



Sun Java™ System

Sun Java Enterprise System 2005Q4

升级指南

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

文件号码 819-3457

版权所有 © 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 保留所有权利。

对于本文档中介绍的产品，Sun Microsystems, Inc. 对其所涉及的技术拥有相关的知识产权。需特别指出的是（但不局限于此），这些知识产权可能包含在 <http://www.sun.com/patents> 中列出的一项或多项美国专利，以及在美国和其他国家/地区申请的一项或多项其他专利或待批专利。

本产品包含 SUN MICROSYSTEMS, INC. 的机密信息和商业秘密。未经 SUN MICROSYSTEMS, INC. 的事先明确书面许可，不得使用、泄露或复制。

美国政府权利——商业软件。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（Federal Acquisition Regulations，即“联邦政府采购法规”）的适用条款及其补充条款。

本发行版可能包含由第三方开发的内容。

本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地区独家许可的注册商标。

Sun、Sun Microsystems、Sun 徽标、Java、Solaris、JDK、Java Naming and Directory Interface、JavaMail、JavaHelp、J2SE、iPlanet、Duke 徽标、Java 咖啡杯徽标、Solaris 徽标、SunTone Certified 徽标和 Sun ONE 徽标是 Sun Microsystems, Inc. 在美国和其他国家/地区的商标或注册商标。

所有的 SPARC 商标的使用均已获得许可，它们是 SPARC International, Inc. 在美国和其他国家/地区的商标或注册商标。标有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。

Legato 和 Legato 徽标是注册商标，它们和 Legato NetWorker 都是 Legato Systems, Inc. 的商标或注册商标。Netscape Communications Corp 徽标是 Netscape Communications Corporation 的商标或注册商标。

OPEN LOOK 和 Sun(TM) 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。Sun 感谢 Xerox 在研究和开发可视或图形用户界面的概念方面为计算机行业所做的开拓性贡献。Sun 已从 Xerox 获得了对 Xerox 图形用户界面的非独占性许可证，该许可证还适用于实现 OPEN LOOK GUI 和在其他方面遵守 Sun 书面许可协议的 Sun 许可证持有者。

本服务手册所介绍的产品以及所包含的信息受美国出口控制法制约，并应遵守其他国家/地区的进出口法律。严禁将本产品直接或间接地用于核设施、导弹、生化武器或海上核设施，也不能直接或间接地出口给核设施、导弹、生化武器或海上核设施的最终用户。严禁出口或转口到美国禁运的国家/地区以及美国禁止出口清单中所包含的实体，包括但不限于被禁止的个人以及特别指定的国家/地区的公民。

本文档按“原样”提供，对于所有明示或默示的条件、陈述和担保，包括对适销性、适用性或非侵权性的默示保证，均不承担任何责任，除非此免责声明的适用范围在法律上无效。

目录

表格列表	15
前言	19
目标读者	20
本书所采用的约定	20
印刷约定	20
符号	21
Shell 提示符	21
相关文档	22
本文档集中的文档	22
联机访问 Sun 资源	23
联系 Sun 技术支持	23
相关的第三方 Web 站点引用	24
Sun 欢迎您提出意见	24
第 1 章 升级规划	25
Java ES 2005Q4 (发行版本 4) 组件	26
发行版本 4 产品组件	26
发行版本 4 共享组件	27
关于 Java ES 升级	29
产品组件升级	29
共享组件升级	29
升级技术	30
操作系统问题	31
所需的操作系统修补程序	31
次发行版本升级	31
非支持平台的升级	31
升级规划	32
什么是升级计划?	32
升级计划注意事项	33
升级途径	33
升级依赖性	35
选择性升级或全部升级	35
多实例升级	36

Java ES 组件依赖性	37
对共享组件的依赖性	37
共享组件依赖性矩阵	37
共享组件升级指导原则	40
对产品组件的依赖性	40
一般先后顺序指导原则	43
第 2 章 升级 Java ES 共享组件	47
共享组件升级概述	48
关于升级计划	48
升级共享组件使用的技术	49
常规升级过程	50
升级发行版本 3 的共享组件	50
升级发行版本 2 的共享组件	52
使用单个修补程序升级共享组件	54
升级到 Java ES 2005Q4 的修补程序	55
修补程序升级回滚	55
使用修补程序群组升级共享组件	56
修补程序群组内容	56
修补程序群组过程	57
从 Solaris 8 和 Solaris 9 升级	57
在 Solaris 10 上升级（仅从 Java ES 发行版本 3）	58
通过替换软件包升级组件	59
软件包替换的升级策略	60
升级途径	60
软件包版本	60
软件包位置	60
软件包替换的过程	61
特殊说明	61
Solaris 平台的软件包	61
在 Solaris 平台上升级软件包	63
Linux 平台的软件包	64
在 Linux 平台上升级软件包	66
需要特殊升级过程的组件	67
升级安全组件（NSS、NSPR、JSS）	67
升级公共代理容器	67
在 Solaris 平台上从 Java ES 发行版本 2 升级	68
在 Linux 平台上从 Java ES 发行版本 2 升级	68
从 Java ES 发行版本 3 升级	68
升级 JATO	71
在 Solaris 平台上从 Java ES 发行版本 3 升级 JATO	71
在 Solaris 平台上从 Java ES 发行版本 2 升级 JATO	71
在 Linux 平台上从 Java ES 发行版本 3 升级 JATO	71
在 Linux 平台上从 Java ES 发行版本 2 升级 JATO	71

在 Linux 平台升级 JavaHelp	71
升级 Sun Java Web Console	72
升级 Sun Explorer Data Collector	72
升级 Java ES 发行版本 4 的 J2SE	73
在 Solaris 平台升级 J2SE	73
在 Linux 平台升级 J2SE	79
第 3 章 Sun Cluster 软件	83
Sun Cluster 软件升级概述	84
关于 Java ES 发行版本 4 Sun Cluster 软件	84
Sun Cluster 软件升级指导说明	84
Sun Cluster 数据	85
兼容性问题	85
Sun Cluster 组件依赖性	86
将 Sun Cluster 软件升级到 Java ES 发行版本 4	86
简介	86
Sun Cluster 升级	87
升级前的任务	87
升级 Sun Cluster 软件	89
检验升级	90
升级后的任务	90
升级回滚	90
第 4 章 Directory Server 和 Administration Server	91
Directory Server 和 Administration Server 升级概述	92
关于 Java ES 发行版本 4	92
Java ES 发行版本 4 升级指导说明	92
Directory Server 和 Administration Server 数据	93
兼容性问题	94
组件依赖性	94
从 Java ES 发行版本 3 升级 Directory Server 和 Administration Server	95
简介	95
发行版本 3 Directory Server 和 Administration Server 的升级	96
升级前的任务	96
升级发行版本 3 Directory Server 和 Administration Server (Solaris)	97
升级发行版本 3 Directory Server 和 Administration Server (Linux)	102
检验升级	106
升级后的任务	106
升级回滚 (Solaris)	106
多实例升级	108
多主复制的滚动升级	108
作为数据服务升级 Directory Server	109
从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server	110

第 5 章 Directory Proxy Server	111
Directory Proxy Server 升级概述	112
关于 Java ES 发行版本 4	112
Java ES 发行版本 4 升级指导说明	112
Directory Proxy Server 数据	113
兼容性问题	114
组件依赖性	114
从 Java ES 发行版本 3 升级 Directory Proxy Server	114
简介	114
发行版本 3 Directory Proxy Server 的升级	115
升级前的任务	116
升级发行版本 3 Directory Proxy Server (Solaris)	117
升级发行版本 3 Directory Proxy Server (Linux)	119
检验升级	121
升级后的任务	121
升级回滚 (Solaris)	121
多实例升级	122
从 Java ES 发行版本 2 升级 Directory Proxy Server	122
第 6 章 Web Server	123
Web Server 升级概述	124
关于 Java ES 发行版本 4 Web Server	124
Web Server 升级指导说明	124
Web Server 数据	125
兼容性问题	125
Web Server 组件依赖性	125
从 Java ES 发行版本 3 升级 Web Server	126
简介	126
发行版本 3 Web Server 的升级	127
升级前的任务	127
升级发行版本 3 Web Server (Solaris)	128
升级发行版本 3 Web Server (Linux)	129
检验升级	131
升级后的任务	131
升级回滚 (Solaris)	131
从 Java ES 发行版本 2 升级 Web Server	132

第 7 章 Message Queue	133
Message Queue 升级概述	134
关于 Java ES 发行版本 4 Message Queue	134
Message Queue 升级指导说明	135
Message Queue 数据	136
兼容性问题	136
协议兼容性	136
代理兼容性	137
管理对象兼容性	137
管理工具兼容性	138
客户机兼容性	138
Message Queue 组件依赖性	138
从 Java ES 发行版本 3 升级 Message Queue	139
简介	139
发行版本 3 Message Queue 的升级	140
升级前的任务	140
升级发行版本 3 Message Queue	141
检验 Message Queue 升级	143
升级后的任务	143
升级回滚	143
多实例升级	144
从 Java ES 发行版本 2 升级 Message Queue	145
升级发行版本 2 的 Message Queue (Solaris)	145
升级发行版本 2 的 Message Queue (Linux)	145
升级过程	145
安装兼容性软件包	146
升级后的任务	147
第 8 章 High Availability Session Store	149
HADB 升级概述	150
关于 Java ES 发行版本 4 HADB	150
HADB 升级指导说明	150
HADB 数据	151
兼容性问题	151
HADB 依赖性	151
从 Java ES 发行版本 3 升级 HADB	152
简介	152
发行版本 3 HADB 的升级	153
升级前的任务	153
升级发行版本 3 HADB	154
检验升级	155
升级后的任务	156
升级回滚	156

第 9 章 Application Server	157
Application Server 升级概述	158
关于 Java ES 发行版本 4 Application Server	158
Application Server 升级指导说明	158
Application Server 数据	159
兼容性问题	160
Application Server 组件依赖性	160
从 Java ES 发行版本 3 升级 Application Server	161
简介	161
发行版本 3 Application Server 的升级	162
升级前的任务	162
升级发行版本 3 Application Server (Solaris)	163
升级发行版本 3 Application Server (Linux)	165
检验升级	168
升级后的任务	168
升级回滚 (Solaris)	168
从 Java ES 发行版本 2 升级 Application Server	169
简介	169
发行版本 2 Application Server 的升级	170
升级前的任务	170
升级发行版本 2 Application Server	171
检验升级	172
升级后的任务	173
升级回滚	173
多实例（群集）升级:	173
在以后再配置安装后配置 Application Server	173
第 10 章 Web Proxy Server	175
Web Proxy Server 升级概述	176
关于 Java ES 发行版本 4 Web Proxy Server	176
Web Proxy Server 升级指导说明	176
Web Proxy Server 数据	177
兼容性问题	177
Web Proxy Server 组件依赖性	177
将 Web Proxy Server 升级到发行版本 4	178
简介	178
Web Proxy Server 升级	179
升级前的任务	179
升级 Web Proxy Server	180
检验升级	181
升级后的任务	181
升级回滚	181

第 11 章 Access Manager	183
Access Manager 升级概述	184
关于 Java ES 发行版本 4 Access Manager	184
Access Manager 升级指导说明	185
Access Manager 数据	186
兼容性问题	187
Access Manager 组件依赖性	188
从 Java ES 发行版本 3 升级 Access Manager	189
简介	189
发行版本 3 Access Manager 的完全升级	190
升级前的任务	190
升级发行版本 3 Access Manager	193
检验 Access Manager 升级	199
升级后的任务	200
升级回滚	200
多实例升级：发行版本 3 与发行版本 4 共存	200
仅升级发行版本 3 Access Manager SDK	201
升级前的任务	201
升级发行版本 3 Access Manager SDK	202
检验 Access Manager SDK 升级	202
升级回滚	202
从 Java ES 发行版本 2 升级 Access Manager	203
升级前的任务	203
升级 Access Manager 依赖项	203
升级目录模式	203
发行版本 2 Access Manager 的升级	204
升级发行版本 2 Access Manager: Web Server Web 容器	204
升级发行版本 2 Access Manager: Application Server Web 容器	204
检验 Access Manager 升级	208
升级后的任务	208
升级回滚	208
第 12 章 Directory Preparation Tool	209
Directory Preparation Tool 升级概述	210
关于 Java ES 发行版本 4 Directory Preparation Tool	210
Directory Preparation Tool 升级指导说明	210
Directory Preparation Tool 数据	211
兼容性问题	211
Directory Preparation Tool 依赖性	211

从 Java ES 发行版本 3 升级 Directory Preparation Tool	212
简介	212
发行版本 3 Directory Preparation Tool 的升级	213
升级前的任务	213
升级发行版本 3 Directory Preparation Tool (Solaris)	214
升级发行版本 3 Directory Preparation Tool (Linux)	215
检验升级	217
升级后的任务	217
升级回滚 (Solaris)	217
从 Java ES 发行版本 2 升级 Directory Preparation Tool	218
发行版本 2 升级过程 (Solaris)	219
发行版本 2 升级过程 (Linux)	220

第 13 章 Messaging Server **223**

Messaging Server 升级概述	224
关于 Java ES 发行版本 4 Messaging Server	224
Messaging Server 升级指导说明	224
Messaging Server 数据	225
兼容性问题	226
Messaging Server 组件依赖性	226
从 Java ES 发行版本 3 升级 Messaging Server	227
简介	227
发行版本 3 Messaging Server 的升级	228
升级前的任务	228
升级发行版本 3 Messaging Server (Solaris)	230
升级发行版本 3 Messaging Server (Linux)	232
检验升级	235
升级后的任务	235
升级回滚 (Solaris)	235
多实例升级	236
从 Java ES 发行版本 2 升级 Messaging Server	237
升级 Messaging Server 依赖项	237
发行版本 2 Messaging Server 的升级	237
升级发行版本 2 Messaging Server (Solaris)	237
升级发行版本 2 Messaging Server (Linux)	238
检验升级	239
升级后的任务	239

第 14 章 Calendar Server	241
Calendar Server 升级概述	242
关于 Java ES 发行版本 4 Calendar Server	242
Calendar Server 升级指导说明	242
Calendar Server 数据	243
兼容性问题	243
Calendar Server 组件依赖性	243
从 Java ES 发行版本 3 升级 Calendar Server	244
简介	244
发行版本 3 Calendar Server 的升级	245
升级前的任务	245
升级发行版本 3 Calendar Server (Solaris)	247
升级发行版本 3 Calendar Server (Linux)	248
检验升级	250
升级后的任务	250
升级回滚 (Solaris)	250
多实例升级	251
从 Java ES 发行版本 2 升级 Calendar Server	251
第 15 章 Communications Express	253
Communications Express 升级概述	254
关于 Java ES 发行版本 4 Communications Express	254
Communications Express 升级指导说明	254
Communications Express 数据	255
兼容性问题	255
Communications Express 组件依赖性	256
从 Java ES 发行版本 3 升级 Communications Express	257
简介	257
发行版本 3 Communications Express 的升级	258
升级前的任务	258
升级发行版本 3 Communications Express (Solaris)	260
升级发行版本 3 Communications Express (Linux)	262
检验升级	264
升级后的任务	265
升级回滚 (Solaris)	265
多实例升级	266
从 Java ES 发行版本 2 升级 Communications Express	266
升级 Communications Express 依赖项	266
发行版本 2 Communications Express 的升级	267
升级发行版本 2 Communications Express: Web Server Web 容器	267
升级发行版本 2 Communications Express: Application Server Web 容器	267

第 16 章 Instant Messaging	269
Instant Messaging 升级概述	270
关于 Java ES 发行版本 4 Instant Messaging	270
Instant Messaging 升级指导说明	270
Instant Messaging 数据	271
兼容性问题	271
Instant Messaging 组件依赖性	271
从 Java ES 发行版本 3 升级 Instant Messaging	272
简介	272
发行版本 3 Instant Messaging 的升级	273
升级前的任务	273
升级发行版本 3 Instant Messaging (Solaris)	274
升级发行版本 3 Instant Messaging (Linux)	276
检验升级	277
升级后的任务	278
升级回滚 (Solaris)	278
多实例升级	278
从 Java ES 发行版本 2 升级 Instant Messaging	279
简介	279
发行版本 2 Instant Messaging 的升级	280
升级前的任务	280
升级发行版本 2 Instant Messaging (Solaris)	281
升级发行版本 2 Instant Messaging (Linux)	283
检验升级	284
升级后的任务	284
升级回滚	285
多实例升级	285
第 17 章 Portal Server	287
Portal Server 升级概述	288
关于 Java ES 发行版本 4 Portal Server	288
Portal Server 升级指导说明	288
Portal Server 数据	289
兼容性问题	290
Portal Server 组件依赖性	290
从 Java ES 发行版本 3 升级 Portal Server	291
简介	291
发行版本 3 Portal Server 的升级	292
升级前的任务	292
升级发行版本 3 Portal Server (Solaris)	294
升级发行版本 3 Portal Server (Linux)	295
检验升级	298

升级后的任务	298
升级回滚 (Solaris)	298
多实例升级	300
从 Java ES 发行版本 2 升级 Portal Server	300
简介	300
发行版本 2 Portal Server 的升级	301
升级前的任务	301
升级发行版本 2 Portal Server (Solaris)	303
升级发行版本 2 Portal Server (Linux)	309
检验升级	314
升级后的任务	314
升级回滚	314
多实例升级	314
第 18 章 Portal Server Secure Remote Access	315
Portal Server Secure Remote Access 升级概述	316
关于 Java ES 发行版本 4 Portal Server Secure Remote Access	316
Portal Server Secure Remote Access 升级指导说明	316
Portal Server Secure Remote Access 数据	317
兼容性问题	317
Portal Server Secure Remote Access 组件依赖性	317
从 Java ES 发行版本 3 升级 Portal Server Secure Remote Access	318
简介	318
发行版本 3 Portal Server Secure Remote Access 的升级	319
升级前的任务	319
升级发行版本 3 Portal Server Secure Remote Access (Solaris)	321
升级发行版本 3 Portal Server Secure Remote Access (Linux)	322
检验升级	325
升级后的任务	325
升级回滚 (Solaris)	325
多实例升级	326
从 Java ES 发行版本 2 升级 Portal Server Secure Remote Access	327
简介	327
发行版本 2 Portal Server Secure Remote Access 的升级	328
升级前的任务	328
升级发行版本 2 Portal Server Secure Remote Access (Solaris)	329
升级发行版本 2 Portal Server Secure Remote Access (Linux)	333
检验升级	336
升级后的任务	337
升级回滚	337
多实例升级	337

第 19 章 Delegated Administrator	339
Delegated Administrator 升级概述	340
关于 Java ES 发行版本 4 Delegated Administrator	340
Delegated Administrator 升级指导说明	340
Delegated Administrator 数据	341
兼容性问题	342
Delegated Administrator 组件依赖性	342
从 Java ES 发行版本 3 升级 Delegated Administrator	343
简介	343
发行版本 3 Delegated Administrator 的升级	344
升级前的任务	344
升级发行版本 3 Delegated Administrator (Solaris)	346
升级发行版本 3 Delegated Administrator (Linux)	348
检验升级	350
升级后的任务	350
升级回滚 (Solaris)	350
从 Java ES 发行版本 2 升级 Delegated Administrator	351
升级 Delegated Administrator 依赖项	351
发行版本 2 Delegated Administrator 的升级	352
升级发行版本 2 Delegated Administrator: Web Server Web 容器	352
升级发行版本 2 Delegated Administrator: Application Server Web 容器	352
附录 A Java Enterprise System 发行版本内容	355
Java ES 2003Q4 (发行版本 1)	356
发行版本 1 安装程序可选组件	356
发行版本 1 共享组件	357
Java ES 2004Q2 (发行版本 2)	358
发行版本 2 安装程序可选组件	358
发行版本 2 共享组件	359
Java ES 2005Q1 (发行版本 3)	361
发行版本 3 安装程序可选组件	361
发行版本 3 共享组件	364
Java ES 2005Q4 (发行版本 4)	366
发行版本 4 安装程序可选组件	366
发行版本 4 共享组件	369
索引	371

表格列表

表 1	印刷约定	20
表 2	符号约定	21
表 3	Shell 提示符	21
表 4	Java Enterprise System 文档	22
表 1-1	Java ES 发行版本 4 产品组件	26
表 1-2	Java ES 发行版本 4 共享组件	27
表 1-3	升级过程中的各个阶段	32
表 1-4	升级到 Java ES 2005Q4（发行版本 4）的途径	34
表 1-5	选择性升级与全部升级之比较	36
表 1-6	Java ES 发行版本 4 产品组件的共享组件依赖性	38
表 1-7	Java ES 产品组件依赖性	41
表 2-1	从 Java ES 发行版本 3 升级共享组件的升级技术	51
表 2-2	从 Java ES 发行版本 2 升级共享组件的升级技术	53
表 2-3	用于在 Solaris 平台上升级共享组件的软件包版本	62
表 2-4	用于在 Linux 平台上升级共享组件的软件包	65
表 3-1	升级到 Java ES 发行版本 4 Sun Cluster 3.1 8/05 (2005Q4) 软件的途径	85
表 3-2	Sun Cluster 数据用途	85
表 3-3	Sun Cluster 版本检验输出	87
表 4-1	升级到 Java ES 发行版本 4: Sun Java System Directory Server 5.2 2005Q4 和 Sun Java System Administration Server 5.2 2005Q4 的途径	92
表 4-2	Directory Server、Administration Server 和 Directory Proxy Server 数据用途	94
表 4-3	Directory Server 版本检验输出	96
表 4-4	用于在 Solaris 上升级 Directory Server 和 Administration Server 的修补程序	98
表 4-5	用于在 Linux 上升级 Directory Server 和 Administration Server 的修补程序	102

表 5-1	升级到 Java ES 发行版本 4: Sun Java System Directory Proxy Server 5.2 2005Q4 的途径	112
表 5-2	Directory Proxy Server 数据用途	113
表 5-3	Directory Proxy Server 版本检验输出	116
表 5-4	在 Solaris 上升级 Directory Proxy Server 的修补程序	118
表 5-5	在 Linux 上升级 Directory Proxy Server 的修补程序	119
表 6-1	升级到 Java ES 发行版本 4: Sun Java System Web Server 6.1 SP5 2005Q4 的途径	124
表 6-2	Web Server 数据用途	125
表 6-3	Web Server 版本检验输出	127
表 6-4	在 Solaris 上升级 Web Server 的修补程序	128
表 6-5	在 Linux 上升级 Web Server 的修补程序	129
表 7-1	升级到 Java ES 发行版本 4 Message Queue 3.6 SP3 2005Q4 的途径	135
表 7-2	Message Queue 数据用途	136
表 7-3	Message Queue 版本检验输出	140
表 7-4	升级 Message Queue 的修补程序	141
表 8-1	升级到 Java ES 发行版本 4: HADB 4.4.2 (2005Q4) 的途径	150
表 8-2	HADB 数据用途	151
表 8-3	HADB 版本检验输出	153
表 8-4	用于在 Solaris 平台上升级 HADB 的软件包版本	154
表 9-1	升级到 Java ES 发行版本 4: Sun Java System Application Server Enterprise Edition 8.1 2005Q4 的途径	158
表 9-2	Application Server 数据用途	159
表 9-3	Application Server 版本检验输出	162
表 9-4	在 Solaris 上升级 Application Server 的修补程序	164
表 9-5	在 Linux 上升级 Application Server 的修补程序	165
表 10-1	升级到 Java ES 发行版本 4: Sun Java System 4: Web Proxy Server 4.0.1 2005Q4 的途径	176
表 10-2	Web Proxy Server 数据用途	177
表 10-3	Web Proxy Server 版本检验输出	179
表 11-1	升级到 Java ES 发行版本 4: Sun Java System Access Manager 7 2005Q4 的途径 ..	185
表 11-2	Access Manager 数据用途	186
表 11-3	Access Manager 版本检验输出	190
表 11-4	Access Manager 配置参数: ampre70upgrade	193

表 11-5	用于升级 Access Manager Mobile Access 软件的修补程序	195
表 11-6	Access Manager 配置参数	196
表 11-7	Access Manager 配置参数: amupgrade	198
表 11-8	Access Manager 配置参数	206
表 12-1	升级到 Java ES 发行版本 4: Sun Java System Directory Preparation Tool 6.3 2005Q4 的途径	210
表 12-2	Directory Preparation Tool 数据用途	211
表 12-3	在 Solaris 上升级 Directory Preparation Tool 的修补程序	214
表 12-4	在 Linux 上升级 Directory Preparation Tool 的修补程序	216
表 12-5	在 Solaris 上升级 Directory Preparation Tool 的起源修补程序	219
表 13-1	升级到 Java ES 发行版本 4: Sun Java System Messaging Server 6.2 2005Q4 的途径	224
表 13-2	Messaging Server 数据用途	225
表 13-3	Messaging Server 版本检验输出	228
表 13-4	在 Solaris 上升级 Messaging Server 的修补程序	230
表 13-5	在 Linux 上升级 Messaging Server 的修补程序	232
表 14-1	升级到 Java ES 发行版本 4: Sun Java System Calendar Server 6.2 2005Q4 的途径	242
表 14-2	Calendar Server 数据用途	243
表 14-3	Calendar Server 版本检验输出	246
表 14-4	在 Solaris 上升级 Calendar Server 的修补程序	247
表 14-5	在 Linux 上升级 Calendar Server 的修补程序	249
表 15-1	升级到 Java ES 发行版本 4: Sun Java System Communications Express 6.2 2005Q4 的途径	254
表 15-2	Communications Express 数据用途	255
表 15-3	Communications Express 版本检验输出	258
表 15-4	在 Solaris 上升级 Communications Express 的修补程序	260
表 15-5	在 Linux 上升级 Communications Express 的修补程序	262
表 16-1	升级到 Java ES 发行版本 4: Sun Java System Instant Messaging 7.0.1 2005Q4 的途径	270
表 16-2	Instant Messaging 数据用途	271
表 16-3	Instant Messaging 版本检验输出	273
表 16-4	在 Solaris 上升级 Instant Messaging 的修补程序	275
表 16-5	在 Linux 上升级 Instant Messaging 的修补程序	276
表 17-1	升级到 Java ES 发行版本 4: Sun Java System Portal Server 6.3.1 2005Q4 的途径	288

表 17-2	Portal Server 数据用途	289
表 17-3	Portal Server 版本检验输出	292
表 17-4	用于在 Solaris 上升级 Portal Server 的修补程序	294
表 17-5	用于在 Linux 上升级 Portal Server 的修补程序	296
表 17-6	在 Solaris 上将 Portal Server 升级到发行版本 4 的修补程序	303
表 17-7	在 Linux 上将 Portal Server 升级到发行版本 4 的修补程序	310
表 18-1	升级到 Java ES 发行版本 4: Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4 的途径	316
表 18-2	Portal Server Secure Remote Access 数据用途	317
表 18-3	Portal Server Secure Remote Access 版本检验输出	320
表 18-4	在 Solaris 上升级 Portal Server Secure Remote Access 的修补程序	321
表 18-5	在 Linux 上升级 Portal Server Secure Remote Access 的修补程序	323
表 18-6	在 Solaris 上将 Portal Server Secure Remote Access 升级到发行版本 3 的修补程序 ...	330
表 18-7	在 Linux 上将 Portal Server Secure Remote Access 升级到发行版本 3 的修补程序	333
表 19-1	升级到 Java ES 发行版本 4: Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 的途径	340
表 19-2	Delegated Administrator 数据用途	341
表 19-3	Delegated Administrator 版本检验输出	344
表 19-4	在 Solaris 上升级 Delegated Administrator 的修补程序	346
表 19-5	在 Linux 上升级 Delegated Administrator 的修补程序	348

前言

Java Enterprise System 升级指南包含在 Sun Solaris™ 操作系统 (Solaris™ Operating System, Solaris OS) 或 Linux 操作系统环境中升级 Sun Java™ Enterprise System (Java ES) 软件所需的信息。本指南涵盖了从 Java ES 2004Q2 (发行版本 2) 和 Java ES 2005Q1 (发行版本 3) 到 Java ES 2005Q4 (发行版本 4) 的升级。

本前言包含以下各节:

- 第 20 页中的“目标读者”
- 第 20 页中的“本书所采用的约定”
- 第 22 页中的“相关文档”
- 第 23 页中的“联机访问 Sun 资源”
- 第 23 页中的“联系 Sun 技术支持”
- 第 24 页中的“相关的第三方 Web 站点引用”
- 第 24 页中的“Sun 欢迎您提出意见”

目标读者

本书适用于想要升级 Java ES 软件的系统管理员或软件技术人员。

本书假定您熟悉以下内容：

- 企业级软件产品的安装
- 所处环境中当前部署的 Java ES 组件
- 所支持 Java ES 平台上的系统管理和联网
- 群集模型（如果安装的是群集软件）

本书所采用的约定

本节各表介绍了本书所采用的约定。

印刷约定

下表介绍了本书所采用的印刷约定。

表 1 印刷约定

字体	含义	示例
AaBbCc123 (等宽字体)	API 和语言元素、HTML 标记、Web 站点 URL、命令名、文件名、目录路径名、计算机屏幕输出、样例代码。	编辑 .login 文件。 使用 <code>ls -a</code> 列出所有文件。 % You have mail.
AaBbCc123 (等宽粗体)	用户键入的内容，与计算机屏幕输出的显示不同。	% su Password:
<i>AaBbCc123</i> (斜体)	保留未译的新词或术语以及要强调的词。 命令或路径名中要使用实名或值替换的占位符。	这些称为 <i>class</i> 选项。 文件位于 <i>install-dir</i> /bin 目录。
新词术语强调	新词或术语以及要强调的词。	切勿 保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

符号

下表介绍了本书所采用的符号约定。

表 2 符号约定

符号	说明	示例	含义
[]	包含可选命令选项。	ls [-l]	-l 选项不是必需的。
{ }	包含所需命令选项的一组选择。	-d {y n}	-d 选项要求您使用 y 参数或 n 参数。
-	结合同时发生的多个击键。	Control-A	按 A 键的同时按 Control 键。
+	结合相继发生的多个击键。	Ctrl+A+N	按 Control 键后松开，然后按后续各键。
>	表示图形用户界面中的菜单项选择。	“文件” > “新建” > “模板”	从“文件”菜单中选择“新建”。从“新建”子菜单中选择“模板”。

Shell 提示符

下表介绍了本书所采用的 shell 提示符。

表 3 Shell 提示符

Shell	提示符
UNIX 或 Linux 上的 C shell	<i>machine-name</i> %
UNIX 或 Linux 上的 C shell 超级用户	<i>machine-name</i> #
UNIX 或 Linux 上的 Bourne shell 和 Korn shell	\$
UNIX 或 Linux 上的 Bourne shell 和 Korn shell 超级用户	#
Windows 命令行	C:\

相关文档

通过 <http://docs.sun.com>SM Web 站点可以联机访问 Sun 技术文档。您可以浏览文档库或者搜索具体的书名或主题。

本文档集中的文档

Java ES 手册以可移植文档格式 (Portable Document Format, PDF) 和超文本标记语言 (Hypertext Markup Language, HTML) 格式的联机文件方式提供。行动不便的用户借助辅助技术也可以阅读这两种格式的文件。可从以下网址访问 Sun™ 文档 Web 站点:

<http://docs.sun.com>

Java ES 文档包括系统的整体信息及其各组件的详细信息。可从以下网址访问此文档:

<http://docs.sun.com/prod/entsys.05q4> 及
<http://docs.sun.com/db/prod/entsys.05q4?l=zh>

下表列出了 Java ES 文档集中的系统级手册。左列提供了各文档的名称和文件号码位置, 右列概括介绍了相应文档的内容。

表 4 Java Enterprise System 文档

文档	内容
Java Enterprise System 发行说明 http://docs.sun.com/doc/819-3422	含有有关 Java Enterprise System 的最新信息, 包括已知问题。此外, 各组件还有各自的发行说明。
Java Enterprise System 文档汇总信息 http://docs.sun.com/doc/819-3415	从系统和单个组件的角度对 Java Enterprise System 的所有相关文档进行说明。
Java Enterprise System 技术概述 http://docs.sun.com/doc/819-3584	介绍 Java Enterprise System 的基本技术和概念。描述组件、体系结构、过程和功能。
Java Enterprise System 部署规划指南 http://docs.sun.com/doc/819-3450	简介如何基于 Java Enterprise System 规划和设计企业部署解决方案。介绍部署规划和设计的基本概念及原理, 对解决方案的生命周期进行论述, 并且提供了基于 Java Enterprise System 规划解决方案时使用的高级示例和策略。
Java Enterprise System 安装规划指南 http://docs.sun.com/doc/819-3942	帮助您在硬件、操作系统和网络方面为 Java Enterprise System 的部署制定实现规范。描述要在安装和配置计划中解决的问题, 如组件依赖性。

表 4 Java Enterprise System 文档 (续)

文档	内容
Java Enterprise System 安装指南 (适用于 UNIX) http://docs.sun.com/doc/819-3315	指导您完成在 Solaris™ 操作系统或 Linux 操作系统上安装 Java Enterprise System 的整个过程。还说明了如何在安装后配置组件以及检验它们是否正常工作。
Java Enterprise System 安装参考 http://docs.sun.com/doc/819-3949	给出有关配置参数的附加信息, 提供要在配置规划中使用的工作表, 并且列出了参考资料, 如默认目录和端口号。
Java Enterprise System 部署示例 系列: 评估方案 http://docs.sun.com/doc/819-3443	描述如何在一个系统上安装 Java Enterprise System、建立一组核心共享联网服务以及如何设置可以访问所建立服务的用户帐户。
Java Enterprise System 升级指南 http://docs.sun.com/doc/819-3457	说明如何在 Solaris™ 操作系统或 Linux 操作系统上升级 Java Enterprise System。
Java Enterprise System 术语表 http://docs.sun.com/doc/819-4631	定义 Java Enterprise System 文档中使用的术语。

联机访问 Sun 资源

有关产品下载、专业服务、修补程序和支持以及其他开发者信息, 请访问以下网址:

- 下载中心
<http://www.sun.com/software/download/>
- 客户解决方案
<http://www.sun.com/service/sunjavasystem/sjssservicessuite.html>
- Sun 企业服务、Solaris 修补程序及支持
<http://sunsolve.sun.com/>
- 开发者信息
<http://developers.sun.com>

以下位置提供了有关 Java ES 及其组件的信息:

<http://www.sun.com/software/javaenterprisesystem/index.html>

联系 Sun 技术支持

如果您遇到通过本文档无法解决的技术问题, 请访问
<http://www.sun.com/service/contacting>。

相关的第三方 Web 站点引用

Sun 对本文档中提到的第三方 Web 站点的可用性不承担任何责任。对于此类站点或资源中的（或通过它们获得的）任何内容、广告、产品或其他材料，Sun 并不表示认可，也不承担任何责任。对于因使用或依靠此类站点或资源中的（或通过它们获得的）任何内容、产品或服务而造成的或连带产生的实际或名义损坏或损失，Sun 概不负责，也不承担任何责任。

Sun 欢迎您提出意见

Sun 致力于提高其文档的质量，并十分乐意收到您的意见和建议。

要共享您的意见，请访问 <http://docs.sun.com>，然后单击“发送意见” (Send Comments)。请在联机表单中提供文档标题和文件号码。文件号码包含七位或九位数字，可在书的标题页或在文档顶部找到该号码。提出意见时您还需要在表格中输入文件的英文文件号码和标题。本文件的英文文件号码是 819-2331-13，文件标题为《Sun Java Enterprise System 2005Q4 Upgrade Guide》。

升级规划

本章提供的信息可用于规划 Sun Java™ Enterprise System (Java ES) 软件到 Java ES 2005Q4（发行版本 4）的升级。其中包括以下各节：

- 第 26 页中的“Java ES 2005Q4（发行版本 4）组件”
- 第 29 页中的“关于 Java ES 升级”
- 第 32 页中的“升级规划”
- 第 37 页中的“Java ES 组件依赖性”
- 第 43 页中的“一般先后顺序指导原则”

Java ES 2005Q4（发行版本 4）组件

本节是对 Java ES 软件升级规划的简介，其中将对 Java ES 发行版本 4 中所包括的组件进行评述。视具体的升级方案，可能需要将其中的一个或多个组件升级到其在发行版本 4 中的版本。

Java ES 组件分成了不同类型的组，如 Java Enterprise System 技术概述 (<http://docs.sun.com/doc/819-3584>) 中所述。相应地，系统服务组件提供主要的 Java ES 基础结构服务，而服务质量组件用于增强这些系统服务。本文将这两种类型的 Java ES 组件统称为产品组件，可在 Java ES 安装程序内选择这些组件。

每个产品组件均依赖于一个或多个称为 Java ES 共享组件的本地共享库。共享组件在产品组件安装期间由 Java ES 安装程序自动进行安装，具体安装哪些共享组件视所安装的产品组件而定。

发行版本 4 产品组件

下表按字母顺序列出了 Java ES 发行版本 4 产品组件。对于其中的服务质量组件，该表包括它们所提供的服务增强类型。

表 1-1 Java ES 发行版本 4 产品组件

产品组件	版本	类型	简称
Access Manager	7.0	系统服务组件	AM
Administration Server	5.2	服务质量：管理组件	ADS
Application Server	8.1	系统服务组件	AS
Calendar Server	6.2	系统服务组件	CS
Communications Express	6.2	服务质量：访问组件	CX
Delegated Administrator	6.3	服务质量：管理组件	DA
Directory Preparation Tool	6.3	服务质量：管理组件	DPT
Directory Proxy Server	5.2	服务质量：访问组件	DPS
Directory Server	5.2	系统服务组件	DS
High Availability Session Store	4.4.2	服务质量：可用性组件	HADB
Instant Messaging	7.0.1	系统服务组件	IM
Message Queue	3.6 SP3	系统服务组件	MQ
Messaging Server	6.2	系统服务组件	MS
Portal Server	6.3	系统服务组件	PS

表 1-1 Java ES 发行版本 4 产品组件（续）

产品组件	版本	类型	简称
Portal Server Secure Remote Access	6.3	服务质量：访问组件	PSRA
Service Registry	3.0	系统服务组件	SR
Sun Cluster	3.1 8/05	服务质量：可用性组件	SC
Web Proxy Server	4.0.1	服务质量：访问组件	WPS
Web Server	6.1 SP%	系统服务组件	WS

发行版本 4 共享组件

无法在 Java ES 安装程序中选择或取消选择单台计算机上安装的产品组件所依赖的 Java ES 共享组件。安装 Java ES 产品组件时，Java ES 安装程序会自动安装已安装的产品组件所需的共享组件。

下表列出了 Java ES 发行版本 4 共享组件。

表 1-2 Java ES 发行版本 4 共享组件

共享组件	版本	缩写
Apache Commons Logging	1.0.3	ACL
Jakarta ANT Java/XML-based build tool	1.6.2	ANT
Berkeley Database	4.2.52	BDB
Common agent container	1.1	CAC
International Components for Unicode	3.2	ICU
Instant Messenger SDK	6.2.8	IM-SDK
Java 2 Platform, Standard Edition	5.0 Update 3	J2SE™
JavaBeans™ Activation Framework	1.0.3	JAF
Java Studio Enterprise Web Application Framework	2.1.5	JATO
JavaHelp™ Runtime	2.0	JHELP
JavaMail™ Runtime	1.3.2	JMAIL
Java Architecture for XML Binding Runtime	1.0.4	JAXB
Java API for XML Processing	1.2.6	JAXP
Java API for XML Registries Runtime	1.0.7	JAXR

表 1-2 Java ES 发行版本 4 共享组件 (续)

共享组件	版本	缩写
Java API for XML-based Remote Procedure Call Runtime	1.1.2	JAX-RPC
Java Calendar API	1.2	JCAPI
Java Dynamic Management™ Kit Runtime	5.1	JDMK
Java Security Services	4.1	JSS
KT Search Engine	1.3.2	KTSE
LDAP C SDK	5.11	LDAP C SDK
LDAP Java SDK	4.18	LDAP J SDK
Mobile Access Core	1.0.6	MA Core
Netscape Portable Runtime	4.5.2	NSPR
Network Security Services	3.10	NSS
SOAP Runtime with Attachments API for Java	1.2.1	SAAJ
Simple Authentication and Security Layer	2.18	SASL
Sun Explorer Data Collector	4.3.1	SEDC
Sun Java Enterprise System Monitoring Framework	1.0.1	MFWK
Sun Java Web Console	2.2.4	SJWC
Web services Common Library	1.0	WSCL

关于 Java ES 升级

通常不使用 Java ES 安装程序或其他任何系统实用程序执行 Java ES 软件到发行版本 4 的升级，而是使用特定于组件的升级过程按组件、计算机逐个执行。

组件升级可从主要升级变动为完全兼容升级，前者可能与组件的先前版本不兼容，后者只提供错误修复。鉴于 Java ES 组件之间的依赖性，升级的性质可能会影响是否还需要升级其他组件。

产品组件升级

Java ES 产品组件升级包括两个基本操作，即镜像 Java ES 产品组件的最初安装和配置：

- **安装已升级的软件。** 新软件可以增强或修复现有软件，也可替换现有软件。通常，新软件是通过以下方法实现的：对现有软件包应用修补程序，替换现有软件包，安装新软件包，或使用 Java ES 安装程序完全重新安装组件。
- **重新配置。** 重新配置包括为支持已升级软件而需要对配置数据、用户数据或动态应用程序数据所做的任何更改。数据更改可以指附加数据、数据格式更改（无论是在属性文件中还是在数据库模式中），也可以指数据位置更改。重新配置有时要求您执行明确的过程，有时无需您介入而自动进行。

在本升级指南中，对于每个 Java ES 产品组件均对组件升级的这两个方面进行了描述。

升级指南还涵盖了产品组件升级的其他重要方面，包括：

- 影响升级的依赖性
- 升级组件前可能需要执行的操作
- 为检验升级是否成功而要执行的操作
- 需要回滚升级时所执行的操作

共享组件升级

Java ES 共享组件升级通常是升级依赖于这些共享组件的产品组件不可或缺的组成部分。

共享组件的升级通常比产品组件的升级更为直接。一般而言，升级是通过向现有软件包应用修补程序或替换现有软件包来实现的。与升级产品组件相比，通常无需进行任何重新配置，也不需要执行升级前或升级后过程。

尽管可以逐个升级共享组件，然而 Java ES 发行版本 4 允许您在一次操作中共同升级多个共享组件。有关更多信息，参见第 2 章“升级 Java ES 共享组件”。

升级技术

如本升级指南中所述，产品组件和共享组件的升级包括修改或替换当前安装的软件包，在某些情况下，还包括安装新的软件包。Solaris 和 Linux 平台运用类似的技术，通过软件包注册表来管理已安装的软件包和跟踪更改。

- **Solaris 平台。** 可以使用 Java ES 软件分发时所带的软件包，通过 Solaris 的 `pkgadd` 和 `pkgrm` 命令安装和删除 Java ES 软件包。一经安装，便可使用通过 `patchadd` 和 `patchrm` 命令应用或删除的修补程序来修改软件包内容。Solaris 软件包的修补程序通过 SunSolve Web 站点进行分发，其网址为：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

Solaris 修补程序可以修补一个或多个软件包。`patchadd` 命令会保存所修补软件包的备份，以方便使用 `patchrm` 命令删除修补程序。修补程序通过修补程序 ID 进行标识，该 ID 包括一个修补程序号，其后紧跟着修订版本号，该版本号会随着修补程序的不断修改而递增。

还可将 Solaris 修补程序集中到一个修补程序包中。修补程序包允许您一同下载和应用包中的所有修补程序。修补程序包专为升级 Java ES 共享组件而提供（参见第 2 章“升级 Java ES 共享组件”）。

- **Linux 平台。** 可以使用 Java ES 软件分发时所带的软件包，通过 `rpm` 命令来安装或更新 Java ES RPM (Red Hat Package Manager) 软件包。但是，一经安装，便无法使用修补程序修改软件包内容。不过，可使用 `rpm -U` 命令选项更新 RPM 软件包，该选项会用较新的软件包替换当前软件包。

为方便起见，许多 RPM 软件包升级内容不但会在 Java ES 软件分发时进行分发，而且还会通过 SunSolve Web 站点进行分发，其网址为：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

在通过 SunSolve 分发时，会将 RPM 软件包包装为修补程序，并为其分配一个与 Solaris 修补程序类似的修补程序 ID 和修订版本号。这些 Linux 修补程序可包括一个或多个 RPM 软件包，每一个均通过唯一的 RPM 名称、RPM 编号和修订版本号加以标识，其中，修订版本号会随着 RPM 软件包的不断修改而递增。

操作系统问题

如下所述，有许多操作系统问题会对 Java ES 软件升级产生影响。

所需的操作系统修补程序

某些情况下，要想成功升级 Java ES 产品组件，必须先修补操作系统或应用特定的修复。通常，最好是在执行 Java ES 升级之前，直接将操作系统更新为最新版本，而不是针对每一种情况应用所需的特定操作系统修补程序。

- Solaris 平台修补程序是通过 SunSolve Web 站点以修补程序包的形式提供的，修补程序包是可以一同应用的操作系统修补程序集。可在以下网址获得 Solaris 8、9 和 10 的操作系统修补程序包：
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>
- Linux 平台最新版本可在以下网址获得：
<https://www.redhat.com/apps/download/>

次发行版本升级

大量 Java ES 共享组件都有特定于 Solaris 发行版本的软件包。这些特定于发行版本的软件包可能无法在其他 Solaris 平台上正常工作。例如，不能期望针对 Solaris 8 操作系统发行的软件包对于 Solaris 9 或 Solaris 10 操作系统也同样适用。

在将操作系统从一个次发行版本升级到另一个次发行版本时，各种已安装的 Java ES 共享组件都会受到影响。当共享组件具有发行版本特定的软件包时，在升级了操作系统之后，还需要对这些软件包进行升级以匹配新升级的操作系统。

非支持平台的升级

Solaris 8 和 9 操作系统以及 Red Hat Enterprise Linux (RHEL) 2.1 都支持 Java ES 2004Q2（发行版本 2）。如果希望将操作系统平台升级到 Solaris 10 或 RHEL 3.0（这些平台不受 Java ES 发行版本 2 支持），则还需要将 Java ES 发行版本 2 升级到支持已升级平台的 Java ES 发行版本，最好是升级到 Java ES 发行版本 4。

由于某些 Java ES 组件的升级要求其他 Java ES 组件处于运行状态，因此，一般来说，在从发行版本 2 升级 Java ES 之前，不能先将操作系统平台升级到 Solaris 10 或 RHEL 3.0（Java ES 发行版本 2 不支持这些平台）。

而应视具体的平台采取所需的方法：

- **Linux 平台。** 应先将 Java ES 发行版本 2 升级到发行版本 4，然后再升级到 RHEL 3.0。
- **Solaris 平台。** 应卸载 Java ES 发行版本 2，将操作系统升级到 Solaris 10，然后执行 Java ES 发行版本 4 的全新安装。此类操作意味着所有 Java ES 组件都必须进行全新配置。在此情况下，比较明智的做法是备份所有 Java ES 发行版本 2 配置文件和自定义内容，以便在配置 Java ES 发行版本 4 组件时使用。

升级规划

在将已部署的 Java ES 软件系统升级到 Java ES 发行版本 4 时，所采取的方法不但取决于升级目的和优先级，而且还取决于部署体系结构的范围和复杂性。

例如，Java ES 部署体系结构可能由单台计算机上运行的单个 Java ES 组件组成，升级目的是为了修复上一软件发行版本中的错误。反之，Java ES 部署体系结构也可能由部署于多台不同计算机上的多个相互依赖的 Java ES 组件组成，而升级目的是要获得一些新的功能并且要以最短的停机时间达到此目的，为此要将所需升级的组件数目降至最低。

这两个例子所代表的升级情况的复杂性相差很大，因而需要截然不同的升级计划。没有任何一个计划对于所有已部署的 Java ES 软件系统都适用。

通常，Java ES 组件数量以及部署体系结构中的计算机数量越多，升级计划就越复杂。

什么是升级计划？

升级计划规定了如何处理升级过程的每个阶段。此过程至少包括下表所示的各个阶段。

表 1-3 升级过程中的各个阶段

升级阶段	说明
准备	制定升级计划。在其中，指定需要升级的 Java ES 组件以及在系统中的各种计算机上升级这些组件所需的顺序。还要规划在生产环境中执行升级过程之前，如何在阶段性环境中对升级过程进行测试。
执行	获取升级所需的所有必备软件包、修补程序和工具。在阶段性环境中执行 Java ES 部署系统的升级和重新配置。这包括备份配置和应用程序数据、升级系统软件以及重新配置数据或将数据迁移到已升级系统。

表 1-3 升级过程中的各个阶段（续）

升级阶段	说明
检验	启动已升级的软件组件并在进行过程中执行检验测试。如果检验不成功，且在合理的时限内无法解决问题，则可能只得回滚升级并将系统恢复到其先前的状态。
回滚/恢复	必要时，可将系统恢复到其在准备阶段中所规定的先前状态。另外，也要执行测试以检验回滚是否成功。

以下各节所提供的信息有助于制定出详细的升级计划。

升级计划注意事项

除了部署体系结构的范围和复杂性以外，升级计划还取决于诸多因素。这些因素含有以下注意事项：

- 升级途径
- 已部署 Java ES 组件间的依赖性
- 执行选择性升级的可能性
- 多实例升级

以上因素将在以下各节中进行讨论。

升级途径

虽然可以将所有先前发行版本的 Java ES 软件升级到 Java ES 2005Q4（发行版本 4），但是只有从 Java ES 2005Q1（发行版本 3）和 Java ES 2004Q2（发行版本 2）的升级经过了证实。本升级指南中未记述从较早发行版本进行的升级。

如第 34 页中的表 1-4 中所述，各种升级途径涉及不同的升级策略。

鉴于发行版本 3 到发行版本 4 升级途径和发行版本 2 到发行版本 4 升级途径的不同特点，并且产品组件的升级过程常常取决于升级途径，所以，在本升级指南中，将分别描述每个产品组件升级的各章分成了两节：一节介绍从发行版本 3 到发行版本 4 的升级，另一节介绍从发行版本 2 到发行版本 4 的升级。

表 1-4 升级到 Java ES 2005Q4（发行版本 4）的途径

产品号	Java ES 发行版本	系统特征	升级策略
2005Q1	发行版本 3	Java ES 发行版本 4 支持在单台计算机上混合使用发行版本 3 和发行版本 4 组件。这包括产品组件和共享组件。发行版本 3 与发行版本 4 组件之间的兼容性已经过测试，并且在 Java Enterprise System 发行说明 (http://docs.sun.com/doc/819-3422) 中指出了所有已知的不兼容性。	发行版本 3 与发行版本 4 组件的共存使得您可能在给定计算机上或在由多台计算机组成的部署体系结构中，有选择地将发行版本 3 组件升级到发行版本 4。
2004Q2	发行版本 2	Java ES 发行版本 4 不支持在单台计算机上混合使用发行版本 2 和发行版本 4 组件。这包括产品组件和共享组件。在这些发行版本之间存在已知的不兼容性，而且发行版本 2 与发行版本 4 组件间的互操作性未经证实。	在给定计算机上将组件从发行版本 2 升级到发行版本 4 时，应将所有发行版本 2 组件都升级到发行版本 4。不过，假如组件兼容，便有可能在由多台计算机组成的部署体系结构内混合使用驻留在不同计算机上的发行版本 2 和发行版本 4 组件。
2003Q4 及以前	发行版本 1 及以前	Java ES 发行版本 4 不支持在单台计算机上混合使用发行版本 1 或以前发行版本的组件和发行版本 4 组件。这包括产品组件和共享组件。这些发行版本之间存在已知的不兼容性，而且发行版本 1 或以前的组件与发行版本 4 组件间的互操作性未经证实。	<p>Java ES 发行版本 4 并未证实发行版本 1 或以前发行版本到发行版本 4 的直接升级。</p> <p>不过，在某些情况下，可以通过先升级到 Java ES 发行版本 3 来执行从发行版本 1 的升级，如发行版本 3 的 Java Enterprise System 升级与迁移指南 (http://docs.sun.com/doc/819-2237) 中所述。</p> <p>在其他情况下，从发行版本 1 到发行版本 4 的升级可以采用与从发行版本 2 或发行版本 3 升级到发行版本 4 相同的方式执行，而且在这些情况下，本升级指南中对应于该组件的升级指导说明指出了这种可能性。</p>

注

有些产品组件已发布了过渡发行版本，这些版本介于正式的 Java ES 发行版本之间。这种情况下，应按照上一 Java ES 发行版本的相同过程来执行过渡发行版本的升级。例如，如果过渡发行版本出现于发行版本 2 和发行版本 3 之间，则要按照从发行版本 2 升级到发行版本 4 的过程来升级组件。

升级依赖性

规划任何给定 Java ES 组件升级的主要问题之一是了解组件对其他 Java ES 组件的依赖性，以及是否还需将这些其他组件升级以支持依赖组件的升级。

在这方面存在两种类型的升级依赖性：

- **硬性升级依赖性。** 硬性升级依赖性是指组件的升级版本要求它所依赖的一些组件的升级版本。这种要求可能归因于新功能、新接口或依赖组件所需的错误修复。不先升级组件所依赖的组件，则无法成功升级和使用该组件。
- **软性升级依赖性。** 软性升级依赖性是指组件的升级版本不要求它所依赖的一些组件的升级版本。不升级组件所依赖的组件，也可以成功升级和使用该组件。

在升级 Java ES 组件时，必须升级它对其具有硬性升级依赖性的所有组件，但是可以不升级它对其具有软性升级依赖性的那些组件。（这条一般原则不适用于单台计算机上从发行版本 2 到发行版本 4 的升级。）

不过，当升级涉及多个互相依赖的组件时，这条一般原则不一定适用。在此情况下，只要其他多个 Java ES 组件中有一个对某个特定组件具有硬性升级依赖性，就必须升级该组件。

选择性升级或全部升级

硬性升级依赖性与软性升级依赖性的差异使得在部署系统内有选择地升级 Java ES 组件成为可能。不过，仅对于单台计算机上从发行版本 3 到发行版本 4 的升级才有这种可能（参见第 33 页中的“升级途径”中的升级途径特征）。不支持单台计算机上从发行版本 2 到发行版本 4 的选择性升级。

- **选择性升级。** 可从一开始就对希望升级到发行版本 4 的 Java ES 组件采用选择性升级方法。应确定该组件的硬性升级依赖性，其中包括对产品组件和共享组件的依赖性。还需要对这些组件进行升级。对每个后续的硬性升级依赖性重复此过程，直至再无其他组件需要升级为止。这样一来便指定了所有需要升级的 Java ES 组件。

选择性升级方法可能很简单，也可能相当复杂，这要视部署体系结构和所涉及的硬性升级依赖性而定。

- **全部升级。** 另外，也可以将所有已部署的 Java ES 组件全都升级到发行版本 4。此方法的复杂性也取决于部署体系结构。在某些情况下，出于业务原因，一次对整个系统进行升级是行不通的。

下表对这两种用于执行升级的方法进行了比较。

表 1-5 选择性升级与全部升级之比较

升级方法	优点	缺点
选择性升级	所要升级的组件数目最少	必须跟踪部署系统中每个组件的版本
全部升级	已部署系统中所有组件的版本都一致	所要升级的组件数目最多

在选择性升级和全部升级之间做出选择并不是一成不变的。例如，您可能选择对特定计算机上的产品组件进行选择性地升级，而希望升级支持所选产品组件所需的所有共享组件。事实上，对于从发行版本 3 到发行版本 4 的升级，在升级所有相应共享组件的同时有选择地升级产品组件常常是首选方法。

多实例升级

升级过程的顺序可能取决于是否以及如何部署体系结构中使用冗余。可以使用 Java ES 组件的多个实例来实现高可用性、可伸缩性、可维护性或这些服务质量的某种组合。有三种在 Java ES 部署体系结构中利用冗余组件的技术：负载平衡、高可用性技术（Sun Cluster 和 High Availability Session Store）以及多主复制（Directory Server）。

在大多数运用冗余的情况下，都是希望在不引起停机的情况下执行升级。进行滚动升级，目的是为了陆续升级组件的冗余实例而不累及它们所提供的服务。

大多数情况下，冗余实例部署在多台计算机上。从升级规划角度来看，这可能意味着要将此类复制组件的升级与其他组件的升级隔离开来，才能最大程度地缩短停机时间。换言之，可能要在执行复制组件的滚动升级之前，先在每台计算机上执行该组件的所有升级前任务。

每种复制技术均有可能影响 Java ES 组件总体升级顺序的配置或重新配置过程。例如，在 Sun Cluster 环境中运行的组件可能要求在升级 Sun Cluster 环境中运行的组件之前，先对 Sun Cluster 进行升级。

Java ES 组件依赖性

如上节所提到的那样，升级计划指定了需要升级的 Java ES 组件，以及升级这些组件所需依照的顺序。升级计划中的一个重要注意事项便是所部署系统中不同 Java ES 组件之间的依赖性。

无论采取选择性升级方法还是升级所有组件，执行组件升级的顺序均要受到组件间依赖性的影响。

本节提供了有关 Java ES 组件依赖性的信息。以下依赖性因素将会对升级计划产生影响。

- 对共享组件的依赖性
- 对产品组件的依赖性
- 多实例升级

以下各节将对上述每个因素进行简要讨论。

对共享组件的依赖性

升级 Java ES 产品组件时，必须考虑这些 Java ES 组件对 Java ES 共享组件的依赖性。当产品组件对共享组件具有硬性升级依赖性时，还必须对共享组件进行升级。

共享组件依赖性矩阵

第 38 页中的表 1-6 显示了 Java ES 2005Q4（发行版本 4）产品组件对 Java ES 共享组件的依赖性。作为表 1-6 列标题的产品组件缩写取自第 26 页中的表 1-1。共享组件的缩写详见第 27 页中的表 1-2。

有四个产品组件未包括在表 1-6 中，它们是：Directory Proxy Server (DPS)、High Availability Session Store (HADB)、Directory Preparation Tool (DPT) 和 Service Registry (SR)。前三个被省略是因为它们对共享组件没有依赖性。第四个被省略是因为它是一个新的产品组件，没有可从中进行升级的先前版本。Web Proxy Server (WPS)，另一个新的发行版本 4 产品组件，亦未包括在表 1-6 中，因为其可以从中升级到发行版本 4 的先前发行版本未包括在 Java ES 中。

在表 1-6 的矩阵中，发行版本 3 到发行版本 4 升级的硬性升级依赖性标记为“H”，而软件升级依赖性标记为“S”。对于发行版本 2 到发行版本 4 升级，按照定义，所有共享组件依赖性均为硬性升级依赖性，必须将所有共享组件从发行版本 2 升级到发行版本 4。

表 1-6 Java ES 发行版本 4 产品组件的共享组件依赖性

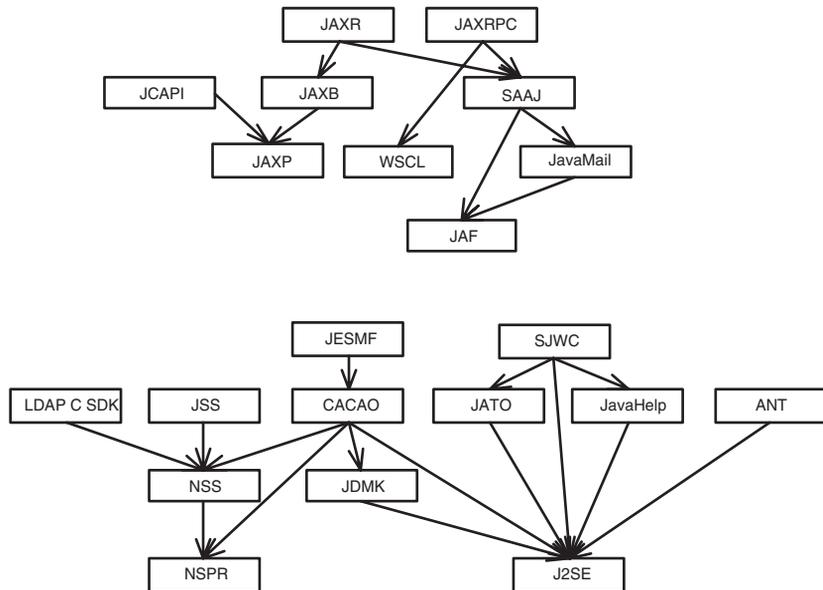
共享组件	AM	ADS	AS	CS	CX	DA	DPS	DS	IM	MQ	MS	PS	PSRA	SC	WPS	WS
ANT			S													
ACL	S															
BDB	S															
CAC									S					S		
ICU		S	S	S			S	S			H	S			S	S
IM-SDK									H			S				
J2SE™	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
JAF	S		S		S				S	S		S	S			
JATO	S		S		S	S						S				
JavaHelp™	S		S							S						
JavaMail™	S		S		S				S	S		S	S			
JAXB	S		S													
JAXP	S		S		S				S	S		S	S			
JAXR	S		S													
JAX-RPC	S		S													
JCAPI					S				S							
JDMK			S						S					S		
JSS	S	S		S		S	S	S		S		S	S		S	S
KTSE												S			S	S
LDAP C SDK		S		S			S	S			H				S	S
LDAP J SDK	S	S			S	S	S	S								
MA Core	S											H	H			
MFWK									S							
NSPR	S	S	S	S		S	S	S	S	S	H	S	S	S	S	H
NSS	S	S	S	S		S	S	S	S	S	H	S	S	S	S	H
SAAJ	S		S							S		S	S			

