



Sun Java™ System

# Sun Java Enterprise System 2005Q4 アップグレードガイド

---

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

Part No: 819-3456

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

Sun Microsystems, Inc. は、この製品に含まれるテクノロジーに関する知的所有権を保持しています。特に限定されることなく、これらの知的所有権は <http://www.sun.com/patents> に記載されている 1 つ以上の米国特許および米国およびその他の国における 1 つ以上の追加特許または特許出願中のものが含まれている場合があります。

このソフトウェアは SUN MICROSYSTEMS, INC. の機密情報と企業秘密を含んでいます。SUN MICROSYSTEMS, INC. の書面による許諾を受けることなく、このソフトウェアを使用、開示、複製することは禁じられています。

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

この配布には、第三者が開発したソフトウェアが含まれている可能性があります。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company, Ltd が独占的にライセンスしている米国およびその他の国における登録商標です。

Sun、Sun Microsystems、Sun のロゴマーク、Java、Solaris、JDK、Java Naming and Directory Interface、JavaMail、JavaHelp、J2SE、iPlanet、Duke のロゴマーク、Java Coffee Cup のロゴ、Solaris のロゴ、SunTone 認定ロゴマークおよび Sun ONE ロゴマークは、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャに基づくものです。

Legato および Legato のロゴマークは Legato Systems, Inc. の商標であり、Legato NetWorker は同社の商標または登録商標です。

Netscape Communications Corp のロゴマークは Netscape Communications Corporation の商標または登録商標です。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカルユーザインタフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

この製品は、米国の輸出規制に関する法規の適用および管理下にあり、また、米国以外の国の輸出および輸入規制に関する法規の制限を受ける場合があります。核、ミサイル、生物化学兵器もしくは原子力船に関連した使用またはかかる使用者への提供は、直接的にも間接的にも、禁止されています。このソフトウェアを、米国の輸出禁止国へ輸出または再輸出すること、および米国輸出制限対象リスト (輸出が禁止されている個人リスト、特別に指定された国籍者リストを含む) に指定された、法人、または団体に輸出または再輸出することは一切禁止されています。

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

# 目次

<b>表目次</b> .....	<b>15</b>
<b>はじめに</b> .....	<b>19</b>
対象読者 .....	19
本書で使用する表記規則 .....	20
表記上の規則 .....	20
記号 .....	20
シェルプロンプト .....	21
関連マニュアル .....	22
このドキュメントセット内のマニュアル .....	22
オンライン上の Sun リソースへのアクセス .....	24
Sun テクニカルサポートへの問い合わせ .....	24
関連するサードパーティーの Web サイト .....	24
ご意見、ご要望の送付先 .....	25
<b>第 1 章 アップグレードの計画</b> .....	<b>27</b>
Java ES 2005Q4 (Release 4) コンポーネント .....	27
Release 4 の製品コンポーネント .....	28
Release 4 の共有コンポーネント .....	29
Java ES のアップグレードについて .....	30
製品コンポーネントのアップグレード .....	30
共有コンポーネントのアップグレード .....	31
アップグレードテクノロジー .....	32
オペレーティングシステムの問題 .....	33
必須のオペレーティングシステムパッチ .....	33
マイナーリリースのアップグレード .....	33
サポートされないプラットフォームへのアップグレード .....	33

アップグレードの計画 .....	34
アップグレード計画とは .....	35
アップグレード計画の注意点 .....	36
アップグレードパス .....	36
アップグレードの依存関係 .....	38
選択的なアップグレードとすべてのアップグレード .....	39
複数インスタンスのアップグレード .....	40
Java ES コンポーネントの依存関係 .....	41
共有コンポーネントの依存関係 .....	42
共有コンポーネントの依存関係のマトリックス .....	42
共有コンポーネントのアップグレードのガイドライン .....	44
製品コンポーネントの依存関係 .....	45
手順に関する一般的なガイドライン .....	51
<b>第 2 章 Java ES 共有コンポーネントのアップグレード .....</b>	<b>55</b>
共有コンポーネントのアップグレードの概要 .....	56
アップグレード計画について .....	56
共有コンポーネントのアップグレードに関するテクノロジー .....	57
一般的なアップグレード手順 .....	58
Release 3 の共有コンポーネントのアップグレード .....	59
Release 2 の共有コンポーネントのアップグレード .....	61
個別のパッチを適用することによる共有コンポーネントのアップグレード .....	64
Java ES 2005Q4 へのパッチアップグレード .....	65
パッチアップグレードのロールバック .....	65
パッチクラスタを使用した共有コンポーネントのアップグレード .....	66
パッチクラスタの内容 .....	66
パッチクラスタの手順 .....	67
Solaris 8 から Solaris 9 へのアップグレード .....	67
Solaris 10 でのアップグレード (Java ES Release 3 からのみ) .....	68
パッケージの置換によるコンポーネントのアップグレード .....	70
パッケージ置換のアップグレード戦略 .....	70
アップグレードパス .....	71
パッケージバージョン .....	71
パッケージの場所 .....	71
パッケージ置換の手順 .....	72
特別な手順 .....	72
Solaris プラットフォームのパッケージ .....	72
Solaris プラットフォームのパッケージのアップグレード .....	75
Linux プラットフォームのパッケージ .....	76
Linux プラットフォームのパッケージのアップグレード .....	79
特別なアップグレード手順を必要とするコンポーネント .....	80
セキュリティーコンポーネント (NSS、NSPR、JSS) のアップグレード .....	80
共通エージェントコンテナのアップグレード .....	81

Solaris プラットフォームでの Java ES Release 2 からのアップグレード	81
Linux プラットフォームでの Java ES Release 2 からのアップグレード	81
Java ES Release 3 からのアップグレード	81
JATO のアップグレード	84
Solaris プラットフォームでの Java ES Release 3 からの JATO のアップグレード	84
Solaris プラットフォームでの Java ES Release 2 からの JATO のアップグレード	84
Linux プラットフォームでの Java ES Release 3 からの JATO のアップグレード	85
Linux プラットフォームでの Java ES Release 2 からの JATO のアップグレード	85
Linux プラットフォームの JavaHelp のアップグレード	85
Sun Java Web Console のアップグレード	85
Sun Explorer Data Collector のアップグレード	86
Java ES Release 4 のための J2SE のアップグレード	86
Solaris プラットフォームの J2SE のアップグレード	87
Linux プラットフォームの J2SE のアップグレード	93
<b>第 3 章 Sun Cluster ソフトウェア</b>	<b>97</b>
Sun Cluster ソフトウェアアップグレードの概要	97
Java ES Release 4 Sun Cluster ソフトウェアについて	98
Sun Cluster ソフトウェアアップグレードのロードマップ	98
Sun Cluster のデータ	99
互換性の問題	99
Sun Cluster の依存関係	99
Sun Cluster ソフトウェアの Java ES Release 4 へのアップグレード	100
はじめに	100
Sun Cluster のアップグレード	101
アップグレード前の作業	101
Sun Cluster ソフトウェアのアップグレード	103
アップグレードの確認	104
アップグレード後の作業	104
アップグレードのロールバック	104
<b>第 4 章 Directory Server と Administration Server</b>	<b>105</b>
Directory Server と Administration Server のアップグレードの概要	106
Java ES Release 4 について	106
Java ES Release 4 へのアップグレードのロードマップ	106
Directory Server と Administration Server のデータ	108
互換性の問題	108
依存関係	108
Java ES Release 3 からの Directory Server と Administration Server のアップグレード	109
はじめに	109
Release 3 の Directory Server と Administration Server のアップグレード	110
アップグレード前の作業	110

Release 3 の Directory Server および Administration Server のアップグレード (Solaris) ..	112
Release 3 の Directory Server および Administration Server のアップグレード (Linux) ....	116
アップグレードの確認 .....	120
アップグレード後の作業 .....	120
アップグレードのロールバック (Solaris) .....	121
複数インスタンスのアップグレード .....	123
マルチマスターレプリケーションのローリングアップグレード .....	123
データサービスとしての Directory Server のアップグレード .....	123
Java ES Release 2 からの Directory Server と Administration Server のアップグレード .....	125
<b>第 5 章 Directory Proxy Server .....</b>	<b>127</b>
Directory Proxy Server のアップグレードの概要 .....	128
Java ES Release 4 について .....	128
Java ES Release 4 へのアップグレードのロードマップ .....	128
Directory Proxy Server のデータ .....	129
互換性の問題 .....	129
依存関係 .....	130
Java ES Release 3 からの Directory Proxy Server のアップグレード .....	130
はじめに .....	130
Release 3 の Directory Proxy Server のアップグレード .....	131
アップグレード前の作業 .....	131
Release 3 の Directory Proxy Server のアップグレード (Solaris) .....	133
Release 3 の Directory Proxy Server のアップグレード (Linux) .....	135
アップグレードの確認 .....	137
アップグレード後の作業 .....	137
アップグレードのロールバック (Solaris) .....	137
複数インスタンスのアップグレード .....	138
Java ES Release 2 からの Directory Proxy Server のアップグレード .....	139
<b>第 6 章 Web Server .....</b>	<b>141</b>
Web Server のアップグレードの概要 .....	142
Java ES Release 4 の Web Server について .....	142
Web Server のアップグレードロードマップ .....	142
Web Server のデータ .....	143
互換性の問題 .....	143
Web Server 依存関係 .....	143
Java ES Release 3 からの Web Server のアップグレード .....	144
はじめに .....	144
Release 3 の Web Server のアップグレード .....	144
アップグレード前の作業 .....	145
Release 3 の Web Server のアップグレード (Solaris) .....	146
Release 3 の Web Server のアップグレード (Linux) .....	147

アップグレードの確認 .....	149
アップグレード後の作業 .....	149
アップグレードのロールバック (Solaris) .....	149
Java ES Release 2 からの Web Server のアップグレード .....	150
<b>第 7 章 Message Queue .....</b>	<b>151</b>
Message Queue のアップグレードの概要 .....	152
Java ES Release 4 の Message Queue について .....	152
Message Queue のアップグレードロードマップ .....	152
Message Queue のデータ .....	154
互換性の問題 .....	155
プロトコルの互換性 .....	155
ブローカの互換性 .....	155
管理対象オブジェクトの互換性 .....	156
管理ツールの互換性 .....	156
クライアントの互換性 .....	157
Message Queue の依存関係 .....	157
Java ES Release 3 からの Message Queue のアップグレード .....	158
はじめに .....	158
Release 3 の Message Queue のアップグレード .....	159
アップグレード前の作業 .....	159
Release 3 の Message Queue のアップグレード .....	160
Message Queue のアップグレードの確認 .....	162
アップグレード後の作業 .....	162
アップグレードのロールバック .....	162
複数インスタンスのアップグレード .....	163
Java ES Release 2 からの Message Queue のアップグレード .....	164
Release 2 の Message Queue のアップグレード (Solaris) .....	164
Release 2 の Message Queue のアップグレード (Linux) .....	164
アップグレード手順 .....	165
互換性パッケージのインストール .....	166
アップグレード後の作業 .....	167
<b>第 8 章 High Availability Session Store .....</b>	<b>169</b>
HADB のアップグレードの概要 .....	170
Java ES Release 4 の HADB について .....	170
HADB のアップグレードのロードマップ .....	170
HADB のデータ .....	171
互換性の問題 .....	171
HADB の依存関係 .....	171
Java ES Release 3 からの HADB のアップグレード .....	172
はじめに .....	172

Release 3 の HADB のアップグレード	173
アップグレード前の作業	173
Release 3 の HADB のアップグレード	174
アップグレードの確認	175
アップグレード後の作業	176
アップグレードのロールバック	176
<b>第 9 章 Application Server</b>	<b>177</b>
Application Server のアップグレードの概要	178
Java ES Release 4 の Application Server について	178
Application Server のアップグレードロードマップ	178
Application Server のデータ	180
互換性の問題	180
Application Server の依存関係	180
Java ES Release 3 からの Application Server のアップグレード	181
はじめに	181
Release 3 の Application Server のアップグレード	182
アップグレード前の作業	183
Release 3 の Application Server のアップグレード (Solaris)	184
Release 3 の Application Server のアップグレード (Linux)	186
アップグレードの確認	188
アップグレード後の作業	188
アップグレードのロールバック (Solaris)	188
Java ES Release 2 からの Application Server のアップグレード	189
はじめに	189
Release 2 の Application Server のアップグレード	190
アップグレード前の作業	190
Release 2 の Application Server のアップグレード	191
アップグレードの確認	192
アップグレード後の作業	193
アップグレードのロールバック	193
複数インスタンス (クラスタ) のアップグレード	193
「あとで設定」インストール後の Application Server の設定	194
<b>第 10 章 Web Proxy Server</b>	<b>197</b>
Web Proxy Server のアップグレードの概要	197
Java ES Release 4 の Web Proxy Server について	198
Web Proxy Server のアップグレードロードマップ	198
Web Proxy Server のデータ	198
互換性の問題	199
Web Proxy Server の依存関係	199
Release 4 への Web Proxy Server のアップグレード	200



はじめに .....	200
Web Proxy Server のアップグレード .....	201
アップグレード前の作業 .....	201
Web Proxy Server のアップグレード .....	202
アップグレードの確認 .....	203
アップグレード後の作業 .....	203
アップグレードのロールバック .....	203
<b>第 11 章 Access Manager .....</b>	<b>205</b>
Access Manager のアップグレードの概要 .....	206
Java ES Release 4 の Access Manager について .....	206
Access Manager のアップグレードロードマップ .....	207
Access Manager のデータ .....	208
互換性の問題 .....	209
Access Manager の依存関係 .....	210
Java ES Release 3 からの Access Manager のアップグレード .....	211
はじめに .....	211
Release 3 の Access Manager の完全アップグレード .....	212
アップグレード前の作業 .....	212
Release 3 の Access Manager のアップグレード .....	215
Access Manager のアップグレードの確認 .....	222
アップグレード後の作業 .....	223
アップグレードのロールバック .....	223
複数インスタンスのアップグレード : Release 3 と Release 4 の共存 .....	224
Release 3 の Access Manager SDK のみのアップグレード .....	225
アップグレード前の作業 .....	225
Release 3 の Access Manager SDK のアップグレード .....	226
Access Manager SDK のアップグレードの確認 .....	226
アップグレードのロールバック .....	227
Java ES Release 2 からの Access Manager のアップグレード .....	227
アップグレード前の作業 .....	227
Access Manager が依存するコンポーネントをアップグレードする .....	227
ディレクトリスキーマをアップグレードする .....	228
Release 2 Access Manager のアップグレード .....	228
Release 2 Access Manager のアップグレード : Web Server Web コンテナ .....	228
Release 2 Access Manager のアップグレード : Application Server Web コンテナ .....	228
Access Manager のアップグレードの確認 .....	232
アップグレード後の作業 .....	232
アップグレードのロールバック .....	233
<b>第 12 章 Directory Preparation Tool .....</b>	<b>235</b>
Directory Preparation Tool のアップグレードの概要 .....	236

Java ES Release 4 Directory Preparation Tool について .....	236
Directory Preparation Tool のアップグレードロードマップ .....	236
Directory Preparation Tool のデータ .....	237
互換性の問題 .....	238
Directory Preparation Tool の依存関係 .....	238
Directory Preparation Tool の Java ES Release 3 からのアップグレード .....	238
はじめに .....	238
Release 3 Directory Preparation Tool のアップグレード .....	239
アップグレード前の作業 .....	239
Release 3 Directory Preparation Tool のアップグレード (Solaris) .....	240
Release 3 Directory Preparation Tool のアップグレード (Linux) .....	242
アップグレードの確認 .....	243
アップグレード後の作業 .....	243
アップグレードのロールバック (Solaris) .....	243
Directory Preparation Tool の Java ES Release 2 からのアップグレード .....	244
Release 2 のアップグレード手順 (Solaris) .....	245
Release 2 のアップグレード手順 (Linux) .....	246
<b>第 13 章 Messaging Server .....</b>	<b>247</b>
Messaging Server のアップグレードの概要 .....	248
Java ES Release 4 の Messaging Server について .....	248
Messaging Server のアップグレードロードマップ .....	248
Messaging Server のデータ .....	249
互換性の問題 .....	250
Messaging Server 依存関係 .....	250
Java ES Release 3 からの Messaging Server のアップグレード .....	251
はじめに .....	251
Release 3 の Messaging Server のアップグレード .....	252
アップグレード前の作業 .....	252
Release 3 の Messaging Server のアップグレード (Solaris) .....	254
Release 3 の Messaging Server のアップグレード (Linux) .....	256
アップグレードの確認 .....	258
アップグレード後の作業 .....	259
アップグレードのロールバック (Solaris) .....	259
複数インスタンスのアップグレード .....	260
Java ES Release 2 からの Messaging Server のアップグレード .....	261
Messaging Server が依存するコンポーネントをアップグレードする .....	261
Release 2 Messaging Server のアップグレード .....	262
Release 2 の Messaging Server のアップグレード (Solaris) .....	262
Release 2 Messaging Server のアップグレード (Linux) .....	262
アップグレードの確認 .....	263
アップグレード後の作業 .....	263

<b>第 14 章 Calendar Server</b> .....	<b>265</b>
Calendar Server のアップグレードの概要 .....	266
Java ES Release 4 の Calendar Server について .....	266
Calendar Server のアップグレードロードマップ .....	266
Calendar Server のデータ .....	267
互換性の問題 .....	267
Calendar Server 依存関係 .....	267
Java ES Release 3 からの Calendar Server のアップグレード .....	268
はじめに .....	268
Release 3 の Calendar Server のアップグレード .....	269
アップグレード前の作業 .....	269
Release 3 の Calendar Server のアップグレード (Solaris) .....	271
Release 3 の Calendar Server のアップグレード (Linux) .....	273
アップグレードの確認 .....	274
アップグレード後の作業 .....	274
アップグレードのロールバック (Solaris) .....	274
複数インスタンスのアップグレード .....	275
Java ES Release 2 からの Calendar Server のアップグレード .....	276
<b>第 15 章 Communications Express</b> .....	<b>277</b>
Communications Express のアップグレードの概要 .....	278
Java ES Release 4 の Communications Express について .....	278
Communications Express のアップグレードロードマップ .....	278
Communications Express のデータ .....	279
互換性の問題 .....	280
Communications Express 依存関係 .....	280
Java ES Release 3 からの Communications Express のアップグレード .....	281
はじめに .....	281
Release 3 の Communications Express のアップグレード .....	282
アップグレード前の作業 .....	283
Release 3 の Communications Express のアップグレード (Solaris) .....	284
Release 3 の Communications Express のアップグレード (Linux) .....	286
アップグレードの確認 .....	289
アップグレード後の作業 .....	289
アップグレードのロールバック (Solaris) .....	289
複数インスタンスのアップグレード .....	290
Java ES Release 2 からの Communications Express のアップグレード .....	290
Communications Express が依存するコンポーネントをアップグレードする .....	291
Release 2 Communications Express のアップグレード .....	292
Release 2 Communications Express のアップグレード : Web Server Web コンテナ .....	292
Release 2 Communications Express のアップグレード : Application Server Web コンテナ .....	292

<b>第 16 章 Instant Messaging</b> .....	<b>293</b>
Instant Messaging のアップグレードの概要 .....	294
Java ES Release 4 の Instant Messaging について .....	294
Instant Messaging のアップグレードロードマップ .....	294
Instant Messaging のデータ .....	295
互換性の問題 .....	295
Instant Messaging 依存関係 .....	296
Java ES Release 3 からの Instant Messaging のアップグレード .....	297
はじめに .....	297
Release 3 の Instant Messaging のアップグレード .....	298
アップグレード前の作業 .....	298
Release 3 の Instant Messaging のアップグレード (Solaris) .....	299
Release 3 の Instant Messaging のアップグレード (Linux) .....	301
アップグレードの確認 .....	303
アップグレード後の作業 .....	303
アップグレードのロールバック (Solaris) .....	304
複数インスタンスのアップグレード .....	304
Java ES Release 2 からの Instant Messaging のアップグレード .....	305
はじめに .....	305
Release 2 の Instant Messaging のアップグレード .....	306
アップグレード前の作業 .....	306
Release 2 の Instant Messaging のアップグレード (Solaris) .....	307
Release 2 の Instant Messaging のアップグレード (Linux) .....	309
アップグレードの確認 .....	310
アップグレード後の作業 .....	311
アップグレードのロールバック .....	311
複数インスタンスのアップグレード .....	311
<b>第 17 章 Portal Server</b> .....	<b>313</b>
Portal Server のアップグレードの概要 .....	314
Java ES Release 4 の Portal Server について .....	314
Portal Server のアップグレードロードマップ .....	314
Portal Server のデータ .....	315
互換性の問題 .....	316
Portal Server 依存関係 .....	316
Java ES Release 3 からの Portal Server のアップグレード .....	317
はじめに .....	317
Release 3 の Portal Server のアップグレード .....	318
アップグレード前の作業 .....	318
Release 3 の Portal Server のアップグレード (Solaris) .....	320
Release 3 の Portal Server のアップグレード (Linux) .....	322
アップグレードの確認 .....	325
アップグレード後の作業 .....	325

アップグレードのロールバック (Solaris) .....	325
複数インスタンスのアップグレード .....	327
Java ES Release 2 からの Portal Server のアップグレード .....	327
はじめに .....	328
Release 2 の Portal Server のアップグレード .....	328
アップグレード前の作業 .....	329
Release 2 Portal Server のアップグレード (Solaris) .....	331
Release 2 Portal Server のアップグレード (Linux) .....	337
アップグレードの確認 .....	342
アップグレード後の作業 .....	342
アップグレードのロールバック .....	342
複数インスタンスのアップグレード .....	342
<b>第 18 章 Portal Server Secure Remote Access .....</b>	<b>343</b>
Portal Server Secure Remote Access のアップグレードの概要 .....	344
Java ES Release 4 の Portal Server Secure Remote Access について .....	344
Portal Server Secure Remote Access のアップグレードロードマップ .....	344
Portal Server Secure Remote Access のデータ .....	345
互換性の問題 .....	346
Portal Server Secure Remote Access 依存関係 .....	346
Java ES Release 3 からの Portal Server Secure Remote Access のアップグレード .....	347
はじめに .....	347
Release 3 の Portal Server Secure Remote Access のアップグレード .....	348
アップグレード前の作業 .....	348
Release 3 の Portal Server Secure Remote Access のアップグレード (Solaris) .....	350
Release 3 の Portal Server Secure Remote Access のアップグレード (Linux) .....	351
アップグレードの確認 .....	354
アップグレード後の作業 .....	354
アップグレードのロールバック (Solaris) .....	354
複数インスタンスのアップグレード .....	355
Java ES Release 2 からの Portal Server Secure Remote Access のアップグレード .....	356
はじめに .....	356
Release 2 Portal Server Secure Remote Access のアップグレード .....	357
アップグレード前の作業 .....	357
Release 2 Portal Server Secure Remote Access のアップグレード (Solaris) .....	359
Release 2 Portal Server Secure Remote Access のアップグレード (Linux) .....	362
アップグレードの確認 .....	365
アップグレード後の作業 .....	366
アップグレードのロールバック .....	366
複数インスタンスのアップグレード .....	366
<b>第 19 章 Delegated Administrator .....</b>	<b>367</b>

Delegated Administrator のアップグレードの概要 .....	368
Java ES Release 4 の Delegated Administrator について .....	368
Delegated Administrator のアップグレードロードマップ .....	368
Delegated Administrator のデータ .....	369
互換性の問題 .....	370
Delegated Administrator 依存関係 .....	370
Java ES Release 3 からの Delegated Administrator のアップグレード .....	371
はじめに .....	371
Release 3 の Delegated Administrator のアップグレード .....	372
アップグレード前の作業 .....	372
Release 3 の Delegated Administrator のアップグレード (Solaris) .....	374
Release 3 の Delegated Administrator のアップグレード (Linux) .....	376
アップグレードの確認 .....	378
アップグレード後の作業 .....	379
アップグレードのロールバック (Solaris) .....	379
Java ES Release 2 からの Delegated Administrator のアップグレード .....	379
Delegated Administrator が依存するコンポーネントをアップグレードする .....	379
Release 2 Delegated Administrator のアップグレード .....	380
Release 2 Delegated Administrator のアップグレード : Web Server Web コンテナ .....	380
Release 2 Delegated Administrator のアップグレード : Application Server Web コンテナ .....	381
<b>付録 A Java Enterprise System リリースの内容 .....</b>	<b>383</b>
Java ES 2003Q4 (Release 1) .....	383
Release 1 のインストーラで選択可能なコンポーネント .....	383
Release 1 の共有コンポーネント .....	384
Java ES 2004Q2 (Release 2) .....	385
Release 2 のインストーラで選択可能なコンポーネント .....	385
Release 2 の共有コンポーネント .....	386
Java ES 2005Q1 (Release 3) .....	388
Release 3 のインストーラで選択可能なコンポーネント .....	388
Release 3 の共有コンポーネント .....	390
Java ES 2005Q4 (Release 4) .....	392
Release 4 のインストーラで選択可能なコンポーネント .....	392
Release 4 の共有コンポーネント .....	394
<b>索引 .....</b>	<b>397</b>

# 表目次

表 1	表記上の規則	20
表 2	記号の表記規則	20
表 3	シェルプロンプト	21
表 4	Java Enterprise System ドキュメント	22
表 1-1	Java ES Release 4 の製品コンポーネント	28
表 1-2	Java ES Release 4 の共有コンポーネント	29
表 1-3	アップグレードプロセスのフェーズ	35
表 1-4	Java ES 2005Q4 (Release 4) へのアップグレードパス	37
表 1-5	選択的なアップグレードとすべてのアップグレードの比較	40
表 1-6	Java ES Release 4 製品コンポーネントの共有コンポーネントに対する依存関係	42
表 1-7	Java ES 製品コンポーネントの依存関係	46
表 2-1	共有コンポーネントを Java ES Release 3 からアップグレードするためのアップグレードテクノロジー	60
表 2-2	共有コンポーネントを Java ES Release 2 からアップグレードするためのアップグレードテクノロジー	62
表 2-3	Solaris プラットフォームの共有コンポーネントアップグレード用のパッケージバージョン	73
表 2-4	Linux プラットフォームの共有コンポーネントアップグレード用のパッケージ	77
表 3-1	Java ES Release 4 Sun Cluster 3.1 8/05 (2005Q4) ソフトウェアへのアップグレードパス	98
表 3-2	Sun Cluster のデータの使用方法	99
表 3-3	Sun Cluster のバージョン確認の出力	101
表 4-1	Java ES Release 4: Sun Java System Directory Server 5.2 2005Q4 および Sun Java System Administration Server 5.2 2005Q4 へのアップグレードパス	106
表 4-2	Directory Server、Administration Server、および Directory Proxy Server のデータの使用方法	108
表 4-3	Directory Server のバージョン確認の出力	111
表 4-4	Solaris で Directory Server および Administration Server をアップグレードするためのバッチ	113

表 4-5	Linux で Directory Server および Administration Server をアップグレードするためのパッチ	116
表 5-1	Java ES Release 4: Sun Java System Directory Proxy Server 5.2 2005Q4 へのアップグレードパス	128
表 5-2	Directory Proxy Server のデータの使用方法	129
表 5-3	Directory Proxy Server のバージョン確認の出力	132
表 5-4	Solaris で Directory Proxy Server をアップグレードするためのパッチ	134
表 5-5	Linux で Directory Proxy Server をアップグレードするためのパッチ	135
表 6-1	Java ES Release 4: Sun Java System Web Server 6.1 SP5 2005Q4 へのアップグレードパス	142
表 6-2	Web Server のデータの使用方法	143
表 6-3	Web Server のバージョン確認の出力	145
表 6-4	Solaris で Web Server をアップグレードするためのパッチ	146
表 6-5	Linux で Web Server をアップグレードするためのパッチ	147
表 7-1	Java ES Release 4 Message Queue 3.6 SP3 2005Q4 へのアップグレードパス	153
表 7-2	Message Queue のデータの使用方法	154
表 7-3	Message Queue のバージョン確認の出力	159
表 7-4	Message Queue をアップグレードするためのパッチ	160
表 8-1	Java ES Release 4: HADB 4.4.2 (2005Q4) へのアップグレードパス	170
表 8-2	HADB のデータの使用方法	171
表 8-3	HADB のバージョン確認の出力	173
表 8-4	Solaris プラットフォーム用 HADB アップグレードパッケージのバージョン	174
表 9-1	Java ES Release 4: Sun Java System Application ServerEnterprise Edition 8.1 2005Q4 へのアップグレードパス	178
表 9-2	Application Server のデータの使用方法	180
表 9-3	Application Server のバージョン確認の出力	183
表 9-4	Solaris で Application Server をアップグレードするためのパッチ	184
表 9-5	Linux で Application Server をアップグレードするためのパッチ	186
表 10-1	Java ES Release 4: Sun Java System 4: Web Proxy Server 4.0.1 2005Q4 へのアップグレードパス	198
表 10-2	Web Proxy Server のデータの使用方法	199
表 10-3	Web Proxy Server のバージョン確認の出力	201
表 11-1	Java ES Release 4: Sun Java System Access Manager 7 2005Q4 へのアップグレードパス	207
表 11-2	Access Manager のデータの使用方法	208
表 11-3	Access Manager のバージョン確認の出力	213
表 11-4	Access Manager 設定パラメータ : ampre70upgrade	216
表 11-5	Access Manager Mobile Access ソフトウェアをアップグレードするためのパッチ	218
表 11-6	Access Manager 設定パラメータ	220



表 11-7	Access Manager 設定パラメータ : amupgrade .....	221
表 11-8	Access Manager 設定パラメータ .....	230
表 12-1	Java ES Release 4: Sun Java System Directory Preparation Tool 6.3 2005Q4 への アップグレードパス .....	236
表 12-2	Directory Preparation Tool のデータの使用方法 .....	237
表 12-3	Solaris で Directory Preparation Tool をアップグレードするためのパッチ .....	241
表 12-4	Linux で Directory Preparation Tool をアップグレードするためのパッチ .....	242
表 12-5	Solaris で Directory Preparation Tool をアップグレードするための開始パッチ .....	245
表 13-1	Java ES Release 4: Sun Java System Messaging Server 6.2 2005Q4 への アップグレードパス .....	248
表 13-2	Messaging Server のデータの使用方法 .....	249
表 13-3	Messaging Server のバージョン確認の出力 .....	253
表 13-4	Solaris で Messaging Server をアップグレードするためのパッチ .....	254
表 13-5	Linux で Messaging Server をアップグレードするためのパッチ .....	257
表 14-1	Java ES Release 4: Sun Java System Calendar Server 6.2 2005Q4 への アップグレードパス .....	266
表 14-2	Calendar Server のデータの使用方法 .....	267
表 14-3	Calendar Server のバージョン確認の出力 .....	270
表 14-4	Solaris で Calendar Server をアップグレードするためのパッチ .....	272
表 14-5	Linux で Calendar Server をアップグレードするためのパッチ .....	273
表 15-1	Java ES Release 4: Sun Java System Communications Express 6.2 2005Q4 への アップグレードパス .....	278
表 15-2	Communications Express のデータの使用方法 .....	279
表 15-3	Communications Express のバージョン確認の出力 .....	283
表 15-4	Solaris で Communications Express をアップグレードするためのパッチ .....	285
表 15-5	Linux で Communications Express をアップグレードするためのパッチ .....	287
表 16-1	Java ES Release 4: Sun Java System Instant Messaging 7.0.1 2005Q4 への アップグレードパス .....	294
表 16-2	Instant Messaging のデータの使用方法 .....	295
表 16-3	Instant Messaging のバージョン確認の出力 .....	298
表 16-4	Solaris で Instant Messaging をアップグレードするためのパッチ .....	300
表 16-5	Linux で Instant Messaging をアップグレードするためのパッチ .....	302
表 17-1	Java ES Release 4: Sun Java System Portal Server 6.3.1 2005Q4 への アップグレードパス .....	314
表 17-2	Portal Server のデータの使用方法 .....	315
表 17-3	Portal Server のバージョン確認の出力 .....	319
表 17-4	Solaris で Portal Server をアップグレードするためのパッチ .....	321
表 17-5	Linux で Portal Server をアップグレードするためのパッチ .....	323
表 17-6	Solaris で Portal Server を Release 4 にアップグレードするためのパッチ .....	331

表 17-7	Linux で Portal Server を Release 4 にアップグレードするためのパッチ .....	337
表 18-1	Java ES Release 4: Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4 へのアップグレードパス .....	344
表 18-2	Portal Server Secure Remote Access のデータの使用方法 .....	345
表 18-3	Portal Server Secure Remote Access のバージョン確認の出力 .....	349
表 18-4	Solaris で Portal Server Secure Remote Access をアップグレードするためのパッチ ..	350
表 18-5	Linux で Portal Server Secure Remote Access をアップグレードするためのパッチ ..	352
表 18-6	Solaris で Portal Server Secure Remote Access を Release 3 にアップグレード するためのパッチ .....	359
表 18-7	Linux で Portal Server Secure Remote Access を Release 3 にアップグレード するためのパッチ 362 .....	
表 19-1	Java ES Release 4: Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 へのアップグレードパス .....	368
表 19-2	Delegated Administrator のデータの使用方法 .....	369
表 19-3	Delegated Administrator のバージョン確認の出力 .....	373
表 19-4	Solaris で Delegated Administrator をアップグレードするためのパッチ .....	375
表 19-5	Linux で Delegated Administrator をアップグレードするためのパッチ .....	377

# はじめに

『Java Enterprise System アップグレードガイド』には、Sun Solaris™ オペレーティングシステム (Solaris OS) または Linux オペレーティングシステム環境での Sun Java™ Enterprise System (Java ES) ソフトウェアのアップグレードに必要な情報が含まれています。本書は、Java ES 2004Q2 (Release 2) および Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードを対象とします。

この章で説明する項目は次のとおりです。

- 19 ページの「対象読者」
- 20 ページの「本書で使用する表記規則」
- 22 ページの「関連マニュアル」
- 24 ページの「オンライン上の Sun リソースへのアクセス」
- 24 ページの「Sun テクニカルサポートへの問い合わせ」
- 24 ページの「関連するサードパーティーの Web サイト」
- 25 ページの「ご意見、ご要望の送付先」

## 対象読者

本書は、システム管理者、および Java ES ソフトウェアのアップグレードが必要なソフトウェア技術者を対象としています。

本書は、次の事項に習熟している方を対象に記述されています。

- エンタープライズレベルのソフトウェア製品のインストール
- ご使用の環境に現在配備されている Java ES コンポーネント
- サポートする Java ES プラットフォーム上のシステム管理およびネットワーク
- クラスタリングモデル (クラスタリングソフトウェアをインストールする場合)

# 本書で使用する表記規則

次の表は、本書で使用する表記規則について説明します。

## 表記上の規則

次の表は、本書で使用する表記上の規則について説明したものです。

表 1 表記上の規則

表記	意味	例
AaBbCc123 (モノスペース)	API および言語の要素、HTML タグ、Web サイトの URL、コマンド名、ファイル名、ディレクトリパス名、画面出力の表示、サンプルコード。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 % You have mail.
<b>AaBbCc123</b> (太字のモノスペース)	ユーザーが入力するテキストを強調し、コンピュータからの画面出力と区別します。	% <b>su</b> Password:
<i>AaBbCc123</i> (イタリック)	実際の名前または値によって置き換えられるコマンドまたはパス名の可変部分。	これらを <i>class</i> オプションと呼びます。 ファイルは <i>install-dir/bin</i> ディレクトリにあります。

## 記号

次の表は、このマニュアルで使用される記号の表記規則を示しています。

表 2 記号の表記規則

記号	説明	例	意味
[ ]	省略できるコマンドオプションを含みます。	ls [-l]	-l オプションは省略可能です。
{   }	必須コマンドオプションの選択肢を含みます。	-d {y n}	-d オプションは、引数 y または n のいずれかを使用する必要があります。
-	同時に押すキーを連結します。	Control-A	Ctrl キーと A キーを同時に押します。

表 2 記号の表記規則 ( 続き )

記号	説明	例	意味
+	連続して押すキーを連結します。	Ctrl+A+N	Ctrl キーを押し、放してから、以後のキーを続けて押します。
>	グラフィカルユーザーインタフェースで選択するメニュー項目を示します。	「ファイル」>「新規」>「テンプレート」	「ファイル」メニューから「新規」を選択します。「新規」サブメニューから、「テンプレート」を選択します。

## シェルプロンプト

次の表は、このマニュアルで使用されるシェルプロンプトを示しています。

表 3 シェルプロンプト

シェル	プロンプト
UNIX または Linux の C シェル	<i>machine-name%</i>
UNIX または Linux の C シェルのスーパーユーザー	<i>machine-name#</i>
UNIX または Linux の Bourne シェルおよび Korn シェル	\$
UNIX または Linux の Bourne シェルおよび Korn シェルのスーパーユーザー	#
Windows のコマンド行	C:¥

## 関連マニュアル

<http://docs.sun.com><sup>SM</sup> Web サイトでは、Sun テクニカルマニュアルにオンラインでアクセスできます。特定のマニュアルのタイトルまたは件名のアーカイブを参照し、検索できます。

### このドキュメントセット内のマニュアル

Java ES マニュアルは、PDF (Portable Document Format) 形式および HTML (Hypertext Markup Language) 形式のオンラインファイルとして用意されています。どちらの形式のファイルも、障害を持つユーザーにも参照可能です。Sun™ のマニュアルには、次の Web サイトからアクセスできます。

<http://docs.sun.com/app/docs?l=ja>

Java ES マニュアルには、システム全体に関する情報とそのコンポーネントに関する情報が含まれます。このマニュアルには次の場所からアクセスできます。

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

次の表は、Java ES ドキュメントセットに含まれる、システムレベルのマニュアルを示しています。左の列は各ドキュメントの名称とアクセスできる URL、右の列はドキュメントの簡単な説明を示しています。

表 4 Java Enterprise System ドキュメント

マニュアル名	内容
『Sun Java Enterprise System リリースノート』 <a href="http://docs.sun.com/doc/819-3421?l=ja">http://docs.sun.com/doc/819-3421?l=ja</a>	既知の問題など、Java Enterprise System に関する最新の情報が記載されています。これ以外に、コンポーネントごとにリリースノートがあります。
『Sun Java Enterprise System ロードマップ』 <a href="http://docs.sun.com/doc/819-3414?l=ja">http://docs.sun.com/doc/819-3414?l=ja</a>	システムとして、および個々のコンポーネントについての Java Enterprise System に関するマニュアルについて説明しています。
『Sun Java Enterprise System 技術の概要』 <a href="http://docs.sun.com/doc/819-3583?l=ja">http://docs.sun.com/doc/819-3583?l=ja</a>	Java Enterprise System の技術的および概念的な基礎について説明します。コンポーネント、アーキテクチャ、プロセス、および機能について説明しています。

表 4 Java Enterprise System ドキュメント ( 続き )

マニュアル名	内容
『Sun Java Enterprise System 配備計画ガイド』 <a href="http://docs.sun.com/doc/819-3449?l=ja">http://docs.sun.com/doc/819-3449?l=ja</a>	Java Enterprise System に基づく企業配備ソリューションの計画および設計に関する情報を提供します。配備の計画および設計に関する基本的概念と原則を示し、ソリューションのライフサイクルについて説明し、Java Enterprise System に基づくソリューションを計画する際に使用する高度な例と戦略を提供します。
『Sun Java Enterprise System インストール計画ガイド』 <a href="http://docs.sun.com/doc/819-3940?l=ja">http://docs.sun.com/doc/819-3940?l=ja</a>	Java Enterprise System 配備のハードウェア、オペレーティングシステム、およびネットワークの見地から実装の仕様を開発する際に役立ちます。インストール計画および設定計画で検討すべきコンポーネントの依存関係などの問題について説明します。
『Sun Java Enterprise System インストールガイド (UNIX 版)』 <a href="http://docs.sun.com/doc/819-3313?l=ja">http://docs.sun.com/doc/819-3313?l=ja</a>	Solaris™ オペレーティングシステムまたは Linux オペレーティングシステムでの Java Enterprise System のインストール手順について説明します。インストール後にコンポーネントを設定する方法、および設定したコンポーネントが正常に機能するかどうかを確認する方法についても説明します。
『Sun Java Enterprise System インストールリファレンス』 <a href="http://docs.sun.com/doc/819-3947?l=ja">http://docs.sun.com/doc/819-3947?l=ja</a>	設定パラメータに関する追加情報、設定計画に使用できるワークシート、およびデフォルトのディレクトリやポート番号などの参照情報の一覧を提供します。
『Sun Java Enterprise System 配備実例集：評価のシナリオ』 <a href="http://docs.sun.com/doc/819-3442?l=ja">http://docs.sun.com/doc/819-3442?l=ja</a>	任意のシステムに Java Enterprise System をインストールし、共有され、ネットワーク化されたコアとなるサービスを確立し、確立したサービスにアクセス可能なユーザーアカウントを設定する方法について説明します。
『Sun Java Enterprise System アップグレードガイド』 <a href="http://docs.sun.com/doc/819-3456?l=ja">http://docs.sun.com/doc/819-3456?l=ja</a>	Solaris™ オペレーティングシステムまたは Linux オペレーティングシステムでの Java Enterprise System のアップグレード手順について説明します。
『Sun Java Enterprise System 用語集』 <a href="http://docs.sun.com/doc/819-4629?l=ja">http://docs.sun.com/doc/819-4629?l=ja</a>	Java Enterprise System のドキュメントで使用される用語について説明します。

## オンライン上の Sun リソースへのアクセス

製品のダウンロード、プロフェショナルサービス、パッチとサポート、および開発者向け追加情報については、次の Web サイトにアクセスしてください。

- ダウンロードセンター  
<http://www.sun.com/software/download/>
- クライアントソリューション  
<http://www.sun.com/service/sunjavasystem/sjssservicesuite.html>
- Sun Enterprise サービス、Solaris パッチ、およびサポート  
<http://sunsolve.sun.com/>
- 開発者向け情報  
<http://jp.sun.com/developers/>

次の場所には、Java ES およびそのコンポーネントに関する情報が用意されています。

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

## Sun テクニカルサポートへの問い合わせ

製品のドキュメントで解決できない、本製品に関する技術的な質問の問い合わせ先については、<http://jp.sun.com/service/support/> を参照してください。

## 関連するサードパーティーの Web サイト

Sun は、このマニュアルに記載されているサードパーティー Web サイトの利用について責任を負いません。Sun は、このようなサイトまたはリソースで得られるあらゆる内容、広告、製品、およびその他の資料を保証するものではなく、責任または義務を負いません。Sun は、このようなサイトまたはリソースで得られるあらゆるコンテンツ、製品、またはサービスによって生じる、または生じたと主張される、または使用に関連して生じる、または信頼することによって生じる、いかなる損害または損失についても責任または義務を負いません。



## ご意見、ご要望の送付先

Sun ではマニュアルの品質向上のため、お客様のご意見、ご要望をお受けしております。

コメントをお送りになる場合は、<http://docs.sun.com/app/docs?l=ja> にアクセスして「コメントの送信」をクリックしてください。オンラインフォームで、ドキュメントのタイトルと Part No. を入力します。Part No. は、マニュアルのタイトルページまたは最上部に記載されている 7 桁または 9 桁の番号です。

ご意見、ご要望の送付先

# アップグレードの計画

この章では、Sun Java™ Enterprise System (Java ES) ソフトウェアの Java ES 2005Q4 (Release 4) へのアップグレード計画に使用される情報について説明します。この章で説明する内容は、次のとおりです。

- 27 ページの「Java ES 2005Q4 (Release 4) コンポーネント」
- 30 ページの「Java ES のアップグレードについて」
- 34 ページの「アップグレードの計画」
- 41 ページの「Java ES コンポーネントの依存関係」
- 51 ページの「手順に関する一般的なガイドライン」

## Java ES 2005Q4 (Release 4) コンポーネント

この節では、Java ES ソフトウェアのアップグレード計画に対する導入として、Java ES Release 4 に含まれるコンポーネントについて確認します。アップグレードシナリオによっては、これらのコンポーネントの 1 つ以上を Release 4 バージョンにアップグレードする必要がある可能性があります。

『Sun Java Enterprise System 技術の概要』(<http://docs.sun.com/doc/819-3583?l=ja>) で説明されるように、Java ES コンポーネントは数種類のタイプに分類されます。システムサービスコンポーネントはメインの Java ES インフラストラクチャーサービスを提供し、サービス品質コンポーネントはこれらのシステムサービスを機能拡張します。ここでは、これら 2 つのタイプの Java ES コンポーネントを合わせて製品コンポーネントと呼びます。これらのコンポーネントは、Java ES インストーラで選択できます。

各製品コンポーネントは、1 つ以上の Java ES 共有コンポーネントというローカルで共有されるライブラリに依存します。共有コンポーネントは、インストールされる製品コンポーネントの内容に基づいて、製品コンポーネントのインストール時に Java ES インストーラによって自動的にインストールされます。

## Release 4 の製品コンポーネント

次の表に Java ES Release 4 の製品コンポーネントをアルファベット順で示します。サービス品質コンポーネントについては、それらが提供するサービス拡張のタイプを示しています。

表 1-1 Java ES Release 4 の製品コンポーネント

製品コンポーネント	バージョン	タイプ	略称
Access Manager	7.0	システムサービスコンポーネント	AM
Administration Server	5.2	サービス品質: 管理コンポーネント	ADS
Application Server	8.1	システムサービスコンポーネント	AS
Calendar Server	6.2	システムサービスコンポーネント	CS
Communications Express	6.2	サービス品質: アクセスコンポーネント	CX
Delegated Administrator	6.3	サービス品質: 管理コンポーネント	DA
Directory Preparation Tool	6.3	サービス品質: 管理コンポーネント	DPT
Directory Proxy Server	5.2	サービス品質: アクセスコンポーネント	DPS
Directory Server	5.2	システムサービスコンポーネント	DS
High Availability Session Store	4.4.2	サービス品質: 可用性コンポーネント	HADB
Instant Messaging	7.0.1	システムサービスコンポーネント	IM
Message Queue	3.6 SP3	システムサービスコンポーネント	MQ
Messaging Server	6.2	システムサービスコンポーネント	MS
Portal Server	6.3	システムサービスコンポーネント	PS
Portal Server Secure Remote Access	6.3	サービス品質: アクセスコンポーネント	PSRA
Service Registry	3.0	システムサービスコンポーネント	SR
Sun Cluster	3.1 8/05	サービス品質: 可用性コンポーネント	SC
Web Proxy Server	4.0.1	サービス品質: アクセスコンポーネント	WPS
Web Server	6.1 SP%	システムサービスコンポーネント	WS

## Release 4 の共有コンポーネント

1 台のコンピュータにインストールされている製品コンポーネントが依存する Java ES 共有コンポーネントは、Java ES インストーラで選択または選択解除することはできません。Java ES インストーラは、Java ES 製品コンポーネントのインストール時に、インストールされる製品コンポーネントが必要とする共有コンポーネントを自動的にインストールします。

次の表に Java ES Release 4 の共有コンポーネントを示します。

表 1-2 Java ES Release 4 の共有コンポーネント

共有コンポーネント	バージョン	略称
Apache Commons Logging	1.0.3	ACL
Jakarta ANT Java/XML ベースの構築ツール	1.6.2	ANT
Berkeley Database	4.2.52	BDB
共通エージェントコンテナ	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
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

表 1-2 Java ES Release 4 の共有コンポーネント ( 続き )

共有コンポーネント	バージョン	略称
Mobile Access コア	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

## Java ES のアップグレードについて

Java ES ソフトウェアを Release 4 にアップグレードする場合、通常は Java ES インストーラやその他のシステムユーティリティを使用しません。アップグレードは、コンポーネントに固有のアップグレード方法によって、コンポーネントごと、コンピュータごとに実行します。

コンポーネントのアップグレードは、以前のバージョンのコンポーネントと互換性がない場合もあるメジャーアップグレードから、単にバグを修正しただけで、完全互換性のあるアップグレードまで、多岐に渡ります。Java ES コンポーネント間の依存関係により、アップグレード時に別のコンポーネントのアップグレードが必要になる場合もあります。

## 製品コンポーネントのアップグレード

Java ES 製品コンポーネントのアップグレードには、Java ES 製品コンポーネントの初期インストールおよび初期設定と同様の次の 2 種類の基本的な操作があります。

- アップグレードされたソフトウェアのインストール:** 新しいソフトウェアは、既存のソフトウェアを拡張または修復、あるいは置換します。通常、新しいソフトウェアは、既存のソフトウェアパッケージへのパッチの適用、既存のパッケージの置換、新しいパッケージのインストール、または Java ES インストーラを使用したコンポーネントの完全な再インストールによって取得されます。

- **再設定** : 再設定は、アップグレードされたソフトウェアをサポートするのに必要な設定データ、ユーザーデータ、または動的なアプリケーションデータに対する変更をすべて網羅します。データの変更とは、データの追加、データ形式の変更 (プロパティファイル内でもデータベーススキーマ内でも)、またはデータの格納場所の変更を意味します。再設定には、ユーザーによる明示的な手順の実行が必要な場合があります。一方、自動的に実行されることもあります。

この『アップグレードガイド』では、Java ES 製品コンポーネントごとに、コンポーネントアップグレードに関する、これらの 2 つの側面について説明します。

その他、『アップグレードガイド』では製品コンポーネントのアップグレードに関する次の重要な側面についても説明します。

- アップグレードに影響を与える依存関係
- コンポーネントをアップグレードする前に実行する必要がある操作
- アップグレードの成功を確認するために実行する操作
- アップグレードのロールバックが必要な場合に実行する操作

## 共有コンポーネントのアップグレード

多くの場合、Java ES 共有コンポーネントに依存する製品コンポーネントのアップグレードでは、共有コンポーネントのアップグレードが必要となります。

通常、共有コンポーネントのアップグレードは製品コンポーネントのアップグレードよりも簡単です。一般的には、共有コンポーネントのアップグレードは、既存のパッケージに対するパッチの適用、または既存のパッケージの置換によって行われます。製品コンポーネントのアップグレードと異なり、通常は再設定が不要で、アップグレードの前や後に実行が必要な手順もありません。

共有コンポーネントは個別にアップグレードできますが、Java ES Release 4 では複数の共有コンポーネントを 1 つの操作でまとめてアップグレードできます。詳細については、[第 2 章「Java ES 共有コンポーネントのアップグレード」](#)を参照してください。

## アップグレードテクノロジー

この『アップグレードガイド』で説明する製品コンポーネントと共有コンポーネントの両方のアップグレードには、現在インストールされているソフトウェアパッケージの修正や置換が含まれます。また、新しいパッケージのインストールが含まれる場合もあります。Solaris プラットフォームと Linux プラットフォームでは、インストールされているソフトウェアパッケージの管理と、パッケージレジストリによる変更の追跡において、同様のテクノロジーが採用されています。

- **Solaris プラットフォーム** : Java ES パッケージは、Java ES ソフトウェアディストリビューション内のパッケージを使用して、Solaris の `pkgadd` コマンドと `pkgrm` コマンドによってインストールまたは削除できます。パッケージをインストールすると、`patchadd` コマンドと `patchrm` コマンドによって適用または削除されるパッチを使用して、パッケージの内容を修正できます。Solaris パッケージに対するパッチは、次の SunSolve Web サイトで配布されています。  
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

Solaris パッチでは、1 つまたは複数のパッケージを処理できます。patchadd コマンドは、patchrm コマンドを使用したパッチの削除を容易に行えるように、パッチされるパッケージのバックアップを保存します。パッチは、パッチ番号とリビジョン番号で構成されるパッチ ID によって識別されます。リビジョン番号は、パッチが修正されるたびに増加します。

また、Solaris パッチはパッチクラスタにまとめることもできます。パッチクラスタを使用すると、クラスタ内のすべてのパッチをまとめてダウンロードし、適用できます。パッチクラスタは、Java ES 共有コンポーネントのアップグレード用に提供されています (第 2 章「Java ES 共有コンポーネントのアップグレード」を参照)。

- **Linux プラットフォーム** : Java ES RPM (Red Hat Package Manager) パッケージは、Java ES ソフトウェアディストリビューション内のパッケージを使用して、`rpm` コマンドによってインストールまたは更新できます。ただし、パッケージをインストールすると、パッチを使用してパッケージの内容を修正することはできません。RPM パッケージを更新するには、現在のパッケージを新しいパッケージで置き換えるための `rpm -U` コマンドオプションを使用します。

便宜上、多くの RPM パッケージアップグレードは Java ES ソフトウェアディストリビューションだけではなく、次の SunSolve Web サイトでも配布されています。  
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

SunSolve による配布では、RPM パッケージはパッチとしてラッパーすることができ、Solaris パッチと類似したパッチ ID とリビジョン番号が割り当てられます。これらの Linux パッチには 1 つまたは複数の RPM パッケージが含まれることがあり、それぞれが一意的な RPM 名、RPM 番号、およびリビジョン番号によって識別されます。リビジョン番号は、RPM パッケージが修正されるたびに増加します。



## オペレーティングシステムの問題

オペレーティングシステムに関する多数の問題が、次に示すように Java ES ソフトウェアのアップグレードに影響を与えます。

### 必須のオペレーティングシステムパッチ

場合によっては、Java ES 製品コンポーネントのアップグレードを成功させるために、最初にオペレーティングシステムにパッチまたは特定の修正を適用する必要があります。通常、各ケースに応じた特定のオペレーティングシステムパッチを適用するよりも、Java ES アップグレードの実行前にオペレーティングシステムを最新バージョンにすることを勧めます。

- Solaris プラットフォームのパッチは SunSolve Web サイトからパッチクラスタとして入手できます。パッチクラスタとは、まとめて適用できるオペレーティングシステムのパッチを集めたものです。Solaris 8、9、および 10 のオペレーティングシステムのパッチクラスタは次のサイトで入手できます。  
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>
- Linux プラットフォームの更新リリースは次のサイトで入手できます。  
<https://www.redhat.com/apps/download/>

### マイナーリリースのアップグレード

Java ES 共有コンポーネントの大多数には、Solaris リリース固有のパッケージが用意されています。リリース固有のパッケージは、ほかの Solaris プラットフォームでは正しく動作しない可能性があります。たとえば、Solaris 8 オペレーティングシステム用にリリースされているパッケージは、Solaris 9 または Solaris 10 オペレーティングシステムで正しく動作することを想定していません。

あるマイナーリリースから別のマイナーリリースにオペレーティングシステムをアップグレードすると、インストールされている各種 Java ES 共有コンポーネントが影響を受けます。共有コンポーネントにリリース固有のパッケージが含まれる場合、オペレーティングシステムのアップグレード後に、それらのパッケージもアップグレードして、新しくアップグレードされたオペレーティングシステムに対応させる必要があります。

### サポートされないプラットフォームへのアップグレード

Java ES 2004Q2 (Release 2) は、Solaris 8 および 9 のオペレーティングシステムと Red Hat Enterprise Linux (RHEL) 2.1 でサポートされています。また、オペレーティングシステムのプラットフォームを Java ES Release 2 でサポートされていない Solaris 10 または RHEL 3.0 にアップグレードする場合、Java ES Release 2 はアップグレードされたプラットフォームをサポートするリリースの Java ES (できれば Java ES Release 4) にアップグレードする必要があります。

一部の Java ES コンポーネントのアップグレードでは別の Java ES コンポーネントを実行していることが必要な場合もあるため、一般的な規則としては、Java ES を Release 2 からアップグレードする前にオペレーティングシステムのプラットフォームを Solaris 10 または RHEL 3.0 にアップグレードすることはできません (Java ES Release 2 は、これらのプラットフォームをサポートしていないため)。

その代わりに、プラットフォームごとに次の対応が必要です。

- **Linux プラットフォーム** : 最初に Java ES Release 2 から Release 4 にアップグレードし、次に RHEL 3.0 にアップグレードしてください。
- **Solaris プラットフォーム** : Java ES Release 2 をアンインストールし、オペレーティングシステムを Solaris 10 にアップグレードしてから、Java ES Release 4 をフレッシュインストールします。これらの操作は、すべての Java ES コンポーネントを新しく設定する必要があることを意味します。この場合、Java ES Release 4 コンポーネントの設定に使用する Java ES Release 2 のすべての設定ファイルとカスタマイズをバックアップすることをお勧めします。

## アップグレードの計画

配備された Java ES ソフトウェアシステムを Java ES Release 4 にアップグレードする方法は、配備アーキテクチャーの範囲と複雑さに加えて、アップグレードの目的と優先事項によって異なります。

たとえば、1 台のコンピュータで実行される単一の Java ES コンポーネントで構成される Java ES 配備アーキテクチャーで、以前のソフトウェアリリースのバグを修正することがアップグレードの目的である場合があります。一方、相互に依存関係を持つ多数の Java ES コンポーネントが複数のコンピュータに配備された構成の Java ES 配備アーキテクチャーで、最小のダウンタイムで必要最小限のコンポーネントをアップグレードすることによって新しい機能を実現することがアップグレードの目的である場合もあります。

これら 2 つの例は、複雑さの大きく異なるアップグレードシナリオであり、大幅に異なるアップグレード計画を必要とします。配備された Java ES ソフトウェアシステムすべてに対応できる計画はありません。

通常、Java ES コンポーネントの数と、配備アーキテクチャー内のコンピュータの数が多くなるほど、アップグレード計画は複雑になります。

## アップグレード計画とは

アップグレード計画は、アップグレードプロセスの段階ごとの手順を特定します。このプロセスには少なくとも、次の表に示すフェーズが含まれます。

表 1-3 アップグレードプロセスのフェーズ

アップグレードフェーズ	説明
準備	アップグレード計画を作成します。アップグレード計画では、アップグレードが必要な Java ES コンポーネントと、それらコンポーネントをシステム内の各コンピュータでアップグレードする順序を示します。また、アップグレード手順を本稼働環境で実行する前に、それらアップグレード手順をステージング環境でテストする方法を計画します。さらに、この手順では、現在のシステムをバックアップし、それらを現在の設定に復元するテストも行います。
実行	アップグレードに必要なパッケージ、パッチ、およびツールをすべて入手します。ステージング環境で、配備済み Java ES システムのアップグレードと再設定を実行します。これには、設定データとアプリケーションデータのバックアップ、システムソフトウェアのアップグレード、およびアップグレードされたシステムへのデータの移行または再設定が含まれます。
検証	アップグレードされたソフトウェアコンポーネントを起動し、引き続き検証テストを実行します。検証に失敗し、適切な時間内に問題を解決できないと、アップグレードをロールバックし、以前の状態にシステムを復元することが必要になる場合があります。
ロールバックと復元	必要に応じて、準備フェーズで指定された以前の状態にシステムを復元します。また、ロールバックの成功を検証するためのテストを実行します。

次に、アップグレード計画の作成に役立つ情報を示します。

## アップグレード計画の注意点

アップグレード計画は、配備アーキテクチャーの範囲と複雑さ以外の要素によっても異なります。これらの要素には、次の検討事項が含まれます。

- アップグレードパス
- 配備された Java ES コンポーネント間の依存関係
- 選択的なアップグレードを実行する可能性
- 複数インスタンスのアップグレード

これらの要素について、次の節以降で説明します。

### アップグレードパス

Java ES 2005Q4 (Release 4) には、以前のすべてのリリースの Java ES ソフトウェアからアップグレードできますが、動作保証されているのは Java ES 2005Q1 (Release 3) および Java ES 2004Q2 (Release 2) からのアップグレードだけです。この『アップグレードガイド』では、それ以前のリリースからのアップグレードについて説明していません。

[37 ページの表 1-4](#) に示すように、各アップグレードパスには異なるアップグレード戦略が含まれます。

Release 3 から Release 4 へのアップグレードパスと Release 2 から Release 4 へのアップグレードパスでは特性が異なることと、製品コンポーネントのアップグレード手順はアップグレードパスによって異なることから、この『アップグレードガイド』では、各製品コンポーネントのアップグレードについて説明する各章は、Release 3 から Release 4 へのアップグレードと Release 2 から Release 4 へのアップグレードに関する 2 つの節に分けられています。

表 1-4 Java ES 2005Q4 (Release 4) へのアップグレードパス

製品番号	Java ES のリリース	システムの特徴	アップグレード戦略
2005Q1	Release 3	Java ES Release 4 では、1 台のコンピュータ上での Release 3 コンポーネントと Release 4 コンポーネントの共存をサポートしています。これには、製品コンポーネントと共有コンポーネントの両方が含まれます。Release 3 コンポーネントと Release 4 コンポーネントの互換性はテスト済みで、既知の非互換性については『Sun Java Enterprise System リリースノート』( <a href="http://docs.sun.com/doc/819-3421?l=jja">http://docs.sun.com/doc/819-3421?l=jja</a> ) で説明されています。	Release 3 と Release 4 のコンポーネントが共存すると、1 台のコンピュータ上で、または複数のコンピュータで構成される配備アーキテクチャー内で、Release 3 コンポーネントの Release 4 への選択的なアップグレードを行う可能性が生じます。
2004Q2	Release 2	Java ES Release 4 では、1 台のコンピュータ上での Release 2 コンポーネントと Release 4 コンポーネントの共存をサポートしていません。これには、製品コンポーネントと共有コンポーネントの両方が含まれます。これらのリリースバージョン間には既知の非互換性があり、Release 2 と Release 4 のコンポーネント間の相互運用性は保証されていません。	1 台のコンピュータで Release 2 から Release 4 にコンポーネントをアップグレードする場合、Release 2 のすべてのコンポーネントを Release 4 にアップグレードする必要があります。ただし、コンポーネントの互換性がある場合、複数のコンピュータで構成される配備アーキテクチャー内でも別のコンピュータであれば Release 2 と Release 4 のコンポーネントを共存させることができます。

表 1-4 Java ES 2005Q4 (Release 4) へのアップグレードパス ( 続き )

製品番号	Java ES のリリース	システムの特性	アップグレード戦略
2003Q4 以前	Release 1 以前	Java ES Release 4 では、1 台のコンピュータ上での Release 1 以前のコンポーネントと Release 4 コンポーネントの共存をサポートしていません。これには、製品コンポーネントと共有コンポーネントの両方が含まれます。これらのリリースバージョン間には既知の非互換性があり、Release 1 以前と Release 4 のコンポーネント間の相互運用性は保証されていません。	<p>Java ES Release 4 では、Release 1 以前のリリースから Release 4 への直接のアップグレードは動作保証されていません。</p> <p>ただし、Release 3 の『Java Enterprise System アップグレードと移行』(<a href="http://docs.sun.com/doc/819-2235?l=ja">http://docs.sun.com/doc/819-2235?l=ja</a>) に説明されているように、最初に Java ES Release 3 にアップグレードすることで Release 1 からアップグレードできることもあります。</p> <p>一方、Release 2 または Release 3 から Release 4 へのアップグレードと同じ方法で Release 1 から Release 4 へのアップグレードを実行できることがあります。この『アップグレードガイド』では、該当するコンポーネントのアップグレードロードマップでその可能性について説明しています。</p>

注	製品コンポーネントによっては、Java ES の公式なリリースの間に暫定リリースが発行されていることがあります。その場合、暫定リリースからのアップグレードでは、以前の Java ES リリースと同じ手順を使用して実行する必要があります。たとえば、Release 2 と Release 3 の間に暫定リリースがある場合、Release 2 から Release 4 へのアップグレード手順を使用してコンポーネントをアップグレードします。
---	--

## アップグレードの依存関係

いずれの Java ES コンポーネントのアップグレード計画においても、主要な課題の 1 つとなるのは、指定されたコンポーネントについて別の Java ES コンポーネントに対する依存関係を理解し、コンポーネントのアップグレードをサポートするために依存先のコンポーネントもアップグレードする必要があるかどうかを判断することです。

この見地から、アップグレードの依存関係には次の 2 つのタイプがあります。

- **強いアップグレード依存関係**：強いアップグレード依存関係とは、アップグレードされたバージョンのコンポーネントが、依存先のコンポーネントについてもアップグレードされたバージョンを必要とする場合です。この要件は、新しい機能、新しいインタフェース、または依存するコンポーネントに必要なバグの修正に起因する場合があります。コンポーネントのアップグレードを成功させ、コンポーネントを使用できるようにするには、先に依存先のコンポーネントをアップグレードする必要があります。
- **弱いアップグレード依存関係**：弱いアップグレード依存関係とは、アップグレードされたバージョンのコンポーネントが、依存先のコンポーネントについてはアップグレードされたバージョンを必要としない場合です。コンポーネントのアップグレードを成功させ、コンポーネントを使用できるようにするために、依存先のコンポーネントをアップグレードする必要はありません。

Java ES コンポーネントのアップグレードでは、強いアップグレード依存関係の依存先となるすべてのコンポーネントをアップグレードする必要がありますが、弱いアップグレード依存関係の依存先となるコンポーネントをアップグレードする必要はありません。ただし、この一般的な規則は、1 台のコンピュータ上での Release 2 から Release 4 へのアップグレードには適用されません。

この一般的な規則は、相互に依存関係を持つ複数のコンポーネントがアップグレードに含まれる場合に適用されるとは限りません。このような場合、ほかの複数の Java ES コンポーネントのうち 1 つだけが該当のコンポーネントに対して強いアップグレード依存関係を持つ場合に、該当のコンポーネントをアップグレードする必要があります。

## 選択的なアップグレードとすべてのアップグレード

強いアップグレード依存関係と弱いアップグレード依存関係の違いによって、配備されたシステム内で Java ES コンポーネントを選択的にアップグレードする可能性が生じます。この可能性は、1 台のコンピュータ上で Release 3 から Release 4 にアップグレードする場合だけに適用されます (36 ページの「アップグレードパス」のアップグレードパスの特性を参照)。1 台のコンピュータでの Release 2 から Release 4 への選択的なアップグレードはサポートされていません。

- **選択的なアップグレード**：選択的なアップグレードの手順は、Release 4 にアップグレードする Java ES コンポーネントから始まります。そのコンポーネントの強いアップグレード依存関係を確認します。これには、製品コンポーネントと共有コンポーネントの両方に対する依存関係が含まれます。それらのコンポーネントもアップグレードする必要があります。このプロセスを強いアップグレード依存関係ごとに、アップグレードの必要があるコンポーネントがなくなるまで継続します。この作業によって、アップグレードが必要なすべての Java ES コンポーネントが特定されます。

選択的なアップグレードの手順は、配備アーキテクチャーとそれに含まれる強いアップグレード依存関係によって、単純な場合もあれば、かなり複雑になる場合もあります。

- **すべてのアップグレード**: その他の方法として、配備された Java ES コンポーネントをすべて Release 4 にアップグレードすることができます。この手順の複雑さも、配備アーキテクチャーによって変わります。場合によっては、システム全体を一度にアップグレードするのは業務上の理由から単純に不可能なことがあります。

アップグレードを実行するための 2 つの手順を、次の表で比較します。

表 1-5 選択的なアップグレードとすべてのアップグレードの比較

アップグレードの手順	利点	欠点
選択的なアップグレード	アップグレードするコンポーネントの数を最小限にとどめます	配備されているシステムでコンポーネントごとにバージョンを追跡する必要があります
すべてのアップグレード	配備されているシステムですべてのコンポーネントについてバージョンの矛盾がありません	アップグレードするコンポーネントの数が最大になります

選択的なアップグレードとすべてのアップグレードのどちらを選択するかについて厳密である必要はありません。たとえば、特定のコンピュータ上で製品コンポーネントについては選択的なアップグレードを選び、選択した製品コンポーネントをサポートするために必要なすべての共有コンポーネントをアップグレードする場合があります。実際、Release 3 から Release 4 へのアップグレードでは、製品コンポーネントについては選択的にアップグレードし、対応する共有コンポーネントについてはすべてをアップグレードする手順が推奨される場合があります。

## 複数インスタンスのアップグレード

アップグレード手順は、配備アーキテクチャーで冗長性が使用されているかどうか、また使用されている場合はどの程度であるかによって変わることがあります。Java ES コンポーネントの複数インスタンスを使用すると、可用性、スケーラビリティ、サービス性、あるいはこれらサービス品質の組み合わせを高めることができます。Java ES 配備アーキテクチャーで冗長性のあるコンポーネントを使用するテクノロジーには、ロードバランス、高可用性の手法 (Sun Cluster および High Availability Session Store)、およびマルチマスターレプリケーション (Directory Server) の 3 種類があります。



冗長性が関係するほとんどの場合で、ダウンタイムなしのアップグレードを実行することが望めます。これらの順次アップグレードでは、コンポーネントが提供するサービスを停止させることなく、それらのコンポーネントの冗長なインスタンスを継続的にアップグレードすることが試みられます。

ほとんどの場合、冗長なインスタンスは複数のコンピュータに配備されます。アップグレード計画の見地からは、このような配備は、複製されたコンポーネントのアップグレードを、その他のコンポーネントのアップグレードから分離することで、ダウンタイムを最小限に抑えることを意味します。言い換えると、複製されたコンポーネントの順次アップグレードを実行する前に、各コンピュータでコンポーネントに対するアップグレード前のすべてのタスクを実行する可能性があるということです。

各複製テクノロジーには、Java ES コンポーネントアップグレードの順序全体に影響を与える可能性がある設定または再設定の手順が含まれます。たとえば、Sun Cluster 環境で実行するコンポーネントでは、Sun Cluster 環境で実行しているコンポーネントをアップグレードする前に、Sun Cluster のアップグレードを必要とすることがあります。

