be modified to support Release 4 capabilities. Update the Access Manager directory structure and schema to Release 4 by running the `amupgrade` script, which is installed in the following directory:

`AccessManager_base/identity/upgrade/scripts`

- a. Obtain the values of the following parameters to be requested by the `amupgrade` script:

Table 9-7 Access Manager Configuration Parameters: `amupgrade`

Parameter	Value
Directory Server Host	Set the fully qualified name: <code>hostname.domain</code>
Directory Server Port	Specify a non-SSL port number ¹ Default: 389
Directory Manager DN	Default: <code>cn=Directory Manager</code>
Directory Manager Password	
Access Manager Administrator User ID	Default: <code>amadmin</code>
Default: <code>amadmin</code>	

Table 9-7 Access Manager Configuration Parameters: amupgrade

Parameter	Value
Access Manager Administrator Password	
Enable Realm Mode	Y/N: Yes means Realm Mode is enabled and services data is migrated to new Realm tree. No (default) means services data remain in Legacy Mode.

- b.** Run the amupgrade script.

```
cd AccessManager-base/upgrade/scripts
export SHLIB_PATH=/opt/sun/private/lib:AccessManager-base/lib
./amupgrade
```

If the upgrade is successful, the script displays “Upgrade completed.”

- c.** Check the following upgrade log file for information about the directory schema extensions:

HP-UX: `opt/sun/install/logs`

Verifying the Access Manager Upgrade

After you finish the upgrade procedure, verify that it was successful as follows:

1. Start Access Manager.

Start the web container in which Access Manager is deployed.

2. Log in to the Access Manager console as amadmin using the following URL:

`http://hostname.domain:port/amconsole.`

where *hostname.domain:port* is the fully qualified host name and port number of the web container you are using.

Verify that new Release 4 services referred to in [“About Java ES Release 4 Access Manager” on page 116](#) are available under the “Service Configuration” tab.

3. Review the status of the upgrade by checking the following upgrade log files in the `/var/opt/sun/install/logs` directory:

Sun Java Enterprise System installer:

- o Java_Shared_Component_Install.*timestamp*
- o Java_Enterprise_System_install.*Atimestamp*
- o Java_Enterprise_System_install.*Btimestamp*
- o Java_Enterprise_System_Summary_Report_install.*timestamp* amupgrade script:
- o Sun_Java_System_Identity_Server_upgrade_dit_log.*timestamp*

4. Review Access Manager trouble shooting files for errors.

The files are located at `/var/opt/sun/identity/debug`

Post-Upgrade Tasks

If you are using the Security Assertion Markup Language (SAML) service, you must add and enable a SAML authentication module using the Access Manager console. For information on creating a SAML authentication module instance, refer to the Sun Java System Access Manager Administration Guide (<http://docs.sun.com/doc/817-7647>).

Rolling Back the Upgrade

No scripts are provided for rolling back Access Manager to its pre-upgrade state. The process must be performed manually using Access Manager data that was backed up as part of the pre-upgrade tasks (see “[Back Up Release 3 Access Manager Log and Debug Files](#)” on page 123). Rollback is too difficult to be practical.

Multiple Instance Upgrades: Release 3 and Release 4 Co-existence

In some deployment architectures Access Manager is deployed on multiple computer systems to provide for high availability and scalability. The Access Manager instances access the same Directory Server. It is often desirable to upgrade the Access Manager instances sequentially without interrupting service. This section discusses the procedure for performing such rolling upgrades.

NOTE Upgrading multiple instances of Access Manager installed on the same host system is not supported in the current release. If you have multiple instances on the same host, after you upgrade the main instance, you must then recreate the additional instances.

The procedure for upgrading Access Manager from Release 3 includes a step for migrating directory schema to support Release 4. Release 3 Access Manager does not support Release 4 directory schema, however Release 4 Access Manager does support Release 3 directory schema.

Java ES Release 4 Access Manager and Release 3 Access Manager instances can coexist and run concurrently against the same Directory Server only if the directory schema has not yet been migrated to Release 4. Therefore, in rolling upgrades, the directory schema should not be migrated to Release 4 until all Access Manager instances have been first upgraded to Release 4.

In performing rolling upgrades, upgrade each instance of Access Manager as described in [“Upgrading Access Manager” on page 124](#), except for [Step 6 on page 124](#). When all instances have been upgraded, then that step can be performed.

Access Manager SDK-only Upgrades

In some deployment architectures, the Access Manager SDK component is installed on one or more computer systems without installing other Access Manager components on those computers. Access Manager SDK serves as a remote interface to Access Manager and must be re-configured for the same operational mode as Access Manager: Legacy or Realm. As a remote interface to Access Manager, the SDK does not need to be configured to access Directory Server.

If Access Manager SDK is being used to support a web component, such as Portal Server or Communications Express, which depends upon web container services, Access Manager SDK must be configured for the corresponding web container. However, Access Manager SDK can also support non-web components, and no web container is needed.

The procedure for upgrading Access Manager SDK is a subset of the procedure for the full Access Manager upgrade, based on the above characteristics.

This section describes how to perform an Access Manager SDK-only upgrade from Java ES Release 3 to Java ES Release 4:

- [Pre-Upgrade Tasks](#)

- [Upgrading Access Manager](#)
- [Verifying the Access Manager Upgrade](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

The pre-upgrade tasks for Access Manager SDK are the same as for the full Access Manager upgrade, but exclude those related to Directory Server and administration tool customizations. The pre-upgrade tasks needed for Access Manager SDK are the following:

- [“Upgrade Access Manager Dependencies” on page 122](#)

However, for Access Manager SDK, there is no dependency on Directory Server, and a dependency on web container software only in the case where Access Manager SDK runs in a web container.

- [“Back Up Release 3 Access Manager Log and Debug Files” on page 123](#)

However, for Access Manager SDK, you only need to back up web container configuration files in the case where Access Manager SDK runs in a web container.

- [“Back Up Release 3 Access Manager Log and Debug Files” on page 123](#)

You also need to obtain the admin username and password for accessing these files.

Upgrading Access Manager SDK

The upgrade procedures for Access Manager SDK are the same as for the full Access Manager upgrade, but exclude those related to administration tool customizations and migrating directory schema.

1. Remove the Java ES Release 3 version of Access Manager SDK.

Follow the instructions in “Remove the Java ES Release 3 Version of Access Manager.” on page 209, except remove only Access Manager SDK.

2. Install Java ES Release 4 version of Access Manager SDK.

Follow the instructions in “Install the Java ES Release 4 Version of Access Manager.” on page 210, except install only Access Manager SDK.

3. Re-configure Access Manager SDK.

Follow the instructions in “Undeploy Access Manager, re-configure, and re-deploy into a Web Container.” on page 212, except set the `DIRECTORY_MODE=5` and the `DEPLOY_LEVEL` parameter as follows:

- If Access Manager SDK is configured for a web container: `DEPLOY_LEVEL=4` (upgrade the SDK and configure the web container).
- If Access Manager SDK is not configured for a web container: `DEPLOY_LEVEL=3` (upgrade the SDK only)

Verifying the Access Manager SDK Upgrade

There are three ways you can verify a successful Access Manager SDK upgrade:

- Run Portal Server, Communications Express, or other component that uses Access Manager SDK to interface with Access Manager, and check that the authentication works.
- Run the Access Manager SDK examples provided in the following location:

`/AccessManager-base/samples/sdk`

- Check the value of the `com.ipplanet.am.version` property, which is in the `AMConfig.properties` file:

`/etc/AccessManager-base/config/AMConfig.properties`

Upgrade Rollback

No scripts are provided for rolling back Access Manager to its pre-upgrade state. The process must be performed manually using Access Manager data that was backed up as part of the pre-upgrade tasks (see “[Back Up Release 3 Access Manager Log and Debug Files](#)” on page 123). Rollback is too difficult to be practical.

Directory Preparation Tool

This chapter describes how to upgrade Directory Preparation Tool to Java ES Release 4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Directory Preparation Tool Upgrades](#)
- [Upgrading Directory Preparation Tool from Java ES Release 3](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *comm_dssetup_base*. At least part of this path might have been specified as an installation directory when Directory Preparation Tool was initially installed. If not, the Java ES installer assigned a default value.

The default value of *comm_dssetup_base* is:
`/opt/sun/comms/dssetup.`

Overview of Directory Preparation Tool Upgrades

This section describes the following general aspects of Directory Preparation Tool that impact upgrading to Java ES Release 4:

- [About Java ES Release 4 Directory Preparation Tool](#)
- [Directory Preparation Tool Upgrade Roadmap](#)
- [Directory Preparation Tool Data](#)
- [Compatibility Issues](#)
- [Directory Preparation Tool Dependencies](#)

About Java ES Release 4 Directory Preparation Tool

Java ES Release 4 versions of Directory Preparation Tool represents a number of minor fixes needed to prepare Directory Server for use by Release 4 communications components (Messaging Server, Calendar Server, and Delegated Administrator).

For details, see the appropriate release notes.

Directory Preparation Tool Upgrade Roadmap

[Table 10-1](#) shows the supported Directory Preparation Tool upgrade paths to Java ES Release 4. The table applies HP-UX for operating systems.

Table 10-1 Upgrade Paths to Java ES Release 4 Directory Preparation Tool 2005Q4

Java ES Release	Directory Preparation Tool Version	General Approach	Re-configuration Required
Release 3	Sun Java System Directory Preparation Tool 6.3 2005Q1	Direct Upgrade: Apply patches.	Prepare Directory for Release 4 communications components

Directory Preparation Tool Data

The following table shows the type of data that could be impacted by an upgrade of Directory Preparation Tool software.

Table 10-2 Directory Preparation Tool Data Usage

Type of Data	Location	Usage
Directory Server schema	Directory Server	Prepare Directory Server for Release 4 communications components: modifies schema, creates new entries and creates indexes

Compatibility Issues

Release 4 Directory Preparation Tool does not introduce any interface changes and is backwardly compatible with earlier versions.

Directory Preparation Tool Dependencies

Directory Preparation Tool has no dependencies on other Java ES components other than Directory Server. Directory Preparation Tool is used to configure Directory Server for use with Java ES communications components.

Upgrading Directory Preparation Tool from Java ES Release 3

This section includes information about upgrading Directory Preparation Tool from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Directory Preparation Tool Upgrade](#)

Introduction

When upgrading Java ES Release 3 Directory Preparation Tool to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Directory Preparation Tool is then used to modify Directory Server as required to support Release 4 Messaging Server, Calendar Server, Communications Express, and Delegated Administrator components.
- **Upgrade Dependencies.** Directory Preparation Tool has no dependencies on Java ES shared components and is compatible with Release 3 Directory Server. Upgrade of Directory Server is therefore optional with respect to upgrade of Directory Preparation Tool to Release 4.
- **Backward Compatibility.** Release 4 Directory Preparation Tool is backwardly compatible with the Release 3 version.

- **Upgrade Rollback.** Rollback of the Release 4 upgrade to Release 3 is achieved by removing the patches applied during the upgrade. The Release 3 Directory Preparation Tool can then be run against Directory Server to back out changes made by the Release 4 version.

Directory Preparation Tool Upgrade

This section describes how to perform an upgrade of Directory Preparation Tool from Java ES Release 3 to Java ES Release 4 on HP-UX platform. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrade Procedure](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Directory Preparation Tool you should perform the tasks described below.

Verify Current Version Information

You can verify the version of Directory Preparation Tool last run against Directory Server by checking attribute values of the *cn=CommServers, o=comms-config* entry written by the tool:

```
./ldapsearch -b cn=config -D bind_dn -w password  
cn=CommServers,o=comms-config
```

The entry has two attributes that specify the current version and the revision:

- *dssetup_ver=version*
- *dssetup_rev=revision*

The tool will write a message to console only if the version of Directory Preparation Tool being run is the same or earlier than the version that was previously run. See [“Upgrade Procedure” on page 140](#), Step 4 on for how to run the tool.