表 1-6 Java ES 发行版本 4 产品组件的共享组件依赖性 (续)

共享组件	AM	ADS	AS	CS	CX	DA	DPS	DS	IM	MQ	MS	PS	PSRA	SC	WPS	WS
SASL		S		H	H		S	S			H				S	S
SEDC														S		
SJWC	S		S											S		
WSCL	S		S													

表 1-6 中所示的任何产品组件的依赖性，既表示直接共享组件依赖性又表示间接共享组件依赖性。换言之，某个产品组件可能依赖于某个特定的共享组件，而该共享组件又依赖于一个或多个其他共享组件。表 1-6 中所示的共享组件依赖性包括所有此类间接依赖性。下图说明了共享组件之间的相互依赖性。

图 1-1 共享组件相互依赖性



共享组件升级指导原则

根据表 1-6，可以在给定计算机上升级一个或多个产品组件时确定要升级的共享组件：

- **发行版本 2 到发行版本 4 的升级。** 如果要从发行版本 2 升级到发行版本 4，必须对表 1-6 中各产品组件所有标记为“S”或“H”的共享组件进行升级。
- **发行版本 3 到发行版本 4 的升级。** 如果要将所有产品组件从发行版本 3 升级到发行版本 4，应对表 1-6 中所示各产品组件的所有共享组件进行升级。

不过，即使是有选择地升级产品组件，亦建议对计算机上所有产品组件所需的共享组件都进行升级；已证实发行版本 4 共享组件可以支持发行版本 3 产品组件。

尽管在大多数情况下都可以有选择地升级共享组件（即，仅升级支持选择性升级产品组件的那些共享组件，或者，与软性升级依赖性相比较，仅升级硬性升级依赖项），但值得注意的是，采用这种方法会冒更大的风险。

如果不涉及任何硬性升级依赖性，则您可能根本不会升级共享组件。但是，一般而言，较好的做法是将底层的 Java ES 基础共享组件升级到最新版本。

注 升级共享组件的顺序可能取决于图 1-1 中所示的共享组件相互依赖性。

另外，如果计划将 J2SE 升级到 J2SE 5.0，应首先升级此共享组件。J2SE 是许多 Java ES 组件的基础组件。

有关如何升级共享组件的信息，参阅第 2 章“升级 Java ES 共享组件”。

对产品组件的依赖性

一个产品组件对另一个产品组件的依赖性确定需要升级的 Java ES 组件及其升级顺序的重要因素。对产品组件的依赖性分为两大类：运行时依赖性和配置依赖性。

- **运行时依赖性。** 软件系统部署组件之间的交互是软件系统正常工作的基础。Java ES 组件之间的基础结构依赖性在 Java Enterprise System 技术概述中进行了讨论。在升级任何 Java ES 产品组件时，均必须考虑此类依赖性。如果一个组件的升级版本对另一个组件具有硬性升级依赖性，则该依赖性意味着依赖组件只应在其所依赖的组件升级之后方可进行升级。

- **配置依赖性。** 许多情况下，必须安装、配置某个 Java ES 组件并使其处于运行状态，方可配置另一组件。例如，Directory Server 配置目录必须正在运行，方可配置 Messaging Server 组件，或者，Directory Server 用户/组目录必须正在运行，方可注册 Access Manager 服务。组件升级过程通常包括已升级组件的重新配置或配置数据的迁移。事实上，一些产品组件的主要功能就是为了向其他组件提供配置或管理支持。因此，配置依赖性可能对升级过程的顺序具有很大影响。

表 1-7 显示了第 26 页中的表 1-1 中所列 Java ES 产品组件之间的依赖性。利用表 1-7，可以图表形式表示升级集中的依赖性链。左列列出了每个产品组件，中间一列列出了其对其他产品组件的依赖性，第三列描述了每种依赖性的特点，最后一列指明了各组件是否必须在本地。

表 1-7 Java ES 产品组件依赖性

产品组件	依赖性	依赖性本质	必须在本地吗？
Access Manager	Directory Server	存储配置数据和启用用户数据查找	否
	J2EE Web 容器，可为以下产品之一： - Application Server - Web Server - BEA WebLogic Server - IBM WebSphere Application Server	提供 Web 容器运行时服务	是
Access Manager SDK	Access Manager	提供 Access Manager 服务	否
	J2EE Web 容器，可为以下产品之一： - Application Server - Web Server - BEA WebLogic Server - IBM WebSphere Application Server	提供 Web 容器运行时服务	是
Administration Server	Directory Server	提供配置目录	否
Application Server	Message Queue	提供可靠的异步消息传送	是
	Web Server（可选）	专用于实例间的负载平衡	是
	High Availability Session Store（可选）	存储支持实例间故障转移所需的会话状态	是
Calendar Server	Directory Server	存储用户数据和启用用户数据查找	否
	Directory Preparation Tool	准备 Calendar Server 所用的目录	否
	Access Manager（可选）	提供单点登录	否
	Messaging Server（可选）	提供电子邮件通知	否
	Delegated Administrator（可选）	为日历服务置备用户	否

表 1-7 Java ES 产品组件依赖性 (续)

产品组件	依赖性	依赖性本质	必须在本地吗?
Communications Express	J2EE Web 容器, 可为以下产品之一: - Application Server - Web Server	提供 Web 容器运行时服务	是
	Directory Server	存储用户数据和启用用户数据查找, 例如在通讯录中	否
	Directory Preparation Tool	准备 Communications Express 所用的目录	否
	Access Manager 或 Access Manager SDK	提供验证和授权服务, 单点登录	是
	Messaging Server	启用基于 Web 的消息传送访问	否
	Calendar Server	启用基于 Web 的日历访问	否
Delegated Administrator	J2EE Web 容器, 可为以下产品之一: - Application Server - Web Server	提供 Web 容器运行时服务	是
	Directory Server	存储用户数据	否
	Directory Preparation Tool	准备 Delegated Administrator 所用的目录	否
	Access Manager 或 Access Manager SDK	提供置备用户所需的 API	是
Directory Preparation Tool	Directory Server	提供准备供通信组件使用的用户/组目录	是
Directory Proxy Server	Administration Server	配置 Directory Proxy Server	否
	Directory Server	提供对目录的访问	否
Directory Server	Administration Server	配置 Directory Server	否
High Availability Session Store	无		
Instant Messaging	J2EE Web 容器, 可为以下产品之一: - Application Server - Web Server	提供 Web 容器运行时服务	是
	Directory Server	存储用户数据	否
	Access Manager (可选)	提供单点登录	否

表 1-7 Java ES 产品组件依赖性 (续)

产品组件	依赖性	依赖性本质	必须在本地吗?
Message Queue	无		
Messaging Server	Directory Server	存储配置数据和启用户数据查找	否
Store MTA MMP MEM	Administration Server	在 Directory Server 配置目录中存储配置数据	是
	Directory Preparation Tool	准备 Messaging Server 所用的目录	否
	Access Manager (可选)	提供单点登录	否
	Delegated Administrator (可选)	为消息传送服务置备用户	否
Portal Server	J2EE Web 容器, 可为以下产品之一: - Application Server - Web Server - BEA WebLogic Server - IBM WebSphere Application Server	提供 Web 容器运行时服务	是
	Directory Server	存储用户配置文件和启用户配置文件查找	否
	Access Manager 或 Access Manager SDK	提供验证和授权服务, 单点登录	是
	Communications Express	提供消息传送和日历频道	否
Portal Server Secure Remote Access	Portal Server	提供对门户的访问	是
	Access Manager 或 Access Manager SDK	提供验证和授权服务, 单点登录	是
Sun Cluster	无		
Sun Cluster 代理	Sun Cluster	提供对 Sun Cluster 服务的访问	
Web Proxy Server	无		
Web Server	无		

一般先后顺序指导原则

前几节中所讨论的因素可能都会对计划升级的 Java ES 组件及其升级顺序产生影响。这些因素还会影响部署于多台计算机上的 Java ES 组件的升级方法。所有这些因素的具体影响取决于部署体系结构。

然而, 可以遵循几条一般的先后顺序指导原则, 尽管它们并不适用于每一种情况。下面列出了在单台计算机上或在已部署系统中成功升级 Java ES 组件可采用的顺序。执行升级时, 只需忽略部署体系结构中不包括的那些组件即可, 或者, 如果执行的是选择性升级, 则忽略升级计划中不包括的那些组件。

注 本升级指南中的各章按照通常升级 Java ES 组件的顺序排列，如以下先后顺序指导原则所示。

- 1. 共享组件**（参见第 47 页中的第 2 章“升级 Java ES 共享组件”）
通常应先升级共享组件，然后再对依赖于它们的组件进行升级。
- 2. Sun Cluster 软件**（参见第 83 页中的第 3 章“Sun Cluster 软件”）
如果有任何组件运行于 Sun Cluster 环境，且 Sun Cluster 软件需要升级，则应先于使用 Sun Cluster 服务的组件对其进行升级。如果进行升级，应在 Sun Cluster 升级过程中升级 Sun Cluster 代理。
- 3. Directory Server 和 Administration Server**（参见第 91 页中的第 4 章“Directory Server 和 Administration Server”）
许多组件都在 Directory Server 中存储用户数据或配置数据，因此通常应先对 Directory Server 执行升级，然后再升级 Directory Server 对其具有运行时或配置依赖性的组件。Administration Server 必须与 Directory Server 一同进行升级。
- 4. Directory Proxy Server**（参见第 111 页中的第 5 章“Directory Proxy Server”）
Directory Proxy Server 对 Directory Server 和 Administration Server 具有硬性升级依赖性，因而应在 Directory Server 和 Administration Server 之后进行升级。其他组件可能要通过 Directory Proxy Server 访问 Directory Server。
- 5. Web Server**（参见第 123 页中的第 6 章“Web Server”）
许多 Java ES 组件都要求 Web 容器支持，如果进行升级，则应先于要求 Web 容器服务的组件对其进行升级。Web 容器服务通常由 Web Server 或 Application Server 提供，但如果体系结构中含有这两者，请首先升级 Web Server。
- 6. Message Queue**（参见第 133 页中的第 7 章“Message Queue”）
如果进行升级，Message Queue 最好是在 Application Server 之前升级，后者要求 Message Queue 与 Java 2 Enterprise Edition (J2EE) 兼容。
- 7. High Availability Session Store**（参见第 149 页中的第 8 章“High Availability Session Store”）
如果进行升级，High Availability Session Store 最好是在 Application Server 之前升级，后者需要 High Availability Session Store 来提供高可用性。
- 8. Application Server**（参见第 157 页中的第 9 章“Application Server”）
Application Server 的负载平衡插件依赖于 Web Server，因此，如果要使用该功能，则应在 Web Server 之后升级 Application Server。

9. Web Proxy Server (参见第 175 页中的第 10 章“Web Proxy Server”)

Web Proxy Server 可在任何时候升级, 然而它通常应在 Web Server 或 Application Server 组件之后进行升级, 因为前者要为后两者提供代理服务。Web Proxy Server 是一个新的 Java ES 发行版本 4 组件, 可从其以前的非 Java ES 发行版本进行升级。

10. Access Manager (参见第 183 页中的第 11 章“Access Manager”)

Access Manager 在验证和授权 (包括单点登录) 方面起着核心作用, 如果进行升级, 则应在其为获得这些服务而依赖的组件之前升级。此外, Access Manager 还要求特定的 Directory Server 模式 (模式 2), 该模式对其他组件使用 Directory Server 的方式有影响。

11. Directory Preparation Tool (参见第 209 页中的第 12 章“Directory Preparation Tool”)

Directory Preparation Tool 依赖于 Directory Server 模式, 因而应在 Access Manager 升级之后针对 Directory Server 运行。(有关这条指导原则的例外情况, 参见第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”。) 如果升级 Directory Preparation Tool, 它应在升级通信组件之前升级, 因为这些通信组件依赖于 Directory Preparation Tool 在目录中进行更改, 它们是: Messaging Server、Calendar Server、Communications Express 和 Delegated Administrator。

12. Messaging Server (参见第 223 页中的第 13 章“Messaging Server”)

如果进行升级, Messaging Server 只应在进行上述升级之后且在依赖于 Messaging Server 组件的 Communications Express 之前升级。

13. Calendar Server (参见第 241 页中的第 14 章“Calendar Server”)

如果进行升级, Calendar Server 应在 Messaging Server 之后升级, 因为它的某些功能要求 Messaging Server 支持。Calendar Server 还应在 Communications Express 之前升级, 因为后者对 Calendar Server 具有依赖性。

14. Communications Express (参见第 253 页中的第 15 章“Communications Express”)

如果进行升级, 由于 Communications Express 依赖于上述许多组件 (Calendar Server、Messaging Server、Directory Preparation Tool、Access Manager、Web Server 和 Directory Server), 因此它应在这些组件之后升级。

15. Instant Messaging (参见第 269 页中的第 16 章“Instant Messaging”)

如果进行升级, Instant Messaging 可在升级完 Access Manager 之后的几乎任何时候升级。

16. Portal Server (参见第 287 页中的第 17 章 “Portal Server”)

Portal Server 与 Communications Express 一样依赖于上述许多组件，但它尤其依赖于 Communications Express 来提供消息传送和日历频道，因此，如果进行升级，它应在 Communications Express 之后升级。

17. Portal Server Secure Remote Access (参见第 315 页中的第 18 章 “Portal Server Secure Remote Access”)

如果进行升级，Portal Server Secure Remote Access 可在升级完 Portal Server 之后的任何时候升级。

18. Delegated Administrator (参见第 339 页中的第 19 章 “Delegated Administrator”)

如果进行升级，Delegated Administrator 可在升级完 Directory Preparation Tool 并对 Directory Server 运行了该工具之后的任何时候升级并用来置备用户。按照惯例，用户是在升级和启动了其他服务之后进行置备的，但是，Delegated Administrator 可在升级依赖于 Delegated Administrator 置备用户的通信组件之前升级。

升级 Java ES 共享组件

本章提供有关将 Java ES 共享组件升级至 Java ES 2005Q4（发行版本 4）的信息。

本章包含以下各节：

- 第 48 页中的“共享组件升级概述”
- 第 54 页中的“使用单个修补程序升级共享组件”
- 第 56 页中的“使用修补程序群组升级共享组件”
- 第 59 页中的“通过替换软件包升级组件”
- 第 67 页中的“需要特殊升级过程的组件”

注 要升级共享组件以便为升级 Sun Cluster 软件做准备，请遵循《Sun Cluster 软件安装指南（适用于 Solaris OS）》的“升级 Sun Cluster 软件”中升级依赖软件的过程，该指南可从以下网址获得：

<http://docs.sun.com/doc/819-0420/6n2r1mncr?a=view>

不过，要从 Java ES 2005Q4（发行版本 4）分发包安装安全组件的软件包，而不要从 Sun Cluster 的两张 CD-ROM 中的某一张安装。

共享组件升级概述

将共享组件升级至 Java ES 2005Q4（发行版本 4）应是一个较大规模升级计划的组成部分，如第 1 章“升级规划”中所述。为确保升级成功，请仔细阅读第 1 章，并制定一个满足需求的升级计划。

关于升级计划

升级计划应涵盖以下几个方面：

- **操作系统问题。** 执行任何操作系统的升级，如第 31 页中的“操作系统问题”中所述。对于 Solaris 10 OS 以外的所有平台，升级共享组件之前，应执行操作系统升级。
- **升级途径。** 确定当前使用的 Java Enterprise System 的版本，并确保了解所支持的升级至 Java ES 发行版本 4 的途径。在多数情况下，升级共享组件时，无论是从 Java ES 2004Q2（发行版本 2）升级，还是从 Java ES 2005Q1（发行版本 3）升级，所遵循的升级过程是相同的。本章说明了依赖于特定升级途径的过程。有关升级途径的更多信息，参阅第 33 页中的“升级途径”。
- **升级依赖性。** 了解要升级的产品组件的相互依赖性。通常根据各产品组件的依赖性排列组件的升级顺序。例如，升级某个组件之前，应升级该组件所依赖的任何组件。还应考虑其他各方面的因素，例如硬性升级依赖性和软性升级依赖性，如第 35 页中的“升级依赖性”中所述。

利用第 38 页中的表 1-6 确定在升级产品组件之前需要升级哪些共享组件。

- **全部升级或选择性升级。** 如果要升级计算机上的全部产品组件，则应升级这些产品组件所依赖的所有共享组件。但是，如果从发行版本 3 升级至发行版本 4，则可以选择性地升级计算机上的一些产品组件，而不升级其他组件。不过，最佳做法是升级计算机上全部产品组件所依赖的所有共享组件。经检验，发行版本 4 的共享组件可支持发行版本 3 的产品组件。有关更多信息，参阅第 40 页中的“共享组件升级指导原则”。
- **排序指导原则。** 查看第 43 页中的“一般先后顺序指导原则”中列出的排序指导原则。通常情况下，首先升级共享组件。但是，在开始升级之前，应了解升级至 Java ES 发行版本 4 的整个顺序。

升级共享组件使用的技术

有三种用于将共享组件升级至 Java ES 发行版本 4 的技术。所采用的方法以升级计划为依据，并取决于要升级的共享组件的数量和类型。

这三种技术分别是：

- **修补程序。** 通过应用修补程序，Solaris 平台上的大多数共享组件都可以升级至 Java ES 发行版本 4。修补程序通常升级单个组件或一组有关联的组件。

如果升级计划要求升级少量共享组件，则可以考虑对可使用升级修补程序的组件应用单个修补程序。

第 51 页中的表 2-1 显示了每个共享组件可使用的升级修补程序。第 54 页中的“使用单个修补程序升级共享组件”提供了下载和应用修补程序的说明。

- **修补程序群组。** 修补程序群组捆绑了共享组件可使用的所有升级修补程序。这样会简化升级过程，因为可通过执行修补程序群组随附的单独升级脚本，升级所有相应的共享组件。

每个 Solaris 平台都有一个单独的修补程序群组。修补程序群组不能用于 Linux 平台。

如果升级计划要求升级多个共享组件，则使用修补程序群组。即使不升级修补程序群组所包括的所有共享组件，修补程序群组可能也是将升级计划中指定的共享组件升级的最有效的方法。

有关下载和应用修补程序群组的信息，参阅第 56 页中的“使用修补程序群组升级共享组件”。

- **替换软件包。** 一些共享组件的升级只能通过用更新版本的软件包替换系统中的现有软件包来进行。共享组件软件包的更新版本可通过 Java ES 发行版本 4 的分发包获得。

由于修补技术不能用于在 Linux 平台升级 Java Enterprise System，故通常通过替换 RPM 软件包升级 Linux 共享组件。但是，一些共享组件将 RPM 软件包作为修补程序提供。

第 51 页中的表 2-1 显示了采用替换软件包的方法升级到 Java ES 发行版本 4 的共享组件。第 59 页中的“通过替换软件包升级组件”提供了有关替换软件包的详细信息。

常规升级过程

下面讨论了升级共享组件所采用的常规步骤：

1. 根据升级计划确定升级途径以及要升级的共享组件。
有关开发升级计划的信息，可查看本概述中的前几个小节。也可以参阅第 32 页中的“升级规划”获得其他信息。
2. 确定可用于将升级计划中指定的共享组件升级的技术。
可在以下几节中找到此信息：
 - 第 52 页中的“升级发行版本 2 的共享组件”
 - 第 50 页中的“升级发行版本 3 的共享组件”
3. 根据特定需要，执行以下几节中的过程：
 - 第 54 页中的“使用单个修补程序升级共享组件”
 - 第 56 页中的“使用修补程序群组升级共享组件”
 - 第 59 页中的“通过替换软件包升级组件”
 - 第 67 页中的“需要特殊升级过程的组件”

升级发行版本 3 的共享组件

第 51 页中的表 2-1 显示了从 Java ES 发行版本 3 升级共享组件时使用的升级技术。请注意下列事项：

- **Solaris 平台。** 提供许多不同的升级选项：
 - 在很多情况下，可以使用特定的修补程序将共享组件升级至 Java ES 发行版本 4。
 - 应用特定于平台的修补程序群组通常是从发行版本 3 升级共享组件的最有效的方法。修补程序群组捆绑了可用于将所有共享组件升级至发行版本 4 的全部修补程序。有关更多信息，参阅第 56 页中的“使用修补程序群组升级共享组件”。
 - 多数情况下，必须将发行版本 3 的软件包替换为 Java ES 分发包随附的发行版本 4 的软件包。
 - 对于自发行版本 3 以来没有变化的共享组件，不必进行升级。

- **Linux 平台。** 必须通过安装或替换 RPM 软件包升级共享组件。如果在表 2-1 中列出了修补程序 ID，则已提供的修补程序捆绑了升级相应组件所需的 RPM 软件包，从而可以轻松下载并安装这些软件包。修补程序提供了特殊说明。另外，可以使用修补程序 ID 跟踪对共享组件进行的修改。修补程序群组不能用于 Linux 平台。

注 在表 2-1 中，修补程序 ID 中的后两位数字指出该修补程序的修订版本号。修订版本号越高，表示版本越新。

表 2-1 列出了升级所需的最低修订版本。如果有更新修订版本的修补程序，应使用这些更新的修订版本，而不使用表中列出的版本。

第 27 页中的“发行版本 4 共享组件”中提供了表 2-2 列出的共享组件的全称。

表 2-1 从 Java ES 发行版本 3 升级共享组件的升级技术

共享组件	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
ANT	替换软件包					
ACL	替换软件包					
BDB	替换软件包					
CAC	替换软件包					
ICU	116103-08	114677-10	119810-01	114678-10	119811-01	替换软件包
IM-SDK	118789-09			118790-09		118791-10
J2SE™	按照第 73 页中的“升级 Java ES 发行版本 4 的 J2SE”中所述安装 J2SE 5.0。					
JAF	自发行版本 3 以来无变化。不必升级					
JATO	替换软件包（可选，参见第 71 页中的“升级 JATO”）					
JavaHelp™	自发行版本 3 以来无变化。不必升级					替换软件包
JavaMail™	自发行版本 3 以来无变化。不必升级					
与 JAX 相关的组件 JAXB JAXP JAXR JAX-RPC SAAJ WSCL	应用特定于平台的修补程序群组 参见第 57 页中的“修补程序群组过程”。					119190-03

表 2-1 从 Java ES 发行版本 3 升级共享组件的升级技术 (续)

共享组件	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
JCAPI	自发行版本 3 以来无变化。不必升级					
JDMK	119044-01					119046-01
JSS	119209-05	119211-05	119213-06	119212-05	119214-06	替换软件包
KTSE	自发行版本 3 以来无变化。不必升级					
LDAP C SDK	116837-02			116838-02		118353-02
LDAP J SDK	119725-02					替换软件包
MA Core	119527-02			119528-02		119529-02
MFWK	119803-02			119804-02		替换软件包
NSPR	119209-05	119211-05	119213-05	119212-05	119214-05	替换软件包
NSS	119209-05	119211-05	119213-05	119212-05	119214-05	替换软件包
SASL	115328-02	115342-02	119345-01	115343-02	119346-01	替换软件包
SEDC	参阅第 72 页中的“升级 Sun Explorer Data Collector”。					
SJWC	替换软件包					

升级发行版本 2 的共享组件

下面的表 2-2 显示了从 Java ES 发行版本 2 升级共享组件时使用的升级技术。请注意下列事项：

- **Solaris 平台。** 多数情况下，应用特定于平台的修补程序群组升级共享组件。而在其他情况下，使用 Java ES 分发提供的发行版本 4 的软件包安装或替换软件包。
- **Linux 平台。** 必须通过安装或替换 RPM 软件包升级共享组件。如果在表 2-2 中列出了修补程序 ID，则已提供的修补程序捆绑了升级相应组件所需的 RPM 软件包，从而可以轻松下载并安装这些软件包。修补程序提供了特殊说明。另外，可以使用修补程序 ID 跟踪对共享组件进行的修改。修补程序群组不能用于 Linux 平台。

注 在表 2-2 中，修补程序 ID 中的后两位数字指出该修补程序的修订版本号。修订版本号越高，表示版本越新。

表 2-2 列出了升级所需的最低修订版本。如果有更新修订版本的修补程序，应使用这些更新的修订版本，而不使用表中列出的版本。

第 27 页中的“发行版本 4 共享组件”中提供了表 2-2 列出的共享组件的全称。

表 2-2 从 Java ES 发行版本 2 升级共享组件的升级技术

共享组件	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
ANT	替换软件包					
ACL	替换软件包					
BDB	替换软件包					
CAC	安装软件包					
ICU	应用特定于平台的修补程序群组					替换软件包
IM-SDK	应用特定于平台的修补程序群组					118791-09
J2SE™	按照第 73 页中的“升级 Java ES 发行版本 4 的 J2SE”中所述安装 J2SE 5.0。					
JAF	应用特定于平台的修补程序群组					安装软件包
JATO	应用特定于平台的修补程序群组					安装软件包
JavaHelp™	替换软件包					
JavaMail™	应用特定于平台的修补程序群组					安装软件包
与 JAX 相关的组件 JAXB JAXP JAXR JAX-RPC SAAJ WSCL	应用特定于平台的修补程序群组					119190-03
JCAPI	应用特定于平台的修补程序群组					118613-01
JDMK	应用特定于平台的修补程序群组					119046-01
JSS	应用特定于平台的修补程序群组					安装软件包
KTSE	应用特定于平台的修补程序群组					替换软件包
LDAP C SDK	应用特定于平台的修补程序群组					118353-02

表 2-2 从 Java ES 发行版本 2 升级共享组件的升级技术 (续)

共享组件	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
LDAP J SDK	应用特定于平台的修补程序群组					安装软件包
MA Core	应用特定于平台的修补程序群组					119529-02
NSPR	应用特定于平台的修补程序群组					替换软件包
NSS	应用特定于平台的修补程序群组					替换软件包
SASL	应用特定于平台的修补程序群组					替换软件包
SEDC	参阅第 72 页中的“升级 Sun Explorer Data Collector”。					
SJWC	安装软件包					

使用单个修补程序升级共享组件

在 Solaris 平台上，通过应用修补程序修改已安装的软件包，可将许多共享组件的软件包从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）。

注 从 Java ES 2004Q2（发行版本 2）升级时，不应使用单个修补程序，而应使用修补程序群组，如第 56 页中的“使用修补程序群组升级共享组件”中所述。

基于软件包替换的修补程序技术的优点是：如有需要，以后可逆向恢复由修补程序应用的修订。通常，修补程序小于已更新软件包的大小，因而使下载和安装更加容易。修补程序通常比提供的最新软件包版本更新。如本节中所述，可随时从 SunSolve 下载修补程序的最新修订版本。

第 51 页中的表 2-1 显示了用于从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 的所有共享组件修补程序的修补程序 ID。

如果升级计划要求在 Solaris 平台上升级多个共享组件，则使用修补程序群组可能是执行升级的最有效的方法。修补程序群组中包含的所有修补程序都可用于升级能采用修补程序技术升级的共享组件。有关修补程序群组的更多信息，参阅第 56 页中的“使用修补程序群组升级共享组件”。

如果升级计划要求仅升级少量共享组件，则可能要按以下过程所述应用单个修补程序。以后可以逆向恢复所应用的修补程序，如以下过程所述。

注 更改系统之前，最好先将系统备份。

升级到 Java ES 2005Q4 的修补程序

可使用第 51 页中的表 2-1 中所示的各个修补程序升级共享组件。请遵循以下过程。

1. 获取如表 2-1 中显示的所需修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 应用修补程序前，请阅读修补程序附带的“自述文件”中的所有特殊说明。

4. 使用 patchadd 命令应用修补程序，如下例所示，该命令将修补程序应用到 Mobile Access 核心组件的共享组件：

```
patchadd 119527-02
```

有关 patchadd 命令的信息，参阅 patchadd(1M) 手册页。

修补程序升级回滚

可按照以下过程回滚单个修补程序升级：

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 使用 patchrm 命令逆向恢复修补程序，如下例所示，该命令将修补程序逆向恢复到 Mobile Access 核心组件的共享组件：

```
patchrm 119527-02
```

有关 patchrm 命令的信息，参阅 patchrm(1M) 手册页。

使用修补程序群组升级共享组件

修补程序群组提供了一个将共享组件升级至 Java ES 2005Q4（发行版本 4）的较为方便的方法。每个 Solaris 平台都有一个修补程序群组，其中包含的所有修补程序都可用于能采用修补程序技术升级的共享组件。应用一个共享组件修补程序群组即是应用修补程序群组中包含的所有修补程序。

使用的修补程序群组并不取决于升级途径。从 Java ES 发行版本 2 升级时，使用的特定于平台的修补程序群组与从 Java ES 发行版本 3 升级时相同。

以下为可用的修补程序群组：

```
Java ES 组件修补程序 Solaris 10 SPARC
Java ES 组件修补程序 Solaris 10 x86
Java ES 组件修补程序 Solaris 9 SPARC
Java ES 组件修补程序 Solaris 9 x86
Java ES 组件修补程序 Solaris 8 SPARC
```

可在以下位置从 SunSolve 下载修补程序群组：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

注 修补程序群组不可用于 Linux 平台，因为在 Linux 平台上只有几个共享组件提供了升级到 Java ES 发行版本 4 的修补程序。

修补程序群组内容

修补程序群组中捆绑的修补程序因每个 Solaris 平台而异。提供新的修补程序时，修补程序群组的内容即被更新。随修补程序群组提供的 `Cluster_readme` 文件列出了它所包含的修补程序。

下载修补程序群组时，即获得 ZIP 格式的特定于平台的文件。将该文件的内容解压缩到应用修补程序群组的目录中。解压缩内容的顶级目录包括下列文件：

- `Cluster_readme`
提供有关应用修补程序群组的信息
- `install_cluster` 脚本
运行此脚本可应用修补程序群组
- `copyright`
修补程序群组和文档的版权声明

解压缩内容还包括修补程序群组中所含的所有修补程序的目录。这些修补程序目录包含适用于每个修补程序的自述文件。

修补程序群组过程

如果升级计划要求只升级少量共享组件，则修补程序群组可能不是升级到 Java ES 发行版本 4 的最有效方法。可能要考虑应用单个修补程序，如第 54 页中的“[使用单个修补程序升级共享组件](#)”中所述。

如果升级计划要求在 Solaris 平台上升级多个共享组件，则可能希望使用修补程序群组升级这些组件。

请记住，`install_cluster` 脚本尝试应用修补程序群组中的所有修补程序。该脚本升级安装在运行该脚本的计算机上的共享组件。当修补程序群组脚本尝试对计算机上没有的共享组件应用修补程序或尝试应用不需要的修补程序时，该脚本通常会出现故障。

`install_cluster` 脚本会在执行期间显示其进度。详细信息还可在日志文件中获得。

以下过程显示了如何应用共享组件修补程序群组。

注 不能回滚修补程序群组，而是必须跟踪修补程序群组脚本应用的所有修补程序，并分别回滚每个修补程序（参见第 55 页中的“[修补程序升级回滚](#)”）。应用修补程序群组之前，最好先将系统备份。

从 Solaris 8 和 Solaris 9 升级

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 如果存在以下软件包，则将其删除：

```
SUNWjato
SUNWjaxb
SUNWjaxp
SUNWjaf
SUNWjmail
SUNWxrgrt
SUNwxrpcrt
SUNWxsrt
```

这些选定的软件包可能来自于 JATO、Java Activation Framework (JAF) 或非 Java ES 安装部分的 JAX 系列共享组件，也可能是无法使用修补程序群组升级的版本。必须删除这些软件包才能确保修补程序群组脚本成功升级到这些软件包的更新版本。

可通过运行以下命令删除这些软件包：

```
pkgrm SUNWjato SUNWjaxe SUNWjaxp SUNWjaf SUNWjmail SUNWxrgt \
SUNWxrprt SUNWxsrt
```

3. 在以下位置从 SunSolve 获取适合 Solaris 平台的修补程序群组：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

4. 将特定于平台的 ZIP 文件的内容解压缩到可从中运行安装脚本的位置。
5. 阅读自述文件，其中包含有关修补程序的重要说明和其他信息。

自述文件包含 "Save and Backout Options"（保存和逆向恢复选项）一节，提供了有关如何在以后可能要逆向恢复更改时应用修补程序群组的信息。

6. 运行 `install_cluster` 脚本，这将安装修补程序群组中捆绑的修补程序。

在 Solaris 10 上升级（仅从 Java ES 发行版本 3）

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 确定可能存在于系统中的以下软件包的版本：

```
SUNWjaxp
SUNWxrgt
SUNWxrprt
SUNWxsrt
```

- a. 使用以下命令确定软件包版本：

```
pkgparam -v <package> | grep VERSION
```

- b. 将该版本与下列版本进行比较：

```
<SPARC> VERSION=7.0,REV=2003.05.07.00.23
<x86>    VERSION=7.0,REV=2003.10.10.14.34
```

如果该软件包版本与列出的平台版本不匹配，或者计算机上未安装软件包，则向前跳至第 58 页中的步骤 3。

如果软件包版本确实与列出的适合所用平台的版本匹配，则继续下面的步骤 c。

- c. 确定 Java ES 分发版中是否包含这些软件包：

完整的 Java ES 分发版中包含这些软件包。如果有一个子分发版，则导航到适合所用平台的目录，以查看这些软件包是否存在：

```
Solaris_sparc/Product/shared_components/Packages/  
Solaris_x86/Product/shared_components/Packages/
```

如果子分发版中不包含这些软件包，则不必升级这些软件包。保留计算机上安装的这些软件包并继续第 58 页中的步骤 3。

如果分发版中确实包含这些软件包，则继续下面的步骤 d。

- d. 通过运行以下命令从系统中删除已安装的软件包：

```
pkgrm SUNWjaxp SUNWxrgrt SUNWxrprt SUNWxsrt
```

- e. 通过发出以下命令从分发版安装相应的软件包：

```
pkgadd -d . SUNWjaxp SUNWxrgrt SUNWxrprt SUNWxsrt
```

3. 在以下位置从 SunSolve 获取适合您的 Solaris 平台的修补程序群组：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

4. 将特定于平台的 ZIP 文件的内容解压缩到可从中运行安装脚本的位置。
5. 阅读自述文件，其中包含有关修补程序的重要说明和其他信息。

自述文件包含 "Save and Backout Options"（保存和逆向恢复选项）一节，提供了有关如何在以后可能要逆向恢复更改时应用修补程序群组的信息。

6. 运行 `install_cluster` 脚本，这将安装修补程序群组中捆绑的修补程序。

通过替换软件包升级组件

许多共享组件不使用修补程序技术升级到 Java ES 2005Q4（发行版本 4），而是通过安装新的软件包或将现有软件包替换为更新版本的软件包，升级这些组件。在少数情况下，运行特殊的过程才能正确升级某个共享组件。

- 针对从 Java ES 2004Q2（发行版本 2）升级，第 53 页中的表 2-2 显示了哪些共享组件使用软件包替换升级至 Java ES 发行版本 4。
- 针对从 Java ES 2005Q1（发行版本 3）升级，第 51 页中的表 2-1 显示了哪些共享组件使用软件包替换升级至 Java ES 发行版本 4。
- 第 62 页中的表 2-3 提供了指向 Solaris 平台可能需要的任何特殊过程的链接。
- 第 65 页中的表 2-4 提供了指向 Linux 平台可能需要的任何特殊过程的链接。

软件包替换的升级策略

升级的共享组件的数量取决于升级计划及计算机上安装的 Java ES 组件。对于从 Java ES 发行版本 2 升级，必须升级所有共享组件。

对于从 Java ES 发行版本 3 升级，既可以升级所有组件，也可以有选择性地产品组件升级到发行版本 4。选择要升级的组件时，虽然可以仅选择支持所选要升级的产品组件所必需的 Java ES 共享组件，不过，最好还是选择升级计算机上的所有共享组件。有关更多信息，参阅第 40 页中的“共享组件升级指导原则”。

注 共享组件的升级顺序由组件的相互依赖性确定，在升级计划中应反应出这种相互依赖性。

但是，如果计划将 J2SE 升级到 J2SE 发行版本 5.0，应首先升级此共享组件。J2SE 是所有 Java ES 组件的基本组件。有关更多信息，参见第 73 页中的“升级 Java ES 发行版本 4 的 J2SE”。

升级途径

通过软件包替换升级共享组件的过程通常并不取决于升级途径。从 Java ES 发行版本 2 升级时，采取的过程可与从 Java ES 发行版本 3 升级相同。

但是，第 61 页中的“Solaris 平台的软件包”和第 64 页中的“Linux 平台的软件包”两节中包含一些表，在这些表中指出了少数几个由升级途径决定所用过程的实例。

软件包版本

替换软件包时，应只用这些软件包的更新版本替换软件包。“Solaris 平台的软件包”和“Linux 平台的软件包”两节中提供了有关如何在升级之前比较软件包版本的信息。

软件包位置

升级多数共享组件的软件包随 Java ES 分发包一起提供，并且位于下面的一个目录中，所在具体目录视平台而定：

```
Solaris_sparc/Product/shared_components/Packages/  
Solaris_x86/Product/shared_components/Packages/  
Linux_x86/Product/shared_component/Packages/
```

Solaris 平台上的某些软件包拥有特定于操作系统的版本。可在下列目录中找到这些软件包：

```
<Solaris_ARCH>/Product/shared_components/Solaris_10/Packages/  
<Solaris_ARCH>/Product/shared_components/Solaris_8/Packages/  
<Solaris_ARCH>/Product/shared_components/Solaris_9/Packages/
```

可在以下目录中找到共享组件软件包的本地化版本：

```
<PLATFORM_ARCH>/Product/shared_components/Packages/locale/
```

软件包替换的过程

除少数特殊情况外，可按照以下常规过程替换软件包。这些过程将在以下各节详细介绍：

- 第 63 页中的“在 Solaris 平台上升级软件包”
- 第 66 页中的“在 Linux 平台上升级软件包”

特殊说明

一些软件包具有针对保留配置信息或其他数据的特殊说明。指向特殊说明的链接包含在第 61 页中的“Solaris 平台的软件包”和第 64 页中的“Linux 平台的软件包”两节所列出的软件包表中。

第 67 页中的“需要特殊升级过程的组件”一节中详细介绍所有特殊说明。

Solaris 平台的软件包

下面的表 2-3 列出了通过替换软件包升级的共享组件的 Solaris 软件包。该表还列出了 Java ES 发行版本 4 中可用的软件包的版本及指向升级该组件的所有特殊说明的链接。

对于每个共享组件，软件包都以安装的顺序列出。

只应将软件包替换为更新的版本。替换某个软件包之前，将系统上的软件包版本与要用来替换该版本的软件包版本进行比较。

要确定已安装的软件包的版本，请使用 `pkgparam` 命令和 `verbose (-v)` 选项。此命令的输出结果提供了软件包版本、其修订版本及 `SUNW_PRODVERS` 版本。例如：

```
pkgparam -v SUNWjato | grep VERSION
VERSION='2.1.4,REV=2004.11.10.16.05'

pkgparam -v SUNWjato | grep SUNW_PRODVERS
SUNW_PRODVERS='2.1.4'
```

注 不同软件包的版本处理系统不同，但通常情况下，版本号越高表示软件包的版本越新。

第 27 页中的“发行版本 4 共享组件”指出了表 2-3 中列出的共享组件的全称。

表 2-3 用于在 Solaris 平台上升级共享组件的软件包版本

共享组件	软件包	版本 修订	SUNW_PRODVERS	特殊说明
ANT	SUNWwant	11.11.0 2005.04.06.16.31.04	1.6.2	无
ACL	SUNWaclg	8.1 2005.05.31.17.01.28	1.0.3	无
BDB	SUNWbdb SUNWbdbj	4.2.52 1.0.3	4.2.52, REV=1.0.3	无
CAC	SUNWcacaocfg SUNWcacao	1.1 15	1.1	按照第 67 页中的“升级公共代理容器”中的特殊说明安装或替换这些软件包。
IM-SDK	SUNWiimdv	6.1 2004.04.16.16.01.40	7.0	无
J2SE™ SPARC (32 位)	SUNWj5rt SUNWj5dev SUNWj5cfg SUNWj5man SUNWj5dmo SUNWj5jmp	1.5.0 2004.12.07.00.07	1.5.0_04/ 1.5.0_04-b05	如第 73 页中的“升级 Java ES 发行版本 4 的 J2SE”中所述安装 J2SE™ 平台 5.0 (Java 2 平台, 标准版)。
J2SE™ SPARC (64 位)	SUNWj5rtx SUNWj5dvx SUNWj5dmx	1.5.0 2004.12.06.22.09	1.5.0_04/ 1.5.0_04-b05	
JATO	SUNWjato SUNWjatodoc SUNWjatodmo	2.1.5 2005.04.06.08.07	2.1.5	根据第 71 页中的“升级 JATO”中所述的说明替换 JATO 软件包。
JavaHelp™	SUNWjhrt SUNWjhdev	2.0 2004.11.23	2.0/FCS	从 Java ES 发行版本 2 升级时替换这些软件包。这些软件包自 Java ES 发行版本 3 以来没有变化。

表 2-3 用于在 Solaris 平台上升级共享组件的软件包版本（续）

共享组件	软件包	版本 修订	SUNW_PRODVERS	特殊说明
SEDC	SUNWexplj SUNWexplu SUNWexplo	4.3.1 2004.06.25.07.21	4.3.1 GA	根据第 72 页中的“升级 Sun Explorer Data Collector”中的说明升级 Sun Explorer。
SJWC	SUNWmctag SUNWmconr SUNWmcon SUNWmcos SUNWmcosx	2.2.4 2005.05.09.14.06	SNAG 开发	有关升级 SJWC 的信息，参阅第 72 页中的“升级 Sun Java Web Console”。对于 Solaris 10 平台，不必升级 Java Enterprise System 的 Sun Java Web Console。

在 Solaris 平台上升级软件包

以下过程显示了在 Solaris 平台上升级软件包的一般说明。

少量共享组件除了需要这些一般说明以外，还需要特殊说明。第 62 页中的表 2-3 为需要特殊说明的组件提供了指向这些说明的链接。

注 更改系统之前，最好先将系统备份。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 查看第 62 页中的表 2-3 可获取可能适用于要升级的组件的特殊说明。

升级软件包之前，请遵循所有特殊说明。如果没有指出任何特殊说明，则继续下一步。

3. 导航到 Java ES 发行版本 4 分发版中软件包的位置。

可在下面的一个目录中找到软件包，具体目录视平台而定：

```
Solaris_sparc/Product/shared_components/Packages/  
Solaris_x86/Product/shared_components/Packages/
```

注 如果拥有 Java Enterprise System 的子分发版，则此子分发版中包含上述位置处的全部必需共享组件。

4. 使用 `pkgrm` 命令可删除要升级的软件包的当前版本。

例如，要删除 JATO 软件包：

```
pkgrm SUNWjatodmo SUNWjatodoc SUNWjato
```

有关删除软件包的详细信息，参阅 `pkgrm(1m)` 手册页。

5. 使用 `pkgadd` 命令从分发安装软件包。

例如，要安装 JATO 软件包：

```
pkgadd -d . SUNWjato SUNWjatodoc SUNWjatodmo
```

有关安装软件包的详细信息，参阅 `pkgadd(1M)` 手册页。

6. 使用 `pkgparam` 和 `pkginfo` 命令，检验是否正确安装软件包。

使用 `pkgparam` 和 `-v` 选项检验版本。`pkginfo` 提供了有关软件包的附加信息。

有关这些命令的详细信息，参见 `pkgparam(1)` 和 `pkginfo(1)` 手册页。

Linux 平台的软件包

下面的表 2-4 列出了用于升级共享组件的 Linux RPM 软件包。该表还提供了指向有关升级组件的所有特殊说明的链接。RPM 软件包的版本嵌在软件包名中。

只应将软件包替换为更新的版本。替换某个软件包之前，将系统上的软件包版本与要用来替换该版本的软件包版本进行比较。版本号越高表示软件包的版本越新。

RPM 命名约定提供了有关软件包版本的信息。不同的共享组件嵌入的版本处理信息不同。通常情况下，文件名中嵌入的数字提供了软件包的版本号和修订版本号。

要确定已安装的软件包版本，请使用 `rpm` 查询命令和 `info (-i)` 选项。该命令显示软件包信息，包括名称、版本及说明。例如：

```
rpm -qi SUNWjato-2.1.5.i386.rpm
```

注 不同软件包的版本处理系统不同，但通常情况下，版本号越高表示软件包的版本越新。

第 27 页中的“发行版本 4 共享组件”指出了表 2-4 中列出的共享组件的全称。

表 2-4 用于在 Linux 平台升级共享组件的软件包

共享组件	软件包	特殊说明
ANT	sun-ant-1.6.2-1.rpm	无
ACL	sun-aclg-1.0.3-1.i386.rpm	无
BDB	sun-berkeleydatabase-core-4.2.52-4.4.i386.rpm sun-berkeleydatabase-java-4.2.52-4.4.i386.rpm	无
CAC	sun-cacao-1.1-15.i386.rpm sun-cacaocfg-1.1-15.i386.rpm sun-cacao-man-1.1-15.i386.rpm	按照第 67 页中的“升级公共代理容器”中的特殊说明安装或替换这些软件包。
ICU	sun-icu-3.2-1.i386.rpm	无
IM-SDK	sun-im-dev-6.2.9.13.i386.rpm	无
J2SE™	jdk-1_5_0_04-linux-i586.rpm	如第 79 页中的“在 Linux 平台升级 J2SE”中所述安装 J2SE™ 平台 5.0 (Java 2 平台, 标准版)。
JATO	SUNWjato-2.1.5.i386.rpm SUNWjatodmo-2.1.5.i386.rpm SUNWjatodoc-2.1.5.i386.rpm	根据第 71 页中的“升级 JATO”中所述的说明安装或替换 JATO 软件包。
JavaHelp™	sun-javahelp-2.0-fcs.i586.rpm	根据第 71 页中的“在 Linux 平台升级 JavaHelp”中所述的说明替换这些软件包。
JavaMail	sun-javamail-1.3.2-34.i386.rpm	仅当从 Java ES 发行版本 2 升级时才安装此软件包。该软件包自 Java ES 发行版本 3 以来没有变化。
JSS	sun-jss-4.1-4.i386.rpm	如果要升级安全共享组件, 以便为升级 Sun Cluster 软件做准备, 请参阅第 67 页中的“升级安全组件 (NSS、NSPR、JSS)”。
KTSE	sun-ktsearch-1.3-2.noarch.rpm	如果从 Java ES 发行版本 2 升级, 则安装此软件包。此软件包自 Java ES 发行版本 3 以来没有变化。
LDAP J SDK	sun-ljdk-4.18-4.i386.rpm	如果从 Java ES 发行版本 2 升级, 则安装此软件包。如果从 Java ES 发行版本 3 升级, 则替换此软件包。
MFWK	sun-mfwk-cfg-1.0.1-1.i386.rpm sun-mfwk-dev-1.0.1-1.i386.rpm sun-mfwk-man-1.0.1-1.i386.rpm	如果从 Java ES 发行版本 2 升级, 则安装这些软件包。如果从 Java ES 发行版本 3 升级, 则替换这些软件包。
NSPR	sun-nspr-4.5.2-4.i386.rpm sun-nspr-devel-4.5.2-4.i386.rpm	如果要升级安全共享组件, 以便为升级 Sun Cluster 软件做准备, 请参阅第 67 页中的“升级安全组件 (NSS、NSPR、JSS)”。
NSS	sun-nss-3.10.1-1.i386.rpm	如果要升级安全共享组件, 以便为升级 Sun Cluster 软件做准备, 请参阅第 67 页中的“升级安全组件 (NSS、NSPR、JSS)”。

表 2-4 用于在 Linux 平台升级共享组件的软件包（续）

共享组件	软件包	特殊说明
SASL	sun-sasl-2.18-1.i386.rpm	无
SJWC	SUNWmcon-2.2.4-1.i386.rpm SUNWmconr-2.2.4-1.i386.rpm SUNWmcos-2.2.4-1.i386.rpm SUNWmcosx-2.2.4-1.i386.rpm SUNWmctag-2.2.4-1.i386.rpm	有关升级 SJWC 的信息，参阅第 72 页中的“升级 Sun Java Web Console”。对于 Solaris 10 平台，不必升级 Java Enterprise System 的 Sun Java Web Console。

在 Linux 平台升级软件包

以下过程显示了在 Linux 平台上更新软件包的一般说明。

少量共享组件除了需要这些一般说明以外，还需要特殊说明。第 65 页中的表 2-4 为需要特殊说明的组件提供了指向这些说明的链接。

注 更改系统之前，最好先将系统备份。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 对于每个组件，查看第 65 页中的表 2-4 可获取有关更新这些软件包的特殊说明。更新软件包之前，请遵循所有特殊说明。如果没有特殊说明，则继续下一步。
3. 在 Java ES 发行版本 4 分发包中找到所需的 RPM 软件包。可在以下目录中找到 RPM:

```
Linux_x86/Product/shared_components/Packages/
```

4. 使用 rpm -U 命令更新适合具体状况的 RPM。

例如:

```
rpm -Uvh SUNWjato-2.1.5.i386.rpm SUNWjatodmo-2.1.5.i386.rpm \
SUNWjatodoc-2.1.5.i386.rpm
```

Linux rpm 实用程序会正确排序要安装的软件包。

有关更新软件包的详细信息，参阅 rpm 手册页。

需要特殊升级过程的组件

本节提供了升级需要特殊过程的共享组件的说明。其中包含升级下列共享组件的特殊过程：

- 第 67 页中的“升级安全组件（NSS、NSPR、JSS）”
- 第 67 页中的“升级公共代理容器”
- 第 71 页中的“升级 JATO”
- 第 71 页中的“在 Linux 平台升级 JavaHelp”
- 第 72 页中的“升级 Sun Java Web Console”
- 第 72 页中的“升级 Sun Explorer Data Collector”
- 第 73 页中的“升级 Java ES 发行版本 4 的 J2SE”
- 第 79 页中的“在 Linux 平台升级 J2SE”

升级安全组件（NSS、NSPR、JSS）

如果要升级安全共享组件（NSS、NSPR 和 JSS）以便为升级 Sun Cluster 软件做准备，请遵循《Sun Cluster 软件安装指南（适用于 Solaris OS）》的“升级 Sun Cluster 软件”中升级依赖软件的过程，该指南可从以下网址获得：

<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>

不过，要从 Java ES 发行版本 4 分发安装安全组件的软件包，而不要从 Sun Cluster 的两张 CD-ROM 中的某一张安装。

升级公共代理容器

如果要升级公共代理容器共享组件，以便为升级 Sun Cluster 软件做准备，请遵循《Sun Cluster 软件安装指南（适用于 Solaris OS）》的“升级 Sun Cluster 软件”中升级依赖软件的过程，该指南可从以下网址获得：

<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>

不过，要从 Java ES 发行版本 4 分发安装公共代理容器组件的软件包，而不要从 Sun Cluster 的两张 CD-ROM 中的某一张进行安装。

以下各节介绍针对独立系统升级公共代理容器软件包的过程。

在 Solaris 平台上从 Java ES 发行版本 2 升级

按照第 63 页中的“在 Solaris 平台上升级软件包”中的过程安装公共代理软件包。

在 Linux 平台上从 Java ES 发行版本 2 升级

遵循第 66 页中的“在 Linux 平台上升级软件包”中的过程。但是，还必须应用修补程序 120677-01，它提供了已更新的 Linux RPM 软件包。可在以下网址从 SunSolve 获得该修补程序：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

从 Java ES 发行版本 3 升级

遵循下述特殊过程。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 确保在更新公共代理容器共享组件前，更新公共代理容器所依赖的共享组件。

公共代理容器所依赖的共享组件为 Java 2 Platform Standard Edition (J2SE)、Java Dynamic Management Kit Runtime (JDMK)、Network Security Services (NSS) 及 Netscape Portable Runtime (NSPR)。有关升级这些共享组件的信息，参阅第 51 页中的表 2-1。

如果将 J2SE 升级到版本 5，则需要更新依赖的组件，如第 70 页中的步骤 8 所示。

3. 如果当前安装使用自定义配置设置（例如，使用哪个端口），则使用以下命令捕获配置设置：

在 Solaris 平台上：

```
/opt/SUNWcacao/bin/cacaoadm list-params
```

在 Linux 平台上：

```
/opt/sun/cacao/bin/cacaoadm list-params
```

输出类似于以下形式：

```
java-flags=-Xms4M -Xmx64M
jmxmp-connector-port=10162
snmp-adaptor-port=10161
snmp-adaptor-trap-port=10162
commandstream-adaptor-port=10163
retries=4
```

上例列出的是默认值。请注意，第 70 页中的步骤 7 中未使用任何默认设置。

4. 使用以下命令停止公共代理容器进程:

在 Solaris 平台上:

```
/opt/SUNWcacao/bin/cacaoadm stop
echo $?
```

如果退出代码不是 0, 则强制停止:

```
/opt/SUNWcacao/bin/cacaoadm stop -f
```

在 Linux 平台上:

```
/opt/sun/cacao/bin/cacaoadm stop
echo $?
```

如果退出代码不是 0, 则强制停止:

```
/opt/sun/cacao/bin/cacaoadm stop -f
```

5. 现在可以升级以下公共代理容器软件包, 如下所示:

对于 Solaris 平台, 遵循第 63 页中的“在 Solaris 平台升级软件包”中的过程升级以下软件包:

```
SUNWcacaoocfg
SUNWcacao
```

对于 Linux 平台, 遵循第 66 页中的“在 Linux 平台升级软件包”中的基本过程升级以下软件包。但是, 注意对下述过程的重要更改:

```
sun-cacaoocfg-1.1-15.i386.rpm
sun-cacao-man-1.1-15.i386.rpm
```

公共代理容器 1.1 不支持 `rpm -U`。要在 Linux 平台升级, 请使用以下命令:

```
rpm -e sun-cacao-man-1.0
rpm -e sun-cacao-1.0
rpm -e sun-cacao-config-1.0
rpm -i sun-cacao-config-1.1
rpm -i sun-cacao-1.1
rpm -i sun-cacao-man-1.1
```

6. 仅在 Linux 平台上, 继续之前先应用修补程序 120677-01。

该修补程序提供了已更新的 Linux RPM 软件包, 必须先安装该软件包才能继续。遵循修补程序随附的说明。可在以下网址从 SunSolve Patch Access 获得修补程序 120677-01:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

7. 应用先前在第 68 页中的步骤 3 中捕获的所有自定义配置设置。

在 Solaris 平台上，使用以下命令：

```
/opt/SUNWcacao/bin/cacaoadm set-param java-flags=<Value>
/opt/SUNWcacao/bin/cacaoadm set-param jmxmp-connector-port=<Value>
/opt/SUNWcacao/bin/cacaoadm set-param snmp-adaptor-port=<Value>
/opt/SUNWcacao/bin/cacaoadm set-param snmp-adaptor-trap-port=<Value>
/opt/SUNWcacao/bin/cacaoadm set-param commandstream-adaptor-port=<Value>
/opt/SUNWcacao/bin/cacaoadm set-param retries=<Value>
```

在 Linux 平台上，使用以下命令：

```
/opt/sun/cacao/bin/cacaoadm set-param java-flags=<Value>
/opt/sun/cacao/bin/cacaoadm set-param jmxmp-connector-port=<Value>
/opt/sun/cacao/bin/cacaoadm set-param snmp-adaptor-port=<Value>
/opt/sun/cacao/bin/cacaoadm set-param snmp-adaptor-trap-port=<Value>
/opt/sun/cacao/bin/cacaoadm set-param commandstream-adaptor-port=<Value>
/opt/sun/cacao/bin/cacaoadm set-param retries=<Value>
```

8. 如果已将 J2SE 升级到了 J2SE 版本 5，则运行 rebuild-dependencies 实用程序：

在 Solaris 平台上：

```
/opt/SUNWcacao/bin/cacaoadm rebuild-dependencies
```

在 Linux 平台上：

```
/opt/sun/cacao/bin/cacaoadm rebuild-dependencies
```

此命令的输出将会是：

```
Property updated: [java-home] .
Property updated: [jdk-home] .
Property updated: [nss-lib-home] .
Property updated: [nss-tools-home] .
```

9. 重新启动公共代理容器服务：

```
cacaoadm start
```

10. 检验公共代理容器是否升级：

```
cacaoadm status
cacaoadm verify-configuration
```

升级 JATO

Java ES 发行版本 4 随附的 JATO 软件包版本包含 Sun Java Studio Enterprise 所需的更新。如果使用 Sun Java Studio Enterprise，可能需要已更新的 JATO 软件包。否则，不必升级 Java ES 发行版本 3 随附的 JATO 的早期版本。

在 Solaris 平台上从 Java ES 发行版本 3 升级 JATO

将随 Java ES 发行版本 3 安装的 JATO 的基本版本替换为 Java ES 发行版本 4 分发随附的 JATO 软件包。

仅在使用 Sun Java Studio Enterprise 并需要此更新版本随附的更新时，才需要 JATO 的更新版本。

在 Solaris 平台上从 Java ES 发行版本 2 升级 JATO

1. 应用适合于所用平台的修补程序群组，如第 57 页中的“[修补程序群组过程](#)”中所述。
2. 将使用修补程序群组脚本安装的 JATO 的基本版本替换为 Java ES 发行版本 4 分发随附的 JATO 软件包。

仅在使用 Sun Java Studio Enterprise 并需要此更新版本随附的更新时，才需要 JATO 的更新版本。

在 Linux 平台上从 Java ES 发行版本 3 升级 JATO

将随 Java ES 发行版本 3 安装的 JATO 的基本版本替换为 Java ES 发行版本 4 分发随附的 JATO RPM 软件包。

仅在使用 Sun Java Studio Enterprise 并需要此更新版本随附的更新时，才需要 JATO 的更新版本。

在 Linux 平台上从 Java ES 发行版本 2 升级 JATO

安装 Java ES 发行版本 4 分发随附的 JATO RPM 软件包。

在 Linux 平台上升级 JavaHelp

此特殊过程仅用于在 Linux 平台上升级 JavaHelp。从 Java ES 发行版本 2 和 Java ES 发行版本 3 升级 JavaHelp 时，均可执行此特殊过程。在这两种情况下，必须将 JavaHelp 替换为 Java ES 发行版本 4 分发随附的 RPM。

替换 JavaHelp RPM 软件包时，不要使用 rpm 实用程序的 -U（升级）选项，而要使用 -e（清除）选项，后接 -i（安装）选项，如下所示：

```
rpm -e sun-javahelp-version.rpm  
rpm -i sun-javahelp-2.0-fcs.i586.rpm
```

升级 Sun Java Web Console

对于 Solaris 8 和 Solaris 9 平台，如果要升级 Sun Java Web Console 以便为升级 Sun Cluster 软件做准备，请遵循《Sun Cluster 软件安装指南（适用于 Solaris OS）》的“升级 Sun Cluster 软件”中升级依赖软件的过程，该指南可从以下网址获得：

<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>

不过，要从 Java ES 发行版本 4 分发安装 Sun Java Web Console 的软件包，而不要从 Sun Cluster 的两张 CD-ROM 中的某一张进行安装。

用于升级 Sun Java Web Console 的软件包不在共享组件软件包的标准位置，而是在 Java ES 分发包的下列目录中：

```
<Architecture>/Product/shared_components/Packages/<OperatingSystem>/
```

其中，*Architecture* 可以是 Solaris_sparc 或 Solaris_x86，*OperatingSystem* 可以是 Solaris_8 或 Solaris_9。

没有用于在 Solaris 10 平台上升级 Sun Java Web Console 的过程。Solaris 10 将 Sun Java Web Console 作为操作系统的一部分提供。

升级 Sun Explorer Data Collector

Sun Cluster 软件是 Sun Explorer 需要的唯一组件。

要升级 Sun Explorer 软件，请遵循《Sun Cluster 软件安装指南（适用于 Solaris OS）》的“升级 Sun Cluster 软件”中升级依赖软件的过程，该指南可从以下网址获得：

<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>

不过，要从 Java ES 发行版本 4 分发安装安全组件的软件包，而不要从 Sun Cluster 的两张 CD-ROM 中的某一张安装。

升级 Java ES 发行版本 4 的 J2SE

Java ES 发行版本 4 已针对 Java 2 平台标准版 (J2SE) 版本 5.0 Update 4 (此处标识为 J2SE 5.0 Update 4) 进行验证。(J2SE 5.0 有时称为开发者版本 1.5.0。)除了下文说明的以外, Java ES 版本 4 将继续支持 J2SE 1.4.2 和 J2SE 5.0 Update 1。

注 随 Java ES 发行版本 4 发布的 High Availability Session Store (HADB) 需要有 J2SE 发行版本 5。如果计划升级 HADB, 还必须升级到 J2SE 5.0。