## Java ES コンポーネントの依存関係

前節で説明したとおり、アップグレード計画では、アップグレードが必要な Java ES コンポーネントと、それらコンポーネントをアップグレードする順序を指定します。アップグレード計画で重要な検討項目の 1 つに、配備されたシステム内の各種 Java ES コンポーネントの間の依存関係があります。

選択的なアップグレードでも、すべてのコンポーネントのアップグレードでも、コンポーネントのアップグレードを実行する順序は、コンポーネント間の依存関係によって影響を受けます。

この節では、Java ES コンポーネントの依存関係について説明します。依存関係に関する次の要素が、アップグレード計画に影響を与えます。

- [共有コンポーネントの依存関係](#)
- [製品コンポーネントの依存関係](#)
- [複数インスタンスのアップグレード](#)

これらの各要素について、次の節以降で簡単に説明します。

## 共有コンポーネントの依存関係

Java ES 製品コンポーネントのアップグレード時には、それら Java ES コンポーネントの Java ES 共有コンポーネントに対する依存関係を考慮する必要があります。製品コンポーネントが共有コンポーネントに対して強いアップグレード依存関係を持つ場合、その共有コンポーネントもアップグレードする必要があります。

### 共有コンポーネントの依存関係のマトリックス

42 ページの表 1-6 は、Java ES 2005Q4 (Release 4) 製品コンポーネントの Java ES 共有コンポーネントに対する依存関係を示しています。表 1-6 の列見出しとして記載されている製品コンポーネントの略称は、28 ページの表 1-1 に記載されているものです。共有コンポーネントの略称は、29 ページの表 1-2 に記載されています。

表 1-6 には、4 つの製品コンポーネントが含まれていません。Directory Proxy Server (DPS)、High Availability Session Store (HADB)、および Directory Preparation Tool (DPT) は、共有コンポーネントに対する依存関係がないため省略されています。Service Registry (SR) は新規の製品コンポーネントであり、アップグレード対象としての以前のバージョンは存在しないため省略されています。ただし、もう 1 つの新規 Release 4 製品コンポーネントである Web Proxy Server (WPS) は、Java ES に含まれていなかった以前のリリースから Release 4 にアップグレードできるため、表 1-6 に記載されています。

表 1-6 のマトリックスでは、Release 3 から Release 4 へのアップグレードについて、強いアップグレード依存関係は「H」で示され、弱いアップグレード依存関係は「S」で示されています。Release 2 から Release 4 へのアップグレードについては、すべての共有コンポーネントの依存関係は、当然ながら強いアップグレード依存関係です。つまり、すべての共有コンポーネントを Release 2 から Release 4 にアップグレードする必要があります。

表 1-6 Java ES Release 4 製品コンポーネントの共有コンポーネントに対する依存関係

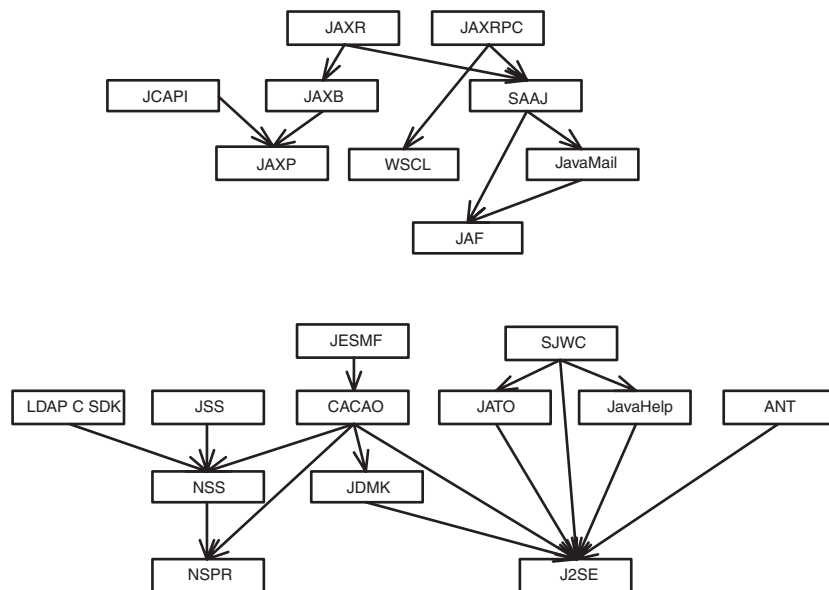
共有コンポーネント	AM	ADS	AS	CS	CX	DA	DPS	DS	IM	MQ	MS	PS	PSRA	SC	WPS	WS
ANT			S													
ACL	S															
BDB	S															
CAC									S					S		
ICU		S	S	S			S	S			H	S			S	S
IM-SDK									H			S				

表 1-6 Java ES Release 4 製品コンポーネントの共有コンポーネントに対する依存関係 ( 続き )

共有コンポーネント	AM	ADS	AS	CS	CX	DA	DPS	DS	IM	MQ	MS	PS	PSRA	SC	WPS	WS
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			
SASL		S		H	H		S	S			H				S	S
SEDC														S		
SJWC	S		S											S		
WSCL	S		S													

表 1-6 に示されているすべての製品コンポーネントの依存関係は、共有コンポーネントに対する直接的な依存関係と間接的な依存関係の両方を表しています。言い換えると、製品コンポーネントは特定の共有コンポーネントに依存し、さらにその共有コンポーネントは別の 1 つまたは複数の共有コンポーネントに依存することもあります。表 1-6 に示されている共有コンポーネントに対する依存関係には、このような間接的な依存関係がすべて含まれています。次の図に、共有コンポーネント間の相互の依存関係を示します。

図 1-1 共有コンポーネントの相互依存関係



## 共有コンポーネントのアップグレードのガイドライン

表 1-6 を参照すると、コンピュータ上で 1 つまたは複数の製品コンポーネントをアップグレードするときに、どの共有コンポーネントをアップグレードするかを判断できます。

- **Release 2 から Release 4 へのアップグレード** : Release 2 から Release 4 にアップグレードする場合、各製品コンポーネントに対して表 1-6 で「S」または「H」と記載されている共有コンポーネントをすべてアップグレードする必要があります。
- **Release 3 から Release 4 へのアップグレード** : すべての製品コンポーネントを Release 3 から Release 4 にアップグレードする場合、各製品コンポーネントに対して表 1-6 で示されている共有コンポーネントをすべてアップグレードしてください。

ただし、製品コンポーネントを選択的にアップグレードする場合でも、コンピュータ上のすべての製品コンポーネントに必要な共有コンポーネントをアップグレードすることをお勧めします。これは、**Release 4** の共有コンポーネントは **Release 3** の製品コンポーネントをサポートすることが保証されているためです。

共有コンポーネントの選択的なアップグレード、つまり、選択的にアップグレードされた製品コンポーネントをサポートする共有コンポーネントだけをアップグレードすること、または弱いアップグレード依存関係ではなく強いアップグレード依存関係のコンポーネントだけをアップグレードすることは、ほとんどの場合に問題ありませんが、この手法には通常よりかなり高いリスクが関係します。

強いアップグレード依存関係が含まれない場合、共有コンポーネントを1つもアップグレードしない選択をすることもできます。ただし、一般的な規則として、基盤となる Java ES 共有コンポーネントベースを最新のバージョンにアップグレードすることをお勧めします。

---

**注** 共有コンポーネントをアップグレードする順序は、[図 1-1](#) に示される共有コンポーネントの相互依存関係によって変わる可能性があります。

また、J2SE から J2SE 5.0 にアップグレードする場合、この共有コンポーネントを最初にアップグレードしてください。J2SE は多くの Java ES コンポーネントのベースコンポーネントです。

---

共有コンポーネントをアップグレードする方法については、[第 2 章「Java ES 共有コンポーネントのアップグレード」](#) を参照してください。

## 製品コンポーネントの依存関係

ある製品コンポーネントが別のコンポーネントに対して持つ依存関係は、アップグレードが必要となる Java ES コンポーネントと、それらをアップグレードする順序を決定する重要な要因です。製品コンポーネントの依存関係は、大きく分けて、実行時の依存関係と設定の依存関係の2つのカテゴリに分類されます。

- 実行時の依存関係**：ソフトウェアシステムの機能は、配備されているコンポーネント間の連携をベースとします。Java ES コンポーネント間のインフラストラクチャの依存関係については、『Sun Java Enterprise System 技術の概要』で説明されています。いずれの Java ES 製品コンポーネントをアップグレードする場合でも、このような依存関係を考慮する必要があります。あるコンポーネントのアップグレードされたバージョンが別のコンポーネントに対して強いアップグレード依存関係を持っている場合、その依存関係は、先に依存先のコンポーネントをアップグレードしたあとで、依存するコンポーネントをアップグレードすべきであることを示しています。

- 設定の依存関係** : 多くの場合、別のコンポーネントを設定するために、Java ES コンポーネントのインストール、設定、および実行が必要です。たとえば、Messaging Server コンポーネントを設定するには、Directory Server 設定ディレクトリを実行する必要があります。また、Access Manager サービスを登録するには、Directory Server ユーザーディレクトリまたはグループディレクトリを実行する必要があります。通常、コンポーネントのアップグレード手順には、アップグレードされたコンポーネントの再設定や設定データの移行が関係しています。実際、いくつかの製品コンポーネントの主要な機能は、別のコンポーネントに対して設定や管理のサポートを提供することです。その結果、設定の依存関係はアップグレード手順に大きな影響を与えます。

表 1-7 は、28 ページの表 1-1 に示される Java ES 製品コンポーネント間の依存関係を示します。表 1-7 を利用して、アップグレードセットの依存連鎖のダイアグラムを作成できます。左の列から順に、製品コンポーネント、別の製品コンポーネントに対する依存関係、各依存関係の特徴、および各コンポーネントをローカルにインストールする必要があるかどうかを示しています。

表 1-7 Java ES 製品コンポーネントの依存関係

製品コンポーネント	依存関係	依存関係の性質	ローカルインストールの必要性
Access Manager	Directory Server	設定データを保存し、ユーザーデータの検索を可能にします	なし
	J2EE Web コンテナ、次のいずれか - Application Server - Web Server - BEA WebLogic Server - IBM WebSphere Application Server	Web コンテナの実行時サービスを提供します	あり
Access Manager SDK	Access Manager	Access Manager サービスを提供します	なし
	J2EE Web コンテナ、次のいずれか - Application Server - Web Server - BEA WebLogic Server - IBM WebSphere Application Server	Web コンテナの実行時サービスを提供します	あり
Administration Server	Directory Server	設定ディレクトリを提供します	なし

表 1-7 Java ES 製品コンポーネントの依存関係 (続き)

製品コンポーネント	依存関係	依存関係の性質	ローカルインストールの必要性
Application Server	Message Queue	信頼できる非同期メッセージングを提供します	あり
	Web Server (省略可能)	インスタンス間のロードバランズを提供します	あり
	High Availability Session Store (省略可能)	インスタンス間のフェイルオーバーをサポートするために必要なセッション状態を保存します	あり
Calendar Server	Directory Server	ユーザーデータを保存し、それらの検索を可能にします	なし
	Directory Preparation Tool	Calendar Server が使用するディレクトリを準備します	なし
	Access Manager (省略可能)	シングルサインオンを提供します	なし
	Messaging Server (省略可能)	電子メールによる通知を提供します	なし
	Delegated Administrator (省略可能)	カレンダーサービスのためのユーザーをプロビジョニングします	なし

表 1-7 Java ES 製品コンポーネントの依存関係 ( 続き )

製品コンポーネント	依存関係	依存関係の性質	ローカルインストールの必要性
Communications Express	J2EE Web コンテナ、次のいずれか - Application Server - Web Server	Web コンテナの実行時サービスを提供します	あり
	Directory Server	アドレス帳などのユーザーデータを保存し、それらの検索を可能にします	なし
	Directory Preparation Tool	Communications Express が使用するディレクトリを準備します	なし
	Access Manager または Access Manager SDK	認証、認証サービス、およびシングルサインオンを提供します	あり
	Messaging Server	メッセージングへの Web ベースのアクセスを可能にします	なし
	Calendar Server	カレンダーへの Web ベースのアクセスを可能にします	なし
Delegated Administrator	J2EE Web コンテナ、次のいずれか - Application Server - Web Server	Web コンテナの実行時サービスを提供します	あり
	Directory Server	ユーザーデータを保存します	なし
	Directory Preparation Tool	Delegated Administrator が使用するディレクトリを準備します	なし
	Access Manager または Access Manager SDK	ユーザーのプロビジョニングに必要な API を提供します	あり
	Directory Preparation Tool	Directory Server	通信コンポーネントが使用するために準備するユーザーディレクトリまたはグループディレクトリを提供します
Directory Proxy Server	Administration Server	Directory Proxy Server を設定します	なし
	Directory Server	ディレクトリへのアクセスを提供します	なし



表 1-7 Java ES 製品コンポーネントの依存関係 ( 続き )

製品コンポーネント	依存関係	依存関係の性質	ローカルインストールの必要性
Directory Server	Administration Server	Directory Server を設定します	なし
High Availability Session Store	なし		
Instant Messaging	J2EE Web コンテナ、次のいずれか - Application Server - Web Server	Web コンテナの実行時サービスを提供します	あり
	Directory Server	ユーザーデータを保存します	なし
	Access Manager ( 省略可能 )	シングルサインオンを提供します	なし
Message Queue	なし		
Messaging Server 次を保存 MTA MMP MEM	Directory Server	設定データを保存し、ユーザーデータの検索を可能にします	なし
	Administration Server	Directory Server 設定ディレクトリに設定データを保存します	あり
	Directory Preparation Tool	Messaging Server が使用するディレクトリを準備します	なし
	Access Manager ( 省略可能 )	シングルサインオンを提供します	なし
	Delegated Administrator ( 省略可能 )	メッセージングサービスのためのユーザーをプロビジョニングします	なし

表 1-7 Java ES 製品コンポーネントの依存関係 (続き)

製品コンポーネント	依存関係	依存関係の性質	ローカルインストールの必要性
Portal Server	J2EE Web コンテナ、次のいずれか - Application Server - Web Server - BEA WebLogic Server - IBM WebSphere Application Server	Web コンテナの実行時サービスを提供します	あり
	Directory Server	ユーザープロフィールを保存し、それらの検索を可能にします	なし
	Access Manager または Access Manager SDK	認証、認証サービス、およびシングルサインオンを提供します	あり
	Communications Express	メッセージングとカレンダーのチャンネルを提供します	なし
Portal Server Secure Remote Access	Portal Server	ポータルへのアクセスを提供します	あり
	Access Manager または Access Manager SDK	認証、認証サービス、およびシングルサインオンを提供します	あり
Sun Cluster	なし		
Sun Cluster エージェント	Sun Cluster	Sun Cluster サービスへのアクセスを提供します	
Web Proxy Server	なし		
Web Server	なし		

# 手順に関する一般的なガイドライン

これまでの節で説明したすべての要因は、どの Java ES コンポーネントをアップグレードするかについての計画と、それらをアップグレードする順序に影響を与える可能性があります。また、これらの要因は複数のコンピュータに配備されている Java ES コンポーネントのアップグレード方法にも影響を与えます。これらすべての要因による特定の影響は、配備アーキテクチャーによって異なります。

とはいえ、すべての場合には適用できませんが、手順に関する一般的なガイドラインがいくつか存在します。1 台のコンピュータまたは 1 つの配備システムで Java ES コンポーネントを正常にアップグレードするための手順を次に示します。アップグレードの実行時、配備アーキテクチャーに含まれていないコンポーネントや、選択的なアップグレードを実行する場合にアップグレード計画に含まれないコンポーネントについては単純に省略してください。

---

**注** この『アップグレードガイド』の各章は、次に示す手順のガイドラインと同様、通常 Java ES コンポーネントをアップグレードする順序に従って説明されています。

---

1. **共有コンポーネント** (55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」を参照)

通常は、共有コンポーネントに依存するコンポーネントよりも前に、共有コンポーネントをアップグレードしてください。

2. **Sun Cluster ソフトウェア** (97 ページの第 3 章「Sun Cluster ソフトウェア」を参照)

Sun Cluster 環境で実行されるコンポーネントがあり、Sun Cluster ソフトウェアをアップグレードする必要がある場合、Sun Cluster サービスを使用するコンポーネントよりも前にアップグレードしてください。Sun Cluster エージェントをアップグレードする場合、Sun Cluster のアップグレードの一部としてアップグレードしてください。

3. **Directory Server および Administration Server** (105 ページの第 4 章「Directory Server と Administration Server」を参照)

多くのコンポーネントが Directory Server にユーザーデータや設定データを保存するため、通常は Directory Server に実行時の依存関係または設定の依存関係を持つコンポーネントをアップグレードする前に、Directory Server のアップグレードを実行してください。Administration Server は、Directory Server とともにアップグレードする必要があります。

4. **Directory Proxy Server** (127 ページの第 5 章「Directory Proxy Server」を参照)

Directory Proxy Server は Directory Server および Administration Server に対して強いアップグレード依存関係を持つため、Directory Server および Administration Server のあとにアップグレードします。その他のコンポーネントは、Directory Proxy Server を通じて Directory Server にアクセスすることがあります。

5. **Web Server** (141 ページの第 6 章「Web Server」を参照)

多数の Java ES コンポーネントが Web コンテナのサポートを必要とするため、Web コンテナをアップグレードする場合、Web コンテナのサービスを必要とするコンポーネントよりも先に Web コンテナをアップグレードしてください。通常、Web コンテナのサービスは、Web Server または Application Server によって提供されますが、これら両方がアーキテクチャーに含まれる場合は最初に Web Server をアップグレードします。

6. **Message Queue** (151 ページの第 7 章「Message Queue」を参照)

Message Queue を最適にアップグレードするには、Java 2 Enterprise Edition (J2EE) に準拠した Message Queue を必要とする Application Server よりも前にアップグレードします。

7. **High Availability Session Store** (169 ページの第 8 章「High Availability Session Store」を参照)

High Availability Session Store を最適にアップグレードするには、高可用性のために High Availability Session Store を必要とする Application Server よりも前にアップグレードします。

8. **Application Server** (177 ページの第 9 章「Application Server」を参照)

Application Server のロードバランスプラグインは Web Server に依存するため、ロードバランスを使用する場合は Web Server のあとに Application Server をアップグレードしてください。

9. **Web Proxy Server** (197 ページの第 10 章「Web Proxy Server」を参照)

Web Proxy Server はどの時点でもアップグレードできますが、通常はプロキシサービスの提供先である Web Server コンポーネントまたは Application Server コンポーネントのあとにアップグレードします。Web Proxy Server は新規の Java ES Release 4 コンポーネントですが、以前の非 Java ES リリースからアップグレードできます。

10. **Access Manager** (205 ページの第 11 章「Access Manager」を参照)

Access Manager は、シングルサインオンを含み、認証および承認で中心的な役割を担うため、これらのサービスについて依存するコンポーネントよりも前にアップグレードしてください。また、Access Manager には特定の Directory Server スキーマ (スキーマ 2) が必要ですが、これはその他のコンポーネントが Directory Server をどのように使用するかに影響を与えます。

11. **Directory Preparation Tool** (235 ページの第 12 章「Directory Preparation Tool」を参照)

Directory Preparation Tool は Directory Server スキーマに依存するため、Access Manager のアップグレード後に Directory Server に対して実行してください。このガイドラインの例外については、[227 ページの「Java ES Release 2 からの Access Manager のアップグレード」](#)を参照してください。Directory Preparation Tool をアップグレードする場合、ディレクトリに対する変更を反映させるために、Directory Preparation Tool に依存する通信コンポーネント、すなわち Messaging Server、Calendar Server、Communications Express、および Delegated Administrator のアップグレードよりも前に Directory Preparation Tool をアップグレードしてください。

**12. Messaging Server (247 ページの第 13 章「Messaging Server」を参照)**

Messaging Server をアップグレードする場合は、必ず先行するアップグレードのあとに行い、また、Messaging Server コンポーネントに依存する Communications Express よりも前にアップグレードしてください。

**13. Calendar Server (265 ページの第 14 章「Calendar Server」を参照)**

Calendar Server のいくつかの機能は Messaging Server のサポートを必要とするため、アップグレードする場合は Messaging Server のあとに行ってください。また、Calendar Server に依存する Communications Express よりも前に Calendar Server をアップグレードしてください。

**14. Communications Express (277 ページの第 15 章「Communications Express」を参照)**

Communications Express は、Calendar Server、Messaging Server、Directory Preparation Tool、Access Manager、Web Server、および Directory Server の先行する多数のコンポーネントに依存するため、それらのあとにアップグレードしてください。

**15. Instant Messaging (277 ページの第 15 章「Communications Express」を参照)**

通常、Instant Messaging は Access Manager のアップグレード後、いつでもアップグレードできます。

**16. Portal Server (313 ページの第 17 章「Portal Server」を参照)**

Communications Express と同様に Portal Server は多数の先行するコンポーネントに依存し、特にメッセージングとカレンダーのチャンネルを提供するために Communications Express に依存するため、アップグレードする場合は Communications Express のあとに行ってください。

**17. Portal Server Secure Remote Access (343 ページの第 18 章「Portal Server Secure Remote Access」を参照)**

Portal Server Secure Remote Access は Portal Server のアップグレード後、いつでもアップグレードできます。

**18. Delegated Administrator (367 ページの第 19 章「Delegated Administrator」を参照)**

Delegated Administrator は Directory Preparation Tool のアップグレード後、いつでもアップグレードでき、ユーザーのプロビジョニングに使用することができます。また、Directory Server を背景にして実行されます。慣例として、その他のサービスをアップグレードして起動したあとにユーザーをプロビジョニングしますが、Delegated Administrator はユーザーのプロビジョニングに関して Delegated Administrator に依存する通信コンポーネントをアップグレードする前にアップグレードできます。

# Java ES 共有コンポーネントのアップグレード

この章では、Java ES 共有コンポーネントの Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。

この章で説明する内容は、次のとおりです。

- 56 ページの「共有コンポーネントのアップグレードの概要」
- 64 ページの「個別のパッチを適用することによる共有コンポーネントのアップグレード」
- 66 ページの「パッチクラスタを使用した共有コンポーネントのアップグレード」
- 70 ページの「パッケージの置換によるコンポーネントのアップグレード」
- 80 ページの「特別なアップグレード手順を必要とするコンポーネント」

---

## 注

**Sun Cluster** ソフトウェアのアップグレードに備えて共有コンポーネントをアップグレードするには、『Sun Cluster Software Installation Guide for Solaris OS』の「Upgrading Sun Cluster Software」に説明されている依存関係を持つソフトウェアのアップグレード手順に従ってください。このマニュアルは次の場所で参照できます。

<http://docs.sun.com/app/docs/doc/817-4229/6mjp4pjuh?l=j&a=view>

ただし、セキュリティーコンポーネントは、Sun Cluster 1 of 2 CD-ROM または Sun Cluster 2 of 2 CD-ROM からではなく、Java ES 2005Q4 (Release 4) ディストリビューションからインストールしてください。

---

# 共有コンポーネントのアップグレードの概要

第1章「アップグレードの計画」で説明したように、共有コンポーネントの Java ES 2005Q4 (Release 4) へのアップグレードは、より規模の大きいアップグレード計画の一部として実行してください。アップグレードを成功させるためには、第1章を詳しく参照し、必要に応じたアップグレード計画を準備してください。

## アップグレード計画について

アップグレード計画には、次の分野が含まれている必要があります。

- **オペレーティングシステムの問題** : 33 ページの「[オペレーティングシステムの問題](#)」で説明した手順に従って、オペレーティングシステムのアップグレードを実行します。Solaris 10 OS 以外のすべてのプラットフォームで、共有コンポーネントをアップグレードする前にオペレーティングシステムをアップグレードします。
- **アップグレードパス** : 現在配備されている Java Enterprise System のバージョンを確認し、Java ES Release 4 へのアップグレードでサポートされるパスを確認します。ほとんどの場合、共有コンポーネントのアップグレード時には、Java ES 2004Q2 (Release 2) からのアップグレードと Java ES 2005Q1 (Release 3) からのアップグレードでは同じ手順に従います。この章では、個別のアップグレードパスに依存する手順を説明します。アップグレードパスの詳細については、36 ページの「[アップグレードパス](#)」を参照してください。
- **アップグレードの依存関係** : アップグレードする製品コンポーネントの相互依存関係を理解する必要があります。通常、依存関係に基づいてアップグレードする製品コンポーネントの順序を決定します。たとえば、あるコンポーネントをアップグレードする前に、そのコンポーネントが依存するコンポーネントをすべてアップグレードします。その他、38 ページの「[アップグレードの依存関係](#)」で説明したとおり、強いアップグレード依存関係や弱いアップグレード依存関係など、考慮すべきさまざまな要因があります。  
  
42 ページの表 1-6 を使用して、製品コンポーネントをアップグレードする前にアップグレードする必要のある共有コンポーネントを確認します。
- **すべてのアップグレードと選択的なアップグレード** : 1 台のコンピュータ上ですべての製品コンポーネントをアップグレードする場合、製品コンポーネントの依存先の共有コンポーネントもすべてアップグレードしてください。ただし、Release 3 から Release 4 にアップグレードする場合、コンピュータ上の一部の製品コンポーネントを選択的にアップグレードし、それ以外のコンポーネントをアップグレードしないこともできます。しかし、最善の方法は、コンピュータ上のすべての製品コンポーネントが依存する共有コンポーネントをすべてアップグレードすることです。Release 4 の共有コンポーネントは Release 3 の製品コンポーネントをサポートすることが保証されています。詳細については、44 ページの「[共有コンポーネントのアップグレードのガイドライン](#)」を参照してください。



- **順序付けのガイドライン**: 51 ページの「[手順に関する一般的なガイドライン](#)」に説明されている順序付けのガイドラインを確認してください。通常、先に共有コンポーネントをアップグレードします。ただし、アップグレードプロセスを開始する前に、Java ES Release 4 にアップグレードするための全体の順番を理解する必要があります。

## 共有コンポーネントのアップグレードに関するテクノロジー

共有コンポーネントを Java ES Release 4 へアップグレードするには 3 つのテクノロジーがあります。使用するテクノロジーは、アップグレード計画に従って、アップグレードする共有コンポーネントの数やタイプによって変わります。

3 つのテクノロジーは次のとおりです。

- **パッチ**: Solaris プラットフォーム上のほとんどの共有コンポーネントは、パッチアプリケーションを使って Java ES Release 4 にアップグレードできます。パッチでは、通常 1 つのコンポーネントまたは関連するコンポーネントのグループがアップグレードされます。

少数の共有コンポーネントをアップグレードする計画の場合、アップグレードパッチが提供されているコンポーネントに対して個別にパッチを適用することができます。

60 ページの表 2-1 は、共有コンポーネントごとに利用できるアップグレードパッチを示しています。64 ページの「[個別のパッチを適用することによる共有コンポーネントのアップグレード](#)」では、パッチのダウンロードと適用の手順を説明しています。

- **パッチクラスタ**: パッチクラスタには、共有コンポーネントに利用できるアップグレードパッチがすべて含まれています。パッチクラスタとともに提供される単一のアップグレードスクリプトを実行することで、関連する共有コンポーネントをすべてアップグレードできるため、アップグレードプロセスが簡略化されます。

Solaris プラットフォームごとに個別のパッチクラスタが用意されています。Linux プラットフォームで利用できるパッチクラスタはありません。

複数の共有コンポーネントをアップグレードする計画の場合、パッチクラスタを使用してください。パッチクラスタで網羅されている共有コンポーネントをすべてアップグレードしない場合でも、アップグレード計画で指定された共有コンポーネントをアップグレードするには、パッチクラスタを使用することがもっとも効率的な方法となる可能性があります。

パッチクラスタのダウンロードと適用方法については、66 ページの「[パッチクラスタを使用した共有コンポーネントのアップグレード](#)」を参照してください。

- **パッケージの置換** : 一部の共有コンポーネントは、システム上の既存のパッケージを新しいバージョンのパッケージで置き換えることによってのみアップグレードできます。新しいバージョンの共有コンポーネントパッケージは、Java ES Release 4 ディストリビューションから入手できます。

パッチテクノロジーは Linux プラットフォームの Java Enterprise System では利用できないため、Linux の共有コンポーネントは通常 RPM パッケージの置換によってアップグレードします。ただし、一部の共有コンポーネントでは RPM パッケージをパッチとして提供しています。

60 ページの表 2-1 に、Java ES Release 4 へのアップグレード時にパッケージの置換を使用する共有コンポーネントを示します。パッケージの置換の詳細については、70 ページの「[パッケージの置換によるコンポーネントのアップグレード](#)」で説明します。

## 一般的なアップグレード手順

ここでは、共有コンポーネントの一般的なアップグレード手順について説明します。

1. アップグレード計画で、アップグレードパスとアップグレードする共有コンポーネントを確認します。

アップグレード計画の作成については、この概要のここまでの説明を参照してください。また、詳細については、34 ページの「[アップグレードの計画](#)」を参照してください。

2. アップグレード計画で指定された共有コンポーネントのアップグレードに利用できるアップグレードテクノロジーを確認します。

詳細については、次の各節を参照してください。

- 61 ページの「[Release 2 の共有コンポーネントのアップグレード](#)」
  - 59 ページの「[Release 3 の共有コンポーネントのアップグレード](#)」
3. 特定の必要に応じ、次の各節で説明される手順に従います。
    - 64 ページの「[個別のパッチを適用することによる共有コンポーネントのアップグレード](#)」
    - 66 ページの「[パッチクラスタを使用した共有コンポーネントのアップグレード](#)」
    - 70 ページの「[パッケージの置換によるコンポーネントのアップグレード](#)」
    - 80 ページの「[特別なアップグレード手順を必要とするコンポーネント](#)」

## Release 3 の共有コンポーネントのアップグレード

60 ページの表 2-1 は、Java ES Release 3 からの共有コンポーネントのアップグレード時に使用するアップグレードテクノロジーを示しています。次の事項に注意してください。

- **Solaris プラットフォーム** : 多様なアップグレードオプションを選択できます。
  - ほとんどの場合、特定のパッチを適用して、共有コンポーネントを Java ES Release 4 にアップグレードできます。
  - 通常、共有コンポーネントを Release 3 からアップグレードするには、プラットフォーム固有のパッチクラスタを適用することがもっとも効率的な方法です。パッチクラスタには、すべての共有コンポーネントを Release 4 にアップグレードするためのすべてのパッチが含まれています。詳細については、66 ページの「[パッチクラスタを使用した共有コンポーネントのアップグレード](#)」を参照してください。
  - 場合によっては、Java ES ディストリビューションとともに提供される Release 4 パッケージで Release 3 パッケージを置換する必要があります。
  - Release 3 以降変更されていない共有コンポーネントについては、アップグレードは不要です。
- **Linux プラットフォーム** : RPM パッケージのインストールまたは置換によって、共有コンポーネントをアップグレードする必要があります。表 2-1 でパッチ ID が示されているものについては、コンポーネントのアップグレードに必要な RPM パッケージを含んだパッチが提供されているため、パッケージのダウンロードとインストールを簡単に行うことができます。パッチには特別の手順があります。また、パッチ ID を使用して、共有コンポーネントに対する変更を追跡できます。Linux プラットフォームで利用できるパッチクラスタはありません。

---

**注** 表 2-1 に示されているパッチ ID の下 2 桁は、パッチのリビジョン番号です。リビジョン番号が大きいほど、バージョンが新しいことを示しています。

表 2-1 には、アップグレードに最低限必要なリビジョンだけを示しています。さらに新しいリビジョンのパッチが提供された場合、表に示されているものではなく新しいほうのリビジョンを適用してください。

表 2-2 に示される共有コンポーネントのフルネームは、29 ページの「[Release 4 の共有コンポーネント](#)」に記載されています。

---

表 2-1 共有コンポーネントを Java ES Release 3 からアップグレードするためのアップグレードテクノロジー

共有コンポーネント	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
ANT	パッケージを置換					
ACL	パッケージを置換					
BDB	パッケージを置換					
CAC	パッケージを置換					
ICU	116103-08	114677-10	119810-01	114678-10	119811-01	パッケージを置換
IM-SDK	118789-09			118790-09		118791-10
J2SE™	86 ページの「Java ES Release 4 のための J2SE のアップグレード」に説明されている手順に従って J2SE 5.0 をインストール					
JAF	Release 3 以降変更なし。アップグレードは不要					
JATO	パッケージを置換 (省略可能、84 ページの「JATO のアップグレード」を参照)					
JavaHelp™	Release 3 以降変更なし。アップグレードは不要					パッケージを置換
JavaMail™	Release 3 以降変更なし。アップグレードは不要					
JAX 関連コンポーネント JAXB JAXP JAXR JAX-RPC SAAJ WSCL	プラットフォーム固有のパッチクラスタを適用 67 ページの「パッチクラスタの手順」を参照					119190-03
JCAPI	Release 3 以降変更なし。アップグレードは不要					
JDMK	119044-01					119046-01
JSS	119209-05	119211-05	119213-06	119212-05	119214-06	パッケージを置換
KTSE	Release 3 以降変更なし。アップグレードは不要					
LDAP C SDK	116837-02			116838-02		118353-02
LDAP J SDK	119725-02					パッケージを置換
MA Core	119527-02			119528-02		119529-02

表 2-1 共有コンポーネントを Java ES Release 3 からアップグレードするためのアップグレードテクノロジー (続き)

共有コンポーネント	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
MFWK	119803-02			119804-02		パッケージを置換
NSPR	119209-05	119211-05	119213-05	119212-05	119214-05	パッケージを置換
NSS	119209-05	119211-05	119213-05	119212-05	119214-05	パッケージを置換
SASL	115328-02	115342-02	119345-01	115343-02	119346-01	パッケージを置換
SEDC	86 ページの「 <a href="#">Sun Explorer Data Collector のアップグレード</a> 」を参照					
SJWC	パッケージを置換					

## Release 2 の共有コンポーネントのアップグレード

次に示す表 2-2 は、Java ES Release 2 からの共有コンポーネントのアップグレード時に使用するアップグレードテクノロジーを示しています。次の事項に注意してください。

- Solaris プラットフォーム**：ほとんどの場合、プラットフォーム固有のパッチクラスタを適用して、共有コンポーネントをアップグレードします。それ以外の場合、Java ES ディストリビューションとともに提供される Release 4 パッケージを使用してパッケージをインストールまたは置換します。
- Linux プラットフォーム**：RPM パッケージのインストールまたは置換によって、共有コンポーネントをアップグレードする必要があります。表 2-2 でパッチ ID が示されているものについては、コンポーネントのアップグレードに必要な RPM パッケージを含んだパッチが提供されているため、パッケージのダウンロードとインストールを簡単に行うことができます。パッチには特別の手順があります。また、パッチ ID を使用して、共有コンポーネントに対する変更を追跡できます。Linux プラットフォームで利用できるパッチクラスタはありません。

注	<p>表 2-2 に示されているパッチ ID の下 2 桁は、パッチのリビジョン番号です。リビジョン番号が大きいほど、バージョンが新しいことを示しています。</p> <p>表 2-2 には、アップグレードに最低限必要なリビジョンだけを示しています。さらに新しいリビジョンのパッチが提供された場合、表に示されているものではなく新しいほうのリビジョンを適用してください。</p> <p>表 2-2 に示される共有コンポーネントのフルネームは、29 ページの「Release 4 の共有コンポーネント」に記載されています。</p>
---	--

表 2-2 共有コンポーネントを Java ES Release 2 からアップグレードするためのアップグレードテクノロジー

共有コンポーネント	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
ANT	パッケージを置換					
ACL	パッケージを置換					
BDB	パッケージを置換					
CAC	パッケージをインストール					
ICU	プラットフォーム固有のパッチクラスタを適用					パッケージを置換
IM-SDK	プラットフォーム固有のパッチクラスタを適用					118791-09
J2SE™	86 ページの「Java ES Release 4 のための J2SE のアップグレード」に説明されている手順に従って J2SE 5.0 をインストール					
JAF	プラットフォーム固有のパッチクラスタを適用					パッケージをインストール
JATO	プラットフォーム固有のパッチクラスタを適用					パッケージをインストール
JavaHelp™	パッケージを置換					
JavaMail™	プラットフォーム固有のパッチクラスタを適用					パッケージをインストール

表 2-2 共有コンポーネントを Java ES Release 2 からアップグレードするためのアップグレードテクノロジー ( 続き )

共有コンポーネント	Solaris 8 SPARC	Solaris 9 SPARC	Solaris 10 SPARC	Solaris 9 x86	Solaris 10 x86	Linux
JAX 関連コンポーネント JAXB JAXP JAXR JAX-RPC SAAJ WSCL	プラットフォーム固有のパッチクラスタを適用					119190-03
JCAPI	プラットフォーム固有のパッチクラスタを適用					118613-01
JDMK	プラットフォーム固有のパッチクラスタを適用					119046-01
JSS	プラットフォーム固有のパッチクラスタを適用					パッケージをインストール
KTSE	プラットフォーム固有のパッチクラスタを適用					パッケージを置換
LDAP C SDK	プラットフォーム固有のパッチクラスタを適用					118353-02
LDAP J SDK	プラットフォーム固有のパッチクラスタを適用					パッケージをインストール
MA Core	プラットフォーム固有のパッチクラスタを適用					119529-02
NSPR	プラットフォーム固有のパッチクラスタを適用					パッケージを置換
NSS	プラットフォーム固有のパッチクラスタを適用					パッケージを置換
SASL	プラットフォーム固有のパッチクラスタを適用					パッケージを置換
SEDC	86 ページの「Sun Explorer Data Collector のアップグレード」を参照					
SJWC	パッケージをインストール					

# 個別のパッチを適用することによる共有コンポーネントのアップグレード

Solaris プラットフォームでは、インストールされているパッケージを変更するパッチアプリケーションによって、共有コンポーネントの多数のパッケージを Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) にアップグレードできます。

---

**注** Java ES 2004Q2 (Release 2) からのアップグレード時には、[66 ページの「パッチクラスタを使用した共有コンポーネントのアップグレード」](#)で説明されているように、個々のパッチを適用せず、パッチクラスタを適用してください。

---

パッケージの置換と比較して、パッチテクノロジーの利点は、パッチで適用したリビジョンをあとで必要に応じてバックアウトできることです。通常、パッチのサイズは更新されたパッケージのサイズよりも小さいため、ダウンロードとインストールを容易に行うことができます。また一般的には、最新のパッケージよりも新しいパッチが用意されています。最新リビジョンのパッチは、この節で説明するように SunSolve からダウンロードできます。

[60 ページの表 2-1](#) には、Java ES Release 3 から Java ES Release 4 へのアップグレードに使用するすべての共有コンポーネントパッチのパッチ ID を示しています。

Solaris プラットフォームのアップグレード計画で、複数の共有コンポーネントをアップグレードする場合、パッチクラスタを使用することがアップグレードを実行するためのもっとも効率的な方法です。パッチクラスタには、パッチテクノロジーを使用する共有コンポーネントのアップグレードに利用できるパッチがすべて含まれます。パッチクラスタの詳細については、[66 ページの「パッチクラスタを使用した共有コンポーネントのアップグレード」](#)を参照してください。

少数の共有コンポーネントをアップグレードする場合、次に示す手順でパッチを個別に適用することもできます。続いて説明するとおり、適用したパッチは、あとでバックアウトできます。

---

**注** システムに変更を加える前に、最初にシステムをバックアップすることをお勧めします。

---



## Java ES 2005Q4 へのパッチアップグレード

60 ページの表 2-1 に示す個々のパッチを使用して共有コンポーネントをアップグレードできます。次の手順に従ってください。

1. 表 2-1 に示されている該当のパッチを取得します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. パッチを適用する前に、パッチとともに提供される README を参照して、特別な指示がないかを確認します。

4. 次の例で示すように、patchadd コマンドを使用してパッチを適用します。これによって、Mobile Access コアの共有コンポーネントにパッチが適用されます。

```
patchadd 119527-02
```

patchadd コマンドの詳細については、patchadd(1M) マニュアルページを参照してください。

## パッチアップグレードのロールバック

次の手順に従って、個々のパッチアップグレードをロールバックできます。

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. 次の例で示すように、patchrm コマンドを使用してパッチをバックアウトします。これによって、Mobile Access Core の共有コンポーネントへのパッチがバックアウトされます。

```
patchrm 119527-02
```

patchrm コマンドの詳細については、patchrm(1M) マニュアルページを参照してください。

# パッチクラスタを使用した共有コンポーネントのアップグレード

パッチクラスタは、共有コンポーネントを Java ES 2005Q4 (Release 4) にアップグレードする便利な方法です。Solaris の各プラットフォームには、パッチテクノロジーを使用する共有コンポーネントのアップグレードに利用できるパッチをすべて含むパッチクラスタが用意されています。共有コンポーネントのパッチクラスタを適用すると、パッチクラスタに含まれるすべてのパッチが適用されます。

使用するパッチクラスタは、アップグレードパスに依存しません。Java ES Release 2 からのアップグレード時には、Java ES Release 3 からのアップグレード時と同じプラットフォーム固有のパッチクラスタを使用します。

次のパッチクラスタを利用できます。

```
Java ES コンポーネントパッチ Solaris 10 SPARC
Java ES コンポーネントパッチ Solaris 10 x86
Java ES コンポーネントパッチ Solaris 9 SPARC
Java ES コンポーネントパッチ Solaris 9 x86
Java ES コンポーネントパッチ Solaris 8 SPARC
```

次の場所にある SunSolve からパッチクラスタをダウンロードします。

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

---

**注** Linux プラットフォームでは Java ES Release 4 にアップグレードするパッチが提供されている共有コンポーネントは少数なので、Linux プラットフォーム用のパッチクラスタはありません。

---

## パッチクラスタの内容

パッチクラスタに含まれているパッチは、Solaris プラットフォームごとに異なります。新規のパッチが提供されると、パッチクラスタの内容は更新されます。パッチクラスタに含まれるパッチの一覧は、パッチクラスタとともに提供される Cluster\_readme ファイルに記載されています。

パッチクラスタをダウンロードすると、ZIP 形式でプラットフォーム固有のファイルを取得できます。ファイルの内容を、パッチクラスタの適用元ディレクトリに展開します。展開された内容の最上位のディレクトリには、次のファイルが含まれます。

- Cluster\_readme  
パッチクラスタの適用に関する情報を提供します
- install\_cluster スクリプト

このスクリプトを実行して、パッチクラスタを適用します

- copyright

パッチクラスタとドキュメントに関する著作権表示です

展開された内容には、パッチクラスタに含まれるパッチごとのディレクトリもあります。これらのパッチディレクトリには、各パッチに適用できる README ファイルが含まれます。

## パッチクラスタの手順

少数の共有コンポーネントだけをアップグレードするアップグレード計画の場合は、パッチクラスタ以外の方法でもっとも効率的に Java ES Release 4 へアップグレードできる場合があります。64 ページの「個別のパッチを適用することによる共有コンポーネントのアップグレード」に説明されるように、個別のパッチを適用する方法を検討することもできます。

Solaris プラットフォームで複数の共有コンポーネントをアップグレードする場合は、パッチクラスタを使用してコンポーネントをアップグレードすることをお勧めします。

install\_cluster スクリプトはパッチクラスタ内のすべてのパッチの適用を試みることに留意してください。このスクリプトは、スクリプトを実行したコンピュータにインストールされている共有コンポーネントをアップグレードします。コンピュータ上に存在しない共有コンポーネントにパッチを適用しようとした場合、または不要なパッチを適用しようとした場合、当然のことながらパッチクラスタスクリプトは処理に失敗します。

install\_cluster スクリプトの実行中、進行状況が表示されます。また、詳細な情報をログファイルで参照できます。

次の手順で、共有コンポーネントのパッチクラスタを適用する方法を示します。

---

**注**           パッチクラスタをロールバックすることはできません。この場合、パッチクラスタスクリプトによって適用されたパッチをすべて追跡し、パッチを個別にロールバックする必要があります (65 ページの「パッチアップグレードのロールバック」を参照)。パッチクラスタを適用する前にシステムをバックアップすることをお勧めします。

---

## Solaris 8 から Solaris 9 へのアップグレード

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. 次のパッケージが存在する場合、それらを削除します。

```
SUNWjato  
SUNWjaxb  
SUNWjxap  
SUNWjaf  
SUNWjmail  
SUNWxrgprt  
SUNWxrpert  
SUNWxsrt
```

これらの選択されたパッケージは、Java ES インストールの一部ではなかった共有コンポーネントの JATO、Java Activation Framework (JAF)、または JAX ファミリーから提供されているか、パッチクラスタを使用してアップグレードすることができないバージョンである可能性があります。パッチクラスタスクリプトによって、これらのパッケージを確実に新しいバージョンにアップグレードするには、これらのパッケージを削除する必要があります。

これらのパッケージを削除するには、次のコマンドを実行します。

```
pkgrm SUNWjato SUNWjaxb SUNWjxap SUNWjaf SUNWjmail SUNWxrgprt  ¥  
SUNWxrpert SUNWxsrt
```

3. 次の場所にある SunSolve から Solaris プラットフォーム用の適切なパッチクラスタを取得します。

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

4. プラットフォーム固有の ZIP ファイルの内容を、インストールスクリプトを実行できる場所に展開します。
5. 重要な指示とパッチに関する情報が記載された README を参照します。

README には「Save and Backout Option」というセクションがあります。ここには、あとで変更をバックアウトする可能性がある場合に、パッチクラスタを適用する方法が説明されています。

6. パッチクラスタに含まれたパッチをインストールする `install_cluster` スクリプトを実行します。

## Solaris 10 でのアップグレード (Java ES Release 3 からのみ)

1. `root` としてログインするか、スーパーユーザーになります。

```
su -
```

2. システムに存在する可能性がある次のパッケージのバージョンを確認します。

```
SUNWjxap  
SUNWxrgprt  
SUNWxrpert  
SUNWxsrt
```

- a. 次のコマンドを使用して、パッケージのバージョンを確認します。

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

- b. 確認したバージョンを次のバージョンと比較します。

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

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

パッケージのバージョンがプラットフォームに対応するバージョンと一致しない場合、またはパッケージがコンピュータにインストールされていない場合、[68 ページの手順 3](#)に進みます。

パッケージのバージョンがプラットフォームに対応するバージョンと一致する場合、次の[手順 c](#)に進みます。

- c. Java ES ディストリビューションにこれらのパッケージが含まれるかどうかを確認します。

完全な Java ES ディストリビューションには、これらのパッケージが含まれます。サブセットディストリビューションを所持している場合、プラットフォームに対応するディレクトリに移動し、パッケージが存在するかどうかを確認します。

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

これらのパッケージがサブセットディストリビューションに含まれない場合、これらのパッケージのアップグレードは不要です。コンピュータにインストールされているこれらのパッケージをそのまま残し、[68 ページの手順 3](#)に進みます。

これらのパッケージがディストリビューションに含まれる場合、次の[手順 d](#)に進みます。

- d. 次のコマンドを実行して、インストールされているパッケージをシステムから削除します。

```
pkgrm SUNWjaxp SUNWxrgprt SUNWxrpprt SUNWxsrt
```

- e. 次のコマンドを発行して、ディストリビューションから対応するパッケージをインストールします。

```
pkgadd -d . SUNWjaxp SUNWxrgprt SUNWxrpprt SUNWxsrt
```

3. 次の場所にある SunSolve から Solaris プラットフォーム用の適切なパッチクラスタを取得します。

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

4. プラットフォーム固有の ZIP ファイルの内容を、インストールスクリプトを実行できる場所に展開します。
5. 重要な指示とパッチに関する情報が記載された README を参照します。

README には「Save and Backout Option」というセクションがあります。ここには、あとで変更をバックアウトする可能性がある場合に、パッチクラスタを適用する方法が説明されています。

6. パッチクラスタに含まれたパッチをインストールする `install_cluster` スクリプトを実行します。

## パッケージの置換によるコンポーネントのアップグレード

多くの場合、Java ES 2005Q4 (Release 4) への共有コンポーネントのアップグレードでパッチテクノロジーは使用されません。代わりに、新規にパッケージをインストールするか、または既存のパッケージを新しいバージョンのパッケージで置換することで、これらのコンポーネントをアップグレードします。まれに、共有コンポーネントを正しくアップグレードするために、特別の手順を実行する場合があります。

- [62 ページの表 2-2](#) には、Java ES 2004Q2 (Release 2) からのアップグレードで Java ES Release 4 へのアップグレードにパッケージの置換を使用する共有コンポーネントが示されています。
- [60 ページの表 2-1](#) には、Java ES 2005Q1 (Release 3) からのアップグレードで Java ES Release 4 へのアップグレードにパッケージの置換を使用する共有コンポーネントが示されています。
- [73 ページの表 2-3](#) には、Solaris プラットフォームで必要となる可能性のある特別な手順へのリンクがあります。
- [77 ページの表 2-4](#) には、Linux プラットフォームで必要となる可能性のある特別な手順へのリンクがあります。

## パッケージ置換のアップグレード戦略

アップグレードする共有コンポーネントの数は、アップグレード計画およびコンピュータにインストールされている Java ES コンポーネントによって決まります。Java ES Release 2 からのアップグレードでは、すべての共有コンポーネントをアップグレードする必要があります。

Java ES Release 3 からアップグレードでは、すべてのコンポーネントをアップグレードする場合と、製品コンポーネントを Release 4 へ選択的にアップグレードする場合があります。アップグレード対象の製品コンポーネントのサポートに必要な Java ES 共有コンポーネントだけをアップグレードすることもできますが、コンピュータ上の共有コンポーネントをすべてアップグレードすることをお勧めします。詳細については、[44 ページの「共有コンポーネントのアップグレードのガイドライン」](#)を参照してください。

---

**注** 共有コンポーネントをアップグレードする順序は、アップグレード計画に反映させるべきコンポーネントの依存関係によって決まります。

ただし、J2SE から J2SE Release 5.0 にアップグレードする場合、この共有コンポーネントを最初にアップグレードしてください。J2SE はすべての Java ES コンポーネントのベースコンポーネントです。詳細については、[86 ページの「Java ES Release 4 のための J2SE のアップグレード」](#)を参照してください。

---

## アップグレードパス

通常、パッケージの置換による共有コンポーネントのアップグレードの順序は、アップグレードパスには依存しません。Java ES Release 2 からのアップグレードでは、Java ES Release 3 からのアップグレードと同じ順序に従います。

ただし、[72 ページの「Solaris プラットフォームのパッケージ」](#) および [76 ページの「Linux プラットフォームのパッケージ」](#) の各節の表に、使用する手順がアップグレードパスによって決定されるわずかなインスタンスを示しています。

## パッケージバージョン

パッケージの置換を行う場合、新しいバージョンが存在するパッケージだけを置換してください。アップグレード前にパッケージバージョンを比較する方法については、[「Solaris プラットフォームのパッケージ」](#) および [「Linux プラットフォームのパッケージ」](#) の各節に示されています。

## パッケージの場所

Java ES ディストリビューションで提供されるほとんどの共有コンポーネントをアップグレードするためのパッケージは、プラットフォームごとに次のディレクトリに格納されています。

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

Solaris プラットフォームの一部のパッケージには、オペレーティングシステム固有のバージョンがあります。それらのパッケージは、次のディレクトリに格納されています。

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

共有コンポーネントパッケージのローカライズ版は、次のディレクトリに格納されています。

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

## パッケージ置換の手順

わずかな例外はありますが、次の一般的な手順に従ってパッケージを置換できます。これらの手順については、次の各節で説明します。

- [75 ページの「Solaris プラットフォームのパッケージのアップグレード」](#)
- [79 ページの「Linux プラットフォームのパッケージのアップグレード」](#)

## 特別な手順

一部のパッケージでは、設定情報やその他のデータを保存しておくための特別な手順があります。特別な手順へのリンクは、[72 ページの「Solaris プラットフォームのパッケージ」](#) および [76 ページの「Linux プラットフォームのパッケージ」](#) の各節に記載されているパッケージ表にあります。

すべての特別な手順については、[80 ページの「特別なアップグレード手順を必要とするコンポーネント」](#) で詳しく説明します。

# Solaris プラットフォームのパッケージ

[表 2-3](#) に、パッケージの置換によってアップグレードされる共有コンポーネントの Solaris パッケージの一覧を示します。この表には、Java ES Release 4 で利用できるパッケージのバージョンも記載され、コンポーネントをアップグレードするための特別な手順へのリンクもあります。

共有コンポーネントごとに、インストールする順序でパッケージが示されています。

新しいバージョンのパッケージだけを置換してください。パッケージの置換を行う前に、システム上のパッケージのバージョンと、それを置換するために使用するパッケージのバージョンを比較してください。



インストールされているパッケージのバージョンを確認するには、冗長 (-v) オプションを指定して pkgparam コマンドを使用します。このコマンドは、パッケージバージョン、パッケージのリビジョン、および SUNW\_PRODVERS のバージョンを出力します。次に例を示します。

```
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** バージョン管理システムはパッケージごとに異なりますが、通常は数値が大きいほど新しいバージョンのパッケージであることを示しています。

表 2-3 に示す共有コンポーネントのフルネームは、29 ページの「[Release 4 の共有コンポーネント](#)」に記載されています。

表 2-3 Solaris プラットフォームの共有コンポーネントアップグレード用のパッケージバージョン

共有コンポーネント	パッケージ	バージョン リビジョン	SUNW_PRODVERS	特別な手順
ANT	SUNWant	11.11.0 2005.04.06.16.31.0 4	1.6.2	なし
ACL	SUNWaclg	8.1 2005.05.31.17.01.2 8	1.0.3	なし
BDB	SUNWbdb SUNWbdbj	4.2.52 1.0.3	4.2.52, REV=1.0.3	なし
CAC	SUNWcacaocfg SUNWcacao	1.1 15	1.1	81 ページの「共通エージェントコンテナのアップグレード」で説明されている特別な手順に従って、これらのパッケージをインストールまたは置換します。
IM-SDK	SUNWiimdv	6.1 2004.04.16.16.01.4 0	7.0	なし

表 2-3 Solaris プラットフォームの共有コンポーネントアップグレード用のパッケージバージョン ( 続き )

共有コンポーネント	パッケージ	バージョン リビジョン	SUNW_PRODVERS	特別な手順
J2SE™ SPARC 32 ビット	SUNWj5rt SUNWj5dev SUNWj5cfg SUNWj5man SUNWj5dmo SUNWj5jmp	1.5.0  2004.12.07.00.07	1.5.0_04/ 1.5.0_04-b05	86 ページの「 <a href="#">Java ES Release 4 のための J2SE のアップグレード</a> 」で説明されている手順に従って、J2SE™ プラットフォーム 5.0 (Java 2 プラットフォーム、Standard Edition) をインストールします。
J2SE™ SPARC 64 ビット	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	84 ページの「 <a href="#">JATO のアップグレード</a> 」で説明されている手順に従って、JATO パッケージを置換します。
JavaHelp™	SUNWjhrt SUNWjhdev	2.0  2004.11.23	2.0/FCS	Java ES Release 2 からアップグレードする場合は、これらのパッケージを置換します。これらのパッケージは Java ES Release 3 から変更されていません。
SEDC	SUNWexplj SUNWexplu SUNWexplo	4.3.1  2004.06.25.07.21	4.3.1 GA	86 ページの「 <a href="#">Sun Explorer Data Collector のアップグレード</a> 」で説明されている手順に従って、Sun Explorer をアップグレードします。
SJWC	SUNWmctag SUNWmconr SUNWmcon SUNWmcos SUNWmcosx	2.2.4  2005.05.09.14.06	SNAG Development	SJWC のアップグレードについては、85 ページの「 <a href="#">Sun Java Web Console のアップグレード</a> 」を参照してください。Solaris 10 プラットフォームでは、Java Enterprise System 版 Sun Java Web Console をアップグレードする必要はありません。

# Solaris プラットフォームのパッケージのアップグレード

Solaris プラットフォームでパッケージをアップグレードするための一般的な手順を、次に示します。

いくつかの共有コンポーネントには、これらの一般的な手順に加えて特別な手順が必要です。[73 ページの表 2-3](#) に、それらのコンポーネントに対する特別な手順へのリンクがあります。

---

**注** システムに変更を加える前に、最初にシステムをバックアップすることをお勧めします。

---

1. **root** としてログインするか、スーパーユーザーになります。

```
su -
```

2. アップグレードするコンポーネントに適用できる特別な手順について、[73 ページの表 2-3](#) をチェックします。

パッケージをアップグレードする前に、すべての特別な手順に従います。特別な手順が示されていない場合、次の手順に進みます。

3. **Java ES Release 4** ディストリビューションで、パッケージの格納場所に移動します。

パッケージは、プラットフォームごとに次のディレクトリに格納されています。

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

---

**注** **Java Enterprise System** のサブセットディストリビューションには、上記の場所に必要な共有コンポーネントがすべて格納されています。

---

4. **pkgrm** コマンドを使用して、アップグレード対象となるパッケージの現在のバージョンを削除します。

たとえば、**JATO** のパッケージを削除するには、次のコマンドを使用します。

```
pkgrm SUNWjatodmo SUNWjatodoc SUNWjato
```

パッケージ削除の詳細については、**pkgrm(1m)** マニュアルページを参照してください。

5. `pkgadd` コマンドを使用して、ディストリビューションからパッケージをインストールします。

たとえば、JATO のパッケージをインストールするには、次のコマンドを使用します。

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

パッケージインストールの詳細については、`pkgadd(1M)` マニュアルページを参照してください。

6. `pkgparam` コマンドおよび `pkginfo` コマンドを使用してパッケージが正しくインストールされたことを確認します。

`-v` オプションを指定した `pkgparam` を使用してバージョンを確認します。 `pkginfo` では、パッケージの詳細情報が得られます。

これらのコマンドの詳細については、`pkgparam(1)` マニュアルページおよび `pkginfo(1)` マニュアルページを参照してください。

## Linux プラットフォームのパッケージ

表 2-4 に、共有コンポーネントのアップグレードに使用される Linux RPM パッケージの一覧を示します。この表には、コンポーネントをアップグレードするための特別な手順へのリンクもあります。RPM パッケージのバージョンは、パッケージ名に埋め込まれています。

新しいバージョンのパッケージだけを置換してください。パッケージの置換を行う前に、システム上のパッケージのバージョンと、それを置換するために使用するパッケージのバージョンを比較してください。バージョン番号が大きいほど、パッケージのバージョンが新しいことを示しています。

RPM のネーミング規則では、パッケージのバージョンに関する情報が提供されます。異なる共有コンポーネントには、異なるバージョン管理情報が埋め込まれます。通常、ファイル名に埋め込まれる番号には、パッケージのバージョン番号とリビジョン番号が含まれます。

インストールされているパッケージのバージョンを確認するには、情報 (`-i`) オプションを指定して `rpm` クエリーコマンドを使用します。このコマンドによって、名前、バージョン、および説明などのパッケージ情報が表示されます。次に例を示します。

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

---

**注**                   バージョン管理システムはパッケージごとに異なりますが、通常は数値が大きいほど新しいバージョンのパッケージであることを示しています。

表 2-4 に示す共有コンポーネントのフルネームは、29 ページの「[Release 4 の共有コンポーネント](#)」に記載されています。

---

表 2-4 Linux プラットフォームの共有コンポーネントアップグレード用のパッケージ

共有コンポーネント	パッケージ	特別な手順
ANT	sun-ant-1.6.2-1.rpm	なし
ACL	sun-aclg-1.0.3-1.i386.rpm	なし
BDB	sun-berkeleydatabase-core-4.2.52-4.4.i386.rpm sun-berkeleydatabase-java-4.2.52-4.4.i386.rpm	なし
CAC	sun-cacao-1.1-15.i386.rpm sun-cacaocfg-1.1-15.i386.rpm sun-cacao-man-1.1-15.i386.rpm	81 ページの「共通エージェントコンテナのアップグレード」で説明されている特別な手順に従って、これらのパッケージをインストールまたは置換します。
ICU	sun-icu-3.2-1.i386.rpm	なし
IM-SDK	sun-im-dev-6.2.9.13.i386.rpm	なし
J2SE™	jdk-1_5_04-linux-i586.rpm	93 ページの「Linux プラットフォームの J2SE のアップグレード」で説明されている手順に従って、J2SE™ プラットフォーム 5.0 (Java 2 プラットフォーム、Standard Edition) をインストールします。
JATO	SUNWjato-2.1.5.i386.rpm SUNWjatodmo-2.1.5.i386.rpm SUNWjatodoc-2.1.5.i386.rpm	84 ページの「JATO のアップグレード」で説明されている手順に従って、JATO パッケージをインストールまたは置換します。
JavaHelp™	sun-javahelp-2.0-fcs.i586.rpm	85 ページの「Linux プラットフォームの JavaHelp のアップグレード」で説明されている手順に従って、これらのパッケージを置換します。
JavaMail	sun-javamail-1.3.2-34.i386.rpm	このパッケージのインストールは、Java ES Release 2 からアップグレードする場合のみ行います。このパッケージは Java ES Release 3 から変更されていません。
JSS	sun-jss-4.1-4.i386.rpm	Sun Cluster ソフトウェアのアップグレードに備えてセキュリティ関連の共有コンポーネントをアップグレードする場合、80 ページの「セキュリティコンポーネント (NSS、NSPR、JSS) のアップグレード」を参照してください。

表 2-4 Linux プラットフォームの共有コンポーネントアップグレード用のパッケージ (続き)

共有コンポーネント	パッケージ	特別な手順
KTSE	sun-ktsearch-1.3-2.noarch.rpm	Java ES Release 2 からアップグレードする場合は、このパッケージをインストールします。このパッケージは Java ES Release 3 から変更されていません。
LDAP J SDK	sun-ljdk-4.18-4.i386.rpm	Java ES Release 2 からアップグレードする場合は、このパッケージをインストールします。Java ES Release 3 からアップグレードする場合は、このパッケージを置換します。
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	Java ES Release 2 からアップグレードする場合は、これらのパッケージをインストールします。Java ES Release 3 からアップグレードする場合は、これらのパッケージを置換します。
NSPR	sun-nspr-4.5.2-4.i386.rpm sun-nspr-devel-4.5.2-4.i386.rpm	Sun Cluster ソフトウェアのアップグレードに備えてセキュリティ関連の共有コンポーネントをアップグレードする場合、 <a href="#">80 ページの「セキュリティコンポーネント (NSS、NSPR、JSS) のアップグレード」</a> を参照してください。
NSS	sun-nss-3.10.1-1.i386.rpm	Sun Cluster ソフトウェアのアップグレードに備えてセキュリティ関連の共有コンポーネントをアップグレードする場合、 <a href="#">80 ページの「セキュリティコンポーネント (NSS、NSPR、JSS) のアップグレード」</a> を参照してください。
SASL	sun-sasl-2.18-1.i386.rpm	なし
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	SJWC のアップグレードについては、 <a href="#">85 ページの「Sun Java Web Console のアップグレード」</a> を参照してください。 Solaris 10 プラットフォームでは、Java Enterprise System 版 Sun Java Web Console をアップグレードする必要はありません。

# Linux プラットフォームのパッケージのアップグレード

Linux プラットフォームでパッケージを更新するための一般的な手順を、次に示します。

いくつかの共有コンポーネントには、これらの一般的な手順に加えて特別な手順が必要です。77 ページの表 2-4 に、それらのコンポーネントに対する特別な手順へのリンクがあります。

---

**注** システムに変更を加える前に、最初にシステムをバックアップすることをお勧めします。

---

1. **root** としてログインするか、スーパーユーザーになります。

```
su -
```

2. コンポーネントごとに、これらのパッケージの更新に対する特別な手順について 77 ページの表 2-4 をチェックします。

パッケージを更新する前に、すべての特別な手順に従います。特別な手順が示されていない場合、次の手順に進みます。

3. Java ES Release 4 ディストリビューションで、必要な RPM パッケージの場所を探します。RPMs は、次のディレクトリに格納されています。

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

4. `rpm -U` コマンドを使用して、該当の RPM を更新します。

次に例を示します。

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

Linux rpm ユーティリティーは、インストールするパッケージを正しく順序付けます。

パッケージ更新の詳細については、rpm マニュアルページを参照してください。

## 特別なアップグレード手順を必要とするコンポーネント

この節では、特別な手順を必要とする共有コンポーネントをアップグレードするための手順を説明します。次の共有コンポーネントに関する特別な手順が含まれます。

- 80 ページの「セキュリティコンポーネント (NSS、NSPR、JSS) のアップグレード」
- 81 ページの「共通エージェントコンテナのアップグレード」
- 84 ページの「JATO のアップグレード」
- 85 ページの「Linux プラットフォームの JavaHelp のアップグレード」
- 85 ページの「Sun Java Web Console のアップグレード」
- 86 ページの「Sun Explorer Data Collector のアップグレード」
- 86 ページの「Java ES Release 4 のための J2SE のアップグレード」
- 93 ページの「Linux プラットフォームの J2SE のアップグレード」

### セキュリティコンポーネント (NSS、NSPR、JSS) のアップグレード

Sun Cluster ソフトウェアのアップグレードに備えてセキュリティ関連の共有コンポーネント (NSS、NSPR、および JSS) をアップグレードするには、『Sun Cluster Software Installation Guide for Solaris OS』の「Upgrading Sun Cluster Software」に説明されている依存関係を持つソフトウェアのアップグレード手順に従います。このマニュアルは次の場所で参照できます。

<http://docs.sun.com/app/docs/doc/817-4229/6mjp4pjuh?l=ja&a=view>

ただし、セキュリティコンポーネントは、Sun Cluster 1 of 2 CD-ROM または Sun Cluster 2 of 2 CD-ROM からではなく、Java ES Release 4 ディストリビューションからインストールしてください。



## 共通エージェントコンテナのアップグレード

Sun Cluster ソフトウェアのアップグレードに備えて共通エージェントコンテナの共有コンポーネントをアップグレードするには、『Sun Cluster Software Installation Guide for Solaris OS』の「Upgrading Sun Cluster Software」に説明されている依存関係を持つソフトウェアのアップグレード手順に従います。このマニュアルは次の場所で参照できます。

<http://docs.sun.com/app/docs/doc/817-4229/6mjp4pjuh?l=ja&a=view>

ただし、共通エージェントコンテナコンポーネントは、Sun Cluster 1 of 2 CD-ROM または Sun Cluster 2 of 2 CD-ROM からではなく、Java ES Release 4 ディストリビューションからインストールしてください。

次の各節で、スタンドアロンシステムで共通エージェントコンテナパッケージをアップグレードする手順を説明します。

### Solaris プラットフォームでの Java ES Release 2 からのアップグレード

75 ページの「Solaris プラットフォームのパッケージのアップグレード」の手順に従って、共通エージェントパッケージをインストールします。

### Linux プラットフォームでの Java ES Release 2 からのアップグレード

79 ページの「Linux プラットフォームのパッケージのアップグレード」の手順に従います。ただし、更新された Linux RPM パッケージを提供する Patch 120677-01 も適用する必要があります。このパッチは、次の場所にある SunSolve から入手できます。

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

### Java ES Release 3 からのアップグレード

次の特別な手順に従います。

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. 共通エージェントコンテナの共有コンポーネントをアップグレードする前に、共通エージェントコンテナが依存関係を持つ共有コンポーネントを確実に更新してください。

共通エージェントコンテナが依存関係を持つ共有コンポーネントは、Java 2 Platform Standard Edition (J2SE)、Java Dynamic Management Kit Runtime (JDMK)、Network Security Services (NSS)、および Netscape Portable Runtime (NSPR) です。これらの共有コンポーネントのアップグレードについては、60 ページの表 2-1 を参照してください。

J2SE を Version 5 に更新する場合、83 ページの [手順 8](#) に従って依存関係を更新する必要があります。

- 現在のインストールで、どのポートを使用するかなどのカスタム設定が使用されている場合、次のコマンドを使用して設定を確認します。

**Solaris** プラットフォームの場合

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

**Linux** プラットフォームの場合

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

出力内容は次のようになります。

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

上記の例は、デフォルト値を示しています。83 ページの [手順 7](#) で使用するためにデフォルト設定以外をすべてメモします。

- 次のコマンドを使用して、共通エージェントコンテナのプロセスを停止します。

**Solaris** プラットフォームの場合

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

終了コードが 0 以外の場合、次のように指定して強制的に停止します。

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

**Linux** プラットフォームの場合

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

終了コードが 0 以外の場合、次のように指定して強制的に停止します。

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

- これで、次に示すように、共通エージェントコンテナのパッケージをアップグレードできます。

**Solaris** プラットフォームでは、75 ページの「[Solaris プラットフォームのパッケージのアップグレード](#)」の手順に従って次のパッケージをアップグレードします。

```
SUNWcacaocfg
SUNWcacao
```

Linux プラットフォームでは、79 ページの「Linux プラットフォームのパッケージのアップグレード」の基本的な手順に従って次のパッケージをアップグレードします。ただし、次の手順に関する重要な変更点に注意してください。

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

rpm -U は共通エージェントコンテナ 1.1 ではサポートされていません。Linux プラットフォームでアップグレードするには、次のコマンドを使用します。