Upgrade Directory Preparation Tool Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, the upgrade of Directory Preparation Tool to Release 4 does not depend upon any other Java ES component.

Back Up Directory Data

The Directory Preparation Tool upgrade from Release 3 to Release 4 does not in itself modify Directory Server data. However, as a safety measure, it is a good idea to back up Directory Server before upgrading the Directory Preparation Tool and running it against Directory Server.

Obtain Required Configuration Information and Passwords

Directory Preparation Tool upgrade requires you to know the superuser password. Otherwise, the tool remembers parameter values used in the previous run and supplies them as defaults when run the next time.

Upgrading Directory Preparation Tool

This section discusses considerations that impact the upgrade procedure for Directory Preparation Tool followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Directory Preparation Tool software to Java ES Release 4 takes into account the following considerations:

- Release 3 Directory Preparation Tool was installed with Directory Server and resides on any computer hosting Directory Server.
- The upgrade of Directory Preparation Tool must be performed on the computer hosting every Directory Server instance being used by Messaging Server, Calendar Server, Communications Express, or Delegated Administrator components.
- The Release 4 upgrade patches for HP-UX are shown in the following table:

Table 10-3 Patches* to Upgrade Directory Preparation Tool

Component	Patch ID
Directory Preparation Tool (DSSETUP)	121513-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to the Directory Preparation Tool installed on the computer where Directory Server resides.

1. Obtain the required patches, based on [Table 10-3](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>

2. Log in as superuser (root).
3. Apply the appropriate Directory Preparation Tool patches as in [Table 10-3](#).

- a. If the product is in the default location (`/opt/sun/comms/dssetup`)

```
swinstall -s patch_location <patch_ID>
```

- b. If the product is in non default location

```
swinstall -s patch_location -x patch_match_target = true
```

4. Run the Directory Preparation Tool against Directory Server. 4. Confirm that the upgrade was successful:

- a. Confirm that Directory Server is running.

- b. Change directory to the location of the Directory Preparation Tool

```
# cd comm_dssetup_base/sbin
```

- c. Run the Directory Preparation Tool (`comm_dssetup.pl` perl script).

```
# perl comm_dssetup.pl
```

The perl version is 5.8.4 and path of the perl is `usr/bin/perl`. Provide the parameters requested by the script.

Verifying the Upgrade

You can verify successful upgrade of Directory Preparation Tool and extension of directory schema by checking the log file created when running the script. The log file is located at:

```
/var/tmp/dssetup_yyyymmddhhmmss
```

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [Upgrade Procedure](#).

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Directory Preparation Tool followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Directory Preparation Tool reverses the procedure for upgrading to Release 4. However, among the changes made by Directory Preparation Tool are modifications to Directory Server schema. These changes are not backed out by the rollback procedure described below, however the schema changes are backwardly compatible.

Rollback Procedure

1. Log in as superuser.
2. Remove the patches in [Table 10-3](#).

```
# swremove <patch_ID>
```

3. Run the rolled-back Directory Preparation Tool against Directory Server.

Directory Server modifications, including indexes are restored to their previous states, however schema changes remain in place. There is no negative impact to the schema extensions; they are backwardly compatible.

Messaging Server

This chapter describes how to upgrade Messaging Server to Java ES Release 4 Messaging Server 6 2005Q4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Messaging Server Upgrades](#)
- [Upgrading Messaging Server from Java ES Release 3](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *msg_svr_base*. At least part of this path might have been specified as an installation directory when Messaging Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *msg_svr_base* is `/opt/sun/messaging`.

Overview of Messaging Server Upgrades

This section describes the following general aspects of Messaging Server that impact upgrading to Java ES Release 4:

- [About Java ES Release 4 Messaging Server](#)
- [Messaging Server Upgrade Roadmap](#)
- [Messaging Server Data](#)
- [Compatibility Issues](#)
- [Messaging Server Dependencies](#)

About Java ES Release 4 Messaging Server

Java ES Release 4 Messaging Server mostly represents bug fixes. There is no major new functionality with respect to Release 3.

Messaging Server Upgrade Roadmap

Table 11-1 Upgrade Paths to Java ES Release 4 Messaging Server 6 2005Q4

Java ES Release	Messaging Server Version	General Approach	Re-configuration Required
Release 3	Sun Java System Messaging Server 6 2005Q1	Direct Upgrade: Performed by applying patches.	Configuration files and configuration directory data

Messaging Server Data

The following table shows the type of data that could be impacted by an upgrade of Messaging Server software.

Table 11-2 Messaging Server Data Usage

Type of Data	Location	Usage
Configuration data	Local configuration directory: <i>/var/msg_svr_base/config/msg.conf</i> and many other configuration files for configuring Messaging Server Store, MTA, MMP, MEM (webmail)	Configuration of Messaging Server components
Configuration data	Directory Server configuration directory	Configuration of Messaging Server components
User data	Directory Server user/group directory	Storing user attributes needed to support messaging for end users
Dynamic application data	Messaging Server store: <i>msg_svr_base/</i>	Store email messages, message transfer queues, and related information on behalf of users

Table 11-2 Messaging Server Data Usage

Type of Data	Location	Usage
Directory schema	Directory Server /opt/sun/mps/serverroot	For user attributes needed to support end users

Compatibility Issues

Release 4 Messaging Server does not introduce any interface changes. The Messaging Server Store, MTA, MMP, and MEM components, logically distinct configurations of Messaging Server, are backwardly compatible with earlier versions.

Messaging Server Dependencies

Messaging Server dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Messaging Server software. Changes in Messaging Server interfaces or functions, for example, could require upgraded version of components upon which Messaging Server depends. The need to upgrade such components depends upon the specific upgrade path.

Messaging Server has dependencies on the following Java ES components:

- **Shared components.** Messaging Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Directory Server.** Messaging Server stores configuration data and user data needed for messaging in Directory Server. As a result, Messaging Server upgrades might require extensions of directory schema.
- **Directory Preparation Tool.** Messaging Server uses the Directory Preparation Tool to prepare Directory Server to support Messaging Server functions.
- **Access Manager (optional).** For software solutions that support single user sign-on for web-based services, Messaging Server can be configured to use Access Manager single sign-on capability.
- **Delegated Admin (optional).** Delegated Admin is the preferred utility to use for provisioning users in Directory Server so that Messaging Server has access to the user data needed to provide messaging services.

Upgrading Messaging Server from Java ES Release 3

This section includes information about upgrading Messaging Server from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Messaging Server Upgrade](#)

Introduction

When upgrading Java ES Release 3 Messaging Server to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Re-configuration is achieved by running two data configuration utilities and by importing configuration data into Directory Server.
- **Upgrade Dependencies.** While Messaging Server has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Messaging Server requires that NSS, NSPR, LDAP C SDK, ICU, and SASL be upgraded to Release 4. Upgrade of J2SE is optional with respect to upgrade of Messaging Server to Release 4.

In addition, Release 4 Messaging Server is dependent upon Directory Server and optionally dependent on Access Manager, as described in [“Messaging Server Dependencies” on page 145](#). However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Messaging Server to Release 4. However, Release 4 Messaging Server has a hard upgrade dependency on Directory Preparation Tool; Release 4 Directory Preparation Tool is required to prepare Directory Server for messaging operations.

However, Release 4 Messaging Server has a hard upgrade dependency on Directory Preparation Tool; Release 4 Directory Preparation Tool is required to prepare Directory Server for messaging operations.

- **Backward Compatibility.** Release 4 Messaging Server is backwardly compatible with the Release 3 version.

- **Upgrade Rollback.** Rollback of the Release 4 upgrade of Messaging Server to Release 3 is achieved by first removing the changes made to Directory Server, removing changes to local configuration files, and removing the patches applied during the upgrade.

Messaging Server Upgrade

This section describes how to perform an upgrade of Messaging Server from Java ES Release 3 to Java ES Release 4 HP-UX. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Messaging Server](#)
- [Upgrade Procedure](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rollback Considerations](#)

Pre-Upgrade Tasks

Before you upgrade Messaging Server you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Messaging Server by entering the following command:

```
# msg_svr_base/sbin/imsimta version
```

Table 11-3 Messaging Server Version Verification Outputs

Java ES Release	Messaging Server Version Number
Release 3	6.2
Release 4	6.2p3

Upgrade Messaging Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Messaging Server has hard upgrade dependencies only on the SASL shared component and on Directory Preparation Tool. Upgrading of other Java ES Release 3 components upon which Messaging Server depends is therefore optional.

However, if you choose to upgrade all Messaging Server dependencies, they should be upgraded in the following order, all before you upgrade Messaging Server. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading NSS, NSPR, LDAP C SDK, ICU, and SASL shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server”](#) on page 61.
3. **Directory Preparation Tool.** Directory Preparation Tool rarely resides on the same computer as Messaging Server, however, instructions for upgrading Directory Preparation Tool and running it against Directory Server are provided in [Chapter 10, “Directory Preparation Tool”](#) on page 135.

Back Up Messaging Server Data

The Messaging Server upgrade from Release 3 to Release 4 requires re-configuration of Messaging Server in local configuration files and in the Directory Server configuration directory. The local changes can be rolled back, but it is a good idea to back up the configuration directory in case you want to roll back the Release 4 upgrade at a future point.

Obtain Required Configuration Information and Passwords

Messaging Server upgrade requires knowing the following information:

- Superuser password
- Directory Manager DN and password

Upgrading Messaging Server

This section discusses considerations that impact the upgrade procedure for Messaging Server followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Messaging Server software to Java ES Release 4 takes into account the following considerations:

- All Messaging Server components, such as Messaging Server Store, MTA, MMP, or MEM, that correspond to the same installed Messaging Server image, are upgraded at the same time. All such components should be shut down before patches are applied to the installed image.
- The Release 4 upgrade patches for HP-UX are shown in the following table:

Table 11-4 Patches* to Upgrade Messaging Server

Component	Patch ID
Messaging Server core	121511-01
Messaging Server locale	121927-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to all Messaging Server components that correspond to the same installed Messaging Server image on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 11-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>

2. Log in as superuser.
3. Stop all running Messaging Server components.


```
# msg_svr_base/sbin/stop-msg
```
4. If not already done, upgrade the required shared components to Release 4.
5. Apply the appropriate Messaging Server patches in [Table 11-4](#).
 - a. If the product is installed in the default location (`/opt/sun/messaging`)

```
swinstall -s patch_location <patch_ID>
```

- b. If the product is installed in other location

```
swinstall -s patch_location -x patch_match_target = true
```

- 6. Migrate configuration data from existing configuration files to Release 4 candidate configuration files.

- a. Create candidate configuration files.

```
# cd msg_svr_base/sbin
```

```
# ./patch-config-hp msg_svr_base/install/patch/<patch_ID>
```

This command backs up existing configuration files. Then it merges configuration parameter values in these files with Release 4 template configuration files to create new Release 4 candidate configuration files. You should examine these new files for possible conflicts, as described in the Special Installation Instructions section of the patch 281005-01 readme file.

This command also generates the following ldif files (LDAP directory import files):

```
msg_svr_base/lib/patch/cfgdir_diff.ldif
```

```
msg_svr_base/lib/patch/ugdir_diff.ldif
```

- b. Install the Release 4 candidate configuration files, making them the active configuration.

```
# ./install-newconfig msg_svr_base/install/patch/<patch_ID>
```

This command installs the new Release 4 configuration files in their correct Release 4 locations.