对于 Java ES 发行版本 4, 建议将 J2SE 升级到版本 5.0 Update 4, 但保留安装的 J2SE 1.4.2。

Java Enterprise System 不使用计算机上安装的默认 J2SE, 而是维护指向所支持的 J2SE 版本的符号链接。升级 J2SE 后, 需要设置 Java ES 符号链接, 使其指向已升级的 J2SE。

不过, 应为需要早期版本的服务维护指向 J2SE 1.4.2 的指针。请查阅相应的产品组件文档, 以获取有关如何维护指向 J2SE 早期版本的符号链接的信息。

以下几节提供了在 Solaris 和 Linux 平台上升级 J2SE 的说明:

- [第 73 页中的“在 Solaris 平台上升级 J2SE”](#)
- [第 79 页中的“在 Linux 平台上升级 J2SE”](#)

在 Solaris 平台上升级 J2SE

升级 J2SE 使用的过程取决于是从 J2SE 1.4 升级, 还是从 J2SE 5.0 Update 4 以前的版本升级。

因此, 应确定 Java ES 安装中当前使用的 J2SE 的版本。默认版本如下所示:

- Java ES 发行版本 2。默认版本为 J2SE 1.4.2
- Java ES 发行版本 3。默认版本为 J2SE 版本 5.0 Update 1

由于各种原因，可能已将 J2SE 从默认版本进行了升级。要确定 Java ES 安装使用的 J2SE 版本，请运行以下命令：

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

- 以下是为 J2SE 1.4.2 Update 5 显示的版本字符串示例：

```
java version "1.4.2_05"
```

- 以下是为 J2SE 5.0 Update 1 显示的版本字符串示例：

```
java version "1.5.0_01"
```

Solaris 平台的常规过程

在 Solaris 平台升级 J2SE 的常规过程取决于从哪一版 J2SE 开始升级：

- 如果是从 J2SE 1.4 升级，则遵循此过程
 - a. 安装 J2SE 5.0 Update 4 或更新版本（参见第 75 页中的“在 Solaris 平台上安装 J2SE 5.0”）。
 - b. 将 Java ES 符号链接设置为指向新安装的 J2SE（参见第 78 页中的“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”）。

注 不必删除 J2SE 1.4。这两个版本可安装在同一台计算机上。
可以选择继续在较早版本的 J2SE 上运行某些 Java ES 服务。为此，请查阅相应的组件产品管理指南。例如，可将 Application Server 使用的 J2SE 符号链接更改为指向 J2SE 的上一版本。

- 如果从 J2SE 5.0 Update 4 之前的版本升级，则遵循此过程
 - a. 删除 J2SE 5.0 的早期版本。
 - b. 安装 J2SE 5.0 Update 4 或更新版本（参见第 75 页中的“在 Solaris 平台上安装 J2SE 5.0”）。
 - c. 将 Java ES 符号链接设置为指向新安装的 J2SE（参见第 78 页中的“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”）。或
 - a. 通过应用修补程序升级当前版本的 J2SE（参见第 77 页中的“在 Solaris 平台上通过应用修补程序升级 J2SE 5.0”）。

- b. 将 Java ES 符号链接设置为指向新更新的 J2SE（参见第 78 页中的“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”）。

升级 J2SE 时，可能希望在继续之前关闭所有依赖于当前安装的 J2SE 的服务。这是为了避免因服务正在使用当前 J2SE 而产生任何问题。如果不关闭依赖于 J2SE 的服务，则在安装 J2SE 并将 Java ES 的符号链接设置为指向此 J2SE 后，应重新引导系统。

在 Solaris 平台上安装 J2SE 5.0

在 Solaris 平台上可使用从以下来源之一获得的软件安装 J2SE 5.0:

- Sun 开发者网络
- Java ES 分发包

注 本节所述的过程是将 J2SE 5.0 安装在默认位置。如果要将 J2SE 安装在非默认位置，请遵循 Sun 开发者网络的说明，网址为：

<http://java.sun.com/j2se/1.5.0/install.html>

从 Sun 开发者网络安装 J2SE 5.0:

1. 在 Sun 开发者网络中导航到下述位置，以检索当前版本的 JDK 5.0 和安装说明：

<http://java.sun.com/j2se/1.5.0/download.jsp>

2. 遵循通过下载获得的安装说明。

安装 J2SE 前，可能要停止依赖于 J2SE 的服务，如第 74 页中的“Solaris 平台的常规过程”中所述。

3. 安装完成后，请进入第 78 页中的“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”一节。

从 Java ES 分发包安装 J2SE 5.0:

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. [可选] 按第 74 页中的“Solaris 平台的常规过程”中所述关闭 Java ES 服务。

3. 如果已安装了 J2SE 5.0 的早期版本，则按以下指示将其删除：

删除以下软件包：

```
pkgrm SUNWj5rt SUNWj5dev SUNWj5cfg SUNWj5man SUNWj5dmo
```

对于 64 位处理器的计算机，删除以下附加软件包：

```
pkgrm SUNWj5rtx SUNWj5dvx SUNWj5dmx
```

4. 导航到 Java ES 发行版本 4 分发版中 J2SE 软件包的位置。

可在下面的一个目录中找到软件包，具体目录视平台而定：

```
Solaris_sparc/Product/shared_components/Packages/  
Solaris_x86/Product/shared_components/Packages/
```

注 如果拥有 Java Enterprise System 的子分发版，则此子分发版中包含安装 J2SE 所必需的软件包。

5. 使用 pkgadd 命令安装 J2SE 软件包：

安装以下软件包：

```
pkgadd -d . SUNWj5rt SUNWj5dev SUNWj5cfg SUNWj5man SUNWj5dmo
```

对于 64 位处理器的计算机，安装以下附加软件包：

```
pkgadd -d . SUNWj5rtx SUNWj5dvx SUNWj5dmx
```

这会将 J2SE 5.0 Update 4 安装到 /usr/jdk/jdk1.5.0_04 中。5.0 版本在 Solaris 8 或 Solaris 9 上不会自动成为默认的 Java 平台（除非没有默认平台），但在 Solaris 10 上可以成为默认平台。

注 在 Solaris 8 和 9 上，可通过将 /usr/java 符号链接修改为指向 /usr/jdk/jdk1.5.0_04 使 J2SE 5.0 成为默认 Java 平台。

但是，以这种方式更改符号链接会导致某些尚未用 J2SE 5.0 进行测试的早期 Java 应用程序出现问题。有关更多信息，参阅 J2SE 5.0 安装说明，网址为：

<http://java.sun.com/j2se/1.5.0/compatibility.html>

6. [可选] 安装日语手册页。

利用 `pkgadd` 命令安装新的日语手册页软件包：

```
pkgadd -d . SUNWj5jmp
```

7. 继续下一节，第 78 页中的“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”。

在 Solaris 平台上通过应用修补程序升级 J2SE 5.0

下述过程显示了如何将 J2SE 5.0 平台的已安装版本升级到所支持的版本。

1. 以 `root` 身份登录，或成为超级用户。

```
su -
```

2. 按下表所示，获取 Solaris 平台所需的修补程序。

平台	修补程序
SPARC	118666-03 或更高版本
SPARC (64 位)	118667-03 或更高版本
X86	118668-03 或更高版本
x86 (64 位)	118669-03 或更高版本

修补程序 ID 的后两位数字指出该修补程序的修订版本号。修订版本号越高，表示该修补程序的版本越新。有关特殊说明，参阅列出的每个修补程序的自述文件。

可从以下网址将修补程序下载到 `/tmp`：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

3. [可选] 按第 74 页中的“Solaris 平台的常规过程”中所述关闭 Java ES 服务。

4. 使用 `patchadd` 命令应用修补程序。

例如，对于 Solaris SPARC 平台：

```
patchadd 118666-03
```

5. 继续下一节，“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”。

在 Solaris 平台上设置 Java ES 的 J2SE 符号链接

Java Enterprise System 维护指向所支持的 J2SE 平台版本的符号链接。Java Enterprise System 维护此链接以确保 Java ES 服务可找到要使用的正确 J2SE 运行时。

如果已从 J2SE 1.4.2 升级到了 J2SE 5.0 Update 4，则需设置此符号链接，以使其指向新安装的 J2SE 5.0。如果已从 J2SE 5.0 的早期版本升级到了 J2SE 5.0 Update 4，则只需检验 Java ES 安装是否使用已更新的版本。

下述过程显示了如何将 Java ES 符号链接设置为指向已升级的 J2SE 安装。

1. 重置符号链接 `/usr/jdk/entsys-j2se`，使其指向新安装的或已更新的 J2SE 安装，如下所示：

如果已在默认位置安装了 J2SE 5.0 Update 4，则按以下方法重置该符号链接：

```
rm /usr/jdk/entsys-j2se
ln -s /usr/jdk/instances/jdk1.5.0 /usr/jdk/entsys-j2se
```

如果在非默认位置安装了 J2SE 5.0，则将默认路径 (`/usr/jdk/instances/jdk1.5.0`) 替换为非默认位置的路径。

2. 如果在升级或安装 J2SE 5.0 Update 4 前先停止了服务，则重新启动这些服务。

如果升级或安装 J2SE 5.0 前未停止服务，则可能要重新引导系统，以使依赖于 J2SE 5.0 的服务使用新的符号链接。

检验 J2SE 的更新

以下命令检验 J2SE 符号链接引用的 J2SE 的版本：

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

该命令返回一个含开发者版本号的字符串。例如，如果安装了 J2SE 5.0 Update 4，则此命令返回以下字符串：

```
java version "1.5.0_04"
```

如果上述命令未返回正确的版本，则检查指向 J2SE 的 Java ES 符号链接的设置是否正确，如“在 Solaris 平台上设置 Java ES 的 J2SE 符号链接”中所述。

在 Linux 平台升级 J2SE

在 Linux 平台升级 J2SE 所使用的过程与升级途径无关。无论是从 J2SE 1.4 升级还是从 J2SE 5.0 的早期版本升级，均使用相同的过程。在 Linux 平台上，可以有多个版本的 J2SE 5.0。

Linux 平台升级 J2SE 的常规过程如下所述。

1. 安装 J2SE 5.0 Update 4 或更新版本（参见第 79 页中的“在 Linux 平台上安装 J2SE 5.0”）。
2. 将 Java ES 符号链接设置为指向新安装的 J2SE（参见第 80 页中的“在 Linux 平台上设置 Java ES 的 J2SE 符号链接”）。

注 可以选择删除 J2SE 的早期版本。如果其他服务依赖于早期版本，则可能希望保留安装的这些早期版本。

在 Linux 平台上安装 J2SE 5.0

在 Solaris 平台上可使用从以下来源之一获得的软件安装 J2SE 5.0:

- Sun 开发者网络
- Java ES 分发包

注 本节所述的过程是将 J2SE 5.0 安装在默认位置。如果要安装 J2SE 在非默认位置，请遵循 Sun 开发者网络的说明，网址为：

<http://java.sun.com/j2se/1.5.0/install.html>

从 Sun 开发者网络安装 J2SE 5.0:

1. 在 Sun 开发者网络中导航到下述位置，以检索当前版本的 JDK 5.0 和安装说明：
<http://java.sun.com/j2se/1.5.0/download.jsp>

2. 遵循通过下载获得的安装说明。

安装 J2SE 前，可能要停止依赖于 J2SE 的服务，如第 74 页中的“Solaris 平台的常规过程”中所述。

3. 安装完成后，请进入第 80 页中的“在 Linux 平台上设置 Java ES 的 J2SE 符号链接”一节。

从 Java ES 分发包安装 J2SE 5.0:

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. [可选] 按第 74 页中的“Solaris 平台的常规过程”中所述关闭 Java ES 服务。
3. 导航到 Java ES 分发包中的下述目录，该目录包含 jdk-1_5_0_04-linux-i586.rpm 文件：

```
Linux_x86/Product/shared_components/Packages/
```

注 如果拥有 Java Enterprise System 的子分发包，则此子分发包中包含安装 J2SE 所必需的软件包。

4. 使用以下命令安装 RPM 软件包：

```
rpm -Uvh jdk-1_5_0_04-linux-i586.rpm
```

可以选择删除 J2SE 的早期版本。如果其他服务依赖于早期版本，则可能希望保留安装的这些版本。

5. 继续下一节中的设置 Java ES 符号链接的过程。

在 Linux 平台上设置 Java ES 的 J2SE 符号链接

Java Enterprise System 维护指向所支持的 J2SE 平台版本的符号链接。Java Enterprise System 维护此链接以确保 Java ES 服务可找到要使用的正确 J2SE 运行时。

下述过程显示如何设置 Java ES 符号链接。

1. 重置符号链接 /usr/jdk/entsys-j2se，使其指向新安装的或已更新的 J2SE 安装，如下所示：

如果已在默认位置安装了 J2SE 5 Update 4，则按以下方法重置该符号链接：

```
rm /usr/jdk/entsys-j2se  
ln -s /usr/java/jdk1.5.0_04 /usr/jdk/entsys-j2se
```

这些命令会修改 J2SE 5.0 Update 4 的路径。根据系统中的版本修改 J2SE 平台的路径。

如果在非默认位置安装了 J2SE 5.0，则将默认路径 (/usr/java/jdk1.5.0_04) 替换为非默认位置的路径。

2. 如果在升级或安装 J2SE 5.0 Update 4 前先停止了服务，则重新启动这些服务。
如果升级或安装 J2SE 5.0 前未停止服务，则可能要重新引导系统，以使依赖于 J2SE 5.0 的服务使用新的符号链接。

检验 J2SE 的更新

以下命令检验 J2SE 符号链接引用的 J2SE 的版本：

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

该命令返回一个含开发者版本号的字符串。例如，如果安装了 J2SE 5.0 Update 4，则此命令返回以下字符串：

```
java version "1.5.0_04"
```

如果上述命令未返回正确的版本，则检查指向 J2SE 的 Java ES 符号链接的设置是否正确，如“[在 Linux 平台上设置 Java ES 的 J2SE 符号链接](#)”中所述。

需要特殊升级过程的组件

Sun Cluster 软件

本章介绍如何将 Sun Cluster 软件升级到 Java ES 2005Q4（发行版本 4）：Sun Cluster 3.1 8/05。

本章概括介绍了将 Sun Cluster 软件升级至 Java ES 发行版本 4 的升级问题和过程。

仅在 Solaris 平台上支持 Sun Cluster 软件。

本章所述的 Sun Cluster 软件的升级包括 Sun Cluster 框架软件和 Sun Cluster 数据服务软件（或代理）这两者的升级。

- [第 84 页中的“Sun Cluster 软件升级概述”](#)
- [第 86 页中的“将 Sun Cluster 软件升级到 Java ES 发行版本 4”](#)

Sun Cluster 软件升级概述

本节将在以下方面对 Sun Cluster 软件进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Sun Cluster 软件](#)
- [Sun Cluster 软件升级指导说明](#)
- [Sun Cluster 数据](#)
- [兼容性问题](#)
- [Sun Cluster 组件依赖性](#)

关于 Java ES 发行版本 4 Sun Cluster 软件

Java ES 发行版本 4 Sun Cluster 软件新增了许多功能，其中包括改进的群集安装和升级功能、对 Network Appliance NAS 设备的增强支持、简化的 SunPlex Manager 界面以及其他在 Sun Cluster 发行说明中详细介绍的功能 (<http://docs.sun.com/doc/819-2084/6n4d8kv8c?a=view>)。

Sun Cluster 软件升级指导说明

表 3-1 显示了所支持的 Sun Cluster 到 Java ES 发行版本 4 的升级途径。该表仅适用于 Solaris 操作系统。

Sun Cluster 版本与 Java ES 发行版本并非是一一对应的映射关系。这是因为 Sun Cluster 软件的过渡功能版本 (interim feature releases, IFRs) 已合并到 Java ES 正式发行版本之间的 Java ES 中。因此，如表 3-1 中所示，将 Java ES 发行版本 3 Sun Cluster 和 Java ES 发行版本 2 Sun Cluster 升级至 Java ES 发行版本 4 Sun Cluster 包括将 Sun Cluster 3.1 4/04 和 Sun Cluster 3.1 9/04 这两个软件升级至 Java ES 发行版本 4。

表 3-1 升级到 Java ES 发行版本 4 Sun Cluster 3.1 8/05 (2005Q4) 软件的途径

Java ES 发行版本	Sun Cluster 软件版本	常规方法	必需的重新配置
发行版本 3	Sun Cluster 3.1 9/04 或 Sun Cluster 3.1 8/05	直接升级： 使用 Sun Cluster scinstall 实用程序 执行。	自动迁移到已升级版本的群集配置
发行版本 2	Sun Cluster 3.1 4/04 或 Sun Cluster 3.1 9/04	直接升级： 使用 Sun Cluster scinstall 实用程序 执行。	自动迁移到已升级版本的群集配置
发行版本 1	Sun Cluster 3.1	直接升级未经验证： 但是可使用 scinstall 实用程序执行。	自动迁移到已升级版本的群集配置
Java ES 以前的 发行版本	Sun Cluster 3.0	直接升级未经验证： 但是可使用 scinstall 实用程序执行。	自动迁移到已升级版本的群集配置

Sun Cluster 数据

下表显示了可能受 Sun Cluster 软件升级影响的数据类型。

表 3-2 Sun Cluster 数据用途

数据类型	位置	用途
群集配置数据	群集配置系统信息库，在所有群集节点范围内复制和同步（ 注意： 切勿手动编辑 CCR 文件，这会导致一个节点或整个群集停止工作）	存储关于 Sun Cluster 所有操作方面的配置信息：群集节点配置、故障转移机制、资源管理等

兼容性问题

Java ES 发行版本 4 Sun Cluster 软件包括新的图形管理界面，但与早期发行版本的 Sun Cluster 代理向后兼容。

Sun Cluster 组件依赖性

Sun Cluster 对其他 Java ES 组件的依赖性会影响升级和重新配置 Sun Cluster 软件的过程。例如，Sun Cluster 接口或功能上的变化可能需要 Sun Cluster 软件所依赖的组件的已升级版本。是否需要升级此类组件取决于特定的升级途径。

Sun Cluster 对以下 Java ES 组件具有依赖性：

- **共享组件。** Sun Cluster 软件对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **数据服务。** Sun Cluster 软件需要特定的数据服务（或代理），以使 Java ES 产品组件具有高可用性。对于在 Sun Cluster 环境中运行的每个产品组件，必须提供对应的数据服务来管理对应的群集资源。代理软件包通常在 Sun Cluster 升级过程中进行升级。

将 Sun Cluster 软件升级到 Java ES 发行版本 4

本节包含有关将 Sun Cluster 软件从 Java ES 2005Q1（发行版本 3）和 Java ES 2004Q2（发行版本 2）升级至 Java ES 发行版本 4 的信息。上述 Java ES 发行版本中包含的两种 Sun Cluster 版本（Sun Cluster 3.1 4/04 和 Sun Cluster 3.1 9/04 软件）的升级过程相同。

本节涵盖了以下主题：

- [简介](#)
- [Sun Cluster 升级](#)

简介

在将 Sun Cluster 软件升级至 Java ES 发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过运行 `scinstall` 脚本执行升级，该脚本将升级 Sun Cluster 软件，并在完成软件升级后应用先前的 Sun Cluster 配置。但必须将群集环境中的所有节点均升级到同一版本，方法有两种：一种是关闭群集然后升级所有节点，另外一种是通过滚动升级而不必关闭群集，期间将逐个升级各个节点，每次升级一个。
- **升级依赖性。** 虽然 Sun Cluster 软件对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但 Java ES 发行版本 4 Sun Cluster 软件与这些组件在发行版本 3 中的版本兼容。因此，对于 Sun Cluster 软件到发行版本 4 的升级，这些共享组件的升级是可选的。

- **向后兼容性。** 发行版本 4 Sun Cluster 软件与早期的群集代理向后兼容，但一个群集中的所有节点必须运行同一个版本的框架和代理软件。
- **升级回滚。** 不支持 Sun Cluster 软件的发行版本 4 到早期版本的升级回滚。
- **平台问题。** 在所有 Solaris 平台上，Sun Cluster 软件的升级方法都是相同的，但在 Linux 平台上不支持 Sun Cluster 软件。

Sun Cluster 升级

本节概述了如何执行 Sun Cluster 软件从 Java ES 发行版本 3 到 Java ES 发行版本 4 的升级：

- [升级前的任务](#)
- [升级 Sun Cluster 软件](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

本节涵盖了 Sun Cluster 非滚动升级的情况。滚动升级的情况有些不同，因为滚动升级时不关闭群集。但是如下所述，对于给定群集节点，这两种情况都包含了相同的常规过程。具体过程可在 [Sun Cluster 安装指南升级一章](#) (<http://docs.sun.com/doc/819-0173/6n2ka5si2?a=view>) 中找到。

升级前的任务

升级 Sun Cluster 软件之前，应执行下述任务。

检验当前版本信息

可通过输入以下命令检验 Sun Cluster 软件的当前版本：

```
% scinstall -pv
```

该命令将返回 Sun Cluster 版本和已安装的各软件包的版本。如果该命令返回了 3.1 8/05 版本 3.1u4，则无需升级到 Java ES 发行版本 4。

表 3-3 Sun Cluster 版本检验输出

Java ES 发行版本	Sun Cluster 版本号
发行版本 1 (Sun Cluster 3.1)	3.1
发行版本 2 (Sun Cluster 3.1 4/04)	3.1u2

表 3-3 Sun Cluster 版本检验输出

Java ES 发行版本	Sun Cluster 版本号
发行版本 2 或 3 (Sun Cluster 3.1 9/04)	3.1u3
发行版本 3 或 4 (Sun Cluster 3.1 8/05)	3.1u4

为升级准备群集节点

必须先从群集环境中删除群集节点，然后才能升级 Sun Cluster 软件：

- **非滚动升级。** 从群集环境中删除节点意味着关闭群集环境：将资源组切换到脱机状态、禁用资源组、关闭正在环境中运行的应用程序、备份共享数据、关闭群集、备份系统磁盘，然后重新引导节点，进入非群集模式。
- **滚动升级。** 从群集环境中删除节点意味着转移节点的所有资源组和设备组、备份共享数据和系统磁盘，然后重新引导节点，进入非群集模式。

有关这些操作以及可能需要在特定情况下执行的其他操作的详细信息，参见 Sun Cluster 安装指南的升级一章 (<http://docs.sun.com/doc/819-0173/6n2ka5si2?a=view>)。

升级操作系统

您可能想利用任何升级停机时间来将操作系统升级到其最新版本，同时也升级正在使用的卷管理器的版本。

有关这些操作的详细信息，参见 Sun Cluster 安装指南的升级一章 (<http://docs.sun.com/doc/819-0173/6n2ka5si2?a=view>)。

升级 Sun Cluster 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。Sun Cluster 软件所依赖的发行版本 3 共享组件的升级是可选的，而发行版本 2 共享组件到发行版本 4 的升级则是必要的。

要升级 Sun Cluster 软件所依赖的所有共享组件（参见第 38 页中的表 1-6），可按照 Sun Cluster 安装指南的升级一章 (<http://docs.sun.com/doc/819-0173/6n2ka5si2?a=view>) 中提供的说明操作，不过，即使满足最低版本要求，也要升级所有共享组件。

获取所需的配置信息和密码

不需要任何与当前安装版本有关的特殊信息，但是必须以超级用户身份登录才能执行升级。

升级 Sun Cluster 软件

本节先讨论影响 Sun Cluster 软件升级过程的若干注意事项，随后对升级过程本身加以说明。

升级注意事项

将 Sun Cluster 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 在升级 Sun Cluster 框架软件时，最好也将管理 Java ES 高可用性组件所需的数据服务或者其他在群集环境中运行的应用程序升级。
- 升级 Sun Cluster 软件的同时也可以升级 Java ES 组件或者在群集环境中运行的其他应用程序。

升级过程

下面的过程适用于在每个群集节点上升级 Sun Cluster 软件。以下步骤只是一般性步骤，有关如何执行这些步骤的详细信息，参见 Sun Cluster 安装指南的升级一章 (<http://docs.sun.com/doc/819-0173/6n2ka5si2?a=view>)。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 将目录切换到 Java ES 发行版的以下目录：

```
cd /Solaris_arch/Product/sun_cluster/Solaris_ver/Tools
```

其中 *arch* 是 *sparc* 或 *x86*，*ver* 是 8、9 或 10，分别代表 Solaris 8、9 或 10。

3. 运行 `scinstall` 实用程序。

```
./scinstall
```

将显示用于执行群集安装、配置和升级任务的主菜单。

4. 升级 Sun Cluster 框架软件以及任何所需的数据服务。

已升级数据服务需要通过将对应的资源迁移到已升级资源类型来进行配置（参见第 90 页中的“升级后的任务”）。

5. 向 Sun Cluster 框架软件和数据服务应用所有必要的修补程序。

有关访问和应用相关修补程序的信息，参见 Sun Cluster 发行说明 (<http://docs.sun.com/doc/819-2084>)。

6. 重新引导节点，进入群集。

检验升级

可按如下步骤检验 Sun Cluster 软件的升级是否成功：

1. 检查 Sun Cluster 框架软件的版本号。

```
scinstall -pv
```

参见第 87 页中的表 3-3 确认输出值。

2. 检查数据服务升级日志文件。

在升级输出消息的末尾提到了日志文件。

升级后的任务

在执行完 Sun Cluster 软件的升级后，可能还需要执行许多附加任务，这取决于您执行的是非滚动升级还是滚动升级。其中，完全恢复群集环境所需的任务有：

- 检验群集配置的状态
- 将资源迁移到新资源类型版本
- 升级安装在群集中的其他 Java ES 组件或者应用程序

有关这些安装后步骤的详细信息，参见 Sun Cluster 安装指南的升级一章 (<http://docs.sun.com/doc/819-0173/6n2ka5si2?a=view>)。

升级回滚

不支持 Sun Cluster 软件的回滚。升级过程中所做的更改无法轻易逆向恢复。

Directory Server 和 Administration Server

本章介绍如何将 Directory Server 和 Administration Server 组件升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Directory Server 5.2 2005Q4 和 Sun Java System Administration Server 5.2 2005Q4。

由于这些升级之间有紧密的联系，因此将它们放在一起进行叙述。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- [第 92 页中的“Directory Server 和 Administration Server 升级概述”](#)
- [第 95 页中的“从 Java ES 发行版本 3 升级 Directory Server 和 Administration Server”](#)
- [第 110 页中的“从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server”](#)

注 本章中的文件位置是相对于一个称为 *serverRoot* 的目录路径而指定的。在最初安装和配置 Directory Proxy Server 时，可能至少已将此路径的一部分指定为安装目录。否则，会分配一个默认值。

serverRoot 的默认值视操作系统平台而定：

- Solaris: `/var/opt/mps/serverroot`
 - Linux: `/var/opt/sun/directory-server`
-

Directory Server 和 Administration Server 升级概述

本节将在以下方面对 Directory Server 和 Administration Server 组件进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4](#)
- [Java ES 发行版本 4 升级指导说明](#)
- [Directory Server 和 Administration Server 数据](#)
- [兼容性问题](#)
- [组件依赖性](#)

关于 Java ES 发行版本 4

Java ES 发行版本 4 的 Directory Server 和 Administration Server 版本仅修复了一些小错误和做出了一些小的改进，并没有新增任何功能。

Java ES 发行版本 4 升级指导说明

表 4-1 显示了所支持的 Directory Server 和 Administration Server 到 Java ES 发行版本 4 的升级途径。该表同时适用于 Solaris 和 Linux 两种操作系统。

表 4-1 升级到 Java ES 发行版本 4: Sun Java System Directory Server 5.2 2005Q4 和 Sun Java System Administration Server 5.2 2005Q4 的途径

Java ES 发行版本	Directory Server、Administration Server 和 Directory Proxy Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Directory Server 5 2005Q1 Sun Java System Administration Server 5 2005Q1	直接升级： 应用修补程序并重新配置配置目录。	配置目录中数据的自动重新配置

表 4-1 升级到 Java ES 发行版本 4: Sun Java System Directory Server 5.2 2005Q4 和 Sun Java System Administration Server 5.2 2005Q4 的途径 (续)

Java ES 发行版本	Directory Server、Administration Server 和 Directory Proxy Server 版本	常规方法	必需的重新配置
发行版本 2	Sun Java System Directory Server 5.2 2004Q2 Sun Java System Administration Server 5.2 2004Q2	直接升级: 应用修补程序并重新配置配置目录。	配置目录中数据的自动重新配置
发行版本 1	Sun One Directory Server 5.2 Sun One Administration Server 5.2	直接升级未经验证: 但可以使用发行版本 2 的升级方法。	配置目录中数据的自动重新配置
Java ES 以前的发行版本	Sun One Directory Server 5.2 Sun One Administration Server 5.2 Sun One Directory Server 5.1、5.0 或 4.x Sun One Administration Server 5.1、5.0 或 4.x	直接升级未经验证: 但可以使用发行版本 2 的升级方法。 不直接升级: 首先升级到发行版本 3。参阅 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237)。 然后从发行版本 3 升级到发行版本 4。	配置目录中数据的自动重新配置 参阅 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237)。

Directory Server 和 Administration Server 数据

Directory Server 和 Administration Server 利用 Directory Server 本身来存储配置数据。数据存储在目录内部的一个特定树结构中。托管配置的 Directory Server 实例称为配置目录。

配置目录可以是专用的 Directory Server 实例 (建议采用这种安全做法), 也可以托管用户身份数据或服务配置数据。配置目录可以与其他 Directory Server 实例或 Administration Server 驻留在同一台计算机上; 但是在大多数部署体系结构中, 配置目录与使用它存储配置信息的其他组件相距甚远。

下表显示了可能受 Directory Server 和 Administration Server 软件升级影响的数据类型。

表 4-2 Directory Server、Administration Server 和 Directory Proxy Server 数据用途

数据类型	位置	用途
Directory Server 配置数据	配置目录	Directory Server 的配置
Administration Server 配置数据	配置目录	Administration Server 的配置

兼容性问题

Java ES 发行版本 4 Directory Server 和 Administration Server 未引入任何接口更改。这些组件在整体上与早期版本向后兼容。但是，这两个组件并不都与其他组件的早期版本向后兼容；两者需要进行集体升级。

组件依赖性

对其他 Java ES 组件的依赖性会影响升级和重新配置 Directory Server 和 Administration Server 软件的过程。上述每个组件均依赖于如下的 Java ES 组件：

- **Directory Server。** Directory Server 依赖于特定的 Java ES 共享组件（参见第 38 页中的表 1-6）。Directory Server 依赖于 Administration Server，后者用于配置 Directory Server 复制以及 Directory Server 功能的其他方面。
- **Administration Server。** Administration Server（和管理控制台用户界面）依赖于特定的 Java ES 共享组件（参见第 38 页中的表 1-6）。Administration Server 还依赖于存储配置数据的 Directory Server（特别是配置目录）。

从 Java ES 发行版本 3 升级 Directory Server 和 Administration Server

本节包含有关将 Directory Server 和 Administration Server 从 Java ES 2005 Q1（发行版本 3）升级到 Java ES 2005Q4（发行版本 4）的信息。其中涵盖了以下主题：

- [简介](#)
- [发行版本 3 Directory Server 和 Administration Server 的升级](#)
- [多实例升级](#)

简介

在将 Java ES 发行版本 3 Directory Server 和 Administration Server 升级到发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过向 Java ES 发行版本 3 的版本应用修补程序来执行升级。通过将配置目录与已升级软件进行同步来实现 Directory Server 和 Administration Server 的重新配置。
- **升级依赖性。** 虽然 Directory Server 和 Administration Server 依赖于许多 Java ES 共享组件（参见第 38 页中的表 1-6），但 Java ES 发行版本 4 Directory Server 和 Administration Server 与这些共享组件在发行版本 3 中的版本兼容。因此，对于 Directory Server 和 Administration Server 到发行版本 4 的升级，这些共享组件的升级是可选的。

Directory Server 对 Administration Server 具有硬性升级依赖性。因此，应将这些组件一同升级到发行版本 4。

- **向后兼容性。** 发行版本 4 Directory Server 和 Administration Server 与其发行版本 3 的版本向后兼容。
- **升级回滚。** 在 Solaris 上，发行版本 4 升级回滚是通过删除发行版本 4 升级修补程序并将配置目录与先前软件状态重新进行同步来实现的。但是在 Linux 上，并不存在回滚发行版本 4 升级的过程。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Directory Server 和 Administration Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Directory Server 和 Administration Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上执行 Directory Server 和 Administration Server 从 Java ES 发行版本 3 到 Java ES 发行版本 4 的升级。在某个主题取决于平台特定过程之处，该主题将指明所适用的操作系统。本节涵盖了以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Directory Server 和 Administration Server \(Solaris\)](#)
- [升级发行版本 3 Directory Server 和 Administration Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

在升级 Directory Server 和 Administration Server 之前，应执行下述任务。

检验当前版本信息

可通过以下方法来检验 Directory Server 和 Administration Server 的当前版本，首先使用 `-v` 选项重新启动 Directory Server 守护进程：

```
cd serverRoot/bin/slapd/server  
./ns-slapd -v
```

然后检查 Directory Server 错误日志中的启动消息：

```
serverRoot/slapd-hostName/logs/errors
```

表 4-3 Directory Server 版本检验输出

Java ES 发行版本	Directory Server 版本号
发行版本 2	Sun Java(TM) System Directory Server/5.2_Patch_2
发行版本 3	Sun Java(TM) System Directory Server/5.2_Patch_3
发行版本 4	Sun Java(TM) System Directory Server/5.2_Patch_4

注意：如果 ns-slapd 命令在 Solaris 10 平台上失败，请在运行该命令时将库路径设置为空：

```
LD_LIBRARY_PATH= ./ns-slapd -v
```

升级 Directory Server 和 Administration Server 依赖项

一般而言，建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，由于 Directory Server 和 Administration Server 并不要求升级发行版本 3 共享组件，所以此任务是可选的。

备份 Directory Server 数据

Directory Server 和 Administration Server 升级过程会修改配置目录数据。因此，在升级之前，建议您使用 Directory Server 控制台或命令行实用程序（如 db2bak）备份配置目录数据。

有关备份 Directory Server 的更多信息，参见 Sun Java System Directory Server 管理指南 (<http://docs.sun.com/doc/819-2013>)。

获取所需的配置信息和密码

您应当知道当前所安装版本的 Directory Server 管理员用户 ID 和密码。

此外，Directory Server 和 Administration Server 必须以同一用户和组的身份运行。也就是说，它们必须以同一 UID 和 GID 运行。

升级发行版本 3 Directory Server 和 Administration Server (Solaris)

本节先讨论影响 Directory Server 和 Administration Server 升级过程的若干注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Directory Server 和 Administration Server 软件升级到 Java ES 发行版本 4 时，应考虑以下注意事项：

- 在升级 Directory Server 实例之前，应先关闭使用该实例的所有 Java ES 组件（如 Access Manager、Communications Express、Messaging Server、Portal Server 等）。但是，大多数部署体系结构都使用 Directory Server 的多个实例来提供高可用性或可伸缩性。在这种情况下，可以执行 Directory Server 的滚动升级，而无需关闭 Directory Server 客户机。
- Administration Server 必须在 Directory Server 之前进行升级，因为数据的新配置必须以特定顺序进行。

- 在应用修补程序时，必须关闭正在升级的组件，但是必须随后运行相关联的配置目录，方可重新配置正在升级的组件。
- 在其中有多个 Directory Server 实例运行于单台计算机上的部署体系结构中（所有实例均对应于同一个已安装的 Directory Server 映像），升级 Directory Server 映像将升级所有实例。在此类体系结构中，每个已安装的 Directory Server 映像仅有一个 Administration Server 实例。
- 在许多部署体系结构中，配置目录是单独的 Directory Server 实例。它可能在本地，也可能位于正在从中执行升级的另一不同计算机系统上。同样，Administration Server 可能在本地，也可能位于正在从中执行 Directory Server 升级的另一不同计算机系统上。
- 在某些部署体系结构中，已通过在安装时取消选择 Administration Server 独立安装了 Directory Server。但是，如随后的说明中所述，在这种情况下，除了 Directory Server 升级过程之外，仍必须执行 Administration Server 升级过程（在独立模式下甚至会安装某些 Administration Server 代码）。
- 下表显示了用于 Solaris OS 的发行版本 4 Directory Server 和 Administration Server 升级修补程序：

表 4-4 用于在 Solaris 上升级 Directory Server 和 Administration Server 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Directory Server	115614-26	115615-26
Directory Server 本地化版本	117015-21	117015-21
Administration Server	115610-23	115611-23
Administration Server 本地化版本	117047-24	117047-24

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于本地驻留在即将进行升级的计算机上的 Directory Server 和 Administration Server 实例。

下面的步骤利用了两个命令：`directoryserver(1m)` 和 `mpsadmserver(1m)`。有关这些命令的更多信息，参见 Directory Server 手册页参考和 Administration Server 手册页参考。

1. 根据表 4-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果管理控制台正在本地运行，则将其停止。

4. 关闭依赖于即将升级的 Directory Server 实例的所有 Java ES 组件。此步骤可能取决于这些组件在部署体系结构中的复制方式。

应按以下顺序关闭组件：

- a. Directory Server 客户机：Access Manager、Communications Express、Messaging Server、Portal Server 等
- b. Directory Proxy Server（如果使用它访问 Directory Server）
- c. Administration Server（如果它在本地运行）
- d. Directory Server
- e. 配置目录（如果它作为单独的 Directory Server 实例在本地运行）

有关如何关闭 Java ES 组件的信息，参见其各自的管理指南。

5. 确保已升级 Directory Server 和 Administration Server 对其具有硬性升级依赖性的所有 Java ES 组件（参见第 97 页中的“升级 Directory Server 和 Administration Server 依赖项”）。

6. 升级 Administration Server。

即使最初已在独立模式下，在进行升级的计算机上安装了 Directory Server（在独立模式下甚至会安装某些 Administration Server 代码），亦需要执行此步骤。

- a. 重新启动要升级的 Administration Server。

- b. 应用表 4-4 中的 Administration Server 修补程序。

确保先应用 Administration Server 本地化修补程序 (117047)，然后再应用 Administration Server 基本修补程序。

```
patchadd patch_ID
```

- c. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 b 中应用的修补程序 ID 的版本。

- d. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。

- e. 使升级后的设置与配置目录同步。

```
/usr/sbin/mpsadmserver sync-cds
```

将提示您输入管理员用户名和密码。

7. 升级 Directory Server。

- a. 如果不运行 Administration Server 而是在独立模式下运行 Directory Server，请执行以下过程，否则，直接转到步骤 7b。

- I. 确保已升级 Administration Server，步骤 6。

- II. 将目录转到 *serverroot* 目录。

```
cd /var/opt/mps/serverroot
```

- III. 创建配置目录：

```
mkdir -p admin-serv/config
```

- IV. 创建 adm.config 文件：

```
vi admin-serv/config/adm.conf
```

V. 添加以下文本

```
isie: cn=Administration Server, cn=Server Group, cn=hostname,  
ou=administration_domain, o=NetscapeRoot
```

所有文本都在一行，其中 *hostname* 为全限定 Directory Server 主机名，*administration_domain* 通常为主机的域名。

- b. 确保关闭了正在升级的 Directory Server 实例。

- c. 应用表 4-5 中的 Directory Server 修补程序。

确保先应用 Directory Server 本地化修补程序 (117015)，然后再应用 Directory Server 基本修补程序。

```
patchadd patch_ID
```

- d. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 c 中应用的修补程序 ID 的版本。

- e. 重置默认的 Directory Server 版本号：

```
/usr/sbin/directoryserver -d 5.2
```

- f. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。

- g. 使升级后的设置与配置目录同步。

```
/usr/sbin/directoryserver -u 5.2 sync-cds
```

将提示您输入管理员用户名和密码。

8. 以步骤 4 中关闭组件的相反顺序重新启动所有 Java ES 组件。

- a. 配置目录（如果它在本地作为单独的 Directory Server 实例运行）

- b. Directory Server

- c. Administration Server（如果它在本地运行）

- d. Directory Proxy Server（如果使用它访问 Directory Server）

- e. Directory Server 客户机：Access Manager、Communications Express、Messaging Server、Portal Server 等

升级发行版本 3 Directory Server 和 Administration Server (Linux)

本节先讨论影响 Directory Server 和 Administration Server 升级过程的若干注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除了 Linux 发行版本 4 升级修补程序与 Solaris 修补程序不同之外，在 Linux 平台上将 Directory Server 及其关联组件升级到 Java ES 发行版本 4 时，应考虑的注意事项与在 Solaris 平台上升级时的注意事项相同（参见第 97 页中的“升级注意事项 (Solaris)”）。

下表显示了用于 Linux OS 的发行版本 4 Directory Server 和 Administration Server 升级修补程序：

表 4-5 用于在 Linux 上升级 Directory Server 和 Administration Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Directory Server	118080-11: sun-directory-server-5.2-25.i386.rpm sun-directory-server-man-5.2-9.i386.rpm
Directory Server 本地化版本	118290-12: sun-directory-server- <i>Locale</i> -5.2-17.i386.rpm
Administration Server	118079-10: sun-admin-server-5.2-18.i386.rpm sun-server-console-5.2-18.i386.rpm sun-admin-server-man-5.2-8.i386.rpm
Administration Server 本地化版本	118289-13: sun-admin-server- <i>Locale</i> -5.2-19.i386.rpm sun-server-console- <i>Locale</i> -5.2-19.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于本地驻留在即将进行升级的计算机上的 Directory Server 和 Administration Server 实例。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

下面的步骤利用了两个命令：`directoryserver(1m)` 和 `mpsadmserver(1m)`。有关这些命令的更多信息，参见 Directory Server 手册页参考和 Administration Server 手册页参考。

1. 使用表 4-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

在以下过程中，*oldVersion* 表示用于发行版本 3 Directory Server 和 Administration Server 版本的 RPM。

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果管理控制台正在本地运行，则将其停止。
4. 关闭依赖于即将升级的 Directory Server 实例的所有 Java ES 组件。此步骤可能取决于这些组件在部署体系结构中的复制方式。

应按以下顺序关闭组件：

- a. Directory Server 客户机：Access Manager、Communications Express、Messaging Server、Portal Server 等
- b. Directory Proxy Server（如果使用它访问 Directory Server）
- c. Administration Server（如果它在本地运行）
- d. Directory Server
- e. 配置目录（如果它作为单独的 Directory Server 实例在本地运行）

有关如何关闭 Java ES 组件的信息，参见其各自的管理指南。

5. 确保已升级 Directory Server 和 Administration Server 对其具有硬性升级依赖性的所有 Java ES 组件（参见第 97 页中的“升级 Directory Server 和 Administration Server 依赖项”）。

6. 应用 Administration Server 的每个 RPM。

a. 应用 Administration Server 的 RPM：产品。

即使最初已在独立模式下，在进行升级的计算机上安装了 Directory Server，亦需要执行此步骤。

I. 按如下所示应用 RPM：

确保先应用 Administration Server 本地化 RPM (118289)，然后再应用 Administration Server 基本 RPM。

```
rpm -Fvh sun-admin-server-Locale-5.2-19.i386.rpm  
rpm -Fvh sun-server-console-Locale-5.2-19.i386.rpm  
rpm -Fvh sun-admin-server-5.2-18.i386.rpm  
...
```

如果以前配置了 Administration Server，将会返回以下错误：

```
error: execution of %preun scriptlet from  
sun-admin-server-5.2-oldVersion failed, exit status 1
```

在这种情况下，请使用 `--noscripts` 选项删除旧版的 RPM，如下所示：

```
rpm -e --noscripts sun-admin-server-5.2-oldVersion
```

II. 如果以前配置了 Administration Server，确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。

III. 使升级后的设置与配置目录同步。

```
/opt/sun/sbin/mpsadmserver sync-cds
```

将提示您输入管理员用户名和密码。

IV. 确认升级成功：

```
rpm -q sun-admin-server
```

应该返回 RPM 的新版本号。

b. 应用 Administration Server 的 RPM：控制台。

```
rpm -Fvh sun-server-console-5.2-18.i386.rpm
```

c. 应用 Administration Server 的 RPM：手册页。

```
rpm -Uvh sun-admin-server-man-5.2-8.i386.rpm
```

7. 应用 Directory Server 的每个 RPM。

- a. 如果不运行 Administration Server 而是在独立模式下运行 Directory Server，请应用 Administration Server RPM。

```
rpm -Fvh sun-admin-server-5.2-18.i386.rpm
```

否则，直接转到步骤 7b。

- b. 应用 Directory Server 的 RPM：产品。

- I. 确保关闭了正在升级的 Directory Server 实例。

- II. 按如下所示应用 RPM：

确保先应用 Directory Server 本地化 RPM (118290)，然后再应用 Directory Server 基本 RPM。

```
rpm -Fvh sun-directory-server-Locale-5.2-17.i386.rpm
rpm -Fvh sun-directory-server-5.2-25.i386.rpm
...
```

如果以前配置了 Directory Server，将会返回以下错误：

```
error: execution of %preun scriptlet from
sun-directory-server-5.2-oldVersion failed, exit status 1
```

在这种情况下，请使用 `--noscripts` 选项删除旧版的 RPM，如下所示：

```
rpm -e --noscripts sun-directory-server-5.2-oldVersion
```

- III. 如果以前配置了 Directory Server，确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。

- IV. 使升级后的设置与配置目录同步。

```
/opt/sun/sbin/directoryserver sync-cds
```

将提示您输入管理员用户名和密码。

- V. 确认升级成功：

```
rpm -q sun-directory-server
```

应该返回 RPM 的新版本号。

- c. 应用 Directory Server 的 RPM：手册页。

```
rpm -Uvh sun-directory-server-man-5.2-9.i386.rpm
```

8. 以步骤 4 中关闭组件的相反顺序重新启动所有 Java ES 组件。
 - a. 配置目录（如果它在本地作为单独的 Directory Server 实例运行）
 - b. Directory Server
 - c. Administration Server（如果它在本地运行）
 - d. Directory Proxy Server（如果使用它访问 Directory Server）
 - e. Directory Server 客户机：Access Manager、Communications Express、Messaging Server、Portal Server 等

检验升级

可通过以下方法检验 Directory Server 和 Administration Server 的升级是否成功，首先使用 `-v` 选项重新启动 Directory Server 守护进程：

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

然后检查 Directory Server 错误日志中的启动消息：

```
/var/opt/mps/serverroot/logs/errors
```

参见第 96 页中的表 4-3 确认输出值。

升级后的任务

除第 99 页中的“升级过程 (Solaris)”和第 103 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节先介绍影响 Directory Server 和 Administration Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

对于到发行版本 4 的 Directory Server 和 Administration Server 的升级，其回滚过程正好与升级到发行版本 4 的过程相反。将会删除修补程序并重新同步配置目录。

需要特别注意的一点是，在应用修补程序时，会将 SSL 证书数据库升级至 `cert8` 格式。修补程序将备份 `cert7` 数据，然后将其转换为 `cert8` 格式。如果您随后决定回滚升级且已将新证书添加到了证书数据库中，则应手动提取这些证书，逆向恢复修补程序，然后再将证书重新添加到先前的 `cert7` 格式的证书数据库中。

当您在更改了 SSL 证书数据库之后回滚升级时，将无法以 SSL 模式启动。要解决此问题，请关闭 SSL 模式，重新启动 Directory Server 和 Administration Server，重新安装证书，然后启用 SSL 模式。

回滚过程 (Solaris)

1. 如果管理控制台正在本地运行，则将其停止。
2. 关闭依赖于即将回滚的 Directory Server 实例的所有 Java ES 组件。此步骤取决于这些组件在部署体系结构中的复制方式。

应按以下顺序关闭组件：

- a. Directory Server 客户机：Access Manager、Communications Express、Messaging Server、Portal Server 等
- b. Directory Proxy Server（如果使用它访问 Directory Server）
- c. Administration Server（如果它在本地运行）
- d. Directory Server
- e. 配置目录（如果它作为单独的 Directory Server 实例在本地运行）

有关如何关闭 Java ES 组件的信息，参见其各自的管理指南。

3. 回滚 Directory Server 升级。
 - a. 确保关闭了正在回滚的 Directory Server 实例。
 - b. 删除表 4-5 中的 Directory Server 修补程序。


```
patchrm patch_ID
```
 - c. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。
 - d. 使回滚后的设置与配置目录同步。


```
/usr/sbin/directoryserver -u 5.2 sync-cds
```

将提示您输入管理员用户名和密码。
 - e. 如果不运行 Administration Server 而独立运行 Directory Server，则必须按照步骤 4 中的说明回滚部分 Administration Server 升级。

4. 回滚 Administration Server 升级。
 - a. 删除表 4-5 中的 Administration Server 修补程序。

```
patchrm patch_ID
```
 - b. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。
 - c. 使升级后的设置与配置目录同步。

```
/usr/sbin/mpsadmserver sync-cds
```

将提示您输入管理员用户名和密码。
5. 对 Directory Server 和 Administration Server 对其具有硬性升级依赖性的所有 Java ES 组件执行升级回滚。
6. 以步骤 2 中关闭组件的相反顺序重新启动所有 Java ES 组件。
 - a. 配置目录（如果它在本地作为单独的 Directory Server 实例运行）
 - b. Directory Server
 - c. Administration Server（如果它在本地运行）
 - d. Directory Proxy Server（如果使用它访问 Directory Server）
 - e. Directory Server 客户机：Access Manager、Communications Express、Messaging Server、Portal Server 等

多实例升级

第 96 页中的“发行版本 3 Directory Server 和 Administration Server 的升级”中的过程没有明确论及一些部署体系结构，在其中为了实现可用性和可伸缩性而复制了 Directory Server。这些体系结构可能包括 Directory Server 多主复制或 Directory Server 的部署，将其作为 Sun Cluster 环境中的数据服务。

本节将对这种情况下的 Directory Server 升级进行讨论。

多主复制的滚动升级

不同计算机系统上的多个 Directory Server 实例（用于多主复制部署体系结构）可以依序一次一个进行升级。每个实例在其各自主机上的升级均是在其他实例仍在运行时执行的。这种滚动升级允许目录服务在升级各个提供该项服务的 Directory Server 实例时保持联机。

作为数据服务升级 Directory Server

本节介绍如何将 Directory Server 作为 Sun Cluster 环境中的数据服务来进行升级和回滚。在将 Directory Server 作为 Sun Cluster 数据服务升级或逆向恢复之前，请考虑以下几点：

- 在执行升级或回滚操作之前备份数据。
- 按顺序而不是以并行方式修补所有群集节点上的 Directory Server 及其关联的 Administration Server。
- 所有群集节点应运行同一版本和发行版本的 Directory Server 及其关联的 Administration Server。
- 如果在故障转移模式下运行群集，请考虑从 HAStorage 升级到 HAStoragePlus。

作为 Sun Cluster 数据服务升级 Directory Server

1. 停止每个 Directory Server 实例及其关联的 Administration Server。

```
serverroot/stop-admin
serverroot/slaped-instanceName/stop-slaped
```

2. 使当前群集节点成为活动节点：

```
scswitch -z -g ldap-group -h this-node-name
```

3. 如第 96 页中的“发行版本 3 Directory Server 和 Administration Server 的升级”中所述升级当前节点上的 Directory Server。

4. 使另一个群集节点成为活动节点：

```
scswitch -z -g ldap-group -h another-node-name
```

5. 重复步骤 3 和步骤 4，直到群集中的所有节点均升级完毕为止。

作为 Sun Cluster 数据服务回滚 Directory Server

1. 停止每个 Directory Server 实例及其关联的 Administration Server。

```
serverroot/stop-admin
serverroot/slaped-instanceName/stop-slaped
```

2. 使当前群集节点成为活动节点：

```
scswitch -z -g ldap-group -h this-node-name
```

3. 如第 106 页中的“升级回滚 (Solaris)”中所述回滚当前节点上的 Directory Server。
4. 使另一个群集节点成为活动节点：

```
scswitch -z -g ldap-group -h another-node-name
```
5. 重复步骤 3 和步骤 4，直到在群集中的所有节点上都回滚了 Directory Server 为止。

从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server

将 Java ES 2004Q2（发行版本 2）Directory Server 和 Administration Server 升级到发行版本 4 的过程与将发行版本 3 Directory Server 和 Administration Server 升级到发行版本 4 的过程相同，只不过在执行升级前的任务时，还应将 Directory Server 和 Administration Server 所依赖的所有共享组件（参见第 38 页中的表 1-6）和本地驻留的产品组件都升级到发行版本 4：

第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级到发行版本 4 的说明。

要将发行版本 2 Directory Server 和 Administration Server 升级到发行版本 4，请按第 95 页中的“从 Java ES 发行版本 3 升级 Directory Server 和 Administration Server”中的说明操作，只不过要在提到发行版本 3 的地方替换为发行版本 2。

Directory Proxy Server

本章介绍如何将 Directory Proxy Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Directory Proxy Server 5.2 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 112 页中的“Directory Proxy Server 升级概述”
- 第 114 页中的“从 Java ES 发行版本 3 升级 Directory Proxy Server”
- 第 122 页中的“从 Java ES 发行版本 2 升级 Directory Proxy Server”

注

本章中的文件位置是相对于一个称为 *serverRoot* 的目录路径而指定的。在最初安装和配置 Directory Proxy Server 时，可能至少已将此路径的一部分指定为安装目录。否则，会分配一个默认值。

serverRoot 的默认值视操作系统平台而定：

- Solaris: /var/opt/mps/serverroot
 - Linux: /var/opt/sun/directory-server
-

Directory Proxy Server 升级概述

本节将在以下方面对 Directory Proxy Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4](#)
- [Java ES 发行版本 4 升级指导说明](#)
- [Directory Proxy Server 数据](#)
- [兼容性问题](#)
- [组件依赖性](#)

关于 Java ES 发行版本 4

Java ES 发行版本 4 Directory Proxy Server 仅修复了一些小错误和做出了一些小的改进，并没有新增任何功能。

Java ES 发行版本 4 升级指导说明

表 5-1 显示了所支持的升级至 Java ES 发行版本 4 的 Directory Proxy Server 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 5-1 升级到 Java ES 发行版本 4：
Sun Java System Directory Proxy Server 5.2 2005Q4 的途径

Java ES 发行版本	Directory Proxy Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Directory Proxy Server 5.2 2005Q1	直接升级： 应用修补程序并重新配置配置目录。	配置目录中数据的自动重新配置
发行版本 2	Sun Java System Directory Proxy Server 5.2 2004Q2	直接升级： 应用修补程序并重新配置配置目录。	配置目录中数据的自动重新配置
发行版本 1	Sun One Directory Proxy Server 5.2	直接升级未经验证： 但可以使用发行版本 2 的升级方法。	配置目录中数据的自动重新配置

表 5-1 升级到 Java ES 发行版本 4:
Sun Java System Directory Proxy Server 5.2 2005Q4 的途径 (续)

Java ES 发行版本	Directory Proxy Server 版本	常规方法	必需的重新配置
Java ES 以前的发行版本	Sun One Directory Proxy Server 5.2	直接升级未经验证: 但可以使用发行版本 2 的升级方法。	配置目录中数据的自动重新配置
	Sun One Directory Access Router 5.0 或 5.0 SP1	不直接升级: 首先升级到发行版本 3, 参阅 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237)。 然后从发行版本 3 升级到发行版本 4。	参阅 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237)。

Directory Proxy Server 数据

Directory Proxy Server 利用 Directory Server 来存储配置数据。数据存储在目录内部的一个特定树结构中。托管配置的 Directory Server 实例称为配置目录。

在大多数部署体系结构中, 配置目录相对于利用其来存储配置信息的其他组件而言是远程的。

下表显示了可能受 Directory Proxy Server 软件升级影响的数据类型。

表 5-2 Directory Proxy Server 数据用途

数据类型	位置	用途
Directory Proxy Server 配置数据	配置目录	Directory Proxy Server 的配置

兼容性问题

Java ES 发行版本 4 Directory Proxy Server 的接口没有任何变化，并与早期版本向后兼容。

组件依赖性

对其他 Java ES 组件的依赖性会影响升级和重新配置 Directory Proxy Server 软件的过程。Directory Proxy Server 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。Directory Proxy Server 提供对 Directory Server 的前端访问，并使用 Administration Server 来实现配置。因此，Directory Proxy Server 对 Directory Server 和 Administration Server 均具有依赖性。

从 Java ES 发行版本 3 升级 Directory Proxy Server

本节包含关于将 Directory Proxy Server 从 Java ES 2005 Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Directory Proxy Server 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Directory Proxy Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过向 Java ES 发行版本 3 的版本应用修补程序来执行升级。通过自动将配置目录与已升级软件进行同步来实现 Directory Proxy Server 的重新配置。
- **升级依赖性。** 虽然 Directory Proxy Server 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但 Java ES 发行版本 4 Directory Proxy Server 与这些共享组件在发行版本 3 中的版本兼容。因此，对于 Directory Proxy Server 到发行版本 4 的升级，这些共享组件的升级是可选的。

Directory Proxy Server 对 Directory Server 和 Administration Server 具有硬性升级依赖性。因此，应将全部三个组件一同升级到发行版本 4。

- **向后兼容性。** 发行版本 4 Directory Proxy Server 与其发行版本 3 的版本向后兼容。
- **升级回滚。** 在 Solaris 平台上，发行版本 4 升级回滚是通过删除发行版本 4 升级修补程序来实现的。但是在 Linux 平台上，则不存在回滚发行版本 4 升级的过程。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Directory Proxy Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Directory Proxy Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Directory Proxy Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Directory Proxy Server \(Solaris\)](#)
- [升级发行版本 3 Directory Proxy Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Directory Proxy Server 之前，应执行下述任务。

检验当前版本信息

可使用以下命令检验 Directory Proxy Server 的当前版本：

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

下表显示了输出结果：

表 5-3 Directory Proxy Server 版本检验输出

Java ES 发行版本	Directory Proxy Server 版本号
发行版本 2	Sun ONE Directory Proxy Server Version 5.2_Patch_2
发行版本 3	Sun ONE Directory Proxy Server Version 5.2_Patch_3
发行版本 4	Sun ONE Directory Proxy Server Version 5.2_Patch_4

升级 Directory Proxy Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。

Directory Proxy Server 对 Directory Server 和 Administration Server 具有硬性升级依赖性（即使这两个组件运行在远程计算机上），因此应该首先升级这两个组件，然后再升级 Directory Proxy Server。

Directory Proxy Server 所依赖的 Java ES 发行版本 3 共享组件的升级是可选的，但建议您执行此操作。

可按以下顺序升级 Directory Proxy Server 依赖项，并且应在升级 Directory Proxy Server 之前进行。您可以跳过任何可能已完成升级的依赖项。

1. **共享组件。** 第 47 页中的“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。

备份 Directory Server 数据

Directory Proxy Server 升级过程将修改配置目录数据。因此，在升级之前，建议您使用 Directory Server 控制台或命令行实用程序（如 db2bak）备份配置目录数据。

有关备份 Directory Server 的更多信息，参见 Sun Java System Directory Server 管理指南 (<http://docs.sun.com/doc/819-2013>)。

获取所需的配置信息和密码

Directory Proxy Server、Directory Server 和 Administration Server 必须以同一用户和组的身份运行。即，全部三个组件必须使用同一个 UID 和 GID 运行。

升级发行版本 3 Directory Proxy Server (Solaris)

本节讨论影响 Directory Proxy Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Directory Proxy Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 在升级 Directory Proxy Server 实例之前，应该首先将使用该实例的所有 Java ES 组件（如 Access Manager、Communications Express、Messaging Server、Portal Server 等）关闭。但是，许多部署体系结构都使用 Directory Proxy Server 的多个实例来提供高可用性或可伸缩性。在这种情况下，可以执行 Directory Proxy Server 的滚动升级，而无需关闭 Directory Proxy Server 客户机。
- Directory Proxy Server 的升级只能在 Administration Server 和 Directory Server 的升级之后执行，因为重新配置必须以特定顺序进行。
- 在应用修补程序时，Directory Proxy Server 必须处于关闭状态，但相关联的配置目录必须处于运行状态以执行重新配置。
- 在其中有多个 Directory Proxy Server 实例运行于单台计算机上的部署体系结构中（所有实例均对应于同一个已安装的 Directory Proxy Server 映像），升级 Directory Proxy Server 映像将升级所有实例。在此类体系结构中，每个已安装的 Directory Proxy Server 映像仅有一个 Administration Server 实例。

- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Directory Proxy Server 升级修补程序：

表 5-4 在 Solaris 上升级 Directory Proxy Server 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Directory Proxy Server	116373-18	116374-18
Directory Proxy Server 本地化版本	117017-20	117017-20

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于本地驻留在即将进行升级的计算机上的 Directory Proxy Server 实例。

1. 根据表 5-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果管理控制台正在本地运行，则将其停止。
4. 关闭依赖于即将升级的 Directory Proxy Server 实例的所有 Java ES 组件。此步骤可能取决于 Directory Proxy Server 在部署体系结构中的复制方式。

有关如何关闭 Java ES 组件的信息，参见其各自的管理指南。

5. 确保已升级 Directory Proxy Server 对其具有硬性升级依赖性的所有 Java ES 组件（参见第 116 页中的“升级 Directory Proxy Server 依赖项”）。