```
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
```

- Linux プラットフォームのみ、次に進む前に Patch 120677-01 を適用します。

このパッチは、次に進む前にインストールする必要がある更新済み Linux RPM パッケージを提供します。パッチとともに提供される手順に従ってください。Patch 120677-01 は、次の場所にある SunSolve Patch Access から入手できます。

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

- 82 ページの手順 3 で確認したカスタム設定をすべて適用します。

Solaris プラットフォームでは、次のコマンドを使用します。

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

Linux プラットフォームでは、次のコマンドを使用します。

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

- J2SE から J2SE Version 5 へのアップグレードでは、rebuild-dependencies ユーティリティを実行します。

Solaris プラットフォームの場合

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

Linux プラットフォームの場合

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

このコマンドの出力内容は、次のとおりです。

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

9. 共通エージェントコンテナのサービスを再起動します。

```
cacaoadm start
```

10. 共通エージェントコンテナがアップグレードされたことを確認します。

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

## JATO のアップグレード

Java ES Release 4 とともに提供される JATO パッケージのバージョンは、Sun Java Studio Enterprise に必要な更新を含んでいます。Sun Java Studio Enterprise を使用している場合、更新された JATO パッケージを必要とすることがあります。使用していない場合、Java ES Release 3 とともに提供される以前のバージョンの JATO をアップグレードする必要はありません。

### Solaris プラットフォームでの Java ES Release 3 からの JATO のアップグレード

Java ES Release 3 とともにインストールされた JATO のベースバージョンを、Java ES Release 4 ディストリビューションとともに提供される JATO パッケージで置換します。

この最新バージョンの JATO が必要となるのは、Sun Java Studio Enterprise を使用していて、この最新バージョンで提供される更新を必要とする場合だけです。

### Solaris プラットフォームでの Java ES Release 2 からの JATO のアップグレード

1. [67 ページの「パッチクラスタの手順」](#)の説明に従って、プラットフォームに対応するパッチクラスタを適用します。
2. パッチクラスタによってインストールされた JATO のベースバージョンを、Java ES Release 4 ディストリビューションとともに提供される JATO パッケージで置換します。

この最新バージョンの JATO が必要となるのは、Sun Java Studio Enterprise を使用していて、この最新バージョンで提供される更新を必要とする場合だけです。

## Linux プラットフォームでの Java ES Release 3 からの JATO のアップグレード

Java ES Release 3 とともにインストールされた JATO のベースバージョンを、Java ES Release 4 ディストリビューションとともに提供される JATO RPM パッケージで置換します。

この最新バージョンの JATO が必要となるのは、Sun Java Studio Enterprise を使用していて、この最新バージョンで提供される更新を必要とする場合だけです。

## Linux プラットフォームでの Java ES Release 2 からの JATO のアップグレード

Java ES Release 4 ディストリビューションとともに提供される JATO RPM パッケージをインストールします。

## Linux プラットフォームの JavaHelp のアップグレード

この特別の手順は、Linux プラットフォームでの JavaHelp のアップグレードのみを対象とします。Java ES Release 2 および Java ES Release 3 のどちらの JavaHelp をアップグレードする場合でも、この特別な手順を実行します。どちらのシナリオでも、Java ES Release 4 ディストリビューションとともに提供される RPM で JavaHelp を置換する必要があります。

JavaHelp RPM パッケージを置換するときに、rpm ユーティリティに `-U` (アップグレード) オプションを指定しないでください。その代わりに、次に示すように `-e` (消去) オプションに続けて `-i` (インストール) オプションを使用します。

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

## Sun Java Web Console のアップグレード

Solaris 8 および Solaris 9 の各プラットフォームで、Sun Cluster ソフトウェアのアップグレードに備えて Sun Java Web Console をアップグレードするには、『Sun Cluster Software Installation Guide for Solaris OS』の「Upgrading Sun Cluster Software」に説明されている依存関係を持つソフトウェアのアップグレード手順に従います。このマニュアルは次の場所で参照できます。

<http://docs.sun.com/app/docs/doc/817-4229/6mjp4pjuh?l=ja&a=view>

ただし、Sun Java Web Console は、Sun Cluster 1 of 2 CD-ROM または Sun Cluster 2 of 2 CD-ROM からではなく、Java ES Release 4 ディストリビューションからインストールしてください。

Sun Java Web Console のアップグレードパッケージは、共有コンポーネントパッケージとは別の場所に格納されています。Java ES ディストリビューションの次のディレクトリでパッケージを探してください。

<Architecture>/Product/shared\_components/Packages/<OperatingSystem>/

*Architecture* は Solaris\_sparc または Solaris\_x86 です。*OperatingSystem* は Solaris\_8 または Solaris\_9 です。

Solaris 10 プラットフォームでは、Sun Java Web Console のアップグレード手順はありません。Solaris 10 では、Sun Java Web Console はオペレーティングシステムの一部として提供されています。

## Sun Explorer Data Collector のアップグレード

Sun Cluster ソフトウェアは、Sun Explorer を必要とする唯一のコンポーネントです。

Sun Explorer ソフトウェアをアップグレードするには、『Sun Cluster Software Installation Guide for Solaris OS』の「Upgrading Sun Cluster Software」に説明されている依存関係を持つソフトウェアのアップグレード手順に従います。このマニュアルは次の場所で参照できます。

<http://docs.sun.com/app/docs/doc/817-4229/6mjp4pjuh?l=ja&a=view>

ただし、セキュリティーコンポーネントは、Sun Cluster 1 of 2 CD-ROM または Sun Cluster 2 of 2 CD-ROM からではなく、Java ES Release 4 ディストリビューションからインストールしてください。

## Java ES Release 4 のための J2SE のアップグレード

Java ES Release 4 は Java 2 Platform, Standard Edition (J2SE) Version 5.0 Update 4 で動作保証されています。ここでは、J2SE 5.0 Update 4 ともいいます。J2SE 5.0 は開発者バージョンの 1.5.0 を表すこともあります。次に示す点を除き、Java ES Release 4 は J2SE 1.4.2 および J2SE 5.0 Update 1 を引き続きサポートしています。

---

**注** Java ES Release 4 とともに配布されている High Availability Session Store (HADB) には J2SE Release 5 が必要です。HADB をアップグレードする場合、J2SE 5.0 にもアップグレードする必要があります。

---

Java ES Release 4 では、J2SE を Version 5.0 Update 4 にアップグレードし、J2SE 1.4.2 のインストールを残すことをお勧めします。

Java Enterprise System はコンピュータにインストールされているデフォルトの J2SE を使用しませんが、サポートされるバージョンの J2SE へのシンボリックリンクを保持します。J2SE のアップグレード後、アップグレードされた J2SE をポイントするように Java ES シンボリックリンクを設定する必要があります。

ただし、以前のバージョンを必要とするサービスのために、J2SE 1.4.2 へのポインタを保持してください。以前のバージョンの J2SE へのシンボリックリンクを保持する方法については、該当する製品コンポーネントのマニュアルを参照してください。

次の各節で、Solaris プラットフォームおよび Linux プラットフォームで J2SE をアップグレードする手順を説明します。

- [87 ページの「Solaris プラットフォームの J2SE のアップグレード」](#)
- [93 ページの「Linux プラットフォームの J2SE のアップグレード」](#)

## Solaris プラットフォームの J2SE のアップグレード

J2SE のアップグレードに使用する手順は、J2SE 1.4 からアップグレードするか、または J2SE 5.0 の Update 4 バージョンよりも前のバージョンからアップグレードするかによって異なります。

そのため、Java ES の現在のインストールが使用する J2SE のバージョンを確認してください。デフォルトのバージョンは、次のとおりです。

- Java ES Release 2。デフォルトは J2SE 1.4.2
- Java ES Release 3。デフォルトは J2SE Version 5.0 Update 1

さまざまな理由により、J2SE をデフォルトバージョンからアップグレードしている可能性があります。Java ES の現在のインストールが使用している J2SE のバージョンを確認するには、次のコマンドを実行します。

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

- J2SE 1.4.2 Update 5 のバージョンを表す文字列の例を次に示します。  

```
java version "1.4.2_05"
```
- J2SE 5.0 Update 1 のバージョンを表す文字列の例を次に示します。  

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

## Solaris プラットフォームの一般的な手順

Solaris プラットフォームで J2SE をアップグレードするための一般的な手順は、どのバージョンの J2SE から開始するかによって異なります。

- J2SE 1.4 からアップグレードする場合は、次の手順に従います。
  - a. J2SE 5.0 Update 4 以降をインストールします (89 ページの「Solaris プラットフォームの J2SE 5.0 のインストール」を参照)。
  - b. 新しくインストールした J2SE をポイントするように Java ES シンボリックリンクを設定します (91 ページの「Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定」を参照)。

---

**注** J2SE 1.4 を削除する必要はありません。同じコンピュータに両方のバージョンをインストールできます。

J2SE の以前のバージョンで、一部の Java ES サービスの実行を継続するように選択できます。このためには、適切なコンポーネント製品の管理ガイドを参照する必要があります。たとえば、以前のバージョンの J2SE をポイントするように、Application Server が使用する J2SE シンボリックリンクを変更できます。

---

- Update 4 バージョンの J2SE 5.0 より前のバージョンからアップグレードする場合は、次の手順に従います。
  - a. 以前のバージョンの J2SE 5.0 を削除します。
  - b. J2SE 5.0 Update 4 以降をインストールします (89 ページの「Solaris プラットフォームの J2SE 5.0 のインストール」を参照)。
  - c. 新しくインストールした J2SE をポイントするように Java ES シンボリックリンクを設定します (91 ページの「Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定」を参照)。または
  - a. パッチを適用して現在のバージョンの J2SE をアップグレードします (91 ページの「パッチの適用による Solaris プラットフォームの J2SE 5.0 のインストール」を参照)。
  - b. 更新した J2SE をポイントするように Java ES シンボリックリンクを設定します (91 ページの「Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定」を参照)。



J2SE のアップグレード時には、手順を進める前に、現在インストールされている J2SE に依存するサービスをすべて停止することをお勧めします。これは、現在の J2SE を使用しているサービスによって生じる問題を回避するためです。J2SE に依存するサービスを停止しない場合、J2SE のインストールと、インストールした J2SE への Java ES シンボリックリンクの設定後、システムを再起動する必要があります。

### Solaris プラットフォームの J2SE 5.0 のインストール

次のソースのいずれかから入手できるソフトウェアを使用して、Solaris プラットフォームに J2SE 5.0 をインストールできます。

- Sun Developer Network
- Java ES ディストリビューション

---

**注** この節で説明する手順では、J2SE 5.0 はデフォルトの場所にインストールされます。デフォルト以外の場所に J2SE をインストールする場合、次の場所で参照できる Sun Developer Network の手順に従います。

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

---

Sun Developer Network から J2SE 5.0 をインストールする

1. Sun Developer Network で次の場所へ移動して、現在のバージョンの JDK 5.0 とインストール手順を取得します。  
<http://java.sun.com/j2se/1.5.0/download.jsp>
2. ダウンロードとともに入手できるインストールの手順に従います。  
88 ページの「Solaris プラットフォームの一般的な手順」に説明されているように、J2SE をインストールする前に J2SE に依存するサービスを停止する場合があります。
3. インストールの完了後、91 ページの「Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定」の手順に進みます。

Java ES ディストリビューションから J2SE 5.0 をインストールする

1. root としてログインするか、スーパーユーザーになります。  
su -
2. (省略可能) 88 ページの「Solaris プラットフォームの一般的な手順」の説明に従って、Java ES サービスを停止します。
3. 以前のバージョンの J2SE 5.0 がインストールされている場合、次のように削除します。  
次のようにパッケージを削除します。

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

64 ビットプロセッサを搭載するコンピュータでは、次のように追加パッケージを削除します。

```
pkgrm SUNWj5rtx SUNWj5dvx SUNWj5dmx
```

4. **Java ES Release 4** ディストリビューションで、**J2SE** パッケージの格納場所に移動します。

パッケージは、プラットフォームごとに次のディレクトリに格納されています。

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

---

**注** **Java Enterprise System** のサブセットディストリビューションには、**J2SE** のインストールに必要なパッケージが格納されています。

---

5. **pkgadd** コマンドを使用して **J2SE** パッケージをインストールします。

次のパッケージをインストールします。

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

64 ビットプロセッサを搭載するコンピュータでは、次の追加パッケージをインストールします。

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

**J2SE 5.0 Update 4** は `/usr/jdk/jdk1.5.0_04` にインストールされます。バージョン 5.0 は、自動的に **Solaris 8** または **Solaris 9** のデフォルトの **Java** プラットフォームにはなりません (その他のデフォルトのプラットフォームがない場合以外) が、**Solaris 10** ではデフォルトになります。

---

**注** **Solaris 8** および **Solaris 9** では、`/usr/jdk/jdk1.5.0_04` をポイントするように `/usr/java` シンボリックリンクを変更することで、**J2SE 5.0** をデフォルトの **Java** プラットフォームに指定できます。

ただし、この方法でシンボリックリンクを変更すると、**J2SE 5.0** でテストされていない以前の **Java** アプリケーションの一部に問題を生じるものがある可能性があります。詳細については、**J2SE** のインストールの注意事項を参照してください。

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

---

6. (省略可能) 日本語のマニュアルページをインストールします。

**pkgadd** コマンドを使用して、最新の日本語マニュアルページのパッケージをインストールします。

```
pkgadd -d . SUNWj5jmp
```

7. 次の 91 ページの「Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定」に進みます。

### パッチの適用による Solaris プラットフォームの J2SE 5.0 のインストール

次の手順は、インストールされているバージョンの J2SE 5.0 プラットフォームを、サポートされるバージョンにアップグレードする方法を示します。

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. 次の表に示される Solaris プラットフォームに必要なパッチを取得します。

プラットフォーム	パッチ
SPARC	118666-03 以上
SPARC 64 ビット	118667-03 以上
x86	118668-03 以上
x86 64 ビット	118669-03 以上

パッチ ID の下 2 桁は、パッチのリビジョン番号です。リビジョン番号が大きいほど、パッチのバージョンが新しいことを示しています。各パッチの特別な手順については、README ファイルを参照してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

3. (省略可能) 88 ページの「Solaris プラットフォームの一般的な手順」の説明に従って、Java ES サービスを停止します。
4. patchadd コマンドを使用してパッチを適用します。  
たとえば、Solaris SPARC プラットフォームでは、次のようにします。  

```
patchadd 118666-03
```
5. 次の「Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定」に進みます。

### Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定

Java Enterprise System は、サポートされるバージョンの J2SE プラットフォームをポイントするシンボリックリンクを保持します。Java Enterprise System がこのリンクを保持することで、使用する正しい J2SE ランタイムを Java ES サービスが確実に見つけることができます。

J2SE 1.4.2 から J2SE 5.0 Update 4 にアップグレードした場合、新しくインストールした J2SE 5.0 をポイントするようにシンボリックリンクを設定する必要があります。以前のバージョンの J2SE 5.0 から J2SE 5.0 Update 4 にアップグレードした場合、Java ES のインストールが更新されたバージョンを使用していることを確認する必要があります。

次の手順は、アップグレードされた J2SE のインストールに Java ES シンボリックリンクを設定する方法を示しています。

1. 次に示すように、新しくインストールされた J2SE または更新された J2SE のインストールをポイントするように、`/usr/jdk/entsys-j2se` シンボリックリンクをリセットします。

デフォルトの場所に J2SE 5.0 Update 4 をインストールした場合、シンボリックリンクを次のようにリセットします。

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

デフォルト以外の場所に J2SE 5.0 をインストールした場合、デフォルトのパス (`/usr/jdk/instances/jdk1.5.0`) をデフォルト以外の場所へのパスで置き換えます。

2. J2SE 5.0 Update 4 のアップグレードまたはインストールを行う前にサービスを停止した場合、サービスを再起動します。

J2SE 5.0 のアップグレードまたはインストールを行う前にサービスを停止しなかった場合、J2SE 5.0 に依存するサービスが新しいシンボリックリンクを使用するようシステムを再起動することをお勧めします。

## J2SE のアップグレードの確認

次のコマンドによって、J2SE シンボリックリンクが参照する J2SE のバージョンを確認します。

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

このコマンドによって、開発者バージョンの番号を含む文字列が返されます。たとえば、J2SE 5.0 Update 4 をインストールした場合、このコマンドは次の文字列を返します。

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

上記のコマンドが正しいバージョンを返さない場合、「[Solaris プラットフォームの Java ES のための J2SE シンボリックリンクの設定](#)」の説明に従って、J2SE への Java ES シンボリックリンクが正しく設定されていることを確認します。

## Linux プラットフォームの J2SE のアップグレード

Linux で J2SE のアップグレードに使用する手順は、アップグレードパスに依存しません。J2SE 1.4 からのアップグレードでも、以前のバージョンの J2SE 5.0 かのアップグレードでも、同じ手順を使用します。Linux プラットフォームでは、複数のバージョンの J2SE 5.0 を共存させることができます。

Linux プラットフォームで J2SE をアップグレードするための一般的な手順は、次のとおりです。

1. J2SE 5.0 Update 4 以降をインストールします (93 ページの「Linux プラットフォームの J2SE 5.0 のインストール」を参照)。
2. 新しくインストールした J2SE をポイントするように Java ES シンボリックリンクを設定します (94 ページの「Linux プラットフォームの Java ES のための J2SE シンボリックリンクの設定」を参照)。

---

**注** 以前のバージョンの J2SE の削除はオプションです。その他のサービスが以前のバージョンに依存する場合、以前のバージョンのインストールを残すことをお勧めします。

---

## Linux プラットフォームの J2SE 5.0 のインストール

次のソースのいずれかから入手できるソフトウェアを使用して、Solaris プラットフォームに J2SE 5.0 をインストールできます。

- Sun Developer Network
- Java ES ディストリビューション

---

**注** この節で説明する手順では、J2SE 5.0 はデフォルトの場所にインストールされます。デフォルト以外の場所に J2SE をインストールする場合、次の場所で参照できる Sun Developer Network の手順に従います。

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

---

Sun Developer Network から J2SE 5.0 をインストールする

1. Sun Developer Network で次の場所に移動して、現在のバージョンの JDK 5.0 とインストール手順を取得します。

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

2. ダウンロードとともに入手できるインストールの手順に従います。

88 ページの「Solaris プラットフォームの一般的な手順」に説明されているように、J2SE をインストールする前に、J2SE に依存するサービスを停止する場合があります。

3. インストールの完了後、94 ページの「Linux プラットフォームの Java ES のための J2SE シンボリックリンクの設定」の手順に進みます。

Java ES ディストリビューションから J2SE 5.0 をインストールする

1. root としてログインするか、スーパーユーザーになります。  
`su -`
2. (省略可能) 88 ページの「Solaris プラットフォームの一般的な手順」の説明に従って、Java ES サービスを停止します。
3. Java ES ディストリビューションで、jdk-1\_5\_0\_04-linux-i586.rpm ファイルを含む次のディレクトリに移動します。

`Linux_x86/Product/shared_components/Packages/`

---

**注** Java Enterprise System のサブセットディストリビューションには、J2SE のインストールに必要なパッケージが格納されています。

---

4. 次のコマンドを使用して RPM パッケージをインストールします。

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

以前のバージョンの J2SE の削除はオプションです。その他のサービスが以前のバージョンに依存する場合、以前のバージョンのインストールを残すことをお勧めします。

5. 次の節で説明する Java ES シンボリックリンクの設定手順に進みます。

### Linux プラットフォームの Java ES のための J2SE シンボリックリンクの設定

Java Enterprise System は、サポートされるバージョンの J2SE プラットフォームをポイントするシンボリックリンクを保持します。Java Enterprise System がこのリンクを保持することで、使用する正しい J2SE ランタイムを Java ES サービスが確実に見つけることができます。

次の手順は、Java ES シンボリックリンクを設定する方法を示しています。

1. 次に示すように、新しくインストールされた J2SE または更新された J2SE のインストールをポイントするように、`/usr/jdk/entsys-j2se` シンボリックリンクをリセットします。

デフォルトの場所に J2SE 5 Update 4 をインストールした場合、シンボリックリンクを次のようにリセットします。

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

これらのコマンドによって J2SE 5.0 Update 4 のパスが変更されます。システム上のバージョンに基づいて J2SE プラットフォームへのパスを更新します。

デフォルト以外の場所に J2SE 5.0 をインストールした場合、デフォルトのパス (/usr/java/jdk1.5.0\_04) をデフォルト以外の場所へのパスで置換します。

2. J2SE 5.0 Update 4 のアップグレードまたはインストールを行う前にサービスを停止した場合、サービスを再起動します。

J2SE 5.0 のアップグレードまたはインストールを行う前にサービスを停止しなかった場合、J2SE 5.0 に依存するサービスが新しいシンボリックリンクを使用するようシステムを再起動することをお勧めします。

### J2SE のアップグレードの確認

次のコマンドによって、J2SE シンボリックリンクが参照する J2SE のバージョンを確認します。

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

このコマンドによって、開発者バージョンの番号を含む文字列が返されます。たとえば、J2SE 5.0 Update 4 をインストールした場合、このコマンドは次の文字列を返します。

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

上記のコマンドが正しいバージョンを返さない場合、「[Linux プラットフォームの Java ES のための J2SE シンボリックリンクの設定](#)」の説明に従って、J2SE への Java ES シンボリックリンクが正しく設定されていることを確認します。

特別なアップグレード手順を必要とするコンポーネント



# Sun Cluster ソフトウェア

この章では、Sun Cluster ソフトウェアを Java ES 2005Q4 (Release 4) の Sun Cluster 3.1 8/05 にアップグレードする方法について説明します。

ここでは、アップグレードに関する一般的な概要と、Sun Cluster ソフトウェアの Java ES Release 4 へのアップグレード手順について説明します。

Sun Cluster ソフトウェアは Solaris プラットフォームだけでサポートされます。

この章で説明する Sun Cluster ソフトウェアのアップグレードでは、Sun Cluster フレームワークソフトウェアおよび Sun Cluster データサービスソフトウェアつまりエージェントの両方を含みます。

- [97 ページの「Sun Cluster ソフトウェアアップグレードの概要」](#)
- [100 ページの「Sun Cluster ソフトウェアの Java ES Release 4 へのアップグレード」](#)

## Sun Cluster ソフトウェアアップグレードの概要

この節では、Java ES 2005Q4 (Release 4) へのアップグレードに影響を与える次の Sun Cluster ソフトウェアの一般的な側面について説明します。

- [Java ES Release 4 Sun Cluster ソフトウェアについて](#)
- [Sun Cluster ソフトウェアアップグレードのロードマップ](#)
- [Sun Cluster のデータ](#)
- [互換性の問題](#)
- [Sun Cluster の依存関係](#)

## Java ES Release 4 Sun Cluster ソフトウェアについて

Java ES Release 4 Sun Cluster ソフトウェアには、クラスタのインストールとアップグレードの拡張機能、Network Appliance NAS デバイスの拡張サポート、SunPlex Manager の簡略化されたインタフェース、および『Sun Cluster Release Notes』(<http://docs.sun.com/doc/819-1405/6n3p13hac?a=view>) に詳しく説明されているその他の機能など、多くの新機能が搭載されています。

## Sun Cluster ソフトウェアアップグレードのロードマップ

表 3-1 に、Java ES Release 4 への Sun Cluster のサポートされるアップグレードパスを示します。この表は、Solaris オペレーティングシステムだけに適用されます。

Sun Cluster のバージョンは、Java ES のリリースと 1 対 1 に対応していません。これは、Sun Cluster ソフトウェアの暫定機能リリース (Interim Feature Releases、IFR) が Java ES の正式リリースの間に Java ES に組み込まれたためです。このため、表 3-1 に示される Java ES Release 3 Sun Cluster および Java ES Release 2 Sun Cluster から Java ES Release 4 Sun Cluster へのアップグレードには、Sun Cluster 3.1 4/04 および Sun Cluster 3.1 9/04 から Java ES Release 4 へのアップグレードが含まれます。

表 3-1 Java ES Release 4 Sun Cluster 3.1 8/05 (2005Q4) ソフトウェアへのアップグレードパス

Java ES のリリース	Sun Cluster ソフトウェアバージョン	一般的な手順	必要な再設定
Release 3	Sun Cluster 3.1 9/04 または Sun Cluster 3.1 8/05	直接アップグレード: Sun Cluster scinstall ユーティリティを使用して実行されます。	クラスタ構成は、アップグレードされたバージョンに自動的に移行されます。
Release 2	Sun Cluster 3.1 4/04 または Sun Cluster 3.1 9/04	直接アップグレード: Sun Cluster scinstall ユーティリティを使用して実行されます。	クラスタ構成は、アップグレードされたバージョンに自動的に移行されます。
Release 1	Sun Cluster 3.1	直接アップグレードは動作保証なし: ただし、scinstall ユーティリティを使用することにより実行できます。	クラスタ構成は、アップグレードされたバージョンに自動的に移行されます。
それ以前の Java ES リリース	Sun Cluster 3.0	直接アップグレードは動作保証なし: ただし、scinstall ユーティリティを使用することにより実行できます。	クラスタ構成は、アップグレードされたバージョンに自動的に移行されます。

## Sun Cluster のデータ

次の表は、Sun Cluster ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 3-2 Sun Cluster のデータの使用方法

データの種類	場所	使用方法
クラスタ設定データ	クラスタ構成リポジトリ。すべてのクラスタノードに渡って複製および同期されます。注意: CCR ファイルを手動で編集しないでください。	ノードやクラスタ全体の動作を停止させてしまう可能性があります。クラスタノードの設定、フェイルオーバー機構、リソース管理など、Sun Cluster 操作のすべての局面に対する設定情報を保存します。

## 互換性の問題

Java ES Release 4 Sun Cluster ソフトウェアには新しいグラフィカル管理インタフェースが含まれます。ただし、以前のリリースの Sun Cluster エージェントと後方互換性があります。

## Sun Cluster の依存関係

ほかの Java ES コンポーネントに対する Sun Cluster の依存関係が、Sun Cluster ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Sun Cluster のインタフェースや機能の変更が原因で、Sun Cluster ソフトウェアが依存するコンポーネントのバージョンのアップグレードが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Sun Cluster は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Sun Cluster ソフトウェアは、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **データサービス** : Sun Cluster ソフトウェアには、Java ES 製品コンポーネントの可用性を高めるために、特定のデータサービス (またはエージェント) が必要です。Sun Cluster 環境で実行している製品コンポーネントごとに、対応するクラスタリソースを管理するための対応するデータサービスが必要です。通常、エージェントパッケージは Sun Cluster アップグレードプロセスの一部としてアップグレードされます。

# Sun Cluster ソフトウェアの Java ES Release 4 へのアップグレード

この節では、Java ES 2005Q1 (Release 3) および Java ES 2004Q2 (Release 2) の両方から Java ES Release 4 への Sun Cluster ソフトウェアのアップグレードに関する情報を提供します。これらの Java ES リリースである Sun Cluster 3.1 4/04 および Sun Cluster 3.1 9/04 のソフトウェアに含まれる 2 つの Sun Cluster バージョンで、アップグレードの手順は同じです。

この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Sun Cluster のアップグレード](#)

## はじめに

Sun Cluster ソフトウェアを Java ES Release 4 にアップグレードするときには、アップグレードプロセスに関する次の事項を検討します。

- **一般的なアップグレード手順** : アップグレードは、scinstall スクリプトを実行して行われます。このスクリプトは、Sun Cluster ソフトウェアをアップグレードし、ソフトウェアのアップグレード後に Sun Cluster の以前の設定を適用します。ただし、クラスタ環境内のすべてのノードを同じバージョンにアップグレードする必要があります。これは、クラスタを停止し、すべてのノードをアップグレードする方法、またはクラスタを停止せずに、ノードを 1 つずつ連続してアップグレードする順次アップグレードによって行います。
- **アップグレードの依存関係** : Sun Cluster ソフトウェアには多数の Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Java ES Release 4 Sun Cluster ソフトウェアには、これらのコンポーネントの Release 3 バージョンと互換性があります。したがって、Sun Cluster ソフトウェアを Release 4 にアップグレードする場合、これらの共有コンポーネントのアップグレードは省略可能です。
- **下位互換性** : Release 4 Sun Cluster ソフトウェアには、以前のクラスタエージェントに対する下位互換性があります。ただし、クラスタ内のすべてのノードでは、同じバージョンのフレームワークとエージェントソフトウェアを実行する必要があります。
- **アップグレードのロールバック** : Sun Cluster ソフトウェアの Release 4 へのアップグレードを以前のバージョンにロールバックすることはサポートされていません。
- **プラットフォームの問題** : Sun Cluster ソフトウェアのアップグレードアプローチは、すべての Solaris プラットフォームで同じです。ただし、Sun Cluster ソフトウェアは Linux プラットフォームでサポートされていません。

## Sun Cluster のアップグレード

この節では、Sun Cluster ソフトウェアを Java ES Release 3 から Java ES Release 4 にアップグレードする方法の概要を示します。

- アップグレード前の作業
- Sun Cluster ソフトウェアのアップグレード
- アップグレードの確認
- アップグレード後の作業
- アップグレードのロールバック

ここでは、Sun Cluster の非順次アップグレードについて説明します。順次アップグレードでは、クラスタを停止しないという点で異なります。ただし、次に説明するように、どちらの場合も1つのクラスタノードに対する一般的な手順は同じです。個別の手順については、『Sun Cluster Installation Guide』(<http://docs.sun.com/doc/819-0420/6n2rlmncr?a=view>) のアップグレードに関する章で説明されています。

### アップグレード前の作業

Sun Cluster ソフトウェアをアップグレードする前に、次に説明される作業を実行してください。

#### 現在のバージョン情報の確認

次のコマンドを入力することにより、Sun Cluster ソフトウェアの現在のバージョンを確認できます。

```
% scinstall -pv
```

このコマンドは、Sun Cluster のバージョンと、インストールされている各ソフトウェアパッケージのバージョンを返します。このコマンドによって 3.1 8/05 のバージョンである 3.1u4 が返された場合、Java ES Release 4 へのアップグレードは不要です。

表 3-3 Sun Cluster のバージョン確認の出力

Java ES のリリース	Sun Cluster のバージョン番号
Release 1 (Sun Cluster 3.1)	3.1
Release 2 (Sun Cluster 3.1 4/04)	3.1u2
Release 2 または Release 3 (Sun Cluster 3.1 9/04)	3.1u3
Release 3 または Release 4 (Sun Cluster 3.1 8/05)	3.1u4

## アップグレードのためのクラスタノードの準備

Sun Cluster ソフトウェアをアップグレードする前に、クラスタノードをクラスタ環境から削除する必要があります。

- **非順次アップグレード**: クラスタ環境からノードを削除することは、環境を停止することを意味します。つまり、リソースグループのオフライン化と無効化、環境内で実行中のアプリケーションの停止、共有データのバックアップ、クラスタの停止、システムディスクのバックアップ、および非クラスタモードでのノードの再起動を行います。
- **順次アップグレード**: クラスタ環境からノードを削除することは、ノードからのすべてのリソースグループとデバイスグループの移動、共有データとシステムディスクのバックアップ、および非クラスタモードでのノードの再起動を意味します。

これらの手順の詳細と、特定の状況で必要なその他の手順については、『Sun Cluster Installation Guide』 (<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>) のアップグレードに関する章で詳しく説明されています。

## オペレーティングシステムのアップグレード

オペレーティングシステムの最新バージョンへのアップグレードや、使用しているボリュームマネージャーのバージョンのアップグレードに、アップグレードによるダウンタイムを利用することができます。

それらの手順については、『Sun Cluster Installation Guide』 (<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>) のアップグレードに関する章で詳しく説明されています。

## Sun Cluster が依存するコンポーネントのアップグレード

通常は、コンピュータシステム上 (およびコンピューティング環境内) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。Sun Cluster ソフトウェアが依存する Release 3 の共有コンポーネントのアップグレードは任意ですが、Release 2 共有コンポーネントの Release 4 へのアップグレードは必須です。

最低限のバージョン要件が満たされる場合でも共有コンポーネントをすべてアップグレードする場合を除き、Sun Cluster ソフトウェアが依存するすべての共有コンポーネント (42 ページの表 1-6 を参照) をアップグレードするには、『Sun Cluster Installation Guide』 (<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>) のアップグレードに関する章の説明に従ってください。

## 必要な設定情報とパスワードの入手

現在インストールされているバージョンに関する特別な情報はありません。ただし、アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。

## Sun Cluster ソフトウェアのアップグレード

ここでは、Sun Cluster ソフトウェアのアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点

Sun Cluster ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Sun Cluster フレームワークソフトウェアのアップグレード時には、高可用性の Java ES コンポーネントやクラスタ環境で実行するその他のアプリケーションを管理するために必要なデータサービスをアップグレードすることをお勧めします。
- Sun Cluster ソフトウェアのアップグレードにより、Java ES コンポーネントやクラスタ環境で実行するその他のアプリケーションをアップグレードする機会も得られます。

### アップグレード手順

次の手順は、各クラスタノードでの Sun Cluster ソフトウェアのアップグレードに適用されます。これらの手順はごく一般的です。実行方法については、『Sun Cluster Installation Guide』 (<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>) のアップグレードに関する章で詳しく説明されています。

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. Java ES ディストリビューションで次のディレクトリに移動します。

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

*arch* は sparc または x86、*ver* は Solaris 8、Solaris 9、および Solaris 10 に対してそれぞれ 8、9、または 10 です。

3. scinstall ユーティリティを実行します。

```
./scinstall
```

クラスタのインストール、設定、およびアップグレードを行うためのメインメニューが表示されます。

4. Sun Cluster フレームワークソフトウェアと必要なデータサービスをアップグレードします。

対応するリソースを、アップグレードされたリソースに移行することによって、アップグレードされたデータサービスを設定する必要があります (104 ページの「アップグレード後の作業」を参照)。

5. 必要なパッチをすべて Sun Cluster フレームワークソフトウェアとデータサービスに適用します。

関連するパッチへのアクセスと適用については、『Sun Cluster Release Notes』(<http://docs.sun.com/doc/819-1405>) に説明されています。

6. クラスタでノードを再起動します。

## アップグレードの確認

Sun Cluster ソフトウェアが正常にアップグレードされたかどうかは、次の手順に従って確認できます。

1. Sun Cluster フレームワークソフトウェアのバージョン番号をチェックします。

```
scinstall -pv
```

出力値については、[101 ページの表 3-3](#) を参照してください。

2. データサービスのアップグレードログファイルをチェックします。

ログファイルは、アップグレード出力メッセージの末尾に記載されています。

## アップグレード後の作業

Sun Cluster ソフトウェアのアップグレード後、多くの作業を実行することが必要な場合があります。これは、非順次アップグレードと順次アップグレードのどちらを行ったかによって異なります。クラスタ環境を完全に復元するために必要なタスクには、次のようなものがあります。

- クラスタ構成の状態の確認。
- 新しいリソースタイプバージョンへのリソースの移行。
- クラスタにインストールされているその他の Java ES コンポーネントやアプリケーションのアップグレード。

インストール後の手順については、『Sun Cluster Installation Guide』

(<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>) のアップグレードに関する章で詳しく説明されています。

## アップグレードのロールバック

Sun Cluster ソフトウェアのロールバックはサポートされていません。アップグレード手順の中で加えた変更を簡単にバックアウトすることはできません。



# Directory Server と Administration Server

この章では、Directory Server コンポーネントと Administration Server コンポーネントを、Java ES 2005Q4 (Release 4): Sun Java System Directory Server 5.2 2005Q4 および Sun Java System Administration Server 5.2 2005Q4 にアップグレードする方法について説明します。

これらのアップグレードは互いに密接に関係することから、まとめて説明しています。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [106 ページの「Directory Server と Administration Server のアップグレードの概要」](#)
- [109 ページの「Java ES Release 3 からの Directory Server と Administration Server のアップグレード」](#)
- [125 ページの「Java ES Release 2 からの Directory Server と Administration Server のアップグレード」](#)

---

## 注

この章で説明するファイルの場所は、*serverRoot* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Directory Proxy Server が最初にインストールおよび設定されたときにインストールディレクトリとして指定されている場合があります。指定されていない場合、デフォルト値が割り当てられています。

*serverRoot* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

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

# Directory Server と Administration Server のアップグレードの概要

この節では、Java ES 2005Q4 (Release 4) へのアップグレードに影響する、Directory Server コンポーネントおよび Administration Server コンポーネントの一般的な側面について説明します。

- [Java ES Release 4 について](#)
- [Java ES Release 4 へのアップグレードのロードマップ](#)
- [Directory Server と Administration Server のデータ](#)
- [互換性の問題](#)
- [依存関係](#)

## Java ES Release 4 について

Directory Server および Administration Server の Java ES Release 4 バージョンでは、若干のバグ修正と機能の改善だけが行われています。新しい機能の追加はありません。

## Java ES Release 4 へのアップグレードのロードマップ

表 4-1 は、Directory Server と Administration Server について、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 4-1 Java ES Release 4: Sun Java System Directory Server 5.2 2005Q4 および Sun Java System Administration Server 5.2 2005Q4 へのアップグレードパス

Java ES のリリース	Directory Server、Administration Server、および Directory Proxy Server のバージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Directory Server 5 2005Q1  Sun Java System Administration Server 5 2005Q1	直接アップグレード: パッチを適用し、設定ディレクトリを再設定します。	設定ディレクトリ内のデータの自動再設定

表 4-1 Java ES Release 4: Sun Java System Directory Server 5.2 2005Q4 および Sun Java System Administration Server 5.2 2005Q4 へのアップグレードパス (続き)

Java ES のリリース	Directory Server、Administration Server、および Directory Proxy Server のバージョン	一般的な手順	必要な再設定
Release 2	Sun Java System Directory Server 5.2 2004Q2 Sun Java System Administration Server 5.2 2004Q2	直接アップグレード: パッチを適用し、設定ディレクトリを再設定します。	設定ディレクトリ内のデータの自動再設定
Release 1	Sun One Directory Server 5.2 Sun One Administration Server 5.2	直接アップグレードは動作保証なし: ただし、Release 2 からのアップグレードと同じ手順を使用できます。	設定ディレクトリ内のデータの自動再設定
それ以前の Java ES リリース	Sun One Directory Server 5.2 Sun One Administration Server 5.2	直接アップグレードは動作保証なし: ただし、Release 2 からのアップグレードと同じ手順を使用できます。	設定ディレクトリ内のデータの自動再設定
	Sun One Directory Server 5.1、5.0、または 4.x Sun One Administration Server 5.1、5.0、または 4.x	直接アップグレード不可: まず Release 3 にアップグレードします。手順については、『Sun Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。  次に、Release 3 から Release 4 にアップグレードします。	手順については、『Sun Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。

## Directory Server と Administration Server のデータ

Directory Server と Administration Server は、設定データの格納に Directory Server を利用します。データはディレクトリ内部の特定のツリー構造に格納されます。設定を格納している Directory Server インスタンスのことを設定ディレクトリと呼びます。

設定ディレクトリは専用の Directory Server インスタンスにすることができます (セキュリティの観点から推奨)。また、ユーザーアイデンティティデータやサービス設定データを格納することもできます。設定ディレクトリは、ほかの Directory Server インスタンスまたは Administration Server と同様、同じコンピュータ上に配置できます。ただし、ほとんどの配備アーキテクチャーでは、設定ディレクトリを使用して設定情報を格納するほかのコンポーネントとは別のコンピュータに設定ディレクトリを配置します。

次の表は、Directory Server および Administration Server ソフトウェアのアップグレードにより影響を受ける可能性があるデータの種類を示しています。

表 4-2 Directory Server、Administration Server、および Directory Proxy Server のデータの使用方法

データの種類	場所	使用方法
Directory Server 設定データ	設定ディレクトリ	Directory Server の設定
Administration Server 設定データ	設定ディレクトリ	Administration Server の設定

## 互換性の問題

Java ES Release 4 の Directory Server および Administration Server では、インタフェースの変更は行われていません。これらのコンポーネントは、グループとして見た場合は以前のバージョンと下位互換性があります。ただし、これらのコンポーネントはどちらも、もう一方のコンポーネントの以前のバージョンとは下位互換性がありません。両方合わせて 1 つの単位としてアップグレードする必要があります。

## 依存関係

ほかの Java ES コンポーネントへの依存関係により、Directory Server および Administration Server ソフトウェアのアップグレードと再設定の手順が影響を受ける場合があります。これらの各コンポーネントは、Java ES コンポーネントに対する次の依存関係を持ちます。

- **Directory Server:** Directory Server は、特定の Java ES 共有コンポーネントへの依存関係を持ちます (42 ページの表 1-6 を参照)。Directory Server は Administration Server に依存します。Administration Server は、Directory Server のレプリケーションや、Directory Server のその他の機能を設定するために使用されます。
- **Administration Server:** Administration Server ( および管理コンソールのユーザーインタフェース ) は、特定の Java ES 共有コンポーネントへの依存関係を持ちます (42 ページの表 1-6 を参照)。Administration Server は Directory Server に設定データを格納するため、Directory Server ( 特に設定ディレクトリ ) に依存します。

## Java ES Release 3 からの Directory Server と Administration Server のアップグレード

この節では、Java ES 2005 Q1 (Release 3) から Java ES 2005Q4 (Release 4) への、Directory Server と Administration Server のアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Directory Server と Administration Server のアップグレード](#)
- [複数インスタンスのアップグレード](#)

### はじめに

Java ES Release 3 の Directory Server および Administration Server を Release 4 にアップグレードする場合は、アップグレードプロセスの次の側面を考慮してください。

- **一般的なアップグレード手順:** このアップグレードは、Java ES Release 3 バージョンにパッチを適用することにより実行されます。Directory Server および Administration Server の再設定は、設定ディレクトリをアップグレードされたソフトウェアと同期することによって行います。
- **アップグレードの依存関係:** Directory Server と Administration Server は多くの Java ES 共有コンポーネントに依存しますが (42 ページの表 1-6 を参照)、Java ES Release 4 の Directory Server と Administration Server は、これらの共有コンポーネントの Release 3 バージョンと互換性があります。したがって、これらの共有コンポーネントのアップグレードは、Release 4 への Directory Server と Administration Server のアップグレードに関しては省略可能です。

Directory Server は、Administration Server に対する強いアップグレード依存関係を持ちます。そのため、これらのコンポーネントは同時に Release 4 にアップグレードすることが推奨されます。

- **下位互換性** : Release 4 の Directory Server と Administration Server は、それぞれの Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Solaris の場合、Release 4 へのアップグレードのロールバックは、Release 4 のアップグレードパッチを削除し、設定ディレクトリを以前のソフトウェア状態と再同期することによって行います。一方、Linux については、Release 4 へのアップグレードをロールバックする手順はありません。
- **プラットフォームの問題** : Directory Server と Administration Server をアップグレードするための一般的な手順は Solaris オペレーティングシステムと Linux オペレーティングシステムのどちらでも同じですが、パッチの適用方法が異なります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Directory Server と Administration Server のアップグレード

この節では、Solaris および Linux プラットフォームで、Directory Server と Administration Server を Java ES Release 3 から Java ES Release 4 にアップグレードする方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Directory Server および Administration Server のアップグレード \(Solaris\)](#)
- [Release 3 の Directory Server および Administration Server のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Directory Server および Administration Server をアップグレードする前に、次に説明されている作業を実行してください。

#### **現在のバージョン情報の確認**

Directory Server および Administration Server の現在のバージョンは、次のように、`-v` オプションを使用して Directory Server デーモンを再起動することによって確認できます。

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

次に、Directory Server のエラーログ内の起動メッセージを確認します。

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

表 4-3 Directory Server のバージョン確認の出力

Java ES のリリース	Directory Server のバージョン番号
Release 2	Sun Java(TM) System Directory Server/5.2_Patch_2
Release 3	Sun Java(TM) System Directory Server/5.2_Patch_3
Release 4	Sun Java(TM) System Directory Server/5.2_Patch_4

注：Solaris 10 プラットフォームで ns-slapd コマンドが失敗した場合は、コマンド実行時にライブラリパスを NULL に設定するようにしてください。

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

### Directory Server と Administration Server が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上（および、コンピューティング環境内）のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。ただし、Directory Server および Administration Server では Release 3 の共有コンポーネントのアップグレードは必須ではないため、この作業は省略可能です。

### Directory Server のデータのバックアップ

Directory Server と Administration Server のアップグレードプロセスでは、設定ディレクトリのデータが変更されます。このため、アップグレードを実行する前に、Directory Server コンソールや、db2bak などのコマンド行ユーティリティを使用して、設定ディレクトリのデータをバックアップすることが推奨されます。

Directory Server のバックアップの詳細については、『Sun Java System Directory Server 管理ガイド』（<http://docs.sun.com/source/819-2011?l=ja>）を参照してください。

### 必要な設定情報とパスワードの入手

現在インストールされているバージョンの Directory Server の管理者ユーザー ID とパスワードを調べておきます。

また、Directory Server と Administration Server は同じユーザーおよびグループとして実行する必要があります。つまり、これらは同一の UID と GID を使って実行される必要があります。

## Release 3 の Directory Server および Administration Server のアップグレード (Solaris)

この節では、Directory Server および Administration Server のアップグレード手順に影響する考慮事項について説明し、続いて手順自体を説明します。

### アップグレードの注意点 (Solaris)

Directory Server および Administration Server ソフトウェアを Java ES Release 4 にアップグレードするときは、次の注意点を考慮に入れてください。

- Directory Server インスタンスをアップグレードする前に、そのインスタンスを使用するすべての Java ES コンポーネント (Access Manager、Communications Express、Messaging Server、Portal Server など) をシャットダウンすることが推奨されます。ただし、ほとんどの配備アーキテクチャーでは、高可用性またはスケラビリティを実現するために、Directory Server の複数のインスタンスを使用します。そのような場合、Directory Server のローリングアップグレードを実行できるため、Directory Server クライアントをシャットダウンする必要はありません。
- データの再設定は決まった順序で行う必要があるため、Directory Server よりも先に Administration Server をアップグレードする必要があります。
- パッチの適用作業中は、アップグレード対象のコンポーネントをシャットダウンする必要があります。ただしその後、アップグレード対象のコンポーネントを再設定するときは、そのコンポーネントと関連付けられた設定ディレクトリが稼働している必要があります。
- Directory Server の複数のインスタンスが 1 台のコンピュータ上で稼働しており、インストールされた同じ Directory Server イメージにそれらすべてのインスタンスが対応しているような配備アーキテクチャーでは、Directory Server イメージをアップグレードするとすべてのインスタンスがアップグレードされます。そのようなアーキテクチャーでは、インストールされた Directory Server イメージごとに Administration Server インスタンスが 1 つだけ存在します。
- 多くの配備アーキテクチャーでは、設定ディレクトリは独立した Directory Server インスタンスです。設定ディレクトリはローカルの場合もあれば、アップグレードが行われているコンピュータとは別のコンピュータシステムに位置している場合もあります。同様に、Administration Server はローカルの場合もあれば、Directory Server のアップグレードの実行元とは別のコンピュータシステムに位置している場合もあります。
- 一部の配備アーキテクチャーでは、Directory Server は、インストール時に Administration Server を選択解除することによってスタンドアロンでインストールされます。ただしその場合、あとで説明するように、Directory Server のアップグレード手順に加えて Administration Server のアップグレード手順も実行する必要があります (スタンドアロンモードを選択した場合でも一部の Administration Server コードはインストールされるため)。



- 次の表は、Solaris OS で Directory Server および Administration Server を Release 4 にアップグレードするためのパッチを示したものです。

表 4-4 Solaris で Directory Server および Administration Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC Solaris 8、9、10	x86 Solaris 9、10
Directory Server	115614-26	115615-26
Directory Server ローカライズ版	117015-21	117015-21
Administration Server	115610-23	115611-23
Administration Server ローカライズ版	117047-24	117047-24

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

ここで説明する手順は、アップグレードを実施するコンピュータにローカルインストールされている Directory Server および Administration Server インスタンスに適用されます。

この手順では、`directoryserver(1m)` と `mpsadmserver(1m)` の 2 つのコマンドを使用します。これらのコマンドの詳細については、『Directory Server Man Page Reference』および『Administration Server Man Page Reference』を参照してください。

1. 表 4-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. `root` としてログインするか、スーパーユーザーになります。

```
su -
```

3. 管理コンソールがローカルで稼働している場合、停止します。
4. アップグレードする Directory Server インスタンスに依存するすべての Java ES コンポーネントをシャットダウンします。この手順は、配備アーキテクチャー内部でのこれらのコンポーネントのレプリケート形態によって異なる場合があります。コンポーネントは次の順序でシャットダウンすることが推奨されます。
  - a. Directory Server クライアント : Access Manager、Communications Express、Messaging Server、Portal Server、その他
  - b. Directory Proxy Server (Directory Server へのアクセスに使用されている場合)

- c. Administration Server (ローカルで稼働している場合)
- d. Directory Server
- e. 設定ディレクトリ (独立した Directory Server インスタンスとしてローカルで稼働している場合)

Java ES コンポーネントのシャットダウン方法については、各コンポーネントの管理ガイドを参照してください。

- 5. Directory Server および Administration Server が強いアップグレード依存関係を持つ Java ES コンポーネント (111 ページの「[Directory Server と Administration Server が依存するコンポーネントのアップグレード](#)」を参照) がすべてアップグレード済みであることを確認します。
- 6. Administration Server をアップグレードします。

この手順は、Directory Server が元々、アップグレードが行われているコンピュータにスタンドアロンモードでインストールされた場合であっても実行する必要があります (スタンドアロンモードでも、一部の Administration Server コードはインストールされるため)。

- a. アップグレードする Administration Server を再起動します。
- b. [表 4-4](#) の Administration Server パッチを適用します。

Administration Server のベースパッチを適用する前に、Administration Server のローカライズ版パッチ (117047) を適用するようにしてください。

```
patchadd patch_ID
```

- c. パッチのアップグレードが成功したことを確認します。

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

この出力では、[手順 b](#) で適用したパッチ ID のバージョンが返されるはずで  
す。

- d. 設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認し  
ます。

- e. アップグレードされた設定を設定ディレクトリと同期します。

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

管理ユーザー名とパスワードの入力を求められます。

- 7. Directory Server をアップグレードします。

- a. Administration Server なしのスタンドアロンモードで Directory Server を実行している場合、次の手順を実行します。そうでない場合、[手順 7b](#) に進みます。

I. **手順 6** で Administration Server が正しくアップグレードされたことを確認します。

II. `serverroot` ディレクトリに移動します。

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

III. 設定ディレクトリを作成します。

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

IV. `adm.config` ファイルを作成します。

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

V. 次のテキストを追加します。

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

このテキストは改行せず 1 行で入力します。`hostname` は Directory Server の完全修飾ホスト名、`administration_domain` は通常はホストのドメイン名です。

b. アップグレード対象の Directory Server インスタンスがシャットダウンされていることを確認します。

c. **表 4-5** の Directory Server パッチを適用します。

Directory Server のベースパッチを適用する前に、Directory Server のローカライズ版パッチ (117015) を適用するようにしてください。

```
patchadd patch_ID
```

d. パッチのアップグレードが成功したことを確認します。

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

この出力では、**手順 c** で適用したパッチ ID のバージョンが返されるはずで  
す。

e. デフォルトの Directory Server バージョン番号をリセットします。

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

f. 設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認し  
ます。

g. アップグレードされた設定を設定ディレクトリと同期します。

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

管理ユーザー名とパスワードの入力を求められます。

8. 手順 4 でシャットダウンしたときと逆の順序で、すべての Java ES コンポーネントを再起動します。
  - a. 設定ディレクトリ (独立した Directory Server インスタンスとしてローカルで稼働している場合)
  - b. Directory Server
  - c. Administration Server (ローカルで稼働している場合)
  - d. Directory Proxy Server (Directory Server へのアクセスに使用されている場合)
  - e. Directory Server クライアント: Access Manager、Communications Express、Messaging Server、Portal Server、その他

## Release 3 の Directory Server および Administration Server のアップグレード (Linux)

この節では、Directory Server および Administration Server のアップグレード手順に影響する考慮事項について説明し、続いて手順自体を説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Directory Server とその関連コンポーネントを Java ES Release 4 にアップグレードするときは、Solaris プラットフォームの場合と同じ事項を考慮します (112 ページの「アップグレードの注意点 (Solaris)」を参照)。ただし、Linux 用の Release 4 アップグレードパッチは Solaris 用のパッチとは異なります。

次の表は、Linux OS で Directory Server および Administration Server を Release 4 にアップグレードするためのパッチを示したものです。

表 4-5 Linux で Directory Server および Administration Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Directory Server	118080-11: sun-directory-server-5.2-25.i386.rpm sun-directory-server-man-5.2-9.i386.rpm
Directory Server ローカライズ版	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

表 4-5 Linux で Directory Server および Administration Server をアップグレードするためのパッチ<sup>1</sup> (続き)

説明	パッチ ID と RPM 名
Administration Server ローカライズ版	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. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

ここで説明する手順は、アップグレードを実施するコンピュータにローカルインストールされている Directory Server および Administration Server インスタンスに適用されます。

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

この手順では、`directoryserver(1m)` と `mpsadmserver(1m)` の 2 つのコマンドを使用します。これらのコマンドの詳細については、『*Directory Server Man Page Reference*』および『*Administration Server Man Page Reference*』を参照してください。

1. 表 4-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

以降の手順中で、`oldVersion` は Directory Server および Administration Server の Release 3 バージョンの RPM を表します。

2. `root` としてログインするか、スーパーユーザーになります。  
su -
3. 管理コンソールがローカルで稼働している場合、停止します。
4. アップグレードする Directory Server インスタンスに依存するすべての Java ES コンポーネントをシャットダウンします。この手順は、配備アーキテクチャー内部でのこれらのコンポーネントのレプリケート形態によって異なる場合があります。コンポーネントは次の順序でシャットダウンすることが推奨されます。
  - a. Directory Server クライアント : Access Manager、Communications Express、Messaging Server、Portal Server、その他

- b. Directory Proxy Server (Directory Server へのアクセスに使用されている場合)
- c. Administration Server (ローカルで稼働している場合)
- d. Directory Server
- e. 設定ディレクトリ (独立した Directory Server インスタンスとしてローカルで稼働している場合)

Java ES コンポーネントのシャットダウン方法については、各コンポーネントの管理ガイドを参照してください。

- 5. Directory Server および Administration Server が強いアップグレード依存関係を持つ Java ES コンポーネント ([111 ページの「Directory Server と Administration Server が依存するコンポーネントのアップグレード」](#)を参照) がすべてアップグレード済みであることを確認します。
- 6. Administration Server の各 RPM を適用します。
  - a. Administration Server: 製品の RPM を適用します。

この手順は、アップグレードが行われているコンピュータに Directory Server がスタンドアロンモードでインストールされた場合でも実行する必要があります。

- I. RPM を次のように適用します。

Administration Server のベース RPM を適用する前に、Administration Server のローカライズ版 RPM (118289) を適用するようにしてください。

```
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
...
```

Administration Server が以前に設定されていた場合、次のエラーが返されます。

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

この場合、旧バージョンの RPM を、次のように `--noscripts` オプションを使って削除します。

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

- II. Administration Server が以前に設定された場合は、設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。

- III. アップグレードされた設定を設定ディレクトリと同期します。

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

管理ユーザー名とパスワードの入力を求められます。

#### IV. アップグレードが成功したことを確認します。

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

RPM の新しいバージョン番号が返されることを確認します。

#### b. Administration Server: コンソールの RPM を適用します。

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

#### c. Administration Server: マニュアルページの RPM を適用します。

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

### 7. Directory Server の各 RPM を適用します。

#### a. Administration Server なしのスタンドアロンモードで Directory Server を実行している場合、Administration Server の RPM を適用します。

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

そうでない場合は、[手順 7b](#)に進みます。

#### b. Directory Server: 製品の RPM を適用します。

##### I. アップグレード対象の Directory Server インスタンスがシャットダウンされていることを確認します。

##### II. RPM を次のように適用します。

Directory Server の RPM を適用する前に、Directory Server のローカライズ版 RPM (118290) を適用するようにしてください。

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

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

...

Directory Server が以前に設定されていた場合、次のエラーが返されます。

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

この場合、旧バージョンの RPM を、次のように `--noscripts` オプションを使って削除します。

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

##### III. Directory Server が以前に設定された場合は、設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。

- IV. アップグレードされた設定を設定ディレクトリと同期します。

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

管理ユーザー名とパスワードの入力を求められます。

- V. アップグレードが成功したことを確認します。

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

RPM の新しいバージョン番号が返されることを確認します。

- c. Directory Server: マニュアルページの RPM を適用します。

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

8. **手順 4** でシャットダウンしたときと逆の順序で、すべての Java ES コンポーネントを再起動します。
- a. 設定ディレクトリ (独立した Directory Server インスタンスとしてローカルで稼働している場合)
  - b. Directory Server
  - c. Administration Server (ローカルで稼働している場合)
  - d. Directory Proxy Server (Directory Server へのアクセスに使用されている場合)
  - e. Directory Server クライアント: Access Manager、Communications Express、Messaging Server、Portal Server、その他

## アップグレードの確認

Directory Server および Administration Server のアップグレードが成功したかどうかは、次のように、`-v` オプションを使用して Directory Server デーモンを再起動することによって確認できます。

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

次に、Directory Server のエラーログ内の起動メッセージを確認します。

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

出力値については、[111 ページの表 4-3](#) を参照してください。

## アップグレード後の作業

[113 ページの「アップグレード手順 \(Solaris\)」](#) および [117 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。



## アップグレードのロールバック (Solaris)

この節では、Directory Server および Administration Server のアップグレードのロールバック手順に影響する注意事項について説明し、続いて手順自体を説明します。

### ロールバックの注意点 (Solaris)

Directory Server および Administration Server の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。パッチを削除してから、設定ディレクトリを再同期します。

特別な注意事項の 1 つは、パッチの適用時に SSL 証明書データベースを cert8 形式にアップグレードすることです。パッチは cert7 データをバックアップしてから、そのデータを cert8 形式に変換します。あとからアップグレードをロールバックすることに決定し、新しい証明書が証明書データベースに追加済みである場合、手動でこれらの証明書を抽出し、パッチをバックアウトしてから、以前の cert7 形式の証明書データベースに証明書を再び追加することをお勧めします。

SSL 証明書データベースを変更したあとでアップグレードをロールバックすると、SSL モードで起動できなくなります。この問題を回避するには、SSL モードをオフにし、Directory Server および Administration Server を再起動し、証明書を再インストールしたあとで、SSL モードを有効にします。

### ロールバック手順 (Solaris)

1. 管理コンソールがローカルで稼働している場合、停止します。
2. ロールバックする Directory Server インスタンスに依存するすべての Java ES コンポーネントをシャットダウンします。この手順は、配備アーキテクチャー内部でのこれらのコンポーネントのレプリケート形態によって異なります。

コンポーネントは次の順序でシャットダウンすることが推奨されます。

- a. Directory Server クライアント : Access Manager、Communications Express、Messaging Server、Portal Server、その他
- b. Directory Proxy Server (Directory Server へのアクセスに使用されている場合)
- c. Administration Server (ローカルで稼働している場合)
- d. Directory Server
- e. 設定ディレクトリ (独立した Directory Server インスタンスとしてローカルで稼働している場合)

Java ES コンポーネントのシャットダウン方法については、各コンポーネントの管理ガイドを参照してください。

3. Directory Server のアップグレードをロールバックします。
  - a. ロールバック対象の Directory Server インスタンスがシャットダウンされていることを確認します。

- b. **表 4-5** の Directory Server パッチを削除します。  
`patchrm patch_ID`
  - c. 設定ディレクトリが稼働していることを確認します。  
設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。
  - d. ロールバックされた設定を設定ディレクトリと同期します。  
`/usr/sbin/directoryserver -u 5.2 sync-cds`  
管理ユーザー名とパスワードの入力を求められます。
  - e. Administration Server なしのスタンドアロンで Directory Server を実行している場合、**手順 4** の指示に従って、Administration Server の部分的アップグレードをロールバックする必要があります。
4. Administration Server のアップグレードをロールバックします。
- a. **表 4-5** の Administration Server パッチを削除します。  
`patchrm patch_ID`
  - b. 設定ディレクトリが稼働していることを確認します。  
設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。
  - c. アップグレードされた設定を設定ディレクトリと同期します。  
`/usr/sbin/mpsadmserver sync-cds`  
管理ユーザー名とパスワードの入力を求められます。
5. Directory Server および Administration Server が強いアップグレード依存関係を持つすべての Java ES コンポーネントへのアップグレードをロールバックします。
6. **手順 2** でシャットダウンしたときと逆の順序で、すべての Java ES コンポーネントを再起動します。
- a. 設定ディレクトリ (独立した Directory Server インスタンスとしてローカルで稼働している場合)
  - b. Directory Server
  - c. Administration Server (ローカルで稼働している場合)
  - d. Directory Proxy Server (Directory Server へのアクセスに使用されている場合)
  - e. Directory Server クライアント : Access Manager、Communications Express、Messaging Server、Portal Server、その他

## 複数インスタンスのアップグレード

110 ページの「Release 3 の Directory Server と Administration Server のアップグレード」の手順は、可用性またはスケーラビリティのために Directory Server がレプリケートされる配備アーキテクチャーには対応していません。これらのアーキテクチャーには、Directory Server マルチマスターレプリケーションや、Sun Cluster 環境におけるデータサービスとしての Directory Server の配備が含まれる場合があります。

この節では、これらの状況における Directory Server のアップグレードについて説明します。

### マルチマスターレプリケーションのローリングアップグレード

マルチマスターレプリケーション配備アーキテクチャーでは、複数の異なるコンピュータシステムに Directory Server の複数のインスタンスを配備します。このような複数のインスタンスを、一度に1つのインスタンスずつ連続してアップグレードすることができます。各ホストコンピュータ上の各インスタンスのアップグレードは、ほかのインスタンスを稼働させたまま実行されます。これはローリングアップグレードと呼ばれる仕組みであり、ディレクトリサービスをオンラインに保ったまま、サービスを提供する個別の Directory Server インスタンスを順次アップグレードすることができます。

### データサービスとしての Directory Server のアップグレード

この節では、Sun Cluster 環境で、データサービスとしての Directory Server をアップグレードおよびロールバックする方法について説明します。Sun Cluster データサービスとしての Directory Server をアップグレードまたはバックアウトする前に、次の事項を考慮してください。

- アップグレードまたはロールバック操作を実行する前にデータをバックアップします。
- すべてのクラスタノード上の Directory Server とそれに関連付けられた Administration Server に、並行してではなく連続的にパッチを適用します。
- すべてのクラスタノードで、Directory Server とそれに関連付けられた Administration Server の同じバージョンおよびリリースを実行することが推奨されます。
- クラスタをフェイルオーバーモードで実行している場合、HAStorage から HAStoragePlus へのアップグレードを検討してください。

### Sun Cluster データサービスとしての Directory Server のアップグレード

1. 各 Directory Server インスタンスと、関連付けられた Administration Server を停止します。

```
serverroot/stop-admin  
serverroot/slapd-instanceName/stop-slapd
```

2. 現在のクラスタノードをアクティブノードにします。  

```
scswitch -z -g ldap-group -h this-node-name
```
3. 110 ページの「[Release 3 の Directory Server と Administration Server のアップグレード](#)」で説明した手順に従って、現在のノード上の Directory Server をアップグレードします。
4. 別のクラスタノードをアクティブノードにします。  

```
scswitch -z -g ldap-group -h another-node-name
```
5. クラスタに含まれるすべてのノードがアップグレードされるまで、[手順 3](#) と [手順 4](#) を繰り返します。

### **Sun Cluster データサービスとしての Directory Server のロールバック**

1. 各 Directory Server インスタンスと、関連付けられた Administration Server を停止します。  

```
serverroot/stop-admin  
serverroot/slapd-instanceName/stop-slapd
```
2. 現在のクラスタノードをアクティブノードにします。  

```
scswitch -z -g ldap-group -h this-node-name
```
3. 121 ページの「[アップグレードのロールバック \(Solaris\)](#)」で説明した手順に従って、現在のノード上の Directory Server をロールバックします。
4. 別のクラスタノードをアクティブノードにします。  

```
scswitch -z -g ldap-group -h another-node-name
```
5. クラスタ内のすべてのノード上の Directory Server がロールバックされるまで、[手順 3](#) および [手順 4](#) を繰り返します。

# Java ES Release 2 からの Directory Server と Administration Server のアップグレード

Java ES 2004Q2 (Release 2) の Directory Server と Administration Server を Release 4 にアップグレードする手順は、Release 3 の Directory Server と Administration Server を Release 4 にアップグレードする手順とほとんど同じです。ただし、異なる点として、Directory Server と Administration Server が依存するすべての共有コンポーネント (42 ページの表 1-6 を参照) およびローカルに位置するすべての製品コンポーネントを Release 4 にアップグレードする作業を、インストール前の作業に含めることが推奨されます。

Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。

Release 2 の Directory Server および Administration Server を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、109 ページの「Java ES Release 3 からの Directory Server と Administration Server のアップグレード」の手順をそのまま使用してください。



# Directory Proxy Server

この章では、Directory Proxy Server を Java ES 2005Q4 (Release 4): Sun Java System Directory Proxy Server 5.2 2005Q4 にアップグレードする方法について説明します。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [128 ページの「Directory Proxy Server のアップグレードの概要」](#)
- [130 ページの「Java ES Release 3 からの Directory Proxy Server のアップグレード」](#)
- [139 ページの「Java ES Release 2 からの Directory Proxy Server のアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*serverRoot* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Directory Proxy Server が最初にインストールおよび設定されたときにインストールディレクトリとして指定されている場合があります。指定されていない場合、デフォルト値が割り当てられています。

*serverRoot* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

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

# Directory Proxy Server のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Directory Proxy Server の次の一般的な側面について説明します。

- [Java ES Release 4 について](#)
- [Java ES Release 4 へのアップグレードのロードマップ](#)
- [Directory Proxy Server のデータ](#)
- [互換性の問題](#)
- [依存関係](#)

## Java ES Release 4 について

Java ES Release 4 の Directory Proxy Server では、若干のバグ修正と機能の改善だけが行われています。新しい機能の追加はありません。

## Java ES Release 4 へのアップグレードのロードマップ

[表 5-1](#) は、Directory Proxy Server の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 5-1 Java ES Release 4: Sun Java System Directory Proxy Server 5.2 2005Q4 へのアップグレードパス

Java ES のリリース	Directory Proxy Server バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Directory Proxy Server 5.2 2005Q1	直接アップグレード：パッチを適用し、設定ディレクトリを再設定します。	設定ディレクトリ内のデータの自動再設定
Release 2	Sun Java System Directory Proxy Server 5.2 2004Q2	直接アップグレード：パッチを適用し、設定ディレクトリを再設定します。	設定ディレクトリ内のデータの自動再設定
Release 1	Sun One Directory Proxy Server 5.2	直接アップグレードは動作保証なし：ただし、Release 2 からのアップグレードと同じ手順を使用できます。	設定ディレクトリ内のデータの自動再設定



表 5-1 Java ES Release 4: Sun Java System Directory Proxy Server 5.2 2005Q4 へのアップグレードパス ( 続き )

Java ES のリリース	Directory Proxy Server バージョン	一般的な手順	必要な再設定
それ以前の Java ES リリース	Sun One Directory Proxy Server 5.2	直接アップグレードは動作保証なし：ただし、Release 2 からのアップグレードと同じ手順を使用できます。	設定ディレクトリ内のデータの自動再設定
	Sun One Directory Access Router 5.0 または 5.0 SP1	直接アップグレード不可：まず Release 3 にアップグレードします。手順については、『Sun Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。  次に、Release 3 から Release 4 にアップグレードします。	手順については、『Sun Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。

## Directory Proxy Server のデータ

Directory Proxy Server は、設定データの格納に Directory Server を利用します。データはディレクトリ内部の特定のツリー構造に格納されます。設定を格納している Directory Server インスタンスのことを設定ディレクトリと呼びます。

ほとんどの配備アーキテクチャでは、設定ディレクトリを使用して設定情報を格納するほかのコンポーネントとは別のコンピュータに設定ディレクトリを配置します。

次の表は、Directory Proxy Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 5-2 Directory Proxy Server のデータの使用方法

データの種類	場所	使用法
Directory Proxy Server 設定データ	設定ディレクトリ	Directory Proxy Server の設定

## 互換性の問題

Java ES Release 4 の Directory Proxy Server では、インタフェースの変更は導入されておらず、以前のバージョンと下位互換性があります。

## 依存関係

ほかの Java ES コンポーネントに対する依存関係によって、Directory Proxy Server ソフトウェアのアップグレードおよび再設定の手順が影響を受ける場合があります。Directory Proxy Server は、特定の Java ES 共有コンポーネントへの依存関係を持ちます (42 ページの表 1-6 を参照)。Directory Proxy Server は Directory Server へのフロントエンドアクセスを提供し、設定のために Administration Server を使用します。そのため Directory Proxy Server は、Directory Server と Administration Server の両方に依存します。

# Java ES Release 3 からの Directory Proxy Server のアップグレード

この節では、Java ES 2005 Q1 (Release 3) から Java ES 2005Q4 (Release 4) への Directory Proxy Server のアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Directory Proxy Server のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 3 Directory Proxy Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : このアップグレードは、Java ES Release 3 バージョンにパッチを適用することにより実行されます。Directory Proxy Server の再設定は、設定ディレクトリをアップグレードされたソフトウェアと自動的に同期することによって行われます。
- **アップグレードの依存関係** : Directory Proxy Server はいくつかの Java ES 共有コンポーネントに依存しますが (42 ページの表 1-6 を参照)、Java ES Release 4 の Directory Proxy Server は、これらの共有コンポーネントの Release 3 バージョンと互換性があります。したがって、これらの共有コンポーネントのアップグレードは、Directory Proxy Server の Release 4 へのアップグレードに関しては省略可能です。

Directory Proxy Server は、Directory Server および Administration Server の両方に対する強いアップグレード依存関係を持ちます。したがって、3つのコンポーネントすべてを同時に Release 4 にアップグレードすることが推奨されます。

- **下位互換性** : Release 4 の Directory Proxy Server は、その Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Solaris プラットフォームでは、Release 4 アップグレードのロールバックは Release 4 アップグレードパッチを削除することによって行います。ただし、Linux プラットフォームでは、Release 4 アップグレードをロールバックする手順はありません。
- **プラットフォームの問題** : Directory Proxy Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Directory Proxy Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Directory Proxy Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Directory Proxy Server のアップグレード \(Solaris\)](#)
- [Release 3 の Directory Proxy Server のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Directory Proxy Server をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Directory Proxy Server の現在のバージョンは、次のコマンドを使用することにより確認できます。

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

出力の一覧を次の表に示します。

表 5-3 Directory Proxy Server のバージョン確認の出力

Java ES のリリース	Directory Proxy Server のバージョン番号
Release 2	Sun ONE Directory Proxy Server Version 5.2_Patch_2
Release 3	Sun ONE Directory Proxy Server Version 5.2_Patch_3
Release 4	Sun ONE Directory Proxy Server Version 5.2_Patch_4

### Directory Proxy Server が依存するコンポーネントのアップグレード

一般には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。

Directory Proxy Server は、Directory Server および Administration Server に対する強いアップグレード依存関係を持ちます。これは、Directory Server および Administration Server がリモートコンピュータ上で動作する場合も同様です。そのため、Directory Proxy Server をアップグレードする前にこれらのコンポーネントをアップグレードすることが推奨されます。

Directory Proxy Server が依存する Java ES Release 3 共有コンポーネントのアップグレードは省略できますが、実行することが推奨されます。

Directory Proxy Server が依存するコンポーネントのアップグレードは次の順序で行います。これらのアップグレードはすべて、Directory Proxy Server をアップグレードする前に行います。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。

### Directory Server のデータのバックアップ

Directory Proxy Server のアップグレードプロセスでは、設定ディレクトリのデータが変更されます。このため、アップグレードを実行する前に、Directory Server コンソールや、db2bak などのコマンド行ユーティリティを使用して、設定ディレクトリのデータをバックアップすることが推奨されます。

Directory Server のバックアップの詳細については、『Sun Java System Directory Server 管理ガイド』 (<http://docs.sun.com/source/819-2011?l=ja>) を参照してください。

## 必要な設定情報とパスワードの入手

Directory Proxy Server は、Directory Server および Administration Server と同じユーザーおよびグループとして実行する必要があります。つまり、これらはすべて同一の UID と GID を使って実行される必要があります。

## Release 3 の Directory Proxy Server のアップグレード (Solaris)

ここでは、Directory Proxy Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Directory Proxy Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Directory Proxy Server インスタンスをアップグレードする前に、そのインスタンスを使用するすべての Java ES コンポーネント (Access Manager、Communications Express、Messaging Server、Portal Server など) をシャットダウンすることが推奨されます。ただし、多くの配備アーキテクチャーでは、高可用性またはスケーラビリティを実現するために Directory Proxy Server の複数のインスタンスを使用します。そのような場合、Directory Proxy Server のローリングアップグレードを実行できるため、Directory Proxy Server クライアントをシャットダウンする必要はありません。
- 再設定は決まった順序で行う必要があるため、Directory Proxy Server のアップグレードは、Administration Server および Directory Server のアップグレード後のみ行うことが推奨されます。
- パッチの適用作業中は Directory Proxy Server をシャットダウンする必要がありますが、再設定を実行するときは、その Directory Proxy Server と関連付けられた設定ディレクトリが稼働している必要があります。
- Directory Proxy Server の複数のインスタンスが 1 台のコンピュータ上で稼働しており、インストールされた同じ Directory Proxy Server イメージにそれらすべてのインスタンスが対応しているような配備アーキテクチャーでは、Directory Proxy Server イメージをアップグレードするとすべてのインスタンスがアップグレードされます。そのようなアーキテクチャーでは、インストールされた Directory Proxy Server イメージごとに Administration Server インスタンスが 1 つだけ存在します。
- Solaris OS での Release 4 Directory Proxy Server のアップグレードパッチを次の表に示します。

表 5-4 Solaris で Directory Proxy Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Directory Proxy Server	116373-18	116374-18
Directory Proxy Server ローカライズ版	117017-20	117017-20

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

ここで説明する手順は、アップグレードを実施するコンピュータにローカルインストールされている Directory Proxy Server インスタンスに適用されます。

1. 表 5-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. 管理コンソールがローカルで稼働している場合、停止します。
4. アップグレードする Directory Proxy Server インスタンスに依存するすべての Java ES コンポーネントをシャットダウンします。この手順は、配備アーキテクチャーの内部での Directory Proxy Server のレプリケート形態によって異なる場合があります。

Java ES コンポーネントのシャットダウン方法については、各コンポーネントの管理ガイドを参照してください。

5. Directory Proxy Server が強いアップグレード依存関係を持つ Java ES コンポーネント (132 ページの「Directory Proxy Server が依存するコンポーネントのアップグレード」を参照) がすべてアップグレード済みであることを確認します。
6. Directory Proxy Server をアップグレードします。
  - a. 設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。

- b. [表 5-4](#) の Directory Proxy Server パッチを適用します。

Directory Proxy Server のベースパッチを適用する前に、Directory Proxy Server のローカライズ版パッチ (117017) を適用するようにしてください。