- c. Import the new configuration data generated in Step a on page 231 into the Directory Server configuration directory being used by Messaging Server. Change to the configuration directory and import the ldif files using the ldapmodify command:

```
# cd/msg_svr_base/lib
```

```
# ./ldapmodify -D bind_dn -w password -c -e patch/cfgdir_diff.rej  
-f patch/cfgdir_diff.ldif
```

```
# ./ldapmodify -D bind_dn -w password -c
```

```
-e patch/ugdir_diff.rej -f patch/ugdir_diff.ldif
```

- Restart the Messaging Server components that were stopped in [Step 3](#).

```
# msg_svr_base/sbin/start-msg
```

Verifying the Upgrade

You can verify the current version of Messaging Server by entering the following command:

```
# msg_svr_base/sbin/imsimta version
```

You can also check the banner displayed when starting up Messaging Server components

See [Table 11-3 on page 147](#) for output values.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in “[Upgrade Procedure](#)” on page 149.

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Messaging Server followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Web Server is pretty much the reverse of the procedure for upgrading to Release 4. The patches are removed.

- Log in as superuser (root).
- Stop all running Messaging Server components.


```
# msg_svr_base/sbin/stop-msg
```
- Roll back the changes made to the Directory Server configuration directory being used by Messaging Server.

Replace the directory with the pre-upgrade directory that you backed up before beginning the upgrade procedure (see “[Back Up Messaging Server Data](#)” on page 148).

- Roll back the re-configuration performed in [Step 6](#).

```
# cd msg_svr_base/sbin
```

```
# ./uninstall-newconfig msg_svr_base/install/patch/<patch_ID>
```

5. Remove the patches in [Table 11-4 on page 149](#).

```
# swremove <patch_ID>
```

6. Restart the Messaging Server components that were stopped in [Step 2](#).

```
# msg_svr_base/sbin/start-msg
```


Calendar Server

This chapter describes how to upgrade Calendar Server to Java ES Release 4 Calendar Server 6 2005Q4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [Overview of Calendar Server Upgrades](#)
- [Upgrading Calendar Server from Java ES Release 3](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *cal_svr_base*. At least part of this path might have been specified as an installation directory when Calendar Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *cal_svr_base* is `/opt/sun/calendar`.

Overview of Calendar Server Upgrades

This section describes the following general aspects of Calendar Server that impact upgrading to Java ES Release 4:

- [About Java ES Release 4 Calendar Server](#)
- [Calendar Server Upgrade Roadmap](#)
- [Calendar Server Data](#)
- [Compatibility Issues](#)
- [Calendar Server Upgrade](#)

About Java ES Release 4 Calendar Server

Java ES Release 4 Calendar Server mostly represents bug fixes. There is no major new functionality with respect to Release 3.

Calendar Server Upgrade Roadmap

Table 12-1 shows the supported Calendar Server upgrade paths to Java ES Release 4.

Table 12-1 Upgrade Paths to Java ES Release 4 Calendar Server 6 2005Q4

Java ES Release	Calendar Server	General Approach	Re-configuration Required
Release 3	Sun Java System Calendar Server 6 Q1	Direct Upgrade: Apply patches.	None

Calendar Server Data

The following table shows the type of data that could be impacted by an upgrade

Table 12-2 Calendar Server Data Usage

Type of Data	Location	Usage
Configuration data	/opt/sun/calendar/config/i cs.conf	Configuration of Calendar Server
Dynamic application data	Calendar Server database:/var/opt/sun/calen dar/csdb	Store calendar entries on behalf of users.
Directory schema	Directory Server user/group directory	For user attributes needed to support end users

Compatibility Issues

Release 4 Calendar Server does not introduce any interface changes. Calendar Server is backwardly compatible with earlier versions.

Calendar Server Dependencies

Calendar Server dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Calendar Server software. Changes in Calendar Server interfaces or functions, for example, could require upgraded version of components upon which Calendar Server depends.

Calendar Server has dependencies on the following Java ES components:

- **Shared components.** Calendar Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)). Calendar Server upgrades might depend upon the upgraded versions of these shared components.
- **Directory Server.** Calendar Server accesses user data stored in Directory Server. As a result, Calendar Server upgrades might require extensions of directory schema.
- **Directory Preparation Tool.** Calendar Server uses the Directory Preparation Tool to prepare the directory to support Calendar Server functions.
- **Access Manager (optional).** For software solutions that support single user sign-on for web-based services, Calendar Server can be configured to use Access Manager single sign-on capability.
- **Messaging Server (optional).** Calendar Server can be configured to use Messaging Server to provide messaging notifications of calendar events.
- **Delegated Admin (optional).** Delegated Admin is the preferred utility to use for provisioning users in Directory Server so that Calendar Server has access to the user data needed to provide calendar services.

Upgrading Calendar Server from Java ES Release 3

This section includes information about upgrading Calendar Server from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Calendar Server Upgrade](#)

Introduction

When upgrading Java ES Release 3 Calendar Server to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version.
- **Upgrade Dependencies.** While Calendar Server has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Calendar Server requires only that SASL be upgraded to Release 4. Upgrade of other shared components is optional with respect to upgrade of Communications Express.

In addition, Release 4 Calendar Server is dependent upon Directory Server and optionally dependent on Access Manager, as described in “[Calendar Server Dependencies](#)” on [page 155](#). However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Calendar Server to Release 4.

However, Release 4 Calendar Server has a hard upgrade dependency on Directory Preparation Tool; Release 4 Directory Preparation Tool is required to prepare Directory Server for calendaring operations.

- **Backward Compatibility.** Release 4 Calendar Server is backwardly compatible with the Release 3 version.
- **Upgrade Rollback.** Rollback of the Release 4 upgrade of Calendar Server to Release 3 is achieved by removing the patches applied during the upgrade.

Calendar Server Upgrade

This section describes how to perform an upgrade of Calendar Server from Java ES Release 3 to Java ES Release 4 on HP-UX platform.

- [Pre-Upgrade Tasks](#)
- [Upgrading Calendar Server](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Calendar Server you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Calendar Server using the following command

```
# /opt/sun/calendar/bin/cshttpd -#
```

Table 12-3 Calendar Server Version Verification Outputs

Java ES Release	Calendar Server Version Number
Release 3	2005Q1
Release 4	2005Q4

Upgrade Calendar Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Calendar Server has a hard upgrade dependency only on Directory Preparation Tool. Upgrading of other Java ES Release 3 components upon which Calendar Server depends is therefore optional.

However, if you choose to upgrade all Calendar Server dependencies, they should be upgraded in the following order, all before you upgrade Calendar Server. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#).
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server”](#).
3. **Directory Preparation Tool.** Directory Preparation Tool rarely resides on the same computer as Calendar Server, however, instructions for upgrading Directory Preparation Tool and running it against Directory Server are provided in [Chapter 10, “Directory Preparation Tool”](#).

Back Up Calendar Server Data

The Calendar Server upgrade from Release 3 to Release 4 requires no re-configuration of Calendar Server. However, as a safety precaution, you might back up your Calendar Server store, located at following path:

```
/var/opt/sun/calendar/csdb
```

Obtain Required Configuration Information and Passwords

No special information about your currently installed version is needed. However you will have to log in as superuser to perform the upgrade.

Upgrading Calendar Server

This section discusses considerations that impact the upgrade procedure for Calendar Server followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Calendar Server software to Java ES Release 4 takes into account the following considerations:

- Calendar Server should be shut down when patches are being applied to the installed image.
- The Release 4 upgrade patches for HP-UX are shown in the following table:

Table 12-4 Patches* Upgrade Calendar Server

Component	Patch ID
Calendar Serve Core	121512-02
Calendar Server locale	121928-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the above table.

The procedure documented below applies to Calendar Server on the computer where the upgrade is taking place.

Upgrade Procedure

1. Obtain the required patches, based on [Table 12-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>.

2. Log in as superuser.
3. Stop Calendar Server if it is running.

```
cal_svr_base/sbin/stop-cal
```

4. If not already done, upgrade the SASL shared component to Release 4 and any other shared components you wish to upgrade.
 - a. Apply the appropriate Calendar Server patches in [Table 12-4](#).

```
# swinstall -s <patch_ID>
```

5. Confirm that the patch upgrade was successful:

```
# swlist -v1 patch | grep <patch_ID>
```

The output should return the versions of patch IDs applied in [Step a](#).

6. Restart the Calendar Server that was stopped in [Step a](#).

```
cal_svr_base/sbin/start-cal
```

Verifying the Upgrade

The upgrade of Calendar Server to Release 4 is verified by confirming that the upgrade patches have been properly applied. The steps for this verification were included in “[Upgrade Procedure](#)” on page 158.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in “[Upgrade Procedure](#)” on page 158.

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Calendar Server followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Calendar Server is pretty much the reverse of the procedure for upgrading to Release 4.

Rollback Procedure

1. Obtain the required patches based on the [Table 12-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>.

2. Log in as superuser.
3. Stop Calendar Server.
`cal_svr_base/sbin/stop-cal`
4. Remove the patches in [Table 12-4](#).
`# swremove <patch_ID>`
5. Restart Calendar Server.
`cal_svr_base/sbin/start-cal`

Communications Express

This chapter describes how to upgrade Communications Express to Java ES 2005Q4 (Release 4): Sun Java System Communications Express 6 2005Q4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [“Overview of Communications Express Upgrades” on page 161](#)
- [“Upgrading Communications Express from Java ES Release 3” on page 164](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *CommsExpress-base*. At least part of this path might have been specified as an installation directory when Web Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *CommsExpress-base* is `/opt/sun/uwc`.

Overview of Communications Express Upgrades

This section describes the following general aspects of Communications Express that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Communications Express](#)
- [Communications Express Upgrade Roadmap](#)
- [Communications Express Data](#)
- [Compatibility Issues](#)
- [Communications Express Dependencies](#)

About Java ES Release 4 Communications Express

Java ES Release 4 Communications Express mostly represents bug fixes. There are a few new features with respect to Release 3: mail filter support, address book sharing, and proxy authentication.

Communications Express Upgrade Roadmap

[Table 13-1](#) shows the supported Communications Express upgrade paths to Java ES Release 4.

Table 13-1 Upgrade Paths to Java ES Release 4:
Sun Java System Communications Express 6 2005Q4

Java ES Release	Communications Express Version	General Approach	Re-configuration Required
Release 3	Sun Java System Communications Express 6 2005Q1	Direct upgrade: Perform by applying patches and re-configuring Messaging Server component.	Configuration files

Communications Express Data

The following table shows the type of data that could be impacted by an upgrade of Communications Express software.

Table 13-2 Communications Express Data Usage

Type of Data	Location	Usage
Configuration data:	Local configuration directory var/ <i>CommsExpress-base</i> /WEB-INF/config/uwcauth.properties var/ <i>CommsExpress-base</i> /WEB-INF/config/uwconfig.properties var/ <i>CommsExpress-base</i> /WEB-INF/config/uwclogging.properties <i>MessagingServer-base</i> /config/msg.conf and other configuration files for configuring Messaging Server MEM (webmail)	Configuration of Communications Express, including Messaging Server MEM (webmail)
Directory schema	Directory Server user/group directory	For user attributes needed to support end users

Compatibility Issues

Release 4 Communications Express does not introduce any interface changes and is backwardly compatible with earlier versions.

Communications Express Dependencies

Communications Express dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Communications Express software. Changes in Communications Express interfaces or functions, for example, could require upgraded version of components upon which Communications Express depends. The need to upgrade such components depends upon the specific upgrade path.

Communications Express has dependencies on the following Java ES components:

- **Shared components.** Communications Express has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).

- **Web Container.** Communications Express depends upon web container services, which can be provided either by Java ES Web Server or Java ES Application Server.
- **Access Manager (or Access Manager SDK).** Communications Express depends upon Access Manager to provide authentication and authorization services for end users, including single sign-on.
- **Messaging Server.** Communications Express is used to provide web-based access to Messaging Server. In fact Communications Express directly employs the Messaging Server MEM component to access other Messaging Server back-end components, such as the Messaging Server Store and MTA components.
- **Calendar Server.** Communications Express is used to provide web-based access to Calendar Server.
- **Directory Server.** Communications Express stores configuration data and also accesses user data stored in Directory Server. As a result, Communications Express upgrades might require upgrades of Directory Server or extensions of directory schema.
- **Directory Preparation Tool.** Communications Express uses the Directory Preparation Tool to prepare Directory Server to support Communications Express functions. As a result, Communications Express upgrades might depend upon preparation of the directory to support new functions.