6. 升级 Directory Proxy Server。

a. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。

b. 应用表 5-4 中的 Directory Proxy Server 修补程序。

确保先应用 Directory Proxy Server 本地化修补程序 (117017)，然后再应用 Directory Proxy Server 基本修补程序。

```
patchadd patch_ID
```

c. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 b 中应用的修补程序 ID 的版本。

7. 重新启动 Directory Proxy Server 及所有依赖于该组件的 Java ES 组件。

重新启动 Directory Proxy Server：

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

升级发行版本 3 Directory Proxy Server (Linux)

本节讨论影响 Directory Proxy Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 操作系统的修补程序不同外，在 Linux 平台上将 Directory Proxy Server 升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 117 页中的“升级注意事项 (Solaris)”）。

下表显示了用于 Linux 操作系统的发行版本 4 的 Directory Proxy Server 升级修补程序：

表 5-5 在 Linux 上升级 Directory Proxy Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Directory Proxy Server	118096-08: sun-directory-proxy-server-5.2-13.i386.rpm
Directory Proxy Server 本地化版本	118288-11: sun-directory-proxy-server-Locale-5.2-16.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于本地驻留在即将进行升级的计算机上的 Directory Proxy Server 实例。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 5-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果管理控制台正在本地运行，则将其停止。
4. 关闭依赖于即将升级的 Directory Proxy Server 实例的所有 Java ES 组件。此步骤可能取决于 Directory Proxy Server 在部署体系结构中的复制方式。

有关如何关闭 Java ES 组件的信息，参见其各自的管理指南。

5. 确保已升级 Directory Proxy Server 对其具有硬性升级依赖性的所有 Java ES 组件（参见第 116 页中的“升级 Directory Proxy Server 依赖项”）。

6. 应用 Directory Proxy Server 的 RPM。

- a. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。

- b. 应用 RPM。

确保先应用 Directory Proxy Server 本地化 RPM，然后再应用 Directory Proxy Server 基本 RPM。

```
rpm -Fvh sun-directory-proxy-server-Locale-5.2-16.i386.rpm
```

```
rpm -Fvh sun-directory-proxy-server-5.2-13.i386.rpm
```

升级后的设置会自动与配置目录保持同步。

7. 重新启动 Directory Proxy Server 及所有依赖于该组件的 Java ES 组件。

重新启动 Directory Proxy Server:

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

检验升级

可使用以下命令检验 Directory Proxy Server 的升级是否成功：

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

参见第 116 页中的表 5-3 确认输出值。

升级后的任务

除第 118 页中的“升级过程 (Solaris)”和第 120 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Directory Proxy Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Directory Proxy Server 的回滚过程几乎就是升级至发行版本 4 的相反过程。将删除修补程序并重新同步配置目录。

需要特别注意的一点是，在应用修补程序时，会将 SSL 证书数据库升级至 cert8 格式。修补程序将备份 cert7 数据，然后将其转换为 cert8 格式。如果您随后决定回滚升级且已将新证书添加到了证书数据库中，则应手动提取这些证书，逆向恢复修补程序，然后再将证书重新添加到先前的 cert7 格式的证书数据库中。

在更改了 SSL 证书数据库之后回滚升级时，将无法在 SSL 模式中启动。要解决这个问题，请关闭 SSL 模式，重新启动 Administration Server 和 Directory Proxy Server，重新安装证书，然后启用 SSL 模式。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 如果管理控制台正在本地运行，则将其停止。
3. 关闭依赖于即将升级的 Directory Proxy Server 实例的所有 Java ES 组件。此步骤可能取决于 Directory Proxy Server 在部署体系结构中的复制方式。

有关如何关闭 Java ES 组件的信息，参见其各自的管理指南。

4. 回滚 Directory Proxy Server 升级。
 - a. 确保配置目录正在运行。

如果在本地，可能必须要将其启动；如果在远程，检查以确保其正在运行。
 - b. 删除表 5-5 中的 Directory Proxy Server 修补程序。

```
patchrm patch_ID
```
5. 对 Directory Proxy Server 对其具有硬性升级依赖性的所有 Java ES 组件执行升级回滚，特别是 Directory Server 和 Administration Server。
6. 重新启动 Directory Proxy Server 及所有依赖于该组件的 Java ES 组件。

多实例升级

在一些部署体系结构中，Directory Proxy Server 被部署在多个计算机系统上，以提供可伸缩性和提高可用性。例如，您可以在多台计算机上运行 Directory Proxy Server 组件，并使用一个负载均衡器来分配负载。

就 Directory Proxy Server 负载均衡的实例而言，可执行滚动升级，期间您可依序升级各个 Directory Proxy Server 实例而不中断服务。在升级每个 Directory Proxy Server 实例时，其他实例仍可保持运行。可按第 115 页中的“[发行版本 3 Directory Proxy Server 的升级](#)”中所述升级每个实例。

从 Java ES 发行版本 2 升级 Directory Proxy Server

升级前的任务应该包括将 Directory Proxy Server 所依赖的所有共享组件（参见第 38 页中的表 1-6）和所有本地驻留的产品组件升级至发行版本 4，除此之外，将 Java ES 2004Q2（发行版本 2）Directory Proxy Server 升级至发行版本 4 的过程与将发行版本 3 Directory Proxy Server 升级至发行版本 4 的过程相同。

第 47 页中的第 2 章“[升级 Java ES 共享组件](#)”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。

要将发行版本 2 的 Directory Proxy Server 升级至发行版本 4，可按照第 114 页中的“[从 Java ES 发行版本 3 升级 Directory Proxy Server](#)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

Web Server

本章介绍如何将 Web Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Web Server 6.1 SP5 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 124 页中的“Web Server 升级概述”
- 第 126 页中的“从 Java ES 发行版本 3 升级 Web Server”
- 第 132 页中的“从 Java ES 发行版本 2 升级 Web Server”

注 本章中的文件位置是相对于一个称为 *WebServer-base* 的目录路径而指定的。在最初安装 Web Server 时，至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

WebServer-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWwbsvr
 - Linux: /opt/sun/webserver
-

Web Server 升级概述

本节将在以下方面对 Web Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Web Server](#)
- [Web Server 升级指导说明](#)
- [Web Server 数据](#)
- [兼容性问题](#)
- [Web Server 组件依赖性](#)

关于 Java ES 发行版本 4 Web Server

Java ES 发行版本 4 的 Web Server 版本修复了许多错误，包括依赖 NSS 共享组件的安全修复。

有关详细信息，参见相应的发行说明。

Web Server 升级指导说明

表 6-1 显示了所支持的升级至 Java ES 发行版本 4 的 Web Server 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 6-1 升级到 Java ES 发行版本 4：Sun Java System Web Server 6.1 SP5 2005Q4 的途径

Java ES 发行版本	Web Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Web Server 6 2005Q1 Update 1 SP 4	直接升级： 通过应用修补程序执行。	无
发行版本 2	Sun Java System Web Server 6 2004Q2 Update 1 SP 2 Platform Edition 和 Enterprise Edition	直接升级： 通过应用修补程序执行。	无
发行版本 1	Sun ONE Web Server 6.1 (2003Q4)	直接升级未经验证： 但可通过应用修补程序执行。	无
Java ES 以前的发行版本		不直接升级。	

Web Server 数据

下表显示了可能受 Web Server 软件升级影响的数据类型。

表 6-2 Web Server 数据用途

数据类型	位置	用途
配置数据	<i>WebServer-base/https-instanceName/config/obj.conf</i> 和同一目录中的其他文件	Web Server 实例的配置

兼容性问题

Java ES 发行版本 4 Web Server 的接口没有任何变化，并与早期版本向后兼容。

Web Server 组件依赖性

Web Server 对 Java ES 共享组件具有依赖性，对其他 Java ES 组件则不然（参见第 38 页中的表 1-6）。

从 Java ES 发行版本 3 升级 Web Server

本节包含关于将 Web Server 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Web Server 的升级](#)

简介

将 Java ES 发行版本 3 Web Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 3 的版本执行升级。从 Java ES 发行版本 3 Web Server 升级到发行版本 4 无需重新配置 Web Server。
- **升级依赖性。** 虽然 Web Server 对许多 Java ES 共享组件具有依赖性（参见 [第 38 页中的表 1-6](#)），但发行版本 4 的 Web Server 仅要求将 NSS 和 NSPR 升级到发行版本 4。因此，对于 Web Server 到发行版本 4 的升级，其他共享组件的升级是可选的。
- **向后兼容性。** 发行版本 4 的 Web Server 与发行版本 3 的版本向后兼容。
- **升级回滚。** 发行版本 4 到发行版本 3 的升级回滚可通过删除升级期间所应用的修补程序实现。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Web Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Web Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Web Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Web Server \(Solaris\)](#)
- [升级发行版本 3 Web Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Web Server 之前，应执行下述任务。

检验当前版本信息

可通过用 `-version` 选项启动 Web Server 实例服务器来检验 Web Server 的当前版本：

```
WebServer-base/https-hostName.domainName/start -version
```

表 6-3 Web Server 版本检验输出

Java ES 发行版本	Web Server 版本号
发行版本 2	6.1SP2
发行版本 3	6.1SP4
发行版本 4	6.1SP5

升级 Web Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件均升级到 Java ES 发行版本 4。但是，将 Web Server 升级到发行版本 4 仅要求在升级 Web Server 之前将 NSS 和 NSPR 共享组件从发行版本 3 升级到其版本 4 的相应版本。第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 NSS 和 NSPR 或可能希望升级的其他 Java ES 共享组件升级到发行版本 4 的说明。

备份 Web Server 数据

Web Server 从发行版本 3 升级到发行版本 4 不会修改配置数据。没有必要备份当前数据。

获取所需的配置信息和密码

不需要任何与当前安装版本有关的特殊信息，但是必须以超级用户身份登录才能执行升级。

升级发行版本 3 Web Server (Solaris)

本节讨论影响 Web Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Web Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 在升级 Web Server 实例之前，应先关闭在该实例中运行的所有 J2EE 组件。
- 所有对应同一个已安装 Web Server 映像的 Web Server 实例将同时被升级。向安装的映像应用修补程序时，应关闭所有此类实例。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Web Server 升级修补程序：

表 6-4 在 Solaris 上升级 Web Server 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Web Server 核心组件 (SUNWwbsvr)	116648-17	116649-17
Web Server 本地化版本	117514-10	117515-10

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于所有这样的 Web Server 实例，它们与即将进行升级的计算机上的同一个已安装 Web Server 映像相对应。

1. 根据表 6-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止所有运行的 Web Server 和 Administration Server 实例。

```
WebServer-base/https-instanceName/stop
WebServer-base/https-admserv/stop
```

4. 将 NSS 和 NSPR 共享组件及希望升级的任何其他共享组件升级到发行版本 4（如果尚未执行此操作）。

参见第 127 页中的“升级 Web Server 依赖项”。

5. 应用表 6-4 中相应的 Web Server 修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 重新启动在步骤 3 中停止的 Web Server 实例。

升级发行版本 3 Web Server (Linux)

本节讨论影响 Web Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Web Server 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 128 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 4 的 Web Server 升级修补程序：

表 6-5 在 Linux 上升级 Web Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Web Server 核心组件	118202-09 <ul style="list-style-type: none"> • sun-websserver-6.1.5-6.i386.rpm
Web Server 本地化版本	118203-06 <ul style="list-style-type: none"> • sun-websserver-Locale-6.1.5-1.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于所有这样的 Web Server 实例，它们与即将进行升级的计算机上的同一个已安装 Web Server 映像相对应。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 6-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止所有运行的 Web Server 和 Administration Server 实例。

```
WebServer-base/https-instanceName/stop
```

```
WebServer-base/https-admserv/stop
```

4. 将 NSS 和 NSPR 共享组件及希望升级的任何其他共享组件升级到发行版本 4（如果尚未执行此操作）。

参见第 127 页中的“升级 Web Server 依赖项”。

5. 应用表 6-5 中用于 Web Server 的 RPM。

```
rpm -Fvh sun-webserver-6.1.5-6.i386.rpm
```

6. 确认升级成功:

```
rpm -q sun-webserver
```

应该会返回 RPM 的新修订版本号。

7. 重新启动在步骤 3 中停止的 Web Server 实例。

```
WebServer-base/https-admserv/start
```

```
WebServer-base/https-instanceName/start
```

检验升级

可通过用 `-version` 选项启动 Web Server 实例服务器来检验 Web Server 到发行版本 4 的升级：

```
WebServer-base/https-hostName.domainName/start -version
```

参见第 127 页中的表 6-3 确认输出值。

还可在以下日志文件中检查相应的条目：

```
WebServer-base/setup/upgrade.log
```

升级后的任务

除第 128 页中的“升级过程 (Solaris)”和第 130 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Web Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Web Server 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。修补程序会被删除。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 停止所有运行的 Web Server 和 Administration Server 实例。

```
WebServer-base/https-instancename/stop
WebServer-base/https-admserv/stop
```

3. 删除第 128 页中的表 6-4 中的修补程序。

```
patchrm patch_ID
```

4. 重新启动在步骤 2 中停止的 Web Server 实例。

从 Java ES 发行版本 2 升级 Web Server

将 Java ES 2004Q2（发行版本 2）的 Web Server 升级至发行版本 4 的过程与将发行版本 3 的 Web Server 升级至发行版本 4 的过程相同，只不过在执行升级前的任务时，还应将 Web Server 所依赖的所有共享组件（参见第 38 页中的表 1-6）升级至发行版本 4。

第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。

要将发行版本 2 的 Web Server 升级到发行版本 4，请按照第 126 页中的“从 Java ES 发行版本 3 升级 Web Server”中的说明操作，但要在提到发行版本 3 的地方用发行版本 2 代替。不过，从发行版本 2 到发行版本 4 的升级还需要修改 `obj.conf` 配置文件，但该操作会自动执行。

Message Queue

本章介绍如何将 Message Queue 软件从先前的 Java ES 版本升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Message Queue 3 Enterprise Edition 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的 Message Queue 升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 134 页中的“Message Queue 升级概述”
- 第 139 页中的“从 Java ES 发行版本 3 升级 Message Queue”
- 第 145 页中的“从 Java ES 发行版本 2 升级 Message Queue”

注 本章所使用的 Message Queue 命令相对于可执行文件的目录位置运行，具体位置取决于操作系统平台：

- Solaris: /usr/bin
 - Linux: /opt/sun/mq/bin
-

Message Queue 升级概述

本节将在以下方面对 Message Queue 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Message Queue](#)
- [Message Queue 升级指导说明](#)
- [Message Queue 数据](#)
- [兼容性问题](#)
- [Message Queue 组件依赖性](#)

关于 Java ES 发行版本 4 Message Queue

Java ES 发行版本 4 的 Message Queue 修复了一些小的代码错误，并无新增功能或增强功能。发行版本 4 本身未引入任何新的兼容性问题（参见第 136 页中的“兼容性问题”）。

Message Queue 软件包括两种版本，Platform Edition 和 Enterprise Edition，每种版本对应不同的功能集和准许的容量。Enterprise Edition 用于在企业生产环境中部署和运行消息传送应用程序。Platform Edition 主要用于开发、调试及负载测试消息传送应用程序和组件。Platform Edition 可从 Sun Web 站点免费下载，同时还与 Solaris OS 和 Java ES Application Server 平台捆绑在一起。从早期 Java ES 发行版本到发行版本 4 的升级会将任何已安装的 Platform Edition 转换成 Enterprise Edition。

Message Queue 升级指导说明

表 7-1 显示了所支持的升级至 Java ES 发行版本 4 的 Message Queue 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 7-1 升级到 Java ES 发行版本 4 Message Queue 3.6 SP3 2005Q4 的途径

Java ES 发行版本	Message Queue 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Message Queue 2005Q2 (3.6) 仅限 Enterprise Edition	直接升级： 使用 mqupgrade 脚本执行。	无
发行版本 2	Sun Java System Message Queue 2004Q2 (3.5) Platform Edition 和 Enterprise Edition	直接升级： 使用 mqupgrade 脚本执行。	在 Solaris 平台自动执行，在 Linux 平台可使用 mqmigrate 脚本。
发行版本 1	Sun Java System Message Queue 3.01 SP2 Platform Edition 和 Enterprise Edition	直接升级未经验证： 但是可使用 mqupgrade 脚本执行。	在 Solaris 平台自动执行，在 Linux 平台可使用 mqmigrate 脚本。
Java ES 以前的发行版本	Sun Java System Message Queue 3.01 SP1 和早期版本 Platform Edition 和 Enterprise Edition	不直接升级： 但可以按照 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237) 中的过程先升级到发行版本 3。 然后从发行版本 3 升级到发行版本 4。	

除了表 7-1 中说明的各 Java ES 发行版本的 Message Queue 之外，Message Queue Platform Edition 也与 Solaris 操作系统软件捆绑在一起。可由 Java ES 安装程序执行将这些 Message Queue 的捆绑版本升级到发行版本 4 Enterprise Edition。如同执行全新的安装一样，您只需选择要由安装程序安装的 Message Queue，随后安装程序软件将自动升级捆绑版本，并对 Message Queue 执行任何可能必要的重新配置。

Message Queue 数据

与其他 Java ES 组件一样，Message Queue 将利用对于任一特定升级而言可能需要迁移至升级版本的各种数据。下表显示了可能受 Message Queue 软件升级影响的数据的类型。

表 7-2 显示了 Solaris 系统中的数据位置。Linux 系统中的位置与之相似，可在 Message Queue 管理指南 (<http://docs.sun.com/doc/819-3561>) 中找到。在表 7-2 中，*instanceName* 标识与数据关联的 Message Queue 代理实例的名称。

表 7-2 Message Queue 数据用途

数据类别	位置 (在 Solaris 上)	用途
代理实例配置属性	<code>/var/imq/instances/instanceName/props/config.properties</code>	代理和相关服务配置
动态应用程序数据的持久存储	<code>/var/imq/instances/instanceName/fs350/</code> 或 JDBC 可以访问的数据存储	存储消息、目标、持久的订阅、事务和其他动态数据
管理对象 (对象存储)	所选的本地目录 或 LDAP Directory Server	用于配置客户机/代理连接的对象
安全: 用户信息库	<code>/var/imq/instances/instanceName/etc/passwd</code> 或 LDAP 目录服务器	存储用于验证和授权的用户数据
安全: 访问控制文件 (默认位置)	<code>/var/imq/instances/instanceName/etc/accesscontrol.properties</code>	设置授予用户对目标和相关功能访问权的规则
安全: 密码文件目录 (默认位置)	<code>/var/imq/instances/instanceName/etc/</code>	存储加密的密码信息
安全: 代理的密钥库文件位置	<code>/etc/imq/</code>	存储用于安全消息传送的加密证书信息

兼容性问题

发行版本 4 的 Message Queue 未在发行版本 3 的基础上增加新的不兼容性。以下一些常规的 Message Queue 兼容性问题与发行版本 3 之前的版本有关。

协议兼容性

Message Queue 依赖 Web 容器在 Message Queue 客户机和代理之间提供 HTTP 协议支持。由于协议变化，使用 Sun Java System Web Server 为 Message Queue `imqhttp.war` 应用程序提供 Web 容器时，若不同时升级 Message Queue，将无法升级 Web Server 组件（参见第 143 页和第 147 页上的“升级后的任务”）。

代理兼容性

发行版本 4 的 Message Queue 代理可以和发行版本 3 或发行版本 2 的代理相互操作，但发行版本 2 中代理属性和持久存储模式的更改会影响兼容性。

发行版本 4 的 Message Queue 可以使用发行版本 3 和发行版本 2 的数据，但对于 Linux 系统上的数据则必须首先将发行版本 2 数据迁移到发行版本 4。

更新到发行版本 4 的 Message Queue 时，请考虑以下事项：

- 可以使用早期的 Message Queue `config.properties` 文件。也可以在配置发行版本 4 的 Message Queue 代理时将它们复制到另一个位置查看其包含的属性设置。
- 首次启动代理时，会根据需要将所有持久 Message Queue 数据（消息、目标、持久订阅）自动转换成发行版本 4 的 Message Queue 数据。例如，必要时，任何现有目标都会自动转换成发行版本 4 的 Message Queue 目标，同时保持现有属性并使用新属性的默认值。
- 如果将 Message Queue 发行版本 2 代理和 Message Queue 发行版本 4 代理混装在一个群集中，主代理必须是 Message Queue 发行版本 2 代理（取较旧者），该群集将运行 Message Queue 发行版本 2 群集。

管理对象兼容性

发行版本 4 的 Message Queue 管理对象和发行版本 3 的管理对象相同。可是，相对于早期版本而言，一些发行版本 3 的管理对象经过重命名并采用了新属性而得到增强。因此，从发行版本 2 Message Queue 升级到发行版本 4 时，应考虑以下事项：

- 可以使用在发行版本 2 中创建的相同对象存储和管理对象；但最好是将管理对象迁移到发行版本 4。执行更新操作时，管理控制台 (`imqadmin`) 和 `ObjectManager` 命令行实用程序 (`imqobjmgr`) 会将发行版本 2 的管理对象转换成发行版本 4 的管理对象。
- 发行版本 4 客户机运行时将会查找并实例化发行版本 2 管理对象，然后转换它们以供发行版本 4 客户机使用。不过，这不会转换驻留在所查找的对象存储中的发行版本 2 管理对象。
- 现有的发行版本 2 客户机（应用程序和 / 或组件），即直接实例化管理对象而非查找它们的客户机，与发行版本 4 兼容。但是，如果它们要使用新的管理对象属性（有关管理对象属性的信息，参见 Message Queue 管理指南 (<http://docs.sun.com/doc/819-3561>) 的第 16 章），则需要被重写。（使用发行版本 4 重新编译发行版本 2 时，将显示在发行版本 4 中已重命名过的那些 Message Queue 发行版本 2 属性。旧名称仍然有效。）

- 启动 Java 客户机并使用命令行选项设置管理对象属性值的脚本与发行版本 4 兼容。然而，如果这些脚本要使用新的管理对象属性（有关管理对象属性的信息，参见 Message Queue 管理指南 (<http://docs.sun.com/doc/819-3561>) 的第 16 章），则需要被重写。

管理工具兼容性

由于在发行版本 3 中添加了新的命令和新的管理功能，因此发行版本 4 管理工具（管理控制台和命令行实用程序）只支持发行版本 3 和发行版本 4 代理。但是，所有发行版本 2 的命令和命令选项仍受支持。

客户机兼容性

发行版本 3 客户机与发行版本 4 的 Message Queue 完全兼容。然而，从发行版本 2 升级到发行版本 4 时，对于 Java 客户机应考虑以下兼容性问题：

- 发行版本 4 代理将支持发行版本 2 客户机（但没有新增的发行版本 4 功能）。
- 发行版本 4 Java 客户机能够连接到发行版本 2 代理（但没有新增的发行版本 4 功能）。
- C 客户机程序仅受发行版本 2、发行版本 3 或者是有试运行许可 (Platform Edition) 或 Enterprise Edition 许可的发行版本 4 代理支持。

Message Queue 组件依赖性

Message Queue 对其他 Java ES 组件的依赖性会影响升级和重新配置 Message Queue 软件的过程。例如，Message Queue 接口或功能上的变化可能要求 Message Queue 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Message Queue 对以下 Java ES 组件具有依赖性：

- **共享组件。** Message Queue 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Directory Server（可选）。** 如果您想配置 Message Queue 以在 LDAP 目录中存储管理对象和 / 或用户数据，而不是本地存储，则可使用 Directory Server 来实现此目的。
- **Web 容器（可选）。** 如果需要客户机和代理之间的 HTTP 消息传送，则 Message Queue 需要 Java ES Web Server、Java ES Application Server 或第三方 Web 容器提供 Web 容器支持。

从 Java ES 发行版本 3 升级 Message Queue

本节包含关于将 Message Queue 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 发行版本 4 的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Message Queue 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Message Queue 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 使用 `mqupgrade` 脚本执行升级，该脚本会用新软件包替代先前的软件包并从发行版本 3 自动迁移配置数据。
- **升级依赖性。** 虽然 Message Queue 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Message Queue 与发行版本 3 中的所有这些组件兼容。因此，对于 Message Queue 到发行版本 4 的升级，这些共享组件的升级是可选的。

此外，如第 138 页中的“[Message Queue 组件依赖性](#)”中所述，发行版本 4 的 Message Queue 选择性地依赖于 Directory Server 和 Web Server（或 Application Server）。但是，这些是软性升级依赖性；对于 Message Queue 到发行版本 4 的升级，这些组件的升级是可选的。

- **向后兼容性。** 发行版本 4 的 Message Queue 与发行版本 3 完全兼容（参见第 136 页中的“[兼容性问题](#)”）。
- **升级回滚。** 没有用于将 Message Queue 升级回滚至发行版本 3 的实用程序。您必须删除已升级的组件，手动恢复先前的版本和配置数据。
- **平台问题。** 对于 Solaris 和 Linux 操作系统，升级 Message Queue 的常规方法是相同的。随后的过程将指明特定于平台的命令或适当的文件位置。

发行版本 3 Message Queue 的升级

本节介绍如何执行 Message Queue 从 Java ES 发行版本 3 到 Java ES 发行版本 4 的升级：

- [升级前的任务](#)
- [升级发行版本 3 Message Queue](#)
- [检验 Message Queue 升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 Message Queue 之前，请执行以下各节所述的过程。在某个过程取决于平台特定命令之处，任务将指明所适用的操作系统。

检验当前版本信息（Solaris 系统）

可通过使用 `-version` 选项启动 Message Queue 代理，确定系统上安装的 Message Queue 的版本和版 (edition)：

```
imqbrokerd -version
```

表 7-3 Message Queue 版本检验输出

Java ES 发行版本	Message Queue 版本号
发行版本 2	Sun Java (tm) System Message Queue 3 2004Q2 Version: 3.5
发行版本 3	Sun Java (tm) System Message Queue 3 2005Q1 Version: 3.6
发行版本 4	Sun Java (tm) System Message Queue 3 2005Q4 Version: 3.6 SP3

升级 Message Queue 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但由于 Message Queue 并不要求一定升级它所依赖的 Java ES 发行版本 3 组件，所以此项任务是可选的。

但是，如果选择升级 Message Queue 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Message Queue 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server（可选）。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件（可选）。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。

备份 Message Queue

没有用于使 Message Queue 回滚至其先前状态的脚本。由于发行版本 4 数据与版本 3 数据兼容，因此无需备份配置数据。此外，若是要将发行版本 4 的 Message Queue 回滚至发行版本 3，您可使用发行版本 3 安装程序，因此也无需备份安装的映像。

升级发行版本 3 Message Queue

将 Message Queue 软件升级至 Java ES 发行版本 4 要利用 mqupgrade 脚本，该脚本将安装包包含表 7-4 中所示修补程序的最新软件包。

表 7-4 升级 Message Queue 的修补程序¹

组件	SPARC	X86	Linux
	Solaris 8、9 和 10	Solaris 9 和 10	
Message Queue Core	119132-06	119133-06	119136-06
Message Queue C-runtime	119134-04	119135-04	
Message Queue jmsclient & xmlclient			119137-04
Message Queue 本地化版本	119691-03	119692-03	119693-03

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程由下列步骤组成：

1. 停止运行任何 Message Queue 客户机应用程序。

如果正在 Application Server 环境中使用 Message Queue，则还要关闭 Application Server。

2. 停止运行任何代理。将提示您输入管理员用户名和密码。

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

3. 如果不想保留动态数据、Message Queue 平面文件用户信息库以及与每个代理实例关联的 Message Queue 访问控制文件，请使用以下命令删除。

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

否则，将会保留动态数据和配置信息供发行版本 4 的 Message Queue 使用。

4. 以 Root 身份登录。

```
su -
```

5. 将目录更改为 Java ES 分发包 Tools 目录的位置。

在 Solaris SPARC 上：

```
cd Solaris_sparc/Product/message_queue/Tools
```

在 Solaris x86 上：

```
cd Solaris_x86/Product/message_queue/Tools
```

在 Linux x86 上：

```
cd Linux_x86/Product/message_queue/Tools
```

6. 运行 mqupgrade 脚本。

- a. 启动脚本：

```
./mqupgrade
```

mqupgrade 脚本会列出已安装的 Message Queue 组件。

- b. 输入 y（是）以升级 Message Queue 组件。

mqupgrade 脚本会检测并列出已经安装的本地化文件。

如果不想升级 Message Queue 组件，请输入 n（否）。mqupgrade 脚本将退出，不升级 Message Queue 组件。

- c. 收到提示时，输入 y（是）以升级本地化文件。

mqupgrade 会发送输出到以下位置的日志文件：

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

检验 Message Queue 升级

完成升级过程后，通过使用 `-version` 选项启动 Message Queue 代理，检验升级是否成功。

```
imqbrokerd -version
```

该命令会返回 Java ES 版本号和特定于 Message Queue 的版本号。

升级后的任务

如果升级了 Web 容器并要使用 Message Queue HTTP 隧道 servlet，您可能需要在新的 Web 容器中重新部署它。要不然，由于在发行版本 3 和发行版本 4 中的 HTTP 隧道 servlet 并没有变化，将 Message Queue 升级到发行版本 4 之后，您就无需重新进行部署。有关 HTTP 支持的更多信息，参见 Message Queue 管理指南 (<http://docs.sun.com/doc/819-3561>)。

升级回滚

尚未提供将 Message Queue 回滚到升级前状态的脚本。该过程必须按照以下步骤手动执行：

1. 停止运行任何 Message Queue 客户机应用程序。
2. 停止运行任何代理。将提示您输入管理员用户名和密码。

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

3. 如果想删除动态数据、Message Queue 平面文件用户信息库以及与每个代理实例关联的 Message Queue 访问控制文件，请使用以下命令删除。

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

4. 以 root 身份登录，或成为超级用户。

```
su -
```

5. 使用以下命令获取安装的 Message Queue 软件包列表：

Solaris:

```
pkginfo | grep -i "message queue"
```

Linux:

```
rpm -qa | grep mq
```

6. 使用以下命令删除 Message Queue 软件包:

Solaris:

```
pkgrm packageName
```

其中, *packageName* 表示任意的 Message Queue 软件包。要删除多个软件包, 请用空格分隔各软件包名称。

Linux:

```
rpm -e --nodeps RPMName
```

其中, *RPMName* 表示任意的 Message Queue rpm 组件。要删除多个组件, 请用空格分隔各 RPM 名称。

由于其他产品可能要使用 Message Queue 软件包, 因此删除它们时要慎重。pkgrm 命令会在删除软件包之前警告您任何依赖该软件包的产品。收到提示时, 键入 **y** (是) 确认删除请求。

7. 键入 “q” 退出。
8. 退出 root shell。
9. 重新安装发行版本 3 的 Message Queue。

使用 Java ES 发行版本 3 安装程序。发行版本 4 的 Message Queue 数据将正常工作。

多实例升级

要升级一个 Message Queue 群集, 且多个代理在此群集中进行交互以提供可伸缩的消息服务, 您可以进行滚动升级, 从而在每个 Message Queue 实例从发行版本 3 升级到发行版本 4 时, 该群集仍保持在线。执行群集升级时, 务必牢记以下两点:

- 因升级而关闭代理时, 其存储的持久性消息要等到重新启动该代理后才可用。
- 主代理应在最后升级。

若不然, 可直接执行以下过程: 每次关闭、升级并重新启动一个代理, 直到全部代理完成升级。

从 Java ES 发行版本 2 升级 Message Queue

将 Java ES 2004Q2（发行版本 2）的 Message Queue 升级至发行版本 4 的过程与将发行版本 3 的 Message Queue 升级至发行版本 4 的过程几乎完全相同（参见第 139 页中的“从 Java ES 发行版本 3 升级 Message Queue”）。但对于从发行版本 2 升级，在操作系统平台之间略有差异。

此外，从发行版本 2 升级至发行版本 4，升级前的任务应包括升级 Message Queue 所依赖的所有共享组件（参见第 38 页中的表 1-6）。

第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级到发行版本 4 的说明。

升级发行版本 2 的 Message Queue (Solaris)

按照第 139 页中的“从 Java ES 发行版本 3 升级 Message Queue”中的说明操作，但要在提到发行版本 3 的地方用发行版本 2 代替。

升级发行版本 2 的 Message Queue (Linux)

在 Linux 系统上，从发行版本 2 到发行版本 4 的升级包括数据迁移步骤（即将代理实例数据迁移到相应的发行版本 4 位置），而从发行版本 3 更新到发行版本 4 则无需该步骤。如果要在升级到发行版本 4 时保留发行版本 2 数据，Message Queue 提供了一个迁移工具 (mqmigrate) 来执行此迁移。

升级过程

要从发行版本 2 升级到发行版本 3，请按照与第 139 页中的“从 Java ES 发行版本 3 升级 Message Queue”中相同的说明操作，只不过需要在运行 mqupgrade 脚本之前先运行 mqmigrate 脚本，这一点在以下过程中有详细说明。

1. 停止运行任何 Message Queue 客户机应用程序。
2. 停止运行任何代理。将提示您输入管理员用户名和密码。

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

3. 如果不想保留动态数据、Message Queue 平面文件用户信息库以及与每个代理实例关联的 Message Queue 访问控制文件，请使用以下命令删除。

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

否则，将会保留动态数据和配置信息供发行版本 4 的 Message Queue 使用。

4. 以 root 身份登录，或成为超级用户。

```
su -
```

5. 将目录更改为 Java ES 分发包 Tools 目录的位置。

```
cd Linux_x86/Product/message_queue/Tools
```

6. 使用以下命令迁移代理程序实例数据：

```
./mqmigrate
```

mqmigrate 脚本会将发行版本 2 代理实例配置数据移动到相应的发行版本 4 位置。

7. 运行 mqupgrade 脚本。

- a. 启动脚本：

```
./mqupgrade
```

mqupgrade 脚本会列出已安装的 Message Queue 组件。

- b. 输入 y（是）以升级 Message Queue 组件。

mqupgrade 脚本会检测并列出已经安装的本地化文件。

如果不想升级 Message Queue 组件，请输入 n（否）。mqupgrade 脚本将退出，不升级 Message Queue 组件。

- c. 收到提示时，输入 y（是）以升级本地化文件。

mqupgrade 会发送输出到以下位置的日志文件：

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

安装兼容性软件包

如果您拥有取决于发行版本 4 安装文件位置的脚本或您的发行版本 2 客户机应用程序包含这样的脚本，就需要安装 sun-mq-compat 软件包，其中包含从发行版本 2 文件位置到发行版本 4 文件位置的符号链接。

sun-mq-compat 软件包位于以下解压缩 Java ES 的位置。

```
Linux_x86/Product/message_queue/Packages
```

执行以下步骤，安装 sun-mq-compat 软件包：

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 从 Packages 目录输入以下命令：

```
rpm -ivh --nodeps sun-mq-compat-3.6-RelNo.i386.rpm
```

升级后的任务

如果当前是使用 HTTP 隧道 servlet 提供 HTTP 连接服务支持，则 Message Queue 从发行版本 2 升级到发行版本 4 时便已升级了该 servlet。这便要求您在 Message Queue 升级到发行版本 4 之后重新部署该 servlet。有关 HTTP 支持的更多信息，参见 Message Queue 管理指南 (<http://docs.sun.com/doc/819-3561>)。

使用管理控制台 (imqadmin) 和 / 或 ObjectManager 命令行实用程序 (imqobjmgr)，将发行版本 2 的管理对象迁移到发行版本 4 以执行更新操作。

High Availability Session Store

本章介绍如何将 High Availability Session Store 升级到 Java ES 2005Q4（版本 4）：High Availability Session Store (HADB) 4.4.2。

本章在介绍 Java ES 发行版本 4 所支持的各种升级途径之前概述一些升级问题。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 150 页中的“HADB 升级概述”
- 第 152 页中的“从 Java ES 发行版本 3 升级 HADB”

注

本章中的文件位置是相对于一个称为 *HADB-base* 的目录路径而指定的。在最初安装 HADB 时，可能至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

HADB-base 的值与 Application Server 目录结构有如下关系：

AppServer8-base/hadb/version_number

HADB-base 的默认值取决于 *AppServer8-base* 的默认值，从而视操作系统平台而定：

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

HADB 升级概述

本节将在以下方面对 HADB 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 HADB](#)
- [HADB 升级指导说明](#)
- [HADB 数据](#)
- [兼容性问题](#)
- [HADB 依赖性](#)

关于 Java ES 发行版本 4 HADB

Java ES 发行版本 4 的 HADB 修复了 Java ES 2005Q1（发行版本 3）的一些错误。

HADB 升级指导说明

[表 8-1](#) 显示了所支持的升级至 Java ES 发行版本 4 的 HADB 升级途径。此表既适用于 Solaris 操作系统，也适用于 Linux 操作系统。

表 8-1 升级到 Java ES 发行版本 4: HADB 4.4.2 (2005Q4) 的途径

Java ES 发行版本	HADB 版本	常规方法	必需的重新配置
发行版本 3	HADB 4.4.1 (2005Q1)	直接升级： 联机升级和脱机升级均可用。	无
发行版本 2	HADB 4.4.0-14 (2004Q2)	不支持升级。	无
发行版本 1	不可用	无升级。	无
Java ES 以前的发行版本	不可用	无升级。	无

HADB 数据

下表显示了可能受 HADB 软件升级影响的数据类型。

表 8-2 HADB 数据用途

数据类型	位置	用途
动态应用程序数据	<code>/var/opt/SUNWhadb</code> 和 <code>/etc/opt/SUNWhadb</code>	高可用性会话存储器和配置信息。

兼容性问题

随 Java ES 发行版本 4 提供的 HADB 与随 Java ES 发行版本 3 提供的 HADB 向后兼容。

HADB 依赖性

随 Java ES 发行版本 4 提供的 HADB 需要 Java™ 2 Platform, Standard Edition (J2SE™) 版本 5.0 或更新版本。

从 Java ES 发行版本 3 升级 HADB

本节包含关于将 HADB 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 HADB 的升级](#)

简介

将 Java ES 发行版本 3 的 HADB 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 升级包括移除 Java ES 发行版本 3 的 HADB 软件包和添加 Java ES 发行版本 4 软件包。有以下两种升级方法可用：
 - **联机升级。** 使用联机升级可避免 HADB 服务中断。
 - **脱机升级。** 如果在用更新版本替换 HADB 软件包时可中断 HADB 服务，则使用脱机升级。
- **升级依赖性。** HADB 要求 J2SE 的 5.0 版本或更新版本。
- **向后兼容性。** 随 Java ES 发行版本 4 提供的 HADB 与随 Java ES 发行版本 3 提供的 HADB 向后兼容。
- **升级回滚。** 通过恢复发行版本 3 版本（该版本在升级到发行版本 4 时在单独的目录中保持原状）来实现从 Java ES 发行版本 4 到 Java ES 发行版本 3 的升级回滚。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 HADB 的常规方法相同。

发行版本 3 HADB 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 HADB 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。其中涵盖以下主题：

- [升级前的任务](#)
- [升级发行版本 3 HADB](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 HADB 之前，应执行下述任务。

检验当前版本信息

可使用标准版检查实用程序检验 HADB 的当前版本。例如：

Solaris:

```
pkgparam -v SUNWhadba
```

Linux:

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

表 8-3 HADB 版本检验输出

Java ES 发行版本	HADB 版本号
发行版本 2	VERSION=4.4.0,REV=14 SUNW_PRODVERS=4.4.0
发行版本 3	VERSION=4.4.1,REV=7 SUNW_PRODVERS=4.4.1
发行版本 4	VERSION=4.4.2,REV=7 SUNW_PRODVERS=4.4.2

升级 HADB 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。HADB 升级到发行版本 4 将依赖 J2SE 5.0 或更新版本。

备份目录数据

HADB 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 本身并不会修改 HADB 动态数据。但是，可以备份 Java ES 发行版本 3 软件包以防您需要回滚升级。

获取所需的配置信息和密码

HADB 升级要求您知道超级用户密码。

升级发行版本 3 HADB

本节讨论影响 HADB 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项

将 HADB 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 您需要根据生产要求确定联机升级或脱机升级哪一个更合适。
- 下表显示了用于 Solaris 和 Linux 平台的 Java ES 发行版本 4 升级软件包。Solaris 软件包按其安装顺序列出。

表 8-4 用于在 Solaris 平台升级 HADB 的软件包版本

Solaris 软件包	Linux 软件包
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

HADB 的联机升级

仅当从 Java ES 发行版本 3 升级时可使用 HADB 的联机升级。

执行联机升级 HADB 时，应首先在群集中要升级的每台服务器上安装 HADB。每台服务器首先从早期安装的 HADB 中注销，然后注册到新安装的 HADB 版本中。

有关执行联机升级的详细信息，参见《Sun Java System Application Server Enterprise Edition 8.1 2005Q2 High Availability Administration Guide》(<http://docs.sun.com/doc/819-2555/6n4r9qo7n?a=view>) 中的以下小节。

HADB 的脱机升级

从 Java ES 发行版本 3 升级时可使用 HADB 的脱机升级。

要执行脱机升级，请关闭 HADB 服务，使用 Java ES 发行版本 4 发行包中提供的更新版本替换现有 HADB 软件包，如第 154 页中的表 8-4 中所示。

请使用第 63 页中的“在 Solaris 平台升级软件包”和第 66 页中的“在 Linux 平台上升级软件包”中介绍的过程。

检验升级

联机升级完成后，使用以下过程检验升级。在核实升级成功之后，可删除旧安装软件包。

要检验当前运行的进程是否在使用已升级的 HADB 服务，可执行以下步骤。

1. 对所有正在运行的 HADB 服务，发出以下一个命令：

```
HADB-base/bin/ma -v
HADB-base/bin/hadbm -v
```

例如，在 Solaris 8 平台上：

```
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. 通过对名为 ExampleDB 的数据库发出以下示例中的命令，检查该数据库是否正在运行：

```
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

所列节点的所有 HADB 服务都应处于“running”状态。

3. 通过对名为 ExampleDB 的数据库发出以下示例中的命令，检验使用 HADB 的所有产品是否正在使用新的 HADB 路径：

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

```
Attribute      Value  
PackageName    V4.4.2.7
```

以上命令会显示 HADB 的当前版本。要获取详细列表，请发出以下命令：

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

升级后的任务

除第 152 页中的“从 Java ES 发行版本 3 升级 HADB”中所述的步骤之外，没有任何升级后的任务。

升级回滚

要回滚对 HADB 的升级，请将您安装的 HADB 软件包的较新版本替换为先前安装的版本。使用与第 156 页中的“升级回滚”中所述相同的过程。不需要更改任何数据或配置文件。

Application Server

本章介绍如何将 Application Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Application Server Enterprise Edition 8.1 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 158 页中的“Application Server 升级概述”
- 第 161 页中的“从 Java ES 发行版本 3 升级 Application Server”
- 第 169 页中的“从 Java ES 发行版本 2 升级 Application Server”

注 本章中的文件位置是相对于若干目录路径指定的，分别称为 *AppServer8-base* 和 *AppServer8Config-base* (Application Server 8.1) 以及 *AppServer7-base* 和 *AppServer7Config-base* (Application Server 7)。在安装 Application Server 时，可能至少已将这部分路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

下表说明了这些目录路径的默认值。

路径名	Solaris OS	Linux OS
<i>AppServer8-base</i>	/opt/SUNWappserver/appserver	/opt/sun/appserver
<i>AppServer8Config-base</i>	/var/opt/SUNWappserver	/var/opt/sun/appserver
<i>AppServer7-base</i>	/opt/SUNWappserver7	/opt/SUNWappserver7
<i>AppServer7Config-base</i>	/var/opt/SUNWappserver7	/var/opt/SUNWappserver7

Application Server 升级概述

本节将在以下方面对 Application Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Application Server](#)
- [Application Server 升级指导说明](#)
- [Application Server 数据](#)
- [兼容性问题](#)
- [Application Server 组件依赖性](#)

关于 Java ES 发行版本 4 Application Server

Java ES 发行版本 4 Application Server 修复了发行版本 3 版本的一些选定的错误。就功能而言，发行版本 4 Application Server 与发行版本 3 相同。

Application Server 升级指导说明

表 9-1 显示了所支持的升级至 Java ES 发行版本 4 的 Application Server 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 9-1 升级到 Java ES 发行版本 4：
Sun Java System Application Server Enterprise Edition 8.1 2005Q4 的途径

Java ES 发行版本	Application Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Application Server Enterprise Edition 8.1 2005Q1	直接升级： 通过应用修补程序执行。	无
发行版本 2	Sun Java System Application Server 7.0 Upgrade 3 (2004Q2) Platform Edition 和 Enterprise Edition	直接升级： 使用 Java ES 安装程序，然后使用重新配置实用程序。	环境变量和其他配置数据。 需将 J2EE 组件和应用程序迁移到新的 Application Server 环境并重新部署。

表 9-1 升级到 Java ES 发行版本 4:
Sun Java System Application Server Enterprise Edition 8.1 2005Q4 的途径 (续)

Java ES 发行版本	Application Server 版本	常规方法	必需的重新配置
发行版本 1	Sun ONE Application Server 7.0 Upgrade 1 (2003Q4) Platform Edition 和 Enterprise Edition	直接升级未经验证: 但可以使用 Java ES 安装程序, 然后使用重新配置实用程序。	环境变量和其他配置数据。 需将 J2EE 组件和应用程序迁移到新的 Application Server 环境并重新部署。
Java ES 以前的发行版本		不直接升级: 但可以按照 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237) 中的过程首先升级到发行版本 3。 然后从发行版本 3 升级到发行版本 4。	

除了表 9-1 中说明的各 Java ES 发行版本的 Application Server 之外, Application Server Platform Edition 也与 Solaris 操作系统软件捆绑在一起。可由 Java ES 安装程序执行将这些 Application Server 的捆绑版本升级到发行版本 4 Enterprise Edition。如同执行全新的安装一样, 您只需选择要由安装程序安装的 Application Server, 随后安装程序软件将自动升级捆绑版本, 并对 Application Server 执行任何可能必要的重新配置。

Application Server 数据

下表显示了可能受 Application Server 软件升级影响的数据类型。

表 9-2 Application Server 数据用途

数据类型	位置	用途
环境变量	<i>AppServer8-base/config/asenv</i>	全局变量
配置数据	发行版本 3 和 4: <i>domain.xml</i> 和 <i>server.policy</i> 文件, 位于 <i>AppServer8Config-base/domains/domainName/config</i> 发行版本 2: <i>server.xml</i> 和 <i>server.policy</i> 文件, 位于 <i>AppServer7Config-base/domains/domainName/instanceName/config</i>	Application Server 实例的配置

表 9-2 Application Server 数据用途

数据类型	位置	用途
部署数据	发行版本 3 和 4: <i>AppServer8Config-base/domains/domainName/ applications</i> 发行版本 2: <i>AppServer7Config-base/domains/domainName/ instanceName/applications</i>	特定 J2EE 组件和应用程序的 J2EE 容器的配置。

兼容性问题

发行版本 4 Application Server 相对于发行版本 3 没有任何接口变化。但是，相对于发行版本 2 则接口变化极大，从而使得发行版本 4 与发行版本 2 不兼容。

Application Server 组件依赖性

Application Server 对其他 Java ES 组件的依赖性可能影响升级和重新配置 Application Server 软件的过程。例如 Application Server 接口或功能上的变化可能需要 Application Server 所依赖的组件的升级版本。是否需要升级此类组件取决于具体升级途径。

Application Server 对以下 Java ES 组件具有依赖性：

- **共享组件。** Application Server 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Message Queue。** Application Server 依赖 Message Queue 提供符合 J2EE Java 消息服务标准的异步消息传送支持。
- **Web 容器（可选）。** Application Server 因其可选的负载平衡插件而依赖于 Web 容器服务。该项支持可通过 Java ES Web Server 或第三方 Web 容器（如 Apache Web Server 和 Microsoft IIS）来提供。
- **High Availability Session Store（可选）。** Application Server 依赖 High Availability Session Store 维护支持实例间故障转移所需的会话状态信息。

从 Java ES 发行版本 3 升级 Application Server

本节包含关于将 Application Server 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Application Server 的升级](#)

简介

将 Java ES 发行版本 3 Application Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 3 的版本执行升级。从发行版本 3 Application Server 升级到发行版本 4 不需要重新配置或迁移 J2EE 组件。
- **升级依赖性。** 虽然 Application Server 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Application Server 与发行版本 3 中的所有这些组件兼容。因此，对于 Application Server 到发行版本 4 的升级，这些共享组件的升级是可选的。

此外，发行版本 4 的 Application Server 依赖发行版本 4 的 Message Queue 以及选择性地依赖 Java ES Web Server 或第三方 Web 容器，如第 160 页中的“[Application Server 组件依赖性](#)”所述。然而，这些是软性升级依赖性，在将 Application Server 升级到发行版本 4 时并不强制升级这些组件。

发行版本 4 的 Application Server 还选择性地依赖于 High Availability Session Store。如果 Application Server 要使用 High Availability Session Store，则应将后者升级到发行版本 4。注意，升级 High Availability Session Store 时会自动将 J2SE 共享组件升级到发行版本 4。

- **向后兼容性。** 发行版本 4 的 Application Server 与发行版本 3 的版本向后兼容。
- **升级回滚。** 发行版本 4 到发行版本 3 的升级回滚可通过删除升级期间所应用的修补程序实现。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Application Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Application Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Application Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Application Server \(Solaris\)](#)
- [升级发行版本 3 Application Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Application Server 之前，应执行下述任务。

检验当前版本信息

可通过输入以下命令检验 Application Server 的当前版本：

```
AppServer8-base/bin/asadmin version --verbose
```

表 9-3 Application Server 版本检验输出

Java ES 发行版本	Application Server 版本号
发行版本 2	Sun ONE Application Server 7.0.0_03c
发行版本 3	Sun Java Enterprise System Application Server Enterprise Edition 8.0.0_01
发行版本 4	Sun Java Enterprise System Application Server Enterprise Edition

升级 Application Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但由于 Application Server 到版本 4 的升级不要求升级其他发行版本 3 的组件，所以此项任务是可选的。如果您选择升级 Application Server 所依赖的组件，则通常会按以下顺序升级这些组件：

1. **共享组件。** 第 47 页中的“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Message Queue。** 第 133 页中的第 7 章“Message Queue”中提供了将 Message Queue 升级至发行版本 4 的说明。
3. **Web 容器软件（可选）。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
4. **High Availability Session Store（可选）。** 第 149 页中的第 8 章“High Availability Session Store”中提供了升级 High Availability Session Store 的说明。

备份 Application Server 数据

Application Server 从发行版本 3 升级到发行版本 4 不会修改配置数据。没有必要备份当前数据。

获取所需的配置信息和密码

您应当知道当前所安装版本的 Application Server 管理员用户 ID 和密码。

升级发行版本 3 Application Server (Solaris)

本节讨论影响 Application Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Application Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 升级 Application Server 实例之前，应先关闭该实例中正在运行的所有 J2EE 组件。但是，如果负载平衡提供了高可用性或可伸缩性，则可以放宽此项要求。
- 将修补程序应用于安装的 Application Server 映像时，必须关闭在单台计算机上运行的所有 Application Server 实例（与同一已安装 Application Server 映像相对应的所有实例）。
- 在多节点部署中，需要对托管 Application Server 实例的每个节点或计算机执行升级过程。

- 下表显示了用于 Solaris 操作系统的发行版本 4 Application Server 升级修补程序:

表 9-4 在 Solaris 上升级 Application Server 的修补程序¹

说明	SPARC Solaris 8、9 和 10	X86 Solaris 9 和 10
Application Server	119166-10	119167-10
Application Server 本地化版本	119024-10	119025-10

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于本地驻留在即将进行升级的计算机上的 Application Server 实例。

1. 根据表 9-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 关闭要升级的 Application Server 实例中正在运行的所有 J2EE 组件。
3. 关闭要升级的计算机上的所有 Application Server 实例。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 应用表 9-4 中相应的 Application Server 修补程序。

```
patchadd patch_ID
```

5. 修改 asant 脚本。

- a. 将现有的 asant 脚本重命名为 asant.bak。

该脚本在如下位置:

```
AppServer8-base/bin/asant
```

- b. 将 asant.template 文件从

```
AppServer8-base/lib/install/templates/ee
```

复制到

```
AppServer8-base/bin/asant
```

- c. 编辑该脚本。

将 `%CONFIG_HOME%` 标记替换为 `AppServer8-base/config`。

- d. 如果在原始脚本文件 (`asant.bak`) 中做了任何手动更改，请将这些更改合并到新的 `asant` 脚本中。

6. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在 [步骤 4](#) 中应用的修补程序 ID 的版本。

7. 重新启动 Application Server 实例。

```
AppServer8-base/bin/asadmin start-domain domainName
```

升级发行版本 3 Application Server (Linux)

本节讨论影响 Application Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Application Server 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 163 页中的“[升级注意事项 \(Solaris\)](#)”）。

下表显示了用于 Linux 操作系统的发行版本 4 的 Application Server 升级修补程序：

表 9-5 在 Linux 上升级 Application Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Application Server	119168-10 sun-asac-8.1.2-10.i386.rpm、sun-asacee-8.1.2-10.i386.rpm sun-ascml-8.1.2-10.i386.rpm、sun-ascmn-8.1.2-10.i386.rpm sun-ascmnse-8.1.2-10.i386.rpm、sun-asdb-8.1.2-10.i386.rpm sun-asdem-8.1.2-10.i386.rpm、sun-asdemdb-8.1.2-10.i386.rpm sun-ashdm-8.1.2-10.i386.rpm sun-asJdbcDrivers-8.1.2-10.i386.rpm sun-asjdoc-8.1.2-10.i386.rpm、sun-aslb-8.1.2-10.i386.rpm sun-asman-8.1.2-10.i386.rpm、sun-asmanee-8.1.2-10.i386.rpm sun-asu-8.1.2-10.i386.rpm、sun-asuee-8.1.2-10.i386.rpm sun-asut-8.1.2-10.i386.rpm、sun-aswbcr-8.1.2-10.i386.rpm

表 9-5 在 Linux 上升级 Application Server 的修补程序¹ (续)

说明	修补程序 ID 和 RPM 名称
Application Server 本地 化版本	119026-10 sun-asacee-Locale-8.1.1-51.i386.rpm sun-ascmse-Locale-8.1.1-51.i386.rpm sun-asu-Locale-8.1.1-51.i386.rpm sun-asuee-Locale-8.1.1-51.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于本地驻留在即将进行升级的计算机上的 Application Server 实例。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 9-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 关闭要升级的 Application Server 实例中正在运行的所有 J2EE 组件。
3. 关闭要升级的计算机上的所有 Application Server 实例。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 备份以下文件:

- o AppServer8-base/pointbase/tools/serveroption 下的所有文件
- o AppServer8-base/samples/common.properties

5. 应用表 9-5 中用于 Application Server 的 RPM。

```
rpm -Fvh sun-asmodule-8.1.2-10.i386.rpm
rpm -Fvh sun-asmodule-Locale-8.1.1-51.i386.rpm
```

6. 将步骤 4 中备份的文件恢复到它们的原始位置：
 - o *AppServer8-base/pointbase/tools/serveroption* 下的所有文件
 - o *AppServer8-base/samples/common.properties*
7. 修改 asant 脚本。
 - a. 将现有的 asant 脚本重命名为 asant.bak。
该脚本在如下位置：
AppServer8-base/bin/asant
 - b. 将 asant.template 文件从
AppServer8-base/lib/install/templates/ee
复制到
AppServer8-base/bin/asant
 - c. 编辑该脚本。
将 %CONFIG_HOME% 标记替换为 *AppServer8-base/config*。
 - d. 如果在原始脚本文件 (asant.bak) 中做了任何手动更改，请将这些更改合并到新的 asant 脚本中。
8. 确认修补程序升级成功：

```
rpm -qa | grep sun-as
```


应返回 RPM 的新版本号。
9. 重新启动 Application Server 实例。
AppServer8-base/bin/asadmin start-domain domainName

检验升级

可以使用以下命令检验是否已经正确应用了修补程序：

```
AppServer8-base/bin/asadmin version --verbose
```

参见第 162 页中的表 9-3 确认输出值。

升级后的任务

除第 164 页中的“升级过程 (Solaris)”和第 166 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Application Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Application Server 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。修补程序会被删除。

回滚过程 (Solaris)

1. 关闭要升级的 Application Server 实例中正在运行的所有 J2EE 组件。
2. 关闭要升级的 Application Server 实例。
3. 删除表 9-4 中的修补程序。

```
patchrm patch_ID
```

4. 重新启动 Application Server 实例。

从 Java ES 发行版本 2 升级 Application Server

本节包含关于将 Application Server 从 Java ES 发行版本 2 升级到 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- 简介
- 发行版本 2 Application Server 的升级

简介

将 Java ES 发行版本 2 Application Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过使用 Java ES 安装程序并选择以后再配置选项安装发行版本 4 的 Application Server 来执行升级。随后使用 `asupgrade` 实用程序实现重新配置。Application Server 升级之后，必须将发行版本 2 的 J2EE 组件和应用程序迁移至发行版本 4。
- **升级依赖性。** 从发行版本 2 升级计算机上的任何 Java ES 组件都需要升级由该计算机托管的所有其他 Java ES 组件；Java ES 组件从发行版本 2 选择性升级到发行版本 4 则不受支持。具体而言，需升级 Application Server 所用的全部 Java ES 共享组件。如果 Message Queue 驻留在同一台计算机上，则也必须将其升级。如果 Web Server 要用于负载平衡，同样必须将其升级。

如果 Application Server 要使用 High Availability Session Store，则应将后者升级到发行版本 4。但由于 High Availability Session Store 已经集成到发行版本 2 的 Application Server 中，故无法单独将其升级到发行版本 4。

- **向后兼容性。** 发行版本 4 的 Application Server 与发行版本 2 版本不能向后兼容。需迁移 J2EE 组件和应用程序以在发行版本 4 的 Application Server 环境中运行。
- **升级回滚。** 只需恢复到发行版本 2 的安装即可实现从发行版本 4 到发行版本 2 的升级回滚（升级过程不会删除发行版本 2 的配置数据）。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Application Server 的常规方法相同。

发行版本 2 Application Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Application Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 2 Application Server](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 Application Server 之前，应执行下述任务。

检验当前版本信息

可通过输入以下命令检验 Application Server 的当前版本：

```
AppServer7-base/bin/asadmin version --verbose
```

参见第 162 页中的表 9-3 确认版本输出信息。

升级 Application Server 依赖项

升级 Application Server 依赖项应包括将 Application Server 所依赖的所有本地驻留的产品组件升级到发行版本 4。Java ES 安装程序会在升级过程中自动升级共享组件（参见第 171 页中的步骤 3）。

升级 Application Server 依赖项时，它们应按以下顺序升级，并且应在升级 Application Server 之前进行。您可以跳过任何可能已完成升级的依赖项。

- **Message Queue。** 参见第 133 页中的第 7 章“Message Queue”
- **Web Server（可选）。** 参见第 123 页中的第 6 章“Web Server”

备份 Application Server 数据

Application Server 从发行版本 2 升级至发行版本 4 不会覆写发行版本 2 的配置数据。但为安全起见，应先备份所有 Application Server 实例的配置目录，然后执行升级到发行版本 4。

这些配置目录在以下位置：

AppServer7-base/domains/domainName/instanceName/config

获取所需的配置信息和密码

您应知道有关当前所安装版本的以下信息：

- Application Server 管理员用户 ID、密码和主密码
- 发行版本 2 的 Application Server 基本目录

升级发行版本 2 Application Server

本节讨论影响 Application Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项

将 Application Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 升级 Application Server 实例之前，应先关闭该实例中正在运行的所有 J2EE 组件。但是，如果您使用负载平衡提供高可用性或可伸缩性，则可以放宽此项要求。
- 在升级安装的 Application Server 映像时，必须关闭在单台计算机上运行的所有 Application Server 实例（与同一已安装 Application Server 映像相对应的所有实例）。

升级过程

下述过程适用于本地驻留在即将进行升级的计算机上的所有 Application Server 实例。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 停止所有 Application Server 进程及相关的进程。
3. 使用 Java ES 安装程序并选择“以后再配置”选项安装发行版本 4 的 Application Server。

选择至少安装前三个子组件，包括节点代理组件。

Application Server 软件安装完成后，务必执行在第 173 页中的“在以后再配置安装后配置 Application Server”中提供的安装后过程。

4. 确定目标和源安装目录，例如：
 - 发行版本 2 在 Solaris 上的默认源安装目录：/opt/SUNWappserver7
 - 发行版本 4 在 Solaris 上的默认目标：/opt/SUNWappserver/appserver

5. 运行 asupgrade 实用程序。

asupgrade 实用程序会创建一个发行版本 4 的节点代理，并将发行版本 2 的 Application Server 实例迁移到该节点代理下。

该实用程序位于 Application Server 目录下，例如：

- 升级向导模式： *AppServer8-base/bin/asupgrade*
- 升级控制台模式： *AppServer8-base/bin/asupgrade -c*

升级向导或升级控制台将指导您逐步升级。

有关 Application Server asupgrade 实用程序的更多信息，参阅 Application Server Enterprise Edition 8.1 Upgrade and Migration Guide 2005Q1 (<http://docs.sun.com/doc/819-0222>) 的第 3 章。

6. 启动 Domain Administration Server (DAS)。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