```
patchadd patch_ID
```

- c. パッチのアップグレードが成功したことを確認します。

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

この出力では、[手順 b](#) で適用したパッチ ID のバージョンが返されるはずで  
す。

7. Directory Proxy Server と、Directory Proxy Server に依存するすべての Java ES コンポーネントを再起動します。

Directory Proxy Server を再起動するには、次のコマンドを実行します。

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

## Release 3 の Directory Proxy Server のアップグレード (Linux)

ここでは、Directory Proxy Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで、Directory Proxy Server を Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームの場合と同じ注意点 ([133 ページの「アップグレードの注意点 \(Solaris\)」](#) を参照) を考慮に入れてください。

Linux OS での Release 4 Directory Proxy Server のアップグレードパッチを次の表に示します。

表 5-5 Linux で Directory Proxy Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Directory Proxy Server	118096-08: sun-directory-proxy-server-5.2-13.i386.rpm
Directory Proxy Server ローカライズ版	118288-11: sun-directory-proxy-server- <i>Locale</i> -5.2-16.i386.rpm

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

## アップグレード手順 (Linux)

ここで説明する手順は、アップグレードを実施するコンピュータにローカルインストールされている Directory Proxy Server インスタンスに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 5-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. 管理コンソールがローカルで稼働している場合、停止します。
4. アップグレードする Directory Proxy Server インスタンスに依存するすべての Java ES コンポーネントをシャットダウンします。この手順は、配備アーキテクチャの内部での Directory Proxy Server のレプリケート形態によって異なる場合があります。

Java ES コンポーネントのシャットダウン方法については、各コンポーネントの管理ガイドを参照してください。

5. Directory Proxy Server が強いアップグレード依存関係を持つ Java ES コンポーネント (132 ページの「[Directory Proxy Server が依存するコンポーネントのアップグレード](#)」を参照) がすべてアップグレード済みであることを確認します。
6. Directory Proxy Server の RPM を適用します。

- a. 設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。

- b. RPM を適用します。

Directory Proxy Server のベース RPM を適用する前に、Directory Proxy Server のローカライズ版 RPM を適用するようにしてください。

```
rpm -Fvh sun-directory-proxy-server-Locale-5.2-16.i386.rpm  
rpm -Fvh sun-directory-proxy-server-5.2-13.i386.rpm
```

アップグレード後の設定と設定ディレクトリとの同期が、自動的に実行されます。



- Directory Proxy Server と、Directory Proxy Server に依存するすべての Java ES コンポーネントを再起動します。

Directory Proxy Server を再起動するには、次のコマンドを実行します。

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

## アップグレードの確認

Directory Proxy Server のアップグレードが成功したかどうかは、次のコマンドを使用して確認できます。

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

出力値については、[132 ページの表 5-3](#) を参照してください。

## アップグレード後の作業

[134 ページの「アップグレード手順 \(Solaris\)」](#) および [136 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

この節では、Directory Proxy Server のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Directory Proxy Server の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。パッチを削除してから、設定ディレクトリを再同期します。

特別な注意事項の 1 つは、パッチの適用時に SSL 証明書データベースを cert8 形式にアップグレードすることです。パッチは cert7 データをバックアップしてから、そのデータを cert8 形式に変換します。あとからアップグレードをロールバックすることに決定し、新しい証明書が証明書データベースに追加済みである場合、手動でこれらの証明書を抽出し、パッチをバックアウトしてから、以前の cert7 形式の証明書データベースに証明書を再び追加することをお勧めします。

SSL 証明書データベースを変更したあとでアップグレードをロールバックすると、SSL モードで起動できなくなります。この問題を回避するには、SSL モードをオフにし、Directory Proxy Server および Administration Server を再起動し、証明書を再インストールしたあとで、SSL モードを有効にします。

### ロールバック手順 (Solaris)

- root としてログインするか、スーパーユーザーになります。

su -

2. 管理コンソールがローカルで稼働している場合、停止します。
3. アップグレードする Directory Proxy Server インスタンスに依存するすべての Java ES コンポーネントをシャットダウンします。この手順は、配備アーキテクチャーの内部での Directory Proxy Server のレプリケート形態によって異なる場合があります。

Java ES コンポーネントのシャットダウン方法については、各コンポーネントの管理ガイドを参照してください。

4. Directory Proxy Server のアップグレードをロールバックします。

- a. 設定ディレクトリが稼働していることを確認します。

設定ディレクトリがローカルの場合、ディレクトリの起動が必要な場合があります。設定ディレクトリがリモートの場合、稼働していることを確認します。

- b. 表 5-5 の Directory Proxy Server パッチを削除します。

```
patchrm patch_ID
```

5. Directory Proxy Server が強いアップグレード依存関係を持つすべての Java ES コンポーネント (特に Directory Server と Administration Server) のアップグレードをロールバックします。
6. Directory Proxy Server と、Directory Proxy Server に依存するすべての Java ES コンポーネントを再起動します。

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、スケーラビリティを実現し、可用性を向上させるために、Directory Proxy Server が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Directory Proxy Server コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。

Directory Proxy Server のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Directory Proxy Server インスタンスをアップグレードする順次アップグレードを実行できます。Directory Proxy Server 各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。131 ページの「[Release 3 の Directory Proxy Server のアップグレード](#)」の説明に従って、各インスタンスのアップグレードを実行します。

# Java ES Release 2 からの Directory Proxy Server のアップグレード

Java ES 2004Q2 (Release 2) の Directory Proxy Server を Release 4 にアップグレードする手順は、Release 3 の Directory Proxy Server を Release 4 にアップグレードする場合とほとんど同じです。ただし、アップグレード前の作業に、すべての共有コンポーネント (42 ページの表 1-6 を参照) と、Directory Proxy Server が依存するローカルにあるすべての製品コンポーネントの Release 4 へのアップグレードを含めるようにしてください。

Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。

Release 2 Directory Proxy Server を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、130 ページの「Java ES Release 3 からの Directory Proxy Server のアップグレード」の手順をそのまま使用してください。



# Web Server

この章では、Web Server を Java ES 2005Q4 (Release 4): Sun Java System Web Server 6.1 SP5 2005Q4 にアップグレードする方法について説明します。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [142 ページの「Web Server のアップグレードの概要」](#)
- [144 ページの「Java ES Release 3 からの Web Server のアップグレード」](#)
- [150 ページの「Java ES Release 2 からの Web Server のアップグレード」](#)

---

**注** この章では、*WebServer-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Web Server が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*WebServer-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

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

# Web Server のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Web Server の次の一般的な側面について説明します。

- [Java ES Release 4 の Web Server について](#)
- [Web Server のアップグレードロードマップ](#)
- [Web Server のデータ](#)
- [互換性の問題](#)
- [Web Server 依存関係](#)

## Java ES Release 4 の Web Server について

Java ES Release 4 バージョンの Web Server では、NSS 共有コンポーネントに依存するセキュリティ修正を含むいくつかのバグ修正が行われています。

詳細については、適切なリリースノートを参照してください。

## Web Server のアップグレードロードマップ

表 6-1 は、Web Server の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 6-1 Java ES Release 4: Sun Java System Web Server 6.1 SP5 2005Q4 へのアップグレードパス

Java ES のリリース	Web Server バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Web Server 6 2005Q1 Update 1 SP 4	直接アップグレード：パッチを適用することにより実行されます。	なし
Release 2	Sun Java System Web Server 6 2004Q2 Update 1 SP 2 Platform Edition および Enterprise Edition	直接アップグレード：パッチを適用することにより実行されます。	なし
Release 1	Sun ONE Web Server 6.1 (2003Q4)	直接アップグレードは動作保証なし：ただし、パッチを適用することにより実行できます。	なし
それ以前の Java ES リリース		直接アップグレード不可。	

## Web Server のデータ

次の表は、Web Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 6-2 Web Server のデータの使用方法

データの種類	場所	使用法
設定データ	<code>WebServer-base/https-instanceName/config/obj.conf</code> および同じディレクトリ内のその他のファイル	Web Server インスタンスの設定

## 互換性の問題

Java ES Release 4 の Web Server では、インタフェースの変更は導入されておらず、以前のバージョンと下位互換性があります。

## Web Server 依存関係

Web Server には、Java ES 共有コンポーネントを除くほかの Java ES コンポーネントへの依存関係はありません ([42 ページの表 1-6](#) を参照)。

# Java ES Release 3 からの Web Server のアップグレード

ここでは、Web Server の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Web Server のアップグレード](#)

## はじめに

Java ES Release 3 Web Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。Java ES Release 3 の Web Server から Release 4 へのアップグレードでは、Web Server の再設定は必要ありません。
- **アップグレードの依存関係** : Web Server は多くの Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Release 4 の Web Server では、NSS および NSPR の Release 4 へのアップグレードだけが必須です。その他の共有コンポーネントのアップグレードは、Release 4 への Web Server のアップグレードでは省略可能です。
- **下位互換性** : Release 4 Web Server は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Release 4 アップグレードの Release 3 へのロールバックは、アップグレード中に適用されたパッチを削除することにより行われません。
- **プラットフォームの問題** : Web Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Web Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Web Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。



- アップグレード前の作業
- Release 3 の Web Server のアップグレード (Solaris)
- Release 3 の Web Server のアップグレード (Linux)
- アップグレードの確認
- アップグレード後の作業
- アップグレードのロールバック (Solaris)

## アップグレード前の作業

Web Server をアップグレードする前に、次に説明されている作業を実行してください。

### 現在のバージョン情報の確認

Web Server の現在のバージョンは、次のように、`-version` オプションを使用して Web Server インスタンスサーバーを起動することによって確認できます。

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

表 6-3 Web Server のバージョン確認の出力

Java ES のリリース	Web Server のバージョン番号
Release 2	6.1SP2
Release 3	6.1SP4
Release 4	6.1SP5

### Web Server が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。ただし、Web Server を Release 4 にアップグレードするための必須要件は、Web Server のアップグレードの前に、NSS および NSPR 共有コンポーネントを Release 3 から Release 4 バージョンにアップグレードすることだけです。NSS および NSPR、またはその他の Java ES 共有コンポーネントを Release 4 にアップグレードする手順については、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明しています。

### Web Server のデータのバックアップ

Web Server の Release 3 から Release 4 へのアップグレードでは、設定データは変更されません。現在のデータをバックアップする必要はありません。

## 必要な設定情報とパスワードの入手

現在インストールされているバージョンに関する特別な情報はありません。ただし、アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。

## Release 3 の Web Server のアップグレード (Solaris)

ここでは、Web Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Web Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Web Server インスタンスをアップグレードする前に、そのインスタンス内で稼働しているすべての J2EE コンポーネントをシャットダウンすることが推奨されます。
- 同じ Web Server インストールイメージに対応するすべての Web Server インスタンスが同時にアップグレードされます。インストールイメージにパッチを適用している間は、このようなインスタンスをすべて停止するようにしてください。
- Solaris OS での Release 4 Web Server のアップグレードパッチを次の表に示します。

表 6-4 Solaris で Web Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC Solaris 8、9、10	x86 Solaris 9、10
Web Server コア (SUNWwbsvr)	116648-17	116649-17
Web Server ローカライズ版	117514-10	117515-10

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

ここで説明する手順は、アップグレードを実施するコンピュータにインストールされている同じ Web Server イメージに対応するすべての Web Server インスタンスに適用されます。

1. 表 6-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web Server および Administration Server の、稼働中のすべてのインスタンスを停止します。

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

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

4. NSS および NSPR 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

145 ページの「Web Server が依存するコンポーネントのアップグレード」を参照してください。

5. 表 6-4 の適切な Web Server パッチを適用します。

```
patchadd patch_ID
```

6. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 5 で適用したパッチ ID のバージョンが返されるはずですが。

7. 手順 3 で停止した Web Server インスタンスを再起動します。

## Release 3 の Web Server のアップグレード (Linux)

ここでは、Web Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Web Server ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (146 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Web Server のアップグレードパッチを次の表に示します。

表 6-5 Linux で Web Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Web Server コア	118202-09 <ul style="list-style-type: none"> <li>• sun-webserver-6.1.5-6.i386.rpm</li> </ul>
Web Server ローカライズ版	118203-06 <ul style="list-style-type: none"> <li>• sun-webserver-Locale-6.1.5-1.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

## アップグレード手順 (Linux)

ここで説明する手順は、アップグレードを実施するコンピュータにインストールされている同じ Web Server イメージに対応するすべての Web Server インスタンスに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 6-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web Server および Administration Server の、稼働中のすべてのインスタンスを停止します。

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

4. NSS および NSPR 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

[145 ページの「Web Server が依存するコンポーネントのアップグレード」](#) を参照してください。

5. 表 6-5 の Web Server の RPM を適用します。

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

6. アップグレードが成功したことを確認します。

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

RPM の新しいリビジョン番号が返されることを確認します。

7. 手順 3 で停止した Web Server インスタンスを再起動します。

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

## アップグレードの確認

Release 4 への Web Server のアップグレードが成功したかどうかは、次のように、`-version` オプションを使用して Web Server インスタンスサーバーを起動することによって確認できます。

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

出力値については、[145 ページの表 6-3](#) を参照してください。

また、次のログファイル内のエントリをチェックすることによっても確認できます。

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

## アップグレード後の作業

[146 ページの「アップグレード手順 \(Solaris\)」](#) および [148 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Web Server のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Web Server の Release 4 へのアップグレードをロールバックするには、Release 4 へのアップグレードと逆の手順を実行します。具体的には、パッチを削除します。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. Web Server および Administration Server の、稼働中のすべてのインスタンスを停止します。

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

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

3. [146 ページの表 6-4](#) のパッチを削除します。

```
patchrm patch_ID
```

4. [手順 2](#) で停止した Web Server インスタンスを再起動します。

# Java ES Release 2 からの Web Server のアップグレード

Java ES 2004Q2 (Release 2) の Web Server から Release 4 にアップグレードする手順は、Release 3 の Web Server を Release 4 にアップグレードする手順とほとんど同じです。ただし、異なる点として、Web Server が依存するすべての共有コンポーネント ([42 ページの表 1-6](#) を参照) を Release 2 バージョンから Release 4 にアップグレードする作業を、アップグレード前の作業に含めることが推奨されます。

Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、[55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」](#) で説明されています。

Release 2 Web Server を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、[144 ページの「Java ES Release 3 からの Web Server のアップグレード」](#) の手順をそのまま使用してください。

Release 2 から Release 4 へのアップグレードには設定ファイル `obj.conf` の変更も必要ですが、これは自動的に行われます。

# Message Queue

この章では、Message Queue ソフトウェアを、以前の Java ES バージョンから Java ES 2005 (Release 4): Sun Java System Message Queue 3 Enterprise Edition 2005Q4 にアップグレードする方法について説明します。

この章では、Java ES Release 4 でサポートされている各種のアップグレードパスでの Message Queue アップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [152 ページの「Message Queue のアップグレードの概要」](#)
- [158 ページの「Java ES Release 3 からの Message Queue のアップグレード」](#)
- [164 ページの「Java ES Release 2 からの Message Queue のアップグレード」](#)

---

**注** この章で使用する Message Queue のコマンドは、実行可能ファイルのディレクトリからの相対パスを使用して実行されます。実行可能ファイルのディレクトリの位置は、オペレーティングシステムプラットフォームによって異なります。

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

# Message Queue のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Message Queue の次の一般的な側面について説明します。

- [Java ES Release 4 の Message Queue について](#)
- [Message Queue のアップグレードロードマップ](#)
- [Message Queue のデータ](#)
- [互換性の問題](#)
- [Message Queue の依存関係](#)

## Java ES Release 4 の Message Queue について

Java ES Release 4 の Message Queue の変更点は若干のコード修正だけであり、新しい機能の追加や機能強化はありません。そのため、互換性に関して Release 4 で新しく生じる問題はありません (155 ページの「互換性の問題」を参照)。

Message Queue ソフトウェアには Platform Edition と Enterprise Edition の 2 つのエディションがあり、エディション間で機能セットおよびライセンス範囲が異なります。Enterprise Edition は、企業の実運用環境でメッセージングアプリケーションを配備し、実行するためのエディションです。Platform Edition は、メッセージングを伴うアプリケーションやコンポーネントの開発、デバッグ、および負荷テストを主な用途とするエディションです。Platform Edition は Sun の Web サイトから無償でダウンロード可能であり、Solaris OS および Java ES Application Server プラットフォームにも付属しています。Java ES の旧リリースバージョンから Release 4 にアップグレードすると、インストール済みの Platform Edition はすべて Enterprise Edition に変換されます。

## Message Queue のアップグレードロードマップ

表 7-1 は、Message Queue の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。



表 7-1 Java ES Release 4 Message Queue 3.6 SP3 2005Q4 へのアップグレードパス

Java ES のリリース	Message Queue バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Message Queue 2005Q2 (3.6) Enterprise Edition のみ	直接アップグレード: mqupgrade スクリプトを使用して行います。	なし
Release 2	Sun Java System Message Queue 2004Q2 (3.5) Platform Edition および Enterprise Edition	直接アップグレード: mqupgrade スクリプトを使用して行います。	Solaris プラットフォームでは自動的に実行され、Linux プラットフォームでは mqmigrate スクリプトを使用できます。
Release 1	Sun Java System Message Queue 3.01 SP2 Platform Edition および Enterprise Edition	直接アップグレードは動作保証なし: ただし、mqupgrade スクリプトを使用することにより実行できます。	Solaris プラットフォームでは自動的に実行され、Linux プラットフォームでは mqmigrate スクリプトを使用できます。
それ以前の Java ES リリース	Sun Java System Message Queue 3.01 SP1 およびそれ以前のバージョン Platform Edition および Enterprise Edition	直接アップグレード不可: まず Release 3 にアップグレードする必要があります。手順については、『Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。  次に、Release 3 から Release 4 にアップグレードします。	

表 7-1 に示した Java ES リリースの Message Queue に加えて、Message Queue Platform Edition は Solaris オペレーティングシステムソフトウェアにも付属します。付属するバージョンの Message Queue は、Java ES インストーラを使用して Release 4 Enterprise Edition にアップグレードできます。このアップグレードを行うには、インストーラで Message Queue の新規インストールを選択します。Platform Edition からのアップグレードと、Message Queue の必要な再設定がインストーラによって自動的に行われます。

## Message Queue のデータ

Message Queue では、ほかの Java ES コンポーネントと同様に、さまざまな種類のデータを利用します。Message Queue のアップグレードでは、これらのデータをアップグレード後のバージョンに移行しなければならない場合があります。次の表は、Message Queue ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 7-2 では、Solaris システムでのデータの位置を示しています。Linux システムでの位置もこれに似ています。詳しくは、『Message Queue 管理ガイド』(<http://docs.sun.com/source/819-3560?l=ja>) を参照してください。表 7-2 で、`instanceName` はデータの関連付けに使用される Message Queue ブローカインスタンスの名前を表します。

表 7-2 Message Queue のデータの使用方法

データカテゴリ	位置 (Solaris)	使用法
ブローカインスタンス設定プロパティ	<code>/var/imq/instances/<i>instanceName</i>/props/config.properties</code>	ブローカおよび関連サービスの設定
動的アプリケーションデータの持続的格納	<code>/var/imq/instances/<i>instanceName</i>/fs350/</code> または JDBC アクセス可能なデータストア	メッセージ、宛先、永続サブスクリプション、トランザクション、およびその他の動的データを格納する
管理対象オブジェクト (オブジェクトストア)	任意のローカルディレクトリ または LDAP Directory Server	クライアントとブローカの接続の設定に使用されるオブジェクト
セキュリティ: ユーザーリポジトリ	<code>/var/imq/instances/<i>instanceName</i>/etc/passwd</code> または LDAP ディレクトリサーバー	認証と承認に使用されるユーザーデータを格納する
セキュリティ: アクセス制御ファイル (デフォルトの位置)	<code>/var/imq/instances/<i>instanceName</i>/etc/accesscontrol.properties</code>	宛先および関連機能へのユーザーアクセスを承認するルールを設定する
セキュリティ: <code>passfile</code> ディレクトリ (デフォルトの位置)	<code>/var/imq/instances/<i>instanceName</i>/etc/</code>	暗号化されたパスワード情報を格納する
セキュリティ: ブローカのキーストアファイルの位置	<code>/etc/imq/</code>	セキュリティ保護されたメッセージングのための暗号化された証明書情報を格納する

## 互換性の問題

Release 4 の Message Queue では、Release 3 との非互換性に関する新しい問題はありません。ここで示す、Message Queue の互換性に関する一般的な問題は、Release 3 よりも前のバージョンに関係するものです。

### プロトコルの互換性

Message Queue は、Message Queue クライアントとブローカの間で HTTP プロトコル サポートを提供するために Web コンテナに依存します。Message Queue の `imqhttp.war` アプリケーションに対して Web コンテナを提供する目的で Sun Java System Web Server を使用しているとき、Web Server コンポーネントをアップグレードするには Message Queue もアップグレードする必要があります。これは、プロトコルの変更が原因です (162 ページおよび 167 ページの「[アップグレード後の作業](#)」を参照)。

### ブローカの互換性

Release 4 の Message Queue ブローカは Release 3 および Release 2 のブローカと相互運用可能ですが、Release 2 と比較した場合、ブローカプロパティと永続的ストアスキーマが変更されており、このことが互換性に影響を及ぼす可能性があります。

Release 4 の Message Queue は Release 3 および Release 2 のデータを使用できます。ただし、Linux システムでは Release 2 のデータをまず Release 4 に移行する必要があります。

Release 4 の Message Queue にアップグレードするときは、次の事項を考慮してください。

- Message Queue の古い `config.properties` ファイルを使用できます。また、それらのファイルを別の場所にコピーし、Release 4 の Message Queue ブローカを設定するときに、それらのファイルのプロパティ設定を参考にすることもできます。
- メッセージ、宛先、永続サブスクリプションなどの持続的 Message Queue データはすべて、ブローカの初回起動時に必要に応じて Release 4 の Message Queue データに自動的に変換されます。たとえば、必要な場合、既存の宛先がすべて Release 4 の Message Queue 宛先に変換されます。その際、既存の属性は保持され、新しい属性についてはデフォルト値が使用されます。
- 1 つのクラスタ内に Message Queue Release 2 のブローカと Message Queue Release 4 のブローカを混在させる場合は、Message Queue Release 2 のブローカ (古いほう) をマスターブローカにする必要があります、そのようなクラスタは Message Queue Release 2 クラスタとして動作します。

## 管理対象オブジェクトの互換性

Release 4 の Message Queue の管理対象オブジェクトは、Release 3 の管理対象オブジェクトと同一です。ただし、Release 3 の管理対象オブジェクトの中には、以前のバージョンから名前が変更されているものや、新しい属性が追加されているものがあります。したがって、Release 2 の Message Queue から Release 4 にアップグレードするときは、次の事項を考慮する必要があります。

- Release 2 で作成したオブジェクトストアおよび管理対象オブジェクトはそのままでも使用できますが、Release 4 に移行することをお勧めします。更新操作を実行すると、管理コンソール (imqadmin) および ObjectManager コマンド行ユーティリティー (imqobjmgr) によって、Release 2 の管理対象オブジェクトが Release 4 の管理対象オブジェクトに変換されます。
- Release 4 のクライアントランタイムは、Release 2 の管理対象オブジェクトを検索してインスタンス化し、Release 4 クライアントで使用できるようにそれらを変換します。ただし、検索実行元のオブジェクトストア内にある Release 2 の管理対象オブジェクトは変換対象にはなりません。
- 管理対象オブジェクトを検索するのではなく直接インスタンス化する、アプリケーションやコンポーネントなどの既存の Release 2 クライアントは、Release 4 と互換性があります。ただし、そのようなクライアントで管理対象オブジェクトの新しい属性を使用する場合、クライアントのコードを変更する必要があります。管理対象オブジェクトの属性の詳細については、『Sun Java System Message Queue 3 2005Q4 管理ガイド』(<http://docs.sun.com/source/819-3560?l=ja>) の第 16 章を参照してください。Release 2 のクライアントを Release 4 で再コンパイルすると、Message Queue Release 2 の属性のうち、どの属性の名前が Release 4 で変更されているかが示されます。古い名前も引き続き正しく機能します。
- Java クライアントを起動するスクリプトや、コマンド行オプションを使用して管理対象オブジェクトの属性値を設定するスクリプトは、Release 4 と互換性があります。ただし、それらのスクリプトで管理対象オブジェクトの新しい属性を使用する場合、スクリプトのコードを変更する必要があります。管理対象オブジェクトの属性の詳細については、『Sun Java System Message Queue 3 2005Q4 管理ガイド』(<http://docs.sun.com/source/819-3560?l=ja>) の第 16 章を参照してください。

## 管理ツールの互換性

Release 3 で新しいコマンドおよび管理機能が追加されたことにより、Release 4 の管理ツール (管理コンソールとコマンド行ユーティリティー) は Release 3 および Release 4 のブローカに対してのみ正常に機能します。ただし、Release 2 のすべてのコマンドおよびコマンドオプションも引き続きサポートされます。

## クライアントの互換性

Release 3 のクライアントは Release 4 の Message Queue と完全に互換性があります。ただし、Release 2 から Release 4 にアップグレードするときは、Java クライアントの互換性に関する次の問題を考慮する必要があります。

- Release 4 のブローカは Release 2 のクライアントをサポートします (ただし、Release 4 で追加された機能はサポートしない)。
- Release 4 の Java クライアントは Release 2 のブローカに接続できます (ただし、Release 4 で追加された機能は使用できない)。
- C クライアントのプログラムは、試用ライセンス (Platform Edition) または Enterprise Edition ライセンスで実行されている Release 2、Release 3、または Release 4 ブローカでのみサポートされます。

## Message Queue の依存関係

ほかの Java ES コンポーネントに対する Message Queue の依存関係が、Message Queue ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Message Queue のインタフェースまたは関数に変更された場合は、Message Queue が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Message Queue は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント**: Message Queue ソフトウェアは、特定の Java ES 共有コンポーネントに依存します ([42 ページの表 1-6](#) を参照)。
- **Directory Server (省略可能)**: 管理対象オブジェクトやユーザーデータをローカルではなく LDAP ディレクトリに格納するように Message Queue を設定する場合、格納先として Directory Server を使用できます。
- **Web コンテナ (省略可能)**: クライアントとブローカの間で HTTP メッセージをやり取りする必要がある場合、Message Queue の Web コンテナ要件を満たすために、Java ES Web Server、Java ES Application Server、またはサードパーティー製 Web コンテナを用意する必要があります。

# Java ES Release 3 からの Message Queue のアップグレード

ここでは、Message Queue の Java ES 2005Q1 (Release 3) から Java ES Release 4 へのアップグレードについて説明します。ここで説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Message Queue のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 3 Message Queue を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : アップグレードは `mqupgrade` スクリプトを使用します。このスクリプトは、古いソフトウェアパッケージを新しいパッケージに置き換え、設定データを Release 3 から自動的に移行します。
- **アップグレードの依存関係** : Message Queue は多くの Java ES 共有コンポーネントに対する依存関係を持ちますが ([42 ページの表 1-6](#) を参照)、Release 4 の Message Queue はこれらすべてのコンポーネントの Release 3 バージョンと互換性があります。したがって、これらの共有コンポーネントのアップグレードは、Message Queue の Release 4 へのアップグレードに関しては省略可能です。

Release 4 の Message Queue は、[157 ページの「Message Queue の依存関係」](#) で説明するように、Directory Server および Web Server (または Application Server) に依存する場合があります。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Message Queue の Release 4 へのアップグレードに関しては省略可能です。

- **下位互換性** : Release 4 の Message Queue は Release 3 と完全互換です ([155 ページの「互換性の問題」](#) を参照)。
- **アップグレードのロールバック** : Message Queue のアップグレードを Release 3 にロールバックするためのユーティリティはありません。アップグレードされたコンポーネントを削除し、以前のバージョンおよび設定データを手動で復元する必要があります。
- **プラットフォームの問題** : Message Queue をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通です。以降の手順説明では、必要に応じてプラットフォーム別のコマンドまたはファイル位置を示します。

## Release 3 の Message Queue のアップグレード

この節では、Message Queue を Java ES Release 3 から Java ES Release 4 にアップグレードする方法について説明します。

- [アップグレード前の作業](#)
- [Release 3 の Message Queue のアップグレード](#)
- [Message Queue のアップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック](#)

### アップグレード前の作業

Message Queue をアップグレードする前に、以降の節で説明する手順を実行してください。プラットフォームによって異なるコマンドを手順内で使用する箇所では、適宜、対象のオペレーティングシステムを示します。

#### 現在のバージョン情報の確認 (Solaris システム)

システムにインストールされている Message Queue のバージョンおよびエディションは、次のように、`-version` オプションを使用して Message Queue ブローカを起動することによって確認できます。

```
imqbrokerd -version
```

表 7-3 Message Queue のバージョン確認の出力

Java ES のリリース	Message Queue のバージョン番号
Release 2	Sun Java(tm) System Message Queue 3 2004Q2 Version: 3.5
Release 3	Sun Java(tm) System Message Queue 3 2005Q1 Version: 3.6
Release 4	Sun Java(tm) System Message Queue 3 2005Q4 Version: 3.6 SP3

### Message Queue が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。ただし Message Queue に関しては、Message Queue が依存する Java ES Release 3 コンポーネントのアップグレードは必須ではないため、この作業は省略可能です。

ただし、Message Queue のすべての依存関係をアップグレードすることを選択した場合は、Message Queue をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server (省略可能)**: Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。
3. **Web コンテナソフトウェア (省略可能)**: Web Server または Application Server をアップグレードする手順は、それぞれ 141 ページの第 6 章「Web Server」および 177 ページの第 9 章「Application Server」で説明されています。

### Message Queue のバックアップ

Message Queue を以前の状態にロールバックするためのスクリプトはありません。Release 4 のデータは Release 3 のデータと互換性があるため、設定データをバックアップする必要はありません。また、Release 4 の Message Queue を Release 3 にロールバックする必要があるときは Release 3 のインストーラを使用できるため、インストール済みのイメージをバックアップする必要もありません。

## Release 3 の Message Queue のアップグレード

Java ES Release 4 への Message Queue ソフトウェアのアップグレードでは、mqupgrade スクリプトを利用します。このスクリプトは、表 7-4 に示すパッチを含む新しいパッケージをインストールします。

表 7-4 Message Queue をアップグレードするためのパッチ<sup>1</sup>

コンポーネント	SPARC Solaris 8、9、10	x86 Solaris 9、10	Linux
Message Queue コア	119132-06	119133-06	119136-06
Message Queue C ランタイム	119134-04	119135-04	
Message Queue jmsclient および xmlclient			119137-04
Message Queue ローカライズ版	119691-03	119692-03	119693-03

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。



アップグレード手順は、次の手順で構成されます。

1. 稼働中の Message Queue クライアントアプリケーションをすべて停止します。

Application Server 環境で Message Queue を使用している場合、Application Server もシャットダウンします。

2. 稼働中のすべてのブローカを停止します。管理ユーザー名とパスワードの入力を求められます。

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

3. 各ブローカインスタンスに関連付けられた動的データ、Message Queue フラットファイルユーザーリポジトリ、および Message Queue アクセス制御ファイルを保持しない場合、次のコマンドを使ってこれらのデータを削除します。

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

削除しない場合、動的データおよび設定情報は保持され、Release 4 の Message Queue で使用されます。

4. root としてログインします。

```
su -
```

5. Java ES 配布パッケージの Tools ディレクトリに移動します。

Solaris SPARC の場合：

```
cd Solaris_sparc/Product/message_queue/Tools
```

Solaris x86 の場合：

```
cd Solaris_x86/Product/message_queue/Tools
```

Linux x86 の場合：

```
cd Linux_x86/Product/message_queue/Tools
```

6. mqupgrade スクリプトを実行します。

- a. スクリプトを起動します。

```
./mqupgrade
```

mqupgrade スクリプトは、インストールされている Message Queue コンポーネントを一覧表示します。

- b. y (yes) と入力して、Message Queue コンポーネントをアップグレードします。

mqupgrade スクリプトは、インストールされているローカライズ版ファイルを検出して一覧表示します。

Message Queue コンポーネントをアップグレードしない場合は、n (no) と入力します。mqupgrade スクリプトは、Message Queue コンポーネントをアップグレードせずに終了します。

- c. プロンプトに対し、*y* (yes) と入力してローカライズ版ファイルをアップグレードします。

mqupgrade スクリプトは、次の場所にあるログファイルに出力を送信します。

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

## Message Queue のアップグレードの確認

アップグレード手順の完了後、`-version` オプションを使用して Message Queue ブローカを起動することにより、アップグレードが成功したことを確認します。

```
imqbrokerd -version
```

このコマンドは、Java ES のバージョン番号と、Message Queue 固有のバージョン番号を返します。

## アップグレード後の作業

Message Queue HTTP トンネリングサーブレットを使用しており、Web コンテナをアップグレードした場合、サーブレットを新しい Web コンテナに再配備しなければならない場合があります。Web コンテナをアップグレードしていない場合、Release 3 と Release 4 で HTTP トンネリングサーブレットの変更はないため、Message Queue を Release 4 にアップグレードしたあとにサーブレットを再配備する必要はありません。HTTP サポートの詳細については、『Sun Java System Message Queue 3 2005Q4 管理ガイド』(<http://docs.sun.com/source/819-3560?l=ja>) を参照してください。

## アップグレードのロールバック

Message Queue をアップグレード前の状態にロールバックするためのスクリプトは用意されていません。ロールバックは、次の手順に従って手動で実行する必要があります。

1. 稼働中の Message Queue クライアントアプリケーションをすべて停止します。
2. 稼働中のすべてのブローカを停止します。管理ユーザー名とパスワードの入力を求められます。

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

3. 各ブローカインスタンスに関連付けられた動的データ、Message Queue フラットファイルユーザーリポジトリ、および Message Queue アクセス制御ファイルを削除する場合、次のコマンドを使ってこれらのデータを削除します。

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

4. `root` としてログインするか、スーパーユーザーになります。

```
su -
```

- インストールされている Message Queue パッケージの一覧を、次のコマンドを使って取得します。

**Solaris:**

```
pkginfo | grep -i "message queue"
```

**Linux:**

```
rpm -qa | grep mq
```

- 次のコマンドを使って Message Queue パッケージを削除します。

**Solaris:**

```
pkgrm packageName
```

ここで、*packageName* は任意の Message Queue パッケージの名前です。複数のパッケージを削除するには、パッケージ名をスペースで区切って入力します。

**Linux:**

```
rpm -e --nodeps RPMName
```

ここで、*RPMName* は任意の Message Queue RPM コンポーネントの名前です。複数のコンポーネントを削除するには、RPM 名をスペースで区切って入力します。

ほかの製品が Message Queue パッケージを使用している可能性があるため、パッケージの削除は慎重に行なってください。パッケージに依存関係が存在する場合、pkgrm コマンドはパッケージを削除する前にその旨をユーザーに警告します。プロンプトが表示されたら、**y** (yes) と入力して削除要求を確認します。

- 「q」と入力して処理を終了します。
- root シェルを終了します。
- Release 3 の Message Queue を再インストールする  
Java ES Release 3 のインストーラを使用します。Release 4 の Message Queue データはそのまま使用できます。

## 複数インスタンスのアップグレード

Message Queue クラスタでは、複数のブローカを相互作用させることで、スケーラブルなメッセージサービスを実現できます。そのようなクラスタをアップグレードする場合、ローリングアップグレードと呼ばれる手法を使用できます。ローリングアップグレードでは、クラスタをオンラインに保ったまま、個々の Message Queue インスタンスを Release 3 から Release 4 に順次アップグレードできます。クラスタのアップグレードを実行するときは、次の 2 つの条件に留意してください。

- アップグレードのためにブローカをシャットダウンしている間、そのブローカが格納している永続メッセージは、ブローカが再起動されるまで取り出すことができません。
- マスターブローカは最後にアップグレードすることが推奨されます。

ローリングアップグレードを使用しない場合の手順は簡単です。すべてのブローカがアップグレードされるまで、ブローカを一度に1つずつシャットダウンし、アップグレードし、再起動します。

## Java ES Release 2 からの Message Queue のアップグレード

Java ES 2004Q2 (Release 2) の Message Queue を Release 4 にアップグレードする手順は、Release 3 の Message Queue を Release 4 にアップグレードする手順 (158 ページの「[Java ES Release 3 からの Message Queue のアップグレード](#)」を参照) とほとんど同じです。ただし、Release 2 からのアップグレードには、オペレーティングシステムプラットフォーム間で若干の違いがあります。

加えて、アップグレード前の作業に、Message Queue が依存するすべての共有コンポーネント (42 ページの表 1-6 を参照) を Release 2 バージョンから Release 4 にアップグレードする作業を含めることが推奨されます。

Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「[Java ES 共有コンポーネントのアップグレード](#)」で説明されています。

### Release 2 の Message Queue のアップグレード (Solaris)

158 ページの「[Java ES Release 3 からの Message Queue のアップグレード](#)」の手順説明に従います。ただし、「Release 3」の記述をすべて「Release 2」に読み替えます。

### Release 2 の Message Queue のアップグレード (Linux)

Linux システムで Release 2 から Release 4 にアップグレードする場合、Release 3 から Release 4 へのアップグレードでは不要なデータ移行作業 (特に、ブローカインスタンスデータを Release 4 の適切な位置に移行する作業) を行う必要があります。Release 4 へのアップグレードで Release 2 のデータを保持したい場合、Message Queue には、この移行を実行するための移行ツール `mqmigrate` が用意されています。

## アップグレード手順

Release 2 から Release 3 にアップグレードするには、158 ページの「[Java ES Release 3 からの Message Queue のアップグレード](#)」で使用したのと同じ手順を使用します。ただし、以降の手順内で説明するように、mqupgrade スクリプトを実行する前に mqmigrate スクリプトを実行する必要があります。

- 稼働中の Message Queue クライアントアプリケーションをすべて停止します。
- 稼働中のすべてのブローカを停止します。管理ユーザー名とパスワードの入力を求められます。

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

- 各ブローカインスタンスに関連付けられた動的データ、Message Queue フラットファイルユーザーリポジトリ、および Message Queue アクセス制御ファイルを保持しない場合、次のコマンドを使ってこれらのデータを削除します。

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

削除しない場合、動的データおよび設定情報は保持され、Release 4 の Message Queue で使用されます。

- root としてログインするか、スーパーユーザーになります。

```
su -
```

- Java ES 配布パッケージの Tools ディレクトリに移動します。

```
cd Linux_x86/Product/message_queue/Tools
```

- 次のコマンドを使ってブローカインスタンスデータを移行します。

```
./mqmigrate
```

mqmigrate スクリプトは、Release 2 のブローカインスタンス設定データを Release 4 の適切な位置に移動します。

- mqupgrade スクリプトを実行します。

- スクリプトを起動します。

```
./mqupgrade
```

mqupgrade スクリプトは、インストールされている Message Queue コンポーネントを一覧表示します。

- b. `y (yes)` と入力して、Message Queue コンポーネントをアップグレードします。

`mqupgrade` スクリプトは、インストールされているローカライズ版ファイルを検出して一覧表示します。

Message Queue コンポーネントをアップグレードしない場合は、`n (no)` と入力します。`mqupgrade` スクリプトは、Message Queue コンポーネントをアップグレードせずに終了します。

- c. プロンプトに対し、`y (yes)` と入力してローカライズ版ファイルをアップグレードします。

`mqupgrade` スクリプトは、次の場所にあるログファイルに出力を送信します。

```
/var/sadm/install/logs/Message_Queue_upgrade_'date' .log
```

## 互換性パッケージのインストール

Release 4 のインストール済みファイルの位置に依存するスクリプトがあるか、Release 2 のクライアントアプリケーションにそのようなスクリプトが含まれる場合、`sun-mq-compat` パッケージをインストールする必要があります。このパッケージには、Release 2 のファイル位置から Release 4 のファイル位置へのシンボリックリンクが含まれます。

`sun-mq-compat` パッケージは、Java ES 配布パッケージの展開先の次の位置にあります。

```
Linux_x86/Product/message_queue/Packages
```

次の手順を実行して、`sun-mq-compat` パッケージをインストールします。

1. `root` としてログインするか、スーパーユーザーになります。

```
su -
```

2. `Packages` ディレクトリから次のコマンドを入力します。

```
rpm -ivh --nodeps sun-mq-compat-3.6-RelNo.i386.rpm
```

## アップグレード後の作業

HTTP 接続サービスのサポートを提供するために HTTP トンネリングサーブレットを使用している場合、Release 2 から Release 4 への Message Queue のアップグレードによってサーブレットがアップグレードされています。そのため、Message Queue を Release 4 にアップグレードしたあとでサーブレットを再配備する必要があります。HTTP サポートの詳細については、『Sun Java System Message Queue 3 2005Q4 管理ガイド』(<http://docs.sun.com/source/819-3560?l=ja>) を参照してください。

管理コンソール (`imqadmin`) や `ObjectManager` コマンド行ユーティリティ (`imqobjmgr`) を使用して更新操作を実行し、Release 2 の管理対象オブジェクトを Release 4 バージョンに移行します。





# High Availability Session Store

この章では、High Availability Session Store を Java ES 2005Q4 (Release 4): High Availability Session Store (HADB) 4.4.2 にアップグレードする方法について説明します。

この章では、アップグレードに伴う一般的な問題の概要を示し、Java ES Release 4 でサポートされている各種のアップグレードパスについて説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [170 ページの「HADB のアップグレードの概要」](#)
- [172 ページの「Java ES Release 3 からの HADB のアップグレード」](#)

---

## 注

この章で説明するファイルの場所は、*HADB-base* というディレクトリパスを基準に指定されています。少なくとも、このパスの一部は、HADB が最初にインストールされたときにインストールディレクトリとして指定されている場合があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*HADB-base* の値は、次に示すように、Application Server のディレクトリ構造を起点とした相対パスです。

*AppServer8-base*/hadb/*version\_number*

*AppServer8-base* のデフォルト値はオペレーティングシステムプラットフォームによって異なり、*HADB-base* のデフォルト値もそれに応じて異なります。

- Solaris: /opt/SUNWappserver/appserver/hadb/*version\_number*
  - Linux: /opt/sun/appserver/hadb/*version\_number*
-

# HADB のアップグレードの概要

この節では、Java ES 2005Q4 (Release 4) へのアップグレードに影響を及ぼす、HADB の一般的な側面について説明します。

- [Java ES Release 4 の HADB について](#)
- [HADB のアップグレードのロードマップ](#)
- [HADB のデータ](#)
- [互換性の問題](#)
- [HADB の依存関係](#)

## Java ES Release 4 の HADB について

Java ES Release 4 バージョンの HADB では、Java ES 2005Q1 (Release 3) バージョンのバグ修正が行われています。

## HADB のアップグレードのロードマップ

表 8-1 は、HADB について、Java ES Release 4 へのサポートされているアップグレードパスを示したものです。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 8-1 Java ES Release 4: HADB 4.4.2 (2005Q4) へのアップグレードパス

Java ES のリリース	HADB のバージョン	一般的な手順	必要な再設定
Release 3	HADB 4.4.1 (2005Q1)	直接アップグレード: オンラインアップグレードとオフラインアップグレードの両方が可能	なし
Release 2	HADB 4.4.0-14 (2004Q2)	アップグレードはサポート対象外	なし
Release 1	なし	アップグレード不可	なし
それ以前の Java ES リリース	なし	アップグレード不可	なし

## HADB のデータ

次の表は、HADB ソフトウェアのアップグレードによって影響を受ける可能性があるデータの種類を示したものです。

表 8-2 HADB のデータの使用方法

データの種類	場所	使用法
動的なアプリケーションデータ	<code>/var/opt/SUNWhadb</code> and <code>/etc/opt/SUNWhadb</code>	高可用性セッションストアおよび設定情報

## 互換性の問題

Java ES Release 4 で提供される HADB は、Java ES Release 3 で提供される HADB と下位互換性があります。

## HADB の依存関係

Java ES Release 4 で提供される HADB を実行するには、Java™ 2 Platform, Standard Edition (J2SE™) バージョン 5.0 以降が必要です。

# Java ES Release 3 からの HADB のアップグレード

この節では、Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) への HADB のアップグレードに関する情報を示します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の HADB のアップグレード](#)

## はじめに

Java ES Release 3 の HADB を Java ES Release 4 にアップグレードするときは、アップグレードプロセスの次の側面を考慮してください。

- **一般的なアップグレード手順** : アップグレードは、Java ES Release 3 の HADB パッケージを削除し、Java ES Release 4 のパッケージを追加することによって行います。アップグレードは 2 とおりの方法で実行できます。
  - **オンラインアップグレード** : HADB サービスの中断を避けるには、オンラインアップグレードを使用します。
  - **オフラインアップグレード** : HADB パッケージを新しいバージョンに置き換えるときに HADB サービスを中断できる場合、オフラインアップグレードを使用します。
- **アップグレードの依存関係** : HADB には J2SE バージョン 5.0 以降が必要です。
- **下位互換性** : Java ES Release 4 で提供される HADB は、Java ES Release 3 で提供される HADB と下位互換性があります。
- **アップグレードのロールバック** : Java ES Release 4 へのアップグレードから Java ES Release 3 へのロールバックは、Release 4 へのアップグレード時に別のディレクトリに残しておいた Release 3 バージョンを復元することによって行います。
- **プラットフォームの問題** : HADB をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通です。

## Release 3 の HADB のアップグレード

この節では、Solaris および Linux プラットフォームで、HADB を Java ES Release 3 から Java ES Release 4 にアップグレードする方法について説明します。この節で説明する内容は、次のとおりです。

- アップグレード前の作業
- Release 3 の HADB のアップグレード
- アップグレードの確認
- アップグレード後の作業
- アップグレードのロールバック

### アップグレード前の作業

HADB をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

HADB の現在のバージョンは、標準のバージョンチェックユーティリティを使用することにより確認できます。次に例を示します。

Solaris:

```
pkgparam -v SUNWhadba
```

Linux:

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

表 8-3 HADB のバージョン確認の出力

Java ES のリリース	HADB のバージョン番号
Release 2	VERSION=4.4.0,REV=14 SUNW_PRODVERS=4.4.0
Release 3	VERSION=4.4.1,REV=7 SUNW_PRODVERS=4.4.1
Release 4	VERSION=4.4.2,REV=7 SUNW_PRODVERS=4.4.2

### HADB が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。HADB を Release 4 にアップグレードするには J2SE 5.0 以降が必要です。

### ディレクトリデータのバックアップ

HADB を Java ES Release 3 から Java ES Release 4 にアップグレードしても、HADB の動的データは変更されません。ただし、アップグレードをロールバックする必要がある場合に備えて、Java ES Release 3 のパッケージをバックアップすることができます。

### 必要な設定情報とパスワードの入手

HADB のアップグレードには、スーパーユーザーのパスワードが必要です。

## Release 3 の HADB のアップグレード

この節では、HADB のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点

HADB ソフトウェアを Java ES Release 4 にアップグレードするときは、次の注意点を考慮に入れてください。

- 運用環境の要件に基づいて、オンラインアップグレードとオフラインアップグレードのどちらが適切かを決定する必要があります。
- 次の表に、Solaris および Linux の各プラットフォーム用の、Java ES Release 4 アップグレードパッケージを示します。Solaris 用のパッケージは、インストールする順に示しています。

表 8-4 Solaris プラットフォーム用 HADB アップグレードパッケージのバージョン

Solaris 用パッケージ	Linux 用パッケージ
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 のオンラインアップグレード

HADB のオンラインアップグレードは、Java ES Release 3 からのアップグレード時のみ実行できます。

HADB のオンラインアップグレードを実行するときは、まず、アップグレード対象のクラスタ内の各サーバーに HADB をインストールします。各サーバーはまず、HADB の以前のインストールから登録解除し、次に、新しくインストールされるバージョンの HADB に登録します。

オンラインアップグレードの実行の詳細については、『Sun Java System Application Server Enterprise Edition 8.1 2005Q2 高可用性 (HA) 管理ガイド』 (<http://docs.sun.com/doc/819-4955?l=ja>) の次の節を参照してください。

## HADB のオフラインアップグレード

HADB のオフラインアップグレードは、Java ES Release 3 からのアップグレード時に実行できます。

オフラインアップグレードを実行するには、HADB サービスをシャットダウンし、既存の HADB パッケージを新しいバージョンと置き換えます。この新しいバージョンは Java ES 配布パッケージに収められており、174 ページの表 8-4 にその一覧を示しています。

75 ページの「Solaris プラットフォームのパッケージのアップグレード」および 79 ページの「Linux プラットフォームのパッケージのアップグレード」で説明されている手順に従います。

## アップグレードの確認

オンラインアップグレードの完了後、次の手順を使用してアップグレードを確認します。アップグレードの成功を確認したら、古いインストールパッケージは削除できません。

稼働中のプロセスがアップグレードされた HADB サービスを使用していることを確認するために、次の手順を実行できます。

1. 稼働中のすべての HADB サービスに対して、次のいずれかのコマンドを発行します。

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

たとえば、Solaris 8 プラットフォームで、次のコマンドを実行します。

```
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. ExampleDB という名前のデータベースに対して、次の例のコマンドを発行することにより、データベースが稼働しているかどうかをチェックします。

```
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

一覧表示されたノードに対して、すべての HADB サービスが「**running**」状態であることを確認します。

3. ExampleDB という名前のデータベースに対して、次の例のコマンドを発行することにより、HADB を使用するすべての製品が新しい HADB パスを使用していることを確認します。

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

```
Attribute      Value
PackageName    V4.4.2.7
```

このコマンドは、HADB の現在のバージョンを表示します。詳細な一覧表示を得るには、次のコマンドを発行します。

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

## アップグレード後の作業

172 ページの「[Java ES Release 3 からの HADB のアップグレード](#)」で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック

HADB へのアップグレードをロールバックするには、インストールした HADB パッケージの新しいバージョンを、以前にインストールしたバージョンと置き換えます。

176 ページの「[アップグレードのロールバック](#)」の説明と同じ手順を使用します。

データまたは設定ファイルの変更は必要ありません。



# Application Server

この章では、Application Server を Java ES 2005Q4 (Release 4): Sun Java System Application Server Enterprise Edition 8.1 2005Q4 にアップグレードする方法について説明します。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [178 ページの「Application Server のアップグレードの概要」](#)
- [181 ページの「Java ES Release 3 からの Application Server のアップグレード」](#)
- [189 ページの「Java ES Release 2 からの Application Server のアップグレード」](#)

---

**注** この章では、*AppServer8-base* および *AppServer8Config-base* (Application Server 8.1 の場合)、または *AppServer7-base* および *AppServer7Config-base* (Application Server 7 の場合) というディレクトリパスを基準に、ファイルの場所が指定されています。これらのパスの少なくとも一部は、Application Server がインストールされたときにインストールディレクトリとして指定されている場合があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

次の表に、これらのディレクトリパスのデフォルト値を示します。

---

パス名	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 のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Application Server の次の一般的な側面について説明します。

- [Java ES Release 4 の Application Server について](#)
- [Application Server のアップグレードロードマップ](#)
- [Application Server のデータ](#)
- [互換性の問題](#)
- [Application Server の依存関係](#)

## Java ES Release 4 の Application Server について

Java ES Release 4 の Application Server では、Release 3 バージョンのバグ修正が行われています。機能的には、Release 4 の Application Server は Release 3 と同じです。

## Application Server のアップグレードロードマップ

表 9-1 は、Application Server の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 9-1 Java ES Release 4: Sun Java System Application Server Enterprise Edition 8.1 2005Q4 へのアップグレードパス

Java ES のリリース	Application Server バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Application Server Enterprise Edition 8.1 2005Q1	直接アップグレード: パッチを適用することにより実行されます。	なし
Release 2	Sun Java System Application Server 7.0 Upgrade 3 (2004Q2) Platform Edition および Enterprise Edition	直接アップグレード: まず Java ES インストーラを使用し、次に再設定ユーティリティを使用します。	環境変数とその他の設定データ。  J2EE コンポーネントおよび J2EE アプリケーションを新しい Application Server 環境に移行し、再配備する必要があります。

表 9-1 Java ES Release 4: Sun Java System Application Server Enterprise Edition 8.1 2005Q4 へのアップグレードパス (続き)

Java ES のリリース	Application Server バージョン	一般的な手順	必要な再設定
Release 1	Sun ONE Application Server 7.0 Upgrade 1 (2003Q4) Platform Edition および Enterprise Edition	直接アップグレードは動作保証なし: ただし、まず Java ES インストーラを使用し、次に再設定ユーティリティを使用する方法でアップグレードが可能です。	環境変数とその他の設定データ。  J2EE コンポーネントおよび J2EE アプリケーションを新しい Application Server 環境に移行し、再配備する必要があります。
それ以前の Java ES リリース		直接アップグレード不可: まず Release 3 にアップグレードする必要があります。手順については、『Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。  次に、Release 3 から Release 4 にアップグレードします。	

表 9-1 に示した Java ES リリースの Application Server に加えて、Application Server Platform Edition は Solaris オペレーティングシステムソフトウェアにも付属します。付属するバージョンの Application Server は、Java ES インストーラを使用して Release 4 Enterprise Edition にアップグレードできます。このアップグレードを行うには、インストーラで Application Server の新規インストールを選択します。Platform Edition からのアップグレードと、Application Server の必要な再設定がインストーラによって自動的に行われます。

## Application Server のデータ

次の表は、Application Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 9-2 Application Server のデータの使用方法

データの種類	場所	使用法
環境変数	<i>AppServer8-base/config/asenv</i>	グローバル変数
設定データ	Release 3 および Release 4: 次の場所にある <i>domain.xml</i> および <i>server.policy</i> ファイル <i>AppServer8Config-base/domains/domainName/config</i>  Release 2: 次の場所にある <i>server.xml</i> および <i>server.policy</i> ファイル <i>AppServer7Config-base/domains/domainName/instanceName/config</i>	Application Server インスタンスの設定
配備データ	Release 3 および Release 4: <i>AppServer8Config-base/domains/domainName/applications</i>  Release 2: <i>AppServer7Config-base/domains/domainName/instanceName/applications</i>	特定の J2EE コンポーネントおよび J2EE アプリケーション用の J2EE コンテナの設定

## 互換性の問題

Release 4 の Application Server では、Release 3 からのインタフェースの変更はありません。ただし、Release 4 と Release 2 ではインタフェースが大きく変更されているため、Release 4 は Release 2 と互換性がありません。

## Application Server の依存関係

ほかの Java ES コンポーネントに対する Application Server の依存関係が、Application Server ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Application Server のインタフェースまたは機能の変更が原因で、Application Server が依存するコンポーネントのバージョンのアップグレードが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Application Server は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Application Server ソフトウェアは、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Message Queue**: Application Server は、J2EE Java Message Service 互換の非同期メッセージングサポートを提供するために Message Queue を利用します。
- **Web コンテナ (省略可能)**: Application Server でオプションのロードバランスプラグインを使用する場合、Web コンテナが必要です。Web コンテナには、Java ES Web Server またはサードパーティー製 Web コンテナ (Apache Web Server や Microsoft IIS など) を使用できます。
- **High Availability Session Store (省略可能)**: Application Server で、インスタンス間のフェイルオーバーをサポートするために必要なセッション状態情報を維持するためには High Availability Session Store が必要です。

## Java ES Release 3 からの Application Server のアップグレード

ここでは、Application Server の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Application Server のアップグレード](#)

### はじめに

Java ES Release 3 Application Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。Release 3 の Application Server から Release 4 へのアップグレードでは、J2EE コンポーネントの再設定または移行は必要ありません。
- **アップグレードの依存関係** : Application Server は多くの Java ES 共有コンポーネントに対する依存関係を持ちますが (42 ページの表 1-6 を参照)、Release 4 の Application Server はこれらすべてのコンポーネントの Release 3 バージョンと互換性があります。したがって、これらの共有コンポーネントのアップグレードは、Application Server の Release 4 へのアップグレードに関しては省略可能です。

Release 4 の Application Server の動作には、Release 4 の Message Queue が必要です。また、使用する機能によっては、Java ES Web Server またはサードパーティー製の Web コンテナも必要です (180 ページの「[Application Server の依存関係](#)」を参照)。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Application Server の Release 4 へのアップグレードに関しては省略可能です。

Release 4 の Application Server の動作には、High Availability Session Store も必要な場合があります。Application Server によって利用される場合、High Availability Session Store を Release 4 にアップグレードすることが推奨されます。High Availability Session Store をアップグレードすると、J2SE 共有コンポーネントも自動的に Release 4 にアップグレードされることに注意してください。

- **下位互換性** : Release 4 Application Server は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Release 4 アップグレードの Release 3 へのロールバックは、アップグレード中に適用されたパッチを削除することにより行われます。
- **プラットフォームの問題** : Application Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Application Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Application Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Application Server のアップグレード \(Solaris\)](#)
- [Release 3 の Application Server のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

## アップグレード前の作業

Application Server をアップグレードする前に、次に説明されている作業を実行してください。

### 現在のバージョン情報の確認

Application Server の現在のバージョンは、次のコマンドを入力することにより確認できます。

```
AppServer8-base/bin/asadmin version --verbose
```

表 9-3 Application Server のバージョン確認の出力

Java ES のリリース	Application Server のバージョン番号
Release 2	Sun ONE Application Server 7.0.0_03c
Release 3	Sun Java Enterprise System Application Server Enterprise Edition 8.0.0_01
Release 4	Sun Java Enterprise System Application Server Enterprise Edition

### Application Server が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。ただし、Release 4 への Application Server のアップグレードではほかの Release 3 コンポーネントのアップグレードは必須ではないため、この作業は省略可能です。Application Server が依存するコンポーネントをアップグレードする場合、それらのコンポーネントのアップグレードは通常、次の順序で行います。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの「[Java ES 共有コンポーネントのアップグレード](#)」で説明されています。
2. **Message Queue** : Message Queue を Release 4 にアップグレードする手順は、151 ページの第 7 章「[Message Queue](#)」で説明されています。
3. **Web コンテナソフトウェア ( 省略可能 )** : Web Server または Application Server をアップグレードする手順は、それぞれ 141 ページの第 6 章「[Web Server](#)」および 177 ページの第 9 章「[Application Server](#)」で説明されています。
4. **High Availability Session Store ( 省略可能 )** : High Availability Session Store をアップグレードする手順は、169 ページの第 8 章「[High Availability Session Store](#)」で示しています。

## Application Server のデータのバックアップ

Application Server の Release 3 から Release 4 へのアップグレードでは、設定データは変更されません。現在のデータをバックアップする必要はありません。

## 必要な設定情報とパスワードの入手

現在インストールされているバージョンの Application Server の管理者ユーザー ID とパスワードを調べておきます。

## Release 3 の Application Server のアップグレード (Solaris)

ここでは、Application Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Application Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Application Server インスタンスをアップグレードする前に、そのインスタンス内で稼働しているすべての J2EE コンポーネントをシャットダウンすることが推奨されます。ただし、高可用性またはスケーラビリティを実現するためにロードバランスを使用している場合、この要件は緩和できます。
- 1 台のコンピュータ上で Application Server の複数のインスタンスが稼働しており、すべてのインスタンスがインストール済みの同じ Application Server イメージに対応している場合、そのイメージにパッチを適用している間はそれらのインスタンスをシャットダウンする必要があります。
- 複数ノード配備の場合、Application Server インスタンスが動作している個々のノードまたはコンピュータ上でアップグレード手順を実行します。
- Solaris OS での Release 4 Application Server のアップグレードパッチを次の表に示します。

表 9-4 Solaris で Application Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Application Server	119166-10	119167-10
Application Server ローカライズ版	119024-10	119025-10

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。



## アップグレード手順 (Solaris)

ここで説明する手順は、アップグレードを実施するコンピュータにローカルインストールされている Application Server インスタンスに適用されます。

1. 表 9-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. アップグレードする Application Server インスタンス内のすべての J2EE コンポーネントをシャットダウンします。
3. アップグレードするコンピュータ上のすべての Application Server インスタンスをシャットダウンします。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 表 9-4 の適切な Application Server のパッチを適用します。

```
patchadd patch_ID
```

5. asant スクリプトを修正します。

- a. 既存の asant スクリプトの名前を asant.bak に変更します。

スクリプトは次の場所にあります。

```
AppServer8-base/bin/asant
```

- b. 次の場所にある asant.template ファイルをコピーします。

```
AppServer8-base/lib/install/templates/ee
```

コピー先は次のとおりです。

```
AppServer8-base/bin/asant
```

- c. スクリプトを編集します。

%CONFIG\_HOME% トークンを *AppServer8-base/config* に置き換えます。

- d. 元のスクリプトファイル (asant.bak) に手動で何らかの変更を行った場合、新しい asant スクリプトにその変更を反映します。

6. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 4 で適用したパッチ ID のバージョンが返されるはずですが。

7. Application Server インスタンスを再起動します。

```
AppServer8-base/bin/asadmin start-domain domainName
```

## Release 3 の Application Server のアップグレード (Linux)

ここでは、Application Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Application Server ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (184 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Application Server のアップグレードパッチを次の表に示します。

表 9-5 Linux で Application Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
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
Application Server ローカライズ版	119026-10  sun-asacee-Locale-8.1.1-51.i386.rpm sun-ascmnse-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. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

ここで説明する手順は、アップグレードを実施するコンピュータにローカルインストールされている Application Server インスタンスに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 9-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. アップグレードする Application Server インスタンス内のすべての J2EE コンポーネントをシャットダウンします。
3. アップグレードを実施するコンピュータ上のすべての Application Server インスタンスをシャットダウンします。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. 次のファイルをバックアップします。
  - *AppServer8-base/pointbase/tools/serveroption* 下のすべてのファイル
  - *AppServer8-base/samples/common.properties*
5. 表 9-5 の Application Server の RPM を適用します。

```
rpm -Fvh sun-asmodule-8.1.2-10.i386.rpm
rpm -Fvh sun-asmodule-Locale-8.1.1-51.i386.rpm
```

6. 手順 4 でバックアップしたファイルを元の場所に復元します。
  - *AppServer8-base/pointbase/tools/serveroption* 下のすべてのファイル
  - *AppServer8-base/samples/common.properties*
7. asant スクリプトを修正します。

- a. 既存の asant スクリプトの名前を asant.bak に変更します。

スクリプトは次の場所にあります。

```
AppServer8-base/bin/asant
```

- b. 次の場所にある asant.template ファイルをコピーします。

```
AppServer8-base/lib/install/templates/ee
```

コピー先は次のとおりです。

```
AppServer8-base/bin/asant
```

- c. スクリプトを編集します。

%CONFIG\_HOME% トークンを *AppServer8-base/config* に置き換えます。

- d. 元のスクリプトファイル (`asant.bak`) に手動で何らかの変更を行った場合、新しい `asant` スクリプトにその変更を反映します。
8. パッチのアップグレードが成功したことを確認します。  

```
rpm -qa | grep sun-as
```

RPM の新しいバージョン番号が返されることを確認します。
9. Application Server インスタンスを再起動します。  

```
AppServer8-base/bin/asadmin start-domain domainName
```

## アップグレードの確認

パッチが正しく適用されたかどうかは、次のコマンドを使用して確認できます。

```
AppServer8-base/bin/asadmin version --verbose
```

出力値については、[183 ページの表 9-3](#) を参照してください。

## アップグレード後の作業

[185 ページの「アップグレード手順 \(Solaris\)」](#) および [186 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Application Server のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Application Server の Release 4 へのアップグレードをロールバックするには、Release 4 へのアップグレードと逆の手順を実行します。具体的には、パッチを削除します。

### ロールバック手順 (Solaris)

1. アップグレードする Application Server インスタンス内で稼働しているすべての J2EE コンポーネントをシャットダウンします。
2. アップグレードする Application Server インスタンスをシャットダウンします。
3. [表 9-4](#) のパッチを削除します。  