Upgrading Communications Express from Java ES Release 3

This section includes information about upgrading Communications Express from Java ES 2005Q1 (Release 3) to Java ES 2005Q4 (Release 4). The section covers the following topics:

- [Introduction](#)
- [Communications Express Upgrade](#)
- [Multiple Instance Upgrades](#)

Introduction

When upgrading Java ES Release 3 Communications Express to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Re-configuration of the included Messaging Server MEM component is achieved using two configuration utilities and by importing configuration data into Directory Server.
- **Upgrade Dependencies.** While Calendar Server has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Calendar Server is compatible with the Release 3 versions of these components. Upgrade of these shared components is therefore optional with respect to upgrade of Calendar Server to Release 4.

In addition, Release 4 Communications Express is dependent upon a web container and on Access Manager, as described in “[Communications Express Dependencies](#)” on page 163. However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Communications Express to Release 4.

However, Communications Express has hard upgrade dependencies on both Calendar Server, for which it provides web-based access, and on Messaging Server for which it also provides web-based access using the Messaging Server MEM component. Both Calendar Server and Messaging Server must therefore be upgraded to Release 4 before Communications Express can be upgraded to Release 4.

In addition, Release 4 Communications Express has a hard upgrade dependency on Directory Preparation Tool; Release 4 Directory Preparation Tool is required to prepare Directory Server for Communications Express functions.

- **Backward Compatibility.** Release 4 Communications Express is backwardly compatible with the Release 3 version.
- **Upgrade Rollback.** Rollback of the Release 4 upgrade of Communications Express to Release 3 is achieved by first removing the changes made to Directory Server, removing changes to local configuration files, and removing the patches applied during the upgrade.

Communications Express Upgrade

This section describes how to perform an upgrade of Communications Express from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Communications Express](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Communications Express you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Communications Express by accessing the Communications Express login page, which shows the current version number.

`http://hostName:port/uwc/auth`

Table 13-3 Communications Express Version Verification Outputs

Java ES Release	Communication Express Version Number
Release 3	Sun Java System Communications Express 6 2004Q1
Release 4	Sun Java System Communications Express 6 2004Q4

Upgrade Communications Express Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Communications Express has hard upgrade dependencies only on Messaging Server, Calendar Server, and Directory Preparation Tool. Upgrading of other Java ES Release 3 components upon which Communications Express depends is therefore optional.

However, if you choose to upgrade all Communications Express dependencies, they should be upgraded in the following order, all before you upgrade Communications Express. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 48.
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server”](#) on page 61.
3. **Web Container Software.** Instructions for upgrading Web Server or Application Server are provided in [Chapter 5, “Web Server”](#) on page 81 and [Chapter 8, “Application Server”](#) on page 107, respectively.
4. **Access Manager (Access Manager SDK).** Instructions for upgrading Access Manager to Release 4 are provided in [Chapter 9, “Access Manager”](#) on page 115.
5. **Directory Preparation Tool.** Release 4 Directory Preparation Tool needs to have been run against Directory Server before configuring Release 4 Communications Express. If Release 4 Directory Preparation Tool has not already been run against Directory Server, upgrade Directory Preparation Tool to Release 4 and use it to modify and extend the schema of Directory Server in (see [Chapter 10, “Directory Preparation Tool”](#) on page 135 for procedures).
6. **Messaging Server.** Messaging Server components need to be upgraded to Release 4 to support Release 4 Communications Express. Instructions for upgrading Messaging Server to Release 4 are provided in [Chapter 11, “Messaging Server”](#) on page 143.
7. **Calendar Server.** Calendar Server components need to be upgraded to Release 4 to support Release 4 Communications Express. Instructions for upgrading Calendar Server to Release 4 are provided in [Chapter 12, “Calendar Server”](#) on page 153.

Back Up Communications Express Data

The Communications Express upgrade from Release 3 to Release 4 requires re-configuration of Messaging Server MEM. The local changes can be rolled back, so there is no need to back up any data.

Obtain Required Configuration Information and Passwords

You have to log in as superuser to perform the upgrade. If you are using Web Server as a web container, no configuration information is needed. However if you are using Application Server as a web container, you will need the Application Server administrator user ID and password.

Upgrading Communications Express

This section discusses considerations that impact the upgrade procedure for Communications Express followed by a description of the procedure itself.

- Communications Express includes components used to provide web-based access to Calendar Server and a Messaging Server MEM component used to provide web-based access to Messaging Server back-end components. The upgrade patches encompass all these components.
- All Communications Express components should be deployed to the same web container. The web container should be shut down before patches are applied to the installed image.
- The Release 4 Communications Express upgrade patches for HP-UX are shown in the following table:

Table 13-4 Patches* to Upgrade Communications Express

Component	Patch-ID
Communications Express	121522-01
Localization	121925-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to all Communications Express components on the computer being updated.

1. Obtain the required patches using the patch numbers from. Use this information to obtain the version numbers for the depots.

Patches can be downloaded from:

[http://sunsolve.sun.com/.](http://sunsolve.sun.com/)

2. Log in as superuser.

```
su root
```

3. Stop Communications Express by stopping its web container.

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
ApplicationServer-base/bin/asadmin stop-domain domainName
```

4. If not already done, upgrade any shared components you wish to upgrade to Release 4.

5. Apply the depots for Communications Express in [Table 13-4](#).

For example:

```
swinstall -s 'pwd' <patch_ID>
```

6. Confirm that the patch upgrade was successful:

```
swlist |grep <patch_ID>
```

The output should return the version details of patch <patch_ID>.

7. Migrate configuration data from existing configuration files to Release 4 candidate configuration files.

```
cd CommsExpress-base/sbin
```

```
./patch-config CommsExpress-base/install/patch
```

This command prompts you for the current configuration directory and then backs up the existing configuration files. Then it merges configuration parameter values in these files with Release 4 template configuration files to create new Release 4 candidate configuration files. You should check these new files for possible conflicts, as described in the Special Installation Instructions section of the patch `readme` file.

8. Install the Release 4 candidate configuration files, making them the active configuration.

```
./install-newconfig CommsExpress-base/install/patch
```

This command installs the new Release 4 configuration files in their correct Release 4 locations.

9. Remove the JSP class cache for Communications Express that is maintained by the web container.

For the procedure, see the documentation for your web container (Web Server or Application Server).

10. Restart Communications Express by restarting its web container.

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
ApplicationServer-base/bin/asadmin start-domain domainName  
--user admin_ID --password password
```

Verifying the Upgrade

You can verify the current version of Communications Express by accessing the Communications Express login page, which shows the current version number.

```
http://hostName:port/uwc/auth
```

Once logged in, check the upgraded user interface for the new mail tab and old email and calendar events (if you are using those channels).

Also, you can check the log files for the various steps in the upgrade procedure:

```
CommsExpress-base/install/patch/121522-02.
```

```
CommsExpress-base/patch-config_20050729164754.log
```

```
CommsExpress-base/install-newconfig_20050729164838.log
```

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [“Upgrade Procedure” on page 168](#).

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Communications Express followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Communications Express is pretty much the reverse of the procedure for upgrading to Release 4. The re-configurations are rolled back and the patches are removed.

Rollback Procedure

1. Log in as superuser.

```
su root
```

2. Stop Communications Express by stopping its web container.

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
ApplicationServer-base/bin/asadmin stop-domain domainName
```

3. Roll back the changes made to the Directory Server configuration directory being used by Communications Express.

4. Roll back the re-configuration performed in [Step 8 on page 169](#).

```
cd CommsExpress-base/sbin
```

```
./uninstall-newconfig CommsExpress-base/install/patch/
```

5. Remove the patches in [Table 13-4 on page 168](#).

```
swremove <patch_ID>
```

6. Restart Communications Express by restarting its web container.

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
ApplicationServer-base/bin/asadmin start-domain domainName
```

```
--user admin_ID --password password
```

Multiple Instance Upgrades

In some deployment architectures Communications Express is deployed on multiple computer systems to provide for scalability and to improve availability.

For example, you might have Communications Express components running on multiple computers with a load balancer to distribute the load.

In the case of load-balanced instances of Communications Express, you can perform a rolling upgrade in which you upgrade the Communications Express instances sequentially without interrupting service. You upgrade each instance of Communications Express while the others remain running. You perform the upgrade of each instance as described in [“Communications Express Upgrade” on page 166](#).

Instant Messaging

This chapter describes how to upgrade Instant Messaging to Java ES 2005Q4 (Release 4): Sun Java System Instant Messaging 7 2005Q4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4:

- [“Overview of Instant Messaging Upgrades” on page 173](#)
- [“Upgrading Instant Messaging from Java ES Release 3” on page 175](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *InstantMessaging-base*. At least part of this path might have been specified as an installation directory when Instant Messaging was initially installed. If not, the Java ES installer assigned a default value.

The default value of *InstantMessaging-base* is `/opt/sun/im`.

Overview of Instant Messaging Upgrades

This section describes the following general aspects of Instant Messaging that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Instant Messaging](#)
- [Instant Messaging Upgrade Roadmap](#)
- [Instant Messaging Data](#)
- [Compatibility Issues](#)
- [Instant Messaging Dependencies](#)

About Java ES Release 4 Instant Messaging

Java ES Release 4 Instant Messaging mostly represents bug fixes. There is no major new functionality with respect to Release 3.

Instant Messaging Upgrade Roadmap

[Table 14-1](#) shows the supported Instant Messaging upgrade paths to Java ES Release 4.

Table 14-1 Upgrade Paths to Java ES Release 4: Sun Java System Instant Messaging 7 2005Q4

Java ES Release	Component	General Approach
Release 3	Sun Java System Instant Messaging 7 2005Q1	Direct upgrade: Performed by applying patches.

Instant Messaging Data

The following table shows the type of data that could be impacted by an upgrade of Instant Messaging software.

Table 14-2 Instant Message Data Usage

Type of Data	Location	Usage
Configuration data:	Local configuration directory <code>/etc/opt/sun/im/default/config/iim.conf</code> <code>/etc/opt/sun/im/default/config/registration.p</code>	Broker and related services configurations
Instant Messaging Server Resources	Local configuration directory <code>InstantMessaging-base/html</code>	customized client files downloaded by end users to launch the Messenger client.
Dynamic data	<i>runtimeFilesDir</i> /default/db where <i>runtimeFilesDir</i> specified at installation: HP-UX: default <i>runtimeFilesDir</i> : <code>/var/opt/sun/im</code>	Track instant messaging clients that are online

Compatibility Issues

Release 4 Instant Messaging does not introduce any interface changes and is backwardly compatible with earlier versions.

Instant Messaging Dependencies

Instant Messaging dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Instant Messaging software. Changes in Instant Messaging interfaces or functions, for example, could require upgraded version of components upon which Instant Messaging depends. The need to upgrade such components depends upon the specific upgrade path.

Instant Messaging has dependencies on the following Java ES components:

- **Shared components.** Instant Messaging has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Web Container.** Instant Messaging depends upon web container services, which can be provided either by Java ES Web Server or Java ES Application Server.
- **Directory Server (optional).** Instant Messaging can be configured to store and access user data in Directory Server. As a result, Instant Messaging upgrades might require extensions of directory schema.
- **Access Manager (optional).** For software solutions that support single user sign-on for web-based services, Instant Messaging can be configured to use Access Manager single sign-on capability.

Upgrading Instant Messaging from Java ES Release 3

This section includes the following information about upgrading Instant Messaging from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Instant Messaging Upgrade](#)
- [Multiple Instance Upgrades](#)

Introduction

When upgrading Java ES Release 3 Instant Messaging to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. There is no re-configuration of Instant Messaging required.
- **Upgrade Dependencies.** While Instant Messaging has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Instant Messaging requires only that IM-SDK be upgraded to Release 4. Upgrade of other shared components is optional with respect to upgrade of Instant Messaging to Release 4.