7. 重新启动已升级的 Application Server 实例。

通过启动节点代理（升级后的 Application Server 实例已迁移到该节点代理下）来执行此操作：

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID  
--password password nodeagentName
```

其中，*nodeagentName* 的格式为 *hostName_domainName*。

默认 *domainName* 为 *domain1*。

检验升级

启动管理控制台，检验是否启动了这些服务器。如果有任何服务器未运行，请检查以下日志文件，确定是否存在因端口冲突而引起的故障：

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/logs/server.log
```

如果存在因端口冲突引起的故障，请使用管理控制台修改端口号以消除冲突，然后停止并重新启动节点代理。

可通过输入以下命令检验 Application Server 到发行版本 4 的升级：

```
AppServer8-base/bin/asadmin version --verbose
```

参见第 162 页中的表 9-3 确认输出值。

升级后的任务

除了第 171 页中的“升级过程”中介绍的步骤之外，还存在许多升级后的任务。这些任务包括迁移发行版本 2 的 J2EE 组件和应用程序以在版本 4 Application Server 环境下运行，以及将它们重新部署到相应的 Application Server 实例。

有关迁移 J2EE 组件和应用程序的更多信息，参阅 Application Server Enterprise Edition 8.1 Upgrade and Migration Guide 2005Q1 (<http://docs.sun.com/doc/819-0222>) 的第 4 章。

升级回滚

发行版本 4 的 Application Server 升级回滚过程仅仅是恢复到发行版本 2 的 Application Server 版本，在升级到发行版本 4 时并不会删除该发行版本。

多实例（群集）升级：

Application Server 的 `asupgrade` 实用程序可用于升级多实例群集。有关说明，参见 Application Server Enterprise Edition 8.1 2005Q2 Upgrade and Migration Guide (<http://docs.sun.com/doc/819-2559>) 的第 3 章。

在以后再配置安装后配置 Application Server

在“以后再配置”安装完成后，您需要运行脚本来设置 Application Server 环境。使用以下步骤。

1. 找到 Application Server 的附件分发包：

Sun Java Enterprise System 2005Q4 Accessory CD 1,
Application Server Add Ons for Solaris SPARC and x86,
CD Image 1 of 1

附件内容可从以下网址处的“Sun 下载中心”下载：

<http://www.sun.com/software/javaenterprisesystem/get.xml>

2. 参阅附件分发包 Addon 文件夹中的自述文件并执行指定的过程。
 - a. 运行 `postInstall` 脚本。

应从附件分发包运行 Addon 文件夹中的主要脚本 `postInstall`，否则将找不到某些文件。

该脚本通过在安装期间安装的模板配置并创建一些 `AppServer8-base/bin/*shell` 脚本和一个 `config/asenv` 文件。（通常安装程序会创建一些 `bin/*shell` 脚本，但如果选择“以后再配置”，则必须按所述方式创建它们。）
 - b. 创建新域。

使用 `asadmin create-domain` 命令创建新域时，需要指定以下两个参数的值：`adminPort` 和 `instancePort`。`adminPort` 的值可与发行版本 2 服务器实例使用的值相同，但 `instancePort` 的值不能与任何发行版本 2 服务器实例所用的值相同。通过选择未使用的 `instancePort` 值，可避免发行版本 4 的 DAS 实例与迁移至发行版本 4 的发行版本 2 服务器实例（参见第 172 页中的步骤 5）之间的冲突。
3. 如有必要，请在 `AppServer8-base/config/asenv` 文件中修改环境设置。

必须手动编辑该文件。

注 要配置 Application Server 以实现负载平衡，参阅 Sun Java System Application Server Enterprise Edition High Availability Administration Guide (<http://docs.sun.com/doc/819-0216>) "Application Server High Availability Features" 一章中的 "Configuring Web Servers for HTTP Load Balancing" 一节。

Web Proxy Server

本章介绍如何将 Web Proxy Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Web Proxy Server 4.0 2005Q4。

同时还概述了升级问题和升级过程。仅支持在 Solaris 平台升级 Web Proxy Server。

- [第 176 页中的“Web Proxy Server 升级概述”](#)
- [第 178 页中的“将 Web Proxy Server 升级到发行版本 4”](#)

注 本章中的文件位置是相对于一个称为 *WebProxyServer-base* 的目录路径而指定的。在最初安装 Web Server 时，至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

WebProxyServer-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWproxy
 - Linux: /opt/sun/webproxyserver
-

Web Proxy Server 升级概述

本节将在以下方面对 Web Proxy Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Web Proxy Server](#)
- [Web Proxy Server 升级指导说明](#)
- [Web Proxy Server 数据](#)
- [兼容性问题](#)
- [Web Proxy Server 组件依赖性](#)

关于 Java ES 发行版本 4 Web Proxy Server

Java ES 发行版本 4 的 Web Proxy Server 与其加入 Java Enterprise System 之前的 Sun One Web Proxy Server 3.6 的相比，性能更佳、体系结构伸缩性更强、更符合标准，并具有新的管理界面。

Web Proxy Server 升级指导说明

[表 10-1](#) 显示 Web Proxy Server 到 Java ES 发行版本 4 的升级途径。先前的 Java ES 发行版本中不包括 Web Proxy Server。此表仅适用于 Solaris 操作系统，因为以前在 Linux 操作系统上不支持 Web Proxy Server。

表 10-1 升级到 Java ES 发行版本 4: Sun Java System 4: Web Proxy Server 4.0.1 2005Q4 的途径

Java ES 发行版本	Web Proxy Server 版本	常规方法	必需的重新配置
Java ES 以前的发行版本	Sun ONE Web Proxy Server 3.6 (下文称为版本 3.6)	直接升级： 使用 Java ES 安装程序在新位置安装，然后使用管理工具迁移配置数据来执行	必须将配置信息迁移到新位置。

Web Proxy Server 数据

下表显示了可能受 Web Proxy Server 软件升级影响的数据类型。

表 10-2 Web Proxy Server 数据用途

数据类型	位置	用途
配置数据	<i>WebProxyServer-base/proxy-serverid/</i> config 目录 包含如 <i>server.xml</i> 、 <i>magnus.conf</i> 、 <i>obj.conf</i> 等文件	存储服务器、高速缓存、过滤器、路由及 Web Proxy Server 其他功能方面的配置信息

兼容性问题

发行版本 4 的 Web Proxy Server 对版本 3.6 所支持的 Netscape Server API (NSAPI) 接口做了较大更改。为解决不兼容性问题，为版本 3.6 开发的任何 NSAPI 插件都需要对 NSAPI 的当前版本重新编译。

Web Proxy Server 组件依赖性

Web Proxy Server 对其他 Java ES 组件的依赖性会影响升级和重新配置 Web Proxy Server 软件的过程。

Web Proxy Server 依赖于特定的 Java ES 共享组件（参见第 38 页中的表 1-6），但不依赖其他 Java ES 产品组件。它可以与 Directory Server、Web Server 和 Application Server 一起使用，但并不依赖这些组件。

将 Web Proxy Server 升级到发行版本 4

本节包含关于将 Web Proxy Server 从发行版本 3.6 升级到 Java ES 2005Q4（发行版本 4）的信息。其中涵盖了以下主题：

- [简介](#)
- [Web Proxy Server 升级](#)

简介

在将 Web Proxy Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过使用 Java ES 安装程序在不同于版本 3.6 的目录安装发行版本 4 的 Web Proxy Server 来执行升级。然后使用 Web Proxy Server 管理服务器将配置设置（非高速缓存中的内容）从版本 3.6 迁移到发行版本 4。
- **升级依赖性。** 虽然 Web Proxy Server 对许多 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 的 Web Proxy Server 与发行版本 3 中的这些组件兼容。不过，在将 Web Proxy Server 升级到发行版本 4 时，Java 安装程序会自动升级这些共享组件。
- **向后兼容性。** 发行版本 4 的 Web Proxy Server 与版本 3.6 向后兼容，但必须用发行版本 4 支持的 NSAPI 接口重新编译用版本 3.6 支持的 NSAPI 接口所开发的插件。
- **升级回滚。** 发行版本 4 的 Web Proxy Server 的升级回滚通过恢复到版本 3.6 来实现，该版本在升级时保持不变。
- **平台问题。** 在所有 Solaris 平台上，Web Proxy Server 软件的升级方法都是相同的，但在 Linux 平台上不支持版本 3.6。

Web Proxy Server 升级

本节概括介绍了如何将 Web Proxy Server 升级到 Java ES 发行版本 4。以前，在 Linux 平台上不支持 Web Proxy Server。因此，只能在 Solaris 平台上将 Web Proxy Server 升级到 Java ES 发行版本 4。本节涵盖了以下主题：

- [升级前的任务](#)
- [升级 Web Proxy Server](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 Web Proxy Server 之前，应执行下述任务。

检验当前版本信息

可通过输入以下命令检验 Web Proxy Server 的当前版本：

```
WebProxyServer-base/proxy-serverid/start -version
```

表 10-3 Web Proxy Server 版本检验输出

Java ES 发行版本	Web Proxy Server 版本号
非 Java ES 发行版本的版本 3.6	3.6
发行版本 4	4.0.1

升级 Web Proxy Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，用于将 Web Proxy Server 升级到发行版本 4 的 Java ES 安装程序会自动升级 Web Proxy Server 所依赖的所有共享组件（参见第 38 页中的表 1-6）。

备份 Web Proxy Server 数据

Web Proxy Server 升级到发行版本 4 时不会修改版本 3.6 的配置数据。但是，在升级之前，必须保存任何未保存的用管理界面对版本 3.6 配置数据所做的更改。

获取所需的配置信息和密码

要从版本 3.6 升级，需要知道该已安装版本的安装目录路径。

升级 Web Proxy Server

本节讨论影响 Web Proxy Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项

所有对应同一个已安装 Web Proxy Server 映像的 Web Proxy Server 实例将同时被升级。但是，必须已完成分别为各实例迁移配置数据。向安装的映像应用修补程序时，应关闭所有此类实例。

升级过程

下述过程适用于要进行升级的计算机中的 Web Proxy Server 软件。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 安装 Web Proxy Server 发行版本 4。

- a. 运行 Java ES 安装程序。
- b. 在选择面板上选择 Web Proxy Server。
- c. 选择“现在配置”选项。
- d. 安装完成后退出 Java ES 安装程序。

3. 将配置设置迁移到新安装的版本中。

必须分别为每个 Web Proxy Server 实例执行此操作。

- a. 启动 Web Proxy Server Administration Server。

```
WebProxyServer-base/proxy-admserv/start
```
- b. 登录到图形管理界面。
- c. 单击 "Server" 选项卡，然后单击 "Migrate Server"。
- d. 指定版本 3.6 安装目录路径。
- e. 选择要迁移的实例。
- f. 单击 "Migrate" 按钮。

迁移成功后，迁移屏幕将提供附加配置的列表，必须手动执行这些配置。屏幕会提供需要添加的数据及相应的配置文件。

有关迁移配置设置的更多信息，参阅《Sun Java System Web Proxy Server 4 2005Q4 Installation and Migration Guide》(<http://docs.sun.com/doc/819-3649>)。

4. 执行**步骤 f**中指定的任何附加配置更改。

有关更多信息，参阅《Sun Java System Web Proxy Server 4 2005Q4 Configuration File Reference》(<http://docs.sun.com/doc/819-3651>)。

检验升级

可通过用 `-version` 选项启动 Web Proxy Server 实例来检验 Web Proxy Server 到发行版本 4 的升级：

```
WebProxyServer-base/proxy-serverid/start -version
```

参见第 179 页中的表 10-3 确认输出值。

升级后的任务

除第 180 页中的“升级过程”中所述的步骤之外，没有任何升级后的任务。

升级回滚

第 180 页中的“升级 Web Proxy Server”中介绍的 Web Proxy Server 到发行版本 4 的升级无法回滚。但是，您可恢复到版本 3.6，该版本在经过发行版本 4 升级过程后仍保留原样。

将 Web Proxy Server 升级到发行版本 4

Access Manager

本章介绍如何将 Access Manager 软件从先前的 Java ES 版本升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Access Manager 7 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的 Access Manager 升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 184 页中的“Access Manager 升级概述”
- 第 189 页中的“从 Java ES 发行版本 3 升级 Access Manager”
- 第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”

注

本章中的文件位置是相对于两个目录路径指定的，分别称为 *AccessManager-base* 和 *AccessManagerConfig-base*。在最初安装 Access Manager 时，可能至少已将这两个路径的一部分指定为安装目录，否则，Java ES 安装程序会分配一个默认值。

AccessManager-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWam
- Linux: /opt/sun/identity

AccessManagerConfig-base 的默认值视操作系统平台而定：

- Solaris: /etc/opt/SUNWam
 - Linux: /etc/opt/sun/identity
-

Access Manager 升级概述

本节将在以下方面对 Access Manager 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Access Manager](#)
- [Access Manager 升级指导说明](#)
- [Access Manager 数据](#)
- [兼容性问题](#)
- [Access Manager 组件依赖性](#)

注 在 Java ES 发行版本 3 之前的 Access Manager 版本名为 Identity Server。因此，当在本章提到 Identity Server 时，它是指 Java ES Access Manager 组件的早期版本。

关于 Java ES 发行版本 4 Access Manager

Java ES 发行版本 4 Access Manager 已在多个主要方面得到增强。在后端，该产品的结构已经过重新设计，能够支持多个身份信息库（或者称为用户数据存储库）。因此，发行版本 4 Access Manager 不仅支持 LDAP 目录（如 Directory Server），还支持其他数据存储协议和格式。发行版本 4 Access Manager 包含新接口和新服务，以支持多个身份信息库的集成。

在前端，通过使用一个新的 Access Manager 控制台来配置新的 Access Manager 服务和身份信息库。

这些新功能和接口使发行版本 4 Access Manager 成为了一个重要的新发行版本。为了提供向后兼容性，发行版本 4 可在传统模式下运行，传统模式支持依赖于发行版本 3 Access Manager 服务的 Java ES 组件（有关更多信息，参见第 187 页中的“兼容性问题”）。

Access Manager 升级指导说明

表 11-1 显示了所支持的升级至 Java ES 发行版本 4 的 Access Manager 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 11-1 升级到 Java ES 发行版本 4: Sun Java System Access Manager 7 2005Q4 的途径

Java ES 发行版本	Access Manager 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Access Manager 6.3 2005Q1	直接升级： 执行方法是删除发行版本 3 的版本，然后执行发行版本 4 的完全安装和重新配置。	配置数据 用于 Access Manager 控制台和验证 UI 的自定义 JSP 目录模式
发行版本 2	Sun Java System Identity Server 6.2 2004Q2 及 6.2 SP1	直接升级： 执行方法是删除发行版本 2 的版本，然后执行发行版本 4 的完全安装和重新配置。	配置数据 用于 Access Manager 控制台和验证 UI 的自定义 JSP 目录模式
发行版本 1	Sun ONE Identity Server 6.1	不直接升级： 但可以按照 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237) 中的过程首先升级到发行版本 3。然后从发行版本 3 升级到发行版本 4。	配置数据 用于 Access Manager 控制台和验证 UI 的自定义 JSP 目录模式
Java ES 以前的发行版本	Sun ONE Identity Server 6.0 或 6.0 SP 1, 或 iPlanet Directory Server Access Management Edition (DSAME) 5.1	不直接升级。	

Access Manager 数据

与其他 Java ES 组件一样，Access Manager 将利用对于任一特定升级而言可能需要迁移至升级版本的各种数据。下表显示了可能受 Access Manager 软件升级影响的数据的类型。

表 11-2 Access Manager 数据用途

数据类型	位置	用途
配置数据	<p><i>AccessManagerConfig-base/config/AMConfig.properties</i></p> <p><i>AccessManagerConfig-base/config/serverconfig.xml</i></p> <p>验证和自定义模块的 JAR 文件</p> <p><i>AccessManager-base/lib</i></p>	Access Manager 的配置及其与后端数据存储器的集成。
Web 容器配置	<p>Web Server: <i>server.policy</i> 和 <i>server.xml</i> 文件，位于 <i>WebServer-base/https-hostname/config</i></p> <p>Application Server (Java ES 发行版本 3 和 4) : <i>server.policy</i> 和 <i>domain.xml</i> 文件，位于 <i>AppServer8Config-base/domains/domainName/config</i></p> <p>Application Server (Java ES 发行版本 2) : <i>server.policy</i> 和 <i>server.xml</i> 文件，位于 <i>AppServer7Config-base/domains/domainName/config</i></p> <p>WebSphere 和 WebLogic: 为这两个 Web 容器配置 Access Manager 时修改它们各自的策略和配置文件。</p>	Access Manager Web 容器实例的配置。
自定义数据 (Web 容器自定义 JSP 文件)	<p>管理控制台: <i>AccessManager-base/web-src/applications</i></p> <p>验证 UI: <i>AccessManager-base/web-src/services</i></p>	Access Manager 管理接口的配置。
目录模式 服务配置 用户数据	Directory Server	Access Manager 根据服务配置、用户以及存储在目录中的策略数据，为最终用户提供验证和授权服务。
动态应用程序数据	无	Access Manager 并不永久存储应用程序数据（如会话状态）。

兼容性问题

发行版本 4 Access Manager 的新功能涉及以下新接口：

- 用于多个后端身份信息库的插件
- 用于存储服务配置信息的新目录信息树结构，这样便可将验证属性和授权策略划分到可与一个用户或一组用户相关联的访问控制领域中
- Access Manager 客户机的新 API
- 新的 Access Manager 控制台用户接口

启用支持这些新接口的 Access Manager 的方法是将 Access Manager 配置为在增强（“领域”）模式下运行。但“领域”模式与早期的 Java ES 发行版本 3 和发行版本 2 Access Manager 不兼容。例如，必须迁移目录数据才能支持“领域”模式操作。需要提供增强的 Access Manager 控制台来支持增强的 Access Manager 服务。

此外，“领域”模式也不支持其他 Java ES 组件，如 Portal Server、Communications Express、Messaging Server 等。

为支持向后兼容性，可将发行版本 4 Access Manager 配置为在“传统”模式下运行。“传统”模式与发行版本 3 Access Manager 向后兼容，但有少数几种例外情况（参见《Sun Java System Access Manager 7 2005Q4 发行说明》（<http://docs.sun.com/doc/819-3475>））。

要支持其他 Java ES 组件和旧版 Access Manager 策略代理（在“领域”模式下，这些策略代理无法与 Access Manager 进行交互操作），必须使用“传统”模式。这一不兼容性是升级时的一个重要注意事项，同时意味着在大多数的 Java ES 部署中，应该将 Access Manager 升级到发行版本 4 传统模式。

然而，即便配置为在“传统”模式下运行，发行版本 4 Access Manager 仍与发行版本 3 Delegated Administrator 不兼容。如果 Access Manager 已升级到发行版本 4，则还必须将 Delegated Administrator 升级到发行版本 4，以便为 Messaging Server 和 Calendar Server 置备用户。但 Messaging Server 和 Calendar Server 本身则不必一定升级到发行版本 4。

Access Manager 组件依赖性

Access Manager 对其他 Java ES 组件的依赖性会影响升级和重新配置 Access Manager 软件的过程。例如，Access Manager 接口或功能上的变化可能要求 Access Manager 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Access Manager 对以下 Java ES 组件具有依赖性：

- **共享组件。** Access Manager 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。Access Manager 升级可能要依赖于这些共享组件的已升级版本。
- **Web 容器。** Access Manager 依赖于 Web 容器服务，这些服务可由 Java ES Web Server、Java ES Application Server 或第三方 Web 容器（来自 Weblogic 和 WebSphere）提供。因此，必须为 Web 容器实例重新配置 Access Manager 升级。此外，还需要将 Access Manager 控制台或验证 UI 的所有自定义 JSP 迁移到已升级 Access Manager 环境中。
- **Directory Server。** Access Manager 存储配置数据，同时还要访问存储在 Directory Server 中的用户数据。因此，Access Manager 升级可能需要目录模式的扩展。

从 Java ES 发行版本 3 升级 Access Manager

本节包含关于将 Access Manager 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Access Manager 的完全升级](#)
- [多实例升级：发行版本 3 与发行版本 4 共存](#)
- [仅升级发行版本 3 Access Manager SDK](#)

简介

将 Java ES 发行版本 3 Access Manager 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过删除先前版本然后重新安装发行版本 4 执行升级。随附了一个 `ampre70upgrade` 脚本以删除发行版本 3 的版本，然后使用 Java ES 安装程序来安装发行版本 4。随后使用 `amconfig` 脚本对 Access Manager 进行重新配置，并使用 `amupgrade` 脚本迁移目录模式。
- **升级依赖性。** 虽然 Access Manager 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Access Manager 与所有这些组件在发行版本 3 中的版本兼容。因此，对于 Access Manager 到发行版本 4 的升级，这些组件的升级是可选的。

此外，如第 188 页中的“[Access Manager 组件依赖性](#)”中所述，发行版本 4 Access Manager 依赖于 Directory Server 和 Web Server（或 Application Server 或第三方 Web 容器）。但它们是软性升级依赖性，对于 Access Manager 到发行版本 4 的升级，这些组件的升级是可选的。

- **向后兼容性。** 发行版本 4 Access Manager 与发行版本 3 不兼容，但它却支持兼容的传统模式（参见第 187 页中的“[兼容性问题](#)”）。
- **升级回滚。** 没有用于回滚 Access Manager 升级的实用程序。事实上，将 Access Manager 回滚至其原始状态所需的重新配置的数目过多，从而导致回滚无法实现。
- **平台问题。** 对于 Solaris 和 Linux 操作系统，升级 Access Manager 的常规方法是相同的。随后的过程将指明特定于平台的命令或适当的文件位置。

发行版本 3 Access Manager 的完全升级

本节介绍如何从 Java ES 发行版本 3 到 Java ES 发行版本 4 执行 Access Manager 的完全升级：

- [升级前的任务](#)
- [升级发行版本 3 Access Manager](#)
- [检验 Access Manager 升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

在升级 Access Manager 之前，请执行以下各节中所述的过程。

检验当前版本信息

可使用以下命令检验 Access Manager 的当前版本：

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

表 11-3 Access Manager 版本检验输出

Java ES 发行版本	Access Manager 版本号
发行版本 2	6.2
发行版本 3	6 2005Q1
发行版本 4	7 2005Q4

升级 Access Manager 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但由于 Access Manager 并不要求一定升级它所依赖的 Java ES 发行版本 3 组件，所以此项任务是可选的。

但是，如果选择升级 Access Manager 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Access Manager 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** [第 47 页中的第 2 章“升级 Java ES 共享组件”](#) 中提供了将 Java ES 共享组件升级至发行版本 4 的说明。

2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。

如果未在升级 Access Manager 之前升级 Web 容器软件，升级过程（使用 amconfig 脚本）会将 Access Manager 配置和重新部署到现有 Web 容器中。

备份 Directory Server 数据

Access Manager 升级过程将使用修改 Directory Server 模式的脚本。因此，在升级 Access Manager 之前，请使用 Directory Server 控制台或命令行实用程序（如 db2bak）备份 Directory Server 数据。

有关备份 Directory Server 的更多信息，参见 Sun Java System Directory Server 管理指南 (<http://docs.sun.com/doc/819-2013>)。

备份发行版本 3 Access Manager 配置信息

因为重新配置发行版本 4 Access Manager 软件需要重新配置发行版本 3 的版本，因此将配置文件备份到一个已知位置是非常重要的。应该备份以下文件：

- AMConfig.properties 文件
AccessManagerConfig-base/config/AMConfig.properties
- serverconfig.xml 文件
AccessManagerConfig-base/config/serverconfig.xml
- Web 容器配置文件：
 - 对于 Web Server: server.policy 和 server.xml 文件，位于
WebServer-base/https-hostname/config
 - 对于 Application Server: server.policy 和 domain.xml 文件，位于
AppServer7Config-base/domain/domain1/config
 - 对于第三方 Web 容器: 相应的配置文件
- 验证和自定义模块的 JAR 文件。
AccessManager-base/lib

备份 Web 容器自定义文件

如果 Access Manager 引用了任何 Web 容器自定义文件，应备份这些文件。此类自定义内容可能包括：

- 自定义的 Access Manager 控制台 JSP 页。
AccessManager-base/web-src/applications
- 自定义的验证 UI JSP 页。
AccessManager-base/web-src/services
- 自定义的 XML 文件。
AccessManagerConfig-base/config/xml

提示 将自定义内容记录下来，以便可以在升级 Access Manager 之后使用备份的代码重新应用它们。

备份发行版本 3 Access Manager 日志和调试文件

要分析系统状态信息，最好的办法是备份日志和调试文件以使其不会丢失。这些文件位于以下位置：

- 调试文件
/var/AccessManager-base/debug
- 日志文件
/var/AccessManager-base/logs

获取所需的配置信息和密码

要升级 Access Manager，必须提供具体的配置信息，其中包括：

- Access Manager 管理员用户 ID 和密码
- LDAP 用户 ID 和密码
- Access Manager 正在使用的 Directory Server 实例的目录管理员名称和密码

升级发行版本 3 Access Manager

将 Access Manager 软件升级到 Java ES 发行版本 4 包括重新配置 Access Manager 和迁移 Access Manager 数据的过程。

升级摘要

升级 Access Manager 的过程由下列步骤组成：

1. 删除 Java ES 发行版本 3 的 Access Manager 版本。使用 `ampre70upgrade` 脚本。
2. 安装 Java ES 发行版本 4 的 Access Manager 版本。使用 Java ES 发行版本 4 安装程序并选择“以后再配置”选项。
3. 升级 `mobile access` 软件。
4. 重新自定义 Access Manager 的 JSP。
5. 取消部署 Access Manager，重新配置并重新部署到 Web 容器中。使用 `amconfig` 脚本。
6. 更新目录结构和模式。使用 `amupgrade` 脚本。

以下过程对这些步骤逐一进行说明。

升级过程

1. 删除 Java ES 发行版本 3 的 Access Manager 版本。
 - a. 以 root 身份登录到托管发行版本 3 Access Manager 的计算机或成为超级用户。
`su -`
 - b. 将目录更改为 Java ES 发行版本 4 分包中的 `platform/Product/identity_svr/Tools` 目录。
 - c. 获取 `ampre70upgrade` 脚本将要请求的以下参数的值：

表 11-4 Access Manager 配置参数：`ampre70upgrade`

参数	值
Directory Server Host	设置全限定名称： <code>hostname.domian</code>
Directory Server Port	指定一个非 SSL 端口号 ¹ 默认值：389
Top-Level Administrator DN	默认值： <code>uid=amadmin,ou=People,dc=iplanet,dc=com</code>
Top-Level Administrator Password	

1. 如果指定一个 Directory Server SSL 端口（如默认的 SSL 值 636），则无法成功完成升级前的过程。

d. 确保 Directory Server 正在运行，否则请将其启动。

e. 运行 ampre70upgrade 脚本。

```
./ampire70upgrade
```

该脚本将备份 Access Manager 配置文件，并删除发行版本 3 基本软件包（必须按照步骤 f 手动删除本地化软件包）。

f. 手动删除计算机上的 Access Manager 本地化软件包。

ampire70upgrade 脚本不会删除 Access Manager 本地化软件包。必须手动将其删除方可执行正确的本地化升级。

- 在 Solaris 平台上使用 pkgrm 删除：SUNWamlLocale、SUNWLocaleammmmap
- 在 Linux 上使用 rpm -e 删除：sun-identity-sdk-Locale

2. 安装 Java ES 发行版本 4 的 Access Manager 版本。

a. 在托管发行版本 3 Access Manager 的计算机上运行 Java ES 安装程序。

b. 在选择面板上选择 Access Manager。

如果屏幕上显示“冲突”消息，则表明安装程序已找到了先前版本的 Access Manager 配置信息，这是预料中的消息。将在后续步骤中执行重新配置。您可以忽略此“冲突”消息然后继续执行操作。

c. 指定安装目录，需与发行版本 3 的安装目录相同。

d. 选择“以后再配置”选项。

e. 安装完成后退出 Java ES 安装程序。

注 如果要使用“Java ES 安装程序”命令行界面安装 Access Manager，则还将自动安装 Directory Server 软件。如果要使用远程 Directory Server，则可按照 Java Enterprise System 安装指南（适用于 UNIX）中的过程卸载本地 Directory Server 软件。

3. 升级 mobile access 软件。

Access Manager mobile access 软件需要通过修补发行版本 3 的版本进行升级。下表显示了所需的修补程序：

表 11-5 用于升级 Access Manager Mobile Access 软件的修补程序¹

说明	Solaris 修补程序 ID	Linux 修补程序 ID
Mobile Access 软件	119530-01 (SPARC) 119531-01 (x86)	119532-01 <ul style="list-style-type: none"> • sun-identity-mobileaccess-6.2-25.i386.rpm • sun-identity-mobileaccess-config-6.2-25.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

a. 利用表 11-5 中的修补程序号获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

b. 以 root 身份登录，或成为超级用户。

```
su -
```

c. 应用表 11-5 中的修补程序。

在 Solaris 上：

```
patchadd patch_ID
```

在 Linux 上：

```
rpm -Fvh sun-identity-mobileaccess-6.2-25.i386.rpm
rpm -Fvh sun-identity-mobileaccess-config-6.2-25.i386.rpm
```

4. 重新自定义 Access Manager 的 JSP。

重新应用按第 192 页中的“备份 Web 容器自定义文件”中所述保存的、用于 Access Manager 控制台和验证用户界面 (User Interface, UI) 的发行版本 3 自定义 JSP 设置。

然后，将自定义 JSP 文件复制到正确的目录中。例如，在 Solaris 系统上：

- o 控制台：*AccessManager-base/web-src/applications/console*

- 验证 UI: `AccessManager-base/web-src/services/config/auth/default` 或 `AccessManager-base/web-src/services/config/auth/default_Locale` (其中 `Locale` 是语言环境指示符, 如 `ja`)

有关更多信息, 参见 Sun Java System Access Manager Developer's Guide (<http://docs.sun.com/doc/819-2139>)。

5. 取消部署 Access Manager, 重新配置并重新部署到 Web 容器中。

通过运行 `amconfig` 脚本为特定 Web 容器配置 Access Manager。 `amconfig` 脚本 (及相关联的 `amsamplesilent` 模板输入文件) 位于以下目录中:

`AccessManager-base/bin`

有关 `amconfig` 脚本和 `amsamplesilent` 模板文件的信息, 参见 Sun Java System Access Manager 管理指南 (<http://docs.sun.com/doc/819-1940>)。

执行以下步骤将 Access Manager 重新配置和重新部署到 Web 容器中:

- 如果选择升级 Web 容器软件 (如第 190 页中的“升级 Access Manager 依赖项”中所述), 需确保升级全部完成。
- 检查 Directory Server 及相应的 Web 容器是否正在运行。
- 根据 `amsamplesilent` 模板输入文件创建一个 `amconfig` 输入文件:
`cp amsamplesilent config-file`
- 设置 `config-file` 中的配置参数。

需正确设置所有参数。一些参数的值可从 `AMConfig.properties` 文件迁移, 相比之下, 其他参数的值则是升级过程的专用值, 如下表所示。

表 11-6 Access Manager 配置参数

参数	值
升级参数	
DEPLOY_LEVEL	26 (用于取消部署) 或 1 (用于重新配置和部署)
DIRECTORY_MODE	5 (现有升级)
AM_REALM	设置为 disabled (“领域模式”为 disabled, 因此“传统模式”为 enabled) (默认值为 enabled)
JAVA_HOME	设置为 JDK 发行版本 4 目录: <code>/usr/java/jdk1.5.0_04/</code>
WEB_CONTAINER	设置为适合于您所使用的 Web 容器类型的值, 且仅填写 <code>config-file</code> 的对应部分。

表 11-6 Access Manager 配置参数 (续)

参数	值
WS61_INSTANCE (如果将 Web Server 用作 Web 容器)	=https-<hostname>.<domain> 其中上述值与 /WebServer-base/SUNWsbsvr/ 中的实例名相匹配 该值区分大小写。
迁移自 AMConfig.properties	
SERVER_PROTOCOL	com.ipplanet.am.server.protocol
SERVER_PORT	com.ipplanet.am.server.port
SERVER_HOST	com.ipplanet.am.server.host
DS_HOST	com.ipplanet.am.directory.host
DS_PORT	com.ipplanet.am.directory.port
ROOT_SUFFIX	com.ipplanet.am.defaultOrg
CONSOLE_DEPLOY_URI	com.ipplanet.am.console.deploymentDescriptor
SERVER_DEPLOY_URI	com.ipplanet.am.services.deploymentDescriptor
PASSWORD_DEPLOY_URI	com.sun.identity.password.deploymentDescriptor
AM_ENC_PWD	am.encryption.pwd

对于其他参数，如果不更改 Web 容器和密码，请提供要升级的发行版本 3 配置中使用的那些值。

- e. 运行 `amconfig` 以取消部署 Access Manager。

将 `config-file` 中 `DEPLOY_LEVEL` 的值设置为 26。

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

- f. 运行 `amconfig` 以重新配置 Access Manager 并将其部署到 Web 容器中。

将 `config-file` 中 `DEPLOY_LEVEL` 的值设置为 1。

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

6. 更新目录结构和模式。

发行版本 4 Access Manager 可与发行版本 3 的目录结构共存，但必须对该结构进行修改以使其支持发行版本 4 的功能。通过运行 `amupgrade` 脚本，将 Access Manager 目录结构和模式更新为发行版本 4，该脚本安装在以下目录中：

- o Solaris:
`AccessManager-base/upgrade/scripts`
- o Linux:
`AccessManager_base/identity/upgrade/scripts`
- a. 获取 `amupgrade` 脚本将要请求的以下参数的值：

表 11-7 Access Manager 配置参数：amupgrade

参数	值
Directory Server Host	设置全限定名称： <code>hostname.domian</code>
Directory Server Port	指定一个非 SSL 端口号 ¹ 默认值：389
Directory Manager DN	默认值： <code>cn=Directory Manager</code>
Directory Manager Password	
Top-Level Administrator DN	默认值： <code>uid=amadmin,ou=People,dc=iplanet,dc=com</code>
Top-Level Administrator Password	
Enable Realm Mode	Y/N: Yes 表明“领域模式”已启用且服务数据已迁移到新的“领域”树。No（默认值）表明服务数据仍然处于“传统模式”下。

1. 如果指定一个 Directory Server SSL 端口（如默认的 SSL 值 636），则无法成功完成升级过程。

b. 运行 `amupgrade` 脚本。

```
cd AccessManager-base/upgrade/scripts
./amupgrade
```

如果升级成功，脚本将显示 "Upgrade completed"。

- c. 查看以下升级日志文件以了解有关目录模式扩展的信息：

Solaris:

```
/var/sadm/install/logs/  
Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm
```

Linux:

```
/var/log/Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm
```

7. 启动 Access Manager。

重新启动部署了 Access Manager 的 Web 容器。

检验 Access Manager 升级

完成升级过程后，可按以下方法检验升级是否成功：

1. 以 amadmin 身份使用以下 URL 登录到 Access Manager 控制台：

```
http://hostname.domain:port/amconsole
```

其中，*hostname.domain:port* 是托管 Access Manager 的 Web 容器的全限定主机名和端口号。

检验“服务配置”选项卡下是否提供了第 184 页中的“关于 Java ES 发行版本 4 Access Manager”中提到的新的发行版本 4 服务。

2. 通过检查 /var/sadm/install/logs 目录中的下列升级日志文件查看升级状态：

Sun Java Enterprise System 安装程序：

- 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 脚本：

- o `Sun_Java_System_Identity_Server_upgrade_dit_log.timestamp`

3. 检查 Access Manager 错误诊断文件以了解错误。

文件位于 /var/opt/SUNWam/debug

升级后的任务

如果要使用安全声明标记语言 (Security Assertion Markup Language, SAML) 服务，必须使用 Access Manager 控制台添加并启用一个 SAML 验证模块。有关创建 SAML 验证模块实例的信息，参阅 Sun Java System Access Manager 管理指南 (<http://docs.sun.com/doc/819-1940>)。

升级回滚

未提供用于将 Access Manager 回滚至其升级前的状态的脚本。必须使用作为升级前任务的一部分而备份的 Access Manager 数据手动执行该过程（参见第 192 页中的“备份发行版本 3 Access Manager 日志和调试文件”）。回滚过程过于困难以至于无法实现。

多实例升级：发行版本 3 与发行版本 4 共存

在一些部署体系结构中，Access Manager 被部署在多个计算机系统上，以提供高可用性和可伸缩性。多个 Access Manager 实例将访问同一个 Directory Server，而用户常常希望在不中断服务的情况下按顺序升级 Access Manager 实例。本节将讨论执行此类滚动升级的过程。

注 当前发行版本不支持升级那些安装在同一主机系统上的多个 Access Manager 实例。如果在同一主机上有多个实例，在升级主实例之后，必须重新创建附加实例。

从发行版本 3 升级 Access Manager 的过程包括迁移目录模式以支持发行版本 4 的步骤。发行版本 3 Access Manager 不支持发行版本 4 目录模式，但发行版本 4 Access Manager 却支持发行版本 3 目录模式。

仅在目录模式尚未迁移到发行版本 4 时，Java ES 发行版本 4 Access Manager 和发行版本 3 Access Manager 实例才可以共存并能够在同一个 Directory Server 上同时运行。因此，滚动升级时，除非已先将所有 Access Manager 实例升级到发行版本 4，否则不应将目录模式迁移至发行版本 4。

在执行滚动升级时，请按第 193 页中的“升级发行版本 3 Access Manager”中所述升级每个 Access Manager 实例，但不要执行第 198 页上的“更新目录结构和模式”步骤。在升级了所有实例之后，方可执行该步骤。

仅升级发行版本 3 Access Manager SDK

在一些部署体系结构中，Access Manager SDK 组件被安装在一个或多个计算机系统上，而在这些计算机上却未安装其他 Access Manager 组件。Access Manager SDK 相当于到 Access Manager 的远程接口，且必须将其操作模式重新配置为与 Access Manager 相同：“传统”或“领域”。作为到 Access Manager 的远程接口，SDK 无需配置便能访问 Directory Server。

如果正在使用 Access Manager SDK 来支持 Web 组件（如依赖于 Web 容器服务的 Portal Server 或 Communications Express），则必须为相应的 Web 容器配置 Access Manager SDK。但 Access Manager SDK 也能支持非 Web 组件，此时无需 Web 容器。

基于上述特点，升级 Access Manager SDK 的过程是 Access Manager 完全升级过程的一部分。

本节介绍如何从 Java ES 发行版本 3 到 Java ES 发行版本 4 执行仅限于 Access Manager SDK 的升级：

- [升级前的任务](#)
- [升级发行版本 3 Access Manager SDK](#)
- [检验 Access Manager SDK 升级](#)
- [升级回滚](#)

升级前的任务

Access Manager SDK 升级前的任务与 Access Manager 完全升级相同，但不包括那些与 Directory Server 和管理工具自定义相关的内容。Access Manager SDK 升级前需完成的任务如下：

- [第 190 页中的“升级 Access Manager 依赖项”](#)

但对于 Access Manager SDK 而言，对 Directory Server 没有依赖性，且仅当 Access Manager SDK 在 Web 容器中运行时才对 Web 容器软件具有依赖性。

- [第 192 页中的“备份发行版本 3 Access Manager 日志和调试文件”](#)

但对于 Access Manager SDK 而言，仅当 Access Manager SDK 在 Web 容器中运行的情况下才需备份 Web 容器配置文件。

- [第 192 页中的“备份发行版本 3 Access Manager 日志和调试文件”](#)

要访问这些文件，还需获得管理员用户名和密码。

升级发行版本 3 Access Manager SDK

Access Manager SDK 的升级过程与 Access Manager 完全升级相同，但不包括那些与管理工具自定义和迁移目录模式相关的内容。

1. 删除 Java ES 发行版本 3 的 Access Manager SDK 版本。

按照第 193 页中的“删除 Java ES 发行版本 3 的 Access Manager 版本”中的说明操作，只不过删除的是 Access Manager SDK。

2. 安装 Java ES 发行版本 4 的 Access Manager SDK 版本。

按照第 194 页中的“安装 Java ES 发行版本 4 的 Access Manager 版本”中的说明操作，只不过安装的是 Access Manager SDK。

3. 重新配置 Access Manager SDK。

按照第 196 页中的“取消部署 Access Manager，重新配置并重新部署到 Web 容器中”中的说明操作，不过要设置 `DIRECTORY_MODE=5` 并按如下所示设置 `DEPLOY_LEVEL` 参数：

- 如果为 Web 容器配置了 Access Manager SDK：
`DEPLOY_LEVEL=4`（升级 SDK 并配置 Web 容器）
- 如果没有为 Web 容器配置 Access Manager SDK：
`DEPLOY_LEVEL=3`（仅升级 SDK）

检验 Access Manager SDK 升级

检验 Access Manager SDK 升级是否成功的方法有三种：

- 运行 Portal Server、Communications Express 或其他使用 Access Manager SDK 的组件以与 Access Manager 连接，并检查验证是否有效。
- 运行以下位置中提供的 Access Manager SDK 示例：

```
/AccessManager-base/samples/sdk
```

- 检查 `com.ipplanet.am.version` 属性的值，该属性在 `AMConfig.properties` 文件中：

```
AccessManagerConfig-base/config/AMConfig.properties
```

升级回滚

未提供用于将 Access Manager 回滚至其升级前的状态的脚本。必须使用作为升级前任务的一部分而备份的 Access Manager 数据手动执行该过程（参见第 192 页中的“备份发行版本 3 Access Manager 日志和调试文件”）。回滚过程过于困难以至于无法实现。

从 Java ES 发行版本 2 升级 Access Manager

将 Java ES 2004Q2（发行版本 2）Access Manager 升级至发行版本 4 的过程与将发行版本 3 Access Manager 升级至发行版本 4 大致相同，下文说明了其中的几处例外。

升级前的任务

在升级 Access Manager 之前，请执行第 190 页中的“升级前的任务”中所述的过程，但要用下节替换第 190 页中的“升级 Access Manager 依赖项”并添加下面的“升级目录模式”一节。

升级 Access Manager 依赖项

与从发行版本 3 升级相比，从发行版本 2 到发行版本 4 的升级前任务应包括将 Access Manager 所依赖的所有共享组件（参见第 38 页中的表 1-6）和所有本地驻留的产品组件升级到发行版本 4。

升级 Access Manager 依赖项时，它们应按以下顺序升级，并且应在升级 Access Manager 之前进行。您可以跳过任何可能已完成升级的依赖项。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** Directory Server 与 Access Manager 很少会驻留在同一台计算机上，但是，第 110 页中的“从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 132 页中的“从 Java ES 发行版本 2 升级 Web Server”和第 169 页中的“从 Java ES 发行版本 2 升级 Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。

升级目录模式

如果已用 Directory Preparation Tool (comm_dssetup.pl) 对 Directory Server 进行了配置，以支持 Messaging Server、Calendar Server 或其他通信组件，则必须首先使用发行版本 4 的 Directory Preparation Tool 版本升级目录模式，然后再升级 Access Manager。请在 Access Manager 依赖项升级完毕后执行这一升级前的任务。第 218 页中的“从 Java ES 发行版本 2 升级 Directory Preparation Tool”中介绍了升级 Directory Preparation Tool 的过程。

发行版本 2 Access Manager 的升级

将 Access Manager 从发行版本 2 升级至发行版本 4 的过程视您在其中部署 Access Manager 软件的 Web 容器而定。

升级发行版本 2 Access Manager: Web Server Web 容器

当部署到 Web Server Web 容器中时，要将发行版本 2 Access Manager 升级至发行版本 4，可按照第 193 页中的“[升级发行版本 3 Access Manager](#)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

升级发行版本 2 Access Manager: Application Server Web 容器

当部署到 Application Server Web 容器中时，要将发行版本 2 Access Manager 升级到发行版本 4，需分两种情况：

- 发行版本 4 Application Server 是新安装的。这种情况下，要将发行版本 2 Access Manager 升级到发行版本 4，可按照第 193 页中的“[升级发行版本 3 Access Manager](#)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。
- 已将发行版本 2 Application Server 升级到发行版本 4。这种情况下，要将发行版本 2 Access Manager 升级到发行版本 4，请按照下面的说明操作。

当 Access Manager 部署到已升级 Application Server Web 容器中时，要将其升级，请按照第 193 页中的[步骤 1 至步骤 4](#)执行操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

现对[步骤 1 至步骤 4](#)总结如下：

1. 删除发行版本 2 的 Access Manager 版本。
使用 `ampre70upgrade` 脚本。按照第 193 页中的“[删除 Java ES 发行版本 3 的 Access Manager 版本](#)”中的说明操作。
2. 安装 Java ES 发行版本 4 的 Access Manager 版本。使用 Java ES 发行版本 4 安装程序并选择“以后再配置”选项。
3. 升级 `mobile access` 软件。
4. [重新自定义 Access Manager 的 JSP](#)。

最初在其中部署 Access Manager 的发行版本 2 Application Server 实例 (*instanceName*) 会在升级到发行版本 4 之后，被迁移到升级过程所创建的一个节点代理下。对于该已升级 Application Server 实例中的 Access Manager 的升级需要执行以下附加步骤：

5. 确保以下支持 Access Manager 的组件正在运行。
 - a. 检查 Directory Server 是否正在运行。
 - b. 启动 Domain Administration Server (DAS)（如果尚未启动）。


```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```
 - c. 启动其中部署了 Access Manager 的已升级 Application Server 实例 (*instanceName*)（如果该服务器实例尚未运行）。

通过启动其下迁移了已升级 Application Server 实例的节点代理来实现此操作：

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

在上述命令及后续步骤中，将使用下面的约定：

- o *nodeAgentName* 的形式为 *hostName_domainName*。
 - o 默认的 *domainName* 为 *domain1*。
 - o 默认的 *instanceName* 为 *server1*。
6. 取消部署 Access Manager，重新配置并重新部署到 Application Server 实例中。使用 *amconfig* 脚本。
 - a. 根据 *amsamplesilent* 模板输入文件创建一个 *amconfig* 输入文件：


```
cp amsamplesilent config-file
```

b. 设置 *config-file* 中的配置参数。

需正确设置所有参数。一些参数的值可从 *AMConfig.properties* 文件迁移，相比之下，其他参数的值则是升级过程的专用值，如下表所示。

表 11-8 Access Manager 配置参数

参数	值
升级参数	
DEPLOY_LEVEL	26（用于取消部署）或 1（用于重新配置和部署）
DIRECTORY_MODE	5（现有升级）
AM_REALM	设置为 disabled（“领域模式”为 disabled，因此“传统模式”为 enabled）；默认值为 enabled
JAVA_HOME	设置为 JDK 发行版本 4 目录： /usr/java/jdk1.5.0_04/
WEB_CONTAINER	设置为 Application Server Web 容器的值，且仅填写 <i>config-file</i> 的对应部分。
AS81_INSTANCE	=instanceName
AS81_ADMIN_IS_SECURE	=false
迁移自 AMConfig.properties	
SERVER_PROTOCOL	com.iplanet.am.server.protocol
SERVER_PORT	com.iplanet.am.server.port
SERVER_HOST	com.iplanet.am.server.host
DS_HOST	com.iplanet.am.directory.host
DS_PORT	com.iplanet.am.directory.port
ROOT_SUFFIX	com.iplanet.am.defaultOrg
CONSOLE_DEPLOY_URI	com.iplanet.am.console.deploymentDescriptor
SERVER_DEPLOY_URI	com.iplanet.am.services.deploymentDescriptor
PASSWORD_DEPLOY_URI	com.sun.identity.password.deploymentDescriptor
AM_ENC_PWD	am.encrypted.pwd

对于其他参数，如果不更改 Web 容器和密码，请提供要升级的发行版本 2 配置中使用的那些值。

- c. 运行 `amconfig` 以取消部署 Access Manager。

将 `config-file` 中 `DEPLOY_LEVEL` 的值设置为 26。

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

- d. 运行 `amconfig` 以重新配置 Access Manager 并将其部署到 Web 容器中。

将 `config-file` 中 `DEPLOY_LEVEL` 的值设置为 1。

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

7. 将 `server.policy` 文件从以下目录：

```
AppServer8Config-base/domains/domainName/config
```

复制到以下目标目录：

```
AppServer8Config-base/nodeagents/nodeagentName/
instanceName/config
```

8. 修改发行版本 4 Application Server `domain.xml` 文件。

- a. 在最初部署 Access Manager 的发行版本 2 Application Server 实例的 `server.xml` 文件中，复制 Access Manager `classpath-suffix` 和 `server-classpath` 信息：

```
AppServer7Config-base/domains/domainName/instanceName/config/server.xml
```

- b. 在已升级 Application Server 实例（其中部署了 Access Manager）的 `domain.xml` 文件中，将复制的类路径信息分别附加到 `classpath-suffix` 和 `server-classpath` 条目上：

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/
config/domain.xml
```

应将类路径信息添加到发行版本 4 Application Server `domain.xml` 文件的 `instanceName-config` 块中。该块的起始行是：

```
<config dynamic-reconfiguration-enabled="true"
name="instanceName-config">
```

在向类路径条目添加信息时，应确保在新旧信息之间加一个冒号 (":")，或者类路径条目中当前所使用的任何一种路径分隔符。也可删除带 `AppServer7-base` 路径的所有条目（注意不要引入错误）。

9. 重新启动 DAS。

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID  
--password password domainName
```

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

10. 重新启动在其中部署了 Access Manager 的服务器实例。

```
AppServer8-base/bin/asadmin stop-node-agent --user admin_ID  
--password password nodeagentName
```

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID  
--password password nodeagentName
```

11. 按第 198 页中的步骤 6 所述更新目录结构和模式。

检验 Access Manager 升级

完成升级过程后，可按第 199 页中的“检验 Access Manager 升级”中所述检验升级是否成功。

升级后的任务

如果要使用安全声明标记语言 (Security Assertion Markup Language, SAML) 服务，必须使用 Access Manager 控制台添加并启用一个 SAML 验证模块。有关创建 SAML 验证模块实例的信息，参阅 Sun Java System Access Manager 管理指南 (<http://docs.sun.com/doc/819-1940>)。

升级回滚

未提供用于将 Access Manager 回滚至其升级前的状态的脚本。必须使用作为升级前任务的一部分而备份的 Access Manager 数据手动执行该过程（参见第 192 页中的“备份发行版本 3 Access Manager 日志和调试文件”）。回滚过程过于困难以至于无法实现。

Directory Preparation Tool

本章介绍如何将 Directory Preparation Tool 升级到 Java ES 2005Q4（发行版本 4）：
Sun Java System Directory Preparation Tool 6.3 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- [第 210 页中的“Directory Preparation Tool 升级概述”](#)
- [第 212 页中的“从 Java ES 发行版本 3 升级 Directory Preparation Tool”](#)
- [第 218 页中的“从 Java ES 发行版本 2 升级 Directory Preparation Tool”](#)

注

本章中的文件位置是相对于一个称为 *DirPrepTool-base* 的目录路径而指定的。在最初安装 Directory Preparation Tool 时，可能至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

DirPrepTool-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWcomds
 - Linux: /opt/sun/comms/dssetup
-

Directory Preparation Tool 升级概述

本节将在以下方面对 Directory Preparation Tool 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Directory Preparation Tool](#)
- [Directory Preparation Tool 升级指导说明](#)
- [Directory Preparation Tool 数据](#)
- [兼容性问题](#)
- [Directory Preparation Tool 依赖性](#)

关于 Java ES 发行版本 4 Directory Preparation Tool

Java ES 发行版本 4 的 Directory Preparation Tool 进行了许多必要的修复，以准备好 Directory Server 供发行版本 4 的通信组件（Messaging Server、Calendar Server、Communications Express 和 Delegated Administrator）使用。

有关详细信息，参见相应的发行说明。

Directory Preparation Tool 升级指导说明

表 12-1 显示了所支持的升级至 Java ES 发行版本 4 的 Directory Preparation Tool 升级途径。此表既适用于 Solaris 操作系统，也适用于 Linux 操作系统。

表 12-1 升级到 Java ES 发行版本 4：
Sun Java System Directory Preparation Tool 6.3 2005Q4 的途径

Java ES 发行版本	Directory Preparation Tool 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Directory Preparation Tool 6.2 2005Q1	直接升级： 通过应用修补程序执行。	为发行版本 4 的通信组件准备 Directory Server
发行版本 2	comm_dssetup.pl 脚本 版本 6.1 修订版 0.2（与 Messaging Server 和 Calendar Server 捆绑在一起）	直接升级： 通过应用起源修补程序后应用升级修补程序执行。	为发行版本 4 的通信组件准备 Directory Server

表 12-1 升级到 Java ES 发行版本 4:
Sun Java System Directory Preparation Tool 6.3 2005Q4 的途径 (续)

Java ES 发行版本	Directory Preparation Tool 版本	常规方法	必需的重新配置
发行版本 1	comm_dssetup.pl 脚本 (与 Messaging Server 和 Calendar Server 捆绑在一起)	直接升级未经验证: 但可以通过应用起源修补程序后 应用升级修补程序执行。	为发行版本 4 的通信组件 准备 Directory Server
Java ES 以前的发行版本	ims_dssetup.pl 脚本 (与 Messaging Server 捆绑在一起)	不直接升级:	

Directory Preparation Tool 数据

下表显示了可能受 Directory Preparation Tool 软件升级影响的数据类型。

表 12-2 Directory Preparation Tool 数据用途

数据类型	位置	用途
Directory Server 模式	Directory Server	为发行版本 4 的通信组件 准备 Directory Server: 修改模 式、创建新条目和创建索引

兼容性问题

发行版本 4 Directory Preparation Tool 的接口没有任何变化, 并与早期版本向后兼容。

Directory Preparation Tool 依赖性

Directory Preparation Tool 依赖 Directory Server, 除此之外不依赖其他 Java ES 组件。Directory Preparation Tool 用于配置 Directory Server 以同 Java ES 通信组件一起使用。

从 Java ES 发行版本 3 升级 Directory Preparation Tool

本节包含关于将 Directory Preparation Tool 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Directory Preparation Tool 的升级](#)

简介

将 Java ES 发行版本 3 的 Directory Preparation Tool 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 3 版本执行升级。然后即可根据需要使用 Directory Preparation Tool 修改 Directory Server，以支持发行版本 4 的 Messaging Server、Calendar Server、Communications Express 和 Delegated Administrator 组件。
- **升级依赖性。** Directory Preparation Tool 不依赖 Java ES 共享组件，并与发行版本 3 的 Directory Server 兼容。因此，相对于 Directory Preparation Tool 到发行版本 4 的升级，Directory Server 的升级是可选的。
- **向后兼容性。** 发行版本 4 的 Directory Preparation Tool 与发行版本 3 版本向后兼容。
- **升级回滚。** 发行版本 4 到发行版本 3 的升级回滚可通过删除升级期间所应用的修补程序实现。然后可针对 Directory Server 运行发行版本 3 的 Directory Preparation Tool，撤消发行版本 4 所做的更改。
- **平台问题。** 在 Solaris 操作系统和 Linux 操作系统中，升级 Directory Preparation Tool 的常规方法相同，但是修补技术却不同。因此该升级过程包括特定于平台的过程，且通常无法在 Linux 平台上回滚修补程序。

发行版本 3 Directory Preparation Tool 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Directory Preparation Tool 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Directory Preparation Tool \(Solaris\)](#)
- [升级发行版本 3 Directory Preparation Tool \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Directory Preparation Tool 之前，应执行下述任务。

检验当前版本信息

可检验上次针对 Directory Server 运行的 Directory Preparation Tool 的版本，方法是检查该工具所写的 `cn=CommServers,o=comms-config` 条目的属性值：

```
./ldapsearch -D "cn=Directory Manager" -w password
             -b cn=CommServers,o=comms-config cn="CommServers"
             sunkeyvalue
```

该条目有两个用于指定当前版本的属性：

- `dssetup_ver=version`（例如，6.3）
- `dssetup_rev=revision`（例如，2.01）

仅在当前运行的 Directory Preparation Tool 的版本等同于或早于先前运行的工具版本时，该工具才会向控制台写一条消息。请参见升级过程，[第 215 页中的步骤 5 \(Solaris\)](#) 或 [第 217 页中的步骤 5 \(Linux\)](#)，了解如何运行该工具。

升级 Directory Preparation Tool 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，Directory Preparation Tool 到发行版本 4 的升级不依赖任何其他 Java ES 组件。

备份目录数据

Directory Preparation Tool 从发行版本 3 升级到发行版本 4 时本身并不会修改 Directory Server 数据。但是，为安全起见，最好先备份 Directory Server，然后升级 Directory Preparation Tool 并针对 Directory Server 运行该工具。

获取所需的配置信息和密码

Directory Preparation Tool 升级要求您知道超级用户密码。该工具会记住上次运行时使用的参数值，并在下次运行时将这些值作为默认值提供。

升级发行版本 3 Directory Preparation Tool (Solaris)

本节讨论影响 Directory Preparation Tool 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Directory Preparation Tool 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 发行版本 3 的 Directory Preparation Tool 已随 Directory Server 一起安装，并驻留在托管 Directory Server 的任意一台计算机上。
- 必须在托管将由 Messaging Server、Calendar Server、Communications Express 或 Delegated Administrator 组件使用的每个 Directory Server 实例的计算机上执行 Directory Preparation Tool 的升级。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Directory Preparation Tool 升级修补程序：

表 12-3 在 Solaris 上升级 Directory Preparation Tool 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Directory Preparation Tool (DSSETUP)	118245-05	118246-05

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于安装在 Directory Server 所驻留的计算机上的 Directory Preparation Tool。

1. 根据表 12-3，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 应用表 12-3 中相应的 Directory Preparation Tool 修补程序。

```
patchadd patch_ID
```

4. 确认修补程序升级成功:

```
showrev -p | grep patch_ID
```

输出应返回在步骤 3 中应用的修补程序 ID 的版本。

5. 针对 Directory Server 运行 Directory Preparation Tool。

- a. 确认 Directory Server 正在运行。

- b. 将目录更改为 Directory Preparation Tool 所在的位置。

```
cd DirPrepTool-base/sbin
```

- c. 运行 Directory Preparation Tool (comm_dssetup.pl perl 脚本)。

```
perl comm_dssetup.pl
```

提供该脚本要求的参数。

升级发行版本 3 Directory Preparation Tool (Linux)

本节讨论影响 Directory Preparation Tool 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Directory Preparation Tool 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 214 页中的“升级注意事项 (Solaris)”）。

下表显示了用于 Linux 操作系统的发行版本 4 的 Directory Preparation Tool 升级修补程序：

表 12-4 在 Linux 上升级 Directory Preparation Tool 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Directory Preparation Tool (DSSETUP)	118247-05 <ul style="list-style-type: none"> sun-comms-dssetup-6.3-2.5.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于安装映像 in Directory Server 所驻留的计算机上的 Directory Preparation Tool。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 12-4 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 应用表 12-4 中用于 Directory Preparation Tool 的 RPM。

```
rpm -Uvh sun-comms-dssetup-6.3-2.5.i386.rpm
```

4. 确认升级成功：

```
rpm -q sun-comms-dssetup
```

应该返回 RPM 的新版本号。

5. 针对 Directory Server 运行 Directory Preparation Tool。
 - a. 确认 Directory Server 正在运行。
 - b. 将目录更改为 Directory Preparation Tool 所在的位置。

```
cd DirPrepTool-base/sbin
```
 - c. 运行 Directory Preparation Tool (comm_dssetup.pl perl 脚本)。

```
perl comm_dssetup.pl
```

提供该脚本要求的参数。

检验升级

通过检查运行脚本时创建的日志文件，可检验 Directory Preparation Tool 升级以及目录模式扩展是否成功。该日志文件位于：

```
/var/tmp/dssetup_YYYYMMDDHHMMSS
```

升级后的任务

除第 215 页中的“升级过程 (Solaris)”和第 216 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Directory Preparation Tool 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Directory Preparation Tool 的升级回滚过程是升级至发行版本 4 的相反过程。但是，Directory Preparation Tool 所做更改中包括了对 Directory Server 模式的修改。这些更改不能通过下述回滚过程撤消，但模式更改可向后兼容。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 删除第 214 页中的表 12-3 中的修补程序。

```
patchrm patch_ID
```

3. 针对 Directory Server 运行回滚的 Directory Preparation Tool。

Directory Server 修改（包括索引）会恢复到先前的状态，但是模式更改仍会保留。这些更改对模式扩展没有负面影响；它们可向后兼容。

从 Java ES 发行版本 2 升级 Directory Preparation Tool

将 Java ES 2004Q2（发行版本 2）Directory Preparation Tool 升级至发行版本 4 的过程与将发行版本 3 的 Directory Preparation Tool 升级至发行版本 4 的过程大致相同，但存在以下例外。

在 Java ES 发行版本 2 中，Directory Preparation Tool（随后称为 `comm_dssetup`）与 Messaging Server 和 Calendar Server 捆绑在一起，不是作为单独的软件包安装。因此，在托管 Directory Server 的计算机上没有已安装的 Directory Preparation Tool 软件包或 RPM。为此，要从发行版本 2 升级到发行版本 4，必须安装 Directory Preparation Tool 软件包：

- 对于 Solaris 平台，将该 DPT 软件包作为起源修补程序安装，其中包含全部 Directory Preparation Tool 软件。然后应用修补程序以升级到发行版本 4。
- 对于 Linux 平台，将直接安装发行版本 4 软件包。