```
patchrm patch_ID
```
4. Application Server インスタンスを再起動します。

# Java ES Release 2 からの Application Server のアップグレード

ここでは、Java ES Release 2 から Java ES 2005Q4 (Release 4) への Application Server のアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 2 の Application Server のアップグレード](#)

## はじめに

Java ES Release 2 Application Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : Java ES インストーラで「あとで設定」オプションを選択して Release 4 の Application Server をインストールすることによってアップグレードを実行します。その後、asupgrade ユーティリティを使用して再設定を行います。Application Server をアップグレードしたあとで、Release 2 の J2EE コンポーネントと J2EE アプリケーションを Release 4 に移行する必要があります。
- **アップグレードの依存関係** : コンピュータ上の任意の Java ES コンポーネントを Release 2 からアップグレードするには、そのコンピュータでホストされているほかのすべての Java ES コンポーネントをアップグレードする必要があります。Java ES コンポーネントの Release 2 から Release 4 への選択的なアップグレードはサポートされていません。特に、Application Server によって使用されるすべての Java ES 共有コンポーネントをアップグレードする必要があります。同じコンピュータに Message Queue がインストールされている場合や、負荷分散用に Web Server を使用している場合、これらのコンポーネントもアップグレードする必要があります。

Application Server で使用している場合、High Availability Session Store も Release 4 にアップグレードすることが推奨されます。ただし、High Availability Session Store は Release 2 の Application Server に統合されているため、単独で Release 4 にアップグレードすることはできません。

- **下位互換性** : Release 4 の Application Server は、Release 2 バージョンとの下位互換性はありません。Release 4 の Application Server 環境で実行するためには、J2EE コンポーネントおよび J2EE アプリケーションを移行する必要があります。
- **アップグレードのロールバック** : Release 4 へのアップグレードは、Release 2 のインストールを復元するだけで Release 2 にロールバックすることができます。アップグレードプロセスでは、Release 2 の設定データは削除されません。
- **プラットフォームの問題** : Application Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通です。

## Release 2 の Application Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Application Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- アップグレード前の作業
- [Release 3 の Application Server のアップグレード \(Solaris\)](#)
- アップグレードの確認
- アップグレード後の作業
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Application Server をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Application Server の現在のバージョンは、次のコマンドを入力することにより確認できます。

```
AppServer7-base/bin/asadmin version --verbose
```

出力されるバージョン情報については、[183 ページの表 9-3](#) を参照してください。

### Application Server が依存するコンポーネントのアップグレード

Application Server が依存するコンポーネントのアップグレードでは、ローカルにインストールされ、Application Server が依存するすべての製品コンポーネントを Release 4 にアップグレードすることが推奨されます。共有コンポーネントは、アップグレード手順の一部として、Java ES インストーラによって自動的にアップグレードされます ([191 ページの手順 3](#) を参照)。

Application Server の依存関係をアップグレードする場合は、Application Server をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

- **Message Queue:** [151 ページの第 7 章「Message Queue」](#) を参照
- **Web Server (省略可能):** [141 ページの第 6 章「Web Server」](#) を参照

## Application Server のデータのバックアップ

Application Server を Release 2 から Release 4 にアップグレードしても、Release 2 の設定データは上書きされません。ただし、不測の事態に備えて、Release 4 へのアップグレードを実行する前に、すべての Application Server インスタンスの設定ディレクトリをバックアップすることが推奨されます。

設定ディレクトリの場所は次の場所にあります。

```
AppServer7-base/domains/domainName/instanceName/config
```

## 必要な設定情報とパスワードの入手

現在インストールされているバージョンに関して、次の情報を確認してください。

- Application Server の管理者ユーザー ID、パスワード、およびマスターパスワード
- Release 2 の Application Server のベースディレクトリ

## Release 2 の Application Server のアップグレード

ここでは、Application Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点

Application Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Application Server インスタンスをアップグレードする前に、そのインスタンス内で稼働しているすべての J2EE コンポーネントをシャットダウンすることが推奨されます。ただし、ロードバランスマグジンを使用して高可用性またはスケールビリティを提供する場合、この要件は緩和できる場合があります。
- 1 台のコンピュータ上で複数の Application Server インスタンスが稼働しており、すべてのインスタンスがインストール済みの同じ Application Server イメージに対応している場合、インストール済みのイメージをアップグレードしている間はすべてのインスタンスをシャットダウンする必要があります。

### アップグレード手順

次に示す手順は、アップグレードを実施するコンピュータにローカルインストールされているすべての Application Server インスタンスに適用されます。

1. root としてログインするか、スーパーユーザーになります。  
su -
2. すべての Application Server と関連プロセスを停止します。
3. Java ES インストーラを使用し、「あとで設定」オプションを選択して Release 4 の Application Server をインストールします。

最低限、「ノードエージェント」コンポーネントを含む最初の3つのサブコンポーネントをインストールするように選択します。

Application Server ソフトウェアをインストールしたあとで必ず、194 ページの「[あとで設定](#)」インストール後の Application Server の設定」で説明しているインストール後の作業を実行します。

4. インストールのターゲットディレクトリとソースディレクトリの両方を識別します。次に例を示します。
  - Solaris 上でのデフォルトの Release 2 ソース : /opt/SUNWappserver7
  - Solaris 上でのデフォルトの Release 4 ターゲット : /opt/SUNWappserver/appserver
5. asupgrade ユーティリティを実行します。

asupgrade ユーティリティは、Release 4 のノードエージェントを作成し、その下に Release 2 の Application Server インスタンスを移行します。

ユーティリティは Application Server ディレクトリの配下にあります。次に例を示します。

- アップグレードウィザードモード : *AppServer8-base*/bin/asupgrade
- アップグレードコンソールモード : *AppServer8-base*/bin/asupgrade -c

アップグレードウィザードまたはアップグレードコンソールによって、アップグレード手順が案内されます。

Application Server の asupgrade ユーティリティの詳細については、『Application Server Enterprise Edition 8.1 Upgrade and Migration Guide 2005Q1』(<http://docs.sun.com/doc/819-0222>) の第3章を参照してください。

6. ドメイン管理サーバー (DAS) を起動します。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

7. アップグレードした Application Server インスタンスを再起動します。

これは、アップグレードした Application Server がその下に移行されたノードエージェントを起動することによって行います。

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

ここで、*nodeagentName* の形式は *hostName\_domainName* です。デフォルトの *domainName* は domain1 です。

## アップグレードの確認

管理コンソールを起動し、これらのサーバーが起動されていることを確認します。稼働していないサーバーがある場合、次のログをチェックして、ポートの競合によるエラーが発生していないかを調べます。



```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/logs/server.log
```

ポートの競合によるエラーが発生していた場合、管理コンソールを使ってポート番号を変更して競合を解消したあと、ノードエージェントをいったん停止し、再起動します。

Release 4 への Application Server のアップグレードが成功したかどうかは、次のコマンドを入力することによって確認できます。

```
AppServer8-base/bin/asadmin version --verbose
```

出力値については、[183 ページの表 9-3](#) を参照してください。

## アップグレード後の作業

[191 ページの「アップグレード手順」](#) で説明している手順以外に、アップグレード後の作業がいくつかあります。これには、Release 2 の J2EE コンポーネントおよび J2EE アプリケーションを Release 4 の Application Server 環境で動作するように移行し、適切な Application Server インスタンスにそれらを再配備する作業が含まれます。

J2EE コンポーネントと J2EE アプリケーションの移行の詳細については、『Application Server Enterprise Edition 8.1 Upgrade and Migration Guide 2005Q1』(<http://docs.sun.com/doc/819-0222>) の第 4 章を参照してください。

## アップグレードのロールバック

Application Server の Release 4 へのアップグレードは、Release 4 へのアップグレードによって削除されなかった Application Server の Release 2 バージョンを復元するだけでロールバックできます。

## 複数インスタンス ( クラスタ ) のアップグレード

Application Server の `asupgrade` ユーティリティを使用すると、複数のインスタンスで構成されるクラスタをアップグレードできます。手順については、『Application Server Enterprise Edition 8.1 2005Q2 Upgrade and Migration Guide』(<http://docs.sun.com/doc/819-2559>) の第 3 章を参照してください。

## 「あとで設定」インストール後の Application Server の設定

「あとで設定」オプションでのインストール後、Application Server 環境を設定するためのスクリプトを実行する必要があります。次の手順を実行します。

1. Application Server のアクセサリ配布パッケージの位置を特定します。

Sun Java Enterprise System 2005Q4 アクセサリ CD 1、  
Solaris SPARC および x86 用 Application Server アドオン、  
CD イメージ (1 / 1)

アクセサリの内容は、Sun Download Center

(<http://www.sun.com/software/javaenterprisesystem/get.xml>) からダウンロード可能です。

2. アクセサリ配布パッケージ内の Addon フォルダ内の ReadMe ファイルを参照して、指定された手順を実行します。

- a. `postInstall` スクリプトを実行します。

Addon フォルダ内のメインスクリプトである `postInstall` は、アクセサリの配布パッケージから実行してください。そうしないと、一部のファイルが見つからない場合があります。

スクリプトは、インストールの間にインストールされたテンプレートを基に、`AppServer8-base/bin/*` シェルスクリプトおよび `config/asenv` ファイルを構成および作成します。通常、`bin/*` シェルスクリプトはインストーラによって作成されますが、「あとで設定」オプションを選択した場合は説明に従って作成する必要があります。

- b. 新しいドメインを作成します。

`asadmin create-domain` コマンドを使用して新しいドメインを作成するとき、`adminPort` と `instancePort` の 2 つのパラメータの値を指定します。`adminPort` の値は、Release 2 のサーバーインスタンスによって使用されるものと同じ値に設定できますが、`instancePort` の値は、Release 2 のどのサーバーインスタンスも使用していない値に設定することが推奨されます。使用されていない `instancePort` の値を選択することにより、Release 4 の DAS インスタンスと、Release 4 に移行される Release 2 のサーバーインスタンスの間で競合が回避されます (192 ページの手順 5 を参照)。

3. 必要に応じて、`AppServer8-base/config/asenv` ファイル内の環境設定を変更します。

ファイルを手動で編集する必要があります。

---

注 負荷分散のために Application Server を設定する方法については、『Sun Java System Application Server Enterprise Edition High Availability Administration Guide』(<http://docs.sun.com/doc/819-0216>)の「Application Server High Availability Features」の章の「Configuring Web Servers for HTTP Load Balancing」を参照してください。

---



# Web Proxy Server

この章では、Web Proxy Server を Java ES 2005Q4 (Release 4): Sun Java System Web Proxy Server 4.0 2005Q4 にアップグレードする方法について説明します。

この章では、アップグレードに伴う一般的な問題の概要を示し、アップグレードの手順を説明します。Web Proxy Server のアップグレードは Solaris プラットフォームでのみ行うことができます。

- [197 ページの「Web Proxy Server のアップグレードの概要」](#)
- [200 ページの「Release 4 への Web Proxy Server のアップグレード」](#)

---

**注** この章では、*WebProxyServer-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Web Server が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*WebProxyServer-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWproxy
  - Linux: /opt/sun/webproxyserver
- 

## Web Proxy Server のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Web Proxy Server の次の一般的な側面について説明します。

- [Java ES Release 4 の Web Proxy Server について](#)
- [Web Proxy Server のアップグレードロードマップ](#)
- [Web Proxy Server のデータ](#)

- 互換性の問題
- Web Proxy Server の依存関係

## Java ES Release 4 の Web Proxy Server について

Java ES Release 4 の Web Proxy Server の特長には、パフォーマンスの向上、よりスケラブルなアーキテクチャー、標準への準拠度の向上、Java Enterprise System に組み込まれる前の Sun One Web Proxy Server 3.6 から一新された新しい管理インタフェースなどがあります。

## Web Proxy Server のアップグレードロードマップ

表 10-1 に、Java ES Release 4 への Web Proxy Server のアップグレードパスを示します。Web Proxy Server は以前の Java ES リリースには含まれていませんでした。Web Proxy Server はこれまで Linux オペレーティングシステムではサポートされていなかったため、この表は Solaris オペレーティングシステムにのみ対応します。

表 10-1 Java ES Release 4: Sun Java System 4: Web Proxy Server 4.0.1 2005Q4 へのアップグレードパス

Java ES のリリース	Web Proxy Server バージョン	一般的な手順	必要な再設定
それ以前の Java ES リリース	Sun ONE Web Proxy Server 3.6 (以後 Version 3.6 と表記)	直接アップグレード: Java ES インストーラを使用して新しい場所にインストールし、管理ツールを使用して設定データを移行することによって行います。	設定情報を新しい場所に移行する必要があります。

## Web Proxy Server のデータ

次の表は、Web Proxy Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 10-2 Web Proxy Server のデータの使用方法

データの種類	場所	使用法
設定データ	<i>WebProxyServer-base/proxy-serverid/</i> config ディレクトリ  含まれるファイルの例: <i>server.xml</i> 、 <i>magnus.conf</i> 、 <i>obj.conf</i> など	サーバー、キャッシュ、フィルタ、ルーティ ング、および Web Proxy Server のその他の機 能に関連する設定情報を格納します

## 互換性の問題

Release 4 の Web Proxy Server では、Version 3.6 でサポートされていた Netscape Server API (NSAPI) インタフェースが大幅に変更されています。Version 3.6 用に開発されたすべての NSAPI プラグインを、NSAPI の現在のバージョンに適合するように再コンパイルして非互換性を解決する必要があります。

## Web Proxy Server の依存関係

ほかの Java ES コンポーネントに対する Web Proxy Server の依存関係が、Web Proxy Server ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。

Web Proxy Server は特定の Java ES 共有コンポーネントへの依存関係を持ちますが (42 ページの表 1-6 を参照)、ほかの Java ES 製品コンポーネントへの依存関係はありません。Directory Server、Web Server、および Application Server と組み合わせて使用できますが、これらのコンポーネントに対する依存関係はありません。

# Release 4 への Web Proxy Server のアップグレード

この節では、Version 3.6 から Java ES 2005Q4 (Release 4) への Web Proxy Server のアップグレードに関する情報を示します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Web Proxy Server のアップグレード](#)

## はじめに

Web Proxy Server を Release 4 にアップグレードするときは、アップグレードプロセスに関する次の事項を考慮してください。

- **一般的なアップグレード手順** : アップグレードは、Java ES インストーラを使用して、Version 3.6 とは異なるディレクトリに Release 4 の Web Proxy Server をインストールすることによって行います。その後、Web Proxy Server 管理サーバーを使用して、キャッシュされた内容を除く設定情報を Version 3.6 から Release 4 に移行します。
- **アップグレードの依存関係** : Web Proxy Server は多くの Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Release 4 の Web Proxy Server はこれらのコンポーネントの Release 3 バージョンと互換性があります。ただし、これらの共有コンポーネントのアップグレードは、Web Proxy Server を Release 4 にアップグレードするときに Java ES インストーラによって自動的に実行されません。
- **下位互換性** : Release 4 の Web Proxy Server は Version 3.6 と下位互換性がありますが、Version 3.6 でサポートされていた NSAPI インタフェースを使用して開発したプラグインは、Release 4 でサポートされている NSAPI インタフェースを使用して再コンパイルする必要があります。
- **アップグレードのロールバック** : Release 4 への Web Proxy Server のアップグレードは、アップグレード時に未変更のまま残されていた Version 3.6 を復元することによってロールバックできます。
- **プラットフォームの問題** : Web Proxy Server をアップグレードする手順はすべての Solaris プラットフォーム上で共通ですが、Version 3.6 は Linux プラットフォームではサポートされていません。



## Web Proxy Server のアップグレード

この節では、Web Proxy Server を Java ES Release 4 にアップグレードする方法の概要を示します。Web Proxy Server はこれまで、Linux プラットフォームではサポートされていませんでした。そのため、Java ES Release 4 への Web Proxy Server のアップグレードは Solaris プラットフォームでのみ実行できます。この節で説明する内容は、次のとおりです。

- アップグレード前の作業
- Web Proxy Server のアップグレード
- アップグレードの確認
- アップグレード後の作業
- アップグレードのロールバック

### アップグレード前の作業

Web Proxy Server をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Web Proxy Server の現在のバージョンは、次のコマンドを入力することにより確認できます。

```
WebProxyServer-base/proxy-serverid/start -version
```

表 10-3 Web Proxy Server のバージョン確認の出力

Java ES のリリース	Web Proxy Server のバージョン番号
Version 3.6 (非 Java ES リリース)	3.6
Release 4	4.0.1

### Web Proxy Server が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上（および、コンピューティング環境内）のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。ただし、Java ES インストーラを使用して Web Proxy Server を Release 4 にアップグレードするとき、Web Proxy Server が依存するすべての共有コンポーネント（42 ページの表 1-6 を参照）が自動的にアップグレードされます。

## Web Proxy Server のデータのバックアップ

Release 4 への Web Proxy Server のアップグレードでは、Version 3.6 の設定データは変更されません。ただし、管理インターフェースを使用して Version 3.6 の設定データを変更し、変更をまだ保存していない場合、アップグレードを実行する前に保存する必要があります。

## 必要な設定情報とパスワードの入手

Version 3.6 からアップグレードするには、Version 3.6 がインストールされているディレクトリのパスがわかっている必要があります。

## Web Proxy Server のアップグレード

ここでは、Web Proxy Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点

同じ Web Proxy Server インストールイメージに対応するすべての Web Proxy Server インスタンスが同時にアップグレードされます。ただし、設定データの移行は各インスタンスに対して個別に行う必要があります。インストールイメージにパッチを適用している間は、このようなインスタンスをすべて停止するようにしてください。

### アップグレード手順

ここで説明する手順は、アップグレード対象のコンピュータ上の Web Proxy Server ソフトウェアに適用されます。

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. Web Proxy Server Release 4 をインストールします。
  - a. Java ES インストーラを実行します。
  - b. 選択パネルから Web Proxy Server を選択します。
  - c. 「今すぐ設定」オプションを選択します。
  - d. インストールが完了したら、Java ES インストーラを終了します。
3. 新しくインストールされたバージョンに設定情報を移行します。

この操作は、各 Web Proxy Server インスタンスに対して個別に実行する必要があります。

- a. Web Proxy Server 管理サーバーを起動します。

```
WebProxyServer-base/proxy-admserv/start
```

- b. 管理グラフィカルインターフェースにログインします。

- c. 「Server」タブをクリックし、「Migrate Server」をクリックします。
- d. Version 3.6 のインストールディレクトリパスを指定します。
- e. 移行するインスタンスを選択します。
- f. 「Migrate」ボタンをクリックします。

移行が成功したら、移行画面に、手動で実行する必要がある追加の設定が一覧表示されます。この画面には、追加する必要があるデータと、対応する設定ファイルが示されます。

設定情報の移行の詳細については、『Sun Java System Web Proxy Server 4 2005Q4 Installation and Migration Guide』(<http://docs.sun.com/doc/819-3649>)を参照してください。

- 4. 手順 f で指定された追加の設定変更を行います。

詳細については、『Sun Java System Web Proxy Server 4 2005Q4 Configuration File Reference』(<http://docs.sun.com/doc/819-3651>)を参照してください。

## アップグレードの確認

Release 4 への Web Proxy Server のアップグレードは、次のように、`-version` オプションを使用して Web Proxy Server インスタンスを起動することによって確認できます。

```
WebProxyServer-base/proxy-serverid/start -version
```

出力値については、[201 ページの表 10-3](#) を参照してください。

## アップグレード後の作業

[202 ページの「アップグレード手順」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック

[202 ページの「Web Proxy Server のアップグレード」](#) で説明しているとおり、Release 4 への Web Proxy Server のアップグレードはロールバックできません。ただし、Release 4 のアップグレード時にそのまま残されていた Version 3.6 に戻すことは可能です。



# Access Manager

この章では、Access Manager ソフトウェアを、以前の Java ES バージョンから Java ES 2005Q4 (Release 4): Sun Java System Access Manager 7 2005Q4 にアップグレードする方法について説明します。

この章では、Java ES Release 4 でサポートされている各種のアップグレードパスでの Access Manager アップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [206 ページの「Access Manager のアップグレードの概要」](#)
- [211 ページの「Java ES Release 3 からの Access Manager のアップグレード」](#)
- [227 ページの「Java ES Release 2 からの Access Manager のアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*AccessManager-base* と *AccessManagerConfig-base* という 2 つのディレクトリパスを基準に指定されています。これらのパスの少なくとも一部は、Access Manager が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*AccessManager-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- **Solaris:** /opt/SUNWam
- **Linux:** /opt/sun/identity

*AccessManagerConfig-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- **Solaris:** /etc/opt/SUNWam
  - **Linux:** /etc/opt/sun/identity
-

# Access Manager のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Access Manager の次の一般的な側面について説明します。

- [Java ES Release 4 の Access Manager について](#)
- [Access Manager のアップグレードロードマップ](#)
- [Access Manager のデータ](#)
- [互換性の問題](#)
- [Access Manager の依存関係](#)

---

**注** Access Manager は、Java ES Release 3 以前のバージョンでは Identity Server という名称でした。そのため、この章での Identity Server という表記は、Java ES Access Manager コンポーネントの以前のバージョンを指しています。

---

## Java ES Release 4 の Access Manager について

Java ES Release 4 の Access Manager は多くの点が拡張されています。内部的には、製品のアーキテクチャーが見直され、複数のアイデンティティリポジトリまたはユーザーデータストアをサポートするようになりました。そのため、Release 4 の Access Manager は、Directory Server などの LDAP ディレクトリだけでなく、その他のデータ格納プロトコルおよび格納形式もサポートします。Release 4 の Access Manager は、新しいインタフェースと、複数のアイデンティティリポジトリの統合をサポートするための新しいサービスを備えています。

フロントエンドでは、新しい Access Manager サービスおよびアイデンティティリポジトリの設定に Access Manager コンソールが使用されます。

新しい機能とインタフェースにより、Release 4 の Access Manager は新しいメジャーリリースといえるものになっています。下位互換性を確保するために、Release 4 は旧バージョンモードで実行できます。このモードは、Release 3 の Access Manager サービスに依存する Java ES コンポーネントをサポートします ( 詳細については [209 ページの「互換性の問題」](#) 参照 )。

# Access Manager のアップグレードロードマップ

表 11-1 は、Access Manager の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 11-1 Java ES Release 4: Sun Java System Access Manager 7 2005Q4 へのアップグレードパス

Java ES のリリース	Access Manager バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Access Manager 6.3 2005Q1	直接アップグレード: Release 3 バージョンを削除してから、Release 4 の完全インストールおよび再設定を行うこと によって実行します。	設定データ  Access Manager コンソールおよび認証 UI のためのカスタマイズされた JSP  ディレクトリスキーマ
Release 2	Sun Java System Identity Server 6.2 2004Q2 および 6.2 SP1	直接アップグレード: Release 2 バージョンを削除してから、Release 4 の完全インストールおよび再設定を行うこと によって実行します。	設定データ  Access Manager コンソールおよび認証 UI のためのカスタマイズされた JSP  ディレクトリスキーマ
Release 1	Sun ONE Identity Server 6.1	直接アップグレード不可: まず Release 3 にアップグレードする必要があります。手順については、『Sun Java Enterprise System 2005Q1 アップグレードと移行』( <a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a> ) を参照してください。  次に、Release 3 から Release 4 にアップグレードします。	設定データ  Access Manager コンソールおよび認証 UI のためのカスタマイズされた JSP  ディレクトリスキーマ
それ以前の Java ES リリース	Sun ONE Identity Server 6.0 または 6.0 SP 1 または  iPlanet Directory Server Access Management Edition (DSAME) 5.1	直接アップグレード不可。	

## Access Manager のデータ

Access Manager では、ほかの Java ES コンポーネントと同様に、さまざまな種類のデータを利用します。Message Queue のアップグレードでは、これらのデータをアップグレード後のバージョンに移行しなければならない場合があります。次の表は、Access Manager ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 11-2 Access Manager のデータの使用方法

データの種類	場所	使用方法
設定データ	<p><i>AccessManagerConfig-base/config/AMConfig.properties</i></p> <p><i>AccessManagerConfig-base/config/serverconfig.xml</i></p> <p>認証モジュールとカスタマイズモジュール用の JAR ファイル <i>AccessManager-base/lib</i></p>	Access Manager の設定と、バックエンドデータストアとの統合。
Web コンテナの設定	<p>Web Server: 次の場所にある <i>server.policy</i> および <i>server.xml</i> ファイル <i>WebServer-base/https-hostname/config</i></p> <p>Application Server (Java ES Release 3 および 4): 次の場所にある <i>server.policy</i> および <i>domain.xml</i> ファイル <i>AppServer8Config-base/domains/domainName/config</i></p> <p>Application Server (Java ES Release 2): 次の場所にある <i>server.policy</i> および <i>server.xml</i> ファイル <i>AppServer7Config-base/domains/domainName/config</i></p> <p>WebSphere および WebLogic: それぞれのポリシーファイルおよび設定ファイルは、これらの Web コンテナ用に Access Manager が設定されるときに変更されます。</p>	Access Manager Web コンテナインスタンスの設定
カスタマイズデータ (Web コンテナのカスタマイズされた JSP ファイル)	<p>管理コンソール: <i>AccessManager-base/web-src/applications</i></p> <p>認証 UI: <i>AccessManager-base/web-src/services</i></p>	Access Manager 管理インタフェースの設定。



表 11-2 Access Manager のデータの使用方法 ( 続き )

データの種類	場所	使用方法
ディレクトリスキーマ サービスの設定 ユーザーデータ	Directory Server	Access Manager は、ディレクトリに格納されたサービス設定データ、ユーザーデータ、およびポリシーデータに基づいて、エンドユーザー向けの認証および承認サービスを提供します。
動的なアプリケーションデータ	なし	Access Manager は、セッション状態などのアプリケーションデータを永続的に格納しません。

## 互換性の問題

Release 4 の Access Manager の新しい機能には、次の新しいインタフェースが関係します。

- 複数のバックエンドアイデンティティリポジトリ用プラグイン
- サービス設定情報を格納するための新しいディレクトリ情報ツリー構造。これにより、認証プロパティおよび承認ポリシーを、ユーザーまたはユーザーのグループと関連付けることのできるアクセス制御レلمにグループ化できます。
- Access Manager クライアント用の新しい API
- 新しい Access Manager コンソールのユーザーインタフェース

これらの新しいインタフェースに対する Access Manager のサポートは、強化モード (レلمモード) で動作するように Access Manager を設定することによって有効化されます。ただし、レلمモードは以前の Java ES Release 3 または Release 2 の Access Manager と互換性がありません。たとえば、レلمモードでの運用を可能にするために、ディレクトリデータを移行する必要があります。強化された Access Manager サービスをサポートするために、強化された Access Manager コンソールが必要です。

加えて、レルムモードは、Portal Server、Communications Express、Messaging Server などのほかの Java ES コンポーネントをサポートしません。

Release 4 の Access Manager は、旧バージョンモードで動作するように設定することで下位互換性を維持できます。旧バージョンモードは、若干の例外を除き Release 3 の Access Manager と互換性があります。例外については、『Sun Java System Access Manager 7 2005Q4 リリースノート』(<http://docs.sun.com/source/819-3474?l=ja>) を参照してください。

旧バージョンモードは、レルムモードの Access Manager と相互運用できないほかのバージョンの Access Manager ポリシーエージェントや、その他の Java ES コンポーネントをサポートするために必要です。この相互運用性はアップグレードの重要な考慮事項であり、ほとんどの Java ES 配備で、Access Manager を Release 4 旧バージョンモードにアップグレードすることが望ましいことを意味します。

ただし、旧バージョンモードで動作するように設定した場合でも、Release 4 の Access Manager は Release 3 の Delegated Administrator と互換性がありません。Access Manager を Release 4 にアップグレードする場合、Messaging Server および Calendar Server のユーザーをプロビジョニングするためには、Delegated Administrator も Release 4 にアップグレードする必要があります。ただし、Messaging Server および Calendar Server 自体は Release 4 にアップグレードする必要はありません。

## Access Manager の依存関係

ほかの Java ES コンポーネントに対する Access Manager の依存関係が、Access Manager ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Access Manager のインタフェースまたは関数に変更された場合は、Access Manager が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Access Manager は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント**：Access Manager は、特定の Java ES 共有コンポーネントに依存します(42 ページの表 1-6 を参照)。Access Manager をアップグレードするために、これらの共有コンポーネントのバージョンをアップグレードしなければならない場合があります。

- **Web コンテナ** : Access Manager は、Java ES Web Server または Java ES Application Server のどちらか (または、Weblogic や WebSphere のサードパーティー Web コンテナ) で提供される Web コンテナサービスに依存します。そのため、アップグレード後の Access Manager を Web コンテナインスタンスに合わせて再設定する必要があります。加えて、Access Manager コンソール用または認証 UI 用にカスタマイズされた JSP はすべて、アップグレードされた Access Manager 環境に移行する必要があります。
- **Directory Server** : Access Manager は、Directory Server に設定データを格納するだけでなく、そこに格納されているユーザーデータへのアクセスも行います。そのため、Access Manager のアップグレードによって、ディレクトリスキーマの拡張が必要になる場合があります。

## Java ES Release 3 からの Access Manager のアップグレード

ここでは、Access Manager の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Access Manager の完全アップグレード](#)
- [複数インスタンスのアップグレード : Release 3 と Release 4 の共存](#)
- [Release 3 の Access Manager SDK のみのアップグレード](#)

### はじめに

Java ES Release 3 Access Manager を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : アップグレードは、以前のバージョンを削除し、Release 4 を新しくインストールすることによって行います。用意されている `ampre70upgrade` スクリプトを使用して Release 3 を削除してから、Java ES インストーラを使用して Release 4 をインストールします。その後、`amconfig` スクリプトを使用して Access Manager を再設定し、`amupgrade` スクリプトを使用してディレクトリスキーマを移行します。

- **アップグレードの依存関係** : Access Manager は多くの Java ES 共有コンポーネントに対する依存関係を持ちますが (42 ページの表 1-6 を参照)、Release 4 の Access Manager はこれらすべてのコンポーネントの Release 3 バージョンと互換性があります。したがって、これらのコンポーネントのアップグレードは、Release 4 への Access Manager のアップグレードに関しては省略可能です。  
加えて、210 ページの「Access Manager の依存関係」で説明されているように、Release 4 の Access Manager は Directory Server および Web Server (あるいは、Application Server またはサードパーティー製 Web コンテナ) に依存します。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Access Manager の Release 4 へのアップグレードに関しては省略可能です。
- **下位互換性** : Release 4 の Access Manager は Release 3 と互換性はありませんが、互換性のための旧バージョンモードをサポートします (209 ページの「互換性の問題」を参照)。
- **アップグレードのロールバック** : Access Manager のアップグレードをロールバックするためのユーティリティはありません。実際、Access Manager を元の状態にロールバックするためには膨大な量の再設定を行う必要があるため、ロールバックは非効率的です。
- **プラットフォームの問題** : Access Manager をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通です。以降の手順説明では、必要に応じてプラットフォーム別のコマンドまたはファイル位置を示します。

## Release 3 の Access Manager の完全アップグレード

この節では、Java ES Release 3 から Java ES Release 4 への、Access Manager の完全アップグレードを実行する方法について説明します。

- [アップグレード前の作業](#)
- [Release 3 の Access Manager のアップグレード](#)
- [Access Manager のアップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック](#)

### アップグレード前の作業

Access Manager をアップグレードする前に、以降の節で説明する手順を実行してください。

## 現在のバージョン情報の確認

Access Manager の現在のバージョンは、次のコマンドを使用することにより確認できます。

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

表 11-3 Access Manager のバージョン確認の出力

Java ES のリリース	Access Manager のバージョン番号
Release 2	6.2
Release 3	6 2005Q1
Release 4	7 2005Q4

## Access Manager が依存するコンポーネントのアップグレード

一般的には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることが推奨されます。ただし Access Manager に関しては、Message Queue が依存する Java ES Release 3 コンポーネントのアップグレードは必須ではないため、この作業は省略可能です。

ただし、Access Manager のすべての依存関係をアップグレードすることを選択した場合は、Access Manager をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、[55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」](#)で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、[105 ページの第 4 章「Directory Server と Administration Server」](#)で説明されています。
3. **Web コンテナソフトウェア** : Web Server または Application Server をアップグレードする手順は、それぞれ [141 ページの第 6 章「Web Server」](#) および [177 ページの第 9 章「Application Server」](#)で説明されています。

Access Manager よりも先に Web コンテナソフトウェアをアップグレードしない場合、アップグレード手順は (amconfig スクリプトを使用して) Access Manager を設定し、既存の Web コンテナに配備します。

### Directory Server のデータのバックアップ

Access Manager のアップグレードプロセスでは、Directory Server のスキーマを変更するスクリプトを使用します。このため、Access Manager をアップグレードする前に、Directory Server コンソールや db2bak などのコマンド行ユーティリティを使用して Directory Server データをバックアップしてください。

Directory Server のバックアップの詳細については、『Sun Java System Directory Server 5 2005Q1 管理ガイド』(<http://docs.sun.com/source/819-2011?l=ja>) を参照してください。

### Release 3 Access Manager の設定情報のバックアップ

Release 4 の Access Manager ソフトウェアの再設定には Release 3 バージョンの再設定が必要なため、設定ファイルをわかりやすい場所にバックアップすることは重要です。バックアップが推奨されるファイルは次のとおりです。

- AMConfig.properties ファイル  
*AccessManagerConfig-base/config/AMConfig.properties*
- serverconfig.xml ファイル  
*AccessManagerConfig-base/config/serverconfig.xml*
- Web コンテナ設定ファイル:
  - Web Server: 次の場所にある server.policy および server.xml ファイル  
*WebServer-base/https-hostname/config*
  - Application Server: 次の場所にある server.policy および domain.xml ファイル  
*AppServer7Config-base/domain/domain1/config*
  - サードパーティー製 Web コンテナ: 所定の設定ファイル
- 認証モジュールとカスタマイズモジュール用の JAR ファイル  
*AccessManager-base/lib*

### Web コンテナのカスタマイズファイルのバックアップ

Access Manager によって参照される Web コンテナカスタマイズファイルがある場合、それらのファイルをバックアップすることが推奨されます。このようなカスタマイズの例には次のものがあります、

- カスタマイズした Access Manager コンソール JSP ページ。  
*AccessManager-base/web-src/applications*
- カスタマイズした認証 UI JSP ページ。  
*AccessManager-base/web-src/services*
- カスタマイズした XML ファイル。  
*AccessManagerConfig-base/config/xml*

---

**ヒント** Access Manager をアップグレードしたあとで、バックアップしたコードを使用してカスタマイズを再適用できるように、カスタマイズ内容を記録しておきます。

---

### Release 3 の Access Manager のログファイルとデバッグファイルのバックアップ

システム状態情報の分析に使用できるように、ログファイルとデバッグファイルをバックアップして消失を防ぐことをお勧めします。これらのファイルは次の場所にあります。

- デバッグファイル  
/var/AccessManager-base/debug
- ログファイル  
/var/AccessManager-base/logs

### 必要な設定情報とパスワードの入手

Access Manager をアップグレードするには、次に示すような、特定の設定情報を提供する必要があります。

- Access Manager 管理者のユーザー ID とパスワード
- LDAP ユーザー ID とパスワード
- Access Manager が使用している Directory Server インスタンスのディレクトリマネージャー名およびパスワード

## Release 3 の Access Manager のアップグレード

Java ES Release 4 への Access Manager ソフトウェアのアップグレードには、Access Manager を再設定する手順と、Access Manager データを移行する手順が含まれます。

### アップグレードの概要

Access Manager をアップグレードするための手順は、次の手順で構成されます。

1. [Java ES Release 3 バージョンの Access Manager](#) を削除します。ampre70upgrade スクリプトを使用します。
2. [Java ES Release 4 バージョンの Access Manager](#) をインストールします。Java ES Release 4 インストーラを使用し、「あとで設定」オプションを指定します。
3. [モバイルアクセスソフトウェア](#) をアップグレードします。
4. [Access Manager 用 JSP](#) を再カスタマイズします。
5. [Access Manager](#) の配備を取り消し、再設定し、[Web コンテナ](#) に再配備します。amconfig スクリプトを使用します。

6. ディレクトリ構造とスキーマを更新します。amupgrade スクリプトを使用します。以降の手順中で、これらの各手順について説明します。

### アップグレード手順

1. Java ES Release 3 バージョンの Access Manager を削除します。
  - a. Release 3 の Access Manager がインストールされたコンピュータに root としてログインするか、スーパーユーザーになります。  
su -
  - b. Java ES Release 4 配布パッケージ内の *platform/Product/identity\_svr/Tools* ディレクトリに移動します。
  - c. ampre70upgrade スクリプトによって要求される、次のパラメータの値を入力します。

表 11-4 Access Manager 設定パラメータ : ampre70upgrade

パラメータ	値
Directory Server ホスト	完全修飾名を設定する : <i>hostname.domian</i>
Directory Server ポート	非 SSL ポート番号を指定する <sup>1</sup> デフォルト : 389
最上位管理者の DN	デフォルト : <b>uid=amadmin,ou=People,dc=iplanet,dc=com</b>
最上位管理者のパスワード	

1. Directory Server の SSL ポートを SSL のデフォルト値である 636 などに指定すると、アップグレード前プロセスは正しく完了しません。

- d. Directory Server が稼働していることを確認し、稼働していない場合は起動します。
- e. ampre70upgrade スクリプトを実行します。

```
./ampire70upgrade
```

このスクリプトは Access Manager の設定ファイルをバックアップし、Release 3 のベースパッケージを削除します。ローカライズ版のパッケージは **手順 f** に従って、手動で削除する必要があります。



- f. コンピュータにインストールされているローカライズ版の Access Manager パッケージを手動で削除します。

ampre70upgrade スクリプトではローカライズ版の Access Manager パッケージは削除されません。ローカライズ版のアップグレードを正しく実行するには、これらを手動で削除する必要があります。

- o Solaris プラットフォームでは `pkgrm` を使用して `SUNWam1Locale`、`SUNWLocaleamrmap` を削除
- o Linux では `rpm -e` を使用して `sun-identity-sdk-Locale` を削除

2. Java ES Release 4 バージョンの Access Manager をインストールします。

- a. Release 3 の Access Manager がホストされたコンピュータ上で Java ES インストーラを実行します。
- b. 選択パネルから Access Manager を選択します。
- 画面上に「衝突」メッセージが表示された場合、インストーラが以前のバージョンの Access Manager 設定情報を検出したことを意味します。これは予期される動作です。後続の手順で再設定が実行されます。この「衝突」メッセージは無視して先に進むことができます。
- c. Release 3 がインストールされていたのと同じインストールディレクトリを指定します。
- d. 「あとで設定」オプションを選択します。
- e. インストールが完了したら、Java ES インストーラを終了します。

---

**注** Java ES インストーラのコマンド行インタフェースを使用して Access Manager をインストールする場合、Directory Server ソフトウェアも自動的にインストールされます。リモート Directory Server を使用している場合、『Sun Java Enterprise System インストールガイド (UNIX 版)』の手順を使用してローカル Directory Server ソフトウェアをアンインストールできます。

---

3. モバイルアクセスソフトウェアをアップグレードします。

Release 3 バージョンにパッチを適用することによって、Access Manager モバイルアクセスソフトウェアをアップグレードする必要があります。必要なパッチを次の表に示します。

表 11-5 Access Manager Mobile Access ソフトウェアをアップグレードするためのパッチ<sup>1</sup>

説明	Solaris パッチ ID	Linux パッチ ID
Mobile Access ソフトウェア	119530-01 (SPARC) 119531-01 (x86)	119532-01 • sun-identity-mobileaccess-6.2-25.i386.rpm • sun-identity-mobileaccess-config-6.2-25.i386.rpm

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

- a. 表 11-5 のパッチ番号を参照して、必要なパッチを入手します。

パッチは

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

- b. root としてログインするか、スーパーユーザーになります。

```
su -
```

- c. 表 11-5 のパッチを適用します。

Solaris の場合：

```
patchadd patch_ID
```

Linux の場合：

```
rpm -Fvh sun-identity-mobileaccess-6.2-25.i386.rpm
rpm -Fvh sun-identity-mobileaccess-config-6.2-25.i386.rpm
```

4. Access Manager 用 JSP を再カスタマイズします。

214 ページの「Web コンテナのカスタマイズファイルのバックアップ」で保存した、Access Manager コンソールおよび認証ユーザーインタフェース (UI) 用 JSP に、Release 3 のカスタマイズを再適用します。

続いて、カスタマイズ JSP ファイルを適切なディレクトリにコピーします。

Solaris システムの例を次に示します。

- コンソール: *AccessManager-base*/web-src/applications/console
- 認証 UI: *AccessManager-base*/web-src/services/config/auth/default または *AccessManager-base*/web-src/services/config/auth/default\_*Locale* (*Locale* は ja などのロケールインジケータ)

詳細については、『Sun Java System Access Manager Developer's Guide』(<http://docs.sun.com/doc/819-2139>) を参照してください。

5. Access Manager の配備を取り消し、再設定し、Web コンテナに再配備します。

amconfig スクリプトを実行して、使用している Web コンテナ用に Access Manager を設定します。amconfig スクリプト ( および、関連付けられた amsamplesilent テンプレート入力ファイル ) は次のディレクトリにあります。

```
AccessManager-base/bin
```

amconfig スクリプトおよび amsamplesilent テンプレートファイルの詳細については、『Sun Java System Access Manager 6 2005Q1 管理ガイド』(<http://docs.sun.com/source/819-1938?l=ja>) を参照してください。

次の手順を実行して、Access Manager を再設定し、Web コンテナに再配備します。

- a. 213 ページの「Access Manager が依存するコンポーネントのアップグレード」で説明しているように Web コンテナソフトウェアをアップグレードする場合、アップグレードが完了していることを確認します。
- b. Directory Server と適切な Web コンテナが稼働していることを確認します。
- c. amsamplesilent テンプレート入力ファイルを基に、amconfig 入力ファイルを作成します。

```
cp amsamplesilent config-file
```

- d. config-file 内の設定パラメータを設定します。

すべてのパラメータを正確に設定する必要があります。次の表に示すように、設定値には、AMConfig.properties ファイルから移行できるものと、アップグレード手順の中で新しく設定するものがあります。

表 11-6 Access Manager 設定パラメータ

パラメータ	値
アップグレードパラメータ	
DEPLOY_LEVEL	26 ( 配備取消し ) または 1 ( 再設定と配備 )
DIRECTORY_MODE	5 ( 既存のアップグレード )
AM_REALM	disabled に設定 ( レルムモード無効、旧バージョンモード有効 ) ( デフォルト = enabled )
JAVA_HOME	JDK Release 4 ディレクトリに設定: /usr/java/jdk1.5.0_04/
WEB_CONTAINER	使用している Web コンテナの種類に適した値を設定し、 <i>config-file</i> の対応するセクションのみを設定する
WS61_INSTANCE (Web Server を Web コンテナとして使用している場合)	=https-<hostname>.<domain> 上の値は /WebServer-base/SUNwsbsvr/ 内のインスタンス名と一致する。 値は大文字と小文字が区別される
AMConfig.properties から移行される値	
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.encryption.pwd

その他のパラメータについては、Web コンテナまたはパスワードを変更する場合を除いて、アップグレードする Release 3 の設定で使われていたのと同じ値を指定します。

- e. `amconfig` を実行して、Access Manager の配備を取り消します。

`config-file` 内の `DEPLOY_LEVEL` の値を 26 に設定します。

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

- f. `amconfig` を実行して Access Manager を再設定し、Web コンテナに配備します。

`config-file` 内の `DEPLOY_LEVEL` の値を 1 に設定します。

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

6. ディレクトリ構造とスキーマを更新します。

Release 4 の Access Manager は Release 3 のディレクトリ構造と共存しますが、Release 4 の機能をサポートするためには構造を変更する必要があります。`amupgrade` スクリプトを実行して、Access Manager のディレクトリ構造とスキーマを Release 4 に更新します。このスクリプトは次のディレクトリにインストールされています。

- o Solaris:  
`AccessManager-base/upgrade/scripts`
  - o Linux:  
`AccessManager_base/identity/upgrade/scripts`
- a. `amupgrade` スクリプトによって要求される、次のパラメータの値を入手します。

表 11-7 Access Manager 設定パラメータ : `amupgrade`

パラメータ	値
Directory Server ホスト	完全修飾名を設定する : <code>hostname.domain</code>
Directory Server ポート	非 SSL ポート番号を指定する <sup>1</sup> デフォルト : 389
ディレクトリマネージャー DN	デフォルト : <code>cn=Directory Manager</code>
ディレクトリマネージャーパスワード	
最上位管理者の DN	デフォルト : <code>uid=amadmin,ou=People,dc=iplanet,dc=com</code>
最上位管理者のパスワード	

表 11-7 Access Manager 設定パラメータ : amupgrade ( 続き )

パラメータ	値
Enable Realm Mode	Y/N: Yes は、レルムモードが有効であり、サービスデータが新しいレルムツリーに移行されたことを意味します。No ( デフォルト ) はサービスデータが旧バージョンモードのままであることを意味します。

1. Directory Server の SSL ポートを SSL のデフォルト値である 636 などに指定すると、アップグレードプロセスは正しく完了しません。

- b. amupgrade スクリプトを実行します。

```
cd AccessManager-base/upgrade/scripts
./amupgrade
```

スクリプトによるアップグレードが成功すると、「Upgrade completed」というメッセージが表示されます。

- c. 次のアップグレードログファイルをチェックし、ディレクトリスキーマ拡張についての情報を確認します。

**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. Access Manager を起動します。

Access Manager が配備される Web コンテナを再起動します。

## Access Manager のアップグレードの確認

アップグレード手順が完了したら、次のようにして、アップグレードが成功したかどうかを確認します。

1. 次の URL を使って、amadmin として Access Manager コンソールにログインします。

```
http://hostname.domain:port/amconsole
```

ここで、hostname.domain:port は、Access Manager をホストする Web コンテナの完全修飾ホスト名とポート番号です。

206 ページの「[Java ES Release 4 の Access Manager について](#)」で言及されている新しい Release 4 のサービスが、「サービス設定」タブで利用可能であることを確認します。

2. `/var/sadm/install/logs` ディレクトリ内の次のアップグレードログファイルをチェックして、アップグレードの状態を確認します。

Sun Java Enterprise System インストーラ :

- 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 スクリプト :

- o `Sun_Java_System_Identity_Server_upgrade_dit_log.timestamp`

3. Access Manager トラブルシューティングファイルで、エラーの有無を確認します。

ファイルは `/var/opt/SUNWam/debug` にあります。

## アップグレード後の作業

SAML (Security Assertion Markup Language) サービスを使用する場合、Access Manager コンソールを使って SAML 認証モジュールを追加および有効化する必要があります。SAML 認証モジュールインスタンスの作成の詳細については、『Sun Java System Access Manager 6 2005Q1 管理ガイド』

(<http://docs.sun.com/source/819-1938?l=ja>) を参照してください。

## アップグレードのロールバック

Access Manager をアップグレード前の状態にロールバックするためのスクリプトは用意されていません。ロールバックプロセスは、アップグレード前の作業でバックアップした Access Manager データ (215 ページの「[Release 3 の Access Manager のログファイルとデバッグファイルのバックアップ](#)」を参照) を使用して手動で行う必要があります。ロールバックは非常に難しく、実際的ではありません。

## 複数インスタンスのアップグレード : Release 3 と Release 4 の共存

配備のアーキテクチャーによっては、高可用性とスケーラビリティを実現するために、Access Manager が複数のコンピュータシステムに配備される場合があります。複数の Access Manager インスタンスが同じ Directory Server にアクセスします。多くの場合、サービスを中断させずに連続的に Access Manager インスタンスをアップグレードするのが理想的です。この節では、そのようなローリングアップグレードを実行するための手順について説明します。

---

**注**           現在のリリースでは、同一ホストシステムにインストールされている Access Manager の複数インスタンスのアップグレードはサポートされていません。同一ホストに複数のインスタンスが存在する場合は、メインインスタンスをアップグレードしてから追加インスタンスを再作成する必要があります。

---

Access Manager を Release 3 からアップグレードする手順には、Release 4 をサポートするようにディレクトリスキーマを移行する手順が含まれます。Release 3 の Access Manager は Release 4 のディレクトリスキーマをサポートしませんが、Release 4 の Access Manager は Release 3 のディレクトリスキーマをサポートします。

Java ES Release 4 の Access Manager インスタンスと Release 3 の Access Manager インスタンスは、ディレクトリスキーマがまだ Release 4 に移行されていない場合に限り、共存して同じ Directory Server に対して同時に動作できます。したがって、ローリングアップグレードでは、すべての Access Manager インスタンスがまず Release 4 にアップグレードされるまで、ディレクトリスキーマを Release 4 に移行しないことが推奨されます。

順次アップグレードの実行では、[221 ページの「ディレクトリ構造とスキーマを更新します。」](#)の手順を除き、[215 ページの「Release 3 の Access Manager のアップグレード」](#)の説明に従って Access Manager の各インスタンスをアップグレードします。すべてのインスタンスがアップグレードされたあとで、省略したその手順を実行できます。



## Release 3 の Access Manager SDK のみのアップグレード

配備構成によっては、1 台以上のコンピュータシステムに、他の Access Manager コンポーネントをインストールせずに、Access Manager SDK コンポーネントがインストールされることがあります。Access Manager SDK は Access Manager に対するリモートインタフェースとして機能し、Access Manager と同じ運用モード (旧バージョンまたはレルム) に合わせて再設定する必要があります。Access Manager へのリモートインタフェースであるこの SDK は、Directory Server にアクセスするように設定する必要はありません。

Portal Server や Communications Express など、Web コンテナサービスに依存する Web コンポーネントをサポートするために Access Manager SDK を使用する場合、対応する Web コンテナに合わせて Access Manager SDK を設定する必要があります。ただし、Access Manager SDK は非 Web コンポーネントをサポートすることもでき、その場合 Web コンテナは必要ありません。

これまでに述べたような特性から、Access Manager SDK をアップグレードする手順は、Access Manager の完全アップグレードの手順の一部です。

この節では、Access Manager SDK のみを Java ES Release 3 から Java ES Release 4 にアップグレードする方法について説明します。

- [アップグレード前の作業](#)
- [Release 3 の Access Manager SDK のアップグレード](#)
- [Access Manager SDK のアップグレードの確認](#)
- [アップグレードのロールバック](#)

### アップグレード前の作業

Access Manager SDK のアップグレード前作業は Access Manager の完全アップグレードの場合と同じですが、Directory Server と管理ツールのカスタマイズに関連する作業は除外します。Access Manager SDK に対して必要なアップグレード前作業は次のとおりです。

- [213 ページの「Access Manager が依存するコンポーネントのアップグレード」](#)  
ただし、Access Manager SDK については、Directory Server への依存関係はなく、Access Manager SDK が Web コンテナ内で動作する場合にのみ Web コンテナソフトウェアへの依存関係があります。
- [215 ページの「Release 3 の Access Manager のログファイルとデバッグファイルのバックアップ」](#)

ただし、Access Manager SDK については、Access Manager SDK が Web コンテナ内で動作する場合にのみ、Web コンテナ設定ファイルをバックアップする必要があります。

- [215 ページの「Release 3 の Access Manager のログファイルとデバッグファイルのバックアップ」](#)

また、これらのファイルにアクセスするための管理ユーザー名とパスワードを入手する必要があります。

## Release 3 の Access Manager SDK のアップグレード

Access Manager SDK のアップグレード手順は Access Manager の完全アップグレードの場合と同じですが、管理ツールのカスタマイズに関連する作業と、ディレクトリスキーマの移行は除外します。

1. Access Manager SDK の Java ES Release 3 バージョンを削除します。

[216 ページの「Java ES Release 3 バージョンの Access Manager を削除します。」](#)の指示に従いますが、削除するのは Access Manager SDK だけです。

2. Java ES Release 4 バージョンの Access Manager SDK をインストールします。

[217 ページの「Java ES Release 4 バージョンの Access Manager をインストールします。」](#)の指示に従いますが、インストールするのは Access Manager SDK だけです。

3. Access Manager SDK を再設定します。

[219 ページの「Access Manager の配備を取り消し、再設定し、Web コンテナに再配備します。」](#)の指示に従いますが、異なる点として、`DIRECTORY_MODE=5` に設定し、`DEPLOY_LEVEL` パラメータを次のように設定します。

- Access Manager SDK を Web コンテナ用に設定する場合：  
`DEPLOY_LEVEL=4` (SDK をアップグレードし Web コンテナを設定する)
- Access Manager SDK を Web コンテナ用に設定しない場合：  
`DEPLOY_LEVEL=3` (SDK のみをアップグレードする)

## Access Manager SDK のアップグレードの確認

Access Manager SDK のアップグレードが成功したかどうかは、3 とおりの方法で確認できます。

- Portal Server、Communications Express、または Access Manager とのインタフェースに Access Manager SDK を使用するその他のコンポーネントを実行し、認証が機能することを確認します。
- 次の場所に用意されている、Access Manager SDK サンプルを実行します。

`/AccessManager-base/samples/sdk`

- AMConfig.properties ファイル内の com.iplanet.am.version プロパティの値を確認します。

`AccessManagerConfig-base/config/AMConfig.properties`

## アップグレードのロールバック

Access Manager をアップグレード前の状態にロールバックするためのスクリプトは用意されていません。ロールバックプロセスは、アップグレード前の作業でバックアップした Access Manager データ (215 ページの「[Release 3 の Access Manager のログファイルとデバッグファイルのバックアップ](#)」を参照) を使用して手動で行う必要があります。ロールバックは非常に難しく、実際的ではありません。

# Java ES Release 2 からの Access Manager のアップグレード

Java ES 2004Q2 (Release 2) Access Manager を Release 4 にアップグレードする手順は、次に説明するいくつかの例外を除き、Release 3 Access Manager を Release 4 にアップグレードする手順と同じです。

## アップグレード前の作業

Access Manager をアップグレードする前に、212 ページの「[アップグレード前の作業](#)」で説明されている手順を実行します。ただし、213 ページの「[Access Manager が依存するコンポーネントのアップグレード](#)」を次の節と置き換え、「[ディレクトリスキーマをアップグレードする](#)」の節を追加します。

## Access Manager が依存するコンポーネントをアップグレードする

Release 3 からのアップグレードと異なり、Release 2 から Release 4 へのアップグレード前の作業には、すべての共有コンポーネント (42 ページの表 1-6 を参照) と、Access Manager が依存するローカルにあるすべての製品コンポーネントの Release 4 へのアップグレードを含めるようにしてください。

Access Manager の依存関係をアップグレードする場合は、Access Manager をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「[Java ES 共有コンポーネントのアップグレード](#)」で説明されています。

2. **Directory Server:** Directory Server が Access Manager と同じコンピュータ上に存在することはほとんどありませんが、Directory Server を Release 4 にアップグレードする手順は、[125 ページの「Java ES Release 2 からの Directory Server と Administration Server のアップグレード」](#)で説明されています。
3. **Web コンテナソフトウェア:** Web Server または Application Server をアップグレードする手順は、それぞれ [150 ページの「Java ES Release 2 からの Web Server のアップグレード」](#) および [189 ページの「Java ES Release 2 からの Application Server のアップグレード」](#)で説明されています。

## ディレクトリスキーマをアップグレードする

Messaging Server、Calendar Server、またはその他の通信コンポーネントをサポートするために、Directory Preparation Tool (comm\_dssetup.pl) を使用して Directory Server を設定した場合、Access Manager をアップグレードする前に、Directory Preparation Tool の Release 4 バージョンを使用してディレクトリスキーマをアップグレードする必要があります。このアップグレード前作業は、Access Manager が依存するコンポーネントをアップグレードしたあとに実行します。Directory Preparation Tool をアップグレードする手順は、[244 ページの「Directory Preparation Tool の Java ES Release 2 からのアップグレード」](#)で説明しています。

## Release 2 Access Manager のアップグレード

Access Manager を Release 2 から Release 4 にアップグレードする手順は、Access Manager ソフトウェアを配備している Web コンテナによって異なります。

### Release 2 Access Manager のアップグレード : Web Server Web コンテナ

Release 2 の Access Manager を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、Web Server Web コンテナへの配備時に、[215 ページの「Release 3 の Access Manager のアップグレード」](#)の手順をそのまま使用します。

### Release 2 Access Manager のアップグレード : Application Server Web コンテナ

配備先 Web コンテナが Application Server のとき、Release 4 への Access Manager のアップグレードには2つのケースがあります。

- Release 4 の Application Server が新規インストールされた場合。この場合、Release 2 の Access Manager を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き [215 ページの「Release 3 の Access Manager のアップグレード」](#)の手順にそのまま従います。

- Release 2 の Application Server が Release 4 にアップグレードされた場合。この場合、Release 2 の Access Manager を Release 4 にアップグレードするには、次の指示に従います。

アップグレードされた Application Server Web コンテナに配備する際に Access Manager をアップグレードするには、[216 ページの手順 1](#) から [手順 4](#) までの指示に従います。その際、「Release 3」の記述をすべての「Release 2」に読み替えます。

[手順 1](#) から [手順 4](#) を要約すると、次のようになります。

1. Access Manager の Release 2 バージョンを削除します。

ampre70upgrade スクリプトを使用します。[216 ページの「Java ES Release 3 バージョンの Access Manager を削除します。」](#)の手順に従ってください。

2. [Java ES Release 4 バージョンの Access Manager](#) をインストールします。Java ES Release 4 インストーラを使用し、「あとで設定」オプションを指定します。
3. [モバイルアクセスソフトウェアをアップグレード](#)します。
4. [Access Manager 用 JSP](#) を再カスタマイズします。

Access Manager が最初に配備されたときの Release 2 Application Server インスタンス (*instanceName*) は、Release 4 にアップグレードされると、アップグレード処理によって作成されたノードエージェントの下で移行されています。このアップグレードされた Application Server インスタンス内の Access Manager のアップグレードには、次の追加の手順が必要です。

5. Access Manager をサポートする次のコンポーネントが稼働していることを確認します。
  - a. Directory Server が稼働していることを確認します。
  - b. まだドメイン管理サーバー (DAS) が稼働していない場合は起動します。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

- c. Access Manager が配備されている、アップグレードされた Application Server インスタンス (*instanceName*) がまだ稼働していない場合、起動します。

これは、アップグレードされた Application Server の移行先ノードエージェントを起動することによって行います。

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

上のコマンドおよび以降の手順では、次の規則が使用されています。

- *nodeAgentName* は、*hostName\_domainName* の形式をしています。
- デフォルトの *domainName* は domain1 です。
- デフォルトの *instanceName* は server1 です。

6. Access Manager の配備を取り消し、再設定し、Application Server インスタンスに再配備します。amconfig スクリプトを使用します。
- a. amsamplesilent テンプレート入力ファイルを基に、amconfig 入力ファイルを作成します。
 

```
cp amsamplesilent config-file
```
  - b. config-file 内の設定パラメータを設定します。
 

すべてのパラメータを正確に設定する必要があります。次の表に示すように、設定値には、AMConfig.properties ファイルから移行できるものと、アップグレード手順の中で新しく設定するものがあります。

表 11-8 Access Manager 設定パラメータ

パラメータ	値
アップグレードパラメータ	
DEPLOY_LEVEL	26 ( 配備取消し ) または 1 ( 再設定と配備 )
DIRECTORY_MODE	5 ( 既存のアップグレード )
AM_REALM	disabled に設定 ( レルムモード無効、旧バージョンモード有効 )。デフォルト = enabled
JAVA_HOME	JDK Release 4 ディレクトリに設定 : /usr/java/jdk1.5.0_04/
WEB_CONTAINER	Application Server Web コンテナ用の値に設定し、config-file の対応するセクションのみを設定する
AS81_INSTANCE	=instanceName
AS81_ADMIN_IS_SECURE	=false
AMConfig.properties から移行される値	
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

表 11-8 Access Manager 設定パラメータ ( 続き )

パラメータ	値
AM_ENC_PWD	am.encryption.pwd

その他のパラメータについては、Web コンテナまたはパスワードを変更する場合を除いて、アップグレードする Release 2 の設定で使われていたのと同じ値を指定します。

- c. `amconfig` を実行して、Access Manager の配備を取り消します。

`config-file` 内の `DEPLOY_LEVEL` の値を 26 に設定します。

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

- d. `amconfig` を実行して Access Manager を再設定し、Web コンテナに配備します。

`config-file` 内の `DEPLOY_LEVEL` の値を 1 に設定します。

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

7. `server.policy` ファイルを次のディレクトリからコピーします。

```
AppServer8Config-base/domains/domainName/config
```

コピー先ディレクトリは次のとおりです。

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/config
```

8. Release 4 の Application Server `domain.xml` ファイルを変更します。

- a. Access Manager が当初配備されていた Release 2 の Application Server インスタンスの `server.xml` ファイルから、Access Manager の `classpath-suffix` および `server-classpath` の情報をコピーします。

```
AppServer7Config-base/domains/domainName/instanceName/config/server.xml
```

- b. コピーしたクラスパス情報を、Access Manager が配備される、アップグレード後の Application Server インスタンスの domain.xml ファイルの classpath-suffix エントリおよび server-classpath エントリにそれぞれ付加します。

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/  
config/domain.xml
```

クラスパス情報は、Release 4 の Application Server の domain.xml ファイルの、instanceName-config ブロックに追加することが推奨されます。このブロックは次の行で始まります。

```
<config dynamic-reconfiguration-enabled="true"  
name="instanceName-config">
```

クラスパスエントリを追加するときは必ず、古い情報と新しい情報の間に、コロン(:) またはクラスパスエントリで使用されているその他のパス区切り文字を入れるようにしてください。AppServer7-base パスを持つすべてのエントリを削除することもできます(エラーを発生させないように注意する)。

9. DAS を再起動します。

```
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. Access Manager が配備されるサーバーインスタンスを再起動します。

```
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. 221 ページの手順 6 の説明に従って、ディレクトリ構造とスキーマを更新します。

## Access Manager のアップグレードの確認

アップグレード手順が完了したら、222 ページの「Access Manager のアップグレードの確認」の説明に従って、アップグレードが成功したかどうかを確認します。

## アップグレード後の作業

SAML (Security Assertion Markup Language) サービスを使用する場合、Access Manager コンソールを使って SAML 認証モジュールを追加および有効化する必要があります。SAML 認証モジュールインスタンスの作成の詳細については、『Sun Java System Access Manager 6 2005Q1 管理ガイド』

(<http://docs.sun.com/source/819-1938?l=ja>) を参照してください。



## アップグレードのロールバック

Access Manager をアップグレード前の状態にロールバックするためのスクリプトは用意されていません。ロールバックプロセスは、アップグレード前の作業でバックアップした Access Manager データ (215 ページの「[Release 3 の Access Manager のログファイルとデバッグファイルのバックアップ](#)」を参照) を使用して手動で行う必要があります。ロールバックは非常に難しく、実際的ではありません。



# Directory Preparation Tool

この章では、Directory Preparation Tool を Java ES 2005Q4 (Release 4): Sun Java System Directory Preparation Tool 6.3 2005Q4 にアップグレードする方法について説明します。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [236 ページの「Directory Preparation Tool のアップグレードの概要」](#)
- [238 ページの「Directory Preparation Tool の Java ES Release 3 からのアップグレード」](#)
- [244 ページの「Directory Preparation Tool の Java ES Release 2 からのアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*DirPrepTool-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Directory Preparation Tool が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*DirPrepTool-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- **Solaris:** /opt/SUNWcomds
  - **Linux:** /opt/sun/comms/dssetup
-

# Directory Preparation Tool のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Directory Preparation Tool の次の一般的な側面について説明します。

- [Java ES Release 4 Directory Preparation Tool について](#)
- [Directory Preparation Tool のアップグレードロードマップ](#)
- [Directory Preparation Tool のデータ](#)
- [互換性の問題](#)
- [Directory Preparation Tool の依存関係](#)

## Java ES Release 4 Directory Preparation Tool について

Directory Preparation Tool の Java ES Release 4 バージョンには、Release 4 の通信コンポーネント (Messaging Server、Calendar Server、Communications Express、および Delegated Administrator) が Directory Server を使用できるように準備するために必要ないくつかのマイナーな修正が含まれています。

詳細については、適切なリリースノートを参照してください。

## Directory Preparation Tool のアップグレードロードマップ

表 12-1 は、サポートされている Directory Preparation Tool の Java ES Release 4 へのアップグレードパスを示しています。この表は、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方に適用されます。

表 12-1 Java ES Release 4: Sun Java System Directory Preparation Tool 6.3 2005Q4 へのアップグレードパス

Java ES のリリース	Directory Preparation Tool のバージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Directory Preparation Tool 6.2 2005Q1	直接アップグレード：パッチを適用することにより実行します。	Directory Server を Release 4 の通信コンポーネント用に準備します。

表 12-1 Java ES Release 4: Sun Java System Directory Preparation Tool 6.3 2005Q4 へのアップグレードパス (続き)

Java ES のリリース	Directory Preparation Tool のバージョン	一般的な手順	必要な再設定
Release 2	comm_dssetup.pl script Version 6.1 Revision 0.2 (Messaging Server および Calendar Server に含まれる)	直接アップグレード: 開始パッチを適用し、次にアップグレードパッチを適用することにより実行します。	Directory Server を Release 4 の通信コンポーネント用に準備します。
Release 1	comm_dssetup.pl script (Messaging Server および Calendar Server に含まれる)	直接アップグレードは動作保証なし: ただし、開始パッチパッチを適用し、次にアップグレードパッチを適用することにより実行できます。	Directory Server を Release 4 の通信コンポーネント用に準備します。
それ以前の Java ES リリース	ims_dssetup.pl script (Messaging Server に含まれる)	直接アップグレード不可:	

## Directory Preparation Tool のデータ

次の表は、Directory Preparation Tool ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類の種類を示しています。

表 12-2 Directory Preparation Tool のデータの使用方法

データの種類	場所	使用法
Directory Server スキーマ	Directory Server	Directory Server を Release 4 の通信コンポーネント用に準備: スキーマの変更、新しいエントリの作成、およびインデックスの作成

## 互換性の問題

Release 4 Directory Preparation Tool ではインタフェースの変更は導入されておらず、以前のバージョンと下位互換性があります。

## Directory Preparation Tool の依存関係

Directory Preparation Tool は、Directory Server を除き、ほかの Java ES コンポーネントには依存しません。Directory Preparation Tool は、Java ES の通信コンポーネントで使用できるように Directory Server を設定するために使用されます。

# Directory Preparation Tool の Java ES Release 3 からのアップグレード

ここでは、Directory Preparation Tool の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 Directory Preparation Tool のアップグレード](#)

## はじめに

Java ES Release 3 Directory Preparation Tool を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順**：このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。次に、Directory Preparation Tool を使用して、Release 4 Messaging Server、Calendar Server、Communications Express、および Delegated Administrator コンポーネントに対するサポートの必要に応じて Directory Server を変更します。
- **アップグレードの依存関係**：Directory Preparation Tool は Java ES 共有コンポーネントには依存せず、Release 3 Directory Server と互換性があります。したがって、Directory Server のアップグレードは、Directory Preparation Tool の Release 4 へのアップグレードに関しては省略可能です。
- **下位互換性**：Release 4 Directory Preparation Tool は、Release 3 バージョンと下位互換性があります。

- **アップグレードのロールバック** : Release 4 アップグレードの Release 3 へのロールバックは、アップグレード中に適用されたパッチを削除することにより行われます。次に、Directory Server に対して Release 3 Directory Preparation Tool を実行することにより、Release 4 バージョンによって加えられた変更をバックアウトできます。
- **プラットフォームの問題** : Directory Preparation Tool をアップグレードするための一般的な手順は、Solaris オペレーティングシステムと Linux オペレーティングシステムのどちらでも同じですが、パッチの適用方法が異なります。そのため、アップグレードプロセスにはプラットフォーム固有の手順が含まれており、通常、Linux プラットフォームではパッチをロールバックできません。

## Release 3 Directory Preparation Tool のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Directory Preparation Tool の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 Directory Preparation Tool のアップグレード \(Solaris\)](#)
- [Release 3 Directory Preparation Tool のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Directory Preparation Tool をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Directory Server に対して最後に実行された Directory Preparation Tool のバージョンは、このツールにより書き込まれた `cn=CommServers,o=comms-config` エントリの属性値を調べることにより確認できます。