In addition, Release 4 Instant Messaging is dependent on a web container and optionally dependent on Access Manager, as described in [“Instant Messaging Dependencies” on page 175](#). However, these are soft upgrade dependencies; upgrade the web container and Access Manager is optional with respect to upgrade of Instant Messaging to Release 4.

- **Backward Compatibility.** Release 4 Instant Messaging is backwardly compatible with the Release 3 version.
- **Upgrade Rollback.** Rollback of the Release 4 upgrade to Release 3 is achieved on by removing the patches applied during the upgrade.

Instant Messaging Upgrade

This section describes how to perform an upgrade of Instant Messaging from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Instant Messaging](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Instant Messaging you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Instant Messaging using standard version checking utilities

```
# Instant_Messagingbase/sbin/imadmin version
```

Table 14-3 Instant Messaging Version Verification Outputs

Java ES Release	Instant Messaging Version Number
Release 3	Version numbers 7.0
Release 4	Version numbers 7.0.1

Upgrade Instant Messaging Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Instant Messaging has a hard upgrade dependency only on the IM-SDK shared component. Upgrading of other Java ES Release 3 components upon which Instant Messaging depends is therefore optional.

If you choose to upgrade all Instant Messaging dependencies, they should be upgraded in the following order, all before you upgrade Instant Messaging. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading IM-SDK and other Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.
2. **Web Container Software.** Instructions for upgrading Web Server or Application Server are provided in [Chapter 5, “Web Server”](#) on page 81 and [Chapter 8, “Application Server”](#) on page 107, respectively.
3. **Access Manager (optional).** Instructions for upgrading Access Manager to Release 4 are provided in [Chapter 9, “Access Manager”](#) on page 115.

Back Up Instant Messaging Data

The Instant Messaging upgrade from Release 3 to Release 4 does not modify configuration data. However, as a safety measure it is a good idea to back up the Instant Messaging database and any existing resource and configuration files you have customized before upgrading Instant Messaging. For more information, see the *Sun Java System Instant Messaging 7 2005Q4 Administration Guide* (<http://docs.sun.com/doc/819-2503>).

Obtain Required Configuration Information and Passwords

You have to log in as superuser to perform the upgrade. If you are using Web Server as a web container, no configuration information is needed. However if you are using Application Server as a web container, you will need the Application Server administrator user ID and password.

Upgrading Instant Messaging

This section discusses considerations that impact the upgrade procedure for Instant Messaging followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Instant Messaging software to Java ES Release 4 takes into account the following considerations:

The upgrade of Instant Messaging software to Java ES Release 4 takes into account the following considerations:

- Instant Messaging components should be shut down when patches are being applied to the installed image.
- In architectures in which different Instant Messaging subcomponents reside on different computers, for example messenger resources on one computer, Instant Messaging server on another, and Instant Messaging Multiplexor on yet another, the upgrade must be performed on all such computers.
- The Instant Messaging upgrade applies to multiple subcomponents of Instant Messaging on one computer using the same installed image.
- The Release 4 Instant Messaging upgrade patch for HP-UX OS are shown in the following table:

Table 14-4 Patches* to Upgrade Instant Messaging

Component	Patch-ID
Instant Messaging	121516-01
Instant Messaging locale	121926-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to Instant Messaging on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 14-4](#).

Patches can be downloaded from:

www.sunsolve.com.

2. Log in as superuser.

```
su root
```

3. Stop Instant Messaging if it is running.

```
InstantMessaging-base/sbin/imadmin stop
```

4. If not already done, upgrade the IM-SDK shared component to Release 4 and any other shared components you wish to upgrade.

5. Apply the appropriate Instant Messaging patches in [Table 14-4](#).

```
swinstall -s <patch-location> <patch-ID>
```

6. Confirm that the upgrade was successful:

```
swlist | grep <patch_ID>
```

The output should return the versions of patch IDs applied in [Step 5](#)

7. Restart the Instant Messaging service that was stopped in [Step 3](#).

```
InstantMessaging-base/sbin/imadmin start
```

Verifying the Upgrade

The upgrade of Instant Messaging to Release 4 is verified by confirming that the upgrade patches have been properly applied. The steps for this verification were included in [“Verifying the Upgrade” on page 179](#).

You can also check the status of the various Instant Messaging subcomponents using the following command:

```
InstantMessaging-base/sbin/imadmin status
```

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [“Verifying the Upgrade” on page 179](#).

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Instant Messaging followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Instant Messaging is pretty much the reverse of the procedure for upgrading to Release 4.

Rollback Procedure

1. Log in as superuser.

```
su root
```

2. Stop Instant Messaging if it is running.

```
InstantMessaging-base/sbin/imadmin stop
```

3. Remove the patches in [Table 14-4 on page 178](#).

```
swremove <patch-ID>
```

4. Restart the Instant Messaging service that was stopped in [Step 2](#)

```
InstantMessaging-base/sbin/imadmin start
```

Multiple Instance Upgrades

Multiple instance upgrades are not applicable to Release 4 Instant Messaging.

Portal Server

This chapter describes how to upgrade Portal Server to Java ES 2005Q4 (Release 4): Sun Java System Portal Server 6 2005Q4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [“Overview of Portal Server Upgrades” on page 181](#)
- [“Upgrading Portal Server from Java ES Release 3” on page 184](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *PortalServer-base*. At least part of this path might have been specified as an installation directory when Portal Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *PortalServer-base* is `/opt/sun/portal_svr`.

Overview of Portal Server Upgrades

This section describes the following general aspects of Portal Server that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Portal Server](#)
- [Portal Server Upgrade Roadmap](#)
- [Portal Server Data](#)
- [Compatibility Issues](#)
- [Portal Server Dependencies](#)

About Java ES Release 4 Portal Server

Java ES Release 4 Portal Server is functionally the same as Release 3, but contains bug fixes made since Release 3.

Portal Server Upgrade Roadmap

[Table 15-1](#) shows the supported Portal Server upgrade paths to Java ES Release 4.

Table 15-1 Upgrade Paths to Java ES Release 4 Portal Server 6 2005Q4

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Portal Server 6 2005Q1	Direct upgrade: Performed by applying patches. Some limitations apply (see procedures).	None

Portal Server Data

The following table shows the type of data that could be impacted by an upgrade of Portal Server software.

Table 15-2 Portal Server Data Usage

Type of Data	Location	Usage
Configuration data	/etc/PortalServer-base/ /var/PortalServer-base Web Server web container: server.policy and server.xml files in WebServer-base/https-hostname/c onfig Application Server web container: server.policy and domain.xml files in /var/ApplicationServer-base/domai n/domain1/config	Configuration of Portal Server and its web container instance.
Customization data	/etc/PortalServer-base/dektop	JAR files for customized modules Customized sample Portal Server desktop
Directory schema	Directory Server	Portal Server depends on
Services configuration		services configurations, such as
User data		the portal desktop, and user profile data that is stored in a directory.
Dynamic application data	None	Portal Server does not persistently store application data such as session state.

Compatibility Issues

Release 4 Portal Server does not introduce any interface changes. Portal Server components, including the mobile access component, are backwardly compatible with earlier versions.

Portal Server Dependencies

Portal Server dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Portal Server software. Changes in Portal Server interfaces or functions, for example, could require upgraded version of components upon which Portal Server depends. The need to upgrade such components depends upon the specific upgrade path.

Portal Server has dependencies on the following Java ES components:

- **Shared components.** Portal Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Web Container.** Portal Server depends upon web container services, which can be provided either by Java ES Web Server or Java ES Application Server.
- **Access Manager (or Access Manager SDK).** Portal Server depends upon Access Manager to provide authentication and authorization services for end users, including single sign-on.
- **Directory Server.** Portal Server accesses user data stored in Directory Server. As a result, Portal Server upgrades might require extensions of directory schema.

Upgrading Portal Server from Java ES Release 3

This section includes information about upgrading Portal Server from Java ES 2005Q1 (Release 3) to Java ES 2005Q4 (Release 4).

NOTE The aforementioned upgrade path is not currently supported. For more information, see the Sun Java System Portal Server 6 2005Q4 Release Notes (<http://docs.sun.com/doc/819-2575>).

The section covers the following topics:

- [Introduction](#)
- [Portal Server Upgrade](#)
- [Multiple Instance Upgrades](#)

Introduction

When upgrading Java ES Release 3 Portal Server to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Re-configuration of Portal Server is required only if the web container upon which Portal Server depends is upgraded or changed.
- **Upgrade Dependencies.** While Portal Server has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Portal Server is compatible with the Release 3 version of these components. Upgrade of these shared components, except for Mobile Access Core (MA Core), is therefore optional with respect to upgrade of Portal Server to Release 4.

In addition, Release 4 Portal Server is dependent upon a web container, Access Manager, and Directory Server as described in “Portal Server Dependencies” on page 306. However, these are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Portal Server to Release 4. Since JES4 portal uses libjss4.sl, sun-jss upgrade may be required.

- **Backward Compatibility.** Release 4 Portal Server is backwardly compatible with the Release 3 version.
- **Upgrade Rollback.** Rollback of the Release 4 upgrade of Portal Server to Release 3 is achieved by rolling back the patches applied during the upgrade. Patch rollback is available on HP-UX platform.

Portal Server Upgrade

This section describes how to perform an upgrade of Portal Server from Java ES Release 3 to Java ES Release 4 on HP-UX platform.

- [Pre-Upgrade Tasks](#)
- [Upgrading Portal Server](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Portal Server you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Portal Server using the following command:

```
PortalServer-base/bin/version
```

Table 15-3 Portal Server Version Verification Outputs*

Java ES Release	Portal Server Version Number
Release 3	6.3.1
Release 4	6.3.1

*. The only difference between Release 3 and Release 4 is a patch. You can check for the Release 4 patches using the `swlist <patch_ID>`.

Upgrade Portal Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Portal Server has a hard upgrade dependency only on the Mobile Access Core (MA Core) shared component. Upgrading of other Java ES Release 3 components upon which Portal Server depends is therefore optional.

However, if you choose to upgrade all Portal Server dependencies, they should be upgraded in the following order, all before you upgrade Portal Server. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [“Upgrading Java ES Shared Components” on page 45](#).
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [“Directory Server and Administration Server” on page 61](#).
3. **Web Container Software.** Instructions for upgrading Web Server or Application Server are provided in [“Web Server” on page 81](#) and [“Application Server” on page 107](#), respectively.
4. **Access Manager (Access Manager SDK).** Instructions for upgrading Access Manager to Release 4 are provided in [“Access Manager” on page 115](#).

Back Up Release 3 Portal Server Configuration Information

Upgrade of Portal Server to Release 4 does not require the re-configuration of Portal Server software. However, as a safety measure you can back up the following directories where configuration information is stored:

```
/etc/PortalServer-base/
```

```
/var/PortalServer-base
```

Obtain Required Configuration Information and Passwords

You have to log in as superuser to perform the upgrade. If you are using Web Server as a web container, no configuration information is needed. However if you are using Application Server as a web container, you will need the Application Server administrator user ID and password.

Upgrading Portal Server

The upgrade of Portal Server software to Java ES Release 4 takes into account the following considerations:

- All Portal Server instances corresponding to the same installed Portal Server image are upgraded at the same time. All such instances should be shut down by shutting down the web container when patches are being applied to the installed image.
- The Release 4 Portal Server upgrade patches for HP-UX are shown in the following table:

Table 15-4 Patches* to Upgrade Portal Server

Component	Patch-ID
Portal Server Core	121519-01
Portal Server Locale	121930-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to Portal Server on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 15-4](#).

Always use the latest patch revision available, unless directed to use a specific revision.

Patches can be downloaded from:

<http://sunsolve.sun.com/>.