从发行版本 2 的 Directory Preparation Tool 升级至发行版本 4 与第 212 页中的“从 Java ES 发行版本 3 升级 Directory Preparation Tool”中所述的过程相似。升级前和升级后的注意事项相同，只需将引用发行版本 3 的地方用发行版本 2 代替。不过，在以下各小节中介绍了具体的升级过程。

发行版本 2 升级过程 (Solaris)

下述过程适用于安装在 Directory Server 所驻留的计算机上的 Directory Preparation Tool。

1. 根据下表，获取所需的起源修补程序：

表 12-5 在 Solaris 上升级 Directory Preparation Tool 的起源修补程序¹

说明	SPARC Solaris 8、9 和 10	X86 Solaris 9 和 10
Directory Preparation Tool (DSSETUP)	118242-03	118243-03

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

可从以下网址将修补程序下载到 /tmp：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 应用表 12-5 中的 Directory Preparation Tool 起源修补程序。

```
patchadd patch_ID
```

4. 根据表 12-3，获取所需的升级修补程序。

可从以下网址将修补程序下载到 /tmp：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

5. 应用表 12-3 中相应的 Directory Preparation Tool 升级修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 针对 Directory Server 运行 Directory Preparation Tool。
 - a. 确认 Directory Server 正在运行。
 - b. 将目录更改为 Directory Preparation Tool 所在的位置。

```
cd DirPrepTool-base/sbin
```
 - c. 运行 Directory Preparation Tool (comm_dssetup.pl perl 脚本)。

```
perl comm_dssetup.pl
```

提供该脚本要求的参数。

发行版本 2 升级过程 (Linux)

下述过程适用于安装在 Directory Server 所驻留的计算机上的 Directory Preparation Tool。

1. 以 root 身份登录，或成为超级用户。

```
su -
```
2. 使用第 216 页中的表 12-4 中的修补程序号和 RPM 名称获取所需的升级修补程序。

可从以下网址将修补程序下载到 /tmp：
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>
3. 应用第 216 页中的表 12-4 中用于 Directory Preparation Tool 的升级 RPM。

```
rpm -Uvh sun-comms-dssetup-6.3-2.5.i386.rpm
```
4. 确认升级成功：

```
rpm -q sun-comms-dssetup
```

应该返回 RPM 的新版本号。

5. 针对 Directory Server 运行 Directory Preparation Tool。
 - a. 确认 Directory Server 正在运行。
 - b. 将目录更改为 Directory Preparation Tool 所在的位置。

```
cd DirPrepTool-base/sbin
```
 - c. 运行 Directory Preparation Tool (comm_dssetup.pl perl 脚本)。

```
perl comm_dssetup.pl
```

提供该脚本要求的参数。

Messaging Server

本章介绍如何将 Messaging Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Messaging Server 6.2 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 224 页中的“Messaging Server 升级概述”
- 第 227 页中的“从 Java ES 发行版本 3 升级 Messaging Server”
- 第 237 页中的“从 Java ES 发行版本 2 升级 Messaging Server”

注

本章中的文件位置是相对于一个称为 *MessagingServer-base* 的目录路径而指定的。在最初安装 Messaging Server 时，可能至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

MessagingServer-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWmsgsr
 - Linux: /opt/sun/messaging
-

Messaging Server 升级概述

本节将在以下方面对 Messaging Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Messaging Server](#)
- [Messaging Server 升级指导说明](#)
- [Messaging Server 数据](#)
- [兼容性问题](#)
- [Messaging Server 组件依赖性](#)

关于 Java ES 发行版本 4 Messaging Server

Java ES 发行版本 4 Messaging Server 主要修复了一些错误，相对于发行版本 3 并未新增任何主要的功能。

Messaging Server 升级指导说明

[表 13-1](#) 显示了所支持的升级至 Java ES 发行版本 4 的 Messaging Server 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 13-1 升级到 Java ES 发行版本 4：
Sun Java System Messaging Server 6.2 2005Q4 的途径

Java ES 发行版本	Messaging Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Messaging Server 5 2006.2Q1	直接升级： 通过应用修补程序执行。	配置文件和配置目录数据
发行版本 2	Sun Java System Messaging Server 6.1 2004Q2	直接升级： 通过应用修补程序执行。	配置文件和配置目录数据
发行版本 1	Sun ONE Messaging Server 6.0 (2003Q4)	不直接升级： 但可以按照 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237) 中的过程首先升级到发行版本 3。 然后从发行版本 3 升级到发行版本 4。	配置文件和配置目录数据

表 13-1 升级到 Java ES 发行版本 4:
Sun Java System Messaging Server 6.2 2005Q4 的途径 (续)

Java ES 发行版本	Messaging Server 版本	常规方法	必需的重新配置
Java ES 以前的发行版本	Sun ONE Messaging Server 5.2	不直接升级: 但可以按照 Java Enterprise System 2005Q1 升级与迁移指南 (http://docs.sun.com/doc/819-2237) 中的过程首先升级到发行版本 3。 然后从发行版本 3 升级到发行版本 4。	配置文件和配置目录数据

Messaging Server 数据

下表显示了可能受 Messaging Server 软件升级影响的数据类型。

表 13-2 Messaging Server 数据用途

数据类型	位置	用途
配置数据	本地配置目录: <i>MessagingServer-base</i> /config/msg.conf 和用于配置 Messaging Server Store、MTA、MMP、MEM (Web 邮件) 的许多其他配置文件	Messaging Server 组件的配置
配置数据	Directory Server 配置目录	Messaging Server 组件的配置
用户数据	Directory Server 用户/组目录	存储用以支持最终用户消息传送的用户属性
动态应用程序数据	Messaging Server Store: <i>MessagingServer-base</i> /	代表用户存储电子邮件消息、消息传输队列及相关信息
目录模式	Directory Server <i>/var/opt/mps/serverroot</i>	用于支持最终用户所需的用户属性

兼容性问题

发行版本 4 Messaging Server 没有任何接口变化。Messaging Server Store、MTA、MMP、MEM 组件以及 Messaging Server 的逻辑上独立的配置与早期版本向后兼容。

Messaging Server 组件依赖性

Messaging Server 对其他 Java ES 组件的依赖性会影响升级和重新配置 Messaging Server 软件的过程。例如，Messaging Server 接口或功能上的变化可能要求 Messaging Server 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Messaging Server 对以下 Java ES 组件具有依赖性：

- **共享组件。** Messaging Server 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Directory Server。** Messaging Server 将消息传送所需的配置数据和用户数据存储在 Directory Server 中。因此，Messaging Server 升级可能需要目录模式的扩展。
- **Directory Preparation Tool。** Messaging Server 使用 Directory Preparation Tool 准备 Directory Server，以支持 Messaging Server 功能。
- **Access Manager（可选）。** 对于支持用户单点登录访问基于 Web 的服务的软件解决方案，可将 Messaging Server 配置为使用 Access Manager 单点登录功能。
- **Delegated Administrator（可选）。** 当在 Directory Server 中置备用户，以使 Messaging Server 有权访问那些提供消息服务所需的用户数据时，Delegated Administrator 是首选实用程序。

从 Java ES 发行版本 3 升级 Messaging Server

本节包含关于将 Messaging Server 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Messaging Server 的升级](#)

简介

将 Java ES 发行版本 3 Messaging Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过向发行版本 3 的版本应用修补程序执行升级。通过运行两个数据配置实用程序并将配置数据导入到 Directory Server 来实现重新配置。
- **升级依赖性。** 虽然 Messaging Server 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Messaging Server 要求必须将 NSS、NSPR、LDAP C SDK、ICU 和 SASL 升级到发行版本 4。对于 Messaging Server 到发行版本 4 的升级，J2SE 的升级是可选的。

此外，如第 226 页中的“[Messaging Server 组件依赖性](#)”中所述，发行版本 4 Messaging Server 依赖于 Directory Server，并且选择性地依赖于 Access Manager。但是，这些是软性升级依赖性；对于 Messaging Server 到发行版本 4 的升级，这些组件的升级是可选的。

但是，发行版本 4 Messaging Server 对 Directory Preparation Tool 具有硬性升级依赖性；在为消息传送操作准备 Directory Server 时，发行版本 4 Directory Preparation Tool 是必需的。

- **向后兼容性。** 发行版本 4 的 Messaging Server 与发行版本 3 的版本向后兼容。
- **升级回滚。** Messaging Server 的发行版本 4 到发行版本 3 的升级回滚的实现方法是：首先删除对 Directory Server 所做的更改，然后删除对本地配置文件的更改，最后删除升级期间所应用的修补程序。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Messaging Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Messaging Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Messaging Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Messaging Server \(Solaris\)](#)
- [升级发行版本 3 Messaging Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Messaging Server 之前，应执行下述任务。

检验当前版本信息

可通过输入以下命令检验 Messaging Server 的当前版本：

```
MessagingServer-base/sbin/imsimta version
```

表 13-3 Messaging Server 版本检验输出

Java ES 发行版本	Messaging Server 版本号
发行版本 2	6.1
发行版本 3	6.2
发行版本 4	6.2p3

升级 Messaging Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，Messaging Server 仅对 SASL 共享组件和 Directory Preparation Tool 具有硬性升级依赖性。因此，升级 Messaging Server 所依赖的其他 Java ES 发行版本 3 组件是可选的。

但是，如果选择升级 Messaging Server 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Messaging Server 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 NSS、NSPR、LDAP C SDK、ICU 和 SASL 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Access Manager（可选）。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
4. **Directory Preparation Tool。** 配置发行版本 4 的 Messaging Server 之前，需要在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool。如果未在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool，则将 Directory Preparation Tool 升级至发行版本 4，并使用它修改并扩展 Directory Server 的模式（有关过程，参见第 209 页中的第 12 章“Directory Preparation Tool”）。

备份 Messaging Server 数据

将 Messaging Server 从发行版本 3 升级至发行版本 4 需要在本地配置文件和 Directory Server 配置目录中重新配置 Messaging Server。本地更改是可以回滚的，但最好备份配置目录，以备将来回滚发行版本 4 升级之需。

获取所需的配置信息和密码

Messaging Server 升级需要了解以下信息：

- 超级用户密码
- 目录管理员 DN 和密码

升级发行版本 3 Messaging Server (Solaris)

本节讨论影响 Messaging Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Messaging Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 所有对应于同一个已安装 Messaging Server 映像的 Messaging Server 组件（如 Messaging Server Store、MTA、MMP 或 MEM）都将会同时升级。在向安装的映像应用修补程序之前，应先将所有此类组件关闭。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Messaging Server 升级修补程序：

表 13-4 在 Solaris 上升级 Messaging Server 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Messaging Server 核心组件	118207-38	118208-38
Messaging Server 本地化版本	117784 -15	117785 -15

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于所有这样的 Messaging Server 组件，它们与即将进行升级的计算机上的同一个已安装 Messaging Server 映像相对应。

1. 根据表 13-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止所有正在运行的 Messaging Server 组件。

```
MessagingServer-base/sbin/stop-msg
```

4. 将必需的共享组件升级到发行版本 4（如果尚未执行此操作）。

第 229 页中的“升级 Messaging Server 依赖项”。

5. 应用表 13-4 中相应的 Messaging Server 修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 将配置数据从现有配置文件迁移至发行版本 4 配置文件。

- a. 创建候选配置文件。

```
cd MessagingServer-base/sbin
./patch-config MessagingServer-base/install/patch/patch_ID
```

此命令将备份现有配置文件，然后将这些文件中的配置参数值与发行版本 4 模板配置文件合并，从而创建新的发行版本 4 候选配置文件。您应该按修补程序 118209 自述文件的“特殊安装说明”部分所述，检查这些新文件是否存在可能的冲突。

此命令还将生成以下 ldif 文件（LDAP 目录导入文件）：

```
MessagingServer-base/lib/patch/cfgdir_diff.ldif
MessagingServer-base/lib/patch/ugdir_diff.ldif
```

- b. 安装发行版本 4 的候选配置文件，使它们成为活动配置。

```
./install-newconfig MessagingServer-base/install/patch/patch_ID
```

该命令会将发行版本 4 的新配置文件安装到它们在发行版本 4 的正确位置中。

注意：如果 install-newconfig 命令在 Solaris 10 平台上失败，请在运行该命令时将库路径设置为空：

```
LD_LIBRARY_PATH= ./install-newconfig
MessagingServer-base/install/patch/patch_ID
```

- c. 将第 231 页中的步骤 a 中生成的新配置数据导入到 Messaging Server 正在使用的 Directory Server 配置目录中。

切换到配置目录并使用 ldapmodify 命令导入 ldif 文件：

```
cd /MessagingServer-base/config/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
```

- 8. 重新启动在步骤 3 中停止的 Messaging Server 组件。

MessagingServer-base/sbin/start-msg

升级发行版本 3 Messaging Server (Linux)

本节讨论影响 Messaging Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Messaging Server 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 230 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 4 的 Messaging Server 升级修补程序：

表 13-5 在 Linux 上升级 Messaging Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
含 S/MIME 的 Messaging Server 核心软件	118209-38 <ul style="list-style-type: none"> • sun-messaging-server-6.1-12.38.i386.rpm
Messaging Server 本地化版本	117786-15 <ul style="list-style-type: none"> • sun-messaging-110n-Locale-6.1-8.15.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于所有这样的 Messaging Server 组件，它们与即将进行升级的计算机上的同一个已安装 Messaging Server 映像相对应。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 13-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止所有正在运行的 Messaging Server 组件。

```
MessagingServer-base/sbin/stop-msg
```

4. 将必需的共享组件升级到发行版本 4（如果尚未执行此操作）。

第 229 页中的“升级 Messaging Server 依赖项”。

5. 应用表 13-5 中用于 Messaging Server 的 RPM。

例如:

```
rpm -Fvh sun-messaging-server-6.1-12.38.i386.rpm
```

6. 确认修补程序升级成功:

```
rpm -q sun-messaging-server
```

应该返回 RPM 的新版本号。

7. 将配置数据从现有配置文件迁移至发行版本 4 配置文件。

a. 创建候选配置文件。

```
cd MessagingServer-base/sbin
./patch-config MessagingServer-base/install/patch/patch_ID
```

此命令将备份现有配置文件，然后将这些文件中的配置参数值与发行版本 4 模板配置文件合并，从而创建新的发行版本 4 候选配置文件。您应该按修补程序 118209 自述文件的“特殊安装说明”部分所述，检查这些新文件是否存在可能的冲突。

此命令还将生成以下 ldif 文件（LDAP 目录导入文件）：

```
MessagingServer-base/lib/patch/cfgdir_diff.ldif
MessagingServer-base/lib/patch/ugdir_diff.ldif
```

b. 安装发行版本 4 的候选配置文件，使它们成为活动配置。

```
./install-newconfig MessagingServer-base/install/patch/patch_ID
```

该命令会将发行版本 4 的新配置文件安装到它们在发行版本 4 的正确位置中。

c. 将第 234 页中的步骤 a 中生成的新配置数据导入到 Messaging Server 正在使用的 Directory Server 配置目录中。

切换到配置目录并使用 ldapmodify 命令导入 ldif 文件：

```
cd /MessagingServer-base/config/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
```

8. 重新启动在步骤 3 中停止的 Messaging Server 组件。

```
MessagingServer-base/sbin/start-msg
```

检验升级

可通过输入以下命令检验 Messaging Server 的当前版本：

```
MessagingServer-base/sbin/imsimta version
```

还可检查在启动 Messaging Server 组件时显示的标题

参见第 228 页中的表 13-3 确认输出值。

升级后的任务

除第 230 页中的“升级过程 (Solaris)”和第 233 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Messaging Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Messaging Server 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。该过程将回滚重新配置并删除修补程序。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 停止所有正在运行的 Messaging Server 组件。

```
MessagingServer-base/sbin/stop-msg
```

3. 回滚对 Messaging Server 正在使用的 Directory Server 配置目录所做的更改。

将目录替换为在开始升级过程之前所备份的升级前目录（参见第 229 页中的“备份 Messaging Server 数据”）。

4. 回滚在[第 231 页](#)中的[步骤 7](#)中执行的重新配置。

```
cd MessagingServer-base/sbin  
./uninstall-newconfig MessagingServer-base/install/patch/patch_ID
```

5. 删除[第 230 页](#)中的[表 13-4](#)中的修补程序。

```
patchrm patch_ID
```

6. 重新启动在[步骤 2](#)中停止的 Messaging Server 组件。

```
MessagingServer-base/sbin/start-msg
```

多实例升级

在一些部署体系结构中，Messaging Server 被部署在多个计算机系统上，以提供高可用性和可伸缩性。例如，您可以在多台计算机上运行 Messaging Server MTA 或 Messaging Server MMP 组件，并使用一个负载均衡器来分配负载。还可以在 Sun Cluster 环境中运行 Messaging Server Store 组件，以提供高可用性。

就 Messaging Server 的负载均衡实例而言，可执行滚动升级，期间您可依序升级各个 Messaging Server 实例而不中断服务。在升级每个 Messaging Server 实例时，其他实例仍可保持运行。在各个 MS 子组件（MS Store、MTA、MMP、MEM）分别部署于不同计算机上的部署体系结构中，组件的升级是从后端层（MS Store）开始，逐步达到前端层（如 MEM）。请按[第 228 页](#)中的“[发行版本 3 Messaging Server 的升级](#)”中所述执行每个实例的升级。

就在群集环境中运行的 Messaging Server 实例而言，这些实例共享同一个配置。因此，需要向每个实例应用 Messaging Server 升级修补程序，但向所有实例应用完修补程序后，仅需执行一次升级过程的重新配置部分。

从 Java ES 发行版本 2 升级 Messaging Server

将 Java ES 2004Q2（发行版本 2）Messaging Server 升级至发行版本 4 的过程与将发行版本 3 Messaging Server 升级至发行版本 4 大致相同，下文说明了其中的几处例外。

升级 Messaging Server 依赖项

与从发行版本 3 升级相比，从发行版本 2 到发行版本 4 的升级前任务应包括将 Messaging Server 所依赖的所有共享组件（参见第 38 页中的表 1-6）和所有本地驻留的产品组件升级到发行版本 4：

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** Directory Server 很少与 Messaging Server 驻留在相同的计算机上，但是，第 110 页中的“从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Access Manager（可选）。** 第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
4. **Directory Preparation Tool。** Directory Preparation Tool 很少与 Messaging Server 驻留在相同的计算机上，但是，第 218 页中的“从 Java ES 发行版本 2 升级 Directory Preparation Tool”中提供了升级 Directory Preparation Tool 和在 Directory Server 上运行 Directory Preparation Tool 的说明。

发行版本 2 Messaging Server 的升级

将 Messaging Server 从发行版本 2 升级至发行版本 4 的过程视操作系统平台而定。

升级发行版本 2 Messaging Server (Solaris)

要将发行版本 2 的 Messaging Server 升级至发行版本 4，可按照第 237 页中的“升级发行版本 2 Messaging Server (Solaris)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

升级发行版本 2 Messaging Server (Linux)

下述过程适用于所有这样的 Messaging Server 组件，它们与即将进行升级的计算机上的同一个已安装 Messaging Server 映像相对应。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 停止所有正在运行的 Messaging Server 组件。

```
MessagingServer-base/sbin/stop-msg
```

3. 将必需的共享组件升级到发行版本 4（如果尚未执行此操作）。

参见第 237 页中的“升级 Messaging Server 依赖项”。

4. 卸载发行版本 2 RPM 软件包。

```
rpm -e --noscripts sun-messaging-lib-6.1-9 \  
                    sun-messaging-store-6.1-9 \  
                    sun-messaging-install-6.1-9 \  
                    sun-messaging-core-6.1-9 \  
                    sun-messaging-mmp-6.1-9 \  
                    sun-messaging-sieveui-6.1-9 \  
                    sun-messaging-webmail-6.1-9 \  
                    sun-messaging-core-en-6.1-9 \  
                    sun-messaging-mta-6.1-9
```

5. 安装第 232 页中的表 13-5 中用于 Messaging Server 的 RPM。

```
rpm -i sun-messaging-server-6.1-12.38.i386.rpm
```

6. 确认修补程序升级成功：

```
rpm -q sun-messaging-server
```

应该返回新安装的 RPM 的版本号。

7. 在其他位置保存旧的发行版本 2 配置。

配置文件位于：*MessagingServer-base*/config

8. 运行 Messaging Server 配置程序。

```
cd MessagingServer-base/sbin  
./configure
```

9. 手动将发行版本 2 配置值和新的发行版本 4 配置条目合并在一起。
10. 重新启动在步骤 2 中停止的 Messaging Server 组件。

```
MessagingServer-base/sbin/start-msg
```

有关更多详细信息（例如，使用 `configutil` 命令更改 HTTP 端口的信息），参见修补程序 118209-38 自述文件的“特殊安装说明”部分。

检验升级

可通过输入以下命令检验 Messaging Server 的当前版本：

```
MessagingServer-base/sbin/imsimta version
```

还可检查在启动 Messaging Server 组件时显示的标题。

参见第 228 页中的表 13-3 确认输出值。

升级后的任务

除第 230 页中的“升级过程 (Solaris)”和第 233 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

Calendar Server

本章介绍如何将 Calendar Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Calendar Server 6.2 2005Q4。其中概括介绍了 Java ES 发行版本 4 所支持的不同升级途径的升级问题和过程。本章内容涵盖在 Solaris 和 Linux 操作系统上的升级：

- [第 242 页中的“Calendar Server 升级概述”](#)
- [第 244 页中的“从 Java ES 发行版本 3 升级 Calendar Server”](#)
- [第 251 页中的“从 Java ES 发行版本 2 升级 Calendar Server”](#)

注 本章中的文件位置是相对于一个称为 *CalendarServer-base* 的目录路径而指定的。在最初安装 Calendar Server 时，至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

CalendarServer-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWics5
 - Linux: /opt/sun/calendar
-

Calendar Server 升级概述

本节将在以下方面对 Calendar Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Calendar Server](#)
- [Calendar Server 升级指导说明](#)
- [Calendar Server 数据](#)
- [兼容性问题](#)
- [Calendar Server 组件依赖性](#)

关于 Java ES 发行版本 4 Calendar Server

Java ES 发行版本 4 Calendar Server 主要修复了一些错误，相对于发行版本 3 并未新增任何主要的功能。

Calendar Server 升级指导说明

[表 14-1](#) 显示了所支持的升级至 Java ES 发行版本 4 的 Calendar Server 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 14-1 升级到 Java ES 发行版本 4：Sun Java System Calendar Server 6.2 2005Q4 的途径

Java ES 发行版本	Calendar Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Calendar Server 6 2005Q1	直接升级： 通过应用修补程序执行。	无
发行版本 2	Sun Java System Calendar Server 6 2004Q2	直接升级： 通过应用修补程序执行。	配置文件
发行版本 1	Sun ONE Calendar Server 6.0 (2003Q4)	直接升级未经验证： 但可通过应用修补程序执行。	配置文件
Java ES 以前的发行版本	所有先前版本	不直接升级。	

Calendar Server 数据

下表显示了可能受 Calendar Server 软件升级影响的数据类型。

表 14-2 Calendar Server 数据用途

数据类型	位置	用途
配置数据	<code>etc/CalendarServer-base/cal/config/ics.conf</code>	Calendar Server 的配置
动态应用程序数据	Calendar Server 数据库： <code>/var/CalendarServer-base/csdb</code>	代表用户存储日历条目
目录模式	Directory Server 用户/组目录	用于支持最终用户所需的用户属性

兼容性问题

发行版本 4 的 Calendar Server 的接口没有任何变化。Calendar Server 与早期版本向后兼容。

Calendar Server 组件依赖性

Calendar Server 对其他 Java ES 组件的依赖性会影响升级和重新配置 Calendar Server 软件的过程。例如，Calendar Server 接口或功能上的变化可能要求 Calendar Server 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Calendar Server 对以下 Java ES 组件具有依赖性：

- **共享组件。** Calendar Server 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Directory Server。** Calendar Server 访问 Directory Server 中存储的用户数据。因此，Calendar Server 升级可能需要目录模式的扩展。
- **Directory Preparation Tool。** Calendar Server 使用 Directory Preparation Tool 准备目录，以支持 Calendar Server 功能。
- **Access Manager（可选）。** 对于支持用户单点登录访问基于 Web 的服务的软件解决方案，可将 Calendar Server 配置为使用 Access Manager 单点登录功能。

- **Messaging Server (可选)**。可对 Calendar Server 进行配置，使之能使用 Messaging Server 提供日历事件的消息传送通知。
- **Delegated Administrator (可选)**。当在 Directory Server 中置备用户，以使 Calendar Server 能访问提供日历服务所需的用户数据时，Delegated Administrator 是首选实用程序。

从 Java ES 发行版本 3 升级 Calendar Server

本节包含关于将 Calendar Server 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Calendar Server 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Calendar Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法**。通过将修补程序应用到发行版本 3 的版本执行升级。
- **升级依赖性**。虽然 Calendar Server 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 的 Calendar Server 只需将 SASL 升级至发行版本 4。相对于 Calendar Server 的升级，其他共享组件的升级是可选的。

此外，如第 243 页中的“[Calendar Server 组件依赖性](#)”中所述，发行版本 4 Calendar Server 依赖于 Directory Server，并且选择性地依赖于 Access Manager。但是，这些是软性升级依赖性；对于 Calendar Server 到发行版本 4 的升级，这些组件的升级是可选的。

但是，发行版本 4 的 Calendar Server 对 Directory Preparation Tool 具有硬性升级依赖性；为日历操作准备 Directory Server 需要发行版本 4 的 Directory Preparation Tool。

- **向后兼容性**。发行版本 4 的 Calendar Server 与发行版本 3 的版本向后兼容。

- **升级回滚。** Calendar Server 的发行版本 4 到发行版本 3 的升级回滚可通过删除升级期间所应用的修补程序实现。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Calendar Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Calendar Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Calendar Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Calendar Server \(Solaris\)](#)
- [升级发行版本 3 Calendar Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Calendar Server 之前，应执行下述任务。

检验当前版本信息

可使用以下命令检验 Calendar Server 的当前版本：

```
Solaris:  
cd CalendarServer-base/cal/bin  
./cshttpd -#
```

```
Linux:  
cd CalendarServer-base/bin  
./cshttpd -#
```

注意：如果 `cshttpd` 命令在 Solaris 10 平台上失败，则在运行该命令时将库路径设置为空：

```
LD_LIBRARY_PATH= ./cshttpd -#
```

表 14-3 Calendar Server 版本检验输出

Java ES 发行版本	Calendar Server 版本号
发行版本 2	2004Q2
发行版本 3	2005Q1

应用必需的操作系统修补程序

在 Solaris 10 操作系统平台上，需要应用操作系统修补程序以执行 Delegated Administrator 升级过程（参见第 31 页中的“所需的操作系统修补程序”）。

升级 Calendar Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，Calendar Server 仅对 Directory Preparation Tool 具有硬性升级依赖性。因此，Calendar Server 所依赖的其他 Java ES 发行版本 3 组件的升级是可选的。

但是，如果选择升级 Calendar Server 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Calendar Server 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Access Manager（可选）。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
4. **Directory Preparation Tool。** 配置发行版本 4 的 Calendar Server 之前，需要在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool。如果未在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool，则将 Directory Preparation Tool 升级至发行版本 4，并使用它修改并扩展 Directory Server 的模式（有关过程，参见第 209 页中的第 12 章“Directory Preparation Tool”）。

备份 Calendar Server 数据

将 Calendar Server 从发行版本 3 升级至发行版本 4 时，无需重新配置 Calendar Server。但是，为安全起见，可备份以下位置的 Calendar Server Store:

```
/var/CalendarServer-base/csdb
```

获取所需的配置信息和密码

不需要任何与当前安装版本有关的特殊信息，但是必须以超级用户身份登录才能执行升级。

升级发行版本 3 Calendar Server (Solaris)

本节讨论影响 Calendar Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Calendar Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 为安装的映像应用修补程序时，应关闭 Calendar Server。
- 在不同的 Calendar Server 子组件分别驻留在不同计算机上的体系结构中（例如，Calendar Server 后端存储在一台计算机上，Calendar Server 前端进程（如 cshttpd）在另一台计算机上），必须在所有此类计算机上均执行升级。
- Calendar Server 升级适用于使用同一已安装映像的单台计算机上的多个 Calendar Server 子组件。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Calendar Server 升级修补程序：

表 14-4 在 Solaris 上升级 Calendar Server 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Calendar Server 核心组件	116577-24	116578-24
Calendar Server 本地化版本	117010 -23	117011 -23

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低修订版本号。如果有较新的修订版本可用，则使用较新的修订版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Calendar Server。

1. 根据表 14-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果 Calendar Server 正在运行，则将其停止。

```
CalendarServer-base/cal/sbin/stop-cal
```

4. 如果未将 SASL 共享组件及希望升级的任何其他共享组件升级至发行版本 4，请执行升级此操作。

参见第 246 页中的“升级 Calendar Server 依赖项”。

5. 应用表 14-4 中相应的 Calendar Server 修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功:

```
showrev -p | grep ics
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 重新启动在步骤 3 中停止的 Calendar Server。

```
CalendarServer-base/cal/sbin/start-cal
```

升级发行版本 3 Calendar Server (Linux)

本节讨论影响 Calendar Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Calendar Server 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 247 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 4 的 Calendar Server 升级修补程序：

表 14-5 在 Linux 上升级 Calendar Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Calendar Server 核心组件	117851-24 <ul style="list-style-type: none"> • sun-calendar-core-6.2-10.7.i386.rpm • sun-calendar-api-6.2-10.7.i386.rpm
Calendar Server 语言环境	117852-23 <ul style="list-style-type: none"> • sun-calendar-core-<i>Locale</i>-6.2-10.3.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Calendar Server。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 14-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果 Calendar Server 正在运行，则将其停止。

```
CalendarServer-base/sbin/stop-cal
```

4. 如果未将 SASL 共享组件及希望升级的任何其他共享组件升级至发行版本 4，请执行升级此操作。

参见第 246 页中的“升级 Calendar Server 依赖项”。

5. 应用表 14-5 中用于 Calendar Server 的 RPM。

```
rpm -Fvh sun-calendar-core-Locale-6.2-10.3.i386.rpm
```

```
rpm -Fvh sun-calendar-core-6.2-10.7.i386.rpm
```

```
rpm -Fvh sun-calendar-api-6.2-10.7.i386.rpm
```

6. 确认修补程序升级成功：

```
rpm -qa | grep sun-calendar
```

应返回 RPM 的新版本号。

7. 重新启动在步骤 3 中停止的 Calendar Server。

```
CalendarServer-base/sbin/start-cal
```

检验升级

通过确认是否已正确应用升级修补程序来检验 Calendar Server 到发行版本 4 的升级。第 248 页中的“升级过程 (Solaris)”和第 249 页中的“升级过程 (Linux)”中包括此检验的步骤。

升级后的任务

除第 248 页中的“升级过程 (Solaris)”和第 249 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Calendar Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Calendar Server 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 停止 Calendar Server。

```
CalendarServer-base/cal/sbin/stop-cal
```

3. 删除第 247 页中的表 14-4 中的修补程序。

```
patchrm patch_ID
```

4. 重新启动 Calendar Server。

```
CalendarServer-base/cal/sbin/start-cal
```

多实例升级

在一些部署体系结构中，Calendar Server 被部署在多个计算机系统上，以提供高可用性。例如，在 Sun Cluster 环境中运行 Calendar Server Store 组件可以提供高可用性。

对于在群集环境中运行的 Calendar Server 实例，这些实例共享同一配置。需要将 Calendar Server 升级修补程序应用到每个实例，对于从发行版本 3 升级到发行版本 4，无需重新进行配置。

从 Java ES 发行版本 2 升级 Calendar Server

除了升级前的任务应包括将所有共享组件（参见第 38 页中的表 1-6）以及对 Calendar Server 具有依赖性的所有驻留在本地的产品组件升级至发行版本 4 之外，将 Java ES 2004Q2（发行版本 2）Calendar Server 升级至发行版本 4 的过程与将发行版本 3 的 Calendar Server 升级至发行版本 4 的过程几乎相同：

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** Directory Server 很少与 Calendar Server 驻留在相同的计算机上，但是，第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Access Manager（可选）。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
4. **Directory Preparation Tool。** Directory Preparation Tool 很少与 Calendar Server 驻留在相同的计算机上，但是，第 209 页中的第 12 章“Directory Preparation Tool”中提供了升级 Directory Preparation Tool 和在 Directory Server 上运行 Directory Preparation Tool 的说明。

要将发行版本 2 的 Calendar Server 升级至发行版本 4，可按照第 244 页中的“从 Java ES 发行版本 3 升级 Calendar Server”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

此外，发行版本 2 升级至发行版本 4 需要执行升级后任务，即配置 Calendar Server 热备份，该任务通过将热备份参数添加到 Calendar Server ics.conf 配置文件完成。此升级后重新配置的说明可在以下位置找到：

<http://docs.sun.com/doc/819-3569/6n5pgriq6?a=view>

Communications Express

本章介绍如何将 Communications Express 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Communications Express 6.2 2005Q4。其中概括介绍了 Java ES 发行版本 4 所支持的不同升级途径的升级问题和过程。本章内容涵盖在 Solaris 和 Linux 操作系统上的升级：

- 第 254 页中的“Communications Express 升级概述”
- 第 257 页中的“从 Java ES 发行版本 3 升级 Communications Express”
- 第 266 页中的“从 Java ES 发行版本 2 升级 Communications Express”

注 本章中的文件位置是相对于一个称为 *CommsExpress-base* 的目录路径而指定的。在最初安装 Communications Express 时，可能至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

CommsExpress-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWuwc
 - Linux: /opt/sun/uwc
-

Communications Express 升级概述

本节将在以下方面对 Communications Express 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Communications Express](#)
- [Communications Express 升级指导说明](#)
- [Communications Express 数据](#)
- [兼容性问题](#)
- [Communications Express 组件依赖性](#)

关于 Java ES 发行版本 4 Communications Express

Java ES 发行版本 4 的 Communications Express 主要修复了一些错误。相对于版本 3 新增了少数功能：邮件过滤支持、通讯录共享和代理验证。

Communications Express 升级指导说明

表 15-1 显示了所支持的升级至 Java ES 发行版本 4 的 Communications Express 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 15-1 升级到 Java ES 发行版本 4：
Sun Java System Communications Express 6.2 2005Q4 的途径

Java ES 发行版本	Communications Express 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Communications Express 6 2005Q1	直接升级： 通过应用修补程序和重新配置 Messaging Server 组件执行。	配置文件
发行版本 2	Sun Java System Communications Express 6 2004Q2	直接升级： 通过应用修补程序和重新配置 Messaging Server 组件执行。	配置文件
发行版本 1	无	无升级	
Java ES 以前的发行版本	无	无升级	

Communications Express 数据

下表显示了可能受 Communications Express 软件升级影响的数据类型。

表 15-2 Communications Express 数据用途

数据类型	位置	用途
配置数据	本地配置目录 var/CommsExpress-base/WEB-INF/config/uwcauth.properties var/CommsExpress-base/WEB-INF/config/uwccconfig.properties var/CommsExpress-base/WEB-INF/config/uwclogging.properties MessagingServer-base/config/msg.conf 和其他配置文件用于配置 Messaging Server MEM (webmail)	配置 Communications Express, 包括 Messaging Server MEM (webmail)
Web 容器配置	Web Server: server.policy 和 server.xml 文件, 位于 WebServer-base/https-hostname/config Application Server (Java ES 发行版本 3 和 4): server.policy 和 domain.xml 文件, 位于 AppServer8Config-base/domains/domainName/config Application Server (Java ES 发行版本 2): server.policy 和 server.xml 文件, 位于 AppServer7Config-base/domains/domainName/config	Communications Express Web 容器实例的配置
目录模式	Directory Server 用户/组目录	用于支持最终用户所需的用户属性

兼容性问题

发行版本 4 Communications Express 的接口没有任何变化, 并与早期版本向后兼容。

Communications Express 组件依赖性

Communications Express 对其他 Java ES 组件的依赖性会影响升级和重新配置 Communications Express 软件的过程。例如，Communications Express 接口或功能上的变化可能要求 Communications Express 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Communications Express 对以下 Java ES 组件具有依赖性：

- **共享组件。** Communications Express 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Web 容器。** Communications Express 依赖于 Web 容器服务，这些服务可由 Java ES Web Server 或 Java ES Application Server 提供。
- **Access Manager（或 Access Manager SDK）。** Communications Express 依赖 Access Manager 为最终用户提供验证和授权服务，包括单点登录。如果 Access Manager 运行在远程计算机上，则 Access Manager SDK 必须在本地可用。
- **Messaging Server。** Communications Express 用于提供对 Messaging Server 实现基于 Web 的访问。事实上，Communications Express 直接使用 Messaging Server MEM 组件访问其他 Messaging Server 后端组件，如 Messaging Server Store 和 MTA 组件。
- **Calendar Server。** Communications Express 用于提供对 Calendar Server 实现基于 Web 的访问。
- **Directory Server。** Communications Express 存储配置数据，同时还访问存储在 Directory Server 中的用户数据。因此，Communications Express 升级可能需要升级 Directory Server 或扩展目录模式。
- **Directory Preparation Tool。** Communications Express 使用 Directory Preparation Tool 准备 Directory Server，以支持 Communications Express 的功能。因此，Communications Express 升级可能依赖目录准备以支持新功能。

从 Java ES 发行版本 3 升级 Communications Express

本节包含关于将 Communications Express 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 2005Q4（发行版本 4）的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Communications Express 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Communications Express 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 3 版本执行升级。使用两个配置实用程序并通过将配置数据导入 Directory Server 来实现重新配置所包括的 Messaging Server MEM 组件。
- **升级依赖性。** 虽然 Calendar Server 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Calendar Server 与这些组件在发行版本 3 中的版本兼容。因此，对于 Calendar Server 到发行版本 4 的升级，这些共享组件的升级是可选的。

此外，如第 256 页中的“[Communications Express 组件依赖性](#)”中所述，发行版本 4 的 Communications Express 依赖于 Web 容器和 Access Manager。但是，这些是软性升级依赖性；对于 Communications Express 到发行版本 4 的升级，这些组件的升级是可选的。

然而，Communications Express 对 Calendar Server（Communication Express 为其提供基于 Web 的访问）和 Messaging Server（Communication Express 使用 Messaging Server MEM 组件也为其提供基于 Web 的访问）具有硬性升级依赖性。因此，必须先将 Calendar Server 和 Messaging Server 升级至发行版本 4，然后才能将 Communications Express 升级至发行版本 4。

此外，发行版本 4 的 Communications Express 对 Directory Preparation Tool 具有硬性升级依赖性；为 Communications Express 功能准备 Directory Server 需要发行版本 4 的 Directory Preparation Tool。

- **向后兼容性。** 发行版本 4 的 Communications Express 与发行版本 3 的版本向后兼容。

- **升级回滚。** Communications Express 的发行版本 4 至发行版本 3 的升级回滚可通过先删除对 Directory Server 所做的更改，删除对本地配置文件的更改，然后删除升级期间应用的修补程序来实现。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Communications Express 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Communications Express 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Communications Express 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Communications Express \(Solaris\)](#)
- [升级发行版本 3 Communications Express \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Communications Express 之前，应执行下述任务。

检验当前版本信息

通过访问 Communications Express 登录页面，可以检验 Communications Express 的当前版本，该登录页面会显示当前的版本号。

```
http://hostName:port/uwc/auth
```

表 15-3 Communications Express 版本检验输出

Java ES 发行版本	Communications Express 版本号
发行版本 2	Sun Java System Communications Express 6 2004Q2
发行版本 3	Sun Java System Communications Express 6 2005Q1
发行版本 4	Sun Java System Communications Express 6 2005Q4

升级 Communications Express 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，Communications Express 仅对 Messaging Server、Calendar Server 和 Directory Preparation Tool 具有硬性升级依赖性。因此，升级 Communications Express 所依赖的其他 Java ES 发行版本 3 组件是可选的。

但是，如果选择升级 Communications Express 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Communications Express 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
4. **Access Manager (Access Manager SDK)。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
5. **Directory Preparation Tool。** 配置发行版本 4 的 Communications Express 之前，需要在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool。如果未在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool，则将 Directory Preparation Tool 升级至发行版本 4，并使用它修改并扩展 Directory Server 的模式（有关过程，参见第 209 页中的第 12 章“Directory Preparation Tool”）。
6. **Messaging Server。** Messaging Server 组件需升级至发行版本 4 以支持发行版本 4 的 Communications Express。第 223 页中的第 13 章“Messaging Server”中提供了将 Messaging Server 升级至发行版本 4 的说明。
7. **Calendar Server。** Calendar Server 组件需升级至发行版本 4 以支持发行版本 4 的 Communications Express。第 241 页中的第 14 章“Calendar Server”中提供了将 Calendar Server 升级至发行版本 4 的说明。

备份 Communications Express 数据

Communications Express 从发行版本 3 升级至发行版本 4 需要重新配置 Messaging Server MEM。可回滚局部更改，因此无需备份任何数据。

获取所需的配置信息和密码

必须以超级用户身份登录才能执行升级。如果要将 Web Server 用作 Web 容器，则无需任何配置信息。但如果要将 Application Server 用作 Web 容器，则将需要 Application Server 管理员用户 ID 和密码。

升级发行版本 3 Communications Express (Solaris)

本节讨论影响 Communications Express 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Communications Express 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- Communications Express 包括许多组件，其中一些组件用于实现对 Calendar Server 的基于 Web 的访问，还包括一个 Messaging Server MEM 组件，用于实现对 Messaging Server 后端组件的基于 Web 的访问。
- 所有 Communications Express 组件应部署到相同的 Web 容器。将修补程序应用于安装的映像之前，应关闭该 Web 容器。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Communications Express 升级修补程序：

表 15-4 在 Solaris 上升级 Communications Express 的修补程序¹

说明	SPARC Solaris 8、9 和 10	X86 Solaris 9 和 10
Communications Express 核心组件	118540-21	118541-21
Communications Express 本地化版本	118042-16	118042-16

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于待更新的计算机上的所有 Communications Express 组件。

1. 根据表 15-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 通过停止其 Web 容器，停止 Communications Express。

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 升级所有您希望升级到发行版本 4 的共享组件（如果尚未执行此操作）。

参见第 259 页中的“升级 Communications Express 依赖项”。

5. 应用表 15-4 中相应的 Communications Express 修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功:

```
showrev -p | grep uwc
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 将配置数据从现有配置文件迁移至发行版本 4 的候选配置文件。

```
cd CommsExpress-base/sbin
```

```
./patch-config CommsExpress-base/install/patch/patch_ID
```

该命令将要求您提供当前的配置目录，然后备份现有配置文件。接下来，该命令会将这些文件中的配置参数值与发行版本 4 模板配置文件合并，创建新的发行版本 4 候选配置文件。您应当检查这些新文件是否存在可能的冲突，如修补程序自述文件中的“特殊安装说明”部分所述。

8. 安装发行版本 4 的候选配置文件，使它们成为活动配置。

```
./install-newconfig CommsExpress-base/install/patch/patch_ID
```

该命令会将发行版本 4 的新配置文件安装到它们在发行版本 4 的正确位置中。

9. 移除 Web 容器为 Communications Express 保持的 JSP 类高速缓存。
有关具体过程，参见 Web 容器（Web Server 或 Application Server）的文档。
10. 通过重新启动其 Web 容器，重新启动 Communications Express。

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

升级发行版本 3 Communications Express (Linux)

本节讨论影响 Communications Express 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Communications Express 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 260 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 4 的 Communications Express 升级修补程序：

表 15-5 在 Linux 上升级 Communications Express 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Communications Express 核心组件	118542-21 <ul style="list-style-type: none"> • sun-uwac-6.1-7.21.i386.rpm
Communications Express 本地化版本	118044-14 <ul style="list-style-type: none"> • sun-uwac-l10n-Locale-6.1-11.9.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于待更新的计算机上的所有 Communications Express 组件。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 15-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 通过停止其 Web 容器，停止 Communications Express。

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 升级所有您希望升级到发行版本 4 的共享组件（如果尚未执行此操作）。

参见第 259 页中的“升级 Communications Express 依赖项”。

5. 应用表 15-5 中用于 Communications Express 的 RPM。

例如:

```
rpm -Fvh sun-uwcc-6.1-7.21.i386.rpm
```

6. 确认修补程序升级成功:

```
rpm -qa | grep uwcc
```

输出应返回在步骤 5 中 RPM 的版本。

7. 将配置数据从现有配置文件迁移至发行版本 4 的候选配置文件。

```
cd CommsExpress-base/sbin  
./patch-config CommsExpress-base/install/patch/patch_ID
```

该命令将要求您提供当前的配置目录，然后备份现有配置文件。接下来，该命令会将这些文件中的配置参数值与发行版本 4 模板配置文件合并，创建新的发行版本 4 候选配置文件。您应当检查这些新文件是否存在可能的冲突，如修补程序自述文件中的“特殊安装说明”部分所述。

8. 安装发行版本 4 的候选配置文件，使它们成为活动配置。

```
./install-newconfig CommsExpress-base/install/patch/patch_ID
```

该命令会将发行版本 4 的新配置文件安装到它们在发行版本 4 的正确位置中。

9. 移除 Web 容器为 Communications Express 保持的 JSP 类高速缓存。

有关具体过程，参见 Web 容器（Web Server 或 Application Server）的文档。

10. 通过重新启动其 Web 容器，重新启动 Communications Express。

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName  
--user admin_ID --password password
```

检验升级

通过访问 Communications Express 登录页面，可以检验 Communications Express 的当前版本，该登录页面会显示当前的版本号。

```
http://hostName:port/uwc/auth
```

登录后，检查新邮件选项卡升级后的用户界面以及原来的电子邮件和日历事件（如果要使用那些频道）。

此外，还可以检查升级过程中各种步骤的日志文件：

```
CommsExpress-base/install/patch/118540-21.  
CommsExpress-base/patch-config_20050729164754.log  
CommsExpress-base/install-newconfig_20050729164838.log
```

升级后的任务

除第 261 页中的“升级过程 (Solaris)”和第 263 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Communications Express 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Communications Express 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。该过程将回滚重新配置并删除修补程序。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 通过停止其 Web 容器，停止 Communications Express。

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

3. 回滚对 Communications Express 正在使用的 Directory Server 配置目录所做的更改。
4. 回滚在第 261 页中的步骤 8 中已执行的重新配置。

```
cd CommsExpress-base/sbin
```

```
./uninstall-newconfig CommsExpress-base/install/patch/patch_ID
```

5. 删除第 260 页中的表 15-4 中的修补程序。

```
patchrm patch_ID
```

6. 通过重新启动其 Web 容器，重新启动 Communications Express。

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
```

```
--user admin_ID --password password
```

多实例升级

在一些部署体系结构中，Communications Express 被部署在多个计算机系统上，以提供可伸缩性和提高可用性。例如，您可以在多台计算机上运行 Communications Express 组件，并使用一个负载均衡器来分配负载。

就 Communications Express 负载均衡的实例而言，可执行滚动升级，期间您可依序升级各个 Communications Express 实例而不中断服务。在升级每个 Communications Express 实例时，其他实例仍可保持运行。可按第 258 页中的“[发行版本 3 Communications Express 的升级](#)”中所述升级每个实例。

从 Java ES 发行版本 2 升级 Communications Express

将 Java ES 2004Q2（发行版本 2）Communications Express 升级至发行版本 4 的过程与将发行版本 3 Communications Express 升级至发行版本 4 大致相同，下文说明了其中的几处例外。

注 本节适用于 Communications Express 部署在发行版本 2 的 Web Server Web 容器中时的情况，但不适用于其部署在发行版本 2 的 Application Server Web 容器中时的情况。目前不支持后一种情况。

升级 Communications Express 依赖项

升级 Communications Express 依赖项应包括将所有共享组件（参见第 38 页中的表 1-6）以及 Communications Express 所依赖的所有本地驻留的产品组件升级至发行版本 4，除这一点外，就升级前的任务而言，将 Java ES 发行版本 2 的 Communications Express 升级至发行版本 4 与将发行版本 3 的 Communications Express 升级至发行版本 4 几乎相同。

升级 Communications Express 依赖项时，它们应按以下顺序升级，并且应在升级 Communications Express 之前进行。您可以跳过任何可能已完成升级的依赖项。

1. **共享组件。** 第 47 页中的第 2 章“[升级 Java ES 共享组件](#)”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。

2. **Directory Server。** Directory Server 很少与 Communications Express 驻留在相同的计算机上，但是，第 110 页中的“从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 132 页中的“从 Java ES 发行版本 2 升级 Web Server”和第 169 页中的“从 Java ES 发行版本 2 升级 Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
4. **Access Manager (Access Manager SDK)。** 第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
5. **Directory Preparation Tool。** Directory Preparation Tool 很少与 Communications Express 驻留在相同的计算机上，但是，第 218 页中的“从 Java ES 发行版本 2 升级 Directory Preparation Tool”中提供了升级 Directory Preparation Tool 和在 Directory Server 上运行 Directory Preparation Tool 的说明。
6. **Messaging Server。** Messaging Server MTA 需升级至发行版本 4 以支持发行版本 4 的 Communications Express。第 237 页中的“从 Java ES 发行版本 2 升级 Messaging Server”中提供了将 Messaging Server 升级至发行版本 4 的说明。
7. **Calendar Server。** Calendar Server 很少与 Communications Express 驻留在相同的计算机上，但是，第 251 页中的“从 Java ES 发行版本 2 升级 Calendar Server”中提供了将 Calendar Server 升级至发行版本 4 的说明。

发行版本 2 Communications Express 的升级

将 Communications Express 从发行版本 2 升级至发行版本 4 的过程视您在其中部署 Communications Express 软件的 Web 容器而定。

升级发行版本 2 Communications Express: Web Server Web 容器

当部署到已经升级至发行版本 4 的 Web Server Web 容器中时，要将发行版本 2 的 Communications Express 升级至发行版本 4，可按照第 260 页中的“升级发行版本 3 Communications Express (Solaris)”或第 262 页中的“升级发行版本 3 Communications Express (Linux)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

升级发行版本 2 Communications Express: Application Server Web 容器

当部署到已升级至发行版本 4 的 Application Server Web 容器中时，目前尚不支持将发行版本 2 的 Communications Express 升级至发行版本 4。

Instant Messaging

本章介绍如何将 Instant Messaging 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Instant Messaging 7.0.1 2005Q4。其中概括介绍了 Java ES 发行版本 4 所支持的不同升级途径的升级问题和过程。本章内容涵盖在 Solaris 和 Linux 操作系统上的升级：

- [第 270 页中的“Instant Messaging 升级概述”](#)
- [第 272 页中的“从 Java ES 发行版本 3 升级 Instant Messaging”](#)
- [第 279 页中的“从 Java ES 发行版本 2 升级 Instant Messaging”](#)

注 本章中的文件位置是相对于一个称为 *InstantMessaging-base* 的目录路径而指定的。在最初安装 Instant Messaging 时，可能至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

InstantMessaging-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWiim
 - Linux: /opt/sun/im
-

Instant Messaging 升级概述

本节将在以下方面对 Instant Messaging 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Instant Messaging](#)
- [Instant Messaging 升级指导说明](#)
- [Instant Messaging 数据](#)
- [兼容性问题](#)
- [Instant Messaging 组件依赖性](#)

关于 Java ES 发行版本 4 Instant Messaging

Java ES 发行版本 4 Instant Messaging 主要修复了一些错误，相对于发行版本 3 并未新增任何主要的功能。

Instant Messaging 升级指导说明

表 16-1 显示了所支持的升级至 Java ES 发行版本 4 的 Instant Messaging 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 16-1 升级到 Java ES 发行版本 4：
Sun Java System Instant Messaging 7.0.1 2005Q4 的途径

Java ES 发行版本	Instant Messaging 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Instant Messaging 7.0 2005Q1	直接升级： 通过应用修补程序执行。	无
发行版本 2	Sun Java System Instant Messaging 6 2004Q2	直接升级： 使用 upgrade 实用程序执行。	配置数据
发行版本 1	Sun Java System Instant Messaging 6.1 (2003Q4)	直接升级未经验证： 但可使用 upgrade 实用程序执行。	配置数据
Java ES 以前的发行版本		不直接升级。	

Instant Messaging 数据

下表显示了可能受 Instant Messaging 软件升级影响的数据类型。

表 16-2 Instant Messaging 数据用途

数据类型	位置	用途
配置数据	本地配置目录 Solaris: /etc/opt/SUNWiim/default/config/iim.conf /etc/opt/SUNWiim/default/config/registration.properties Linux: /etc/opt/sun/im/default/config/iim.conf /etc/opt/sun/im/default/config/registration.properties	Instant Messaging 进程和注册属性的配置。
Instant Messaging 服务器资源	本地配置目录 InstantMessaging-base/html	由最终用户下载的自定义客户机文件，用以启动 Messenger 客户机。
动态数据	runtimeFilesDir/default/db 其中 runtimeFilesDir 在安装时指定： Solaris: 默认的 runtimeFilesDir 为: /var/opt/SUNWiim Linux: 默认的 runtimeFilesDir 为: /var/opt/sun/im	所有变量数据，如 Instant Messaging 数据库、日志文件和锁定文件。

兼容性问题

发行版本 4 Instant Messaging 的接口没有任何变化，并与早期版本向后兼容。

Instant Messaging 组件依赖性

Instant Messaging 对其他 Java ES 组件的依赖性会影响升级和重新配置 Instant Messaging 软件的过程。例如，Instant Messaging 接口或功能上的变化可能要求 Instant Messaging 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Instant Messaging 对以下 Java ES 组件具有依赖性：

- **共享组件。** Instant Messaging 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Web 容器。** Instant Messaging 依赖于 Web 容器服务，这些服务可由 Java ES Web Server 或 Java ES Application Server 提供。
- **Directory Server（可选）。** 可将 Instant Messaging 配置为在 Directory Server 中存储和访问用户数据。因此，Instant Messaging 升级可能需要目录模式的扩展。
- **Access Manager（可选）。** 对于支持用户单点登录访问基于 Web 的服务的软件解决方案，可将 Instant Messaging 配置为使用 Access Manager 单点登录功能。

从 Java ES 发行版本 3 升级 Instant Messaging

本节包含关于将 Instant Messaging 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 的以下信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Instant Messaging 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Instant Messaging 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过向发行版本 3 的版本应用修补程序执行升级。不要求重新配置 Instant Messaging。
- **升级依赖性。** 虽然 Instant Messaging 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 的 Instant Messaging 仅要求将 IM-SDK 升级至发行版本 4。对于 Instant Messaging 到发行版本 4 的升级，其他共享组件的升级是可选的。

此外，如第 271 页中的“[Instant Messaging 组件依赖性](#)”中所述，发行版本 4 的 Instant Messaging 依赖于 Web 容器，并且选择性地依赖于 Access Manager。但是，这些是软性升级依赖性；对于 Instant Messaging 到发行版本 4 的升级，Web 容器和 Access Manager 的升级是可选的。

- **向后兼容性。** 发行版本 4 的 Instant Messaging 与发行版本 3 的版本向后兼容。
- **升级回滚。** 在 Solaris 平台上，发行版本 4 到发行版本 3 的升级回滚可通过删除升级期间所应用的修补程序实现。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Instant Messaging 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Instant Messaging 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Instant Messaging 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Instant Messaging \(Solaris\)](#)
- [升级发行版本 3 Instant Messaging \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Instant Messaging 之前，应执行下述任务。

检验当前版本信息

可使用标准版检查实用程序检验 Instant Messaging 的当前版本

```
Solaris:
pkginfo -l SUNWiimin

Linux:
rpm -qa | grep sun-im
```

表 16-3 Instant Messaging 版本检验输出

Java ES 发行版本	Instant Messaging 版本号
发行版本 2	版本号 6.x
发行版本 3	版本号 7.0
发行版本 4	版本号 7.0.1

升级 Instant Messaging 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，Instant Messaging 仅对 IM-SDK 共享组件具有硬性升级依赖性。因此，Instant Messaging 所依赖的其他 Java ES 发行版本 3 组件的升级是可选的。

如果选择升级 Instant Messaging 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Instant Messaging 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 IM-SDK 和其他 Java ES 共享组件升级至发行版本 4 的说明。
2. **Web 容器软件。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
3. **Access Manager（可选）。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。

备份 Instant Messaging 数据

Instant Messaging 从发行版本 3 到发行版本 4 的升级并不修改配置数据。但是，为安全起见，最好在升级 Instant Messaging 之前，对 Instant Messaging 数据库及自定义的所有现有资源和配置文件进行备份。参见《Sun Java System Instant Messaging 7 2005Q4 Administration Guide》(<http://docs.sun.com/doc/819-2503>)。

获取所需的配置信息和密码

必须以超级用户身份登录才能执行升级。如果要将 Web Server 用作 Web 容器，则无需任何配置信息。但如果要将 Application Server 用作 Web 容器，则将需要 Application Server 管理员用户 ID 和密码。

升级发行版本 3 Instant Messaging (Solaris)

本节讨论影响 Instant Messaging 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Instant Messaging 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 向安装的映像应用修补程序时，应关闭 Instant Messaging 组件。
- 在不同的 Instant Messaging 子组件分别驻留在不同计算机上的体系结构中（例如，信使资源在一台计算机上，Instant Messaging 服务器在另一台计算机上，而 Instant Messaging 多路复用器又在另一台计算机上），必须在所有此类计算机上均执行升级。

- Instant Messaging 升级适用于使用同一已安装映像的单台计算机上的多个 Instant Messaging 子组件。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Instant Messaging 升级修补程序：

表 16-4 在 Solaris 上升级 Instant Messaging 的修补程序¹

说明	SPARC Solaris 8、9 和 10	X86 Solaris 9 和 10
Instant Messaging	118786-08	118787-08
Instant Messaging 本地化版本	119707-06	119707-06

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Instant Messaging。

1. 根据表 16-4，获取所需的修补程序。
 可从以下网址将修补程序下载到 /tmp：
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>
2. 以 root 身份登录，或成为超级用户。

```
su -
```
3. 如果 Instant Messaging 正在运行，则将其停止。

```
InstantMessaging-base/sbin/imadmin stop
```
4. 将 IM-SDK 共享组件及希望升级的任何其他共享组件升级至发行版本 4（如果尚未执行此操作）。
 参见第 274 页中的“升级 Instant Messaging 依赖项”。
5. 应用表 16-4 中相应的 Instant Messaging 修补程序。
 确保先应用 Instant Messaging 本地化修补程序 (119707)，然后再应用 Instant Messaging 基本修补程序。

```
patchadd patch_ID
```

6. 确认升级成功:

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 重新启动在步骤 3 中停止的 Instant Messaging 服务。

```
InstantMessaging-base/sbin/imadmin start
```

升级发行版本 3 Instant Messaging (Linux)

本节讨论影响 Instant Messaging 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Instant Messaging 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 274 页中的“升级注意事项 (Solaris)”）。

下表显示了用于 Linux 操作系统的发行版本 4 的 Instant Messaging 升级修补程序：

表 16-5 在 Linux 上升级 Instant Messaging 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Instant Messaging	118788-11 <ul style="list-style-type: none"> • sun-im-client-7.0-13.8.i386.rpm • sun-im-server-7.0-13.8.i386.rpm • sun-im-mux-7.0-13.8.i386.rpm • sun-im-olh-7.0-13.8.i386.rpm • sun-im-install-7.0-13.8.i386.rpm • sun-im-ident-7.0-13.8.i386.rpm • sun-im-apidoc-7.0-13.8.i386.rpm
Instant Messaging 本地化版本	119708-06 <ul style="list-style-type: none"> • sun-im-client-Locale-7.0-12.i386.rpm • sun-im-ident-Locale-7.0-12.i386.rpm • sun-im-install-Locale-7.0-12.i386.rpm • sun-im-olh-Locale-7.0-12.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Instant Messaging。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 16-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 如果 Instant Messaging 正在运行，则将其停止。

```
InstantMessaging-base/sbin/imadmin stop
```

4. 将 IM-SDK 共享组件及希望升级的任何其他共享组件升级至发行版本 4（如果尚未执行此操作）。

参见第 274 页中的“升级 Instant Messaging 依赖项”。

5. 应用表 16-5 中用于 Instant Messaging 的 RPM。

确保先应用 Instant Messaging 本地化修补程序 (119708)，然后再应用 Instant Messaging 基本修补程序。

```
rpm -Fvh sun-im-module-Locale-7.0-2.8.i386.rpm
```

```
rpm -Fvh sun-im-module-7.0-13.8.i386.rpm
```

6. 确认升级成功:

```
rpm -qa | grep sun-im
```

应返回 RPM 的新版本号。

7. 重新启动在步骤 3 中停止的 Instant Messaging 服务。

```
InstantMessaging-base/sbin/imadmin start
```

检验升级

通过确认是否已正确应用升级修补程序来检验 Instant Messaging 到发行版本 4 的升级。第 275 页中的“升级过程 (Solaris)”和第 277 页中的“升级过程 (Linux)”中包括此检验的步骤。

还可使用以下命令检查各个 Instant Messaging 子组件的状态：

```
InstantMessaging-base/sbin/imadmin status
```