```
./ldapsearch -D "cn=Directory Manager" -w password
             -b cn=CommServers,o=comms-config cn="CommServers"
             sunkeyvalue
```

このエントリには、現在のバージョンを指定する次の 2 つの属性が含まれています。

- `dssetup_ver=version` (例: 6.3)
- `dssetup_rev=revision` (例: 2.01)

このツールは、実行している Directory Preparation Tool のバージョンが、以前に実行されたバージョンと同じかそれ以前の場合にのみ、コンソールにメッセージを書き込みます。ツールの実行方法については、[241 ページの手順 5 \(Solaris\)](#) または [243 ページの手順 5 \(Linux\)](#) のアップグレード手順を参照してください。

### **Directory Preparation Tool が依存するコンポーネントのアップグレード**

一般には、コンピュータシステム上 (および、コンピューティング環境内) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Directory Preparation Tool の Release 4 へのアップグレードは、ほかの Java ES コンポーネントには依存しません。

### **ディレクトリデータのバックアップ**

Directory Preparation Tool を Release 3 から Release 4 にアップグレードしただけでは、Directory Server データは変更されません。ただし、安全策として、Directory Preparation Tool をアップグレードし、Directory Server に対して実行する前に Directory Server をバックアップすることをお勧めします。

### **必要な設定情報とパスワードの入手**

Directory Preparation Tool をアップグレードするには、スーパーユーザーのパスワードが必要です。このツールは、以前の実行で使用されたパラメータ値を記憶しており、次の実行時にはその値をデフォルトとして表示します。

## **Release 3 Directory Preparation Tool のアップグレード (Solaris)**

ここでは、Directory Preparation Tool のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### **アップグレードの注意点 (Solaris)**

Directory Preparation Tool ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Release 3 Directory Preparation Tool は Directory Server とともにインストールされており、Directory Server をホストしている任意のコンピュータ上に存在します。
- Directory Preparation Tool のアップグレードは、Messaging Server、Calendar Server、Communications Express、または Delegated Administrator コンポーネントで使用されているすべての Directory Server インスタンスをホストしているコンピュータ上で実行する必要があります。



- Solaris OS での Release 4 Directory Preparation Tool のアップグレードパッチを次の表に示します。

表 12-3 Solaris で Directory Preparation Tool をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC Solaris 8、9、10	x86 Solaris 9、10
Directory Preparation Tool (DSSETUP)	118245-05	118246-05

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

次に示す手順は、Directory Server が存在するコンピュータにインストールされた Directory Preparation Tool に適用されます。

1. 表 12-3 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. 表 12-3 の適切な Directory Preparation Tool のパッチを適用します。

```
patchadd patch_ID
```

4. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 3 で適用したパッチ ID のバージョンが返されるはずですが。

5. Directory Server に対して Directory Preparation Tool を実行します。

- a. Directory Server が稼働していることを確認します。

- b. Directory Preparation Tool が格納されているディレクトリに移動します。

```
cd DirPrepTool-base/sbin
```

- c. Directory Preparation Tool (comm\_dssetup.pl Perl スクリプト) を実行します。

```
perl comm_dssetup.pl
```

スクリプトに必要なパラメータを指定します。

## Release 3 Directory Preparation Tool のアップグレード (Linux)

ここでは、Directory Preparation Tool のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Directory Preparation Tool ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (240 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Directory Preparation Tool のアップグレードパッチを次の表に示します。

表 12-4 Linux で Directory Preparation Tool をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Directory Preparation Tool (DSSETUP)	118247-05 <ul style="list-style-type: none"> <li>sun-comms-dssetup-6.3-2.5.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、Directory Server が存在するコンピュータ上の Directory Preparation Tool インストールイメージに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 12-4 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. 表 12-4 の Directory Preparation Tool の RPM を適用します。

```
rpm -Uvh sun-comms-dssetup-6.3-2.5.i386.rpm
```

4. アップグレードが成功したことを確認します。

```
rpm -q sun-comms-dssetup
```

RPM の新しいバージョン番号が返されることを確認します。

5. Directory Server に対して Directory Preparation Tool を実行します。

- a. Directory Server が稼働していることを確認します。

- b. Directory Preparation Tool が格納されているディレクトリに移動します。

```
cd DirPrepTool-base/sbin
```

- c. Directory Preparation Tool (comm\_dssetup.pl Perl スクリプト) を実行します。

```
perl comm_dssetup.pl
```

スクリプトに必要なパラメータを指定します。

## アップグレードの確認

Directory Preparation Tool が正常にアップグレードされ、ディレクトリスキーマが拡張されたかどうかは、スクリプトの実行時に作成されたログファイルを調べることで確認できます。このログファイルは、次の場所に格納されています。

```
/var/tmp/dssetup_yyyymmddhhmmss
```

## アップグレード後の作業

[241 ページの「アップグレード手順 \(Solaris\)」](#) および [242 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Directory Preparation Tool のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Directory Preparation Tool の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順の逆になります。ただし、Directory Preparation Tool によって加えられた変更の中には、Directory Server スキーマへの変更が含まれています。これらの変更は、次に説明するロールバック手順ではバックアウトされませんが、スキーマの変更は下位互換性があります。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. [241 ページの表 12-3](#) のパッチを削除します。

`patchrm patch_ID`

3. Directory Server に対して、ロールバックされた Directory Preparation Tool を実行します。

インデックスを含む Directory Server の変更は以前の状態に復元されますが、スキーマの変更はそのまま残ります。スキーマの変更は下位互換性があるため、スキーマの拡張に悪影響はありません。

## Directory Preparation Tool の Java ES Release 2 からのアップグレード

Java ES 2004Q2 (Release 2) Directory Preparation Tool を Release 4 にアップグレードする手順は、次の例外を除き、Release 3 Directory Preparation Tool の Release 4 へのアップグレードとほぼ同じです。

Java ES Release 2 では、Directory Preparation Tool ( そのときの名称は `comm_dssetup` ) は Messaging Server および Calendar Server に含まれており、別のパッケージとしてはインストールされませんでした。したがって、Directory Server をホストしているコンピュータには、Directory Preparation Tool のインストールされたパッケージや RPM は存在しません。このため、Release 2 から Release 4 にアップグレードするには、Directory Preparation Tool パッケージをインストールする必要があります。

- Solaris プラットフォームでは、完全な Directory Preparation Tool ソフトウェアを含む DPT パッケージが開始パッチとしてインストールされます。次に、Release 4 にアップグレードするためのパッチを適用します。
- Linux プラットフォームでは、Release 4 パッケージが直接インストールされます。

Release 2 Directory Preparation Tool から Release 4 へのアップグレードは、[238 ページの「Directory Preparation Tool の Java ES Release 3 からのアップグレード」](#)で説明されている手順とほぼ同じです。アップグレード前とアップグレード後の注意点も、手順に示されている Release 3 をすべて Release 2 に置き換えることを除いて同じです。ただし、具体的なアップグレード手順は次の節で説明されています。

## Release 2 のアップグレード手順 (Solaris)

次に示す手順は、Directory Server が存在するコンピュータにインストールされた Directory Preparation Tool に適用されます。

1. 次の表に基づいて、必要な開始パッチを入手します。

表 12-5 Solaris で Directory Preparation Tool をアップグレードするための開始パッチ<sup>1</sup>

説明	SPARC Solaris 8、9、10	x86 Solaris 9、10
Directory Preparation Tool (DSSETUP)	118242-03	118243-03

1. この表に示されているパッチのリリース番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリリース番号のパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. 表 12-5 の Directory Preparation Tool の開始パッチを適用します。

```
patchadd patch_ID
```

4. 表 12-3 に基づいて、必要なアップグレードパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

5. 表 12-3 の適切な Directory Preparation Tool のアップグレードパッチを適用します。

```
patchadd patch_ID
```

6. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 5 で適用したパッチ ID のバージョンが返されるはずですが。

7. Directory Server に対して Directory Preparation Tool を実行します。

- a. Directory Server が稼働していることを確認します。

- b. Directory Preparation Tool が格納されているディレクトリに移動します。

```
cd DirPrepTool-base/sbin
```

- c. Directory Preparation Tool (`comm_dssetup.pl` Perl スクリプト) を実行します。

```
perl comm_dssetup.pl
```

スクリプトに必要なパラメータを指定します。

## Release 2 のアップグレード手順 (Linux)

次に示す手順は、Directory Server が存在するコンピュータにインストールされた Directory Preparation Tool に適用されます。

1. `root` としてログインするか、スーパーユーザーになります。

```
su -
```

2. 242 ページの表 12-4 のパッチ番号と RPM 名に基づいて、必要なアップグレードパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

3. 242 ページの表 12-4 の Directory Preparation Tool のアップグレード RPM を適用します。

```
rpm -Uvh sun-comms-dssetup-6.3-2.5.i386.rpm
```

4. アップグレードが成功したことを確認します。

```
rpm -q sun-comms-dssetup
```

RPM の新しいバージョン番号が返されることを確認します。

5. Directory Server に対して Directory Preparation Tool を実行します。

- a. Directory Server が稼働していることを確認します。

- b. Directory Preparation Tool が格納されているディレクトリに移動します。

```
cd DirPrepTool-base/sbin
```

- c. Directory Preparation Tool (`comm_dssetup.pl` Perl スクリプト) を実行します。

```
perl comm_dssetup.pl
```

スクリプトに必要なパラメータを指定します。

# Messaging Server

この章では、Messaging Server を Java ES 2005Q4 (Release 4): Sun Java System Messaging Server 6.2 2005Q4 にアップグレードする方法について説明します。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [248 ページの「Messaging Server のアップグレードの概要」](#)
- [251 ページの「Java ES Release 3 からの Messaging Server のアップグレード」](#)
- [261 ページの「Java ES Release 2 からの Messaging Server のアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*MessagingServer-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Messaging Server が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*MessagingServer-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWmsgsr
  - Linux: /opt/sun/messaging
-

# Messaging Server のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Messaging Server の次の一般的な側面について説明します。

- [Java ES Release 4 の Messaging Server について](#)
- [Messaging Server のアップグレードロードマップ](#)
- [Messaging Server のデータ](#)
- [互換性の問題](#)
- [Messaging Server 依存関係](#)

## Java ES Release 4 の Messaging Server について

Java ES Release 4 Messaging Server は、大部分がバグ修正です。Release 3 に対して特に重要な新機能は追加されていません。

## Messaging Server のアップグレードロードマップ

表 13-1 は、Messaging Server の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 13-1 Java ES Release 4: Sun Java System Messaging Server 6.2 2005Q4 へのアップグレードパス

Java ES のリリース	Messaging Server バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Messaging Server 6.2 2005Q1	直接アップグレード: パッチを適用することにより実行されます。	設定ファイルおよび設定ディレクトリデータ
Release 2	Sun Java System Messaging Server 6.1 2004Q2	直接アップグレード: パッチを適用することにより実行されます。	設定ファイルおよび設定ディレクトリデータ



表 13-1 Java ES Release 4: Sun Java System Messaging Server 6.2 2005Q4 へのアップグレードパス ( 続き )

Java ES のリリース	Messaging Server バージョン	一般的な手順	必要な再設定
Release 1	Sun ONE Messaging Server 6.0 (2003Q4)	<p>直接アップグレード不可: まず Release 3 にアップグレードする必要があります。手順については、『Sun Java Enterprise System 2005Q1 アップグレードと移行』(<a href="http://docs.sun.com/source/819-2235?l=ja">http://docs.sun.com/source/819-2235?l=ja</a>) を参照してください。</p> <p>次に、Release 3 から Release 4 にアップグレードします。</p>	設定ファイルおよび設定ディレクトリデータ
それ以前の Java ES リリース	Sun ONE Messaging Server 5.2	<p>直接アップグレード不可: まず Release 3 にアップグレードする必要があります。手順については、『Java Enterprise System 2005Q1 アップグレードと移行』(<a href="http://docs.sun.com/doc/819-2235?l=ja">http://docs.sun.com/doc/819-2235?l=ja</a>) を参照してください。</p> <p>次に、Release 3 から Release 4 にアップグレードします。</p>	設定ファイルおよび設定ディレクトリデータ

## Messaging Server のデータ

次の表は、Messaging Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 13-2 Messaging Server のデータの使用方法

データの種類	場所	使用法
設定データ	ローカルの設定ディレクトリ: <i>MessagingServer-base</i> /config/msg.conf および、Messaging Server Store、MTA、MMP、MEM (webmail) を設定するためのその他多くの設定ファイル	Messaging Server コンポーネントの設定
設定データ	Directory Server の設定ディレクトリ	Messaging Server コンポーネントの設定

表 13-2 Messaging Server のデータの使用方法 ( 続き )

データの種類	場所	使用方法
ユーザーデータ	Directory Server のユーザー / グループディレクトリ	エンドユーザーへのメッセージングのサポートに必要なユーザー属性の格納
動的なアプリケーションデータ	Messaging Server ストア : <i>MessagingServer-base/</i>	ユーザーの代わりに、電子メールメッセージ、メッセージ転送キュー、および関連情報を格納
ディレクトリスキーマ	Directory Server <i>/var/opt/mps/serverroot</i>	エンドユーザーのサポートに必要なユーザー属性の格納

## 互換性の問題

Release 4 Messaging Server では、インタフェースの変更は導入されていません。Messaging Server の論理的に別個の構成である Messaging Server Store、MTA、MMP、および MEM コンポーネントは、以前のバージョンと下位互換性があります。

## Messaging Server 依存関係

ほかの Java ES コンポーネントに対する Messaging Server の依存関係が、Messaging Server ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Messaging Server のインタフェースまたは関数を変更された場合は、Messaging Server が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Messaging Server は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント :** Messaging Server は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Directory Server:** Messaging Server は、Directory Server のメッセージングに必要な設定データとユーザーデータを格納します。そのため、Messaging Server のアップグレードによって、ディレクトリスキーマの拡張が必要になる場合があります。
- **Directory Preparation Tool:** Messaging Server は Directory Preparation Tool を使用して、Directory Server が Messaging Server の機能をサポートできるように準備します。

- **Access Manager (任意):** Web ベースのサービスでシングルユーザーサインオンをサポートするソフトウェアソリューションのために、Messaging Server は、Access Manager のシングルサインオン機能を使用するように設定できます。
- **Delegated Administrator (任意):** Delegated Administrator は、Messaging Server がメッセージングサービスの提供に必要なユーザーデータにアクセスできるように、Directory Server でユーザーをプロビジョニングする場合に使用することを推奨されるユーティリティーです。

## Java ES Release 3 からの Messaging Server のアップグレード

ここでは、Messaging Server の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Messaging Server のアップグレード](#)

### はじめに

Java ES Release 3 Messaging Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順:** このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。再設定は、2つのデータ設定ユーティリティーを実行し、Directory Server に設定データをインポートすることにより行われます。
- **アップグレードの依存関係:** Messaging Server はいくつかの Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Release 4 Messaging Server では NSS、NSPR、LDAP C SDK、ICU、および SASL を Release 4 にアップグレードする必要があります。J2SE のアップグレードは、Messaging Server の Release 4 へのアップグレードに関しては省略可能です。

さらに、[250 ページの「Messaging Server 依存関係」](#) で説明されているように、Release 4 Messaging Server は Directory Server に依存し、また任意で Access Manager に依存します。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Messaging Server の Release 4 へのアップグレードに関しては省略可能です。

ただし、Release 4 Messaging Server は Directory Preparation Tool に対するアップグレード依存関係が強いため、メッセージング操作が可能なように Directory Server を準備するには、Release 4 Directory Preparation Tool が必要です。

- **下位互換性** : Release 4 Messaging Server は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Messaging Server の Release 4 アップグレードの Release 3 へのロールバックは、まず Directory Server に加えられた変更を削除し、ローカルの設定ファイルへの変更を削除してから、アップグレード中に適用されたパッチを削除することにより行われます。
- **プラットフォームの問題** : Messaging Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Messaging Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Messaging Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Messaging Server のアップグレード \(Solaris\)](#)
- [Release 3 の Messaging Server のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Messaging Server をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Messaging Server の現在のバージョンは、次のコマンドを入力することにより確認できます。

```
MessagingServer-base/sbin/imsimta version
```

表 13-3 Messaging Server のバージョン確認の出力

Java ES のリリース	Messaging Server のバージョン番号
Release 2	6.1
Release 3	6.2
Release 4	6.2p3

### Messaging Server が依存するコンポーネントのアップグレード

一般には、コンピュータシステム上(および、コンピューティング環境内)のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Messaging Server は、SASL 共有コンポーネントと Directory Preparation Tool に対してのみ強いアップグレード依存関係を持っています。したがって、Messaging Server が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

ただし、Messaging Server のすべての依存関係をアップグレードすることを選択した場合は、Messaging Server をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: NSS、NSPR、LDAP C SDK、ICU、および SASL 共有コンポーネントを Release 4 にアップグレードする手順は、[55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」](#)で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、[105 ページの第 4 章「Directory Server と Administration Server」](#)で説明されています。
3. **Access Manager (省略可能)**: Access Manager を Release 4 にアップグレードする手順は、[205 ページの第 11 章「Access Manager」](#)で説明されています。
4. **Directory Preparation Tool**: Release 4 Messaging Server を設定する前に、Directory Server に対して Release 4 Directory Preparation Tool を実行する必要があります。Directory Server に対して Release 4 Directory Preparation Tool をまだ実行していない場合は、Directory Preparation Tool を Release 4 にアップグレードし、それを使用して Directory Server のスキーマを変更および拡張します(手順については、[235 ページの第 12 章「Directory Preparation Tool」](#)を参照)。

## Messaging Server のデータのバックアップ

Messaging Server の Release 3 から Release 4 へのアップグレードでは、ローカルの設定ファイルと Directory Server の設定ディレクトリでの Messaging Server の再設定が必要です。ローカルの変更はロールバックできますが、将来のある時点で Release 4 アップグレードのロールバックが必要になった場合に備えて、設定ディレクトリをバックアップすることをお勧めします。

## 必要な設定情報とパスワードの入手

Messaging Server のアップグレードでは、次の情報が必要です。

- スーパーユーザーのパスワード
- ディレクトリマネージャーの DN とパスワード

## Release 3 の Messaging Server のアップグレード (Solaris)

ここでは、Messaging Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Messaging Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- 同じ Messaging Server インストールイメージに対応する、Messaging Server Store、MTA、MMP、MEM などのすべての Messaging Server コンポーネントが同時にアップグレードされます。インストールイメージにパッチを適用する前に、このようなコンポーネントをすべて停止するようにしてください。
- Solaris OS での Release 4 Messaging Server のアップグレードパッチを次の表に示します。

表 13-4 Solaris で Messaging Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Messaging Server コア	118207-38	118208-38
Messaging Server ローカライズ版	117784 -15	117785 -15

1. この表に示されているパッチのリリース番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリリース番号のパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

## アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の同じ Messaging Server インストールイメージに対応するすべての Messaging Server コンポーネントに適用されます。

1. 表 13-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. 実行中のすべての Messaging Server コンポーネントを停止します。

```
MessagingServer-base/sbin/stop-msg
```

4. 必要な共有コンポーネントの Release 4 へのアップグレードをまだ実行していない場合は、実行します。

[253 ページの「Messaging Server が依存するコンポーネントのアップグレード」](#)

5. 表 13-4 の適切な Messaging Server パッチを適用します。

```
patchadd patch_ID
```

6. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 5 で適用したパッチ ID のバージョンが返されるはずですが。

7. 設定データを、既存の設定ファイルから Release 4 設定ファイルに移行します。

- a. 設定ファイルの候補を作成します。

```
cd MessagingServer-base/sbin
./patch-config MessagingServer-base/install/patch/patch_ID
```

このコマンドは、既存の設定ファイルをバックアップします。次に、これらのファイル内の設定パラメータ値を Release 4 のテンプレート設定ファイルにマージして、新しい Release 4 設定ファイルの候補を作成します。パッチ 118209 の readme ファイルの「Special Installation Instructions」の説明に従って、競合が発生していないかどうか、これらの新規ファイルを確認してください。

このコマンドはまた、次の ldif ファイル (LDAP ディレクトリインポートファイル) も生成します。

```
MessagingServer-base/lib/patch/cfgdir_diff.ldif
MessagingServer-base/lib/patch/ugdir_diff.ldif
```

- b. Release 4 設定ファイルの候補をインストールして、それをアクティブな設定にします。

```
./install-newconfig MessagingServer-base/install/patch/patch_ID
```

このコマンドにより、新しい Release 4 設定ファイルが Release 4 の正しい場所にインストールされます。

**注 :** Solaris 10 プラットフォームで install-newconfig コマンドが失敗した場合は、コマンド実行時にライブラリパスを NULL に設定するようにしてください。

```
LD_LIBRARY_PATH= ./install-newconfig  
MessagingServer-base/install/patch/patch_ID
```

- c. 255 ページの[手順 a](#) で生成された新しい設定データを、Messaging Server が使用している Directory Server の設定ディレクトリにインポートします。

設定ディレクトリに移動し、ldapmodify コマンドを使用して ldif ファイルをインポートします。

```
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. [手順 3](#) で停止した Messaging Server コンポーネントを再起動します。

```
MessagingServer-base/sbin/start-msg
```

## Release 3 の Messaging Server のアップグレード (Linux)

ここでは、Messaging Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Messaging Server ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 ([254 ページの「アップグレードの注意点 \(Solaris\)」](#)を参照) を考慮に入れてください。

Linux OS での Release 4 Messaging Server のアップグレードパッチを次の表に示します。



表 13-5 Linux で Messaging Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
S/MIME を含む Messaging Server コアソフトウェア	118209-38 <ul style="list-style-type: none"> <li>sun-messaging-server-6.1-12.38.i386.rpm</li> </ul>
Messaging Server ローカライズ版	117786-15 <ul style="list-style-type: none"> <li>sun-messaging-110n-<i>Locale</i>-6.1-8.15.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の同じ Messaging Server インストールイメージに対応するすべての Messaging Server コンポーネントに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

- 表 13-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。  
 パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。
- root としてログインするか、スーパーユーザーになります。  

```
su -
```
- 実行中のすべての Messaging Server コンポーネントを停止します。  

```
MessagingServer-base/sbin/stop-msg
```
- 必要な共有コンポーネントの Release 4 へのアップグレードをまだ実行していない場合は、実行します。  
[253 ページの「Messaging Server が依存するコンポーネントのアップグレード」](#)
- 表 13-5 の Messaging Server の RPM を適用します。  
 次に例を示します。  

```
rpm -Fvh sun-messaging-server-6.1-12.38.i386.rpm
```
- パッチのアップグレードが成功したことを確認します。

```
rpm -q sun-messaging-server
```

RPM の新しいバージョン番号が返されることを確認します。

7. 設定データを、既存の設定ファイルから Release 4 設定ファイルに移行します。

- a. 設定ファイルの候補を作成します。

```
cd MessagingServer-base/sbin
./patch-config MessagingServer-base/install/patch/patch_ID
```

このコマンドは、既存の設定ファイルをバックアップします。次に、これらのファイル内の設定パラメータ値を Release 4 のテンプレート設定ファイルにマージして、新しい Release 4 設定ファイルの候補を作成します。パッチ 118209 の readme ファイルの「Special Installation Instructions」の説明に従って、競合が発生していないかどうか、これらの新規ファイルを確認してください。

このコマンドはまた、次の ldif ファイル (LDAP ディレクトリインポートファイル) も生成します。

```
MessagingServer-base/lib/patch/cfgdir_diff.ldif
MessagingServer-base/lib/patch/ugdir_diff.ldif
```

- b. Release 4 設定ファイルの候補をインストールして、それをアクティブな設定にします。

```
./install-newconfig MessagingServer-base/install/patch/patch_ID
```

このコマンドにより、新しい Release 4 設定ファイルが Release 4 の正しい場所にインストールされます。

- c. [258 ページの手順 a](#) で生成された新しい設定データを、Messaging Server が使用している Directory Server の設定ディレクトリにインポートします。

設定ディレクトリに移動し、ldapmodify コマンドを使用して ldif ファイルをインポートします。

```
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. [手順 3](#) で停止した Messaging Server コンポーネントを再起動します。

```
MessagingServer-base/sbin/start-msg
```

## アップグレードの確認

Messaging Server の現在のバージョンは、次のコマンドを入力することにより確認できます。

`MessagingServer-base/sbin/imsimta` version

また、Messaging Server コンポーネントを起動したときに表示されるバナーを確認することもできます。

出力値については、[253 ページの表 13-3](#) を参照してください。

## アップグレード後の作業

[255 ページの「アップグレード手順 \(Solaris\)」](#) および [257 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Messaging Server のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Messaging Server の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。再設定がロールバックされ、パッチが削除されます。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. 実行中のすべての Messaging Server コンポーネントを停止します。

```
MessagingServer-base/sbin/stop-msg
```

3. Messaging Server が使用している Directory Server の設定ディレクトリに加えられた変更をロールバックします。

ディレクトリを、アップグレード手順を開始する前にバックアップしたアップグレード前のディレクトリに置き換えます ([254 ページの「Messaging Server のデータのバックアップ」](#) を参照)。

4. [255 ページの手順 7](#) で実行された再設定をロールバックします。

```
cd MessagingServer-base/sbin
./uninstall-newconfig MessagingServer-base/install/patch/patch_ID
```

5. [254 ページの表 13-4](#) のパッチを削除します。

```
patchrm patch_ID
```

6. [手順 2](#) で停止した Messaging Server コンポーネントを再起動します。

```
MessagingServer-base/sbin/start-msg
```

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、高可用性とスケーラビリティを実現するために、Messaging Server が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Messaging Server MTA または Messaging Server MMP コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。また、Sun Cluster 環境で Messaging Server Store コンポーネントを実行して、高可用性を実現することもできます。

Messaging Server のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Messaging Server インスタンスをアップグレードする順次アップグレードを実行できます。Messaging Server の各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。さまざまな MS サブコンポーネント (MS Store、MTA、MMP、MEM) が別のコンピュータに配備されている配備アーキテクチャーでは、バックエンド層 (MS Store) から開始し、フロントエンド層 (MEM など) に向かってコンポーネントをアップグレードします。[252 ページの「Release 3 の Messaging Server のアップグレード」](#)の説明に従って、各インスタンスのアップグレードを実行します。

クラスタ環境内で実行されている Messaging Server インスタンスの場合は、これらのインスタンスが同じ設定を共有します。そのため、Messaging Server のアップグレードパッチを各インスタンスに適用する必要がありますが、すべてのインスタンスにパッチを適用したら、アップグレード手順の再設定の部分は 1 回実行するだけで済みます。

# Java ES Release 2 からの Messaging Server のアップグレード

Java ES 2004Q2 (Release 2) Messaging Server を Release 4 にアップグレードする手順は、次に説明するいくつかの例外を除き、Release 3 Messaging Server を Release 4 にアップグレードする手順と同じです。

## Messaging Server が依存するコンポーネントをアップグレードする

Release 3 からのアップグレードと異なり、Release 2 から Release 4 へのアップグレード前の作業には、すべての共有コンポーネント ([42 ページの表 1-6](#) を参照) と、Messaging Server が依存するローカルにあるすべての製品コンポーネントの Release 4 へのアップグレードを含めるようにしてください。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、[55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」](#) で説明されています。
2. **Directory Server**: Directory Server が Messaging Server と同じコンピュータ上に存在することはほとんどありませんが、Directory Server を Release 4 にアップグレードする手順は、[125 ページの「Java ES Release 2 からの Directory Server と Administration Server のアップグレード」](#) で説明されています。
3. **Access Manager (省略可能)**: Access Manager を Release 4 にアップグレードする手順は、[227 ページの「Java ES Release 2 からの Access Manager のアップグレード」](#) で説明されています。
4. **Directory Preparation Tool**: Directory Preparation Tool が Messaging Server と同じコンピュータ上に存在することはほとんどありませんが、Directory Preparation Tool をアップグレードし、それを Directory Server に対して実行する手順は、[244 ページの「Directory Preparation Tool の Java ES Release 2 からのアップグレード」](#) で説明されています。

## Release 2 Messaging Server のアップグレード

Messaging Server を Release 2 から Release 4 にアップグレードする手順は、オペレーティングシステムプラットフォームによって異なります。

### Release 2 の Messaging Server のアップグレード (Solaris)

Release 2 Messaging Server を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、[262 ページの「Release 2 の Messaging Server のアップグレード \(Solaris\)」](#)の手順をそのまま使用してください。

### Release 2 Messaging Server のアップグレード (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の同じ Messaging Server インストールイメージに対応するすべての Messaging Server コンポーネントに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. root としてログインするか、スーパーユーザーになります。  
`su -`
2. 実行中のすべての Messaging Server コンポーネントを停止します。  
`MessagingServer-base/sbin/stop-msg`
3. 必要な共有コンポーネントの Release 4 へのアップグレードをまだ実行していない場合は、実行します。

[261 ページの「Messaging Server が依存するコンポーネントをアップグレードする」](#)を参照してください。

4. Release 2 の RPM パッケージをアンインストールします。

```
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. [257 ページの表 13-5](#) の Messaging Server の RPM をインストールします。

```
rpm -i sun-messaging-server-6.1-12.38.i386.rpm
```

- パッチのアップグレードが成功したことを確認します。

```
rpm -q sun-messaging-server
```

新しくインストールされた RPM のバージョン番号が返されるはずです。

- 古い Release 2 の設定を保存します。

設定ファイルは、*MessagingServer-base/config* に格納されています。

- Messaging Server 設定プログラムを実行します。

```
cd MessagingServer-base/sbin
./configure
```

- Release 2 の設定値を新しい Release 4 の設定エントリに手動でマージします。

- 手順 2 で停止した Messaging Server コンポーネントを再起動します。

```
MessagingServer-base/sbin/start-msg
```

configutil コマンドを使用して HTTP ポートを変更する方法などの詳細については、パッチ 118209-38 の readme ファイルの「Special Installation Instructions」を参照してください。

## アップグレードの確認

Messaging Server の現在のバージョンは、次のコマンドを入力することにより確認できます。

```
MessagingServer-base/sbin/imsimta version
```

また、Messaging Server コンポーネントを起動したときに表示されるバナーを確認することもできます。

出力値については、253 ページの表 13-3 を参照してください。

## アップグレード後の作業

255 ページの「アップグレード手順 (Solaris)」および 257 ページの「アップグレード手順 (Linux)」で説明している手順以外に、アップグレード後の作業はありません。





# Calendar Server

この章では、Calendar Server を Java ES 2005Q4 (Release 4): Sun Java System Calendar Server 6.2 2005Q4 にアップグレードする方法について説明します。ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [266 ページの「Calendar Server のアップグレードの概要」](#)
- [268 ページの「Java ES Release 3 からの Calendar Server のアップグレード」](#)
- [276 ページの「Java ES Release 2 からの Calendar Server のアップグレード」](#)

---

**注**

この章で説明するファイルの場所は、*CalendarServer-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Calendar Server が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*CalendarServer-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWics5
  - Linux: /opt/sun/calendar
-

# Calendar Server のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Calendar Server の次の一般的な側面について説明します。

- [Java ES Release 4 の Calendar Server について](#)
- [Calendar Server のアップグレードロードマップ](#)
- [Calendar Server のデータ](#)
- [互換性の問題](#)
- [Calendar Server 依存関係](#)

## Java ES Release 4 の Calendar Server について

Java ES Release 4 Calendar Server は、大部分がバグ修正です。Release 3 に対して特に重要な新機能は追加されていません。

## Calendar Server のアップグレードロードマップ

表 14-1 は、Calendar Server の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 14-1 Java ES Release 4: Sun Java System Calendar Server 6.2 2005Q4 へのアップグレードパス

Java ES のリリース	Calendar Server バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Calendar Server 6 2005Q1	直接アップグレード： パッチを適用することにより実行します。	なし
Release 2	Sun Java System Calendar Server 6 2004Q2	直接アップグレード： パッチを適用することにより実行します。	設定ファイル
Release 1	Sun ONE Calendar Server 6.0 (2003Q4)	直接アップグレードは動作保証なし： ただし、パッチを適用することにより実行できます。	設定ファイル
それ以前の Java ES リリース	以前のすべてのバージョン	直接アップグレード不可。	

## Calendar Server のデータ

次の表は、Calendar Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 14-2 Calendar Server のデータの使用方法

データの種類	場所	使用法
設定データ	<code>etc/CalendarServer-base/cal/config/ics.conf</code>	Calendar Server の設定
動的なアプリケーションデータ	Calendar Server データベース： <code>/var/CalendarServer-base/csdb</code>	ユーザーの代わりにカレンダーエントリを格納します。
ディレクトリスキーマ	Directory Server のユーザー / グループディレクトリ	エンドユーザーのサポートに必要なユーザー属性の格納

## 互換性の問題

Release 4 Calendar Server では、インタフェースの変更は導入されていません。Calendar Server は、以前のバージョンと下位互換性があります。

## Calendar Server 依存関係

ほかの Java ES コンポーネントに対する Calendar Server の依存関係が、Calendar Server ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Calendar Server のインタフェースまたは関数に変更された場合は、Calendar Server が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Calendar Server は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Calendar Server は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Directory Server**: Calendar Server は、Directory Server に格納されているユーザーデータにアクセスします。そのため、Calendar Server のアップグレードによって、ディレクトリスキーマの拡張が必要になる場合があります。
- **Directory Preparation Tool**: Calendar Server は Directory Preparation Tool を使用して、ディレクトリが Calendar Server の機能をサポートできるように準備します。

- **Access Manager (任意):** Web ベースのサービスでシングルユーザーサインオンをサポートするソフトウェアソリューションのために、Calendar Server は、Access Manager のシングルサインオン機能を使用するように設定できます。
- **Messaging Server (任意):** Calendar Server は、Messaging Server を使用してカレンダーイベントのメッセージ通知を提供するように設定できます。
- **Delegated Administrator (任意):** Delegated Administrator は、Calendar Server がカレンダーサービスの提供に必要なユーザーデータにアクセスできるように、Directory Server でユーザーをプロビジョニングする場合に使用することが推奨されるユーティリティです。

## Java ES Release 3 からの Calendar Server のアップグレード

ここでは、Calendar Server の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Calendar Server のアップグレード](#)
- [複数インスタンスのアップグレード](#)

### はじめに

Java ES Release 3 Calendar Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順:** このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。
- **アップグレードの依存関係:** Calendar Server はいくつかの Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Release 4 Calendar Server では SASL を Release 4 にアップグレードすることだけが必要です。ほかの共有コンポーネントのアップグレードは、Calendar Server のアップグレードに関しては省略可能です。

さらに、[267 ページの「Calendar Server 依存関係」](#)で説明されているように、Release 4 Calendar Server は Directory Server に依存し、また任意で Access Manager に依存します。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Calendar Server の Release 4 へのアップグレードに関しては省略可能です。

ただし、Release 4 Calendar Server は、Directory Preparation Tool に対するアップグレード依存関係が強いため、カレンダー操作が可能のように Directory Server を準備するには、Release 4 Directory Preparation Tool が必要です。

- **下位互換性** : Release 4 Calendar Server は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Calendar Server の Release 4 アップグレードの Release 3 へのロールバックは、アップグレード中に適用されたパッチを削除することにより行われます。
- **プラットフォームの問題** : Calendar Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Calendar Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Calendar Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Calendar Server のアップグレード \(Solaris\)](#)
- [Release 3 の Calendar Server のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Calendar Server をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Calendar Server の現在のバージョンは、次のコマンドを使用することにより確認できます。

```
Solaris:
cd CalendarServer-base/cal/bin
./cshttpd -#
```

```
Linux:
cd CalendarServer-base/bin
./cshttpd -#
```

注: Solaris 10 プラットフォームで cshttpd コマンドが失敗した場合は、コマンド実行時にライブラリパスを NULL に設定してください。

```
LD_LIBRARY_PATH= ./cshttpd -#
```

表 14-3 Calendar Server のバージョン確認の出力

Java ES のリリース	Calendar Server のバージョン番号
Release 2	2004Q2
Release 3	2005Q1

### 必要なオペレーティングシステムのパッチの適用

Solaris 10 オペレーティングシステムプラットフォームでは、Delegated Administrator のアップグレード手順を実行するために、オペレーティングシステムのパッチを適用する必要があります (33 ページの「必須のオペレーティングシステムパッチ」を参照)。

### Calendar Server が依存するコンポーネントをアップグレードする

一般には、コンピュータシステム上 (および、コンピューティング環境内) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Calendar Server は、Directory Preparation Tool に対してのみ強いアップグレード依存関係を持っています。したがって、Calendar Server が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

ただし、Calendar Server のすべての依存関係をアップグレードすることを選択した場合は、Calendar Server をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。
3. **Access Manager (省略可能)**: Access Manager を Release 4 にアップグレードする手順は、205 ページの第 11 章「Access Manager」で説明されています。

4. **Directory Preparation Tool: Release 4 Calendar Server** を設定する前に、**Directory Server** に対して **Release 4 Directory Preparation Tool** を実行する必要があります。**Directory Server** に対して **Release 4 Directory Preparation Tool** をまだ実行していない場合は、**Directory Preparation Tool** を **Release 4** にアップグレードし、それを使用して **Directory Server** のスキーマを変更および拡張します (手順については、[235 ページの第 12 章「Directory Preparation Tool」](#) を参照)。

### Calendar Server のデータをバックアップする

Calendar Server の Release 3 から Release 4 へのアップグレードでは、Calendar Server の再設定は必要ありません。ただし、安全策として、次の場所に格納されている Calendar Server ストアをバックアップすることもできます。

```
/var/CalendarServer-base/csdb
```

### 必要な設定情報とパスワードの入手

現在インストールされているバージョンに関する特別な情報はありません。ただし、アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。

## Release 3 の Calendar Server のアップグレード (Solaris)

ここでは、Calendar Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Calendar Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- インストールイメージにパッチを適用している間は、Calendar Server を停止するようにしてください。
- たとえば、あるコンピュータに Calendar Server バックエンドストアが、別のコンピュータに Calendar Server フロントエンドプロセス (cshttpd など) が存在するなど、Calendar Server の各種サブコンポーネントがさまざまなコンピュータに存在するアーキテクチャーの場合は、サブコンポーネントが存在するすべてのコンピュータ上でアップグレードを実行する必要があります。
- Calendar Server のアップグレードは、同じインストールイメージを使用している 1 つのコンピュータ上の Calendar Server の複数のサブコンポーネントに適用されます。
- Solaris OS での Release 4 Calendar Server のアップグレードパッチを次の表に示します。

表 14-4 Solaris で Calendar Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Calendar Server コア	116577-24	116578-24
Calendar Server ローカライズ版	117010 -23	117011 -23

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Calendar Server に適用されます。

1. 表 14-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Calendar Server が稼働している場合は停止します。

```
CalendarServer-base/cal/sbin/stop-cal
```

4. SASL 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

270 ページの「Calendar Server が依存するコンポーネントをアップグレードする」を参照してください。

5. 表 14-4 の適切な Calendar Server パッチを適用します。

```
patchadd patch_ID
```

6. パッチのアップグレードが成功したことを確認します。

```
showrev -p | grep ics
```

この出力では、手順 5 で適用したパッチ ID のバージョンが返されるはずですが。

7. 手順 3 で停止した Calendar Server を再起動します。

```
CalendarServer-base/cal/sbin/start-cal
```



## Release 3 の Calendar Server のアップグレード (Linux)

ここでは、Calendar Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Calendar Server ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (271 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Calendar Server のアップグレードパッチを次の表に示します。

表 14-5 Linux で Calendar Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Calendar Server コア	117851-24 <ul style="list-style-type: none"> <li>sun-calendar-core-6.2-10.7.i386.rpm</li> <li>sun-calendar-api-6.2-10.7.i386.rpm</li> </ul>
Calendar Server ロケール	117852-23 <ul style="list-style-type: none"> <li>sun-calendar-core-<i>Locale</i>-6.2-10.3.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Calendar Server に適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 14-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

su -

- Calendar Server が稼働している場合は停止します。

```
CalendarServer-base/sbin/stop-cal
```

- SASL 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

270 ページの「[Calendar Server が依存するコンポーネントをアップグレードする](#)」を参照してください。

- 表 14-5 の Calendar Server の RPM を適用します。

```
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
```

- パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-calendar
```

RPM の新しいバージョン番号が返されることを確認します。

- 手順 3 で停止した Calendar Server を再起動します。

```
CalendarServer-base/sbin/start-cal
```

## アップグレードの確認

Calendar Server の Release 4 へのアップグレードは、アップグレードパッチが正常に適用されていることを確認することによって検証されます。この確認の手順は、272 ページの「[アップグレード手順 \(Solaris\)](#)」および 273 ページの「[アップグレード手順 \(Linux\)](#)」に記載されています。

## アップグレード後の作業

272 ページの「[アップグレード手順 \(Solaris\)](#)」および 273 ページの「[アップグレード手順 \(Linux\)](#)」で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Calendar Server のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Calendar Server の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. Calendar Server を停止します。

```
CalendarServer-base/cal/sbin/stop-cal
```

3. 272 ページの表 14-4 のパッチを削除します。

```
patchrm patch_ID
```

4. Calendar Server を再起動します。

```
CalendarServer-base/cal/sbin/start-cal
```

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、高可用性を実現するために、Calendar Server が複数のコンピュータシステムに配備される場合があります。たとえば、Sun Cluster 環境で Calendar Server Store コンポーネントを実行して、高可用性を実現することができます。

クラスター環境内で実行されている Calendar Server インスタンスの場合は、これらのインスタンスが同じ設定を共有します。これらの各インスタンスに Calendar Server アップグレードパッチを適用する必要があります。Release 3 から Release 4 へのアップグレードの場合、必要な再設定はありません。

# Java ES Release 2 からの Calendar Server のアップグレード

Java ES 2004Q2 (Release 2) Calendar Server を Release 4 にアップグレードする手順は、Release 3 Calendar Server を Release 4 にアップグレードする手順とほぼ同じです。ただし、アップグレード前の作業に、すべての共有コンポーネント (42 ページの表 1-6 を参照) と、Calendar Server が依存するローカルにあるすべての製品コンポーネントの Release 4 へのアップグレードを含めるようにしてください。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server**: Directory Server が Calendar Server と同じコンピュータ上に存在することはほとんどありませんが、Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。
3. **Access Manager (省略可能)**: Access Manager を Release 4 にアップグレードする手順は、205 ページの第 11 章「Access Manager」で説明されています。
4. **Directory Preparation Tool**: Directory Preparation Tool が Calendar Server と同じコンピュータ上に存在することはほとんどありませんが、Directory Preparation Tool をアップグレードし、それを Directory Server に対して実行する手順は、235 ページの第 12 章「Directory Preparation Tool」で説明されています。

Release 2 Calendar Server を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、268 ページの「Java ES Release 3 からの Calendar Server のアップグレード」の手順をそのまま使用してください。

さらに、Release 2 から Release 4 へのアップグレードでは、Calendar Server のホットバックアップを設定するというアップグレード後の作業が必要になります。この作業は、Calendar Server の `ics.conf` 設定ファイルにホットバックアップパラメータを追加することにより行われます。このアップグレード後の再設定の手順については、次のサイトを参照してください。<http://docs.sun.com/doc/819-3568/6n4n1fjnjq?a=view>

# Communications Express

この章では、Communications Express を Java ES 2005Q4 (Release 4): Sun Java System Communications Express 6.2 2005Q4 にアップグレードする方法について説明します。ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- 278 ページの「[Communications Express のアップグレードの概要](#)」
- 281 ページの「[Java ES Release 3 からの Communications Express のアップグレード](#)」
- 290 ページの「[Java ES Release 2 からの Communications Express のアップグレード](#)」

---

**注** この章で説明するファイルの場所は、*CommsExpress-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Communications Express が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*CommsExpress-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWuwc
  - Linux: /opt/sun/uwc
-

# Communications Express のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Communications Express の次の一般的な側面について説明します。

- [Java ES Release 4 の Communications Express について](#)
- [Communications Express のアップグレードロードマップ](#)
- [Communications Express のデータ](#)
- [互換性の問題](#)
- [Communications Express 依存関係](#)

## Java ES Release 4 の Communications Express について

Java ES Release 4 Communications Express は、大部分がバグ修正です。Release 3 と比較して、メールフィルタのサポート、アドレス帳の共有、プロキシ認証など、いくつかの新機能があります。

## Communications Express のアップグレードロードマップ

表 15-1 は、Communications Express の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 15-1 Java ES Release 4: Sun Java System Communications Express 6.2 2005Q4 へのアップグレードパス

Java ES のリリース	Communications Express バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Communications Express 6 2005Q1	直接アップグレード：パッチを適用し、Messaging Server コンポーネントを再設定することにより実行します。	設定ファイル
Release 2	Sun Java System Communications Express 6 2004Q2	直接アップグレード：パッチを適用し、Messaging Server コンポーネントを再設定することにより実行します。	設定ファイル
Release 1	なし	アップグレードはなし：	

表 15-1 Java ES Release 4: Sun Java System Communications Express 6.2 2005Q4 へのアップグレードパス ( 続き )

Java ES のリリース	Communications Express バージョン	一般的な手順	必要な再設定
それ以前の Java ES リリース	なし	アップグレードはなし:	

## Communications Express のデータ

次の表は、Communications Express ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 15-2 Communications Express のデータの使用方法

データの種類	場所	使用法
設定データ	ローカルの設定ディレクトリ <code>var/CommsExpress-base/WEB-INF/config/uwcauth.properties</code> <code>var/CommsExpress-base/WEB-INF/config/uwconfig.properties</code> <code>var/CommsExpress-base/WEB-INF/config/uwclogging.properties</code> <code>MessagingServer-base/config/msg.conf</code> および、Messaging Server MEM (webmail) を設定するためのその他の設定ファイル	Messaging Server MEM (webmail) を含む、 Communications Express の設定
Web コンテナ の設定	Web Server: 次の場所にある <code>server.policy</code> および <code>server.xml</code> ファイル <code>WebServer-base/https-hostname/config</code>  Application Server (Java ES Release 3 および 4): 次の場所にある <code>server.policy</code> および <code>domain.xml</code> ファイル <code>AppServer8Config-base/domains/domainName/config</code>  Application Server (Java ES Release 2): 次の場所にある <code>server.policy</code> および <code>server.xml</code> ファイル <code>AppServer7Config-base/domains/domainName/config</code>	Communications Express Web コンテ ナインスタンスの設 定
ディレクトリ スキーマ	Directory Server のユーザー / グループディレクトリ	エンドユーザーのサ ポートに必要なユー ザー属性の格納

## 互換性の問題

Release 4 Communications Express ではインタフェースの変更は導入されておらず、以前のバージョンと下位互換性があります。

## Communications Express 依存関係

ほかの Java ES コンポーネントに対する Communications Express の依存関係が、Communications Express ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Communications Express のインタフェースまたは関数を変更された場合は、Communications Express が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Communications Express は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Communications Express は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Web コンテナ** : Communications Express は、Java ES Web Server または Java ES Application Server のどちらかで提供される Web コンテナサービスに依存します。
- **Access Manager (または Access Manager SDK)** : Communications Express は、シングルサインオンを含む、エンドユーザーへの認証および承認サービスを提供するために Access Manager に依存します。Access Manager をリモートコンピュータ上で動作させる場合、Access Manager SDK がローカルで利用可能である必要があります。
- **Messaging Server** : Communications Express は、Messaging Server への Web ベースアクセスを提供するために使用されます。実際、Communications Express は Messaging Server MEM コンポーネントを直接使用して、Messaging Server の Store コンポーネントや MTA コンポーネントなど、Messaging Server のほかのバックエンドコンポーネントにアクセスします。
- **Calendar Server** : Communications Express は、Calendar Server への Web ベースアクセスを提供するために使用されます。
- **Directory Server** : Communications Express は、Directory Server に設定データを格納するだけでなく、そこに格納されているユーザーデータへのアクセスも行います。そのため、Communications Express のアップグレードによって、Directory Server のアップグレードまたはディレクトリスキーマの拡張が必要になる場合があります。



- **Directory Preparation Tool: Communications Express** は Directory Preparation Tool を使用して、Directory Server が Communications Express の機能をサポートできるように準備します。そのため、Communications Express のアップグレードは、ディレクトリが新機能をサポートできるように準備されているかどうかにかかわらず場合があります。

## Java ES Release 3 からの Communications Express のアップグレード

ここでは、Communications Express の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Communications Express のアップグレード](#)
- [複数インスタンスのアップグレード](#)

### はじめに

Java ES Release 3 Communications Express を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。含まれている Messaging Server MEM コンポーネントの再設定は、2つの設定ユーティリティーを使用し、設定データを Directory Server インポートにすることにより行われます。
- **アップグレードの依存関係** : Calendar Server は多くの Java ES 共有コンポーネントに対する依存関係を持ちますが ([42 ページの表 1-6](#) を参照)、Release 4 の Calendar Server はこれらすべてのコンポーネントの Release 3 バージョンと互換性があります。したがって、これらの共有コンポーネントのアップグレードは、Calendar Server の Release 4 へのアップグレードに関しては省略可能です。

さらに、[280 ページの「Communications Express 依存関係」](#) で説明されているように、Release 4 Communications Express は Web コンテナと Access Manager に依存します。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Communications Express の Release 4 へのアップグレードに関しては省略可能です。

ただし、Communications Express は、Web ベースアクセスを提供する相手の Calendar Server と、Messaging Server MEM コンポーネントを使用して同様に Web ベースアクセスを提供する相手の Messaging Server の両方に対して強いアップグレード依存関係を持っています。したがって、Communications Express を Release 4 にアップグレードするには、Calendar Server と Messaging Server の両方を Release 4 にアップグレードする必要があります。

さらに、Release 4 Communications Express は、Directory Preparation Tool に対して強いアップグレード依存関係を持っています。Communications Express の機能を使用できるように Directory Server を準備するには、Release 4 Directory Preparation Tool が必要です。

- **下位互換性** : Release 4 Communications Express は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Communications Express の Release 4 アップグレードの Release 3 へのロールバックは、まず Directory Server に加えられた変更を削除し、ローカルの設定ファイルへの変更を削除してから、アップグレード中に適用されたパッチを削除することにより行われます。
- **プラットフォームの問題** : Communications Express をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Communications Express のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Communications Express の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Communications Express のアップグレード \(Solaris\)](#)
- [Release 3 の Communications Express のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

## アップグレード前の作業

Communications Express をアップグレードする前に、次に説明されている作業を実行してください。

### 現在のバージョン情報の確認

Communications Express の現在のバージョンは、現在のバージョン番号が表示される Communications Express のログインページにアクセスすることにより確認できます。

```
http://hostName:port/uwc/auth
```

表 15-3 Communications Express のバージョン確認の出力

Java ES のリリース	Communications Express のバージョン番号
Release 2	Sun Java System Communications Express 6 2004Q2
Release 3	Sun Java System Communications Express 6 2005Q1
Release 4	Sun Java System Communications Express 6 2005Q4

### Communications Express が依存するコンポーネントのアップグレード

一般には、コンピュータシステム上（および、コンピューティング環境内）のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Communications Express は、Messaging Server、Calendar Server、および Directory Preparation Tool に対してのみ強いアップグレード依存関係を持っています。したがって、Communications Express が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

ただし、Communications Express のすべての依存関係をアップグレードすることを選択した場合は、Communications Express をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server** : Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。
3. **Web コンテナソフトウェア** : Web Server または Application Server をアップグレードする手順は、それぞれ 141 ページの第 6 章「Web Server」および 177 ページの第 9 章「Application Server」で説明されています。

4. **Access Manager (Access Manager SDK):** Access Manager を Release 4 にアップグレードする手順は、[205 ページの第 11 章「Access Manager」](#) で説明されています。
5. **Directory Preparation Tool:** Release 4 Communications Express を設定する前に、Directory Server に対して Release 4 Directory Preparation Tool を実行する必要があります。Directory Server に対して Release 4 Directory Preparation Tool をまだ実行していない場合は、Directory Preparation Tool を Release 4 にアップグレードし、それを使用して Directory Server のスキーマを変更および拡張します (手順については、[235 ページの第 12 章「Directory Preparation Tool」](#) を参照)。
6. **Messaging Server:** Release 4 Communications Express をサポートするには、Messaging Server コンポーネントを Release 4 にアップグレードする必要があります。Messaging Server を Release 4 にアップグレードする手順は、[247 ページの第 13 章「Messaging Server」](#) で説明されています。
7. **Calendar Server:** Release 4 Communications Express をサポートするには、Calendar Server コンポーネントを Release 4 にアップグレードする必要があります。Calendar Server を Release 4 にアップグレードする手順は、[265 ページの第 14 章「Calendar Server」](#) で説明されています。

### *Communications Express のデータのバックアップ*

Communications Express の Release 3 から Release 4 へのアップグレードでは、Messaging Server MEM の再設定が必要です。ローカルの変更はロールバックできるため、データをバックアップする必要はありません。

### *必要な設定情報とパスワードの入手*

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。Web Server を Web コンテナとして使用している場合は、設定情報は必要ありません。ただし、Application Server を Web コンテナとして使用している場合は、Application Server 管理者のユーザー ID とパスワードが必要になります。

## **Release 3 の Communications Express のアップグレード (Solaris)**

ここでは、Communications Express のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### *アップグレードの注意点 (Solaris)*

Communications Express ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- Communications Express には、Calendar Server への Web ベースアクセスを提供するために使用されるコンポーネントと、Messaging Server バックエンドコンポーネントへの Web ベースアクセスを提供するために使用される Messaging Server MEM コンポーネントが含まれています。アップグレードパッチには、これらのすべてのコンポーネントが含まれています。

- すべての Communications Express コンポーネントを、同じ Web コンテナに配備するようにしてください。インストールイメージにパッチを適用する前に、Web コンテナを停止するようにしてください。
- Solaris OS での Release 4 Communications Express のアップグレードパッチを次の表に示します。

表 15-4 Solaris で Communications Express をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Communications Express コア	118540-21	118541-21
Communications Express ローカライズ版	118042-16	118042-16

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

次に示す手順は、更新するコンピュータ上のすべての Communications Express コンポーネントに適用されます。

1. 表 15-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web コンテナを停止することにより、Communications Express を停止します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. Release 4 にアップグレードする任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

[283 ページの「Communications Express が依存するコンポーネントのアップグレード」](#)を参照してください。

5. 表 15-4 の適切な Communications Express パッチを適用します。

```
patchadd patch_ID
```

6. パッチのアップグレードが成功したことを確認します。

```
showrev -p | grep uwc
```

この出力では、[手順 5](#) で適用したパッチ ID のバージョンが返されるはずですが。

7. 設定データを、既存の設定ファイルから Release 4 設定ファイルの候補に移行します。

```
cd CommsExpress-base/sbin
./patch-config CommsExpress-base/install/patch/patch_ID
```

このコマンドは、現在の設定ディレクトリの入力を求めたあと、既存の設定ファイルをバックアップします。次に、これらのファイル内の設定パラメータ値を Release 4 のテンプレート設定ファイルにマージして、新しい Release 4 設定ファイルの候補を作成します。パッチの `readme` ファイルの「**Special Installation Instructions**」の説明に従って、競合が発生していないかどうか、これらの新規ファイルを確認してください。

8. Release 4 設定ファイルの候補をインストールして、それをアクティブな設定にします。

```
./install-newconfig CommsExpress-base/install/patch/patch_ID
```

このコマンドにより、新しい Release 4 設定ファイルが Release 4 の正しい場所にインストールされます。

9. Web コンテナに保持されている、Communications Express に対する JSP クラスキャッシュを削除します。

手順については、Web コンテナ (Web Server または Application Server) のドキュメントを参照してください。

10. Web コンテナを再起動することにより、Communications Express を再起動します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

## Release 3 の Communications Express のアップグレード (Linux)

ここでは、Communications Express のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Communications Express ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 ([284 ページ](#)の「[アップグレードの注意点 \(Solaris\)](#)」を参照) を考慮に入れてください。

Linux OS での Release 4 Communications Express のアップグレードパッチを次の表に示します。

表 15-5 Linux で Communications Express をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Communications Express コア	118542-21 <ul style="list-style-type: none"> <li>sun-uwc-6.1-7.21.i386.rpm</li> </ul>
Communications Express ローカライズ版	118044-14 <ul style="list-style-type: none"> <li>sun-uwc-110n-<i>Locale</i>-6.1-11.9.i386.rpm</li> </ul>

1. この表に示されているパッチのリリース番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリリース番号のパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、更新するコンピュータ上のすべての Communications Express コンポーネントに適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 15-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web コンテナを停止することにより、Communications Express を停止します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. Release 4 にアップグレードする任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

283 ページの「[Communications Express が依存するコンポーネントのアップグレード](#)」を参照してください。

5. 表 15-5 の Communications Express の RPM を適用します。

次に例を示します。

```
rpm -Fvh sun-uwc-6.1-7.21.i386.rpm
```

6. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep uwc
```

この出力では、手順 5 で適用した RPM のバージョンが返されるはずですが。

7. 設定データを、既存の設定ファイルから Release 4 設定ファイルの候補に移行します。

```
cd CommsExpress-base/sbin
./patch-config CommsExpress-base/install/patch/patch_ID
```

このコマンドは、現在の設定ディレクトリの入力を求めたあと、既存の設定ファイルをバックアップします。次に、これらのファイル内の設定パラメータ値を Release 4 のテンプレート設定ファイルにマージして、新しい Release 4 設定ファイルの候補を作成します。パッチの readme ファイルの「**Special Installation Instructions**」の説明に従って、競合が発生していないかどうか、これらの新規ファイルを確認してください。

8. Release 4 設定ファイルの候補をインストールして、それをアクティブな設定にします。

```
./install-newconfig CommsExpress-base/install/patch/patch_ID
```

このコマンドにより、新しい Release 4 設定ファイルが Release 4 の正しい場所にインストールされます。

9. Web コンテナに保持されている、Communications Express に対する JSP クラスキャッシュを削除します。

手順については、Web コンテナ (Web Server または Application Server) のドキュメントを参照してください。

10. Web コンテナを再起動することにより、Communications Express を再起動します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```



## アップグレードの確認

Communications Express の現在のバージョンは、現在のバージョン番号が表示される Communications Express のログインページにアクセスにすることにより確認できます。

```
http://hostName:port/uwc/auth
```

ログインしたら、新しいメールタブと古い電子メールおよびカレンダーイベントに対するアップグレードされたユーザーインタフェースを確認します (これらのチャンネルを使用している場合)。

また、アップグレード処理のさまざまな手順に対する次のログファイルも確認できます。

```
CommsExpress-base/install/patch/118540-21.  
CommsExpress-base/patch-config_20050729164754.log  
CommsExpress-base/install-newconfig_20050729164838.log
```

## アップグレード後の作業

285 ページの「アップグレード手順 (Solaris)」および 287 ページの「アップグレード手順 (Linux)」で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Communications Express のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Communications Express の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。再設定がロールバックされ、パッチが削除されます。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. Web コンテナを停止することにより、Communications Express を停止します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

3. Communications Express が使用している Directory Server の設定ディレクトリに加えられた変更をロールバックします。

4. [286 ページの手順 8](#) で実行された再設定をロールバックします。

```
cd CommsExpress-base/sbin  
./uninstall-newconfig CommsExpress-base/install/patch/patch_ID
```

5. [285 ページの表 15-4](#) のパッチを削除します。

```
patchrm patch_ID
```

6. Web コンテナを再起動することにより、Communications Express を再起動します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName  
--user admin_ID --password password
```

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、スケーラビリティを実現し、可用性を向上させるために、Communications Express が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Communications Express コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。

Communications Express のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Communications Express インスタンスをアップグレードする順次アップグレードを実行できます。Communications Express の各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。[282 ページの「Release 3 の Communications Express のアップグレード」](#)の説明に従って、各インスタンスのアップグレードを実行します。

## Java ES Release 2 からの Communications Express のアップグレード

Java ES 2004Q2 (Release 2) Communications Express を Release 4 にアップグレードする手順は、次に説明するいくつかの例外を除き、Release 3 Communications Express を Release 4 にアップグレードする手順と同じです。

---

**注** ここでの説明は、Communications Express が Release 2 Web Server Web コンテナに配備されている場合に適用されますが、Communications Express が Release 2 Application Server Web コンテナに配備されている場合には適用されません。後者のケースは、現在サポートされていません。

---

## Communications Express が依存するコンポーネントをアップグレードする

Java ES Release 2 Communications Express を Release 4 にアップグレードするときのアップグレード前の作業は、Release 3 Communications Express を Release 4 にアップグレードする手順とほぼ同じです。ただし、Communications Express の依存関係のアップグレードに、すべての共有コンポーネント (42 ページの表 1-6 を参照) と、Communications Express が依存するローカルにあるすべての製品コンポーネントの Release 4 へのアップグレードを含めるようにしてください。

Communications Express の依存関係をアップグレードする場合は、Communications Express をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server**: Directory Server が Communications Express と同じコンピュータ上に存在することはほとんどありませんが、Directory Server を Release 4 にアップグレードする手順は、125 ページの「Java ES Release 2 からの Directory Server と Administration Server のアップグレード」で説明されています。
3. **Web コンテナソフトウェア** : Web Server または Application Server をアップグレードする手順は、それぞれ 150 ページの「Java ES Release 2 からの Web Server のアップグレード」および 189 ページの「Java ES Release 2 からの Application Server のアップグレード」で説明されています。
4. **Access Manager (Access Manager SDK)**: Access Manager を Release 4 にアップグレードする手順は、227 ページの「Java ES Release 2 からの Access Manager のアップグレード」で説明されています。
5. **Directory Preparation Tool**: Directory Preparation Tool が Communications Express と同じコンピュータ上に存在することはほとんどありませんが、Directory Preparation Tool をアップグレードし、それを Directory Server に対して実行する手順は、244 ページの「Directory Preparation Tool の Java ES Release 2 からのアップグレード」で説明されています。

6. **Messaging Server:** Release 4 Communications Express をサポートするには、Messaging Server の MTA を Release 4 にアップグレードする必要があります。Messaging Server を Release 4 にアップグレードする手順は、[261 ページの「Java ES Release 2 からの Messaging Server のアップグレード」](#)で説明されています。
7. **Calendar Server:** Calendar Server が Communications Express と同じコンピュータ上に存在することはほとんどありませんが、Calendar Server を Release 4 にアップグレードする手順は、[276 ページの「Java ES Release 2 からの Calendar Server のアップグレード」](#)で説明されています。

## Release 2 Communications Express のアップグレード

Communications Express を Release 2 から Release 4 にアップグレードする手順は、Communications Express ソフトウェアを配備している Web コンテナによって異なります。

### Release 2 Communications Express のアップグレード : Web Server Web コンテナ

Release 4 にアップグレードされた Web Server Web コンテナに配備する際に Release 2 Communications Express を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、[284 ページの「Release 3 の Communications Express のアップグレード \(Solaris\)」](#)または [286 ページの「Release 3 の Communications Express のアップグレード \(Linux\)」](#)の手順に従います。

### Release 2 Communications Express のアップグレード : Application Server Web コンテナ

Release 4 にアップグレードされた Application Server Web コンテナに配備する際に Release 2 Communications Express の Release 4 へのアップグレードは、現在サポートされていません。

# Instant Messaging

この章では、Instant Messaging を Java ES 2005Q4 (Release 4): Sun Java System Instant Messaging 7.0.1 2005Q4 にアップグレードする方法について説明します。ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [294 ページの「Instant Messaging のアップグレードの概要」](#)
- [297 ページの「Java ES Release 3 からの Instant Messaging のアップグレード」](#)
- [305 ページの「Java ES Release 2 からの Instant Messaging のアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*InstantMessaging-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Instant Messaging が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*InstantMessaging-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWiim
  - Linux: /opt/sun/im
-

# Instant Messaging のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Instant Messaging の次の一般的な側面について説明します。

- [Java ES Release 4 の Instant Messaging について](#)
- [Instant Messaging のアップグレードロードマップ](#)
- [Instant Messaging のデータ](#)
- [互換性の問題](#)
- [Instant Messaging 依存関係](#)

## Java ES Release 4 の Instant Messaging について

Java ES Release 4 Instant Messaging は、大部分がバグ修正です。Release 3 に対して特に重要な新機能は追加されていません。

## Instant Messaging のアップグレードロードマップ

表 16-1 は、Instant Messaging の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 16-1 Java ES Release 4: Sun Java System Instant Messaging 7.0.1 2005Q4 へのアップグレードパス

Java ES のリリース	Instant Messaging バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Instant Messaging 7.0 2005Q1	直接アップグレード: パッチを適用することにより実行されます。	なし
Release 2	Sun Java System Instant Messaging 6 2004Q2	直接アップグレード: upgrade ユーティリティを使用して実行されます。	設定データ
Release 1	Sun Java System Instant Messaging 6.1 (2003Q4)	直接アップグレードは動作保証なし: ただし、upgrade ユーティリティを使用することにより実行できます。	設定データ
それ以前の Java ES リリース		直接アップグレード不可	

## Instant Messaging のデータ

次の表は、Instant Messaging ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 16-2 Instant Messaging のデータの使用方法

データの種類	場所	使用方法
設定データ	ローカルの設定ディレクトリ  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 プロセスと登録属性の設定
Instant Messaging サーバーのリソース	ローカルの設定ディレクトリ <i>InstantMessaging-base/html</i>	Messenger クライアントを起動するためにエンドユーザーがダウンロードするカスタマイズされたクライアントファイル
動的データ	<i>runtimeFilesDir/default/db</i> ここで、 <i>runtimeFilesDir</i> はインストール時に指定されます。 Solaris: デフォルトの <i>runtimeFilesDir</i> : /var/opt/SUNWiim Linux: デフォルトの <i>runtimeFilesDir</i> : /var/opt/sun/im	すべての変数データ (Instant Messaging データベース、ログファイル、ロックファイルなど)

## 互換性の問題

Release 4 Instant Messaging ではインタフェースの変更は導入されておらず、以前のバージョンと下位互換性があります。

## Instant Messaging 依存関係

ほかの Java ES コンポーネントに対する Instant Messaging の依存関係が、Instant Messaging ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Instant Messaging のインタフェースまたは関数に変更された場合は、Instant Messaging が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Instant Messaging は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Instant Messaging は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Web コンテナ** : Instant Messaging は、Java ES Web Server または Java ES Application Server のどちらかで提供される Web コンテナサービスに依存します。
- **Directory Server (任意)** : Instant Messaging は、Directory Server にユーザーデータを格納し、それにアクセスするように設定できます。そのため、Instant Messaging のアップグレードによって、ディレクトリスキーマの拡張が必要になる場合があります。
- **Access Manager (任意)** : Web ベースのサービスでシングルユーザーサインオンをサポートするソフトウェアソリューションのために、Instant Messaging は、Access Manager のシングルサインオン機能を使用するように設定できます。



# Java ES Release 3 からの Instant Messaging のアップグレード

ここでは、Instant Messaging の Java ES Release 3 から Java ES Release 4 へのアップグレードについて説明します。ここで説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Instant Messaging のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 3 Instant Messaging を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順**：このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。Instant Messaging の再設定は必要ありません。
- **アップグレードの依存関係**：Instant Messaging はいくつかの Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Release 4 Instant Messaging では IM-SDK を Release 4 にアップグレードすることだけが必要です。ほかの共有コンポーネントのアップグレードは、Instant Messaging の Release 4 へのアップグレードに関しては省略可能です。

さらに、[296 ページの「Instant Messaging 依存関係」](#) で説明されているように、Release 4 Instant Messaging は Web コンテナに依存し、また任意で Access Manager に依存します。ただし、これらはアップグレード依存関係が弱いものであり、Web コンテナおよび Access Manager のアップグレードは、Instant Messaging の Release 4 へのアップグレードに関しては省略可能です。

- **下位互換性**：Release 4 Instant Messaging は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック**：Release 4 アップグレードの Release 3 へのロールバックは、Solaris プラットフォームで、アップグレード中に適用されたパッチを削除することにより行われます。
- **プラットフォームの問題**：Instant Messaging をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

# Release 3 の Instant Messaging のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Instant Messaging の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Instant Messaging のアップグレード \(Solaris\)](#)
- [Release 3 の Instant Messaging のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

## アップグレード前の作業

Instant Messaging をアップグレードする前に、次に説明されている作業を実行してください。

### 現在のバージョン情報の確認

Instant Messaging の現在のバージョンは、標準のバージョン確認ユーティリティを使用することにより確認できます。

```
Solaris:  
pkginfo -l SUNWiimin
```

```
Linux:  
rpm -qa | grep sun-im
```

表 16-3 Instant Messaging のバージョン確認の出力

Java ES のリリース	Instant Messaging のバージョン番号
Release 2	バージョン番号 6.x
Release 3	バージョン番号 7.0
Release 4	バージョン番号 7.0.1

### *Instant Messaging が依存するコンポーネントのアップグレード*

一般には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Instant Messaging は、IM-SDK 共有コンポーネントに対してのみ強いアップグレード依存関係を持っています。したがって、Instant Messaging が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

Instant Messaging のすべての依存関係をアップグレードすることを選択した場合は、Instant Messaging をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: IM-SDK やその他の Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Web コンテナソフトウェア**: Web Server または Application Server をアップグレードする手順は、それぞれ 141 ページの第 6 章「Web Server」および 177 ページの第 9 章「Application Server」で説明されています。
3. **Access Manager ( 省略可能 )**: Access Manager を Release 4 にアップグレードする手順は、205 ページの第 11 章「Access Manager」で説明されています。

### *Instant Messaging のデータのバックアップ*

Instant Messaging の Release 3 から Release 4 へのアップグレードでは、設定データは変更されません。ただし、安全策として、Instant Messaging をアップグレードする前に、Instant Messaging データベースおよびカスタマイズしている既存の任意のリソースや設定ファイルをバックアップすることをお勧めします。詳細については、『Sun Java System Instant Messaging 7 2005Q4 Administration Guide』 (<http://docs.sun.com/doc/819-2503>) を参照してください。

### *必要な設定情報とパスワードの入手*

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。Web Server を Web コンテナとして使用している場合は、設定情報は必要ありません。ただし、Application Server を Web コンテナとして使用している場合は、Application Server 管理者のユーザー ID とパスワードが必要になります。

### **Release 3 の Instant Messaging のアップグレード (Solaris)**

ここでは、Instant Messaging のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

## アップグレードの注意点 (Solaris)

Instant Messaging ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- インストールイメージにパッチを適用している間は、Instant Messaging コンポーネントを停止するようにしてください。
- 各種の Instant Messaging サブコンポーネントが別のコンピュータに存在するアーキテクチャー、たとえば、あるコンピュータにメッセージリソースが、別のコンピュータに Instant Messaging サーバーが、さらに別のコンピュータに Instant Messaging マルチプレクサが存在するような場合は、このようなすべてのコンピュータでアップグレードを実行する必要があります。
- Instant Messaging のアップグレードは、同じインストールイメージを使用している 1 つのコンピュータ上の Instant Messaging の複数のサブコンポーネントに適用されます。
- Solaris OS での Release 4 Instant Messaging のアップグレードパッチを次の表に示します。

表 16-4 Solaris で Instant Messaging をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Instant Messaging	118786-08	118787-08
Instant Messaging ローカライズ版	119707-06	119707-06

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

## アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Instant Messaging に適用されます。

1. 表 16-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Instant Messaging が稼働している場合は停止します。