2. Log in as superuser.

```
su root
```

3. Stop Portal Server by stopping its web container.

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
ApplicationServer-base/bin/asadmin stop-domain domainName
```

```
ApplicationServer-base/bin/asadmin stop-node-agent hostName_domainName
```

4. If you have not already done so, upgrade the MA Core shared component and any others you wish to upgrade.

5. Apply the appropriate Portal Server patch in [Table 15-4](#).

```
swinstall -s <absolute-location>/<patch_ID> <patch_ID>
```

6. Confirm that the patch upgrade was successful:

```
swlist <patch_ID>
```

7. Restart Portal Server by restarting its web container.

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
ApplicationServer-base/bin/asadmin start-domain domainName --user  
admin_ID --password password
```

8. Re-deploy the Portal Server web application to your web container.

```
PortalServer-base/bin/deploy redeploy
```

The redeploy command redeploys content from *PortalServer-base*/web-src to /var/*PortalServer-base*/https-*hostName*/deploy-dir/web-apps. Any customizations to the Portal Server web application should therefore be first made to /web-src and then deployed to /web-apps. Any changes you might make under /web-apps should be replicated in /web-src before running the deploy command, or such changes will be overwritten.

9. Stop and restart the web container.

While not required in all situations, restarting the web container ensures that Portal Server starts in a clean state.

Verifying the Upgrade

The upgrade of Portal Server to Release 4 is verified by confirming that the upgrade patches have been properly applied. The steps for this verification were included in [“Upgrade Procedure” on page 187](#).

In addition, you can use the following command:

```
PortalServer-base/bin/version
```

See [Table 15-3 on page 186](#) for output values.

Beyond these tests of the patch upgrade you can verify that what previously worked still works and that bug fixes of interest have actually been fixed.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [“Upgrade Procedure” on page 187](#).

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Portal Server followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Portal Server is pretty much the reverse of the procedure for upgrading to Release 4. The re-configurations are rolled back and the patches are removed.

Rollback Procedure

1. Log in as superuser.

```
su root
```

2. Stop Portal Server by stopping its web container.

```
WebServer-base/https-instanceName/stop (Web Server)
```

```
ApplicationServer-base/bin/asadmin stop-domain --d instanceName (Application Server)
```

3. Remove the patches in [Table 15-4 on page 187](#).

```
swremove <patch_ID>
```

4. Restart Portal Server by restarting its web container.

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
ApplicationServer-base/bin/asadmin start-domain domainName --user  
admin_ID --password password
```

5. Re-deploy the Portal Server web application to your web container.

```
PortalServer-base/bin/deploy redeploy
```

The redeploy command redeploys content from *PortalServer-base*/web-src to /var/*PortalServer-base*/https-*hostName*/*deploy-dir*/web-apps. Any customizations to the Portal Server web application should therefore be first made to /web-src and then deployed to /web-apps. Any changes you might make under /web-apps should be replicated in /web-src *before* running the deploy command, or such changes will be overwritten.

6. Stop and restart the web container.

While not required in all situations, restarting the web container ensures that Portal Server starts in a clean state.

Multiple Instance Upgrades

In some deployment architectures Portal Server is deployed on multiple computer systems to provide for scalability and to improve availability. For example, you might have Portal Server components running on multiple computers with a load balancer to distribute the load.

In the case of load-balanced instances of Portal Server, you can perform a rolling upgrade in which you upgrade the Portal Server instances sequentially without interrupting service. You upgrade each instance of Portal Server while the others remain running. You perform the upgrade of each instance as described in [“Portal Server Upgrade” on page 185](#).

Portal Server Remote Access

This chapter describes how to upgrade Portal Server Secure Remote Access to Java ES 2005Q4 (Release 4): Sun Java System Portal Server Secure Remote Access 6 2005Q4.

The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [“Overview of Portal Server Secure Remote Access Upgrades” on page 191](#)
- [“Upgrading Portal Server Secure Remote Access from Java ES Release 3” on page 194](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *PortalServer-base*. It is the same directory path used for Portal Server file locations. At least part of this path might have been specified as an installation directory when Portal Server was initially installed. If not, the Java ES installer assigned a default value.

The default value of *PortalServer-base* is `/opt/sun/portal_svr`.

Overview of Portal Server Secure Remote Access Upgrades

This section describes the following general aspects of Portal Server Secure Remote Access that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Portal Server Secure Remote Access](#)

- [Portal Server Secure Remote Access Upgrade Roadmap](#)
- [Portal Server Secure Remote Access Data](#)
- [Compatibility Issues](#)
- [Portal Server Secure Remote Access Dependencies](#)

About Java ES Release 4 Portal Server Secure Remote Access

Java ES Release 4 Portal Server Secure Remote Access mostly represents bug fixes. There is no major new functionality with respect to Release 3.

Portal Server Secure Remote Access Upgrade Roadmap

[Table 16-1](#) shows the supported Portal Server Secure Remote Access upgrade paths to Java ES Release 4.

Table 16-1 Upgrade Paths to Java ES Release 4: Sun Java System Portal Server Secure Remote Access 6 2005Q4

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Portal Server Secure Remote Access 6 2005Q1	Direct upgrade: Performed by applying patches.	None

Portal Server Secure Remote Access Data

The following table shows the type of data that could be impacted by an upgrade of Portal Server Secure Remote Access software.

Table 16-2 Portal Server Secure Remote Access Data Usage

Type of Data	Location	Usage
Configuration data	/etc/PortalServer-base/ /var/PortalServer-base	Configuration of Portal Server Remote Access
Directory schema User data	Directory Server	Portal Server Secure Remote Access depends on user profile data that is stored in the directory.
Dynamic application data	None	Portal Server Secure Remote Access does not persistently store application data such as session state.

Compatibility Issues

Release 4 Portal Server Secure Remote Access does not introduce any interface changes. Portal Server Secure Remote Access is backwardly compatible with earlier versions.

Portal Server Secure Remote Access Dependencies

Portal Server Secure Remote Access dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Portal Server Secure Remote Access software. Changes in Portal Server Secure Remote Access interfaces or functions, for example, could require upgraded version of components upon which Portal Server Secure Remote Access depends. The need to upgrade such components depends upon the specific upgrade path.

Portal Server Secure Remote Access has dependencies on the following Java ES components:

- **Shared components.** Portal Server has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Portal Server.** Portal Server Secure Remote Access provides secure remote access to Portal Server.

- **Access Manager (or Access Manager SDK).** Portal Server depends upon Access Manager to provide authentication and authorization services for end users, including single sign-on.
- **Directory Server.** Portal Server accesses user data stored in Directory Server. As a result, Portal Server upgrades might require extensions of directory schema.

Upgrading Portal Server Secure Remote Access from Java ES Release 3

This section includes information about upgrading Portal Server Secure Remote Access from Java ES 2005Q1 (Release 3) to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Portal Server Secure Remote Access Upgrade](#)
- [Multiple Instance Upgrades](#)

Introduction

When upgrading Java ES Release 3 Portal Server Secure Remote Access to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Re-configuration of Portal Server Secure Remote Access is not required.
- **Upgrade Dependencies.** While Portal Server Secure Remote Access has dependencies on a number of Java ES shared components (see Table 1-6 on page 39), Release 4 Portal Server Secure Remote Access is compatible with the Release 3 version of these components. Upgrade of these shared components, except for Mobile Access Core (MA Core), is therefore optional with respect to upgrade of Portal Server Secure Remote Access to Release 4.

In addition, Release 4 Portal Server Secure Remote Access is dependent upon Portal Server, Access Manager, and Directory Server as described in “[Portal Server Secure Remote Access Dependencies](#)” on page 193. Upgrade of Portal Server Secure Remote Access to Release 4 requires that Portal Server also be

upgraded. However, the dependencies on Access Manager and Directory Server are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Portal Server Secure Remote Access to Release 4.

- **Backward Compatibility.** Release 4 Portal Server Secure Remote Access is backwardly compatible with the Release 3 version.
- **Upgrade Rollback.** Rollback of the Release 4 upgrade of Portal Server Secure Remote Access to Release 3 is achieved by rolling back the patches applied during the upgrade.

Portal Server Secure Remote Access Upgrade

This section describes how to perform an upgrade of Portal Server Secure Remote Access from Java ES Release 3 to Java ES Release 4. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Portal Server Secure Remote Access](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Portal Server Secure Remote Access you should perform the tasks described below

PortalServer-base/bin/gateway version

PortalServer-base/bin/rwproxyd version

PortalServer-base/bin/netletd version

Table 16-3 Portal Server Version Verification Outputs*

Java ES Release	Portal Server Version Number
Release 3	6.3.1
Release 4	6.3.1

*. The only difference between Release 3 and Release 4 is a patch. You can check for the Release 4 patches using the `swlist <patch_ID>`.

Upgrade Portal Server Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Portal Server has a hard upgrade dependency only on the Mobile Access Core (MA Core) shared component. Upgrading of other Java ES Release 3 components upon which Portal Server depends is therefore optional.

However, if you choose to upgrade all Portal Server Secure Remote Access dependencies, they should be upgraded in the following order, all before you upgrade Portal Server Secure Remote Access. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [“Upgrading Java ES Shared Components” on page 45](#).
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [“Directory Server and Administration Server” on page 61](#).
3. **Access Manager (Access Manager SDK).** Instructions for upgrading Access Manager to Release 4 are provided in [“Access Manager” on page 115](#).
4. **Portal Server.** Instructions for upgrading Portal Server are provided in [Chapter 15, “Portal Server” on page 181](#).

Back Up Release 3 Portal Server Secure Remote Access Configuration Information

Upgrade of Portal Server Secure Remote Access to Release 4 does not require the re-configuration of Portal Server Secure Remote Access software. However, as a safety measure you can back up the following directories where configuration information is stored:

```
/etc/PortalServer-base/
```

```
/var/PortalServer-base
```

Upgrading Portal Server Secure Remote Access

This section discusses considerations that impact the upgrade procedure for Portal Server Secure Remote Access followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Portal Server Secure Remote Access software to Java ES Release 4 takes into account the following considerations:

- All Portal Server Secure Remote Access instances corresponding to the same installed Portal Server Secure Remote Access image are upgraded at the same time. All such instances should be shut down when patches are being applied to the installed image.
- The Release 4 Portal Server Secure Remote Access upgrade patches for HP-UX are the same as those used to upgrade Portal Server and are shown in the following table:

Table 16-4 Patches* to Upgrade Portal Server Secure Remote Access

Component	Patch-ID
Portal Server Core	121519-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to Portal Server Secure Remote Access on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 16-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com>.

2. Log in as superuser.

```
su root
```

3. Stop Portal Server Secure Remote Access processes.

```
/sbin/init.d/gateway stop
```

```
/sbin/init.d/netletd stop
```

```
/sbin/init.d/rwproxyd stop
```

4. If you have not already done so, upgrade the MA Core shared component and any others you wish to upgrade.

5. Apply the appropriate Portal Server patches in [Table 16-4](#).

```
swinstall -s <absolute-location>/<patch_ID> <patch_ID>.
```

6. Confirm that the patch upgrade was successful:

```
swlist <patch_ID>
```

The output should return the versions of patch IDs applied in [Step 5](#).

7. Restart Portal Server Secure Remote Access processes.

```
/sbin/init.d/gateway start
```

```
/sbin/init.d/netletd start
```

```
/sbin/init.d/rwproxyd start
```

Verifying the Upgrade

The upgrade of Portal Server Secure Remote Access to Release 4 is verified by confirming that the upgrade patches have been properly applied. The steps for this verification were included in [“Verifying the Upgrade” on page 198](#).

In addition, you can use the following commands:

```
PortalServer-base/bin/gateway version
```

```
PortalServer-base/bin/rwproxyd version
```

```
PortalServer-base/bin/netletd version
```

See [Table 16-3 on page 195](#) for output values.

Beyond these tests of the patch upgrade you can verify that what previously worked still works and that bug fixes of interest have actually been fixed.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [“Upgrade Procedure” on page 197](#).

Rolling Back the Upgrade

This section describes considerations that impact the upgrade rollback procedure for Portal Server Secure Remote Access followed by the procedure itself.

Rollback Considerations

The procedure for rolling back the upgrade to Release 4 of Portal Server Secure Remote Access is pretty much the reverse of the procedure for upgrading to Release 4. The re-configurations are rolled back and the patches are removed.

Rollback Procedure

1. Log in as superuser.