或者可检查位于 *iim.instancevardir*/log 处的日志文件，

其中 *instancevardir* 在 *iim.conf* 文件中指定（有关路径，参见第 271 页中的表 16-2）。

升级后的任务

除第 275 页中的“升级过程 (Solaris)”和第 277 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Instant Messaging 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Instant Messaging 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 如果 Instant Messaging 正在运行，则将其停止。

```
InstantMessaging-base/sbin/imadmin stop
```

3. 删除第 275 页中的表 16-4 中的修补程序。

```
patchrm patch_ID
```

4. 重新启动在步骤 2 中停止的 Instant Messaging 服务。

```
InstantMessaging-base/sbin/imadmin start
```

多实例升级

多实例升级对发行版本 4 的 Instant Messaging 不适用。

从 Java ES 发行版本 2 升级 Instant Messaging

本节包含关于将 Instant Messaging 从 Java ES 2004Q2（发行版本 2）升级至 Java ES 2005Q4（发行版本 4）的信息。将发行版本 2 的 Instant Messaging 升级到发行版本 4 的过程完全不同于从发行版本 3 的 Instant Messaging 进行升级的过程。

本节涵盖了以下主题：

- [简介](#)
- [发行版本 2 Instant Messaging 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 2 Instant Messaging 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 使用升级实用程序执行升级，该实用程序可执行升级 Instant Messaging 软件所需的所有操作。
- **升级依赖性。** 将一台计算机上的任一 Java ES 组件从发行版本 2 升级都需要升级由该计算机托管的所有其他 Java ES 组件；不支持有选择地将 Java ES 组件从发行版本 2 升级至发行版本 4。特别是需要将 Instant Messaging、Web 容器和 Access Manager 使用的所有 Java ES 共享组件都升级到发行版本 4。
- **向后兼容性。** 发行版本 4 的 Instant Messaging 与发行版本 2 的版本向后兼容。
- **升级回滚。** 通过保存所有发行版本 2 软件和数据并手动恢复到发行版本 2 的版本，可实现从发行版本 4 到发行版本 2 的升级回滚。没有用于回滚此升级的实用程序。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Instant Messaging 的常规方法相同。升级过程包括任何特定于平台的详细信息。

发行版本 2 Instant Messaging 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Instant Messaging 从 Java ES 发行版本 2 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 2 Instant Messaging \(Solaris\)](#)
- [升级发行版本 2 Instant Messaging \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 Instant Messaging 之前，应执行下述任务。

检验当前版本信息

可使用标准版检查实用程序检验 Instant Messaging 的当前版本

```
Solaris:  
pkginfo -l SUNwiimin
```

```
Linux:  
rpm -qa | grep sun-im
```

参见第 273 页中的表 16-3 确认输出值。

升级 Instant Messaging 依赖项

Java ES 发行版本 4 不支持发行版本 2 和发行版本 4 的共享组件在一台计算机上共存。

因此，要求您将 Instant Messaging 所依赖的所有本地 Java ES 发行版本 2 组件都升级至发行版本 4。升级计算机上 Instant Messaging 所依赖的所有组件时，应按以下顺序升级，并且应在升级 Instant Messaging 之前进行。

1. **共享组件。** 必须将 Instant Messaging 所依赖的所有共享组件升级至发行版本 4。如果其他 Java ES 产品组件与 Instant Messaging 共存于一台计算机上，则必须将驻留于该计算机上的所有 Java ES 共享组件都进行升级。第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了有关将 Java ES 共享组件升级至发行版本 4 的说明。

2. **Web 容器软件。** 第 132 页中的“从 Java ES 发行版本 2 升级 Web Server”和第 169 页中的“从 Java ES 发行版本 2 升级 Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
3. **Access Manager (Access Manager SDK)。** 第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
4. **Directory Server。** Instant Messaging 很少依赖于本地 Directory Server，但在第 110 页中的“从 Java ES 发行版本 2 升级 Directory Server 和 Administration Server”中提供了将 Directory Server 升级到发行版本 4 的说明。

备份 Instant Messaging 数据

将 Instant Messaging 从发行版本 2 升级到发行版本 4 会修改配置数据和自定义内容。在升级 Instant Messaging 之前，最好对 Instant Messaging 数据库及所有现有资源和配置文件进行备份。有关更多信息，参见《Sun Java System Instant Messaging 7 2005Q4 Administration Guide》(<http://docs.sun.com/doc/819-2503>)。

获取所需的配置信息和密码

必须以超级用户身份登录才能执行升级。如果要将 Web Server 用作 Web 容器，则无需任何配置信息。但如果要将 Application Server 用作 Web 容器，则将需要 Application Server 管理员用户 ID 和密码。

升级发行版本 2 Instant Messaging (Solaris)

本节讨论影响 Instant Messaging 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Instant Messaging 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 在不同的 Instant Messaging 子组件分别驻留在不同计算机上的体系结构中（例如，信使资源在一台计算机上，Instant Messaging 服务器在另一台计算机上，而 Instant Messaging 多路复用器又在另一台计算机上），必须在所有此类计算机上均执行升级。但是，该升级适用于使用同一已安装映像的单台计算机上的多个 Instant Messaging 子组件。
- 将发行版本 2 的 Instant Messaging 软件升级到 Java ES 发行版本 4 可利用升级实用程序，该升级实用程序执行以下操作：
 - 创建一个用来存储工作文件的临时目录。
 - 收集和临时存储安装在系统上的所有软件包以前的软件包安装参数。
 - 关闭先前版本的 Instant Messaging 服务器。

- 安装新的软件包并修补现有的软件包。
- 安装 Instant Messaging 所需的所有新的共享组件包。
- 保存来自 IIM_DOCROOT 的以前的图形内容，并将其恢复到新的资源文件位置。
- 重新启动所有服务。
- 删除临时目录及其内容。

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Instant Messaging。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 将 IM-SDK 共享组件及希望升级的任何其他共享组件升级至发行版本 4（如果尚未执行此操作）。

参见第 280 页中的“升级 Instant Messaging 依赖项”。

3. 运行 Java ES 分发包 Instant Messaging 工具目录下的 upgrade 实用程序。

```
cd /Solaris_arch/Product/instant_messaging/Tools
./upgrade
```

upgrade 实用程序会创建一个日志文件，显示升级过程的进度。该日志文件位于以下位置：

```
/var/sadm/install/logs/Instant_Messaging_Upgrade.timestamp
```

其中 *timestamp* 的格式为 *yyyymmddhhss*。

4. （可选）必要时更改配置以使用自发行版本 2 之后所引入的新功能。有关配置信息，参见 Sun Java System Instant Messaging 管理指南 (<http://docs.sun.com/doc/819-1489>)。
5. 重新自定义信使资源。

如果已自定义信使资源，需要将这些自定义内容重新应用于以下文件：

```
InstantMessaging-base/html/Locale/imbrand.jar
InstantMessaging-base/html/Locale/imb[ssl].html|jnlp
```

请查阅按第 281 页中的“备份 Instant Messaging 数据”中所述保存的自定义文件。

6. 将信使资源重新部署到 Web 容器中。

```
InstantMessaging-base/lib/deployHtml -f webcontainerDeployLocation
```

其中 *webcontainerDeployLocation* 是要部署信使资源的目录位置。该位置通常取决于所使用的 Web 容器，例如：

Web Server

```
WebServer-base/docs/im
```

Application Server

```
AppServer8Config-base/nodeagents/hostname_domainName/instanceName/  
docroot/iim
```

升级发行版本 2 Instant Messaging (Linux)

本节讨论影响 Instant Messaging 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

在 Linux 平台上将 Instant Messaging 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 281 页中的“升级注意事项 (Solaris)”）。

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Instant Messaging。

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 将 IM-SDK 共享组件及希望升级的任何其他共享组件升级至发行版本 4（如果尚未执行此操作）。

参见第 280 页中的“升级 Instant Messaging 依赖项”。

3. 运行 Java ES 分发包 Instant Messaging 工具目录下的 upgrade 实用程序。

```
cd /Linux_x86/Product/instant_messaging/Tools  
./upgrade
```

upgrade 实用程序会创建一个日志文件，显示升级过程的进度。该日志文件位于以下位置：

```
/var/sadm/install/logs/Instant_Messaging_Upgrade.timestamp
```

其中 *timestamp* 的格式为 *yyyymmddhhss*。

4. （可选）必要时更改配置以使用自发行版本 2 之后所引入的新功能。有关配置信息，参见 Sun Java System Instant Messaging 管理指南 (<http://docs.sun.com/doc/819-1489>)。

5. 重新自定义信使资源。

如果已自定义信使资源，需要将这些自定义内容重新应用于以下文件：

```
InstantMessaging-base/html/Locale/imbrand.jar
```

```
InstantMessaging-base/html/Locale/imb[ssl].html|jnlp
```

请查阅按第 281 页中的“备份 Instant Messaging 数据”中所述保存的自定义文件。

6. 将信使资源重新部署到 Web 容器中。

```
InstantMessaging-base/lib/deployHtml -f webcontainerDeployLocation
```

其中 *webcontainerDeployLocation* 是要部署信使资源的目录位置。该位置通常取决于所使用的 Web 容器，例如：

Web Server

```
WebServer-base/docs/im
```

Application Server

```
AppServer8Config-base/nodeagents/hostname_domainName/instanceName/  
docroot/iim
```

检验升级

可使用以下命令检查各个 Instant Messaging 子组件的状态：

```
InstantMessaging-base/sbin/imadmin status
```

或者可检查位于 *iim.instancevardir/log* 处的日志文件，

其中 *instancevardir* 在 *etc/InstantMessaging-base/config/iim.conf* 文件中指定。

升级后的任务

除第 282 页中的“升级过程 (Solaris)”和第 283 页中的“升级过程 (Linux)”中所述的步骤外，没有任何升级后的任务。不过，如果希望使用发行版本 4 的增强功能以取代发行版本 2 的相应功能，需要重新配置 Instant Messaging，并重新部署到 Web 容器中。

升级回滚

通过保存所有发行版本 2 软件和数据（参见第 281 页中的“[备份 Instant Messaging 数据](#)”）并手动恢复到发行版本 2 的版本，可实现从发行版本 4 到发行版本 2 的升级回滚。没有用于回滚此升级的实用程序。

多实例升级

多实例升级对发行版本 4 的 Instant Messaging 不适用。

Portal Server

本章介绍如何将 Portal Server 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Portal Server 6.3.1 2005Q4。其中概括介绍了 Java ES 发行版本 4 所支持的不同升级途径的升级问题和过程。本章内容涵盖了 Solaris 和 Linux 操作系统上的升级：

- [第 288 页中的“Portal Server 升级概述”](#)
- [第 291 页中的“从 Java ES 发行版本 3 升级 Portal Server”](#)
- [第 300 页中的“从 Java ES 发行版本 2 升级 Portal Server”](#)

注

本章中的文件位置是相对于两个目录路径指定的，分别称为 *PortalServer-base* 和 *PortalServerConfig-base*。在最初安装 Portal Server 时，可能至少已将这两个路径的一部分指定为安装目录，否则，Java ES 安装程序会分配一个默认值。

PortalServer-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWps
- Linux: /opt/sun/portal

PortalServerConfig-base 的默认值视操作系统平台而定：

- Solaris: /etc/opt/SUNWps
 - Linux: /etc/opt/sun/portal
-

Portal Server 升级概述

本节将在以下方面对 Portal Server 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Portal Server](#)
- [Portal Server 升级指导说明](#)
- [Portal Server 数据](#)
- [兼容性问题](#)
- [Portal Server 组件依赖性](#)

关于 Java ES 发行版本 4 Portal Server

Java ES 发行版本 4 Portal Server 在功能上与发行版本 3 相同，但包含了自发行版本 3 以来所做的错误修复。

Portal Server 升级指导说明

[表 17-1](#) 显示了所支持的升级至 Java ES 发行版本 4 的 Portal Server 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 17-1 升级到 Java ES 发行版本 4: Sun Java System Portal Server 6.3.1 2005Q4 的途径

Java ES 发行版本	Portal Server 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Portal Server 6.3.1 2005Q1	直接升级： 通过应用修补程序执行。对此有一些限制（参见过程说明）。	无
发行版本 2	Sun Java System Portal Server 6.3 2004Q2	直接升级： 执行方式是通过应用修补程序升级到发行版本 4，重新配置软件，然后重新部署到 Web 容器。	配置数据

表 17-1 升级到 Java ES 发行版本 4: Sun Java System Portal Server 6.3.1 2005Q4 的途径 (续)

Java ES 发行版本	Portal Server 版本	常规方法	必需的重新配置
发行版本 1	Sun ONE Portal Server 6.2 (2003Q4)	不直接升级； 但可通过以下方式执行：首先升级到发行版本 3，然后通过应用修补程序升级到发行版本 4。对此有一些限制（参见过程说明）。	配置数据
Java ES 以前的发行版本		不直接升级。	

Portal Server 数据

下表显示了可能受 Portal Server 软件升级影响的数据类型。

表 17-2 Portal Server 数据用途

数据类型	位置	用途
配置数据	<i>PortalServerConfig-base/</i>	Portal Server 的配置。
Web 容器配置	Web Server: server.policy 和 server.xml 文件，位于 <i>WebServer-base/https-hostname/config</i> Application Server (Java ES 发行版本 3 和 4) : server.policy 和 domain.xml 文件，位于 <i>AppServer8Config-base/domains/domainName/config</i> Application Server (Java ES 发行版本 2) : server.policy 和 server.xml 文件，位于 <i>AppServer7Config-base/domains/domainName/config</i>	Portal Server Web 容器实例的配置。
自定义数据	<i>PortalServerConfig-base/desktop</i>	自定义模块的 JAR 文件。 自定义的 Portal Server 桌面样例。

表 17-2 Portal Server 数据用途 (续)

数据类型	位置	用途
目录模式	Directory Server	Portal Server 依赖于服务配置（如门户桌面）和存储在目录中的用户配置文件数据。
服务配置		
用户数据		
动态应用程序数据	无	Portal Server 并不永久存储应用程序数据（如会话状态）。

兼容性问题

发行版本 4 Portal Server 的接口没有任何变化。Portal Server 组件（包括移动访问组件）与早期版本向后兼容。

Portal Server 组件依赖性

Portal Server 对其他 Java ES 组件的依赖性会影响升级和重新配置 Portal Server 软件的过程。例如，Portal Server 接口或功能上的变化可能要求 Portal Server 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Portal Server 对以下 Java ES 组件具有依赖性：

- **共享组件。** Portal Server 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Web 容器。** Portal Server 依赖于 Web 容器服务，这些服务可由 Java ES Web Server 或 Java ES Application Server（或来自 Weblogic 和 WebSphere 的第三方 Web 容器）提供。
- **Access Manager（或 Access Manager SDK）。** Portal Server 依赖 Access Manager 为最终用户提供验证和授权服务，包括单点登录。如果 Access Manager 运行在远程计算机上，则 Access Manager SDK 必须在本地可用。
- **Directory Server。** Portal Server 访问 Directory Server 中存储的用户数据。因此，Portal Server 升级可能需要目录模式的扩展。

从 Java ES 发行版本 3 升级 Portal Server

本节包含将 Portal Server 从 Java ES 2005Q1（发行版本 3）升级到 Java ES 2005Q4（发行版本 4）的有关信息。

注 对于 Portal Server 已部署于 Application Server Web 容器中且在升级到发行版本 4 之前已从发行版本 2 升级到发行版本 3 的特殊情况，本节不适用。当前不支持上述升级途径。

本节涵盖了以下主题：

- [简介](#)
- [发行版本 3 Portal Server 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Portal Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 执行升级的方法是先将修补程序应用于发行版本 3 的版本，然后将 Portal Server 重新部署到 Web 容器。
- **升级依赖性。** 虽然 Portal Server 对许多 Java ES 共享组件具有依赖性（参见 [第 38 页中的表 1-6](#)），但发行版本 4 Portal Server 与发行版本 3 的这些组件兼容。因此，对于 Portal Server 到发行版本 4 的升级，除 Mobile Access Core (MA Core) 之外，这些共享组件的升级是可选的。

此外，如 [第 290 页中的“Portal Server 组件依赖性”](#) 中所述，发行版本 4 Portal Server 依赖于 Web 容器、Access Manager 和 Directory Server。但是，这些是软性升级依赖性；对于 Portal Server 到发行版本 4 的升级，这些组件的升级是可选的。

- **向后兼容性。** 发行版本 4 的 Portal Server 与发行版本 3 的版本向后兼容。
- **升级回滚。** Portal Server 的发行版本 4 到发行版本 3 的升级回滚可通过回滚升级期间应用的修补程序来实现。修补程序回滚在 Linux 平台上不可用。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Portal Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Portal Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Portal Server 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Portal Server \(Solaris\)](#)
- [升级发行版本 3 Portal Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Portal Server 之前，应执行下述任务。

检验当前版本信息

可使用以下命令检验 Portal Server 的当前版本：

```
PortalServer-base/bin/version
```

表 17-3 Portal Server 版本检验输出

Java ES 发行版本	Portal Server 版本号
发行版本 2	6.3
发行版本 3	6.3.1
发行版本 4	6.3.1 ¹

1. 发行版本 3 与发行版本 4 的不同之处在于修补程序。可使用 Solaris `showrev -p | grep patch_ID` 命令和 Linux `rpm -qa sun-portal-core` 命令来检查第 296 页中的表 17-5 和第 310 页中的表 17-7 中所示的发行版本 4 修补程序，然后查找字符串 "25.12" 或更大者。

升级 Portal Server 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件均升级到 Java ES 发行版本 4。但是，Portal Server 仅对 Mobile Access Core (MA Core) 共享组件具有硬性升级依赖性。因此，Portal Server 所依赖的其他 Java ES 发行版本 3 组件的升级是可选的。

但是，如果选择升级 Portal Server 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Portal Server 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级到发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 或 Application Server 的说明。

注 升级诸如来自 WebLogic 和 WebSphere 的第三方 Web 容器可能导致 Portal Server 中断，因为在容器升级时会改写了为了支持 Portal Server 而对这些容器所做的自定义设置。

在此情况下，必须为升级后的 Web 容器环境重新安装和重新配置 Portal Server。

4. **Access Manager (Access Manager SDK)。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。

备份发行版本 3 Portal Server 配置信息

将 Portal Server 升级至发行版本 4 时，无需重新配置 Portal Server 软件。但是，为安全起见，可以备份存储配置信息的下列目录：

PortalServerConfig-base/

获取所需的配置信息和密码

必须以超级用户身份登录才能执行升级。如果要将 Web Server 用作 Web 容器，则无需任何配置信息。但如果要将 Application Server 用作 Web 容器，则将需要 Application Server 管理员用户 ID 和密码。

升级发行版本 3 Portal Server (Solaris)

本节讨论影响 Portal Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Portal Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 与同一 Portal Server 安装映像对应的所有 Portal Server 实例将同时进行升级。向安装映像应用修补程序时，应通过关闭 Web 容器关闭所有此类实例。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Portal Server 升级修补程序：

表 17-4 用于在 Solaris 上升级 Portal Server 的修补程序¹

说明	SPARC Solaris 8、9 和 10	X86 Solaris 9 和 10
Portal Server 核心组件	118950-12	118951-12
Portal Server 本地化版本	119425-08	119425-08
Portal Server 本地化配置程序	118115-11	118115-11

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Portal Server。

1. 根据表 17-4，获取所需的修补程序。

若未指明使用某个特定修订版本，始终应使用可用的最新修补程序修订版本。

可从以下网址将修补程序下载到 /tmp：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 通过停止其 Web 容器，停止 Portal Server。

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 升级 MA Core 共享组件以及您希望升级的任何其他共享组件（如果尚未执行此操作）。

参见第 293 页中的“升级 Portal Server 依赖项”。

5. 应用表 17-4 中相应的 Portal Server 修补程序。

确保先应用 Portal Server 核心修补程序，然后再应用两个 Portal Server 本地化修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 通过重新启动其 Web 容器，重新启动 Portal Server。

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

8. 将 Portal Server Web 应用程序重新部署到 Web 容器。

```
PortalServer-base/bin/deploy redeploy
```

redeploy 命令会将内容从 *PortalServer-base/web-src* 重新部署至 */var/PortalServer-base/https-hostName/deploy-dir/web-apps*。因此，应首先对 */web-src* 进行 Portal Server Web 应用程序的任何自定义设置，然后再将其部署至 */web-apps*。运行 *deploy* 命令之前，应在 */web-src* 中复制可能在 */web-apps* 下所做的任何更改，否则，此类更改将会被改写。

9. 停止并重新启动 Web 容器。

虽然并非在所有情况下都需要重新启动 Web 容器，但这样做可确保 Portal Server 在干净的状态下启动。

升级发行版本 3 Portal Server (Linux)

本节讨论影响 Portal Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Portal Server 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 294 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 4 的 Portal Server 升级修补程序：

表 17-5 用于在 Linux 上升级 Portal Server 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Portal Server 核心组件	118952-12 <ul style="list-style-type: none"> sun-portal-core-6.3-25.12.i386.rpm 以及用于 Portal 桌面和 Portal Server 移动访问的许多其他 RPM。
Portal Server 本地化版本	119426-07 <ul style="list-style-type: none"> sun-portal-core-Locale-6.3-24.i386.rpm 以及用于 Portal Server 移动访问、配置、标识和其他组件的许多其他 RPM。
Portal Server 本地化配置程序	118116-08 <ul style="list-style-type: none"> sun-portal-l10n-configurator-6.3-24.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Portal Server。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 17-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 通过停止其 Web 容器，停止 Portal Server。

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 升级 MA Core 共享组件以及您希望升级的任何其他共享组件（如果尚未执行此操作）。

参见第 293 页中的“升级 Portal Server 依赖项”。

5. 应用表 17-5 中用于 Portal Server 核心修补程序的 RPM。

```
cd /tmp
```

其中，/tmp 是步骤 1 中修补程序的下载目录。

```
./update
```

更新脚本会安装 RPM 并确保因应用了修补程序而产生正确的配置变化。

6. 确认修补程序升级成功：

```
rpm -qa | grep sun-portal-core-6.3-25
```

应返回 RPM 的升级修订版本号。

7. 应用表 17-5 中用于两个 Portal Server 本地化修补程序的 RPM。

```
rpm -Fvh --replacefiles sun-portal-*-Locale-6.3-24.i386.rpm
```

```
rpm -Fvh --replacefiles
```

```
sun-portal-l10n-configurator-6.3-24.i386.rpm
```

8. 确认修补程序升级成功：

```
rpm -qa | grep sun-portal-l10n-configurator-6.3-24
```

应返回 RPM 的升级修订版本号。

9. 通过重新启动其 Web 容器，重新启动 Portal Server。

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
```

```
--password password domainName
```

10. 将 Portal Server Web 应用程序重新部署到 Web 容器。

```
PortalServer-base/bin/deploy redeploy
```

`redeploy` 命令会将内容从 `PortalServer-base/web-src` 重新部署至 `/var/PortalServer-base/https-hostName/deploy-dir/web-apps`。因此，应首先对 `/web-src` 进行 Portal Server Web 应用程序的任何自定义设置，然后再将其部署至 `/web-apps`。运行 `deploy` 命令之前，应在 `/web-src` 中复制可能在 `/web-apps` 下所做的任何更改，否则，此类更改将会被改写。

11. 停止并重新启动 Web 容器。

虽然并非在所有情况下都需要重新启动 Web 容器，但这样做可确保 Portal Server 在干净的状态下启动。

检验升级

通过确认是否已正确应用升级修补程序来检验 Portal Server 到发行版本 4 的升级。第 294 页中的“升级过程 (Solaris)”和第 296 页中的“升级过程 (Linux)”中包括此检验的步骤。

此外，还可使用以下命令：

```
PortalServer-base/bin/version
```

参见第 292 页中的表 17-3 确认输出值。

除了对修补程序升级进行这些测试外，还可以检验先前正常运行的程序是否仍然正常，以及是否确实修复了所关注的错误。

升级后的任务

除第 294 页中的“升级过程 (Solaris)”和第 296 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Portal Server 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Portal Server 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。该过程将回滚重新配置并删除修补程序。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 通过停止其 Web 容器，停止 Portal Server。

Web Server:

```
WebServer-base/https-instanceName/stop
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

3. 删除第 294 页中的表 17-4 中的修补程序。

```
patchrm patch_ID
```

4. 通过重新启动其 Web 容器，重新启动 Portal Server。

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

5. 将 Portal Server Web 应用程序重新部署到 Web 容器。

```
PortalServer-base/bin/deploy redeploy
```

redeploy 命令会将内容从 *PortalServer-base/web-src* 重新部署至 */var/PortalServer-base/https-hostName/deploy-dir/web-apps*。因此，应首先对 */web-src* 进行 Portal Server Web 应用程序的任何自定义设置，然后再将其部署至 */web-apps*。运行 *deploy* 命令之前，应在 */web-src* 中复制可能在 */web-apps* 下所做的任何更改，否则，此类更改将会被改写。

6. 停止并重新启动 Web 容器。

虽然并非在所有情况下都需要重新启动 Web 容器，但这样做可确保 Portal Server 在干净的状态下启动。

多实例升级

在一些部署体系结构中，Portal Server 被部署在多个计算机系统上，以提供可伸缩性和提高可用性。例如，您可以在多台计算机上运行 Portal Server 组件，并使用一个负载均衡器来分配负载。

就 Portal Server 负载均衡的实例而言，可执行滚动升级，期间您可依序升级各个 Portal Server 实例而不中断服务。在升级每个 Portal Server 实例时，其他实例仍可保持运行。可按第 292 页中的“[发行版本 3 Portal Server 的升级](#)”中所述升级每个实例。

从 Java ES 发行版本 2 升级 Portal Server

本节包含将 Portal Server 从 Java ES 2004Q2（发行版本 2）升级到 Java ES 2005Q4（发行版本 4）的有关信息。

鉴于发行版本 2 Portal Server 升级的复杂性和可能出现的相当长的停机时间，您可能会选择在另一台计算机上执行平行升级，而不是在生产系统上进行就地升级。此种方法对于只容许有限停机时间的业务攸关或复杂的 Portal Server 解决方案较为可取。升级过程的持续时间还取决于重新实现和测试任何必要的 Portal Server 自定义内容所需的时间。

另外，为适应特定的升级方案，可能有必要对本节中的说明进行修改或调整。在此情况下，建议在执行升级时向 Sun Microsystems 支持服务寻求帮助。

本节包含有关从发行版本 2 升级到发行版本 4 的以下主题：

- [简介](#)
- [发行版本 2 Portal Server 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 2 Portal Server 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 2 执行升级。还必须使用升级实用程序重新配置 Portal Server。
- **升级依赖性。** Portal Server 依赖于许多 Java ES 共享组件（参见第 38 页中的表 1-6），并且需要将所有这些组件全都升级到发行版本 4，因为 Java ES 不支持在单台计算机上混合使用发行版本 2 和发行版本 4 组件。

此外，如第 290 页中的“Portal Server 组件依赖性”中所述，发行版本 4 的 Portal Server 依赖于 Web 容器、Access Manager 和 Directory Server。Portal Server 对 Web 容器和 Access Manager（或 Access Manager SDK）具有硬性升级依赖性（因为它们驻留在本地），而对 Directory Server 只具有软性升级依赖性（因为它很少驻留在本地）。

- **向后兼容性。** 发行版本 4 的 Portal Server 与发行版本 2 的版本向后兼容。
- **升级回滚。** 只要重新配置过 Portal Server，就无法实现 Portal Server 的发行版本 4 到发行版本 2 的升级回滚。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Portal Server 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 2 Portal Server 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Portal Server 从 Java ES 发行版本 2 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 2 Portal Server \(Solaris\)](#)
- [升级发行版本 2 Portal Server \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 Portal Server 之前，应执行下述任务。

检验当前版本信息

可使用以下命令检验 Portal Server 的当前版本：

```
PortalServer-base/bin/version
```

参见第 292 页中的表 17-3 确认输出值。

升级 Portal Server 依赖项

Java ES 发行版本 4 不支持发行版本 2 和发行版本 4 的共享组件在一台计算机上共存。

因此，必须将 Portal Server 所依赖的所有本地 Java ES 发行版本 2 组件全部升级到发行版本 4。在升级计算机上 Portal Server 所依赖的所有本地组件时，应按以下顺序升级，并且全部应在升级 Portal Server 之前进行。请注意，对于特定升级方案有特殊的要求。

1. **共享组件。** 第 47 页中的“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级到发行版本 4 的说明。
2. **Directory Server。** Portal Server 很少依赖于本地 Directory Server。但是在第 91 页中的第 4 章“Directory Server 和 Administration Server”中依然提供了将 Directory Server 升级到发行版本 4 的说明。
3. **Web 容器软件。** Portal Server 可在 Web Server 或 Application Server 所提供的 Web 容器中运行。
 - Web Server: 按照第 132 页中的“从 Java ES 发行版本 2 升级 Web Server”中的过程升级到发行版本 4 Web Server。
 - Application Server: 不按照第 169 页中的“从 Java ES 发行版本 2 升级 Application Server”中的过程，而是通过使用 Java ES 安装程序执行 Application Server 的全新安装，将其升级到发行版本 4 Application Server。安装发行版本 4 Application Server 8 之前，请务必从发行版本 2 Application Server 7 中获取管理端口和服务器实例端口。

注 升级诸如来自 WebLogic 和 WebSphere 的第三方 Web 容器可能导致 Portal Server 中断，因为在容器升级时会改写了为了支持 Portal Server 而对这些容器所做的自定义设置。

在此情况下，必须为升级后的 Web 容器环境重新安装和重新配置 Portal Server。

4. **Access Manager (Access Manager SDK)。** Portal Server 可以与 Access Manager 在相同或不同的 Web 容器中运行。
 - 如果 Portal Server 与 Access Manager 在不同的 Web 容器中运行（例如，如果 Access Manager 在远程运行），请按照第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”中的过程将 Access Manager 或 Access Manager SDK 从发行版本 2 升级到发行版本 4。如果只升级 Access Manager SDK，请参阅第 202 页中的“升级发行版本 3 Access Manager SDK”并设置 `DEPLOY_LEVEL = 3`。

- 如果 Portal Server 与 Access Manager 在相同的 Web 容器中运行，且该 Web 容器是由 Web Server 提供的，则按照第 204 页中的“升级发行版本 2 Access Manager: Web Server Web 容器”中的过程将 Access Manager 从发行版本 2 升级到发行版本 4。
- 如果 Portal Server 与 Access Manager 在相同的 Web 容器中运行，且该 Web 容器是由 Application Server 提供的，则按照第 204 页中的“升级发行版本 2 Access Manager: Application Server Web 容器”中的过程将 Access Manager 从发行版本 2 升级到发行版本 4，但务必要使用全新安装 AS 时所采用的方案。

备份发行版本 2 Portal Server 配置信息

将 Portal Server 升级到发行版本 4 时，必须重新配置 Portal Server 软件。为安全起见，可以备份存储配置信息的下列目录：

PortalServerConfig-base/

获取所需的配置信息和密码

必须以超级用户身份登录才能执行升级。如果使用 Web Server 作为 Web 容器，无需任何管理员密码。但如果使用 Application Server 作为 Web 容器，则需要 Application Server 管理员用户 ID 和密码。

升级发行版本 2 Portal Server (Solaris)

本节讨论影响 Portal Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Portal Server 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 与同一 Portal Server 安装映像对应的所有 Portal Server 实例将同时进行升级。向安装映像应用修补程序时，应通过关闭 Web 容器关闭所有此类实例。
- 下表显示了用于 Solaris 操作系统的发行版本 4 的 Portal Server 升级修补程序：

表 17-6 在 Solaris 上将 Portal Server 升级到发行版本 4 的修补程序

说明	SPARC	X86	Solaris 10
	Solaris 8 和 9	Solaris 9	
Portal Server sync-up	118195-07	118196-07	不适用
Portal Server 核心组件	118128-13	118129-13	不适用
Mobile Access 核心组件	119527-02	119528-02	不适用
Portal Server 修复	118950-15 (或更高版本)	118951-15 (或更高版本)	不适用

- 在 Solaris 平台上升级 Portal Server 的过程取决于部署 Portal Server 的 Web 容器是由 Web Server 还是由 Application Server 提供。因此，下面针对这两个 Web 容器分别提供了升级说明。

升级过程（Solaris: Web Server）

下述过程适用于要进行升级的计算机中的 Portal Server。

1. 根据表 17-6，获取所需的修补程序。

确保下载恰如表 17-6 中所示的修补程序修订版本，但 Portal Server 修复除外，对于这些修复可能有较新的修补程序可用。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 通过停止其 Web 容器，停止 Portal Server。

```
WebServer-base/https-instanceName/stop
```

4. 升级所有共享组件、Web 容器和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 302 页中的“升级 Portal Server 依赖项”。

5. 启动 Directory Server 和 Access Manager（如果它们尚未运行）。

6. 应用表 17-6 中相应的 Portal Server 修补程序。

确保按照表 17-6 中所示从上到下的顺序应用修补程序。

```
patchadd patch_ID
```

7. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 7 中应用的修补程序 ID 的版本。

8. 重新配置 Portal Server 软件：

```
ksh
```

```
$ cd PortalServer-base/lib
```

```
$ ./upgradePS04Q205Q1
```

9. 通过重新启动其 Web 容器，重新启动 Portal Server。

```
WebServer-base/https-instanceName/start
```

10. 将 Portal Server Web 应用程序重新部署到 Web 容器。

PortalServer-base/bin/deploy redeploy

redeploy 命令会将内容从 *PortalServer-base/web-src* 重新部署至 */var/PortalServer-base/https-hostName/deploy-dir/web-apps*。因此，应首先对 */web-src* 进行 Portal Server Web 应用程序的任何自定义设置，然后再将其部署至 */web-apps*。运行 *deploy* 命令之前，应在 */web-src* 中复制可能在 */web-apps* 下所做的任何更改，否则，此类更改将会被改写。

11. 停止并重新启动 Web 容器。

虽然并非在所有情况下都需要重新启动 Web 容器，但这样做可确保 Portal Server 在干净的状态下启动。

升级过程（Solaris: Application Server）

下述过程适用于要进行升级的计算机中的 Portal Server。

1. 根据表 17-6，获取所需的修补程序。

确保下载恰如表 17-6 中所示的修补程序修订版本，但 Portal Server 修复除外，对于这些修复可能有较新的修补程序可用。

可从以下网址将修补程序下载到 */tmp*：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

su -

3. 确保 Portal Server 不再在其发行版本 2 的 Application Server 实例中运行。

AppServerConfig7-base/domains/domainName/instanceName/bin/stopserv

在上述命令及后续步骤中采用了以下约定：

- 默认的 *domainName* 为 *domain1*
- 默认的 *instanceName* 为 *server1*

4. 升级所有共享组件、Web 容器和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 302 页中的“升级 Portal Server 依赖项”。

5. 确保已升级的 Access Manager 未在其发行版本 4 的 Application Server 实例中运行。

AppServer8-base/bin/asadmin stop-domain domainName

6. 确保 Access Manager 配置文件

AccessManagerConfig-base/config/AMConfig.properties

包含以下属性值:

```
com.iplanet.am.notification.url=  
    http://hostName:port/amserver/notificationsservice  
com.sun.identity.webcontainer=IAS8.1  
com.iplanet.am.cookie.encode=true
```

其中, *hostName:port* 为托管 Access Manager 实例的计算机和端口。

7. 应用表 17-6 中相应的 Portal Server 修补程序。

确保按照表 17-6 中所示从上到下的顺序应用修补程序。

```
patchadd patch_ID
```

8. 确认修补程序升级成功:

```
showrev -p | grep patch_ID
```

输出应返回在步骤 7 中应用的修补程序 ID 的版本。

9. 确保 Portal Server 配置文件

PortalServerConfig-base/PSConfig.properties

包含以下属性值, 这些属性值引用了 Application Server 的 Domain Administration Server (DAS) 实例:

```
DEPLOY_TYPE=SUNONE8  
DEPLOY_INSTANCE_DIR=AppServer8Config-base/domains/domainName  
DEPLOY_DOMAIN=AppServer8Config-base/domains/domainName  
DEPLOY_PRODUCT_DIR=AppServer8Config-base/domains/domainName  
DEPLOY_ADMIN_PROTOCOL=https  
DEPLOY_ADMIN_PORT=DAS_adminPort (例如, 默认值 = 4848)  
DEPLOY_ADMIN_HOST=DAS_hostName  
LOAD_BALANCER_URL=http://DAS_hostName:DAS_hostPort/portal  
DEPLOY_DOCROOT=AppServer8Config-base/domains/domainName/docroot  
PS_PORT=DAS_hostPort (例如, 默认值 = 80)  
DEPLOY_DIR=AppServer8-base  
PS_PROTOCOL=http
```

假定分配给全新安装的发行版本 4 Application Server 8 的端口值与分配给所安装的发行版本 2 Application Server 7 的端口值相同, 并且这些端口值是默认端口值, 则上面所示的默认值将适用。

10. 按如下所示修改 `PSconfig.properties` 文件:

```
DEPLOY_INSTANCE=temporary_instanceName
```

其中, *temporary_instanceName* 是一个未使用的临时值。

11. 确保 DAS 正在运行。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

12. 执行以下命令:

```
cd PortalServer-base/bin  
./multiserverinstance
```

`multiserverinstance` 脚本询问的许多问题使用第 306 页中的步骤 9 中所示文件 `PSConfig.properties` 中设置的值作为默认值, 以下说明假设这些默认值是正确的。

按如下所示回答 `multiserverinstance` 脚本所问的问题:

1. 为 Create a new portalserver instance (创建新的 portalserver 实例) 选择选项 1。
2. 为 Sun Java System Application Server 8.1 选择选项 3。
3. Where is the Web Container installed? (Web 容器安装在何处?) 按回车键。
4. What is the domain name? (域名是什么?) 按回车键。
5. What is the domain (DAS) path? (域 (DAS) 路径是什么?) 输入第 4 个问题中显示的那个默认值。
6. What is the Web Container instance path? (Web 容器实例的路径是什么?) 输入第 5 个问题中所输入的那个值。
7. What is the Web Container administrator? (Web 容器管理员是什么?) 按回车键。
8. What is the Web Container administration port? (Web 容器管理端口是什么?) 按回车键。
9. Is the Web Container administration port secure? (Web 容器管理端口是否安全?) 按回车键。
10. Instance name? (实例名?) 输入值 `server`。
11. Instance port? (实例端口?) 输入在 `PSConfig.properties` 文件中为 `PS_PORT` 值输入的那个值。
12. Is the instance port secure? (实例端口是否安全?) 按回车键。
13. What is the Web Container document root directory? (Web 容器文档根目录是什么?) 按回车键。
14. What is the Application Server administration password? (Application Server 管理密码是什么?) 输入密码。
15. What is the Identity Server administration password? (Identity Server 管理密码是什么?) 输入密码。

- 按如下所示修改 `PSconfig.properties` 文件:

```
DEPLOY_INSTANCE=server
```

其中, 值 `server` 为 DAS 实例的默认实例名。

- 重新启动 DAS。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

- 部署 Portal Server Web 应用程序。

```
cd PortalServer-base/bin
```

```
./deploy redeploy
```

忽略指示 `deploy.log` 中可能有错误的消息。

`redeploy` 命令会将内容从 `PortalServer-base/web-src` 重新部署至 `/var/PortalServer-base/https-hostName/deploy-dir/web-apps`。因此, 应首先对 `/web-src` 进行 Portal Server Web 应用程序的任何自定义设置, 然后再将其部署至 `/web-apps`。运行 `deploy` 命令之前, 应在 `/web-src` 中复制可能在 `/web-apps` 下所做的任何更改, 否则, 此类更改将会被改写。

- 重新配置 Portal Server 软件:

```
ksh
```

```
$ cd PortalServer-base/lib
```

```
$ ./postinstall_PortletSamples
```

```
$ ./upgradePS04Q205Q1
```

忽略 `upgradePS04Q205Q1` 脚本所发出的与 CLI137 相关的错误以及与 (取消) 部署相关的错误。

- 重新启动 DAS。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

虽然并非在所有情况下都需要重新启动 Web 容器, 但这样做可确保 Portal Server 在干净的状态下启动。

18. 更新 Portal Server 显示配置文件。

- a. 执行以下命令：

```
PortalServer-base/bin/dpadmin list -g -u amadminDN
-w amadminPassword /tmp/GlobalDP.xml
```

其中，*amadminDN* 的值可在 *AccessManagerConfig-base/config/AMConfig.properties* 文件的属性 *com.sun.identity.authentication.super.user* 中找到。

- b. 打开文件 */tmp/GlobalDP.xml* 进行编辑
c. 将下面的值：

```
org.apache.xalan.xsltc.trax.TransformerFactoryImpl
修改为
com.sun.org.apache.xalan.internal.xsltc.trax.
TransformerFactoryImpl
```

- d. 将出现的所有以下值：

```
Sun Java™ System Portal Server 6 2004Q2
修改为
Sun Java™ System Portal Server 6 2005Q4
```

- e. 执行以下命令：

```
PortalServer-base/bin/dpadmin modify -g -u amadminDN
-w amadminPassword /tmp/GlobalDP.xml
```

其中，*amadminDN* 的值与步骤 a 中的值相同。

升级发行版本 2 Portal Server (Linux)

本节讨论影响 Portal Server 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Portal Server 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 294 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 4 的 Portal Server 升级修补程序：

表 17-7 在 Linux 上将 Portal Server 升级到发行版本 4 的修补程序

说明	修补程序 ID 和 RPM 名称
Portal Server 核心组件	118020-16 sun-portal-module-6.3-25.i386.rpm 其中， <i>module</i> 是约 70 个不同软件模块的任意一个
Mobile Access 核心组件	119529-02 <ul style="list-style-type: none"> • sun-mobileaccess-1.0-25.2.i386.rpm • sun-mobileaccess-config-1.0-25.2.i386.rpm
Portal Server 修复	118952-15（或更高版本） <ul style="list-style-type: none"> • sun-portal-core-6.3-xx.y.i386.rpm • sun-portal-configurator-6.3-xx.y.i386.rpm • sun-portal-mobileaccess-6.3-xx.y.i386.rpm • sun-portal-desktop-6.3-xx.y.i386.rpm • sun-portal-sample-6.3-xx.y.i386.rpm • sun-portal-mobileaccess-config-6.3-xx.y.i386.rpm

升级过程（Linux: Web Server）

下述过程适用于要进行升级的计算机中的 Portal Server。

1. 使用表 17-7 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 通过停止其 Web 容器，停止 Portal Server。

```
WebServer-base/https-instanceName/stop
```

4. 升级所有共享组件、Web 容器和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 302 页中的“升级 Portal Server 依赖项”。

5. 启动 Directory Server 和 Access Manager（如果它们尚未运行）。
6. 应用表 17-7 中用于 Portal Server 的 RPM。
 - a. `cd /tmp`
其中，`/tmp` 是步骤 1 中修补程序的下载目录。
 - b. 解压缩 118020 修补程序文件，阅读自述文件并运行以下脚本：
`./upgradeportalrpms`
`upgradeportalrpms` 脚本会安装 RPM，同时确保由修补程序所引起的配置变化是正确的。
 - c. 解压缩 119529 修补程序文件，然后从解压缩修补程序时所创建的目录内运行 `./update` 脚本。
 - d. 解压缩 118952 修补程序文件，然后从解压缩修补程序时所创建的目录内运行 `./update` 脚本。
7. 确认修补程序升级成功：
`rpm -qa | grep sun-portal`
`rpm -qa | grep sun-mobileaccess`
应返回 RPM 的新版本号。
8. 重新配置 Portal Server 软件：
`ksh`
`$ cd PortalServer-base/lib`
`$./upgradePS04Q205Q1`
9. 按如下所示编辑 `PortalServer-base/export/deploy.import` 文件：
如果存在以下内容：
`%JATO_LIB_DIR%/jato.tld %WEB_SRC_DIR%/WEB-INF/jato.tld`
`%JATO_LIB_DIR%/jato.jar %WEB_SRC_DIR%/WEB-INF/lib/jato.jar`
替换为：
`/usr/share/lib/jato/jato.tld %WEB_SRC_DIR%/WEB-INF/jato.tld`
`/usr/share/lib/jato/jato.jar %WEB_SRC_DIR%/WEB-INF/lib/jato.jar`
换言之，就是用 `/usr/share/lib/jato` 替换 `%JATO_LIB_DIR%`
10. 通过重新启动其 Web 容器，重新启动 Portal Server。
`WebServer-base/https-instanceName/start`

11. 将 Portal Server Web 应用程序重新部署到 Web 容器。

PortalServer-base/bin/deploy redeploy

redeploy 命令会将内容从 *PortalServer-base/web-src* 重新部署至 */var/PortalServer-base/https-hostName/deploy-dir/web-apps*。因此，应首先对 */web-src* 进行 Portal Server Web 应用程序的任何自定义设置，然后再将其部署至 */web-apps*。运行 *deploy* 命令之前，应在 */web-src* 中复制可能在 */web-apps* 下所做的任何更改，否则，此类更改将会被改写。

12. 停止并重新启动 Web 容器。

虽然并非在所有情况下都需要重新启动 Web 容器，但这样做可确保 Portal Server 在干净的状态下启动。

升级过程（Linux: Application Server）

下述过程适用于要进行升级的计算机中的 Portal Server。

1. 根据表 17-7，获取所需的修补程序。

确保下载恰如表 17-7 中所示的修补程序修订版本，但 Portal Server 修复除外，对于这些修复可能有较新的修补程序可用。

可从以下网址将修补程序下载到 */tmp*：

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

su -

3. 确保 Portal Server 不再在其发行版本 2 的 Application Server 实例中运行。

AppServerConfig7-base/domains/domainName/instanceName/bin/stopserv

在上述命令及后续步骤中采用了以下约定：

- 默认的 *domainName* 为 *domain1*
- 默认的 *instanceName* 为 *server1*

4. 升级所有共享组件、Web 容器和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 302 页中的“升级 Portal Server 依赖项”。

5. 确保已升级的 Access Manager 未在其发行版本 4 的 Application Server 实例中运行。

AppServer8-base/bin/asadmin stop-domain domainName

6. 确保 Access Manager 配置文件

AccessManagerConfig-base/config/AMConfig.properties

包含以下属性值：

```
com.ipplanet.am.notification.url=
    http://hostName:port/amserver/notificationservice
com.sun.identity.webcontainer=IAS8.1
com.ipplanet.am.cookie.encode=true
```

其中，*hostName:port* 为托管 Access Manager 实例的计算机和端口。

7. 应用表 17-7 中用于 Portal Server 的 RPM。

a. `cd /tmp`

其中，`/tmp` 是步骤 1 中修补程序的下载目录。

b. 解压缩 118020 修补程序文件，阅读自述文件并运行以下脚本：

```
./upgradeportalrpms
```

`upgradeportalrpms` 脚本会安装 RPM，同时确保由修补程序所引起的配置变化是正确的。

c. 解压缩 119529 修补程序文件，然后从解压缩修补程序时所创建的目录内运行 `./update` 脚本。

d. 解压缩 118952 修补程序文件，然后从解压缩修补程序时所创建的目录内运行 `./update` 脚本。

8. 确认修补程序升级成功：

```
rpm -qa | grep sun-portal
rpm -qa | grep sun-mobileaccess
```

应返回 RPM 的新版本号。

9. 按如下所示编辑 *PortalServer-base/export/deploy.import* 文件：

如果存在以下内容：

```
%JATO_LIB_DIR%/jato.tld %WEB_SRC_DIR%/WEB-INF/jato.tld
%JATO_LIB_DIR%/jato.jar %WEB_SRC_DIR%/WEB-INF/lib/jato.jar
```

替换为：

```
/usr/share/lib/jato/jato.tld %WEB_SRC_DIR%/WEB-INF/jato.tld
/usr/share/lib/jato/jato.jar %WEB_SRC_DIR%/WEB-INF/lib/jato.jar
```

换言之，就是用 `/usr/share/lib/jato` 替换 `%JATO_LIB_DIR%`

10. 按“升级过程（Solaris: Application Server）”下的第 306 页中的步骤 9 到第 309 页中的步骤 18 进行操作。

检验升级

通过确认是否已正确应用升级修补程序来检验 Portal Server 到发行版本 4 的升级。第 294 页中的“升级过程 (Solaris)”和第 296 页中的“升级过程 (Linux)”中包括此检验的步骤。

此外，还可使用以下命令：

```
PortalServer-base/bin/version
```

参见第 292 页中的表 17-3 确认输出值。

除了对修补程序升级进行这些测试外，还可以检验先前正常运行的程序是否仍然正常，以及是否确实修复了所关注的错误。

升级后的任务

除第 305 页中的“升级过程 (Solaris: Application Server)”和第 310 页中的“升级过程 (Linux: Web Server)”中所述的步骤之外，没有任何升级后的任务。

升级回滚

Portal Server 从发行版本 2 到发行版本 4 的升级无法回滚。

多实例升级

在一些部署体系结构中，Portal Server 被部署在多个计算机系统上，以提供可伸缩性和提高可用性。例如，您可以在多台计算机上运行 Portal Server 组件，并使用一个负载均衡器来分配负载。

就 Portal Server 负载均衡的实例而言，可执行滚动升级，期间您可依序升级各个 Portal Server 实例而不中断服务。在升级每个 Portal Server 实例时，其他实例仍可保持运行。可按第 301 页中的“发行版本 2 Portal Server 的升级”中所述升级每个实例。

Portal Server Secure Remote Access

本章介绍如何将 Portal Server Secure Remote Access 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Portal Server Secure Remote Access 6 2005Q4。

本章概括介绍了有关 Java ES 发行版本 4 所支持的不同升级途径的升级问题及过程。其内容涵盖了在 Solaris 和 Linux 操作系统上的升级：

- 第 316 页中的“Portal Server Secure Remote Access 升级概述”
- 第 318 页中的“从 Java ES 发行版本 3 升级 Portal Server Secure Remote Access”
- 第 327 页中的“从 Java ES 发行版本 2 升级 Portal Server Secure Remote Access”

注

本章中的文件位置是相对于两个目录路径指定的，分别称为 *PortalServer-base* 和 *PortalServerConfig-base*。在最初安装 Portal Server 时，可能至少已将这两个路径的一部分指定为安装目录，否则，Java ES 安装程序会分配一个默认值。

PortalServer-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWps
- Linux: /opt/sun/portal

PortalServerConfig-base 的默认值视操作系统平台而定：

- Solaris: /etc/opt/SUNWps
 - Linux: /etc/opt/sun/portal
-

Portal Server Secure Remote Access 升级概述

本节将在以下方面对 Portal Server Secure Remote Access 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Portal Server Secure Remote Access](#)
- [Portal Server Secure Remote Access 升级指导说明](#)
- [Portal Server Secure Remote Access 数据](#)
- [兼容性问题](#)
- [Portal Server Secure Remote Access 组件依赖性](#)

关于 Java ES 发行版本 4 Portal Server Secure Remote Access

Java ES 发行版本 4 Portal Server Secure Remote Access 主要修复了一些错误，相对于发行版本 3 并未新增任何主要的功能。

Portal Server Secure Remote Access 升级指导说明

表 18-1 显示了所支持的升级至 Java ES 发行版本 4 的 Portal Server Secure Remote Access 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 18-1 升级到 Java ES 发行版本 4：
Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4 的途径

Java ES 发行版本	Portal Server Secure Remote Access 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q1	直接升级： 通过应用修补程序执行。	无
发行版本 2	Sun Java System Portal Server Secure Remote Access 6.3 2004Q2	不直接升级： 通过先升级至发行版本 3 后应用修补程序升级至发行版本 4 来执行。	配置数据

表 18-1 升级到 Java ES 发行版本 4:
Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4 的途径 (续)

Java ES 发行版本	Portal Server Secure Remote Access 版本	常规方法	必需的重新配置
发行版本 1	Sun ONE Portal Server Secure Remote Access 6.2 (2003Q4)	不直接升级: 通过先升级至发行版本 3 后应用修补程序升级至发行版本 4 来执行。	配置数据
Java ES 以前的发行版本		不直接升级。	

Portal Server Secure Remote Access 数据

下表显示了可能受 Portal Server Secure Remote Access 软件升级影响的数据类型。

表 18-2 Portal Server Secure Remote Access 数据用途

数据类型	位置	用途
配置数据	<i>PortalServerConfig-base/</i>	Portal Server Secure Remote Access 的配置。
目录模式 用户数据	Directory Server	Portal Server Secure Remote Access 取决于目录中存储的用户配置文件数据。
动态应用程序数据	无	Portal Server Secure Remote Access 并不永久存储应用程序数据 (如会话状态)。

兼容性问题

发行版本 4 的 Portal Server Secure Remote Access 的接口没有任何变化。Portal Server Secure Remote Access 与早期版本向后兼容。

Portal Server Secure Remote Access 组件依赖性

Portal Server Secure Remote Access 对其他 Java ES 组件的依赖性会影响升级和重新配置 Portal Server Secure Remote Access 软件的过程。例如, Portal Server Secure Remote Access 接口或功能上的变化可能要求 Portal Server Secure Remote Access 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Portal Server Secure Remote Access 对以下 Java ES 组件具有依赖性：

- **共享组件。** Portal Server Secure Remote Access 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Portal Server。** Portal Server Secure Remote Access 提供对 Portal Server 的安全远程访问。
- **Access Manager（或 Access Manager SDK）。** Portal Server Secure Remote Access 依赖 Access Manager 为最终用户提供验证和授权服务，包括单点登录。如果 Access Manager 运行在远程计算机上，则 Access Manager SDK 必须在本地可用。
- **Directory Server。** Portal Server Secure Remote Access 访问 Directory Server 中存储的用户数据。因此，Portal Server Secure Remote Access 升级可能需要目录模式的扩展。

从 Java ES 发行版本 3 升级 Portal Server Secure Remote Access

本节包含关于将 Portal Server Secure Remote Access 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 发行版本 4 的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Portal Server Secure Remote Access 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 3 Portal Server Secure Remote Access 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 3 执行升级。不要求重新配置 Portal Server Secure Remote Access。
- **升级依赖性。** 虽然 Portal Server Secure Remote Access 对许多 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Portal Server Secure Remote Access 与发行版本 3 的这些组件兼容。因此，对于 Portal Server Secure Remote Access 到发行版本 4 的升级，除 Mobile Access Core (MA Core) 之外，这些共享组件的升级是可选的。

此外，发行版本 4 的 Portal Server Secure Remote Access 还依赖 Portal Server、Access Manager 和 Directory Server，如第 317 页中的“[Portal Server Secure Remote Access 组件依赖性](#)”中所述。Portal Server Secure Remote Access 到发行版本 4 的升级还需要升级 Portal Server。但是，它对 Access Manager 和 Directory Server 的依赖性为软性升级依赖性；因此，对于 Portal Server Secure Remote Access 到发行版本 4 的升级，这些组件的升级是可选的。

- **向后兼容性。** 发行版本 4 的 Portal Server Secure Remote Access 与发行版本 3 的版本向后兼容。
- **升级回滚。** Portal Server Secure Remote Access 的发行版本 4 到发行版本 3 的升级回滚可通过回滚在升级期间应用的修补程序实现。修补程序回滚在 Linux 平台上不可用。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Portal Server Secure Remote Access 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Portal Server Secure Remote Access 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Portal Server Secure Remote Access 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Portal Server Secure Remote Access \(Solaris\)](#)
- [升级发行版本 3 Portal Server Secure Remote Access \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Portal Server Secure Remote Access 之前，应执行下述任务。

检验当前版本信息

可使用以下命令检验 Portal Server Secure Remote Access 的当前版本：

```
PortalServer-base/bin/gateway version
PortalServer-base/bin/rwproxyd version
PortalServer-base/bin/netletd version
```

表 18-3 Portal Server Secure Remote Access 版本检验输出

Java ES 发行版本	Portal Server 版本号
发行版本 2	早于发行版本 3
发行版本 3	Thu Dec 16 03:30:34 PST 2004
发行版本 4	晚于发行版本 3 ¹

1. 发行版本 3 与发行版本 4 的不同之处仅在于修补程序。可使用 Solaris `showrev -p | grep patch_ID` 命令和 Linux `rpm -qa sun-portal-core` 命令来检查第 323 页中的表 18-5 和第 333 页中的表 18-7 中所示的发行版本 4 修补程序，然后查找字符串 "25.12" 或更大者。

升级 Portal Server Secure Remote Access 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件均升级到 Java ES 发行版本 4。但是，Portal Server 仅对 Mobile Access Core (MA Core) 共享组件具有硬性升级依赖性。因此，Portal Server 所依赖的其他 Java ES 发行版本 3 组件的升级是可选的。

但是，如果选择升级 Portal Server Secure Remote Access 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Portal Server Secure Remote Access 之前进行。您可以跳过任何可能已经升级的依赖组件。

- 1. 共享组件。** 第 47 页中的“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
- 2. Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
- 3. Access Manager (Access Manager SDK)。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
- 4. Portal Server。** 第 287 页中的第 17 章“Portal Server”中提供了升级 Portal Server 的说明。

备份发行版本 3 Portal Server Secure Remote Access 配置信息

将 Portal Server Secure Remote Access 升级至发行版本 4 时，无需重新配置 Portal Server Secure Remote Access 软件。但是，为安全起见，可以备份存储配置信息的下列目录：

PortalServerConfig-base/

获取所需的配置信息和密码

您必须以超级用户的身份登录以执行升级。

升级发行版本 3 Portal Server Secure Remote Access (Solaris)

本节讨论影响 Portal Server Secure Remote Access 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Portal Server Secure Remote Access 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 所有对应同一个已安装 Portal Server Secure Remote Access 映像的 Portal Server Secure Remote Access 实例将同时被升级。向安装的映像应用修补程序时，应关闭所有此类实例。
- 用于 Solaris 操作系统的发行版本 4 Portal Server Secure Remote Access 升级修补程序与用于升级 Portal Server 的修补程序相同，下表显示了这些修补程序：

表 18-4 在 Solaris 上升级 Portal Server Secure Remote Access 的修补程序¹

说明	SPARC	X86
	Solaris 8、9 和 10	Solaris 9 和 10
Portal Server 核心组件	118950-12	118951-12
Portal Server 本地化版本	119425-08	119425-08
Portal Server 本地化配置程序	118115-11	118115-11

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Portal Server Secure Remote Access。

1. 根据表 18-4，获取所需的修补程序。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway stop  
/etc/init.d/netletd stop  
/etc/init.d/rwproxyd stop
```

4. 升级 MA Core 共享组件以及您希望升级的任何其他共享组件（如果尚未执行此操作）。

参见第 320 页中的“升级 Portal Server Secure Remote Access 依赖项”。

5. 应用表 18-4 中相应的 Portal Server 修补程序。

确保先应用 Portal Server 核心修补程序，然后再应用两个 Portal Server 本地化修补程序。

```
patchadd patch_ID
```

6. 确认修补程序升级成功:

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

7. 重新启动 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway start  
/etc/init.d/netletd start  
/etc/init.d/rwproxyd start
```

升级发行版本 3 Portal Server Secure Remote Access (Linux)

本节讨论影响 Portal Server Secure Remote Access 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Portal Server Secure Remote Access 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 321 页中的“升级注意事项 (Solaris)”）。

用于 Linux 操作系统的发行版本 4 Portal Server Secure Remote Access 升级修补程序与用于升级 Portal Server 的修补程序相同，下表显示了这些修补程序：

表 18-5 在 Linux 上升级 Portal Server Secure Remote Access 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Portal Server 核心组件	118952-12 <ul style="list-style-type: none"> sun-portal-core-6.3-25.12.i386.rpm 以及用于 Portal 桌面和 Portal Server 移动访问的许多其他 RPM。
Portal Server 本地化版本	119426-07 <ul style="list-style-type: none"> sun-portal-core-Locale-6.3-24.i386.rpm 以及用于 Portal Server 移动访问、配置、标识和其他组件的许多其他 RPM。
Portal Server 本地化配置程序	118116-08 <ul style="list-style-type: none"> sun-portal-l10n-configurator-6.3-24.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Portal Server Secure Remote Access。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 18-4 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway stop  
/etc/init.d/netletd stop  
/etc/init.d/rwproxyd stop
```

4. 升级 MA Core 共享组件以及您希望升级的任何其他共享组件（如果尚未执行此操作）。

参见第 320 页中的“升级 Portal Server Secure Remote Access 依赖项”。

5. 应用表 18-5 中用于 Portal Server 核心修补程序的 RPM。

```
cd /tmp  
./update
```

更新脚本会安装 RPM 并确保因应用了修补程序而产生正确的配置变化。

6. 确认修补程序升级成功：

```
rpm -qa | grep sun-portal-gateway
```

应返回 RPM 的新版本号。

7. 应用表 18-5 中用于两个 Portal Server 本地化修补程序的 RPM。

```
rpm -Fvh --replacefiles sun-portal-*-Locale-6.3-24.i386.rpm  
rpm -Fvh --replacefiles  
    sun-portal-l10n-configurator-6.3-24.i386.rpm
```

8. 确认修补程序升级成功：

```
rpm -qa | grep sun-portal-l10n-configurator-6.3-24
```

应返回 RPM 的升级修订版本号。

9. 重新启动 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway start  
/etc/init.d/netletd start  
/etc/init.d/rwproxyd start
```