```
InstantMessaging-base/sbin/imadmin stop
```

4. IM-SDK 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

299 ページの「[Instant Messaging が依存するコンポーネントのアップグレード](#)」を参照してください。

5. [表 16-4](#) の適切な Instant Messaging パッチを適用します。

Instant Messaging のベースパッチを適用する前に、Instant Messaging のローカライズ版パッチ (119707) を適用するようにしてください。

```
patchadd patch_ID
```

6. アップグレードが成功したことを確認します。

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

この出力では、[手順 5](#) で適用したパッチ ID のバージョンが返されるはずですが。

7. [手順 3](#) で停止した Instant Messaging サービスを再起動します。

```
InstantMessaging-base/sbin/imadmin start
```

## Release 3 の Instant Messaging のアップグレード (Linux)

ここでは、Instant Messaging のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Instant Messaging ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 ([300 ページの「アップグレードの注意点 \(Solaris\)」](#)を参照) を考慮に入れてください。

Linux OS での Release 4 Instant Messaging のアップグレードパッチを次の表に示します。

表 16-5 Linux で Instant Messaging をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Instant Messaging	118788-11 <ul style="list-style-type: none"> <li>• sun-im-client-7.0-13.8.i386.rpm</li> <li>• sun-im-server-7.0-13.8.i386.rpm</li> <li>• sun-im-mux-7.0-13.8.i386.rpm</li> <li>• sun-im-olh-7.0-13.8.i386.rpm</li> <li>• sun-im-install-7.0-13.8.i386.rpm</li> <li>• sun-im-ident-7.0-13.8.i386.rpm</li> <li>• sun-im-apidoc-7.0-13.8.i386.rpm</li> </ul>
Instant Messaging ローカライズ版	119708-06 <ul style="list-style-type: none"> <li>• sun-im-client-<i>Locale</i>-7.0-12.i386.rpm</li> <li>• sun-im-ident-<i>Locale</i>-7.0-12.i386.rpm</li> <li>• sun-im-install-<i>Locale</i>-7.0-12.i386.rpm</li> <li>• sun-im-olh-<i>Locale</i>-7.0-12.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Instant Messaging に適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 16-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

su -

- Instant Messaging が稼働している場合は停止します。

```
InstantMessaging-base/sbin/imadmin stop
```

- IM-SDK 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

299 ページの「Instant Messaging が依存するコンポーネントのアップグレード」を参照してください。

- 表 16-5 の Instant Messaging の RPM を適用します。

Instant Messaging のベースパッチを適用する前に、Instant Messaging のローカライズ版パッチ (119708) を適用するようにしてください。

```
rpm -Fvh sun-im-module-Locale-7.0-2.8.i386.rpm
rpm -Fvh sun-im-module-7.1-11.i386.rpm
```

- アップグレードが成功したことを確認します。

```
rpm -qa | grep sun-im
```

RPM の新しいバージョン番号が返されることを確認します。

- 手順 3 で停止した Instant Messaging サービスを再起動します。

```
InstantMessaging-base/sbin/imadmin start
```

## アップグレードの確認

Instant Messaging の Release 4 へのアップグレードは、アップグレードパッチが正常に適用されていることを確認することによって検証されます。この確認の手順は、300 ページの「アップグレード手順 (Solaris)」および 302 ページの「アップグレード手順 (Linux)」に記載されています。

次のコマンドを使用して、各種の Instant Messaging サブコンポーネントの状態を確認することもできます。

```
InstantMessaging-base/sbin/imadmin status
```

または、`iim.instancevardir/log` に格納されているログファイルを確認できます。

この `instancevardir` は、`iim.conf` ファイルで指定されています (パスについては、295 ページの表 16-2 を参照)。

## アップグレード後の作業

300 ページの「アップグレード手順 (Solaris)」および 302 ページの「アップグレード手順 (Linux)」で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Instant Messaging のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Instant Messaging の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。  
`su -`
2. Instant Messaging が稼働している場合は停止します。  
`InstantMessaging-base/sbin/imadmin stop`
3. [300 ページの表 16-4](#) のパッチを削除します。  
`patchrm patch_ID`
4. [手順 2](#) で停止した Instant Messaging サービスを再起動します。  
`InstantMessaging-base/sbin/imadmin start`

## 複数インスタンスのアップグレード

複数インスタンスのアップグレードは、Release 4 Instant Messaging には適用できません。



# Java ES Release 2 からの Instant Messaging のアップグレード

ここでは、Instant Messaging の Java ES 2004Q2 (Release 2) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。Release 2 Instant Messaging を Release 4 にアップグレードする手順は、Release 3 Instant Messaging からのアップグレードとはまったく異なります。

この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 2 の Instant Messaging のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 2 Instant Messaging を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード方法**：このアップグレードは、Instant Messaging ソフトウェアのアップグレードに必要なすべての操作を実行するアップグレードユーティリティを使用することにより実行されます。
- **アップグレードの依存関係**：コンピュータ上の任意の Java ES コンポーネントを Release 2 からアップグレードするには、そのコンピュータでホストされているほかのすべての Java ES コンポーネントをアップグレードする必要があります。Java ES コンポーネントの Release 2 から Release 4 への選択的なアップグレードはサポートされていません。特に、Instant Messaging、Web コンテナ、および Access Manager で使用されているすべての Java ES 共有コンポーネントを Release 4 にアップグレードする必要があります。
- **下位互換性**：Release 4 Instant Messaging は、Release 2 バージョンと下位互換性があります。
- **アップグレードのロールバック**：Release 4 アップグレードの Release 2 へのロールバックは、すべての Release 2 ソフトウェアおよびデータを保存し、手動で Release 2 バージョンに戻すことにより行われます。アップグレードをロールバックするためのユーティリティはありません。
- **プラットフォームの問題**：Instant Messaging をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通です。アップグレード処理には、プラットフォーム固有の詳細事項が含まれています。

## Release 2 の Instant Messaging のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Instant Messaging の Java ES Release 2 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- アップグレード前の作業
- [Release 2 の Instant Messaging のアップグレード \(Solaris\)](#)
- [Release 2 の Instant Messaging のアップグレード \(Linux\)](#)
- アップグレードの確認
- アップグレード後の作業
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Instant Messaging をアップグレードする前に、次に説明されている作業を実行してください。

#### *現在のバージョン情報の確認*

Instant Messaging の現在のバージョンは、標準のバージョン確認ユーティリティを使用することにより確認できます。

```
Solaris:  
pkginfo -l SUNWiimin
```

```
Linux:  
rpm -qa | grep sun-im
```

出力値については、[298 ページの表 16-3](#) を参照してください。

#### *Instant Messaging が依存するコンポーネントのアップグレード*

Java ES Release 4 では、1 つのコンピュータ上での Release 2 および Release 4 共有コンポーネントの共存はサポートされていません。

そのため、Instant Messaging が依存するローカルのすべての Java ES Release 2 コンポーネントを Release 4 にアップグレードする必要があります。コンピュータ上のすべての Instant Messaging の依存関係をアップグレードする場合は、Instant Messaging をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。

1. **共有コンポーネント**: Instant Messaging が依存するすべての共有コンポーネントを Release 4 にアップグレードする必要があります。1つのコンピュータ上でほかの Java ES 製品コンポーネントが Instant Messaging と共存している場合は、そのコンピュータ上に存在するすべての Java ES 共有コンポーネントをアップグレードする必要があります。Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Web コンテナソフトウェア**: Web Server または Application Server をアップグレードする手順は、それぞれ 150 ページの「Java ES Release 2 からの Web Server のアップグレード」および 189 ページの「Java ES Release 2 からの Application Server のアップグレード」で説明されています。
3. **Access Manager (Access Manager SDK)**: Access Manager を Release 4 にアップグレードする手順は、227 ページの「Java ES Release 2 からの Access Manager のアップグレード」で説明されています。
4. **Directory Server**: Instant Messaging がローカルの Directory Server に依存することはほとんどありませんが、Directory Server を Release 4 にアップグレードする手順は、125 ページの「Java ES Release 2 からの Directory Server と Administration Server のアップグレード」で説明されています。

### Instant Messaging のデータのバックアップ

Instant Messaging の Release 2 から Release 4 へのアップグレードでは、設定データとカスタマイズが変更されます。Instant Messaging をアップグレードする前に、Instant Messaging データベースおよび既存の任意のリソースや設定ファイルをバックアップすることをお勧めします。詳細については、『Sun Java System Instant Messaging 7 2005Q4 Administration Guide』(<http://docs.sun.com/doc/819-2503>) を参照してください。

### 必要な設定情報とパスワードの入手

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。Web Server を Web コンテナとして使用している場合は、設定情報は必要ありません。ただし、Application Server を Web コンテナとして使用している場合は、Application Server 管理者のユーザー ID とパスワードが必要になります。

## Release 2 の Instant Messaging のアップグレード (Solaris)

ここでは、Instant Messaging のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Instant Messaging ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- 各種の Instant Messaging サブコンポーネントが別のコンピュータに存在するアーキテクチャー、たとえば、あるコンピュータにメッセージリソースが、別のコンピュータに Instant Messaging サーバーが、さらに別のコンピュータに Instant Messaging マルチプレクサが存在するような場合は、このようなすべてのコンピュータでアップグレードを実行する必要があります。ただし、アップグレードは、同じインストールイメージを使用している 1 つのコンピュータ上の Instant Messaging の複数のサブコンポーネントに適用されます。
- Release 2 Instant Messaging ソフトウェアの Java ES Release 4 へのアップグレードでは、次の操作を実行するアップグレードユーティリティを使用します。
  - 作業ファイルを格納するための一時ディレクトリを作成します。
  - システムにインストールされているすべてのパッケージに対する以前のパッケージインストールパラメータを収集し、一時的に格納します。
  - 旧バージョンの Instant Messaging サーバーをシャットダウンします。
  - 新しいパッケージをインストールし、既存のパッケージにパッチを当てます。
  - Instant Messaging に必要な新しい共有コンポーネントパッケージをすべてインストールします。
  - 以前のグラフィックスコンテンツを IIM\_DOCROOT から保存し、それを新しいリソースファイルの場所に復元します。
  - すべてのサービスを再起動します。
  - 一時ディレクトリとその中身を削除します。

### アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Instant Messaging に適用されます。

1. root としてログインするか、スーパーユーザーになります。  
su -
2. IM-SDK 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。  
[306 ページの「Instant Messaging が依存するコンポーネントのアップグレード」](#)を参照してください。

3. Java ES ディストリビューションの Instant Messaging ツールディレクトリから upgrade ユーティリティを実行します。

```
cd /Solaris_arch/Product/instant_messaging/Tools
./upgrade
```

upgrade ユーティリティによって、アップグレード処理の進行状況を示すログファイルが作成されます。このログファイルは、次の場所に格納されています。

```
/var/sadm/install/logs/Instant_Messaging_Upgrade.timestamp
```

ここで、*timestamp* は、*yyyymmddhhss* の形式になります。

4. (省略可能) 必要に応じて、Release 2 のあとに導入された新機能を使用するように設定を変更します。設定情報については、『Sun Java System Instant Messaging 管理ガイド』(<http://docs.sun.com/doc/819-1487?l=ja>) を参照してください。
5. メッセージャリソースを再カスタマイズします。

メッセージャリソースをカスタマイズしている場合は、次のファイルにそのカスタマイズを再適用する必要があります。

```
InstantMessaging-base/html/Locale/imbrand.jar
InstantMessaging-base/html/Locale/imb[ssl].html|jnlp
```

307 ページの「Instant Messaging のデータのバックアップ」で保存したカスタマイズされたファイルを参照してください。

6. メッセージャリソースを Web コンテナに再配備します。

```
InstantMessaging-base/lib/deployHtml -f webcontainerDeployLocation
```

ここで、*webcontainerDeployLocation* は、メッセージャリソースを配備するディレクトリの場所です。この場所は一般に、使用されている Web コンテナによって異なります。次に例を示します。

Web Server

```
WebServer-base/docs/im
```

Application Server

```
AppServer8Config-base/nodeagents/hostname_domainName/instanceName/
docroot/iim
```

## Release 2 の Instant Messaging のアップグレード (Linux)

ここでは、Instant Messaging のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Instant Messaging ソフトウェアを Java ES Release 4 にアップグレードする場合は、Solaris プラットフォームのときと同じ注意点 (307 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Instant Messaging に適用されます。

1. `root` としてログインするか、スーパーユーザーになります。

```
su -
```

2. IM-SDK 共有コンポーネントの Release 4 へのアップグレードや、ほかの任意の共有コンポーネントのアップグレードをまだ実行していない場合は、実行します。

306 ページの「Instant Messaging が依存するコンポーネントのアップグレード」を参照してください。

3. Java ES ディストリビューションの Instant Messaging ツールディレクトリから upgrade ユーティリティを実行します。

```
cd /Linux_x86/Product/instant_messaging/Tools
./upgrade
```

upgrade ユーティリティによって、アップグレード処理の進行状況を示すログファイルが作成されます。このログファイルは、次の場所に格納されています。

```
/var/sadm/install/logs/Instant_Messaging_Upgrade.timestamp
```

ここで、*timestamp* は、*yyyymmddhhss* の形式になります。

4. (省略可能) 必要に応じて、Release 2 のあとに導入された新機能を使用するように設定を変更します。設定情報については、『Sun Java System Instant Messaging 管理ガイド』(<http://docs.sun.com/doc/819-1487?l=ja>)を参照してください。
5. メッセージャリソースを再カスタマイズします。

メッセージャリソースをカスタマイズしている場合は、次のファイルにそのカスタマイズを再適用する必要があります。

```
InstantMessaging-base/html/Locale/imbrand.jar
```

```
InstantMessaging-base/html/Locale/imb[ssl].html|jnlp
```

307 ページの「Instant Messaging のデータのバックアップ」で保存したカスタマイズされたファイルを参照してください。

6. メッセージャリソースを Web コンテナに再配備します。

```
InstantMessaging-base/lib/deployHtml -f webcontainerDeployLocation
```

ここで、*webcontainerDeployLocation* は、メッセージャリソースを配備するディレクトリの場所です。この場所は一般に、使用されている Web コンテナによって異なります。次に例を示します。

Web Server

```
WebServer-base/docs/im
```

Application Server

```
AppServer8Config-base/nodeagents/hostName_domainName_instanceName/
docroot/iim
```

## アップグレードの確認

次のコマンドを使用して、各種の Instant Messaging サブコンポーネントの状態を確認することができます。

```
InstantMessaging-base/sbin/imadmin status
```

または、`iim.instancevardir/log` に格納されているログファイルを確認できます。

この `instancevardir` は、`etc/InstantMessaging-base/config/iim.conf` ファイルで指定されています。

## アップグレード後の作業

308 ページの「アップグレード手順 (Solaris)」および 309 ページの「アップグレード手順 (Linux)」で説明している手順以外に、アップグレード後の作業はありません。ただし、Release 4 で Release 2 から拡張された機能を使用する場合は、Instant Messaging を再設定し、Web コンテナに再配備する必要があります。

## アップグレードのロールバック

Release 4 アップグレードの Release 2 へのロールバックは、すべての Release 2 ソフトウェアおよびデータを保存し (307 ページの「Instant Messaging のデータのバックアップ」を参照)、手動で Release 2 バージョンに戻すことにより行われます。アップグレードをロールバックするためのユーティリティーはありません。

## 複数インスタンスのアップグレード

複数インスタンスのアップグレードは、Release 4 Instant Messaging には適用できません。





# Portal Server

この章では、Portal Server を Java ES 2005Q4 (Release 4): Sun Java System Portal Server 6.3.1 2005Q4 にアップグレードする方法について説明します。ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [314 ページの「Portal Server のアップグレードの概要」](#)
- [317 ページの「Java ES Release 3 からの Portal Server のアップグレード」](#)
- [327 ページの「Java ES Release 2 からの Portal Server のアップグレード」](#)

---

## 注

この章で説明するファイルの場所は、*PortalServer-base* と *PortalServerConfig-base* という 2 つのディレクトリパスを基準に指定されています。これらのパスの少なくとも一部は、Portal Server が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*PortalServer-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWps
- Linux: /opt/sun/portal

*PortalServerConfig-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /etc/opt/SUNWps
  - Linux: /etc/opt/sun/portal
-

# Portal Server のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Portal Server の次の一般的な側面について説明します。

- [Java ES Release 4 の Portal Server について](#)
- [Portal Server のアップグレードロードマップ](#)
- [Portal Server のデータ](#)
- [互換性の問題](#)
- [Portal Server 依存関係](#)

## Java ES Release 4 の Portal Server について

Java ES Release 4 Portal Server は、機能的には Release 3 と同じですが、Release 3 以降のバグ修正が含まれています。

## Portal Server のアップグレードロードマップ

表 17-1 は、Portal Server の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 17-1 Java ES Release 4: Sun Java System Portal Server 6.3.1 2005Q4 へのアップグレードパス

Java ES のリリース	Portal Server バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Portal Server 6.3.1 2005Q1	直接アップグレード: パッチを適用することにより実行されます。いくつかの制限が適用されます (手順を参照)。	なし
Release 2	Sun Java System Portal Server 6.3 2004Q2	直接アップグレード: Release 4 にアップグレードするためのパッチを適用し、ソフトウェアを再設定して、Web コンテナに再配備することにより実行されます。	設定データ

表 17-1 Java ES Release 4: Sun Java System Portal Server 6.3.1 2005Q4 へのアップグレードパス ( 続き )

Java ES のリリース	Portal Server バージョン	一般的な手順	必要な再設定
Release 1	Sun ONE Portal Server 6.2 (2003Q4)	直接アップグレード不可: ただし、まず Release 3 にアップグレードし、次に Release 4 にアップグレードするためのパッチを適用することにより実行できます。いくつかの制限が適用されません ( 手順を参照 )。	設定データ
それ以前の Java ES リリース		直接アップグレード不可。	

## Portal Server のデータ

次の表は、Portal Server ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 17-2 Portal Server のデータの使用方法

データの種類	場所	使用法
設定データ	<i>PortalServerConfig-base/</i>	Portal Server の設定
Web コンテナの設定	<p>Web Server: 次の場所にある <code>server.policy</code> および <code>server.xml</code> ファイル <i>WebServer-base/https-hostname/config</i></p> <p>Application Server (Java ES Release 3 および 4): 次の場所にある <code>server.policy</code> および <code>domain.xml</code> ファイル <i>AppServer8Config-base/domains/domainName/config</i></p> <p>Application Server (Java ES Release 2): 次の場所にある <code>server.policy</code> および <code>server.xml</code> ファイル <i>AppServer7Config-base/domains/domainName/config</i></p>	Portal Server Web コンテナインスタンスの設定
カスタマイズデータ	<i>PortalServerConfig-base/desktop</i>	<p>カスタマイズしたモジュールの JAR ファイル</p> <p>カスタマイズされたサンプル Portal Server デスクトップ</p>

表 17-2 Portal Server のデータの使用方法 (続き)

データの種類	場所	使用方法
ディレクトリスキーマ サービスの設定 ユーザーデータ	Directory Server	Portal Server は、ポータルデスクトップなどのサービスの設定、およびディレクトリに格納されているユーザープロフィールデータに依存します。
動的なアプリケーションデータ	なし	Portal Server は、セッション状態などのアプリケーションデータを永続的に格納しません。

## 互換性の問題

Release 4 Portal Server では、インタフェースの変更は導入されていません。Portal Server コンポーネントは、モバイルアクセスコンポーネントを含め、以前のバージョンと下位互換性があります。

## Portal Server 依存関係

ほかの Java ES コンポーネントに対する Portal Server の依存関係が、Portal Server ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Portal Server のインタフェースまたは関数に変更された場合は、Portal Server が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Portal Server は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Portal Server は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Web コンテナ** : Portal Server は、Java ES Web Server または Java ES Application Server のどちらか (または、Weblogic や WebSphere のサードパーティー Web コンテナ) で提供される Web コンテナサービスに依存します。
- **Access Manager (または Access Manager SDK)** : Portal Server は、シングルサインオンを含む、エンドユーザーへの認証および承認サービスを提供するために Access Manager に依存します。Access Manager をリモートコンピュータ上で動作させる場合、Access Manager SDK がローカルで利用可能である必要があります。

- **Directory Server:** Portal Server は、Directory Server に格納されているユーザーデータにアクセスします。そのため、Portal Server のアップグレードによって、ディレクトリスキーマの拡張が必要になる場合があります。

## Java ES Release 3 からの Portal Server のアップグレード

ここでは、Portal Server の Java ES 2005Q1 (Release 3) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。

---

**注**      ここでの説明は、Portal Server が Application Server Web コンテナに配備されており、Release 4 にアップグレードされる前に Release 2 から Release 3 にアップグレードされているという特殊なケースには適用されません。上記のアップグレードパスは、現在サポートされていません。

---

この節で説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Portal Server のアップグレード](#)
- [複数インスタンスのアップグレード](#)

### はじめに

Java ES Release 3 Portal Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード方法:** このアップグレードは、Release 3 バージョンにパッチを適用し、Portal Server を Web コンテナに再配備することにより実行されます。
- **アップグレードの依存関係:** Portal Server は多くの Java ES 共有コンポーネントに対する依存関係を持ちますが ([42 ページの表 1-6](#) を参照)、Release 4 の Portal Server はこれらのコンポーネントの Release 3 バージョンと互換性があります。したがって、Mobile Access コア (MA コア) を除き、これらの共有コンポーネントのアップグレードは、Portal Server の Release 4 へのアップグレードに関しては省略可能です。

さらに、316 ページの「Portal Server 依存関係」で説明されているように、Release 4 Portal Server は Web コンテナ、Access Manager、および Directory Server に依存します。ただし、これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Portal Server の Release 4 へのアップグレードに関しては省略可能です。

- **下位互換性** : Release 4 Portal Server は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Portal Server の Release 4 アップグレードの Release 3 へのロールバックは、アップグレード中に適用されたパッチをロールバックすることにより行われます。パッチのロールバックは、Linux プラットフォームでは使用できません。
- **プラットフォームの問題** : Portal Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Portal Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Portal Server の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Portal Server のアップグレード \(Solaris\)](#)
- [Release 3 の Portal Server のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Portal Server をアップグレードする前に、次に説明されている作業を実行してください。

#### **現在のバージョン情報の確認**

Portal Server の現在のバージョンは、次のコマンドを使用することにより確認できません。

`PortalServer-base/bin/version`

表 17-3 Portal Server のバージョン確認の出力

Java ES のリリース	Portal Server のバージョン番号
Release 2	6.3
Release 3	6.3.1
Release 4	6.3.1 <sup>1</sup>

1. Release 3 と Release 4 の違いはパッチだけです。323 ページの表 17-5 および 337 ページの表 17-7 に示されている Release 4 のパッチは、Solaris の `showrev -p grep patch_ID` コマンドおよび Linux の `rpm -qa sun-portal-core` コマンドを使用して "25.12" 以上の文字列を探すことにより確認できます。

### Portal Server が依存するコンポーネントのアップグレード

一般には、コンピュータシステム上 ( および、コンピューティング環境内 ) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Portal Server は、Mobile Access コア (MA コア) 共有コンポーネントに対してのみ強いアップグレード依存関係を持っています。したがって、Portal Server が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

ただし、Portal Server のすべての依存関係をアップグレードすることを選択した場合は、Portal Server をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。
3. **Web コンテナソフトウェア** : Web Server または Application Server をアップグレードする手順は、それぞれ、141 ページの第 6 章「Web Server」および 177 ページの第 9 章「Application Server」で説明されています。

---

**注** WebLogic や WebSphere などのサードパーティー Web コンテナをアップグレードすると、Portal Server をサポートするためにこれらのコンテナに加えられたカスタマイズがコンテナのアップグレードによって上書きされるため、Portal Server が動作しなくなる場合があります。

この場合は、アップグレードされた Web コンテナ環境で Portal Server を再インストールおよび再設定する必要があります。

---

4. **Access Manager (Access Manager SDK):** Access Manager を Release 4 にアップグレードする手順は、[205 ページの第 11 章「Access Manager」](#) で説明されています。

### **Release 3 Portal Server の設定情報のバックアップ**

Portal Server の Release 4 へのアップグレードでは、Portal Server ソフトウェアの再設定は必要ありません。ただし、安全策として、設定情報が格納されている次のディレクトリをバックアップすることができます。

`PortalServerConfig-base/`

### **必要な設定情報とパスワードの入手**

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。Web Server を Web コンテナとして使用している場合は、設定情報は必要ありません。ただし、Application Server を Web コンテナとして使用している場合は、Application Server 管理者のユーザー ID とパスワードが必要になります。

## **Release 3 の Portal Server のアップグレード (Solaris)**

ここでは、Portal Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### **アップグレードの注意点 (Solaris)**

Portal Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- 同じ Portal Server インストールイメージに対応するすべての Portal Server インスタンスが同時にアップグレードされます。インストールイメージにパッチを適用している間は、Web コンテナを停止することにより、このようなインスタンスをすべて停止するようにしてください。
- Solaris OS での Release 4 Portal Server のアップグレードパッチを次の表に示します。



表 17-4 Solaris で Portal Server をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC	x86
	Solaris 8、9、10	Solaris 9、10
Portal Server コア	118950-12	118951-12
Portal Server ローカライズ版	119425-08	119425-08
Portal Server ローカリゼーション コンフィギュレータ	118115-11	118115-11

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server に適用されます。

1. 表 17-4 に基づいて、必要なパッチを入手します。

特定のリビジョンを使用するように指示された場合以外は、常に、使用可能な最新のパッチリビジョンを使用してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web コンテナを停止することにより、Portal Server を停止します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin stop-domain domainName
```

4. MA コア共有コンポーネントやその他の任意のコンポーネントのアップグレードをまだ実行していない場合は、実行します。

319 ページの「Portal Server が依存するコンポーネントのアップグレード」を参照してください。

5. 表 17-4 の適切な Portal Server のパッチを適用します。

この 2 つの Portal Server ローカライズ版パッチを適用する前に、Portal Server コアパッチを適用するようにしてください。

```
patchadd patch_ID
```

- パッチのアップグレードが成功したことを確認します。

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

この出力では、[手順 5](#) で適用したパッチ ID のバージョンが返されるはずですが。

- Web コンテナを再起動することにより、Portal Server を再起動します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

- Portal Server Web アプリケーションを Web コンテナに再配備します。

```
PortalServer-base/bin/deploy redeploy
```

redeploy コマンドによって、コンテンツが *PortalServer-base/web-src* から */var/PortalServer-base/https-hostName/deploy-dir/web-apps* に再配備されます。そのため、Portal Server Web アプリケーションをカスタマイズする場合は、最初に */web-src* に対して行い、次に */web-apps* に配備するようにしてください。*/web-apps* の下に変更を加えた場合は、deploy コマンドを実行する前に */web-src* にレプリケートするようにしてください。そうしないと、これらの変更が上書きされます。

- Web コンテナを停止し、再起動します。

すべての状況で必要なわけではありませんが、Web コンテナを再起動すると、Portal Server がクリーンな状態で起動されるようになります。

## Release 3 の Portal Server のアップグレード (Linux)

ここでは、Portal Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Portal Server ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 ([320 ページの「アップグレードの注意点 \(Solaris\)」](#) を参照) を考慮に入れてください。

Linux OS での Release 4 Portal Server のアップグレードパッチを次の表に示します。

表 17-5 Linux で Portal Server をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Portal Server コア	118952-12 <ul style="list-style-type: none"> <li>sun-portal-core-6.3-25.12.i386.rpm</li> </ul> および、ポータルデスクトップと Portal Server モバイルアクセスのための他のいくつかの RPM。
Portal Server ローカライズ版	119426-07 <ul style="list-style-type: none"> <li>sun-portal-core-Locale-6.3-24.i386.rpm</li> </ul> および、Portal Server モバイルアクセス、設定、アイデンティティ、その他のコンポーネントのための他のいくつかの RPM。
Portal Server ローカリゼーションコンフィ ギュレータ	118116-08 <ul style="list-style-type: none"> <li>sun-portal-110n-configurator-6.3-24.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server に適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 17-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web コンテナを停止することにより、Portal Server を停止します。

Web Server:

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

**Application Server:**

*AppServer8-base/bin/asadmin stop-domain domainName*

4. MA コア共有コンポーネントやその他の任意のコンポーネントのアップグレードをまだ実行していない場合は、実行します。

[319 ページの「Portal Server が依存するコンポーネントのアップグレード」](#)を参照してください。

5. [表 17-5](#) の Portal Server コアパッチの RPM を適用します。

```
cd /tmp
```

ここで、/tmp は、[手順 1](#) でパッチをダウンロードしたディレクトリです。

```
./update
```

update スクリプトにより RPM がインストールされます。また、パッチの結果として正しい設定変更が行われることも保証されます。

6. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-portal-core-6.3-25
```

RPM のアップグレードドリビジョン番号が返されることを確認します。

7. [表 17-5](#) の 2 つの Portal Server ローカライズ版パッチの RPM を適用します。

```
rpm -Fvh --replacefiles sun-portal-*-Locale-6.3-24.i386.rpm
```

```
rpm -Fvh --replacefiles
```

```
sun-portal-l10n-configurator-6.3-24.i386.rpm
```

8. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-portal-l10n-configurator-6.3-24
```

RPM のアップグレードドリビジョン番号が返されることを確認します。

9. Web コンテナを再起動することにより、Portal Server を再起動します。

**Web Server:**

*WebServer-base/https-instanceName/start*

**Application Server:**

*AppServer8-base/bin/asadmin start-domain --user admin\_ID*

*--password password domainName*

10. Portal Server Web アプリケーションを Web コンテナに再配備します。

*PortalServer-base/bin/deploy redeploy*

redeploy コマンドによって、コンテンツが *PortalServer-base/web-src* から */var/PortalServer-base/https-hostName/deploy-dir/web-apps* に再配備されます。そのため、Portal Server Web アプリケーションをカスタマイズする場合は、最初に */web-src* に対して行い、次に */web-apps* に配備するようにしてください。*/web-apps* の下に変更を加えた場合は、deploy コマンドを実行する前に */web-src* にレプリケートするようにしてください。そうしないと、これらの変更が上書きされます。

#### 11. Web コンテナを停止し、再起動します。

すべての状況で必要なわけではありませんが、Web コンテナを再起動すると、Portal Server がクリーンな状態で起動されるようになります。

## アップグレードの確認

Portal Server の Release 4 へのアップグレードは、アップグレードパッチが正常に適用されていることを確認することによって検証されます。この確認の手順は、[321 ページの「アップグレード手順 \(Solaris\)」](#) および [323 ページの「アップグレード手順 \(Linux\)」](#) に記載されています。

さらに、次のコマンドを使用することもできます。

```
PortalServer-base/bin/version
```

出力値については、[319 ページの表 17-3](#) を参照してください。

パッチのアップグレードに対するこれらのテスト以外に、前に動作していた機能が引き続き正常に動作すること、および目的のバグ修正が実際に修正されていることを確認することができます。

## アップグレード後の作業

[321 ページの「アップグレード手順 \(Solaris\)」](#) および [323 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Portal Server のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Portal Server の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。再設定がロールバックされ、パッチが削除されます。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

su -

2. Web コンテナを停止することにより、Portal Server を停止します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

3. [321 ページの表 17-4](#) のパッチを削除します。

```
patchrm patch_ID
```

4. Web コンテナを再起動することにより、Portal Server を再起動します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName  
--user admin_ID --password password
```

5. Portal Server Web アプリケーションを Web コンテナに再配備します。

```
PortalServer-base/bin/deploy redeploy
```

redeploy コマンドによって、コンテンツが *PortalServer-base/web-src* から */var/PortalServer-base/https-hostName/deploy-dir/web-apps* に再配備されます。そのため、Portal Server Web アプリケーションをカスタマイズする場合は、最初に */web-src* に対して行い、次に */web-apps* に配備するようにしてください。*/web-apps* の下に変更を加えた場合は、deploy コマンドを実行する前に */web-src* にレプリケートするようにしてください。そうしないと、これらの変更が上書きされます。

6. Web コンテナを停止し、再起動します。

すべての状況で必要なわけではありませんが、Web コンテナを再起動すると、Portal Server がクリーンな状態で起動されるようになります。

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、スケーラビリティを実現し、可用性を向上させるために、Portal Server が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Portal Server コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。

Portal Server のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Portal Server インスタンスをアップグレードする順次アップグレードを実行できます。Portal Server の各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。[318 ページの「Release 3 の Portal Server のアップグレード」](#)の説明に従って、各インスタンスのアップグレードを実行します。

# Java ES Release 2 からの Portal Server のアップグレード

ここでは、Portal Server の Java ES 2004Q2 (Release 2) から Java ES 2005Q4 (Release 4) へのアップグレードについて説明します。

Release 2 Portal Server のアップグレードは複雑であり、また停止時間がかなり長くなる可能性があるため、本稼働システムを直接アップグレードするのではなく、別のコンピュータで並列アップグレードを実行することも選択できます。このような方法は、限られた停止時間しか許容されないビジネスクリティカルな、または複雑な Portal Server ソリューションに対してお勧めします。アップグレード処理の所要時間はまた、必要な Portal Server カスタマイズを再実装したりテストしたりするためにかかる時間によっても異なります。

また、ここで説明する手順を、特定のアップグレードシナリオに合うように変更したり適応させたりすることも必要になる可能性があります。このような場合は、ご購入先のサポートサービスに連絡して、アップグレードの実行について支援を受けることをお勧めします。

ここでは、Release 2 から Release 4 へのアップグレードに関して次の内容を説明します。

- [はじめに](#)
- [Release 2 の Portal Server のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 2 Portal Server を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順**：このアップグレードは、Release 2 バージョンにパッチを適用することにより実行されます。また、Portal Server の再設定も必要であり、アップグレードユーティリティを使用して実行されます。
- **アップグレードの依存関係**：Portal Server は、いくつかの Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。Java ES では 1 つのコンピュータ上での Release 2 および Release 4 コンポーネントの混在がサポートされていないため、これらの共有コンポーネントをすべて Release 4 にアップグレードする必要があります。

さらに、316 ページの「Portal Server 依存関係」で説明されているように、Release 4 Portal Server は Web コンテナ、Access Manager、および Directory Server に依存します。Portal Server は、ローカルに存在する Web コンテナと Access Manager (または Access Manager SDK) に対しては強いアップグレード依存関係を持っており、ローカルに存在することがほとんどない Directory Server に対しては弱いアップグレード依存関係を持っています。

- **下位互換性**：Release 4 Portal Server は、Release 2 バージョンと下位互換性があります。
- **アップグレードのロールバック**：Portal Server の再設定が実行されたあとは、Portal Server の Release 4 アップグレードを Release 2 にロールバックすることはできません。
- **プラットフォームの問題**：Portal Server をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 2 の Portal Server のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Portal Server の Java ES Release 2 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Portal Server のアップグレード \(Solaris\)](#)
- [Release 3 の Portal Server のアップグレード \(Linux\)](#)



- アップグレードの確認
- アップグレード後の作業
- アップグレードのロールバック (Solaris)

## アップグレード前の作業

Portal Server をアップグレードする前に、次に説明されている作業を実行してください。

### 現在のバージョン情報の確認

Portal Server の現在のバージョンは、次のコマンドを使用することにより確認できます。

```
PortalServer-base/bin/version
```

出力値については、[319 ページの表 17-3](#) を参照してください。

### Portal Server が依存するコンポーネントのアップグレード

Java ES Release 4 では、1 つのコンピュータ上での Release 2 および Release 4 共有コンポーネントの共存はサポートされていません。

そのため、Portal Server が依存するローカルのすべての Java ES Release 2 コンポーネントを Release 4 にアップグレードする必要があります。コンピュータ上のローカルのすべての Portal Server の依存関係をアップグレードする場合は、Portal Server をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。特定のアップグレードシナリオに対して特殊な要件が存在することに注意してください。

1. **共有コンポーネント** : Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、[55 ページの「Java ES 共有コンポーネントのアップグレード」](#) で説明されています。
2. **Directory Server** : Portal Server が、ローカルの Directory Server に依存することはほとんどありません。ただし、Directory Server を Release 4 にアップグレードする手順は、[105 ページの第 4 章「Directory Server と Administration Server」](#) で説明されています。
3. **Web コンテナソフトウェア** : Portal Server は、Web Server または Application Server のどちらかで提供される Web コンテナで稼働しています。
  - **Web Server** : [150 ページの「Java ES Release 2 からの Web Server のアップグレード」](#) の手順を使用して、Release 4 Web Server にアップグレードします。

- Application Server: 189 ページの「[Java ES Release 2 からの Application Server のアップグレード](#)」の手順を使用するのではなく、Java ES インストーラを使用して Application Server のフレッシュインストールを実行することにより、Release 4 Application Server にアップグレードします。Release 4 Application Server 8 をインストールする前に、Release 2 Application Server 7 から管理ポートとサーバーインスタンスポートを入手することを忘れないでください。

---

**注**

WebLogic や WebSphere などのサードパーティー Web コンテナをアップグレードすると、Portal Server をサポートするためにこれらのコンテナに加えられたカスタマイズがコンテナのアップグレードによって上書きされるため、Portal Server が動作しなくなる場合があります。

この場合は、アップグレードされた Web コンテナ環境で Portal Server を再インストールおよび再設定する必要があります。

---

4. **Access Manager (Access Manager SDK):** Portal Server は、Access Manager と同じ Web コンテナで稼働している場合と、別の Web コンテナで稼働している場合があります。
  - Portal Server が Access Manager とは別の Web コンテナで稼働している場合、たとえば Access Manager がリモートで稼働している場合は、[227 ページの「Java ES Release 2 からの Access Manager のアップグレード」](#)の手順を使用して Access Manager または Access Manager SDK を Release 2 から Release 4 にアップグレードします。Access Manager SDK のみをアップグレードする場合は、[226 ページの「Release 3 の Access Manager SDK のアップグレード」](#)を参照し、DEPLOY\_LEVEL = 3 を設定します。
  - Portal Server が Access Manager と同じ Web コンテナで稼働しており、その Web コンテナが Web Server で提供されている場合は、[228 ページの「Release 2 Access Manager のアップグレード: Web Server Web コンテナ」](#)の手順を使用して Access Manager を Release 2 から Release 4 にアップグレードします。
  - Portal Server が Access Manager と同じ Web コンテナで稼働しており、その Web コンテナが Application Server で提供されている場合は、[228 ページの「Release 2 Access Manager のアップグレード: Application Server Web コンテナ」](#)の手順を使用して Access Manager を Release 2 から Release 4 にアップグレードしますが、AS がフレッシュインストールされたシナリオを使用するようにしてください。

### **Release 2 Portal Server の設定情報のバックアップ**

Portal Server の Release 4 へのアップグレードでは、Portal Server ソフトウェアの再設定が必要です。安全策として、設定情報が格納されている次のディレクトリをバックアップすることができます。

*PortalServerConfig-base/*

## 必要な設定情報とパスワードの入手

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。Web Server を Web コンテナとして使用している場合は、管理者パスワードは必要ありません。ただし、Application Server を Web コンテナとして使用している場合は、Application Server 管理者のユーザー ID とパスワードが必要になります。

## Release 2 Portal Server のアップグレード (Solaris)

ここでは、Portal Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Portal Server ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- 同じ Portal Server インストールイメージに対応するすべての Portal Server インスタンスが同時にアップグレードされます。インストールイメージにパッチを適用している間は、Web コンテナを停止することにより、このようなインスタンスをすべて停止するようにしてください。
- Solaris OS での Release 4 Portal Server のアップグレードパッチを次の表に示します。

表 17-6 Solaris で Portal Server を Release 4 にアップグレードするためのパッチ

説明	SPARC Solaris 8、9	x86 Solaris 9	Solaris 10
Portal Server sync-up	118195-07	118196-07	適用なし
Portal Server コア	118128-13	118129-13	適用なし
Mobile Access コア	119527-02	119528-02	適用なし
Portal Server の修正	118950-15 (以上)	118951-15 (以上)	適用なし

- Solaris プラットフォームで Portal Server をアップグレードする手順は、Portal Server が Web Server または Application Server のどちらで提供される Web コンテナに配備されているかによって異なります。そのため、以降の説明では、この 2 つの Web コンテナに対して別のアップグレード手順を示します。

### アップグレード手順 (Solaris: Web Server)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server に適用されます。

1. 表 17-6 に基づいて、必要なパッチを入手します。

表 17-6 に示されている正確なパッチリビジョンをダウンロードするようにしてください。ただし、あとでパッチを使用できる可能性のある Portal Server の修正は除きます。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web コンテナを停止することにより、Portal Server を停止します。

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

4. すべての共有コンポーネント、Web コンテナ、および Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

329 ページの「Portal Server が依存するコンポーネントのアップグレード」を参照してください。

5. まだ稼働していない場合は、Directory Server と Access Manager を起動します。

6. 表 17-6 の適切な Portal Server のパッチを適用します。

パッチは、表 17-6 に示されている上から下の順序で適用するようにしてください。

```
patchadd patch_ID
```

7. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 7 で適用したパッチ ID のバージョンが返されるはずです。

8. Portal Server ソフトウェアを再設定します。

```
ksh
```

```
$ cd PortalServer-base/lib
```

```
$ ./upgradePS04Q205Q1
```

9. Web コンテナを再起動することにより、Portal Server を再起動します。

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

10. Portal Server Web アプリケーションを Web コンテナに再配備します。

```
PortalServer-base/bin/deploy redeploy
```

redeploy コマンドによって、コンテンツが *PortalServer-base/web-src* から */var/PortalServer-base/https-hostName/deploy-dir/web-apps* に再配備されます。そのため、Portal Server Web アプリケーションをカスタマイズする場合は、最初に */web-src* に対して行い、次に */web-apps* に配備するようにしてください。*/web-apps* の下に変更を加えた場合は、deploy コマンドを実行する前に */web-src* にレプリケートするようにしてください。そうしないと、これらの変更が上書きされます。

#### 11. Web コンテナを停止し、再起動します。

すべての状況で必要なわけではありませんが、Web コンテナを再起動すると、Portal Server がクリーンな状態で起動されるようになります。

### アップグレード手順 (Solaris: Application Server)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server に適用されます。

#### 1. 表 17-6 に基づいて、必要なパッチを入手します。

表 17-6 に示されている正確なパッチリビジョンをダウンロードするようにしてください。ただし、あとでパッチを使用できる可能性のある Portal Server の修正は除きます。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

#### 2. root としてログインするか、スーパーユーザーになります。

```
su -
```

#### 3. Portal Server がすでに Release 2 Application Server インスタンスでは稼働していないことを確認します。

```
AppServerConfig7-base/domains/domainName/instanceName/bin/stopserv
```

上のコマンドおよび以降の手順では、次の規則が使用されています。

- デフォルトの *domainName* は domain1 です。
- デフォルトの *instanceName* は server1 です。

#### 4. すべての共有コンポーネント、Web コンテナ、および Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

329 ページの「Portal Server が依存するコンポーネントのアップグレード」を参照してください。

#### 5. アップグレードされた Access Manager が Release 4 Application Server インスタンスでは稼働していないことを確認します。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

## 6. 次の Access Manager 設定ファイル

*AccessManagerConfig-base*/config/AMConfig.properties

に、次の属性値が含まれていることを確認します。

```
com.ipplanet.am.notification.url=
    http://hostName:port/amserver/notificationservice
com.sun.identity.webcontainer=IAS8.1
com.ipplanet.am.cookie.encode=true
```

ここで、*hostName:port* は、Access Manager インスタンスをホストしているコンピュータとポートです。

## 7. 表 17-6 の適切な Portal Server パッチを適用します。

パッチは、表 17-6 に示されている上から下の順序で適用するようにしてください。

```
patchadd patch_ID
```

## 8. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 7 で適用したパッチ ID のバージョンが返されるはずですが。

## 9. 次の Portal Server 設定ファイル

*PortalServerConfig-base*/PSConfig.properties

に、次の属性値が含まれていることを確認します。これらの属性値は、Application Server のドメイン管理サーバー (DAS) インスタンスを参照していません。

```
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 (for example, default=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 (for example, default=80)
DEPLOY_DIR=AppServer8-base
PS_PROTOCOL=http
```

Release 4 Application Server 8 フレッシュインストールに割り当てられたポート値が Release 2 Application Server 7 インストールのポート値と同じであり、それらがデフォルトのポート値であると仮定すると、上に示したデフォルト値が適用されます。

## 10. PSconfig.properties ファイルを次のように変更します。

DEPLOY\_INSTANCE=*temporary\_instanceName*

ここで、*temporary\_instanceName* は、使用されていない一時的な値です。

11. DAS が稼働していることを確認します。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

12. 次のコマンドを実行します。

```
cd PortalServer-base/bin./multiserverinstance
```

multiserverinstance スクリプトから尋ねられるいくつかの質問では、[334 ページの手順 9](#) に示されている PSConfig.properties ファイル内の設定値がデフォルトとして使用され、次の手順ではこれらのデフォルト値が正しいと仮定しています。

multiserverinstance スクリプトから尋ねられる質問に対して、次のように答えます。

1. 新しい portalserver インスタンスの作成 に対しては、オプション 1 を選択します。
  2. Sun Java System Application Server 8.1 に対しては、オプション 3 を選択します。
  3. コンテナがインストールされている場所。Return キーを押します。
  4. ドメイン名。Return キーを押します。
  5. ドメイン (DAS) のパス。手順 4 の質問でデフォルト値として示されていたものと同じ値を入力します。
  6. Web コンテナインスタンスのパス。手順 5 で入力したものと同じ値を入力します。
  7. Web コンテナの管理者。Return キーを押します。
  8. Web コンテナの管理ポート。Return キーを押します。
  9. Web コンテナの管理ポートはセキュリティー保護されているか。Return キーを押します。
  10. インスタンス名。server の値を入力します。
  11. インスタンスポート。PSConfig.properties ファイル内の PS\_PORT 値に対して入力されているものと同じ値を入力します。
  12. インスタンスポートはセキュリティー保護されているか。Return キーを押します。
  13. Web コンテナのドキュメントルートディレクトリ。Return キーを押します。
  14. Application Server の管理パスワード。パスワードを入力します。
  15. Identity Server の管理パスワード。パスワードを入力します。
13. PSconfig.properties ファイルを次のように変更します。

```
DEPLOY_INSTANCE=server
```

ここで、server の値は、DAS インスタンスのデフォルトのインスタンス名です。

14. DAS を再起動します。

```
AppServer8-base/bin/asadmin stop-domain domainName
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

15. Portal Server Web アプリケーションを配備します。

```
cd PortalServer-base/bin
./deploy redeploy
```

deploy.log にエラーが存在する可能性があることを示すメッセージは無視してください。

redeploy コマンドによって、コンテンツが *PortalServer-base*/web-src から */var/PortalServer-base/https-hostName/deploy-dir/web-apps* に再配備されます。そのため、Portal Server Web アプリケーションをカスタマイズする場合は、最初に */web-src* に対して行い、次に */web-apps* に配備するようにしてください。*/web-apps* の下に変更を加えた場合は、deploy コマンドを実行する前に */web-src* にレプリケートするようにしてください。そうしないと、これらの変更が上書きされます。

16. Portal Server ソフトウェアを再設定します。

```
ksh
$ cd PortalServer-base/lib
$ ./postinstall_PortletSamples
$ ./upgradePS04Q205Q1
```

CLI137 関連のエラー、および upgradePS04Q205Q1 スクリプトによって発行される配備 (解除) 関連のエラーは無視してください。

17. DAS を再起動します。

```
AppServer8-base/bin/asadmin stop-domain domainName
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

すべての状況で必要なわけではありませんが、Web コンテナを再起動すると、Portal Server がクリーンな状態で起動されるようになります。

18. Portal Server のディスプレイプロファイルを更新します。

- a. 次のコマンドを実行します。

```
PortalServer-base/bin/dpadmin list -g -u amadminDN
-w amadminPassword /tmp/GlobalDP.xml
```

ここで、*amadminDN* の値は、*AccessManagerConfig-base/config/AMConfig.properties* ファイル内のプロパティ `com.sun.identity.authentication.super.user` に含まれています。



b. 編集のために、ファイル /tmp/GlobalDP.xml を開きます。

c. 次の値

```
org.apache.xalan.xsltc.trax.TransformerFactoryImpl
を次のように変更します。
com.sun.org.apache.xalan.internal.xsltc.trax.
    TransformerFactoryImpl
```

d. 次の値

```
Sun Java™ System Portal Server 6 2004Q2
をすべて次のように変更します。
Sun Java™ System Portal Server 6 2005Q4
```

e. 次のコマンドを実行します。

```
PortalServer-base/bin/dpadmin modify -g -u amadminDN
-w amadminPassword /tmp/GlobalDP.xml
```

ここで、*amadminDN* の値は、手順 a で入力したものと同じです。

## Release 2 Portal Server のアップグレード (Linux)

ここでは、Portal Server のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Portal Server ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (320 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Portal Server のアップグレードパッチを次の表に示します。

表 17-7 Linux で Portal Server を Release 4 にアップグレードするためのパッチ

説明	パッチ ID と RPM 名
Portal Server コア	118020-16  sun-portal-module-6.3-25.i386.rpm  ここで、 <i>module</i> は、約 70 種類あるソフトウェアモジュールのいずれかです
Mobile Access コア	119529-02  • sun-mobileaccess-1.0-25.2.i386.rpm  • sun-mobileaccess-config-1.0-25.2.i386.rpm

表 17-7 Linux で Portal Server を Release 4 にアップグレードするためのパッチ ( 続き )

説明	パッチ ID と RPM 名
Portal Server の修正	118952-15 ( 以上 ) <ul style="list-style-type: none"> <li>• sun-portal-core-6.3-xx.y.i386.rpm</li> <li>• sun-portal-configurator-6.3-xx.y.i386.rpm</li> <li>• sun-portal-mobileaccess-6.3-xx.y.i386.rpm</li> <li>• sun-portal-desktop-6.3-xx.y.i386.rpm</li> <li>• sun-portal-sample-6.3-xx.y.i386.rpm</li> <li>• sun-portal-mobileaccess-config-6.3-xx.y.i386.rpm</li> </ul>

### アップグレード手順 (Linux: Web Server)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server に適用されます。

1. 表 17-7 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Web コンテナを停止することにより、Portal Server を停止します。

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

4. すべての共有コンポーネント、Web コンテナ、および Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

[329 ページ](#)の「Portal Server が依存するコンポーネントのアップグレード」を参照してください。

5. まだ稼働していない場合は、Directory Server と Access Manager を起動します。
6. 表 17-7 の Portal Server の RPM を適用します。

- a. cd /tmp

ここで、/tmp は、手順 1 でパッチをダウンロードしたディレクトリです。

- b. 118020 パッチファイルを解凍し、README ファイルを参照して、次のスクリプトを実行します。

```
./upgradeportalrpms
```

upgradeportalrpms スクリプトにより RPM がインストールされます。また、パッチの結果として正しい設定変更が行われることも保証されます。

- c. 119529 パッチファイルを解凍し、パッチを解凍したときに作成されたディレクトリから ./update スクリプトを実行します。
- d. 118952 パッチファイルを解凍し、パッチを解凍したときに作成されたディレクトリから ./update スクリプトを実行します。

7. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-portal
rpm -qa | grep sun-mobileaccess
```

RPM の新しいバージョン番号が返されることを確認します。

8. Portal Server ソフトウェアを再設定します。

```
ksh
$ cd PortalServer-base/lib
$ ./upgradePS04Q205Q1
```

9. PortalServer-base/export/deploy.import ファイルを次のように編集します。

次の行が存在した場合は、

```
%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
```

次のように置き換えます。

```
/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
```

つまり、%JATO\_LIB\_DIR% を /usr/share/lib/jato に置き換えます。

10. Web コンテナを再起動することにより、Portal Server を再起動します。

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

11. Portal Server Web アプリケーションを Web コンテナに再配備します。

```
PortalServer-base/bin/deploy redeploy
```

redeploy コマンドによって、コンテンツが *PortalServer-base/web-src* から */var/PortalServer-base/https-hostName/deploy-dir/web-apps* に再配備されます。そのため、Portal Server Web アプリケーションをカスタマイズする場合は、最初に */web-src* に対して行い、次に */web-apps* に配備するようにしてください。*/web-apps* の下に変更を加えた場合は、deploy コマンドを実行する前に */web-src* にレプリケートするようにしてください。そうしないと、これらの変更が上書きされます。

## 12. Web コンテナを停止し、再起動します。

すべての状況で必要なわけではありませんが、Web コンテナを再起動すると、Portal Server がクリーンな状態で起動されるようになります。

### アップグレード手順 (Linux: Application Server)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server に適用されます。

#### 1. 表 17-7 に基づいて、必要なパッチを入手します。

表 17-7 に示されている正確なパッチリビジョンをダウンロードするようにしてください。ただし、あとでパッチを使用できる可能性のある Portal Server の修正は除きます。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から */tmp* にダウンロードできます。

#### 2. root としてログインするか、スーパーユーザーになります。

```
su -
```

#### 3. Portal Server がすでに Release 2 Application Server インスタンスでは稼働していないことを確認します。

```
AppServerConfig7-base/domains/domainName/instanceName/bin/stopserv
```

上のコマンドおよび以降の手順では、次の規則が使用されています。

- デフォルトの *domainName* は *domain1* です。
- デフォルトの *instanceName* は *server1* です。

#### 4. すべての共有コンポーネント、Web コンテナ、および Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

[329 ページの「Portal Server が依存するコンポーネントのアップグレード」](#) を参照してください。

#### 5. アップグレードされた Access Manager が Release 4 Application Server インスタンスでは稼働していないことを確認します。

```
AppServer8-base/bin/asadmin stop-domain domainName
```

## 6. 次の Access Manager 設定ファイル

*AccessManagerConfig-base*/config/AMConfig.properties

に、次の属性値が含まれていることを確認します。

```
com.iplanet.am.notification.url=
    http://hostName:port/amserver/notificationservice
com.sun.identity.webcontainer=IAS8.1
com.iplanet.am.cookie.encode=true
```

ここで、*hostName:port* は、Access Manager インスタンスをホストしているコンピュータとポートです。

## 7. 表 17-7 の Portal Server の RPM を適用します。

a. `cd /tmp`

ここで、/tmp は、手順 1 でパッチをダウンロードしたディレクトリです。

## b. 118020 パッチファイルを解凍し、README ファイルを参照して、次のスクリプトを実行します。

```
./upgradeportalrpms
```

upgradeportalrpms スクリプトにより RPM がインストールされます。また、パッチの結果として正しい設定変更が行われることも保証されます。

## c. 119529 パッチファイルを解凍し、パッチを解凍したときに作成されたディレクトリから ./update スクリプトを実行します。

## d. 118952 パッチファイルを解凍し、パッチを解凍したときに作成されたディレクトリから ./update スクリプトを実行します。

## 8. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-portal
rpm -qa | grep sun-mobileaccess
```

RPM の新しいバージョン番号が返されることを確認します。

9. *PortalServer-base*/export/deploy.import ファイルを次のように編集します。

次の行が存在した場合は、

```
%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
```

次のように置き換えます。

```
/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
```

つまり、%JATO\_LIB\_DIR% を /usr/share/lib/jato に置き換えます。

10. 「アップグレード手順 (Solaris: Application Server)」の 334 ページの手順 9 から 336 ページの手順 18 に従います。

## アップグレードの確認

Portal Server の Release 4 へのアップグレードは、アップグレードパッチが正常に適用されていることを確認することによって検証されます。この確認の手順は、[321 ページの「アップグレード手順 \(Solaris\)」](#) および [323 ページの「アップグレード手順 \(Linux\)」](#) に記載されています。

さらに、次のコマンドを使用することもできます。

```
PortalServer-base/bin/version
```

出力値については、[319 ページの表 17-3](#) を参照してください。

パッチのアップグレードに対するこれらのテスト以外に、前に動作していた機能が引き続き正常に動作すること、および目的のバグ修正が実際に修正されていることを確認することができます。

## アップグレード後の作業

[333 ページの「アップグレード手順 \(Solaris: Application Server\)」](#) および [338 ページの「アップグレード手順 \(Linux: Web Server\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック

Portal Server の Release 2 から Release 4 へのアップグレードはロールバックできません。

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、スケーラビリティを実現し、可用性を向上させるために、Portal Server が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Portal Server コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。

Portal Server のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Portal Server インスタンスをアップグレードする順次アップグレードを実行できます。Portal Server の各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。[328 ページの「Release 2 の Portal Server のアップグレード」](#) の説明に従って、各インスタンスのアップグレードを実行します。

# Portal Server Secure Remote Access

この章では、Portal Server Secure Remote Access を Java ES 2005Q4 (Release 4): Sun Java System Portal Server Secure Remote Access 6 2005Q4 にアップグレードする方法について説明します。

ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [344 ページの「Portal Server Secure Remote Access のアップグレードの概要」](#)
- [347 ページの「Java ES Release 3 からの Portal Server Secure Remote Access のアップグレード」](#)
- [356 ページの「Java ES Release 2 からの Portal Server Secure Remote Access のアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*PortalServer-base* と *PortalServerConfig-base* という 2 つのディレクトリパスを基準に指定されています。これらのパスの少なくとも一部は、Portal Server が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*PortalServer-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /opt/SUNWps
- Linux: /opt/sun/portal

*PortalServerConfig-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- Solaris: /etc/opt/SUNWps
  - Linux: /etc/opt/sun/portal
-

# Portal Server Secure Remote Access のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Portal Server Secure Remote Access の次の一般的な側面について説明します。

- [Java ES Release 4 の Portal Server Secure Remote Access について](#)
- [Portal Server Secure Remote Access のアップグレードロードマップ](#)
- [Portal Server Secure Remote Access のデータ](#)
- [互換性の問題](#)
- [Portal Server Secure Remote Access 依存関係](#)

## Java ES Release 4 の Portal Server Secure Remote Access について

Java ES Release 4 Portal Server Secure Remote Access は、大部分がバグ修正です。Release 3 に対して特に重要な新機能は追加されていません。

## Portal Server Secure Remote Access のアップグレードロードマップ

表 18-1 は、Portal Server Secure Remote Access の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 18-1 Java ES Release 4: Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4 へのアップグレードパス

Java ES のリリース	Portal Server Secure Remote Access バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q1	直接アップグレード: パッチを適用することにより実行されます。	なし



表 18-1 Java ES Release 4: Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4 へのアップグレードパス (続き)

Java ES のリリース	Portal Server Secure Remote Access バージョン	一般的な手順	必要な再設定
Release 2	Sun Java System Portal Server Secure Remote Access 6.3 2004Q2	直接アップグレード不可: まず Release 3 にアップグレードし、次に Release 4 にアップグレードするためのパッチを適用することにより実行されます。	設定データ
Release 1	Sun ONE Portal Server Secure Remote Access 6.2 (2003Q4)	直接アップグレード不可: まず Release 3 にアップグレードし、次に Release 4 にアップグレードするためのパッチを適用することにより実行されます。	設定データ
それ以前の Java ES リリース		直接アップグレード不可。	

## Portal Server Secure Remote Access のデータ

次の表は、Portal Server Secure Remote Access ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類の種類を示しています。

表 18-2 Portal Server Secure Remote Access のデータの使用方法

データの種類	場所	使用方法
設定データ	<i>PortalServerConfig-base/</i>	Portal Server Secure Remote Access の設定
ディレクトリスキーマ ユーザーデータ	Directory Server	Portal Server Secure Remote Access は、ディレクトリに格納されているユーザープロファイルデータに依存します。
動的なアプリケーション データ	なし	Portal Server Secure Remote Access は、セッション状態などのアプリケーションデータを永続的に格納しません。

## 互換性の問題

Release 4 Portal Server Secure Remote Access では、インタフェースの変更は導入されていません。Portal Server Secure Remote Access は、以前のバージョンと下位互換性があります。

## Portal Server Secure Remote Access 依存関係

ほかの Java ES コンポーネントに対する Portal Server Secure Remote Access の依存関係が、Portal Server Secure Remote Access ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Portal Server Secure Remote Access のインタフェースまたは関数が変更された場合は、Portal Server Secure Remote Access が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Portal Server Secure Remote Access は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Portal Server Secure Remote Access は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Portal Server**: Portal Server Secure Remote Access は、Portal Server へのセキュリティ保護されたリモートアクセスを提供します。
- **Access Manager (または Access Manager SDK)**: Portal Server Secure Remote Access は、シングルサインオンを含む、エンドユーザーへの認証および承認サービスを提供するために Access Manager に依存します。Access Manager をリモートコンピュータ上で動作させる場合、Access Manager SDK がローカルで利用可能である必要があります。
- **Directory Server**: Portal Server Secure Remote Access は、Directory Server に格納されているユーザーデータにアクセスします。そのため、Portal Server Secure Remote Access のアップグレードによって、ディレクトリスキーマの拡張が必要になる場合があります。

# Java ES Release 3 からの Portal Server Secure Remote Access のアップグレード

ここでは、Portal Server Secure Remote Access の Java ES 2005Q1 (Release 3) から Java ES Release 4 へのアップグレードについて説明します。ここで説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Portal Server Secure Remote Access のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 3 Portal Server Secure Remote Access を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順**：このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。Portal Server Secure Remote Access の再設定は必要ありません。
- **アップグレードの依存関係**：Portal Server Secure Remote Access はいくつかの Java ES 共有コンポーネントに依存しますが ([42 ページの表 1-6](#) を参照)、Release 4 Portal Server Secure Remote Access はこれらのコンポーネントの Release 3 バージョンと互換性があります。したがって、Mobile Access コア (MA コア) を除き、これらの共有コンポーネントのアップグレードは、Portal Server Secure Remote Access の Release 4 へのアップグレードに関しては省略可能です。

さらに、[346 ページの「Portal Server Secure Remote Access 依存関係」](#) で説明されているように、Release 4 Portal Server Secure Remote Access は Portal Server、Access Manager、および Directory Server に依存します。また、Portal Server Secure Remote Access の Release 4 へのアップグレードでは、Portal Server もアップグレードする必要があります。ただし、Access Manager と Directory Server に対する依存関係は弱いアップグレード依存関係であり、これらのコンポーネントのアップグレードは、Portal Server Secure Remote Access の Release 4 へのアップグレードに関しては省略可能です。

- **下位互換性**：Release 4 Portal Server Secure Remote Access は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック**：Portal Server Secure Remote Access の Release 4 アップグレードの Release 3 へのロールバックは、アップグレード中に適用されたパッチをロールバックすることにより行われます。パッチのロールバックは、Linux プラットフォームでは使用できません。

- **プラットフォームの問題** : Portal Server Secure Remote Access をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Portal Server Secure Remote Access のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Portal Server Secure Remote Access の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Portal Server Secure Remote Access のアップグレード \(Solaris\)](#)
- [Release 3 の Portal Server Secure Remote Access のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Portal Server Secure Remote Access をアップグレードする前に、次に説明されている作業を実行してください。

#### **現在のバージョン情報の確認**

Portal Server Secure Remote Access の現在のバージョンは、次のコマンドを使用することにより確認できます。

```
PortalServer-base/bin/gateway version  
PortalServer-base/bin/rwproxyd version  
PortalServer-base/bin/netletd version
```

表 18-3 Portal Server Secure Remote Access のバージョン確認の出力

Java ES のリリース	Portal Server のバージョン番号
Release 2	Release 3 より前のバージョン
Release 3	Thu Dec 16 03:30:34 PST 2004
Release 4	Release 3 より後のバージョン <sup>1</sup>

1. Release 3 と Release 4 の違いはパッチだけです。352 ページの表 18-5 および 362 ページの表 18-7 に示されている Release 4 のパッチは、Solaris の `showrev -p grep patch_ID` コマンドおよび Linux の `rpm -qa sun-portal-core` コマンドを使用して "25.12" 以上の文字列を探すことにより確認できます。

### Portal Server Secure Remote Access が依存するコンポーネントのアップグレード

一般には、コンピュータシステム上 (および、コンピューティング環境内) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Portal Server は、Mobile Access コア (MA コア) 共有コンポーネントに対してのみ強いアップグレード依存関係を持っています。したがって、Portal Server が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

ただし、Portal Server Secure Remote Access のすべての依存関係をアップグレードすることを選択した場合は、Portal Server Secure Remote Access をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの「Java ES 共有コンポーネントのアップグレード」で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「Directory Server と Administration Server」で説明されています。
3. **Access Manager (Access Manager SDK)**: Access Manager を Release 4 にアップグレードする手順は、205 ページの第 11 章「Access Manager」で説明されています。
4. **Portal Server**: Portal Server をアップグレードする手順は、313 ページの第 17 章「Portal Server」で説明されています。

## Release 3 Portal Server Secure Remote Access の設定情報のバックアップ

Portal Server Secure Remote Access の Release 4 へのアップグレードでは、Portal Server Secure Remote Access ソフトウェアの再設定は必要ありません。ただし、安全策として、設定情報が格納されている次のディレクトリをバックアップすることができます。

`PortalServerConfig-base/`

## 必要な設定情報とパスワードの入手

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。

## Release 3 の Portal Server Secure Remote Access のアップグレード (Solaris)

ここでは、Portal Server Secure Remote Access のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Portal Server Secure Remote Access ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- 同じ Portal Server Secure Remote Access インストールイメージに対応するすべての Portal Server Secure Remote Access インスタンスが同時にアップグレードされます。インストールイメージにパッチを適用している間は、このようなインスタンスをすべて停止するようにしてください。
- Solaris OS での Release 4 Portal Server Secure Remote Access のアップグレードパッチは、Portal Server のアップグレードに使用されるものと同じであり、次の表に示されています。

表 18-4 Solaris で Portal Server Secure Remote Access をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC Solaris 8、9、10	x86 Solaris 9、10
Portal Server コア	118950-12	118951-12
Portal Server ローカライズ版	119425-08	119425-08
Portal Server ローカリゼーション コンフィギュレータ	118115-11	118115-11

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

## アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server Secure Remote Access に適用されます。

1. 表 18-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Portal Server Secure Remote Access プロセスを停止します。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

4. MA コア共有コンポーネントやその他の任意のコンポーネントのアップグレードをまだ実行していない場合は、実行します。

349 ページの「Portal Server Secure Remote Access が依存するコンポーネントのアップグレード」を参照してください。

5. 表 18-4 の適切な Portal Server パッチを適用します。

この 2 つの Portal Server ローカライズ版パッチを適用する前に、Portal Server コアパッチを適用するようにしてください。

```
patchadd patch_ID
```

6. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 5 で適用したパッチ ID のバージョンが返されるはずですが。

7. Portal Server Secure Remote Access プロセスを再起動します。

```
/etc/init.d/gateway start
/etc/init.d/netletd start
/etc/init.d/rwproxyd start
```

## Release 3 の Portal Server Secure Remote Access のアップグレード (Linux)

ここでは、Portal Server Secure Remote Access のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

## アップグレードの注意点 (Linux)

Linux プラットフォームで Portal Server Secure Remote Access ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (350 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Portal Server Secure Remote Access のアップグレードパッチは、Portal Server のアップグレードに使用されるものと同じであり、次の表に示されています。

表 18-5 Linux で Portal Server Secure Remote Access をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Portal Server コア	118952-12 <ul style="list-style-type: none"> <li>sun-portal-core-6.3-25.12.i386.rpm</li> </ul> および、ポータルデスクトップと Portal Server モバイルアクセスのためのその他のいくつかの RPM。
Portal Server ローカライズ版	119426-07 <ul style="list-style-type: none"> <li>sun-portal-core-<i>Locale</i>-6.3-24.i386.rpm</li> </ul> および、Portal Server モバイルアクセス、設定、アイデンティティ、その他のコンポーネントのためのその他のいくつかの RPM。
Portal Server ローカリゼーションコンフィギュレータ	118116-08 <ul style="list-style-type: none"> <li>sun-portal-l10n-configurator-6.3-24.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

## アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server Secure Remote Access に適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---



1. [表 18-4](#) のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Portal Server Secure Remote Access プロセスを停止します。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

4. MA コア共有コンポーネントやその他の任意のコンポーネントのアップグレードをまだ実行していない場合は、実行します。

[349 ページの「Portal Server Secure Remote Access が依存するコンポーネントのアップグレード」](#) を参照してください。

5. [表 18-5](#) の Portal Server コアパッチの RPM を適用します。

```
cd /tmp
./update
```

update スクリプトにより RPM がインストールされます。また、パッチの結果として正しい設定変更が行われることも保証されます。

6. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-portal-gateway
```

RPM の新しいバージョン番号が返されることを確認します。

7. [表 18-5](#) の 2 つの Portal Server ローカライズ版パッチの RPM を適用します。

```
rpm -Fvh --replacefiles sun-portal-*-Locale-6.3-24.i386.rpm
rpm -Fvh --replacefiles
    sun-portal-l10n-configurator-6.3-24.i386.rpm
```

8. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-portal-l10n-configurator-6.3-24
```

RPM のアップグレードドリビジョン番号が返されることを確認します。

9. Portal Server Secure Remote Access プロセスを再起動します。

```
/etc/init.d/gateway start
/etc/init.d/netletd start
/etc/init.d/rwproxyd start
```

## アップグレードの確認

Portal Server Secure Remote Access の Release 4 へのアップグレードは、アップグレードパッチが正常に適用されていることを確認することによって検証されます。この確認の手順は、[351 ページの「アップグレード手順 \(Solaris\)」](#) および [352 ページの「アップグレード手順 \(Linux\)」](#) に記載されています。

さらに、次のコマンドを使用することもできます。

```
PortalServer-base/bin/gateway version
PortalServer-base/bin/rwproxyd version
PortalServer-base/bin/netletd version
```

出力値については、[349 ページの表 18-3](#) を参照してください。

パッチのアップグレードに対するこれらのテスト以外に、前に動作していた機能が引き続き正常に動作すること、および目的のバグ修正が実際に修正されていることを確認することができます。

## アップグレード後の作業

[351 ページの「アップグレード手順 \(Solaris\)」](#) および [352 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

ここでは、Portal Server Secure Remote Access のアップグレードのロールバック手順に影響する注意点を示したあと、その手順自体について説明します。

### ロールバックの注意点 (Solaris)

Portal Server Secure Remote Access の Release 4 へのアップグレードをロールバックする手順は、Release 4 にアップグレードする手順のほぼ逆になります。再設定がロールバックされ、パッチが削除されます。

### ロールバック手順 (Solaris)

1. root としてログインするか、スーパーユーザーになります。

```
su -
```

2. Portal Server Secure Remote Access プロセスを停止します。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

3. [350 ページの表 18-4](#) のパッチを削除します。

```
patchrm patch_ID
```

4. Portal Server Secure Remote Access プロセスを再起動します。

```
/etc/init.d/gateway start  
/etc/init.d/netletd start  
/etc/init.d/rwproxyd start
```

## 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、スケーラビリティを実現し、可用性を向上させるために、Portal Server Secure Remote Access が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Portal Server Secure Remote Access コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。

Portal Server Secure Remote Access のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Portal Server Secure Remote Access インスタンスをアップグレードする順次アップグレードを実行できます。Portal Server Secure Remote Access の各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。348 ページの「[Release 3 の Portal Server Secure Remote Access のアップグレード](#)」の説明に従って、各インスタンスのアップグレードを実行します。

# Java ES Release 2 からの Portal Server Secure Remote Access のアップグレード

ここでは、Portal Server Secure Remote Access の Java ES Release 2 から Java ES Release 4 へのアップグレードについて説明します。

このアップグレードは、まず Release 2 から Release 3 にアップグレードし、次に Release 3 から Release 4 にアップグレードするという 2 段階で実行されます。この 2 つのアップグレードパスは異なるため、ここでは、Release 2 から Release 3 へのアップグレードに焦点を絞って説明します。

Release 2 から Release 3 へのアップグレードが完了したら、[347 ページの「Java ES Release 3 からの Portal Server Secure Remote Access のアップグレード」](#)の説明に従って、Release 3 から Release 4 へのアップグレードに進むことができます。

ここでは、Release 2 から Release 3 へのアップグレードに関して次の内容を説明します。

- [はじめに](#)
- [Release 3 の Portal Server Secure Remote Access のアップグレード](#)
- [複数インスタンスのアップグレード](#)

## はじめに

Java ES Release 2 Portal Server Secure Remote Access を Release 3 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順**：このアップグレードは、Release 2 バージョンにパッチを適用することにより実行されます。また、Portal Server Secure Remote Access の再設定も必要であり、アップグレードユーティリティーを使用して実行されます。
- **アップグレードの依存関係**：Portal Server Secure Remote Access は、いくつかの Java ES 共有コンポーネントに依存します ([42 ページの表 1-6](#) を参照)。Java ES では 1 つのコンピュータ上での Release 2 および Release 3 コンポーネントの混在がサポートされていないため、これらの共有コンポーネントをすべて Release 3 にアップグレードする必要があります。

さらに、346 ページの「Portal Server Secure Remote Access 依存関係」で説明されているように、Release 3 Portal Server Secure Remote Access は Portal Server、Access Manager、および Directory Server に依存します。Portal Server Secure Remote Access は、ローカルに存在する Portal Server と Access Manager (または Access Manager SDK) に対しては強いアップグレード依存関係を持っており、ローカルに存在することがほとんどない Directory Server に対しては弱いアップグレード依存関係を持っています。

- **下位互換性** : Release 3 Portal Server Secure Remote Access は、Release 2 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Portal Server Secure Remote Access の再設定が実行されたあとは、Portal Server Secure Remote Access の Release 3 アップグレードを Release 2 にロールバックすることはできません。
- **プラットフォームの問題** : Portal Server Secure Remote Access をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 2 Portal Server Secure Remote Access のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Portal Server Secure Remote Access の Java ES Release 2 から Java ES Release 3 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Portal Server Secure Remote Access のアップグレード \(Solaris\)](#)
- [Release 3 の Portal Server Secure Remote Access のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Portal Server Secure Remote Access をアップグレードする前に、次に説明されている作業を実行してください。

### 現在のバージョン情報の確認

Portal Server Secure Remote Access の現在のバージョンは、次のコマンドを使用することにより確認できます。

```
PortalServer-base/bin/gateway version
PortalServer-base/bin/rwproxyd version
PortalServer-base/bin/netletd version
```

出力値については、[349 ページの表 18-3](#) を参照してください。

### Portal Server Secure Remote Access が依存するコンポーネントのアップグレード

Java ES Release 3 では、1 つのコンピュータ上での Release 2 および Release 3 共有コンポーネントの共存はサポートされていません。

そのため、Portal Server Secure Remote Access が依存するローカルのすべての Java ES Release 2 コンポーネントを Release 3 にアップグレードする必要があります。アップグレードの手順については、『Java Enterprise System 2005Q1 アップグレードと移行』(<http://docs.sun.com/doc/819-2235?l=ja>) を参照してください。

コンピュータ上のローカルのすべての Portal Server の依存関係をアップグレードする場合は、Portal Server Secure Remote Access をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 3 にアップグレードする手順は、『Java Enterprise System 2005Q1 アップグレードと移行』(<http://docs.sun.com/doc/819-2235?l=ja>) で説明されています。
2. **Portal Server**: Portal Server Secure Remote Access が、ローカルの Portal Server に依存することはほとんどありません。
3. **Access Manager (Access Manager SDK)**: Access Manager を Release 3 にアップグレードする手順は、『Java Enterprise System 2005Q1 アップグレードと移行』(<http://docs.sun.com/doc/819-2235?l=ja>) で説明されています。
4. **Directory Server**: Portal Server が、ローカルの Directory Server に依存することはほとんどありません。

### Release 2 Portal Server Secure Remote Access の設定情報のバックアップ

Portal Server Secure Remote Access の Release 3 へのアップグレードでは、Portal Server Secure Remote Access ソフトウェアの再設定が必要です。安全策として、設定情報が格納されている次のディレクトリをバックアップすることができます。

```
PortalServerConfig-base/
```

## 必要な設定情報とパスワードの入手

アップグレードを実行するには、スーパーユーザーとしてログインする必要があります。

## Release 2 Portal Server Secure Remote Access のアップグレード (Solaris)

ここでは、Portal Server Secure Remote Access のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Solaris)

Portal Server Secure Remote Access ソフトウェアの Java ES Release 3 へのアップグレードでは、次の注意点を考慮に入れてください。

- 同じ Portal Server Secure Remote Access インストールイメージに対応するすべての Portal Server Secure Remote Access インスタンスが同時にアップグレードされます。インストールイメージにパッチを適用する前に、このようなインスタンスをすべて停止するようにしてください。
- Rewriter プロキシ、Netlet プロキシ、およびゲートウェイプロセスが別のコンピュータで稼働している場合は、ゲートウェイサービスをホストしている任意のコンピュータでアップグレードを実行する前に、Rewriter および Netlet プロキシサービスをホストしているコンピュータでアップグレード手順を実行してください。
- Solaris OS での Release 3 Portal Server Secure Remote Access のアップグレードパッチは、Portal Server のアップグレードに使用されるものと同じであり、次の表に示されています。

表 18-6 Solaris で Portal Server Secure Remote Access を Release 3 にアップグレードするためのパッチ

説明	SPARC Solaris 8、9	x86 Solaris 9	Solaris 10
Portal Server sync-up	118195-07	118196-07	適用なし
Portal Server コア	118128-13	118129-13	適用なし
Mobile Access コア	118219-12	118219-12	適用なし
Portal Server の修正	118950-15 (以上)	118951-15 (以上)	適用なし

### アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server Secure Remote Access に適用されます。

Portal Server Secure Remote Access が Portal Server と同じコンピュータにインストールされている場合は、317 ページの「Java ES Release 3 からの Portal Server のアップグレード」の手順に従ってください。次に、360 ページの手順 8 に進みます。

1. 表 18-6 に基づいて、必要なパッチを入手します。

表 18-6 に示されている正確なパッチリビジョンをダウンロードするようにしてください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Portal Server Secure Remote Access プロセスを停止します。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

4. すべての共有コンポーネントおよび Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

358 ページの「Portal Server Secure Remote Access が依存するコンポーネントのアップグレード」を参照してください。

5. まだ稼働していない場合は、Directory Server と Access Manager (または Access Manager SDK) を起動します。

6. 表 18-6 の適切な Portal Server のパッチを適用します。

パッチは、表 18-6 に示されている上から下の順序で適用するようにしてください。

```
patchadd patch_ID
```

7. パッチのアップグレードが成功したことを確認します。

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

この出力では、手順 5 で適用したパッチ ID のバージョンが返されるはずです。

8. Portal Server Secure Remote Access ソフトウェアを再設定します。

```
ksh
$ cd PortalServer-base/lib
$ ./upgradeSRA-04Q4-05Q1
```

9. Portal Server Secure Remote Access プロセスを再起動します。

```
/etc/init.d/gateway start
/etc/init.d/netletd start
/etc/init.d/rwproxyd start
```



## 10. プロキシレットサービスと Netlet サービスを再設定します。

- a. Access Manager コンソール (AMCONSOLE) に管理ユーザーとしてログオンします。
- b. プロキシレットサービスと Netlet サービスを削除します。

「アイデンティティ管理」タブで「サービス」オプションを選択します。すると、左側のパネル上に、登録されているすべてのサービスが一覧表示されます。「SRA 設定」で「プロキシレット」と「Netlet」の両チェックボックスを選択します。スクロールして左側のパネルの最上部に移動し、「削除」ボタンをクリックします。すると、プロキシレットサービスと Netlet サービスが、組織レベルから削除されます。

この手順の完了を手動で確認するには、LDAP ディレクトリ (所属する組織の下) で、それらのサービス (srapProxylet と srapNetlet) が実際に削除されていることを確認します。

- c. サービスを再度追加します。

「アイデンティティ管理」タブで「サービス」オプションを選択します。「サービス」の下にある「追加」ボタンをクリックします。すると、右側のパネル上に、利用可能なすべてのサービスが表示されます。プロキシレットサービスと Netlet サービスのチェックボックスを選択し、「了解」をクリックします。左側のパネルの「SRA 設定」の下に、新しく追加したサービスが表示されます。

- d. 新しく追加したサービスをクリックし、テンプレートファイルを作成します。「保存」ボタンをクリックします。
- e. ゲートウェイサービスの下の非認証 URL リストに、  
/portal/netlet/jnlpclient.jar と /portal/netlet/netletjsse.jar を追加します。\*
  - o 「サービス設定」タブをクリックします。
  - o 「SRA 設定」の下にある「ゲートウェイ」リンクをクリックします。すると、利用可能なすべてのゲートウェイプロファイルが一覧表示されます。
  - o リンクをクリックして適切なプロファイルを選択します。
  - o 「セキュリティ」タブをクリックします。
  - o 非認証 URL の下にある編集フィールドに /portal/netlet/jnlpclient.jar を追加し、「追加」ボタンをクリックします。
  - o 非認証 URL の下にある編集フィールドに /portal/netlet/netletjsse.jar を追加し、「追加」ボタンをクリックします。
  - o ページの最下部にある「保存」ボタンをクリックします。

- f. ゲートウェイサーバーを再起動します。

```
/etc/init.d/gateway stop
/etc/init.d/gateway start
```

11. Portal Server Secure Remote Access を Release 3 から Release 4 にアップグレードします。

347 ページの「Java ES Release 3 からの Portal Server Secure Remote Access のアップグレード」の手順に従ってください。

## Release 2 Portal Server Secure Remote Access のアップグレード (Linux)

ここでは、Portal Server Secure Remote Access のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Portal Server Secure Remote Access ソフトウェアを Java ES Release 3 にアップグレードするときは、Linux の Release 3 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (350 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 3 Portal Server Secure Remote Access のアップグレードパッチを次の表に示します。

表 18-7 Linux で Portal Server Secure Remote Access を Release 3 にアップグレードするためのパッチ

説明	パッチ ID と RPM 名
Portal Server コア	118020-16  sun-portal- <i>module</i> -6.3-25.i386.rpm  ここで、 <i>module</i> は、約 70 種類あるソフトウェアモジュールのいずれかです
Mobile Access コア	119515-01  <ul style="list-style-type: none"> <li>• sun-mobileaccess-1.0-25.i386.rpm</li> <li>• sun-mobileaccess-config-1.0-25.i386.rpm</li> </ul>

表 18-7 Linux で Portal Server Secure Remote Access を Release 3 にアップグレードするためのパッチ (続き)

説明	パッチ ID と RPM 名
Portal Server の修正	118952-01
	<ul style="list-style-type: none"> <li>• sun-portal-core-6.3-xx.y.i386.rpm</li> <li>• sun-portal-configurator-6.3-xx.y.i386.rpm</li> <li>• sun-portal-mobileaccess-6.3-xx.y.i386.rpm</li> <li>• sun-portal-desktop-6.3-xx.y.i386.rpm</li> <li>• sun-portal-sample-6.3-xx.y.i386.rpm</li> <li>• sun-portal-mobileaccess-config-6.3-xx.y.i386.rpm</li> </ul>

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Portal Server Secure Remote Access に適用されます。

Portal Server Secure Remote Access が Portal Server と同じコンピュータにインストールされている場合は、317 ページの「Java ES Release 3 からの Portal Server のアップグレード」の手順に従ってください。次に、364 ページの手順 8 に進みます。

1. 表 18-6 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. Portal Server Secure Remote Access プロセスを停止します。

```
/etc/init.d/gateway stop
/etc/init.d/netletd stop
/etc/init.d/rwproxyd stop
```

4. すべての共有コンポーネントおよび Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

358 ページの「Portal Server Secure Remote Access が依存するコンポーネントのアップグレード」を参照してください。

5. まだ稼働していない場合は、Directory Server と Access Manager (または Access Manager SDK) を起動します。
6. 表 18-6 の Portal Server の RPM を適用します。

- a. `cd /tmp`
  - b. 118020 パッチファイルを解凍し、README ファイルを参照して、次のスクリプトを実行します。  

```
./upgradportalrpms
```

**update** スクリプトにより RPM がインストールされます。また、パッチの結果として正しい設定変更が行われることも保証されます。
  - c. 119515 パッチファイルを解凍し、含まれている README ファイルの指示に従います。
  - d. 118952 パッチファイルを解凍し、含まれている README ファイルの指示に従います。
7. パッチのアップグレードが成功したことを確認します。
- ```
rpm -qa | grep sun-portal-gateway  
rpm -qa | grep sun-mobileaccess
```
- RPM の新しいバージョン番号が返されることを確認します。
8. Portal Server Secure Remote Access ソフトウェアを再設定します。
- ```
ksh  
$ cd PortalServer-base/lib  
$ ./upgradeSRA-04Q4-05Q1
```
9. Portal Server Secure Remote Access プロセスを再起動します。
- ```
/etc/init.d/gateway start  
/etc/init.d/netletd start  
/etc/init.d/rwproxyd start
```
10. プロキシレットサービスと Netlet サービスを再設定します。
- a. **Access Manager** コンソール (AMCONSOLE) に管理ユーザーとしてログオンします。
  - b. プロキシレットサービスと Netlet サービスを削除します。  

「アイデンティティ管理」タブで「サービス」オプションを選択します。すると、左側のパネル上に、登録されているすべてのサービスが一覧表示されます。「SRA 設定」で「プロキシレット」と「Netlet」の両チェックボックスを選択します。スクロールして左側のパネルの最上部に移動し、「削除」ボタンをクリックします。すると、プロキシレットサービスと Netlet サービスが、組織レベルから削除されます。

この手順の完了を手動で確認するには、LDAP ディレクトリ (所属する組織の下) で、それらのサービス (srapProxylet と srapNetlet) が実際に削除されていることを確認します。

- c. サービスを再度追加します。
- 「アイデンティティ管理」タブで「サービス」オプションを選択します。「サービス」の下にある「追加」ボタンをクリックします。すると、右側のパネル上に、利用可能なすべてのサービスが表示されます。プロキシレットサービスと Netlet サービスのチェックボックスを選択し、「了解」をクリックします。左側のパネルの「SRA 設定」の下に、新しく追加したサービスが表示されます。
- d. 新しく追加したサービスをクリックし、テンプレートファイルを作成します。「保存」ボタンをクリックします。
- e. ゲートウェイサービスの下の非認証 URL リストに、  
/portal/netlet/jnlpclient.jar と /portal/netlet/netletjsse.jar を追加します。\*
- o 「サービス設定」タブをクリックします。
  - o 「SRA 設定」の下にある「ゲートウェイ」リンクをクリックします。すると、利用可能なすべてのゲートウェイプロファイルが一覧表示されます。
  - o リンクをクリックして適切なプロファイルを選択します。
  - o 「セキュリティ」タブをクリックします。
  - o 非認証 URL の下にある編集フィールドに /portal/netlet/jnlpclient.jar を追加し、「追加」ボタンをクリックします。
  - o 非認証 URL の下にある編集フィールドに /portal/netlet/netletjsse.jar を追加し、「追加」ボタンをクリックします。
  - o ページの最下部にある「保存」ボタンをクリックします。
- f. ゲートウェイサーバーを再起動します。
- ```
/etc/init.d/gateway stop
/etc/init.d/gateway start
```

11. Portal Server Secure Remote Access を Release 3 から Release 4 にアップグレードします。

347 ページの「Java ES Release 3 からの Portal Server Secure Remote Access のアップグレード」の手順に従ってください。

## アップグレードの確認

Portal Server Secure Remote Access の Release 3 へのアップグレードは、アップグレードパッチが正常に適用されていることを確認することによって検証されます。この確認の手順は、351 ページの「アップグレード手順 (Solaris)」および 352 ページの「アップグレード手順 (Linux)」に記載されています。

さらに、次のコマンドを使用することもできます。

```
PortalServer-base/bin/gateway version  
PortalServer-base/bin/rwproxyd version  
PortalServer-base/bin/netletd version
```

出力値については、[349 ページの表 18-3](#) を参照してください。

パッチのアップグレードに対するこれらのテスト以外に、前に動作していた機能が引き続き正常に動作すること、および目的のバグ修正が実際に修正されていることを確認することができます。

## アップグレード後の作業

[359 ページの「アップグレード手順 \(Solaris\)」](#) および [363 ページの「アップグレード手順 \(Linux\)」](#) で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック

Portal Server Secure Remote Access の Release 2 から Release 3 へのアップグレードはロールバックできません。

# 複数インスタンスのアップグレード

配備のアーキテクチャーによっては、スケーラビリティを実現し、可用性を向上させるために、Portal Server Secure Remote Access が複数のコンピュータシステムに配備される場合があります。たとえば、ロードバランサを使用して Portal Server Secure Remote Access コンポーネントを複数のコンピュータで実行することにより、負荷を分散させることができます。

Portal Server Secure Remote Access のインスタンスが負荷分散されている場合は、サービスを中断することなく連続して Portal Server Secure Remote Access インスタンスをアップグレードする順次アップグレードを実行できます。Portal Server Secure Remote Access の各インスタンスを、ほかのインスタンスが稼働している間にアップグレードします。[348 ページの「Release 3 の Portal Server Secure Remote Access のアップグレード」](#) の説明に従って、各インスタンスのアップグレードを実行します。

# Delegated Administrator

この章では、Delegated Administrator を Java ES 2005Q4 (Release 4): Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 にアップグレードする方法について説明します。ここでは、Java ES Release 4 でサポートされている各種のアップグレードパスでのアップグレードの問題と手順の概要について説明します。また、Solaris オペレーティングシステムと Linux オペレーティングシステムの両方でのアップグレードを取り上げます。

- [368 ページの「Delegated Administrator のアップグレードの概要」](#)
- [371 ページの「Java ES Release 3 からの Delegated Administrator のアップグレード」](#)
- [379 ページの「Java ES Release 2 からの Delegated Administrator のアップグレード」](#)

---

**注** この章で説明するファイルの場所は、*DelegatedAdmin-base* というディレクトリパスを基準に指定されています。このパスの少なくとも一部は、Delegated Administrator が最初にインストールされたときにインストールディレクトリとして指定されている可能性があります。指定されていない場合、Java ES インストーラによってデフォルト値が割り当てられています。

*DelegatedAdmin-base* のデフォルト値は、オペレーティングシステムプラットフォームによって異なります。

- **Solaris:** /opt/SUNWcomm
  - **Linux:** /opt/sun/comms/commcli
-

# Delegated Administrator のアップグレードの概要

ここでは、Java ES 2005Q4 (Release 4) へのアップグレードに影響する Delegated Administrator の次の一般的な側面について説明します。

- [Java ES Release 4 の Delegated Administrator について](#)
- [Delegated Administrator のアップグレードロードマップ](#)
- [Delegated Administrator のデータ](#)
- [互換性の問題](#)
- [Delegated Administrator 依存関係](#)

## Java ES Release 4 の Delegated Administrator について

Java ES Release 4 Delegated Administrator の Release 3 と比較した場合の新機能には、カレンダーサービスのプロビジョニング、メールグループのプロビジョニング、ユーザビリティフィードバックに基づく UI ナビゲーションの強化、および各種のバグ修正があります。

## Delegated Administrator のアップグレードロードマップ

表 19-1 は、Delegated Administrator の、Java ES Release 4 へのサポートされているアップグレードパスを示しています。この表は Solaris および Linux の両方のオペレーティングシステムに適用されます。

表 19-1 Java ES Release 4: Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 へのアップグレードパス

Java ES のリリース	Delegated Administrator バージョン	一般的な手順	必要な再設定
Release 3	Sun Java System Communication Services Delegated Administrator 6 2005Q1	直接アップグレード: パッチを適用し、config-commda ユーティリティを実行することにより実行されます。	設定データ



表 19-1 Java ES Release 4: Sun Java System Communication Services Delegated Administrator 6.3 2005Q4 へのアップグレードパス (続き)

Java ES のリリース	Delegated Administrator バージョン	一般的な手順	必要な再設定
Release 2	Sun Java System Communication Services ユーザー管理ユーティリティ 1.1 (2004Q2)	直接アップグレード: パッチを適用し、config-commanda ユーティリティを実行することにより実行されます。	設定データ
Release 1	ユーザー管理ユーティリティ (2003Q4)	直接アップグレードは動作保証なし: ただし、パッチを適用し、config-commanda ユーティリティを実行することにより実行できます。	設定データ
それ以前の Java ES リリース	iPlanet Delegated Administrator	直接アップグレード不可	

## Delegated Administrator のデータ

次の表は、Delegated Administrator ソフトウェアのアップグレードにより影響を受ける可能性のあるデータの種類を示しています。

表 19-2 Delegated Administrator のデータの使用方法

データの種類	場所	使用方法
ディレクトリスキーマ	Directory Server のユーザー / グループディレクトリ	エンドユーザー、組織、およびサービススキーマのサポートに必要な属性の格納
設定データ	<i>DelegatedAdmin-base</i> /data/WEB-INF/classes/sun/com/m/cli/server/servlet/resource.properties	Delegated Administrator サーバーの設定とカスタマイズ
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/com/sun/comm/da/resources/daconfig.properties	Delegated Administrator コンソールの設定
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/com/sun/comm/da/resources/logger.properties	Delegated Administrator コンソールのログ
	<i>DelegatedAdmin-base</i> /data/da/WEB-INF/classes/com/sun/comm/da/resources/security.properties	Delegated Administrator コンソールの設定

表 19-2 Delegated Administrator のデータの使用方法 ( 続き )

データの種類	場所	使用法
Web コンテナ の設定	<p>Web Server: 次の場所にある <code>server.policy</code> および <code>server.xml</code> ファイル <code>WebServer-base/https-hostname/config</code></p> <p>Application Server (Java ES Release 3 および 4): 次の場所にある <code>server.policy</code> および <code>domain.xml</code> ファイル <code>AppServer8Config-base/domains/domainName/config</code></p> <p>Application Server (Java ES Release 2): 次の場所にある <code>server.policy</code> および <code>server.xml</code> ファイル <code>AppServer7Config-base/domains/domainName/config</code></p>	Delegated Administrator Web コ ンテナインスタンスの設定

## 互換性の問題

Release 4 Delegated Administrator ではグラフィカルユーザーインターフェースの変更が導入されていますが、以前のバージョンとは下位互換性があります。

## Delegated Administrator 依存関係

ほかの Java ES コンポーネントに対する Delegated Administrator の依存関係が、Delegated Administrator ソフトウェアのアップグレードおよび再設定の手順に影響を及ぼす可能性があります。たとえば、Delegated Administrator のインターフェースまたは関数を変更された場合は、Delegated Administrator が依存するコンポーネントのアップグレードされたバージョンが必要になる場合があります。このようなコンポーネントのアップグレードの必要性は、個々のアップグレードパスによって異なります。

Delegated Administrator は、次の Java ES コンポーネントに依存します。

- **共有コンポーネント** : Delegated Administrator は、特定の Java ES 共有コンポーネントに依存します (42 ページの表 1-6 を参照)。
- **Web コンテナ** : Delegated Administrator は、Java ES Web Server または Java ES Application Server のどちらかで提供される Web コンテナサービスに依存します。
- **Directory Server** : Delegated Administrator は、アプリケーションとユーザーのデータを Directory Server に格納します。

- **Directory Preparation Tool: Delegated Administrator** は Directory Preparation Tool を使用して、ディレクトリが Delegated Administrator のユーザープロビジョニング機能をサポートできるように準備します。そのため、Delegated Administrator のアップグレードは、ディレクトリが新機能をサポートできるように準備されているかどうかにかかわらず依存する場合があります。
- **Access Manager (Access Manager SDK): Delegated Administrator** は、サービスを登録し、Directory Server にエントリを作成するために Access Manager に依存します。

## Java ES Release 3 からの Delegated Administrator のアップグレード

ここでは、Delegated Administrator の Java ES 2005Q1 (Release 3) から Java ES Release 4 へのアップグレードについて説明します。ここで説明する内容は、次のとおりです。

- [はじめに](#)
- [Release 3 の Delegated Administrator のアップグレード](#)

### はじめに

Java ES Release 3 Delegated Administrator を Release 4 にアップグレードする場合は、アップグレード処理の次の側面を考慮してください。

- **一般的なアップグレード手順** : このアップグレードは、Release 3 バージョンにパッチを適用することにより実行されます。Delegated Administrator の再設定は、`config-commanda` 設定ユーティリティを実行することにより行われます。
- **アップグレードの依存関係** : Delegated Administrator は多くの Java ES 共有コンポーネントに対する依存関係を持ちますが ([42 ページの表 1-6](#) を参照)、Release 4 の Delegated Administrator はこれらすべてのコンポーネントの Release 3 バージョンと互換性があります。したがって、これらの共有コンポーネントのアップグレードは、Delegated Administrator の Release 4 へのアップグレードに関しては省略可能です。

さらに、[370 ページの「Delegated Administrator 依存関係」](#) で説明されているように、Release 4 Delegated Administrator は Web コンテナと Access Manager に依存します。これらはアップグレード依存関係が弱いものであり、これらのコンポーネントのアップグレードは、Delegated Administrator の Release 4 へのアップグレードに関しては省略可能です。ただし、Access Manager をアップグレードする場合は、Delegated Administrator もアップグレードする必要があります。

ただし、Release 4 Delegated Administrator は、Directory Preparation Tool に対して強いアップグレード依存関係を持っています。ユーザープロビジョニング操作が可能のように Directory Server を準備するには、Release 4 Directory Preparation Tool が必要です。

- **下位互換性** : Release 4 Delegated Administrator は、Release 3 バージョンと下位互換性があります。
- **アップグレードのロールバック** : Delegated Administrator の Release 4 アップグレードの Release 3 へのロールバックはサポートされていません。
- **プラットフォームの問題** : Delegated Administrator をアップグレードするための一般的な手順は、Solaris および Linux の両オペレーティングシステムで共通ですが、パッチの適用に関して技術的に異なる点があります。したがって、アップグレードプロセスにはプラットフォーム固有の手順が含まれます。

## Release 3 の Delegated Administrator のアップグレード

ここでは、Solaris プラットフォームと Linux プラットフォームの両方で、Delegated Administrator の Java ES Release 3 から Java ES Release 4 へのアップグレードを実行する方法について説明します。プラットフォーム固有の手順に依存する項目には、その手順が適用されるオペレーティングシステムが示されています。この節で説明する内容は、次のとおりです。

- [アップグレード前の作業](#)
- [Release 3 の Delegated Administrator のアップグレード \(Solaris\)](#)
- [Release 3 の Delegated Administrator のアップグレード \(Linux\)](#)
- [アップグレードの確認](#)
- [アップグレード後の作業](#)
- [アップグレードのロールバック \(Solaris\)](#)

### アップグレード前の作業

Delegated Administrator をアップグレードする前に、次に説明されている作業を実行してください。

#### 現在のバージョン情報の確認

Delegated Administrator の現在のバージョンは、次のコマンドを入力することにより確認できます。

```
DelegatedAdmin-base/bin/commadmin -V
```

表 19-3 Delegated Administrator のバージョン確認の出力

Java ES のリリース	Delegated Administrator のバージョン番号
Release 2	ユーザー管理ユーティリティ 1.1
Release 3	ユーザー管理ユーティリティ 6 2005Q1
Release 4	Delegated Administrator 6.3-0.09

### 必要なオペレーティングシステムのパッチの適用

Solaris 10 オペレーティングシステムプラットフォームでは、Delegated Administrator のアップグレード手順を実行するために、オペレーティングシステムのパッチを適用する必要があります (33 ページの「[必須のオペレーティングシステムパッチ](#)」を参照)。

### Delegated Administrator が依存するコンポーネントのアップグレード

一般には、コンピュータシステム上 (および、コンピューティング環境内) のすべての Java ES コンポーネントを Java ES Release 4 にアップグレードすることをお勧めします。ただし、Delegated Administrator は、Directory Preparation Tool に対してのみ強いアップグレード依存関係を持っています。したがって、Delegated Administrator が依存するほかの Java ES Release 3 コンポーネントのアップグレードは省略可能です。

ただし、Delegated Administrator のすべての依存関係をアップグレードすることを選択した場合は、Delegated Administrator をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの「[Java ES 共有コンポーネントのアップグレード](#)」で説明されています。
2. **Directory Server**: Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「[Directory Server と Administration Server](#)」で説明されています。
3. **Web コンテナソフトウェア**: Web Server または Application Server をアップグレードする手順は、それぞれ 141 ページの第 6 章「[Web Server](#)」および 177 ページの第 9 章「[Application Server](#)」で説明されています。
4. **Access Manager (Access Manager SDK)**: Access Manager を Release 4 にアップグレードする手順は、205 ページの第 11 章「[Access Manager](#)」で説明されています。

5. **Directory Preparation Tool: Release 4 Delegated Administrator** を使用する前に、Directory Server に対して Release 4 Directory Preparation Tool を実行する必要があります。Directory Server に対して Directory Preparation Tool をまだ実行していない場合は、Directory Preparation Tool を Release 4 にアップグレードし、それを使用して Directory Server のスキーマを変更および拡張します (手順については、[235 ページの第 12 章「Directory Preparation Tool」](#) を参照)。

### *Delegated Administrator のデータのバックアップ*

Delegated Administrator の Release 3 から Release 4 へのアップグレードでは、Delegated Administrator の再設定が必要です。安全策として、設定データおよびカスタマイズしている任意の Release 3 グラフィカルユーザーインターフェースをバックアップすることをお勧めします。

### *必要な設定情報とパスワードの入手*

現在インストールされているバージョンに関して、次の情報を確認してください。

- Access Manager 管理者のユーザー ID とパスワード
- Access Manager 内部の LDAP パスワード
- 最上位管理者のユーザー ID とパスワード
- Application Server を Web コンテナとして使用している場合は、Application Server 管理者のユーザー ID とパスワード

## **Release 3 の Delegated Administrator のアップグレード (Solaris)**

ここでは、Delegated Administrator のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### *アップグレードの注意点 (Solaris)*

Delegated Administrator ソフトウェアの Java ES Release 4 へのアップグレードでは、次の注意点を考慮に入れてください。

- インストールイメージにパッチを適用している間は、Delegated Administrator を使用しないようにしてください。
- Delegated Administrator を再設定するときは、Access Manager と同じ Web コンテナに配備するようにしてください。
- Solaris OS での Release 4 Delegated Administrator のアップグレードパッチを次の表に示します。

表 19-4 Solaris で Delegated Administrator をアップグレードするためのパッチ<sup>1</sup>

説明	SPARC Solaris 8、9、10	x86 Solaris 9、10
Delegated Administrator	119777-09	119778-09

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Solaris)

次に示す手順は、アップグレードを実行するコンピュータ上の Delegated Administrator に適用されます。

1. 表 19-4 に基づいて、必要なパッチを入手します。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. すべての共有コンポーネント、Web コンテナ、および Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

[373 ページの「Delegated Administrator が依存するコンポーネントのアップグレード」](#)を参照してください。

4. 表 19-4 の適切な Delegated Administrator パッチを適用します。

```
patchadd patch_ID
```

5. パッチのアップグレードが成功したことを確認します。

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

この出力では、[手順 4](#) で適用したパッチ ID のバージョンが返されるはずですが。

6. Delegated Administrator を再設定します。
  - a. Directory Server が稼働していることを確認します。

- b. Web コンテナで Access Manager が実行され、Delegated Administrator が稼働していることを確認します。

Web コンテナを起動するには、次のコマンドを実行します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName  
--user admin_ID --password password
```

ここで、Access Manager と Delegated Administrator は、*domainName* ドメイン内のサーバーインスタンスに配備されています。デフォルトの *domainName* は *domain1* であり、デフォルトのサーバーインスタンスは *server1* です。

- c. Delegated Administrator 設定ユーティリティーを実行します。

```
DelegatedAdmin-base/sbin/config-commda
```

このユーティリティーの方法使用の詳細については、『Sun Java System Communications Services 6 2005Q4 Delegated Administrator 管理ガイド』(<http://docs.sun.com/doc/819-4103?l=ja>) の第 3 章「Delegated Administrator の設定」を参照してください。

7. Delegated Administrator Web コンテナを停止し、再起動します。

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
```

## Release 3 の Delegated Administrator のアップグレード (Linux)

ここでは、Delegated Administrator のアップグレード手順に影響する注意点を示したあと、その手順自体について説明します。

### アップグレードの注意点 (Linux)

Linux プラットフォームで Delegated Administrator ソフトウェアを Java ES Release 4 にアップグレードするときは、Linux の Release 4 アップグレードパッチが Solaris のパッチとは異なる点を除き、Solaris プラットフォームのときと同じ注意点 (374 ページの「アップグレードの注意点 (Solaris)」を参照) を考慮に入れてください。

Linux OS での Release 4 Delegated Administrator のアップグレードパッチを次の表に示します。



表 19-5 Linux で Delegated Administrator をアップグレードするためのパッチ<sup>1</sup>

説明	パッチ ID と RPM 名
Delegated Administrator	119779-09
	<ul style="list-style-type: none"> <li>• sun-commcli-client-1.1-11.9.i386.rpm</li> <li>• sun-commcli-server-1.1-11.9.i386.rpm</li> </ul>

1. この表に示されているパッチのリビジョン番号は、Java ES Release 4 にアップグレードするための最低要件です。表に記載されているものよりも新しいリビジョンのパッチが使用可能になった場合は、新しいほうのパッチを使用してください。

### アップグレード手順 (Linux)

次に示す手順は、アップグレードを実行するコンピュータ上の Delegated Administrator に適用されます。

---

**注意** Linux での Java ES Release 3 から Java ES Release 4 へのアップグレードはロールバックできません。

---

1. 表 19-5 のパッチ番号と RPM 名に基づいて、必要なパッチを入手します。この情報を参照して、RPM のバージョン番号を確認してください。

パッチは <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> から /tmp にダウンロードできます。

2. root としてログインするか、スーパーユーザーになります。

```
su -
```

3. すべての共有コンポーネント、Web コンテナ、および Access Manager (または Access Manager SDK) のアップグレードをまだ実行していない場合は、実行します。

[373 ページの「Delegated Administrator が依存するコンポーネントのアップグレード」](#)を参照してください。

4. 表 19-5 の Delegated Administrator の RPM を適用します。

```
rpm -Fvh sun-commcli-client-1.1-11.9.i386.rpm
rpm -Fvh sun-commcli-server-1.1-11.9.i386.rpm
```

5. パッチのアップグレードが成功したことを確認します。

```
rpm -qa | grep sun-commcli
```

RPM の新しいバージョン番号が返されることを確認します。

6. Delegated Administrator を再設定します。
  - a. Web コンテナで Access Manager が実行され、Delegated Administrator が稼働していることを確認します。

Web コンテナを起動するには、次のコマンドを実行します。

Web Server:

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

Application Server:

```
AppServer8-base/bin/asadmin start-domain domainName
--user admin_ID --password password
```

ここで、Access Manager と Delegated Administrator は、*domainName* ドメイン内のサーバーインスタンスに配備されています。デフォルトの *domainName* は *domain1* であり、デフォルトのサーバーインスタンスは *server1* です。

- b. Delegated Administrator 設定ユーティリティーを実行します。

```
DelegatedAdmin-base/sbin/config-commda
```

このユーティリティーの方法使用の詳細については、『Sun Java System Communications Services 6 2005Q4 Delegated Administrator 管理ガイド』(<http://docs.sun.com/doc/819-4103?l=ja>) の第 3 章「Delegated Administrator の設定」を参照してください。

7. Delegated Administrator Web コンテナを停止し、再起動します。

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
```

## アップグレードの確認

Delegated Administrator が正常にアップグレードされたかどうかは、次のようにして確認できます。

1. バージョン番号を確認します。

```
DelegatedAdmin-base/bin/commadmin -V
```

出力値については、[373 ページの表 19-3](#) を参照してください。

2. Delegated Administrator の再設定中に指定した最上位管理者のユーザー ID とパスワードを使用して、Delegated Administrator コンソールにログインします。

```
http://hostName:port/da/DA/Login
```

ここで、`hostName:port` は、Delegated Administrator の再設定中に指定した値です。

## アップグレード後の作業

375 ページの「アップグレード手順 (Solaris)」および 377 ページの「アップグレード手順 (Linux)」で説明している手順以外に、アップグレード後の作業はありません。

## アップグレードのロールバック (Solaris)

Delegated Administrator のロールバックはサポートされていません。Directory Server のエントリなどのアップグレード処理中に加えられた変更や、Delegated Administrator を Web コンテナに配備したときの変更は、簡単にはバックアウトできません。

# Java ES Release 2 からの Delegated Administrator のアップグレード

Java ES 2004Q2 (Release 2) Delegated Administrator を Release 4 にアップグレードする手順は、次に説明するいくつかの例外を除き、Release 3 Delegated Administrator を Release 4 にアップグレードする手順と同じです。

## Delegated Administrator が依存するコンポーネントをアップグレードする

Java ES Release 2 Delegated Administrator を Release 4 にアップグレードするときのアップグレード前の作業は、Release 3 Delegated Administrator を Release 4 にアップグレードする手順とほぼ同じです。ただし、Delegated Administrator の依存関係のアップグレードに、すべての共有コンポーネント (42 ページの表 1-6 を参照) と、Delegated Administrator が依存するローカルにあるすべての製品コンポーネントの Release 4 へのアップグレードを含めるようにしてください。

Delegated Administrator の依存関係をアップグレードする場合は、Delegated Administrator をアップグレードする前に、そのすべてを次の順序でアップグレードするようにしてください。アップグレード済みのコンポーネントについては、その手順を省略できます。

1. **共有コンポーネント**: Java ES 共有コンポーネントを Release 4 にアップグレードする手順は、55 ページの第 2 章「Java ES 共有コンポーネントのアップグレード」で説明されています。

2. **Directory Server:** Directory Server を Release 4 にアップグレードする手順は、105 ページの第 4 章「[Directory Server と Administration Server](#)」で説明されています。
3. **Web コンテナソフトウェア:** Web Server または Application Server をアップグレードする手順は、それぞれ 150 ページの「[Java ES Release 2 からの Web Server のアップグレード](#)」および 189 ページの「[Java ES Release 2 からの Application Server のアップグレード](#)」で説明されています。
4. **Access Manager (Access Manager SDK):** Access Manager を Release 4 にアップグレードする手順は、227 ページの「[Java ES Release 2 からの Access Manager のアップグレード](#)」で説明されています。
5. **Directory Preparation Tool:** Directory Preparation Tool が Delegated Administrator と同じコンピュータ上に存在することはほとんどありませんが、Directory Preparation Tool をアップグレードし、それを Directory Server に対して実行する手順は、244 ページの「[Directory Preparation Tool の Java ES Release 2 からのアップグレード](#)」で説明されています。

## Release 2 Delegated Administrator のアップグレード

Delegated Administrator を Release 2 から Release 4 にアップグレードする手順は、Delegated Administrator ソフトウェアを配備している Web コンテナによって異なります。

### Release 2 Delegated Administrator のアップグレード : Web Server Web コンテナ

Release 4 にアップグレードされた Web Server Web コンテナに配備する際に Release 2 Delegated Administrator を Release 4 にアップグレードするには、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、374 ページの「[Release 3 の Delegated Administrator のアップグレード \(Solaris\)](#)」または 376 ページの「[Release 3 の Delegated Administrator のアップグレード \(Linux\)](#)」の手順に従います。

## Release 2 Delegated Administrator のアップグレード : Application Server Web コンテナ

Release 4 にアップグレードされた Application Server Web コンテナに配備する際に Release 2 Delegated Administrator を Release 4 にアップグレードするには、まず、手順に示されている Release 3 をすべて Release 2 に置き換えることを除き、[374 ページ](#)の「[Release 3 の Delegated Administrator のアップグレード \(Solaris\)](#)」または [376 ページ](#)の「[Release 3 の Delegated Administrator のアップグレード \(Linux\)](#)」の手順に従います。

Delegated Administrator が最初に配備されたときの Release 2 Application Server インスタンス (*instanceName*) は、Release 4 にアップグレードされると、アップグレード処理によって作成されたノードエージェントの下で移行されています。

この状況での Delegated Administrator のアップグレードでは、[375 ページ](#)の「[Delegated Administrator を再設定します。](#)」(Solaris) または [378 ページ](#) (Linux) の手順 6 を次のように変更する必要があります。

6. Delegated Administrator を再設定します。
  - a. *AccessManagerConfig-base/config/AMConfig.properties* ファイルを変更します。

次の行

```
com.sun.identity.webcontainer=IAS7.0
```

を次の行に置き換えます。

```
com.sun.identity.webcontainer=IAS8.1
```

- b. Delegated Administrator が配備されているアップグレードされた Application Server インスタンス (*instanceName*) が稼働していることを確認します。

Application Server インスタンスを起動するには、ドメイン管理サーバー (DAS) と、そのインスタンスを移行したノードエージェントを起動します。

```
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
```

上のコマンドおよび以降の手順では、次の規則が使用されています。

- *nodeAgentName* は、*hostName\_domainName* の形式をしています。
- デフォルトの *domainName* は domain1 です。
- デフォルトの *instanceName* は server1 です。

- c. Application Server インスタンス (*instanceName*) から `commcli Delegated Administrator Web` アプリケーションの配備を取り消します。

```
AppServer8-base/bin/asadmin undeploy --secure=false --user admin
--password password --target instanceName commcli
```

- d. Delegated Administrator 設定ユーティリティーを実行します。

```
DelegatedAdmin-base/sbin/config-commda
```

このユーティリティーの方法使用の詳細については、『Sun Java System Communications Services 6 2005Q4 Delegated Administrator 管理ガイド』(<http://docs.sun.com/doc/819-4103?l=ja>) の第 3 章「Delegated Administrator の設定」を参照してください。

次のパラメータを指定します。

- o Access Manager ホストおよびポートの入力を求められたら、DAS インスタンスのポート (デフォルトは 8080) を指定します。
- o Delegated Administrator コンソールおよび Delegated Administrator サーバーをどこに配備するかを求められた場合も、アップグレードされた Application Server インスタンス (*instanceName*) の情報ではなく、DAS インスタンスの情報 (デフォルトポートは 8080) を指定します。

これらのパラメータ値により、Delegated Administrator が DAS インスタンスに再配備されます。これは標準的な方法ではありませんが、正常に動作します。

- e. `classpath-prefix` と `classpath-suffix` の Access Manager クラスパス情報を、*instanceName* インスタンスの次の `domain.xml` ファイルから

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/
config/domain.xml
```

DAS (デフォルト名は `server`) の次の `domain.xml` ファイルにコピーします。

```
AppServer8Config-base/domains/domainName/config/domain.xml
```

ここで、デフォルトの *domainName* は `domain1` です。

この手順は、Application Server 管理コンソールからも実行できます。

- f. DAS を再起動します。

この DAS は、Delegated Administrator が再配備された先のサーバーインスタンスです。

```
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 リリースの内容

この付録では、各種の Java Enterprise System リリースの内容について説明します。  
この章で説明する内容は、次のとおりです。

- [383 ページの「Java ES 2003Q4 \(Release 1\)」](#)
- [385 ページの「Java ES 2004Q2 \(Release 2\)」](#)
- [388 ページの「Java ES 2005Q1 \(Release 3\)」](#)
- [392 ページの「Java ES 2005Q4 \(Release 4\)」](#)

## Java ES 2003Q4 (Release 1)

この節では、Java Enterprise System 2003Q4 の内容について説明します。

### Release 1 のインストーラで選択可能なコンポーネント

Sun Open Network Environment (Sun ONE) および Sun Cluster コンポーネント製品は、分散型のエンタープライズアプリケーションをサポートするために必要なインフラストラクチャーサービスを提供します。これらのコンポーネント製品は、次のとおりです。

- Sun Cluster 3.1 と Sun ONE 用 Sun Cluster エージェント
- 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

## Release 1 の共有コンポーネント

共有コンポーネントは、コンポーネント製品が依存するローカルサービスとテクノロジーサポートを提供します。コンポーネント製品をインストールすると、そのコンポーネントに必要な共有コンポーネントがインストールされていない場合に、Java ES インストーラは自動的にこれらのコンポーネントをインストールします。

Java Enterprise System には次の共有コンポーネントが含まれます。

- ANT (Jakarta ANT Java/XML ベースの構築ツール)
- 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 検索エンジン
- 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**           また、Perl も Application Server と Directory Server のシステムに必要ですが、Java ES 共有コンポーネントとして自動的にインストールされません。

---

## Java ES 2004Q2 (Release 2)

この節では、Java Enterprise System 2004Q2 の内容について説明します。

### Release 2 のインストーラで選択可能なコンポーネント

コンポーネント製品は、分散型のエンタープライズアプリケーションをサポートするために必要なインフラストラクチャーサービスを提供します。Java Enterprise System を特定のホストにインストールするときは、配備の全体的なアーキテクチャーに基づいて、そのホストにインストールするコンポーネント製品を選択します。

Java Enterprise System 2005Q4 には、次のコンポーネント製品が含まれます。

#### 通信サービスと共同作業サービス

- 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 サービスとアプリケーションサービス

- Sun Java System Application Server 7.0 Update 3 (Standard および Platform Edition)

- Sun Java System Web Server 6 2004Q2 Update 1 Service Pack 2
- Sun Java System Message Queue 3.5 SP1 (Platform および Enterprise Edition)

#### ディレクトリサービスとアイデンティティサービス

- Sun Java System Identity Server 6.2 2004Q2  
(Sun Java System Communications Services 6 2004Q2 ユーザー管理ユーティリティを含む)
- Sun Java System Directory Server 5 2004Q2
- Sun Java System Directory Proxy Server 5 2004Q2

#### 可用性サービス

- Sun Cluster 3.1 4/04 および Sun Java System 用 Sun Cluster エージェント

#### 管理サービス

- Sun Java System Administration Server 5 2004Q2
- Sun Remote Services Net Connect 3.5

Sun Cluster、Sun Cluster Agents、および Sun Remote Services Net Connect は Linux オペレーティングシステムでは使用できません。

## Release 2 の共有コンポーネント

共有コンポーネントは、コンポーネント製品が依存するローカルサービスとテクノロジーサポートを提供します。コンポーネント製品をインストールすると、そのコンポーネントに必要な共有コンポーネントがインストールされていない場合に、Java ES インストーラは自動的にこれらのコンポーネントをインストールします。

Java Enterprise System 2005Q4 には次の共有コンポーネントが含まれます。

- Ant (Jakarta ANT Java/XML ベースの構築ツール)
- 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 検索エンジン
- LDAP C Language SDK
- LDAP Java SDK
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- Perl LDAP (NSPERL を含む)
- 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 (Release 3)

この節では、Java Enterprise System 2005Q1 の内容について説明します。

## Release 3 のインストーラで選択可能なコンポーネント

Java ES インストーラのコンポーネント選択ページでは、選択可能なコンポーネントが提供するサービスごとにグループ化されています。また、次のリストは各コンポーネントとともにインストールされるサブコンポーネントを示します。

### 通信サービスと共同作業サービス

- Sun Java System Messaging Server 6 2005Q1
- Sun Java System Calendar Server 6 2005Q1
- Sun Java System Instant Messaging 7 2005Q1
  - Instant Messaging サーバーコア (サーバーおよびマルチプレクサソフトウェアを含む)
  - Instant Messaging リソース
  - Access Manager Instant Messaging サービス
- Sun Java System Calendar Server 6 2005Q1
- Sun Java System Portal Server Secure Remote Access 6 2005Q1
  - Secure Remote Access コア
  - ゲートウェイ
  - Netlet プロキシ
  - Rewriter プロキシ
- Sun Java System Communications Express 2005Q1
- Sun Java System Directory Preparation Tool

### Web サービスとアプリケーションサービス

- Sun Java System Application Server Enterprise Edition 8.1 2005Q1
  - ドメイン Administration Server
  - Application Server ノードエージェント
  - コマンド行管理ツール
  - 負荷分散プラグイン

設定時の選択により、Web Server または Apache Web Server のいずれかで使用できます。デフォルトは Web Server です。

- PointBase
- サンプルアプリケーション
- Sun Java System Web Server 6 2005Q1 Update 1 Service Pack 4
- Sun Java System Message Queue 3 2005Q1

#### ディレクトリサービスとアイデンティティサービス

- Sun Java System Access Manager 6.3 2005Q1

Portal Server および Messaging Server 用の Delegated Administrator プロビジョニングツールは、Access Manager とともに自動的にインストールされます。

- アイデンティティ管理およびポリシーサービスコア (Delegated Administrator ユーティリティを含む)
- Access Manager 管理コンソール
- 連携管理の共有ドメインサービス
- Access Manager SDK
- Sun Java System Directory Server 5 2005Q1
- Sun Java System Directory Proxy Server 5 2005Q1

#### 可用性サービス

- Sun Cluster 3.1 9/04
  - Sun Cluster コア
- Sun Java System 用 Sun Cluster エージェント
  - 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 (高可用性セッションストレージで使用)

#### 管理サービス

- Sun Java System Administration Server 5 2005Q1
- Sun<sup>SM</sup> Remote Services Net Connect 3.1.1

---

注	<p>Sun Cluster, Sun Cluster エージェント、および Sun Remote Services Net Connect は Solaris 10 または Linux オペレーティングシステムでは利用できません。</p> <p>Sun Remote Services Net Connect は Solaris x86 プラットフォームでは利用できません。</p>
---	--

---

## Release 3 の共有コンポーネント

共有コンポーネントは、選択可能なコンポーネントに対してローカルサービスとテクノロジーサポートを提供します。Java ES コンポーネントをインストールすると、そのコンポーネントに必要な共有コンポーネントがインストールされていない場合に、インストーラは自動的にこれらのコンポーネントをインストールします。

今回の Java ES のリリースには、次の共有コンポーネントが含まれています。

- Ant (Jakarta ANT Java/XML ベースの構築ツール)
- Apache SOAP (Simple Object Access Protocol) Runtime
- Berkeley Database
- 共通エージェントコンテナ
- 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 (NSPERL を含む)
- 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 (Solaris のみ)
- Sun Java Monitoring Framework
- Sun Java Web Console
- Tomcat Servlet JSP Container
- XML C Library (libxml)
- WSCL (Web services Common Library)

# Java ES 2005Q4 (Release 4)

この節では、Java Enterprise System 2005Q4 の内容について説明します。

## Release 4 のインストーラで選択可能なコンポーネント

Java ES インストーラのコンポーネント選択ページでは、選択可能なコンポーネントが提供するサービスごとにグループ化されています。また、次のリストは各コンポーネントとともにインストールされるサブコンポーネントを示します。

### 通信サービスと共同作業サービス

- 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 サーバーコア (サーバーおよびマルチプレクサソフトウェアを含む)
  - Instant Messaging リソース
  - Access Manager Instant Messaging サービス
- Sun Java System Portal Server 6.3.1 2005Q4
- Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4
  - Secure Remote Access コア
  - ゲートウェイ
  - Netlet プロキシ
  - Rewriter プロキシ
- 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 コンソールおよびユーティリティー
  - Delegated Administrator Server

### Web サービスとアプリケーションサービス

- Sun Java System Application Server Enterprise Edition 8.1 2005Q4
  - ドメイン Administration Server



- Application Server ノードエージェント
- コマンド行管理ツール
- 負荷分散プラグイン
 

設定時の選択により、Web Server または Apache Web Server のいずれかで使用できます。デフォルトは Web Server です。
- PointBase データベース
- サンプルアプリケーション
- 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

#### ディレクトリサービスとアイデンティティサービス

- Sun Java System Access Manager 7.0 2005Q4
  - アイデンティティ管理およびポリシーサービスコア
  - Access Manager 管理コンソール
  - 連携管理の共有ドメインサービス
  - Access Manager SDK
- Sun Java System Directory Server 5.2 2005Q4
- Sun Java System Directory Proxy Server 5.2 2005Q4

#### 可用性サービス

- Sun Cluster 3.1 8/05
  - Sun Cluster コア
  - Sun Java System 用 Sun Cluster エージェント
    - 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

#### 管理サービス

- Sun Java System Administration Server 5.2 2005Q4

---

#### 注

Sun Cluster, Sun Cluster エージェント、および Sun Remote Services Net Connect は Solaris 10 または Linux オペレーティングシステムでは利用できません。

Sun Remote Services Net Connect は Solaris x86 プラットフォームでは利用できません。

---

## Release 4 の共有コンポーネント

共有コンポーネントは、選択可能なコンポーネントに対してローカルサービスとテクノロジーサポートを提供します。Java ES コンポーネントをインストールすると、そのコンポーネントに必要な共有コンポーネントがインストールされていない場合に、インストーラは自動的にこれらのコンポーネントをインストールします。

今回の Java ES のリリースには、次の共有コンポーネントが含まれています。

- ANT (Jakarta ANT Java/XML ベースの構築ツール)
- ACL (Apache Common Logging)
- BDB (Berkeley Database)
- CAC (共通エージェントコンテナ)
- Derby Database
- ICU (International Components for Unicode)
- IM-SDK (Instant Messenger SDK)
- J2SEa (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
- JCAPAPI (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) コア
- 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、Solaris のみ)
- MFWK (Java ES Monitoring Framework)
- SJWC (Sun Java Web Console)
- WSCL (Web services Common Library)



# 索引

## A

### Access Manager

- 共有コンポーネントに対する依存関係, 42
- サブコンポーネント, 393
- 製品コンポーネントの依存関係, 46
- 略称, 28

*AccessManager-base* パス, 205

*AccessManagerConfig-base* パス, 205

### ACL 共有コンポーネント

- Release 2 からのアップグレード, 62
- Release 3 からのアップグレード, 60
- 依存関係の表, 42
- パッケージバージョン, 73, 77
- フルネーム, 29

### Administration Server

- 共有コンポーネントに対する依存関係, 42
- 製品コンポーネントの依存関係, 46
- パッチ, 113, 116
- 略称, 28

*amconfig* スクリプト (Access Manager), 219

*ampre70upgrade* スクリプト (Access Manager), 216

*amupgrade* スクリプト (Access Manager), 221

Ant (Jakarta ANT Java/XML ベースの構築ツール)、  
「ANT」を参照

### ANT 共有コンポーネント

- Release 2 からのアップグレード, 62
- Release 3 からのアップグレード, 60
- 依存関係の表, 42
- パッケージバージョン, 73, 77

フルネーム, 29

Apache Common Logging、「ACL」を参照

### Application Server

- インストール後の設定, 194
- 共有コンポーネントに対する依存関係, 42
- サブコンポーネント, 392
- 製品コンポーネントの依存関係, 47
- 略称, 28

*AppServer7-base* のパス, 177

*AppServer7Config-base* のパス, 177

*AppServer8-base* のパス, 177

*AppServer8Config-base* のパス, 177

*asant* スクリプト (Application Server), 185

*asupgrade* スクリプト (Application Server), 194

## B

### BDB 共有コンポーネント

- Release 2 からのアップグレード, 62
- Release 3 からのアップグレード, 60
- 依存関係の表, 42
- パッケージバージョン, 73, 77
- フルネーム, 29

BEA WebLogic Server、Web コンテナとしての、46,  
50

Berkeley Database、「BDB」を参照

## C

### CAC 共有コンポーネント

Release 2 からのアップグレード, 62

Release 3 からのアップグレード, 60

依存関係の表, 42

パッケージバージョン, 73, 77

フルネーム, 29

### Calendar Server

共有コンポーネントに対する依存関係, 42

製品コンポーネントの依存関係, 47

略称, 28

### CalendarServer-base パス, 265

### comm\_dssetup.pl (Directory Preparation Tool) スクリプト, 241

### CommsExpress-base パス, 277

### Communications Express

共有コンポーネントに対する依存関係, 42

製品コンポーネントの依存関係, 48

略称, 28

## D

### db2bak ユーティリティ, 111, 132, 214

### DelegatedAdmin-base パス, 367

### Delegated Administrator

共有コンポーネントに対する依存関係, 42

製品コンポーネントの依存関係, 48

略称, 28

### Derby Database 共有コンポーネント, 394

### Directory Proxy Server

共有コンポーネントに対する依存関係, 42

製品コンポーネントの依存関係, 48

パッチ, 113, 116

略称, 28

### Directory Server

共有コンポーネントに対する依存関係, 42

製品コンポーネントの依存関係, 49

パッチ, 113, 116

略称, 28

### DirPrepTool-base パス, 235

## H

### HADB-base のパス, 169

### High Availability Session Store

略称, 28

## I

### ICU 共有コンポーネント

Release 2 からのアップグレード, 62

Release 3 からのアップグレード, 60

依存関係の表, 42

パッケージバージョン, 77

フルネーム, 29

### IM-SDK 共有コンポーネント

Release 2 からのアップグレード, 62

Release 3 からのアップグレード, 60

依存関係の表, 42

パッケージバージョン, 73, 77

フルネーム, 29

### install\_cluster (パッチクラスタ) スクリプト, 66

### Instant Messaging

共有コンポーネントに対する依存関係, 42

製品コンポーネントの依存関係, 49

略称, 28

### InstantMessaging-base パス, 293

Instant Messenger SDK、「IM-SDK」を参照

International Components for Unicode、「ICU」を参照

## J

### J2SE 共有コンポーネント

Release 2 からのアップグレード, 62

Release 3 からのアップグレード, 60

依存関係の表, 43

パッケージバージョン, 74, 77

フルネーム, 29

### JAF 共有コンポーネント

- Release 2 からのアップグレード, 62
- Release 3 からのアップグレード, 60
- 依存関係の表, 43
- フルネーム, 29
- JATO 共有コンポーネント
  - Release 2 からのアップグレード, 62
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - パッケージバージョン, 74, 77
  - フルネーム, 29
- Java 2 Platform, Standard Edition, 「J2SE」を参照
- Java API for XML-based Remote Procedure Call, 「JAX-RPC」を参照
- Java API for XML Processing, 「JAXP」を参照
- Java API for XML Registries, 「JAXR」を参照
- Java Architecture for XML Binding, 「JAXB」を参照
- JavaBeans Activation Framework, 「JAF」を参照
- Java Calendar API, 「JCAPI」を参照
- Java Dynamic Management Kit, 「JDMK」を参照
- Java ES 2003Q4 (Release 1)
  - 共有コンポーネント, 384
  - 製品コンポーネント, 383
- Java ES 2004Q2 (Release 2)
  - 共有コンポーネント, 386
  - 製品コンポーネント, 385
- Java ES 2005Q1 (Release 3)
  - 共有コンポーネント, 390
  - 製品コンポーネント, 388
- Java ES 2005Q4 (Release 4)
  - 共有コンポーネント, 394
  - 製品コンポーネント, 392
- Java ES Monitoring Framework, 「MFWK」を参照
- Java ES パッチクラスタスクリプト, 67, 84
- JavaHelp 共有コンポーネント
  - Release 2 からのアップグレード, 62
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - パッケージバージョン, 74, 77
- JavaMail 共有コンポーネント
  - Release 2 からのアップグレード, 62
  - Release 3 からのアップグレード, 60
- 依存関係の表, 43
- Java Security Services, 「JSS」を参照
- Java Studio Enterprise Web Application Framework, 「JATO」を参照
- JAXB 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - フルネーム, 29
- JAXP 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - フルネーム, 29
- JAX-RPC 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - フルネーム, 29
- JAXR 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - フルネーム, 29
- JCAPI 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - フルネーム, 29
- JDMK 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43
  - フルネーム, 29
- JHELP 共有コンポーネント, 「JavaHelp」を参照
- JMAIL 共有コンポーネント, 「JavaMail」を参照
- JSP ファイル, カスタマイズ, 208, 214, 218
- JSS 共有コンポーネント
  - Release 2 からのアップグレード, 63
  - Release 3 からのアップグレード, 60
  - 依存関係の表, 43

パッケージバージョン, 77  
フルネーム, 29

## K

KT Search Engine、「KTSE」を参照

KTSE 共有コンポーネント

Release 2 からのアップグレード, 63  
Release 3 からのアップグレード, 60  
依存関係の表, 43  
パッケージバージョン, 78  
フルネーム, 29

## L

LDAP C Language SDK 共有コンポーネント, 395

LDAP C SDK 共有コンポーネント

Release 2 からのアップグレード, 63  
Release 3 からのアップグレード, 60  
依存関係の表, 43  
フルネーム, 29

LDAP J SDK 共有コンポーネント, 395

Release 2 からのアップグレード, 63  
Release 3 からのアップグレード, 60  
依存関係の表, 43  
パッケージバージョン, 78  
フルネーム, 29

## M

MA Core 共有コンポーネント

Release 2 からのアップグレード, 63  
Release 3 からのアップグレード, 60  
依存関係の表, 43  
フルネーム, 30

Message Queue

共有コンポーネントに対する依存関係, 42  
製品コンポーネントの依存関係, 49

略称, 28

Messaging Server

共有コンポーネントに対する依存関係, 42  
略称, 28

*MessagingServer-base* パス, 247

MFWK 共有コンポーネント

Release 3 からのアップグレード, 61  
依存関係の表, 43  
パッケージバージョン, 78  
フルネーム, 30

Mobile Access コア、「MA コア」を参照

mqupgrade スクリプト (Message Queue), 161, 165

multiserverinstance (Portal Server) スクリプト, 335

## N

Netscape Portable Runtime、「NSPR」を参照

Network Security Services、「NSS」を参照

NSPR 共有コンポーネント

Release 2 からのアップグレード, 63  
Release 3 からのアップグレード, 61  
依存関係の表, 43  
パッケージバージョン, 78  
フルネーム, 30

NSS 共有コンポーネント

Release 2 からのアップグレード, 63  
Release 3 からのアップグレード, 61  
依存関係の表, 43  
パッケージバージョン, 78  
フルネーム, 30

## P

Portal Server

Secure Remote Access

略称, 28  
共有コンポーネントに対する依存関係, 42  
製品コンポーネントの依存関係, 50  
略称, 28



*PortalServer-base* パス , 313  
*PortalServerConfig-base* パス , 313  
Portal Server Secure Remote Access  
共有コンポーネントに対する依存関係 , 42  
製品コンポーネントの依存関係 , 50  
postInstall スクリプト (Application Server), 194

## S

SAAJ 共有コンポーネント  
Release 2 からのアップグレード , 63  
Release 3 からのアップグレード , 60  
依存関係の表 , 43  
フルネーム , 30  
SAML, 223, 232  
SASL 共有コンポーネント  
Release 2 からのアップグレード , 63  
Release 3 からのアップグレード , 61  
依存関係の表 , 43  
パッケージバージョン , 78  
フルネーム , 30  
Security Assertion Markup Language、「SAML」を参照  
SEDC 共有コンポーネント  
Release 2 からのアップグレード , 63  
Release 3 からのアップグレード , 61  
依存関係の表 , 43  
パッケージバージョン , 74  
フルネーム , 30  
*serverRoot* のパス (Directory Server), 105  
Service Registry  
略称 , 28  
Simple Authentication and Security Layer、「SASL」を参照  
SJWC 共有コンポーネント  
依存関係の表 , 43  
パッケージバージョン , 78  
フルネーム , 30  
SOAP Runtime with Attachments API for Java、「SAAJ」を参照

srapNetlet サービス , 361, 364  
srapProxylet サービス , 361, 364  
Sun Cluster  
共有コンポーネントに対する依存関係 , 42  
製品コンポーネントの依存関係 , 50  
略称 , 28  
Sun Cluster エージェント , 393  
Sun Explorer Data Collector、「SEDC」を参照  
Sun Java Web Console、「SJWC」を参照

## U

update スクリプト (Portal Server), 324  
update スクリプト (Portal Server SRA), 353  
upgradeportalrpms スクリプト (Portal Server), 339, 341

## W

Web Proxy Server  
共有コンポーネントに対する依存関係 , 42  
製品コンポーネントの依存関係 , 50  
略称 , 28  
*WebProxyServer-base* のパス , 197  
Web Server  
共有コンポーネントに対する依存関係 , 42  
製品コンポーネントの依存関係 , 50  
略称 , 28  
*WebServer-base* パス , 141  
Web services Common Library、「WSCL」を参照  
WSCL 共有コンポーネント  
Release 2 からのアップグレード , 63  
Release 3 からのアップグレード , 60  
依存関係の表 , 43  
フルネーム , 30

## い

### 依存関係

- 共有コンポーネント間, [44](#)
- 製品コンポーネント間, [46](#)
- 製品コンポーネント、共有コンポーネントに対する, [42](#)

## き

### 記号の表記規則, [20](#)

#### 規則

- 記号, [20](#)
- 表記, [20](#)

共通エージェントコンテナ、「CAC」を参照

### 共有コンポーネント

- Java ES 2003Q4 (Release 1), [384](#)
- Java ES 2004Q2 (Release 2), [386](#)
- Java ES 2005Q1 (Release 3), [390](#)
- Java ES 2005Q4 (Release 4), [394](#)
- 依存する製品コンポーネント, [42](#)
- 相互依存関係, [44](#)

## く

### クラスタのアップグレード, [193](#)

## さ

### サービス

- srapNetlet, [361, 364](#)
- srapProxylet, [361, 364](#)

## す

### スクリプト

- amconfig (Access Manager), [219](#)
- ampre70upgrade (Access Manager), [216](#)

- amupgrade (Access Manager), [221](#)
- asant (Application Server), [185](#)
- asupgrade (Application Server), [194](#)
- comm\_dssetup.pl (Directory Preparation Tool), [241](#)
- install\_cluster (パッチクラスタ), [66](#)
- Java ES パッチクラスタ, [67, 84](#)
- mqmigrate (Message Queue), [165](#)
- mqupgrade (Message Queue), [161](#)
- multiserverinstance (Portal Server), [335](#)
- postInstall (Application Server), [194](#)
- update (Portal Server), [324](#)
- update (Portal Server SRA), [353](#)
- upgradeportalrpms (Portal Server), [339, 341](#)

## せ

### 製品コンポーネント

- Java ES 2003Q4 (Release 1), [383](#)
- Java ES 2004Q2 (Release 2), [385](#)
- Java ES 2005Q1 (Release 3), [388](#)
- Java ES 2005Q4 (Release 4), [392](#)
- 共有コンポーネントに対する依存関係, [42](#)
- 相互依存関係, [46](#)

## は

### パッチクラスタ、Solaris OS, [67](#)

## ひ

### 表記上の規則, [20](#)