```
su root
```

2. Stop Portal Server Secure Remote Access processes.

```
/sbin/init.d/gateway stop
```

```
/sbin/init.d/netletd stop
```

```
/sbin/init.d/rwproxyd stop
```

3. Remove the patches in [Table 16-4 on page 197](#).

```
swremove <patch_ID>
```

4. Restart Portal Server Secure Remote Access processes.

```
/sbin/init.d/gateway start
```

```
/sbin/init.d/netletd start
```

```
/sbin/init.d/rwproxyd start
```

Multiple Instance Upgrades

In some deployment architectures Portal Server Secure Remote Access is deployed on multiple computer systems to provide for scalability and to improve availability. For example, you might have Portal Server Secure Remote Access components running on multiple computers with a load balancer to distribute the load.

In the case of load-balanced instances of Portal Server Secure Remote Access, you can perform a rolling upgrade in which you upgrade the Portal Server Secure Remote Access instances sequentially without interrupting service. You upgrade each instance of Portal Server Secure Remote Access while the others remain running. You perform the upgrade of each instance as described in [“Portal Server Secure Remote Access Upgrade” on page 195](#).

Delegated Administrator

This chapter describes how to upgrade Delegated Administrator to Java ES 2005Q4 (Release 4): Sun Java System Communication Services Delegated Administrator 6 2005Q4. The chapter provides a general overview of upgrade issues and procedures for the different upgrade paths supported by Java ES Release 4.

- [“Overview of Delegated Administrator Upgrades” on page 201](#)
- [“Upgrading Delegated Administrator from Java ES Release 3” on page 204](#)

NOTE File locations in this chapter are specified with respect to a directory path referred to as *DelegatedAdmin-base*. At least part of this path might have been specified as an installation directory when Delegated Administrator was initially installed. If not, the Java ES installer assigned a default value.

The default value of *DelegatedAdmin-base* is
`/opt/sun/comms/commcli.`

Overview of Delegated Administrator Upgrades

This section describes the following general aspects of Delegated Administrator that impact upgrading to Java ES 2005Q4 (Release 4):

- [About Java ES Release 4 Delegated Administrator](#)
- [Delegated Administrator Upgrade Roadmap](#)
- [Delegated Administrator Data](#)
- [Compatibility Issues](#)

- [Delegated Administrator Dependencies](#)

About Java ES Release 4 Delegated Administrator

Java ES Release 4 Delegated Administrator new features with respect to Release 3 include calendar service provisioning, mail group provisioning, improved UI navigation based on usability feedback, and various bug fixes.

Delegated Administrator Upgrade Roadmap

[Table 17-1](#) shows the supported Delegated Administrator upgrade paths to Java ES Release 4.

Table 17-1 Upgrade Paths to Java ES Release 4: Sun Java System Communication Services Delegated Administrator 6 2005Q4

Java ES Release	Component	General Approach	Re-configuration Required
Release 3	Sun Java System Communication Services Delegated Administrator 66 2005Q1	Direct upgrade: Performed by applying patches and running <code>config-commda</code> utility.	Configuration data

Delegated Administrator Data

The following table shows the type of data that could be impacted by an upgrade of Delegated Administrator software.

Table 17-2 Delegated Administrator Data Usage

Type of Data	Location	Usage
Directory schema	Directory Server <code>user/group</code> directory	For attributes needed to support end users, organizations, and services schema

Table 17-2 Delegated Administrator Data Usage

Type of Data	Location	Usage
Configuration data	<i>DelegatedAdmin-base</i> /data/WEB-INF/classes/sun/comm/cli/server/servlet/resource.properties	Delegated Administrator server configuration and customizations
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/comm/sun/comm/da/resources/daconfig.properties	Delegated Administrator console configuration
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/comm/sun/comm/da/resources/logger.properties	Delegated Administrator console logging
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/comm/sun/comm/da/resources/security.properties	Delegated Administrator console configuration

Compatibility Issues

Release 4 Delegated Administrator introduces graphical user interface changes but is backwardly compatible with earlier versions.

Delegated Administrator Dependencies

Delegated Administrator dependencies on other Java ES components can impact the procedure for upgrading and re-configuring Delegated Administrator software. Changes in Delegated Administrator interfaces or functions, for example, could require upgraded version of components upon which Delegated Administrator depends. The need to upgrade such components depends upon the specific upgrade path.

Delegated Administrator has dependencies on the following Java ES components:

- **Shared components.** Delegated Administrator has dependencies on specific Java ES shared components (see [Table 1-6 on page 33](#)).
- **Web Container.** Delegated Administrator depends upon web container services, which can be provided either by Java ES Web Server or Java ES Application Server.

- **Directory Server.** Delegated Administrator stores application and user data in Directory Server.
- **Directory Preparation Tool.** Delegated Administrator uses the Directory Preparation Tool to prepare the directory to support Delegated Administrator user provisioning functions. As a result, Delegated Administrator upgrades might depend upon preparation of the directory to support new functions.
- **Access Manager (Access Manager SDK).** Delegated Administrator depends upon Access Manager to register services and to make entries into Directory Server.

Upgrading Delegated Administrator from Java ES Release 3

This section includes information about upgrading Delegated Administrator from Java ES 2005Q1 (Release 3) to Java ES Release 4. The section covers the following topics:

- [Introduction](#)
- [Delegated Administrator Upgrade](#)

Introduction

When upgrading Java ES Release 3 Delegated Administrator to Release 4, consider the following aspects of the upgrade process:

- **General Upgrade Approach.** The upgrade is performed by applying patches to the Release 3 version. Re-configuration of Delegated Administrator is achieved by running the config-commda configuration utility.
- **Upgrade Dependencies.** While Delegated Administrator has dependencies on a number of Java ES shared components (see [Table 1-6 on page 33](#)), Release 4 Delegated Administrator is compatible with the Release 3 versions of these components. Upgrade of these shared components is therefore optional with respect to upgrade of Delegated Administrator to Release 4.

In addition, Release 4 Delegated Administrator is dependent upon a web container and on Access Manager, as described in “[Delegated Administrator Dependencies](#)” on page 203. These are soft upgrade dependencies; upgrade of these components is optional with respect to upgrade of Delegated Administrator to Release 4.

However, Release 4 Delegated Administrator has a hard upgrade dependency on Directory Preparation Tool; Release 4 Directory Preparation Tool is required to prepare Directory Server for user provisioning operations.

- **Backward Compatibility.** Release 4 Delegated Administrator is backwardly compatible with the Release 3 version.

Delegated Administrator Upgrade

This section describes how to perform an upgrade of Delegated Administrator from Java ES Release 3 to Java ES Release 4 on HP-UX. The section covers the following topics:

- [Pre-Upgrade Tasks](#)
- [Upgrading Delegated Administrator](#)
- [Verifying the Upgrade](#)
- [Post-Upgrade Tasks](#)
- [Rolling Back the Upgrade](#)

Pre-Upgrade Tasks

Before you upgrade Delegated Administrator you should perform the tasks described below.

Verify Current Version Information

You can verify the current version of Delegated Administrator by entering the following command:

```
DelegatedAdmin-base/bin/commadmin -V
```

Table 17-3 Delegated Administrator Version Verification Outputs

Java ES Release	Delegated Administrator Version Number
Release 3	User Management Utility 6 2005Q1

Table 17-3 Delegated Administrator Version Verification Outputs

Java ES Release	Delegated Administrator Version Number
Release 4	Delegated Administrator 6.3-0.09

Upgrade Delegated Administrator Dependencies

It is generally recommended that all Java ES components on a computer system (and in a computing environment) be upgraded to Java ES Release 4. However, Delegated Administrator has a hard upgrade dependency only on Directory Preparation Tool. Upgrading of other Java ES Release 3 components upon which Delegated Administrator depends is therefore optional.

However, if you choose to upgrade all Delegated Administrator dependencies, they should be upgraded in the following order, all before you upgrade Delegated Administrator. You can skip any that might already have been upgraded.

1. **Shared Components.** Instructions for upgrading Java ES shared components to Release 4 are provided in [Chapter 2, “Upgrading Java ES Shared Components”](#) on page 45.
2. **Directory Server.** Instructions for upgrading Directory Server to Release 4 are provided in [Chapter 3, “Directory Server and Administration Server”](#) on page 61.
3. **Web Container Software.** Instructions for upgrading Web Server or Application Server are provided in [Chapter 5, “Web Server”](#) on page 81 and [Chapter 8, “Application Server”](#) on page 107, respectively.
4. **Access Manager (Access Manager SDK).** Instructions for upgrading Access Manager to Release 4 are provided in [Chapter 9, “Access Manager”](#) on page 115.
5. **Directory Preparation Tool.** Release 4 Directory Preparation Tool needs to have been run against Directory Server before using Release 4 Delegated Administrator. If Directory Preparation Tool has not already been run against Directory Server, upgrade Directory Preparation Tool to Release 4 and use it to modify and extend the schema of Directory Server (see [Chapter 10, “Directory Preparation Tool”](#) on page 135 for procedures).

Back Up Delegated Administrator Data

The Delegated Administrator upgrade from Release 3 to Release 4 requires re-configuration of Delegated Administrator. It is a good idea to back up configuration data as a safety precaution, and to back up any Release 3 graphical user interface customizations.

Obtain Required Configuration Information and Passwords

You should know the following information about your currently installed version:

- Access Manager administrator user ID and password
- Access Manager internal LDAP password
- Top level administrator user ID and password
- Application Server administrator user ID and password, if you are using
- Application Server as a web container

Upgrading Delegated Administrator

This section discusses considerations that impact the upgrade procedure for Delegated Administrator followed by a description of the procedure itself.

Upgrade Considerations

The upgrade of Delegated Administrator software to Java ES Release 4 takes into account the following considerations:

- Delegated Administrator should not be used while patches are being applied to the installed image.
- When re-configuring Delegated Administrator, it should be deployed to the same web container as Access Manager.
- The Release 4 Delegated Administrator upgrade patch for HP-UX is shown in the following table:

Table 17-4 Patches* to Upgrade Delegated Administrator

Component	Patch-ID
Delegated Administrator	121517-01

*. Patch revision numbers are the minimum required for upgrade to Java ES Release 4. If newer revisions become available, use the newer ones instead of those shown in the table.

Upgrade Procedure

The procedure documented below applies to Delegated Administrator on the computer where the upgrade is taking place.

1. Obtain the required patches, based on [Table 17-4](#).

Patches can be downloaded from:

<http://sunsolve.sun.com/>.

2. Log in as superuser.

```
su root
```

3. Apply the appropriate Delegated Administrator patches in [Table 17-4](#).

- a. If the product is in the default location (`/opt/sun/comms/commcli`)

```
swinstall -s patch_location <patch_ID>
```

- b. If the product is in non default location

```
swinstall -s patch_location -x patch_match_target = true
```

4. Reconfigure Delegated Administrator.

- a. Make sure the web container running Access Manager and Delegated Administrator is running.

To start the web container:

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
ApplicationServer-base/bin/asadmin start-domain domainName
```

```
--user admin_ID --password password
```

where Access Manager and Delegated Administrator are deployed in a server instance in the `domainName` domain. The default `domainName` is `domain1` and the default server instance is `server1`.