检验升级

通过确认是否已正确应用升级修补程序来检验 Portal Server Secure Remote Access 到发行版本 4 的升级。第 322 页中的“升级过程 (Solaris)”和第 323 页中的“升级过程 (Linux)”中包括此检验的步骤。

此外，还可使用下列命令：

```
PortalServer-base/bin/gateway version
PortalServer-base/bin/rwproxyd version
PortalServer-base/bin/netletd version
```

参见第 320 页中的表 18-3 确认输出值。

除了对修补程序升级进行这些测试外，还可以检验先前正常运行的程序是否仍然正常，以及是否确实修复了所关注的错误。

升级后的任务

除第 322 页中的“升级过程 (Solaris)”和第 323 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

本节介绍影响 Portal Server Secure Remote Access 升级回滚过程的一些注意事项，随后对升级回滚过程本身加以说明。

回滚注意事项 (Solaris)

升级至发行版本 4 的 Portal Server Secure Remote Access 的升级回滚过程几乎就是升级至发行版本 4 的相反过程。该过程将回滚重新配置并删除修补程序。

回滚过程 (Solaris)

1. 以 root 身份登录，或成为超级用户。

```
su -
```

2. 停止 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

3. 删除第 321 页中的表 18-4 中的修补程序。

```
patchrm patch_ID
```

4. 重新启动 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway start  
/etc/init.d/netletd start  
/etc/init.d/rwproxyd start
```

多实例升级

在一些部署体系结构中，Portal Server Secure Remote Access 被部署在多个计算机系统上，以提供可伸缩性和提高可用性。例如，您可以在多台计算机上运行 Portal Server Secure Remote Access 组件，并使用一个负载平衡器来分配负载。

就 Portal Server Secure Remote Access 负载平衡的实例而言，可执行滚动升级，期间您可依序升级各个 Portal Server Secure Remote Access 实例而不中断服务。在升级每个 Portal Server Secure Remote Access 实例时，其他实例仍可保持运行。可按第 319 页中的“[发行版本 3 Portal Server Secure Remote Access 的升级](#)”中所述升级每个实例。

从 Java ES 发行版本 2 升级 Portal Server Secure Remote Access

本节包含关于将 Portal Server Secure Remote Access 从 Java ES 发行版本 2 升级至 Java ES 发行版本 4 的信息。

该升级分两步执行：首先是将发行版本 2 升级到发行版本 3，然后从发行版本 3 升级到发行版本 4。由于这两条升级途径截然不同，本节着重讨论从发行版本 2 升级到发行版本 3。

完成从发行版本 2 到发行版本 3 的升级后，即可继续从发行版本 3 升级到发行版本 4，该过程已在第 318 页中的“[从 Java ES 发行版本 3 升级 Portal Server Secure Remote Access](#)”中介绍过。

本节包含有关从发行版本 2 升级到发行版本 3 的以下主题：

- [简介](#)
- [发行版本 2 Portal Server Secure Remote Access 的升级](#)
- [多实例升级](#)

简介

将 Java ES 发行版本 2 Portal Server Secure Remote Access 升级至发行版本 3 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过将修补程序应用到发行版本 2 执行升级。还必须使用升级实用程序重新配置 Portal Server Secure Remote Access。
- **升级依赖性。** Portal Server Secure Remote Access 对许多 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6），并且所有这些组件都必须升级到发行版本 3，因为 Java ES 不支持在一台计算机上同时有发行版本 2 和发行版本 3 的组件。

此外，发行版本 3 的 Portal Server Secure Remote Access 依赖 Portal Server、Access Manager 和 Directory Server，如第 317 页中的“[Portal Server Secure Remote Access 组件依赖性](#)”中所述。Portal Server Secure Remote Access 由于驻留在本地，故对 Portal Server 和 Access Manager（或 Access Manager SDK）具有硬性升级依赖性；而对 Directory Server 具有软性升级依赖性，因为后者很少驻留在本地。

- **向后兼容性。** 发行版本 3 的 Portal Server Secure Remote Access 与发行版本 2 的版本向后兼容。

- **升级回滚。** 只要重新配置过 Portal Server Secure Remote Access，就无法实现 Portal Server Secure Remote Access 的发行版本 3 到发行版本 2 的升级回滚。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Portal Server Secure Remote Access 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 2 Portal Server Secure Remote Access 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Portal Server Secure Remote Access 从 Java ES 发行版本 2 升级至 Java ES 发行版本 3。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 2 Portal Server Secure Remote Access \(Solaris\)](#)
- [升级发行版本 2 Portal Server Secure Remote Access \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚](#)

升级前的任务

升级 Portal Server Secure Remote Access 之前，应执行下述任务。

检验当前版本信息

可使用以下命令检验 Portal Server Secure Remote Access 的当前版本：

```
PortalServer-base/bin/gateway version  
PortalServer-base/bin/rwproxyd version  
PortalServer-base/bin/netletd version
```

参见第 320 页中的表 18-3 确认输出值。

升级 Portal Server Secure Remote Access 依赖项

Java ES 发行版本 3 不支持发行版本 2 和发行版本 3 的共享组件在一台计算机上共存。

因此，必须将 Portal Server Secure Remote Access 所依赖的所有本地 Java ES 发行版本 2 组件升级到发行版本 3。有关升级过程，参见 Java Enterprise System 2005Q1 升级与迁移指南 (<http://docs.sun.com/doc/819-2237>)。

升级某台计算机上所有本地 Portal Server 依赖项时，它们应按以下顺序升级，并且应在升级 Portal Server Secure Remote Access 之前进行。

1. **共享组件。** Java Enterprise System 2005Q1 升级与迁移指南 (<http://docs.sun.com/doc/819-2237>) 中提供了将 Java ES 共享组件升级到发行版本 3 的说明。
2. **Portal Server。** Portal Server Secure Remote Access 很少依赖本地的 Portal Server。
3. **Access Manager (Access Manager SDK)。** Java Enterprise System 2005Q1 升级与迁移指南 (<http://docs.sun.com/doc/819-2237>) 中提供了将 Access Manager 升级到发行版本 3 的说明。
4. **Directory Server。** Portal Server 很少依赖本地的 Directory Server。

备份发行版本 2 Portal Server Secure Remote Access 配置信息

将 Portal Server Secure Remote Access 升级到发行版本 3 时，必须重新配置 Portal Server Secure Remote Access 软件。为安全起见，可以备份存储配置信息的下列目录：

PortalServerConfig-base/

获取所需的配置信息和密码

您必须以超级用户的身份登录以执行升级。

升级发行版本 2 Portal Server Secure Remote Access (Solaris)

本节讨论影响 Portal Server Secure Remote Access 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Portal Server Secure Remote Access 软件升级至 Java ES 发行版本 3 时，应考虑以下注意事项：

- 所有对应同一个已安装 Portal Server Secure Remote Access 映像的 Portal Server Secure Remote Access 实例将同时被升级。向安装的映像应用修补程序之前，应关闭所有此类实例。

- 如果重写器代理、netlet 代理和网关进程在不同的计算机上运行，则需要先在托管重写器和 netlet 代理服务的计算机上执行升级过程，然后在所有托管网关服务的计算机上执行升级。
- 用于 Solaris 操作系统的发行版本 3 Portal Server Secure Remote Access 升级修补程序与用于升级 Portal Server 的修补程序相同，下表显示了这些修补程序：

表 18-6 在 Solaris 上将 Portal Server Secure Remote Access 升级到发行版本 3 的修补程序

说明	SPARC Solaris 8 和 9	X86 Solaris 9	Solaris 10
Portal Server sync-up	118195-07	118196-07	不适用
Portal Server 核心组件	118128-13	118129-13	不适用
Mobile Access 核心组件	118219-12	118219-12	不适用
Portal Server 修复	118950-15 (或更高版本)	118951-15 (或更高版本)	不适用

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Portal Server Secure Remote Access。

如果 Portal Server Secure Remote Access 与 Portal Server 安装在相同的计算机上，则按照第 291 页中的“从 Java ES 发行版本 3 升级 Portal Server”中的说明操作。然后转到第 331 页中的步骤 8。

1. 根据表 18-6，获取所需的修补程序。

确保下载了表 18-6 中所示的正确修补程序修订版本

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

4. 升级所有共享组件和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 329 页中的“升级 Portal Server Secure Remote Access 依赖项”。

5. 启动 Directory Server 和 Access Manager（或 Access Manager SDK）（如果它们尚未运行）。
6. 应用表 18-6 中相应的 Portal Server 修补程序。

确保按照表 18-6 中所示从上到下的顺序应用修补程序。

```
patchadd patch_ID
```

7. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在步骤 5 中应用的修补程序 ID 的版本。

8. 重新配置 Portal Server Secure Remote Access 软件：

```
ksh
```

```
$ cd PortalServer-base/lib
$ ./upgradeSRA-04Q4-05Q1
```

9. 重新启动 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway start
/etc/init.d/netletd start
/etc/init.d/rwproxyd start
```

10. 重新配置 Proxylet 和 Netlet 服务。

- a. 以管理员用户身份登录 Access Manager 控制台 (AMCONSOLE)。
- b. 移除 Proxylet 和 Netlet 服务。

在“身份管理”选项卡下，选择“服务”选项。这会在左面板上列出所有已注册的服务。在“SRA 配置”中，选中 Proxylet 和 Netlet 复选框。滚动到左面板顶部，单击“移除”按钮。这将从 ORG 级别移除 Proxylet 和 Netlet 服务。

要手动检验此步骤，可以检查 LDAP 目录（在您的组织下）以确定服务 (srapProxylet, srapNetlet) 已确实移除。

c. 再次添加服务。

在“身份管理”选项卡下，选择“服务”选项。单击“服务”下的“添加”按钮。这会在右面板中显示所有可用的服务。选中 **Proxylet** 和 **Netlet** 服务复选框，然后单击“确定”。新增的服务将会出现在左面板的“SRA 配置”下。

d. 单击新增的服务并建立模板文件。单击“保存”按钮。

e. 将 `/portal/netlet/jnlpclient.jar` 和 `/portal/netlet/netletjsse.jar` 添加到网关服务下非验证 URL 列表中。*

- 单击“服务配置”选项卡。
- 单击“SRA 配置”下的“网关”链接。这会列出所有可用的网关配置文件。
- 单击链接以选择适当的配置文件。
- 单击“安全”选项卡。
- 在“非验证 URL”下的编辑字段中添加 `/portal/netlet/jnlpclient.jar`，然后单击“添加”按钮。
- 在“非验证 URL”下的编辑字段中添加 `/portal/netlet/netletjsse.jar`，然后单击“添加”按钮。
- 单击页面底部的“保存”按钮。

f. 重新启动网关服务器。

```
/etc/init.d/gateway stop  
/etc/init.d/gateway start
```

11. 将 Portal Server Secure Remote Access 从发行版本 3 升级到发行版本 4。

按照第 318 页中的“从 Java ES 发行版本 3 升级 Portal Server Secure Remote Access”中的说明操作。

升级发行版本 2 Portal Server Secure Remote Access (Linux)

本节讨论影响 Portal Server Secure Remote Access 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 3 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Portal Server Secure Remote Access 软件升级到 Java ES 发行版本 3 的注意事项与在 Solaris 平台上相同（参见第 321 页中的“升级注意事项 (Solaris)”）。

下表显示了 Linux 操作系统的发行版本 3 的 Portal Server Secure Remote Access 升级修补程序：

表 18-7 在 Linux 上将 Portal Server Secure Remote Access 升级到发行版本 3 的修补程序

说明	修补程序 ID 和 RPM 名称
Portal Server 核心组件	118020-16 sun-portal-module-6.3-25.i386.rpm 其中， <i>module</i> 是约 70 个不同软件模块的任意一个
Mobile Access 核心组件	119515-01 <ul style="list-style-type: none"> • sun-mobileaccess-1.0-25.i386.rpm • sun-mobileaccess-config-1.0-25.i386.rpm
Portal Server 修复	118952-01 <ul style="list-style-type: none"> • sun-portal-core-6.3-xx.y.i386.rpm • sun-portal-configurator-6.3-xx.y.i386.rpm • sun-portal-mobileaccess-6.3-xx.y.i386.rpm • sun-portal-desktop-6.3-xx.y.i386.rpm • sun-portal-sample-6.3-xx.y.i386.rpm • sun-portal-mobileaccess-config-6.3-xx.y.i386.rpm

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Portal Server Secure Remote Access。

如果 Portal Server Secure Remote Access 与 Portal Server 安装在相同的计算机上，则按照第 291 页中的“从 Java ES 发行版本 3 升级 Portal Server”中的说明操作。然后转到第 335 页中的步骤 8。

1. 使用表 18-6 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 停止 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway stop  
/etc/init.d/netletd stop  
/etc/init.d/rwproxyd stop
```

4. 升级所有共享组件和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 329 页中的“升级 Portal Server Secure Remote Access 依赖项”。

5. 启动 Directory Server 和 Access Manager（或 Access Manager SDK）（如果它们尚未运行）。

6. 应用表 18-6 中用于 Portal Server 的 RPM。

- a. cd /tmp

- b. 解压缩 118020 修补程序文件，阅读自述文件并运行以下脚本：

```
./upgradeportalrpms
```

该更新脚本会安装 RPM 并确保因应用了修补程序而产生正确的配置变化。

- c. 解压缩 119515 修补程序文件，并按照其自述文件中的说明操作。

- d. 解压缩 118952 修补程序文件，并按照其自述文件中的说明操作。

7. 确认修补程序升级成功：

```
rpm -qa | grep sun-portal-gateway  
rpm -qa | grep sun-mobileaccess
```

应返回 RPM 的新版本号。

8. 重新配置 Portal Server Secure Remote Access 软件：

```
ksh
$ cd PortalServer-base/lib
$ ./upgradeSRA-04Q4-05Q1
```

9. 重新启动 Portal Server Secure Remote Access 进程。

```
/etc/init.d/gateway start
/etc/init.d/netletd start
/etc/init.d/rwproxyd start
```

10. 重新配置 Proxylet 和 Netlet 服务。

- a. 以管理员用户身份登录 Access Manager 控制台 (AMCONSOLE)。
- b. 移除 Proxylet 和 Netlet 服务。

在“身份管理”选项卡下，选择“服务”选项。这会在左面板上列出所有已注册的服务。在“SRA 配置”中，选中 Proxylet 和 Netlet 复选框。滚动到左面板顶部，单击“移除”按钮。这将从 ORG 级别移除 Proxylet 和 Netlet 服务。

要手动检验此步骤，可以检查 LDAP 目录（在您的组织下）以确定服务 (srapProxylet, srapNetlet) 已确实移除。

- c. 再次添加服务。

在“身份管理”选项卡下，选择“服务”选项。单击“服务”下的“添加”按钮。这会在右面板中显示所有可用的服务。选中 Proxylet 和 Netlet 服务复选框，然后单击“确定”。新增的服务将会出现在左面板的“SRA 配置”下。

- d. 单击新增的服务并建立模板文件。单击“保存”按钮。
- e. 将 /portal/netlet/jnlclient.jar 和 /portal/netlet/netletjsse.jar 添加到网关服务下非验证 URL 列表中。*
 - 单击“服务配置”选项卡。
 - 单击“SRA 配置”下的“网关”链接。这会列出所有可用的网关配置文件。

- 单击链接以选择适当的配置文件。
 - 单击“安全”选项卡。
 - 在“非验证 URL”下的编辑字段中添加
/portal/netlet/jnlpclient.jar，然后单击“添加”按钮。
 - 在“非验证 URL”下的编辑字段中添加
/portal/netlet/netletjsse.jar，然后单击“添加”按钮。
 - 单击页面底部的“保存”按钮。
- f. 重新启动网关服务器。

```
/etc/init.d/gateway stop  
/etc/init.d/gateway start
```

11. 将 Portal Server Secure Remote Access 从发行版本 3 升级到发行版本 4。

按照第 318 页中的“[从 Java ES 发行版本 3 升级 Portal Server Secure Remote Access](#)”中的说明操作。

检验升级

通过确认是否已正确应用升级修补程序来检验 Portal Server Secure Remote Access 到发行版本 3 的升级。第 322 页中的“[升级过程 \(Solaris\)](#)”和第 323 页中的“[升级过程 \(Linux\)](#)”中包括此检验的步骤。

此外，还可使用下列命令：

```
PortalServer-base/bin/gateway version  
PortalServer-base/bin/rwproxyd version  
PortalServer-base/bin/netletd version
```

参见第 320 页中的表 18-3 确认输出值。

除了对修补程序升级进行这些测试外，还可以检验先前正常运行的程序是否仍然正常，以及是否确实修复了所关注的错误。

升级后的任务

除第 330 页中的“升级过程 (Solaris)”和第 334 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚

Portal Server Secure Remote Access 从发行版本 2 到发行版本 3 的升级无法回滚。

多实例升级

在一些部署体系结构中，Portal Server Secure Remote Access 被部署在多个计算机系统上，以提供可伸缩性和提高可用性。例如，您可以在多台计算机上运行 Portal Server Secure Remote Access 组件，并使用一个负载均衡器来分配负载。

就 Portal Server Secure Remote Access 负载均衡的实例而言，可执行滚动升级，期间您可依序升级各个 Portal Server Secure Remote Access 实例而不中断服务。在升级每个 Portal Server Secure Remote Access 实例时，其他实例仍可保持运行。可按第 319 页中的“发行版本 3 Portal Server Secure Remote Access 的升级”中所述升级每个实例。

Delegated Administrator

本章介绍如何将 Delegated Administrator 升级到 Java ES 2005Q4（发行版本 4）：Sun Java System Communication Services Delegated Administrator 6.3 2005Q4。其中概括介绍了 Java ES 发行版本 4 所支持的不同升级途径的升级问题和过程。本章内容涵盖在 Solaris 和 Linux 操作系统上的升级：

- [第 340 页中的“Delegated Administrator 升级概述”](#)
- [第 343 页中的“从 Java ES 发行版本 3 升级 Delegated Administrator”](#)
- [第 351 页中的“从 Java ES 发行版本 2 升级 Delegated Administrator”](#)

注 本章中的文件位置是相对于一个称为 *DelegatedAdmin-base* 的目录路径而指定的。在最初安装 Delegated Administrator 时，可能至少已将此路径的一部分指定为安装目录。否则，Java ES 安装程序会分配一个默认值。

DelegatedAdmin-base 的默认值视操作系统平台而定：

- Solaris: /opt/SUNWcomm
 - Linux: /opt/sun/comms/commcli
-

Delegated Administrator 升级概述

本节将在以下方面对 Delegated Administrator 进行介绍，这些方面会对升级到 Java ES 2005Q4（发行版本 4）产生影响：

- [关于 Java ES 发行版本 4 Delegated Administrator](#)
- [Delegated Administrator 升级指导说明](#)
- [Delegated Administrator 数据](#)
- [兼容性问题](#)
- [Delegated Administrator 组件依赖性](#)

关于 Java ES 发行版本 4 Delegated Administrator

相对于发行版本 3 而言，Java ES 发行版本 4 Delegated Administrator 的新增功能包括日历服务置备、邮件组置备、根据可用性反馈而改进的 UI 导航以及各种错误修复。

Delegated Administrator 升级指导说明

表 19-1 显示了所支持的升级至 Java ES 发行版本 4 的 Delegated Administrator 升级途径。此表既适用于 Solaris，也适用于 Linux 操作系统。

表 19-1 升级到 Java ES 发行版本 4:
Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 的途径

Java ES 发行版本	Delegated Administrator 版本	常规方法	必需的重新配置
发行版本 3	Sun Java System Communication Services Delegated Administrator 6 2005Q1	直接升级： 通过应用修补程序和运行 config-commda 实用程序执行。	配置数据
发行版本 2	Sun Java System Communication Services User Management Utility 1.1 (2004Q2)	直接升级： 通过应用修补程序和运行 config-commda 实用程序执行。	配置数据

表 19-1 升级到 Java ES 发行版本 4:
Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 的途径 (续)

Java ES 发行版本	Delegated Administrator 版本	常规方法	必需的重新配置
发行版本 1	User Management Utility (2003Q4)	直接升级未经验证: 但可通过应用修补程序和运行 config-commda 实用程序执行。	配置数据
Java ES 以前的发行版本	iPlanet Delegated Administrator	不直接升级。	

Delegated Administrator 数据

下表显示了可能受 Delegated Administrator 软件升级影响的数据类型。

表 19-2 Delegated Administrator 数据用途

数据类型	位置	用途
目录模式	Directory Server 用户/组目录	用于支持最终用户、组织和服务模式所需的属性
配置数据	<i>DelegatedAdmin-base</i> /data/WEB-INF/classes/sun/comm/cli/server/servlet/resource.properties	Delegated Administrator 服务器配置和定制
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/com/sun/comm/da/resources/daconfig.properties	Delegated Administrator 控制台配置
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/com/sun/comm/da/resources/logger.properties	Delegated Administrator 控制台登录
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/com/sun/comm/da/resources/security.properties	Delegated Administrator 控制台配置
Web 容器配置	<p>Web Server: server.policy 和 server.xml 文件, 位于 <i>WebServer-base</i>/https-hostname/config</p> <p>Application Server (Java ES 发行版本 3 和 4): server.policy 和 domain.xml 文件, 位于 <i>AppServer8Config-base</i>/domains/domainName/config</p> <p>Application Server (Java ES 发行版本 2): server.policy 和 server.xml 文件, 位于 <i>AppServer7Config-base</i>/domains/domainName/config</p>	Delegated Administrator Web 容器实例的配置

兼容性问题

发行版本 4 Delegated Administrator 的图形用户界面有所变化，但与早期版本向后兼容。

Delegated Administrator 组件依赖性

Delegated Administrator 对其他 Java ES 组件的依赖性会影响升级和重新配置 Delegated Administrator 软件的过程。例如，Delegated Administrator 接口或功能上的变化可能要求 Delegated Administrator 所依赖组件的升级版本。是否需要升级此类组件取决于具体的升级途径。

Delegated Administrator 对以下 Java ES 组件具有依赖性：

- **共享组件。** Delegated Administrator 对特定的 Java ES 共享组件具有依赖性（参见第 38 页中的表 1-6）。
- **Web 容器。** Delegated Administrator 依赖于 Web 容器服务，这些服务可由 Java ES Web Server 或 Java ES Application Server 提供。
- **Directory Server。** Delegated Administrator 将应用程序和用户数据存储在 Directory Server 中。
- **Directory Preparation Tool。** Delegated Administrator 使用 Directory Preparation Tool 准备目录以支持 Delegated Administrator 用户置备功能。因此，Delegated Administrator 升级可能依赖于目录的准备以支持新功能。
- **Access Manager (Access Manager SDK)。** Delegated Administrator 依赖于 Access Manager 注册服务及在 Directory Server 中创建条目。

从 Java ES 发行版本 3 升级 Delegated Administrator

本节包含关于将 Delegated Administrator 从 Java ES 2005Q1（发行版本 3）升级至 Java ES 发行版本 4 的信息。其中涵盖以下主题：

- [简介](#)
- [发行版本 3 Delegated Administrator 的升级](#)

简介

将 Java ES 发行版本 3 Delegated Administrator 升级至发行版本 4 时，请考虑升级过程中的以下方面：

- **常规升级方法。** 通过向发行版本 3 的版本应用修补程序来执行升级。通过运行 `config-command` 配置实用程序来实现 Delegated Administrator 的重新配置。
- **升级依赖性。** 虽然 Delegated Administrator 对许多 Java ES 共享组件都具有依赖性（参见第 38 页中的表 1-6），但发行版本 4 Delegated Administrator 与这些组件在发行版本 3 中的版本兼容。因此，对于 Delegated Administrator 到发行版本 4 的升级，这些共享组件的升级是可选的。

此外，发行版本 4 的 Delegated Administrator 还依赖于 Web 容器和 Access Manager，如第 342 页中的“[Delegated Administrator 组件依赖性](#)”中所述。这些是软性升级依赖性；对于 Delegated Administrator 到发行版本 4 的升级，这些组件的升级是可选的。（不过，如果升级了 Access Manager，则还必须升级 Delegated Administrator。）

但是，发行版本 4 的 Delegated Administrator 对 Directory Preparation Tool 具有硬性升级依赖性；为用户置备操作准备 Directory Server 需要发行版本 4 的 Directory Preparation Tool。

- **向后兼容性。** 发行版本 4 的 Delegated Administrator 与发行版本 3 的版本向后兼容。
- **升级回滚。** 不支持 Delegated Administrator 的发行版本 4 到发行版本 3 的升级回滚。
- **平台问题。** 在 Solaris 和 Linux 操作系统上，升级 Delegated Administrator 的常规方法相同，但是修补技术却不同。因此，升级过程包括特定于平台的操作过程。

发行版本 3 Delegated Administrator 的升级

本节介绍如何在 Solaris 和 Linux 平台上将 Delegated Administrator 从 Java ES 发行版本 3 升级至 Java ES 发行版本 4。如果其中某个主题取决于具体平台的过程，则该主题将指明所适用的操作系统。本节包括以下主题：

- [升级前的任务](#)
- [升级发行版本 3 Delegated Administrator \(Solaris\)](#)
- [升级发行版本 3 Delegated Administrator \(Linux\)](#)
- [检验升级](#)
- [升级后的任务](#)
- [升级回滚 \(Solaris\)](#)

升级前的任务

升级 Delegated Administrator 之前，应执行下述任务。

检验当前版本信息

可通过输入以下命令检验 Delegated Administrator 的当前版本：

```
DelegatedAdmin-base/bin/commadmin -V
```

表 19-3 Delegated Administrator 版本检验输出

Java ES 发行版本	Delegated Administrator 版本号
发行版本 2	User Management Utility 1.1
发行版本 3	User Management Utility 6 2005Q1
发行版本 4	Delegated Administrator 6.3-0.09

应用必需的操作系统修补程序

在 Solaris 10 操作系统平台上，需要应用操作系统修补程序以执行 Delegated Administrator 升级过程（参见第 31 页中的“所需的操作系统修补程序”）。

升级 Delegated Administrator 依赖项

通常建议将计算机系统（以及计算环境）中的所有 Java ES 组件都升级到 Java ES 发行版本 4。但是，Delegated Administrator 仅对 Directory Preparation Tool 具有硬性升级依赖性。因此，Delegated Administrator 所依赖的其他 Java ES 发行版本 3 组件的升级是可选的。

但是，如果选择升级 Delegated Administrator 所依赖的所有组件，则它们应按以下顺序升级，并且应在升级 Delegated Administrator 之前进行。您可以跳过任何可能已经升级的依赖组件。

1. **共享组件。** 第 47 页中的“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级到发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 123 页中的第 6 章“Web Server”和第 157 页中的第 9 章“Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
4. **Access Manager (Access Manager SDK)。** 第 183 页中的第 11 章“Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
5. **Directory Preparation Tool。** 使用发行版本 4 的 Delegated Administrator 之前，需要在 Directory Server 中运行发行版本 4 的 Directory Preparation Tool。如果未在 Directory Server 中运行 Directory Preparation Tool，则将 Directory Preparation Tool 升级至发行版本 4，并使用它修改和扩展 Directory Server 的模式（有关过程，参见第 209 页中的第 12 章“Directory Preparation Tool”）。

备份 Delegated Administrator 数据

将 Delegated Administrator 从发行版本 3 升级到发行版本 4 需要重新配置 Delegated Administrator。为安全起见，最好对配置数据以及发行版本 3 的所有图形用户界面自定义进行备份。

获取所需的配置信息和密码

您应知道有关当前所安装版本的以下信息：

- Access Manager 管理员用户 ID 和密码
- Access Manager 内部 LDAP 密码
- 顶层管理员用户 ID 和密码
- Application Server 管理员用户 ID 和密码（如果要将 Application Server 用作 Web 容器）

升级发行版本 3 Delegated Administrator (Solaris)

本节讨论影响 Delegated Administrator 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Solaris)

将 Delegated Administrator 软件升级至 Java ES 发行版本 4 时，应考虑以下注意事项：

- 向安装的映像应用修补程序时，不应使用 Delegated Administrator。
- 重新配置 Delegated Administrator 时，应将其部署到与 Access Manager 相同的 Web 容器中。
- 下表显示了用于 Solaris 操作系统的发行版本 4 Delegated Administrator 升级修补程序：

表 19-4 在 Solaris 上升级 Delegated Administrator 的修补程序¹

	SPARC	X86
说明	Solaris 8、9 和 10	Solaris 9 和 10
Delegated Administrator	119777-09	119778-09

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Solaris)

下述过程适用于要进行升级的计算机中的 Delegated Administrator。

1. 根据表 19-4，获取所需的修补程序。
 可从以下网址将修补程序下载到 /tmp：
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>
2. 以 root 身份登录，或成为超级用户。

```
su -
```
3. 升级所有共享组件、Web 容器和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。
 参见第 345 页中的“升级 Delegated Administrator 依赖项”。
4. 应用表 19-4 中相应的 Delegated Administrator 修补程序。

```
patchadd patch_ID
```

5. 确认修补程序升级成功：

```
showrev -p | grep patch_ID
```

输出应返回在 [步骤 4](#) 中应用的修补程序 ID 的版本。

6. 重新配置 Delegated Administrator。

a. 确保 Directory Server 正在运行。

b. 确保其中运行 Access Manager 和 Delegated Administrator 的 Web 容器正在运行。

启动 Web 容器：

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

其中 Access Manager 和 Delegated Administrator 被部署在 *domainName* 域的一个服务器实例中。默认的 *domainName* 为 `domain1`，默认的服务器实例为 `server1`。

c. 运行 Delegated Administrator 配置实用程序。

```
DelegatedAdmin-base/sbin/config-commda
```

有关如何使用此实用程序的详细信息，参见《Sun Java System Communications Services 6 2005Q4 Delegated Administrator 指南》(<http://docs.sun.com/doc/819-4105>) 的第 3 章“配置 Delegated Administrator”。

7. 停止 Delegated Administrator Web 容器，然后将其重新启动。

Web Server:

```
WebServer-base/https-instanceName/stop
```

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

升级发行版本 3 Delegated Administrator (Linux)

本节讨论影响 Delegated Administrator 升级过程的一些注意事项，随后对升级过程本身加以说明。

升级注意事项 (Linux)

除 Linux 发行版本 4 的升级修补程序与 Solaris 的修补程序不同外，在 Linux 平台上将 Delegated Administrator 软件升级到 Java ES 发行版本 4 的注意事项与在 Solaris 平台上相同（参见第 346 页中的“升级注意事项 (Solaris)”）。

下表显示了用于 Linux 操作系统的发行版本 4 的 Delegated Administrator 升级修补程序：

表 19-5 在 Linux 上升级 Delegated Administrator 的修补程序¹

说明	修补程序 ID 和 RPM 名称
Delegated Administrator	119779-09 <ul style="list-style-type: none"> • sun-commcli-client-1.1-11.9.i386.rpm • sun-commcli-server-1.1-11.9.i386.rpm

1. 修补程序修订版本号是升级到 Java ES 发行版本 4 所需的最低版本号。如果有较新的修订版本可用，请用较新版本替代该表中所示的版本。

升级过程 (Linux)

下述过程适用于要进行升级的计算机中的 Delegated Administrator。

注意 在 Linux 上从 Java ES 发行版本 3 升级至 Java ES 发行版本 4 无法回滚。

1. 使用表 19-5 中的修补程序号和 RPM 名称，获取所需的修补程序。使用此信息获取 RPM 的版本号。

可从以下网址将修补程序下载到 /tmp:

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

2. 以 root 身份登录，或成为超级用户。

```
su -
```

3. 升级所有共享组件、Web 容器和 Access Manager（或 Access Manager SDK）（如果尚未执行此操作）。

参见第 345 页中的“升级 Delegated Administrator 依赖项”。

4. 应用表 19-5 中用于 Delegated Administrator 的 RPM。

```
rpm -Fvh sun-commcli-client-1.1-11.9.i386.rpm
rpm -Fvh sun-commcli-server-1.1-11.9.i386.rpm
```

5. 确认修补程序升级成功：

```
rpm -qa | grep sun-commcli
```

应返回 RPM 的新版本号。

6. 重新配置 Delegated Administrator。

- a. 确保其中运行 Access Manager 和 Delegated Administrator 的 Web 容器正在运行。

启动 Web 容器：

Web Server:

```
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

其中 Access Manager 和 Delegated Administrator 被部署在 *domainName* 域的一个服务器实例中。默认的 *domainName* 为 *domain1*，默认的服务器实例为 *server1*。

- b. 运行 Delegated Administrator 配置实用程序。

```
DelegatedAdmin-base/sbin/config-commda
```

有关如何使用此实用程序的详细信息，参见《Sun Java System Communications Services 6 2005Q4 Delegated Administrator 指南》(<http://docs.sun.com/doc/819-4105>) 的第 3 章“配置 Delegated Administrator”。

7. 停止 Delegated Administrator Web 容器，然后将其重新启动。

Web Server:

```
WebServer-base/https-instanceName/stop  
WebServer-base/https-instanceName/start
```

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName  
AppServer8-base/bin/asadmin start-domain domainName  
--user admin_ID --password password
```

检验升级

可按如下步骤检验 Delegated Administrator 的升级是否成功:

1. 检查版本号。

```
DelegatedAdmin-base/bin/commadmin -V
```

参见第 344 页中的表 19-3 确认输出值。

2. 使用 Delegated Administrator 重新配置过程中所指定的顶层管理员用户 ID 和密码登录至 Delegated Administrator 控制台。

```
http://hostName:port/da/DA/Login
```

其中 *hostName:port* 是 Delegated Administrator 重新配置过程中所提供的值。

升级后的任务

除第 346 页中的“升级过程 (Solaris)”和第 348 页中的“升级过程 (Linux)”中所述的步骤之外，没有任何升级后的任务。

升级回滚 (Solaris)

不支持 Delegated Administrator 的回滚。升级过程中所做的更改（如 Directory Server 中的条目或在将 Delegated Administrator 部署到 Web 容器中时的条目）无法轻易逆向恢复。

从 Java ES 发行版本 2 升级 Delegated Administrator

将 Java ES 2004Q2（发行版本 2）Delegated Administrator 升级至发行版本 4 的过程与将发行版本 3 Delegated Administrator 升级至发行版本 4 大致相同，下文说明了其中的几处例外。

升级 Delegated Administrator 依赖项

升级 Delegated Administrator 依赖项应包括将 Delegated Administrator 所依赖的所有共享组件（参见第 38 页中的表 1-6）以及所有本地驻留的产品组件升级至发行版本 4，除这一点外，就升级前的任务而言，将 Java ES 发行版本 2 的 Delegated Administrator 升级至发行版本 4 与将发行版本 3 的 Delegated Administrator 升级至发行版本 4 几乎相同。

升级 Delegated Administrator 依赖项时，它们应按以下顺序升级，并且应在升级 Delegated Administrator 之前进行。您可以跳过任何可能已完成升级的依赖项。

1. **共享组件。** 第 47 页中的第 2 章“升级 Java ES 共享组件”中提供了将 Java ES 共享组件升级至发行版本 4 的说明。
2. **Directory Server。** 第 91 页中的第 4 章“Directory Server 和 Administration Server”中提供了将 Directory Server 升级至发行版本 4 的说明。
3. **Web 容器软件。** 第 132 页中的“从 Java ES 发行版本 2 升级 Web Server”和第 169 页中的“从 Java ES 发行版本 2 升级 Application Server”中分别提供了升级 Web Server 和 Application Server 的说明。
4. **Access Manager (Access Manager SDK)。** 第 203 页中的“从 Java ES 发行版本 2 升级 Access Manager”中提供了将 Access Manager 升级至发行版本 4 的说明。
5. **Directory Preparation Tool。** Directory Preparation Tool 很少与 Delegated Administrator 驻留在相同的计算机上，但是，第 218 页中的“从 Java ES 发行版本 2 升级 Directory Preparation Tool”中提供了升级 Directory Preparation Tool 和在 Directory Server 上运行 Directory Preparation Tool 的说明。

发行版本 2 Delegated Administrator 的升级

将 Delegated Administrator 从发行版本 2 升级至发行版本 4 的过程视您在其中部署 Delegated Administrator 软件的 Web 容器而定。

升级发行版本 2 Delegated Administrator: Web Server Web 容器

当部署到已经升级至发行版本 4 的 Web Server Web 容器中时，要将发行版本 2 的 Delegated Administrator 升级至发行版本 4，可按照第 346 页中的“升级发行版本 3 Delegated Administrator (Solaris)”或第 348 页中的“升级发行版本 3 Delegated Administrator (Linux)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

升级发行版本 2 Delegated Administrator: Application Server Web 容器

当部署到已经升级至发行版本 4 的 Application Server Web 容器中时，要将发行版本 2 的 Delegated Administrator 升级至发行版本 4，首先应按照第 346 页中的“升级发行版本 3 Delegated Administrator (Solaris)”或第 348 页中的“升级发行版本 3 Delegated Administrator (Linux)”中的说明操作，但需要在提到发行版本 3 的地方用发行版本 2 代替。

最初在其中部署 Delegated Administrator 的发行版本 2 Application Server 实例 (*instanceName*) 会在升级到发行版本 4 之后，被迁移到升级过程所创建的一个节点代理下。

这种情况下，升级 Delegated Administrator 需要如下更改第 347 页 (Solaris) 或第 349 页 (Linux) 中的步骤 6 “重新配置 Delegated Administrator”：

5. 重新配置 Delegated Administrator。

- a. 修改 *AccessManagerConfig-base/config/AMConfig.properties* 文件。

将下行：

```
com.sun.identity.webcontainer=IAS7.0
```

替换为：

```
com.sun.identity.webcontainer=IAS8.1
```

- b. 确保在其中部署了 Delegated Administrator 的已升级 Application Server 实例 (*instanceName*) 正在运行。

要启动 Application Server 实例，请启动 Domain Administration Server (DAS) 和已将实例迁移到其下的节点代理：

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

在上述命令及后续步骤中，将使用下面的约定：

- *nodeAgentName* 的形式为 *hostName_domainName*。
 - 默认的 *domainName* 为 *domain1*
 - 默认的 *instanceName* 为 *server1*
- c. 取消 `commcli` Delegated Administrator Web 应用程序在 Application Server 实例 (*instanceName*) 中的部署。

```
AppServer8-base/bin/asadmin undeploy --secure=false --user admin
--password password --target instanceName commcli
```

- d. 运行 Delegated Administrator 配置实用程序。

```
DelegatedAdmin-base/sbin/config-commda
```

有关如何使用此实用程序的详细信息，参见《Sun Java System Communications Services 6 2005Q4 Delegated Administrator 指南》(<http://docs.sun.com/doc/819-4105>) 的第 3 章“配置 Delegated Administrator”。

指定下列参数：

- 当要求提供 Access Manager 主机和端口时，指定 DAS 实例的端口（默认值为 8080）。
- 当询问部署 Delegated Administrator 控制台和 Delegated Administrator 服务器的位置时，同样指定 DAS 实例信息（默认端口为 8080），而非已升级 Application Server 实例 (*instanceName*) 的信息。

这些参数值将使得 Delegated Administrator 重新部署到 DAS 实例中。这不是标准做法，但却可行。

- e. 将 Access Manager 类路径信息 `classpath-prefix` 和 `classpath-suffix` 从 `instanceName` 实例的 `domain.xml` 文件:

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/  
config/domain.xml
```

复制到 DAS (默认名称为 `server`) 的 `domain.xml` 文件中:

```
AppServer8Config-base/domains/domainName/config/domain.xml
```

其中, 默认的 `domainName` 为 `domain1`。

也可从 Application Server 管理控制台执行此步骤。

- f. 重新启动 DAS。

DAS 是已在其中重新部署了 Delegated Administrator 的服务器实例。

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID  
--password password domainName
```

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

Java Enterprise System 发行版本内容

本附录列出了各个 Java Enterprise System 发行版本的内容。包括以下各节：

- 第 356 页中的“Java ES 2003Q4（发行版本 1）”
- 第 358 页中的“Java ES 2004Q2（发行版本 2）”
- 第 361 页中的“Java ES 2005Q1（发行版本 3）”
- 第 366 页中的“Java ES 2005Q4（发行版本 4）”

Java ES 2003Q4 (发行版本 1)

本节列出 Java Enterprise System 2003Q4 的内容。

发行版本 1 安装程序可选组件

Sun Open Network Environment (Sun ONE) 和 Sun Cluster 组件产品提供了支持分布式企业应用程序所需的基础结构服务。这些组件产品包括：

- Sun Cluster 3.1 和 Sun Cluster Agents for Sun ONE
- Sun ONE Administration Server 5.2
- Sun ONE Application Server 7 Update 1
- Sun ONE Calendar Server 6.0
- Sun ONE Directory Server 5.2
- Sun ONE Directory Proxy Server 5.2
- Sun ONE Identity Server 6.1
- Sun ONE Instant Messaging 6.1
- Sun ONE Message Queue 3.0.1 Service Pack 2
- Sun ONE Messaging Server 6.0
- Sun ONE Portal Server 6.2
- Sun ONE Portal Server, Secure Remote Access 6.2
- Sun ONE Web Server 6.1

发行版本 1 共享组件

共享组件提供组件产品所依赖的本地服务和技术支持。安装组件产品时，Java ES 安装程序自动安装所需的共享组件（如果尚未安装）。

Java Enterprise System 包括以下共享组件：

- ANT (Jakarta ANT Java/XML-based build tool)
- Apache Commons Logging
- ICU (International Components for Unicode)
- J2SE™ platform 1.4.1_06 (Java 2 Platform, Standard Edition)
- JAF (JavaBeans™ Activation Framework)
- JATO (Sun ONE Application Framework)
- JavaHelp™ Runtime
- JAXM (Java API for XML Messaging) Client Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries)
- JAX-RPC (Java APIs for XML-based Remote Procedure Call)
- JSS (Java Security Services)
- KT search engine
- LDAP C Language SDK
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- SAAJ (SOAP with Attachments API for Java)
- SASL (Simple Authentication and Security Layer)
- XML C Library (libxml)

注 Application Server 和 Directory Server 还要求在系统中安装 Perl，但 Perl 不会作为 Java ES 共享组件自动安装。

Java ES 2004Q2 (发行版本 2)

本节列出 Java Enterprise System 2004Q2 的内容。

发行版本 2 安装程序可选组件

组件产品提供支持分布式企业应用程序所需的基础结构服务。在特定主机上安装 Java Enterprise System 时，可根据总体部署体系结构来选择要将哪些组件产品安装到该主机。

Java Enterprise System 2005Q4 包括以下组件产品：

通信和协作服务

- Sun Java System Messaging Server 6 2004Q2
- Sun Java System Calendar Server 6 2004Q2
- Sun Java System Instant Messaging 6 2004Q2
- Sun Java System Portal Server 2004Q2
- Sun Java System Portal Server Mobile Access 2004Q2
- Sun Java System Portal Server Secure Remote Access 2004Q2
- Sun Java System Communications Express 6 2004Q2

Web 和应用程序服务

- Sun Java System Application Server 7.0 Update 3 (Standard and Platform Editions)
- Sun Java System Web Server 6 2004Q2 Update 1 Service Pack 2
- Sun Java System Message Queue 3.5 SP1 (Platform and Enterprise Editions)

目录和身份认证服务

- Sun Java System Identity Server 6.2 2004Q2, 包括 Sun Java System Communications Services 6 2004Q2 User Management Utility
- Sun Java System Directory Server 5 2004Q2
- Sun Java System Directory Proxy Server 5 2004Q2

可用性服务

- Sun Cluster 3.1 4/04 和 Sun Cluster Agents for Sun Java System

管理服务

- Sun Java System Administration Server 5 2004Q2
- Sun Remote Services Net Connect 3.5

注意，在 Linux 操作系统中，Sun Cluster、Sun Cluster Agents 和 Sun Remote Services Net Connect 不可用。

发行版本 2 共享组件

共享组件提供组件产品所依赖的本地服务和技术支持。安装组件产品时，Java ES 安装程序自动安装所需的共享组件（如果尚未安装）。

Java Enterprise System 2005Q4 共享组件包括：

- Ant (Jakarta ANT Java/XML-based build tool)
- Apache Commons Logging
- Apache SOAP (Simple Object Access Protocol)
- ICU (International Components for Unicode)
- J2SE™ platform 1.4.2_04 (Java 2 Platform, Standard Edition)
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Application Framework)
- JavaHelp™ Runtime
- JAXB (Java Architecture for XML Binding)
- JAXM (Java API for XML Messaging) Client Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries)

- JAX-RPC (Java APIs for XML-based Remote Procedure Call)
- JCAPI (Java Calendar API)
- JSS (Java Security Services)
- KT search engine
- LDAP C Language SDK
- LDAP Java SDK
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- Perl LDAP, 包括 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
- Sun Explorer Data Collector
- XML C Library (libxml)

Java ES 2005Q1 (发行版本 3)

本节列出 Java Enterprise System 2005Q1 的内容。

发行版本 3 安装程序可选组件

在 Java ES 安装程序的组件选择页面中，可选组件按其帮助提供的服务分组。下面的列表还显示了随每个组件安装子组件。

通信和协作服务

- 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 ; 包括服务器和多路复用器软件
 - 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 Tool

Web 和应用程序服务

- Sun Java System Application Server Enterprise Edition 8.1 2005Q1
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin

可以与 Web Server 或 Apache Web Server 一起使用，在配置时可以选择。
默认情况下与 Web Server 一起使用。

 - PointBase
 - 样例应用程序
- Sun Java System Web Server 6 2005Q1 Update 1 Service Pack 4
- Sun Java System Message Queue 3 2005Q1

目录和身份认证服务

- Sun Java System Access Manager 6.3 2005Q1

Portal Server 和 Messaging Server 的 Delegated Administrator 置备工具自动随 Access Manager 安装。

 - Identity Management and Policy Services Core (包括 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

可用性服务

- Sun Cluster 3.1 9/04
 - Sun Cluster Core
- Sun Cluster Agents for Sun Java System
 - HA/Scalable Sun Java System Web Server
 - HA Sun Java System Message Queue
 - HA Sun Java System Portal Server
 - HA Sun Java System Administration Server
 - HA Sun Java System Directory Server
 - HA Sun Java System Messaging Server
- HADB (用于高可用性会话存储)

管理服务

- Sun Java System Administration Server 5 2005Q1
- SunSM Remote Services Net Connect 3.1.1

注 Sun Cluster、Sun Cluster Agents 和 Sun Remote Services Net Connect 在 Solaris 10 或 Linux 操作系统中不可用。

Sun Remote Services Net Connect 在 Solaris x86 平台中不可用。

发行版本 3 共享组件

共享组件为可选组件提供本地服务和技术支持。安装 Java ES 组件时，安装程序自动安装所需的共享组件（如果尚未安装）。

此版本的 Java ES 包括以下共享组件：

- 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)
- J2SE™ (Java 2 Platform, Standard Edition) platform 5.0
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio Enterprise Web Application Framework)
- JavaHelp™ Runtime
- JavaMail™ Runtime
- 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, 包括 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
- Sun Explorer Data Collector (仅限 Solaris)
- Sun Java Monitoring Framework
- Sun Java Web Console
- Tomcat Servlet JSP Container
- XML C Library (libxml)
- WSCL (Web services Common Library)

Java ES 2005Q4 (发行版本 4)

本节列出 Java Enterprise System 2005Q4 的内容。

发行版本 4 安装程序可选组件

在 Java ES 安装程序的组件选择页面中，可选组件按其帮助提供的服务分组。下面的列表还显示了随每个组件安装的子组件。

通信和协作服务

- 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 ; 包括服务器和多路复用器软件
 - 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 6.3.1 2005Q4
 - Secure Remote Access Core
 - Gateway
 - Netlet Proxy
 - Rewriter Proxy
- Sun Java System Communications Express 6.2 2005Q4
- Sun Java System Directory Preparation Tool 6.3 2005Q4
- Sun Java System Communications Services Delegated Administrator 6.3 2005Q4
 - Delegated Administrator Console and Utility
 - Delegated Administrator Server

Web 和应用程序服务

- Sun Java System Application Server Enterprise Edition 8.1 2005Q4
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin

可以与 Web Server 或 Apache Web Server 一起使用，在配置时可以选择。默认情况下与 Web Server 一起使用。

- PointBase Database
- 样例应用程序
- Sun Java System Web Server 6.1 Service Pack 5 2005Q4
- Sun Java Web Proxy Server 4.0.1 2005Q4
- Sun Java System Message Queue Enterprise Edition 3.6 SP3 2005Q4
- Sun Java Service Registry 3.0

目录和身份认证服务

- Sun Java System Access Manager 7.0 2005Q4
 - Identity Management and Policy Services Core
 - Access Manager Administration Console
 - Common Domain Services for Federation Management
 - Access Manager SDK
- Sun Java System Directory Server 5.2 2005Q4
- Sun Java System Directory Proxy Server 5.2 2005Q4

可用性服务

- Sun Cluster 3.1 8/05
 - Sun Cluster Core
 - Sun Cluster Agents for Sun Java System
 - HA Sun Java System Directory Server
 - HA Sun Java System Administration Server
 - HA/Scalable Sun Java System Web Server
 - HA Sun Java System Message Queue
 - HA Sun Java System Application Server
 - HA Sun Java System Messaging Server
 - HA Sun Java System Calendar Server
 - HA Sun Java System Instant Messaging
- High Availability Session Store (HADB) 4.4.2

管理服务

- Sun Java System Administration Server 5.2 2005Q4

注 Sun Cluster、Sun Cluster Agents 和 Sun Remote Services Net Connect 在 Solaris 10 或 Linux 操作系统中不可用。

Sun Remote Services Net Connect 在 Solaris x86 平台中不可用。

发行版本 4 共享组件

共享组件为可选组件提供本地服务和技术支持。安装 Java ES 组件时，安装程序自动安装所需的共享组件（如果尚未安装）。

此版本的 Java ES 包括以下共享组件：

- ANT (Jakarta ANT Java/XML-based build tool)
- ACL (Apache Commons Logging)
- BDB (Berkeley Database)
- CAC (Common agent container)
- Derby Database
- ICU (International Components for Unicode)
- IM-SDK (Instant Messenger SDK)
- J2SE™ (Java 2 Platform, Standard Edition) platform 5.0
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio Enterprise Web Application Framework)
- JavaHelp™ Runtime
- JavaMail™ Runtime
- 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
- MA (Mobile Access) Core
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- SAAJ (SOAP runtime with Attachments API for Java)
- SASL (Simple Authentication and Security Layer)
- SEDC (Sun Explorer Data Collector, 仅限 Solaris)
- MFWK (Java ES Monitoring Framework)
- SJWC (Sun Java Web Console)
- WSCL (Web services Common Library)

A

Access Manager

- 产品组件依赖性 41
- 共享组件依赖性 38
- 缩写 26
- 子组件 367

AccessManager-base 路径 183

AccessManagerConfig-base 路径 183

ACL 共享组件

- 从发行版本 2 升级 53
- 从发行版本 3 升级 51
- 全名 27
- 软件包版本 62, 65
- 依赖性表中 38

Administration Server

- 产品组件依赖性 41
- 共享组件依赖性 38
- 缩写 26
- 修补程序 98, 102

amconfig 脚本 (Access Manager) 196

ampre70upgrade 脚本 (Access Manager) 193

amupgrade 脚本 (Access Manager) 198

ANT 共享组件

- 从发行版本 2 升级 53
- 从发行版本 3 升级 51
- 全名 27
- 软件包版本 62, 65
- 依赖性表中 38

Apache Commons Logging, 参见 ACL

Application Server

- 安装后配置 173
- 产品组件依赖性 41
- 共享组件依赖性 38
- 缩写 26
- 子组件 367

AppServer7-base 路径 157

AppServer7Config-base 路径 157

AppServer8-base 路径 157

AppServer8Config-base 路径 157

asant 脚本 (Application Server) 164

asupgrade 脚本 (Application Server) 173

安全声明标记语言, 参见 SAML

B

BDB 共享组件

- 从发行版本 2 升级 53
- 从发行版本 3 升级 51
- 全名 27
- 软件包版本 62, 65
- 依赖性表中 38

BEA WebLogic Server, 作为 Web 容器 41, 43

Berkeley Database, 参见 BDB

C

C

CAC 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

全名 27

软件包版本 62, 65

依赖性表中 38

Calendar Server

产品组件依赖性 41

共享组件依赖性 38

缩写 26

CalendarServer-base 路径 241

comm_dssetup.pl (Dir. Prep. Tool) 脚本 215

Common agent container, 参见 CAC

CommsExpress-base 路径 253

Communications Express

产品组件依赖性 42

共享组件依赖性 38

缩写 26

产品组件

对共享组件的依赖性 37

Java ES 2003Q4 (发行版本 1) 356

Java ES 2004Q2 (发行版本 2) 358

Java ES 2005Q1 (发行版本 3) 361

Java ES 2005Q4 (发行版本 4) 366

相互依赖性 41

D

db2bak 实用程序 97, 117, 191

Delegated Administrator

产品组件依赖性 42

共享组件依赖性 38

缩写 26

DelegatedAdmin-base 路径 339

Derby Database 共享组件 369

Directory Proxy Server

产品组件依赖性 42

共享组件依赖性 38

缩写 26

修补程序 98, 102

Directory Server

产品组件依赖性 42

共享组件依赖性 38

缩写 26

修补程序 98, 102

DirPrepTool-base 路径 209

F

符号约定 21

服务

srapNetlet 331, 335

srapProxylet 331, 335

G

更新脚本 (Portal Server SRA) 324

更新脚本 (Portal Server) 297

共享组件

Java ES 2003Q4 (发行版本 1) 357

Java ES 2004Q2 (发行版本 2) 359

Java ES 2005Q1 (发行版本 3) 364

Java ES 2005Q4 (发行版本 4) 369

相互依赖性 39

依赖产品组件 37

H

HADB-base 路径 149

High Availability Session Store

缩写 26

I

ICU 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

全名 27

软件包版本 65

依赖性表中 38

IM-SDK 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

全名 27

软件包版本 62, 65

依赖性表中 38

install_cluster (修补程序群组) 脚本 56

Instant Messaging

产品组件依赖性 42

共享组件依赖性 38

缩写 26

Instant Messenger SDK, 参见 IM-SDK

InstantMessaging-base 路径 269

International Components for Unicode, 参见 ICU

J

J2SE 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

全名 27

软件包版本 62, 65

依赖性表中 38

JAF 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

全名 27

依赖性表中 38

Jakarta ANT Java/XML-based build tool, 参见 ANT

JATO 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

全名 27

软件包版本 62, 65

依赖性表中 38

Java 2 Platform, Standard Edition, 参见 J2SE

Java API for XML Processing, 参见 JAXP

Java API for XML Registries, 参见 JAXR

Java API for XML-based Remote Procedure Call,
参见 JAX-RPC

Java Architecture for XML Binding, 参见 JAXB

Java Calendar API, 参见 JCAPI

Java Dynamic Management Kit, 参见 JDMK

Java ES 2003Q4 (发行版本 1)

产品组件 356

共享组件 357

Java ES 2004Q2 (发行版本 2)

产品组件 358

共享组件 359

Java ES 2005Q1 (发行版本 3)

产品组件 361

共享组件 364

Java ES 2005Q4 (发行版本 4)

产品组件 366

共享组件 369

Java ES Monitoring Framework, 参见 MFWK

Java ES 修补程序群组脚本 57, 71

Java Security Services, 参见 JSS

Java Studio Enterprise Web Application Framework,
参见 JATO

JavaBeans Activation Framework, 参见 JAF

JavaHelp 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

软件包版本 62, 65

依赖性表中 38

JavaMail 共享组件

从发行版本 2 升级 53

从发行版本 3 升级 51

依赖性表中 38

JAXB 共享组件

从发行版本 2 升级 53

K

- 从发行版本 3 升级 51
- 全名 27
- 依赖性表中 38
- JAXP 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 51
 - 全名 27
 - 依赖性表中 38
- JAXR 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 51
 - 全名 27
 - 依赖性表中 38
- JAX-RPC 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 51
 - 全名 28
 - 依赖性表中 38
- JCAPI 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 52
 - 全名 28
 - 依赖性表中 38
- JDMK 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 52
 - 全名 28
 - 依赖性表中 38
- JHELP 共享组件, 参见 JavaHelp
- JMAIL 共享组件, 参见 JavaMail
- JSP 文件, 自定义 186, 192, 195
- JSS 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 52
 - 全名 28
 - 软件包版本 65
 - 依赖性表中 38
- 脚本
 - amconfig (Access Manager) 196
 - ampre70upgrade (Access Manager) 193
 - amupgrade (Access Manager) 198
 - asant (Application Server) 164
 - asupgrade (Application Server) 173
 - comm_dssetup.pl (Dir. Prep. Tool) 215
 - 更新 (Portal Server SRA) 324
 - 更新 (Portal Server) 297
 - install_cluster (修补程序群组) 56
 - Java ES 修补程序群组 57, 71
 - mqmigrate (Message Queue) 146
 - mqupgrade (Message Queue) 142
 - multiserverinstance (Portal Server) 307
 - postInstall (Application Server) 174
 - upgradeportalrps (Portal Server) 311, 313

K

KT Search Engine, 参见 KTSE

KTSE 共享组件

- 从发行版本 2 升级 53
- 从发行版本 3 升级 52
- 全名 28
- 软件包版本 65
- 依赖性表中 38

L

LDAP C Language SDK 共享组件 370

LDAP C SDK 共享组件

- 从发行版本 2 升级 53
- 从发行版本 3 升级 52
- 全名 28
- 依赖性表中 38

LDAP J SDK 共享组件 370

- 从发行版本 2 升级 54
- 从发行版本 3 升级 52
- 全名 28
- 软件包版本 65
- 依赖性表中 38

M

MA Core 共享组件

从发行版本 2 升级 [54](#)

从发行版本 3 升级 [52](#)

全名 [28](#)

依赖性表中 [38](#)

Message Queue

产品组件依赖性 [43](#)

共享组件依赖性 [38](#)

缩写 [26](#)

Messaging Server

共享组件依赖性 [38](#)

缩写 [26](#)

MessagingServer-base 路径 [223](#)

MFWK 共享组件

从发行版本 3 升级 [52](#)

全名 [28](#)

软件包版本 [65](#)

依赖性表中 [38](#)

Mobile Access Core, 参见 MA core

mqupgrade 脚本 (Message Queue) [142](#), [146](#)

multiserverinstance (Portal Server) 脚本 [307](#)

N

Netscape Portable Runtime, 参见 NSPR

Network Security Services, 参见 NSS

NSPR 共享组件

从发行版本 2 升级 [54](#)

从发行版本 3 升级 [52](#)

全名 [28](#)

软件包版本 [65](#)

依赖性表中 [38](#)

NSS 共享组件

从发行版本 2 升级 [54](#)

从发行版本 3 升级 [52](#)

全名 [28](#)

软件包版本 [65](#)

依赖性表中 [38](#)

P

Portal Server

产品组件依赖性 [43](#)

共享组件依赖性 [38](#)

Secure Remote Access

缩写 [27](#)

缩写 [26](#)

Portal Server Secure Remote Access

产品组件依赖性 [43](#)

共享组件依赖性 [38](#)

PortalServer-base 路径 [287](#)

PortalServerConfig-base 路径 [287](#)

postInstall 脚本 (Application Server) [174](#)

Q

群集升级 [173](#)

S

SAAJ 共享组件

从发行版本 2 升级 [53](#)

从发行版本 3 升级 [51](#)

全名 [28](#)

依赖性表中 [38](#)

SAML [200](#), [208](#)

SASL 共享组件

从发行版本 2 升级 [54](#)

从发行版本 3 升级 [52](#)

全名 [28](#)

软件包版本 [66](#)

依赖性表中 [39](#)

SEDC 共享组件

从发行版本 2 升级 [54](#)

从发行版本 3 升级 [52](#)

全名 [28](#)

软件包版本 [63](#)

依赖性表中 [39](#)

U

- serverRoot* 路径 (Directory Server) 91
- Service Registry
 - 缩写 27
- Simple Authentication and Security Layer, 参见 SASL
- SJWC 共享组件
 - 全名 28
 - 软件包版本 66
 - 依赖性表中 39
- SOAP runtime with Attachments API for Java, 参见 SAAJ
- srapNetlet 服务 331, 335
- srapProxylet 服务 331, 335
- Sun Cluster
 - 产品组件依赖性 43
 - 共享组件依赖性 38
 - 缩写 27
- Sun Cluster Agents 368
- Sun Explorer Data Collector, 参见 SEDC
- Sun Java Web Console, 参见 SJWC

U

- upgradeportalrps 脚本 (Portal Server) 311, 313

W

- Web Proxy Server
 - 产品组件依赖性 43
 - 共享组件依赖性 38
 - 缩写 27
- Web Server
 - 产品组件依赖性 43
 - 共享组件依赖性 38
 - 缩写 27
- Web services Common Library, 参见 WSCL
- WebProxyServer-base* 路径 175
- WebServer-base* 路径 123

- WSCL 共享组件
 - 从发行版本 2 升级 53
 - 从发行版本 3 升级 51
 - 全名 28
 - 依赖性表中 39

X

- 修补程序群组, Solaris OS 57

Y

- 依赖性
 - 产品组件, 对共享组件 37
 - 产品组件间 41
 - 共享组件之间 39
- 印刷约定 20
- 约定
 - 符号 21
 - 印刷 20