- b. Run the Delegated Administrator configuration utility.

```
DelegatedAdmin-base/sbin/config-commda
```

For details of how to use this utility, see [Chapter 3 Configuring Delegated Administrator of the Sun Java System Communications Services 6 2005Q4 Delegated Administration Guide](#) (<http://docs.sun.com/doc/819-2658>).

5. Stop and restart the Delegated Administrator web container.

Web Server:

```
WebServer-base/https-instanceName/stop
```

```
WebServer-base/https-instanceName/start
```

Application Server:

```
ApplicationServer-base/bin/asadmin stop-domain domainName
```

```
ApplicationServer-base/bin/asadmin start-domain domainName
```

```
--user admin_ID --password password
```

Verifying the Upgrade

You can verify successful upgrade of Delegated Administrator as follows:

1. Check the version number.

```
DelegatedAdmin-base/bin/commadmin -V
```

See [Table 17-4 on page 207](#) for output values.

2. Log in to the Delegated Administrator console using the top level administrator user ID and password specified during re-configuration of Delegated Administrator.

```
http://hostName:port/da/DA/Login
```

where *hostName:port* are values provided during re-configuration of Delegated Administrator.

Post-Upgrade Tasks

There are no post-upgrade tasks beyond the steps described in [“Upgrade Procedure” on page 207](#).

Rolling Back the Upgrade

Rollback of Delegated Administrator is not supported. Changes made during the upgrade procedure, such as entries in Directory Server or in deploying Delegated Administrator into the web container cannot easily be backed out.

Java System Release Contents

This appendix lists the selectable and shared components that are part of the Sun Java™ Enterprise System (Java ES) software.

- [Java ES 2005Q1 \(Release 3\)](#)
- [Java ES 2005Q4 \(Release 4\)](#)

Java ES 2005Q1 (Release 3)

This section lists the contents of Java Enterprise System 2005Q1.

Release 3 Installer Selectable Components

In the component selection page of the Java ES installer, the selectable components are grouped by the services they help to provide. The following list also shows the subcomponents that are installed with each component.

Communication & Collaboration Services

- Sun Java System Messaging Server 6 2005Q1
- Sun Java System Calendar Server 6 2005Q1
- Sun Java System Instant Messaging 7 2005Q1
 - Instant Messaging Server Core; includes server and multiplexor software
 - Instant Messaging Resources
 - Access Manager Instant Messaging Service
- Sun Java System Portal Server 6 2005Q1

- Sun Java System Portal Server Secure Remote Access 6 2005Q1
 - Secure Remote Access Core
 - Gateway
 - Netlet Proxy
 - Rewriter Proxy
- Sun Java System Communications Express 2005Q1
- Sun Java System Directory Preparation Script

Web & Application Services

- Sun Java System Application Server Enterprise Edition 8.1 2005Q1
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin
 - Can be used with Sun Java System WebServer 6.1SP4
 - Sample Applications
 - PointBase
- Sun Java System Web Server 6 2005Q1 Update 1 Service Pack 4
- Sun Java System Message Queue 3.6 2005Q1

Directory & Identity Services

- Sun Java System Access Manager 6.3 2005Q1

Delegated Administrator provisioning tools for Portal Server and Messaging Server are automatically installed with Access Manager.

- Identity Management and Policy Services Core (includes Delegated Administrator Utility)
 - Access Manager Administration Console
 - Common Domain Services for Federation Management
 - Access Manager SDK
- Sun Java System Directory Server 5 2005Q1

- Sun Java System Directory Proxy Server 5 2005Q1

Administrative Services

- Sun Java System Administration Server 5 2005Q1

Release 3 Shared Components

This release of Java ES includes these shared components:

- Ant (Jakarta ANT Java/XML-based build tool)
- Apache SOAP (Simple Object Access Protocol) Runtime
- Berkeley Database
- Common agent container
- ICU (International Components for Unicode)
- Instant Messenger SDK
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio Enterprise Web Application Framework)
- JavaHelp™ Runtime
- JavaMail™
- JAXB (Java Architecture for XML Binding) Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries) Runtime
- JAX-RPC (Java API for XML-based Remote Procedure Call) Runtime
- JCAPI (Java Calendar API)
- JDMK (Java Dynamic Management™ Kit) Runtime
- JSS (Java Security Services)
- KTSE (KT Search Engine)
- LDAP C SDK
- LDAP Java SDK
- NSPR (Netscape Portable Runtime)

- NSS (Network Security Services)
- Perl LDAP, including NSPERL
- SAAJ (SOAP with Attachments API for Java)
- SAML (Security Assertions Markup Language)
- SASL (Simple Authentication and Security Layer)
- SNMP (Simple Network Management Protocol) Peer
- MFWK (Sun Java Monitoring Framework)
- Sun Java Web Console
- Tomcat Servlet JSP Container
- XML C Library (libxml)
- WSCL (Web services Common Library)
- ZLib

Java ES 2005Q4 (Release 4)

This section lists the contents of Java Enterprise System 2005Q4.

Release 4 Installer-Selectable Components

In the component selection page of the Java ES installer, the selectable components are grouped by the services they help to provide. The following list also shows the subcomponents that are installed with each component.

Communication & Collaboration Services

- Sun Java System Messaging Server 6.2 2005Q4
- Sun Java System Calendar Server 6.2 2005Q4
- Sun Java System Instant Messaging 7.0.1 2005Q4
 - Instant Messaging Server Core; includes server and multiplexor software
 - Instant Messaging Resources
 - Access Manager Instant Messaging Service

- Sun Java System Portal Server 6.3.1 2005Q4
- Sun Java System Portal Server Secure Remote Access 6 2005Q4
 - Secure Remote Access Core
 - Gateway
 - Netlet Proxy
 - Rewriter Proxy
- Sun Java System Communications Express 6.3 2005Q4
- Sun Java System Directory Preparation Script 6.3

Web & Application Services

- Sun Java System Application Server Enterprise Edition 8.1 2005Q4
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin
 - Can be used with Sun Java System WebServer 6.1SP5
 - PointBase
 - Sample Applications
- Sun Java System Web Server 6 Update 1 Service Pack 5 2005Q4
- Sun Java Proxy Server 4.0 2005Q4
- Sun Java System Message Queue 3.6 SP3 2005Q4
- Sun Java Registry Server 3.0

Directory & Identity Services

- Sun Java System Access Manager 7.0 2005Q4

Delegated Administrator provisioning tools for Portal Server and Messaging Server are automatically installed with Access Manager.

 - Identity Management and Policy Services Core (includes Delegated Administrator 6.3)
 - Access Manager Administration Console

- Common Domain Services for Federation Management
- Access Manager SDK

Administrative Services

- Sun Java System Administration Server 5.2 2005Q4

Release 4 Shared Components

Shared components provide the local services and technology support for the selectable components.

This release of Java ES includes these shared components:

- ANT (Jakarta ANT Java/XML-based build tool)
- ACL (Apache Common Logging)
- BDB (Berkeley Database)
- Common agent container
- Derby
- ICU (International Components for Unicode)
- Instant Messenger SDK
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio Enterprise Web Application Framework)
- JavaHelp™ Runtime
- JavaMail™
- JAXB (Java Architecture for XML Binding) Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries) Runtime
- JAX-RPC (Java API for XML-based Remote Procedure Call) Runtime
- JCAPI (Java Calendar API)
- JDMK (Java Dynamic Management™ Kit) Runtime
- JSS (Java Security Services)
- KTSE (KT Search Engine)

- LDAP C SDK
- LDAP Java SDK
- MFWK (Sun Java Monitoring Framework)
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- SAAJ (SOAP runtime with Attachments API for Java)
- SASL (Simple Authentication and Security Layer)
- WSCL (Web services Common Library)
- Zlib

Index

A

- Abbreviation 25
- abbreviation 24
- Access Manager
 - abbreviation 24
- Administration Server
 - abbreviation 24
 - compatibility issues 63
 - dependencies 37
 - patches 67
 - rollback upgrade 69
- amconfig script 121, 122
- amupgrade script 121, 129, 131
- Ant 25, 48, 53, 213, 216
- Apache Common Logging 25, 216
- Apache SOAP (Simple Object Access Protocol)
 - Runtime 213
- Application Server
 - abbreviation 24
 - dependencies 37
 - subcomponents 212, 215

B

- backing up
 - Directory Server data 66, 75, 123

- Messaging Server data 148
- BDB 25, 48, 53, 213, 216

C

- Calendar Server
 - abbreviation 24
 - compatibility issues 154
 - dependencies 38
 - patches 158
 - rollback upgrade 159
- common agent container 25, 48, 53
 - 213, 216
- Communications Express
 - abbreviation 24
 - dependencies 38, 40
- component products
 - dependencies 32
 - Java ES 2005Q1 211
 - Java ES 2005Q4 214
- configuration files
 - instance 89
 - location 89
- customized JSPs 127

D

db2bak [66, 75, 123](#)

Delegated Admin
 abbreviation [24](#)
 component [39](#)
 dependencies [33](#)

Delegated Administrator [212, 215](#)

DerbyDB [25, 48, 53, 216](#)

Directory Preparation Tool
 abbreviation [24](#)
 dependencies [39](#)
 patches [139](#)

Directory Proxy Server
 abbreviation [24](#)
 dependencies [39](#)
 patches [67, 76](#)

Directory Server
 abbreviation [24](#)
 compatibility issues [63](#)
 dependencies [39](#)
 patches [67, 76](#)
 rollback upgrade [69](#)

documentation [18](#)

I

ICU
 [25, 48, 53, 213, 216](#)

Instant Messaging
 abbreviation [24](#)
 dependencies [39](#)
 subcomponents [211, 214](#)

J

J2SE [58](#)

JAF [25, 213, 216](#)

JATO [25, 48, 53, 213, 216](#)

Java ES 2005Q1
 selectable components [211](#)

 shared components [213](#)

Java ES 2005Q4
 selectable components [214](#)
 shared components [216](#)

JAXB [25, 49, 54, 213, 216](#)

JAXP [25, 49, 54, 213, 216](#)

JAXR [25, 49, 54, 213, 216](#)

JAX-RPC [25, 213, 216](#)

JCAPI [25, 49, 54, 213, 216](#)

JDMK [25, 213, 216](#)

JHELP [25, 213, 216](#)

JMAIL [25, 213, 216](#)

JSS [25, 49, 55, 213, 216](#)

K

KTSE [25, 49, 55, 213, 216](#)

L

LDAP C SDK [25, 49, 55, 213, 217](#)

LDAP J SDK [49, 55](#)

LDAP Java SDK [25, 213, 217](#)

M

Message Queue
 abbreviation [24](#)
 dependencies [39](#)

Messaging Server
 abbreviation [24](#)
 compatibility issues [145](#)
 dependencies [40](#)
 patches [149](#)
 rollback upgrade [151](#)

MFWK [26, 49, 55, 214, 217](#)

N

NSPR [25](#), [49](#), [55](#), [213](#), [217](#)

NSS [25](#), [49](#), [55](#), [214](#), [217](#)

Z

Zlib [26](#), [49](#), [56](#), [214](#), [217](#)

P

passfile [89](#)

password requirements [66](#), [76](#), [84](#), [112](#), [124](#), [139](#), [148](#),
[158](#), [168](#), [178](#), [187](#), [207](#)

patches [67](#), [76](#), [126](#), [139](#)

Portal Server

 abbreviation [24](#)

 dependencies [40](#)

Portal Server Secure Remote Access

 abbreviation [24](#)

 dependencies [40](#)

S

SAAJ [25](#), [49](#), [55](#), [214](#), [217](#)

SASL [25](#), [49](#), [55](#), [214](#), [217](#)

Service Registry

 abbreviation [24](#)

W

Web Proxy Server

 abbreviation [24](#)

Web Server

 abbreviation [24](#)

 compatibility issues [82](#)

 dependencies [40](#)

 patches [85](#)

 roll back upgrade [86](#)

WSCL [26](#), [49](#), [56](#), [214](#), [217](#)

