



Sun Java™ System

適用於 UNIX 的
Sun Java Enterprise System 5 升級指南

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

文件號碼：820-0891
2007 年 6 月

Copyright © 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 版權所有。

Sun Microsystems, Inc. 對於本文件所述產品中涉及的技术擁有智慧財產權。需特別指出的是，這些智慧財產權可能包含 <http://www.sun.com/patents> 上所列的一項或多項美國專利，以及在美國與其他國家 / 地區擁有的一項或多項其他專利或申請中專利，但並不以此為限。

本產品包含 SUN MICROSYSTEMS, INC. 的機密資訊和商業秘密。未經 SUN MICROSYSTEMS, INC. 事先明確的書面許可，不得使用、透露或複製。

美國政府權利 - 商業軟體。政府使用者均應遵守 Sun Microsystems, Inc. 的標準授權合約和 FAR 及其增補文件中的適用條款。

本發行物可能包含由協力廠商開發的材料。

本產品中的某些部分可能源自加州大學授權的 Berkeley BSD 系統的開發成果。UNIX 是在美國及其他國家 / 地區的註冊商標，已獲得 X/Open Company, Ltd. 專屬授權。

Sun、Sun Microsystems、Sun 標誌、Java、Solaris、JDK、Java Naming and Directory Interface、JavaMail、JavaHelp、J2SE、iPlanet、Duke 標誌、Java 咖啡杯標誌、Solaris 標誌、SunTone Certified 標誌與 Sun ONE 標誌是 Sun Microsystems, Inc. 在美國及其他國家 / 地區的商標或註冊商標。

所有 SPARC 商標都是 SPARC International, Inc. 在美國及其他國家 / 地區的商標或註冊商標，經授權後使用。凡具有 SPARC 商標的產品都是採用 Sun Microsystems, Inc. 所開發的架構。

Legato 和 Legato 標誌是註冊商標，它們和 Legato NetWorker 都是 Legato Systems, Inc. 的商標或註冊商標。Netscape Communications Corp 標誌是 Netscape Communications Corporation 的商標或註冊商標。

OPEN LOOK 與 Sun(TM) Graphical User Interface (Sun 圖形化使用者介面) 都是由 Sun Microsystems, Inc. 為其使用者與授權者所開發的技術。Sun 感謝 Xerox 公司在研究和開發視覺化或圖形化使用者介面之概念上，為電腦工業所做的開拓性貢獻。Sun 已向 Xerox 公司取得 Xerox 圖形化使用者介面之非獨占性授權，該授權亦適用於使用 OPEN LOOK GUI 並遵守 Sun 書面授權合約的 Sun 公司授權者。

本服務手冊所涵蓋的產品和包含的資訊受到美國出口控制法規的控制，並可能受到其他國家 / 地區進出口法規的制約。嚴禁核子武器、飛彈、生化武器或海上核動力裝備等最終用途或一般使用者直接或間接使用本產品。嚴禁向受到美國禁運的國家 / 地區或美國出口除外清單 (包括但不僅限於被拒人清單和特別指定的國家 / 地區清單) 上標識的實體出口或再出口本產品。

本文件以其「原狀」提供，對任何明示或暗示的條件、陳述或擔保，包括對適銷性、特殊用途的適用性或非侵權性的暗示保證，均不承擔任何責任，除非此免責聲明的適用範圍在法律上無效。

目錄

表目錄	15
前言	19
本書適用對象	20
本書中所使用的慣例	20
管理介面	20
印刷排版慣例	21
符號	21
Shell 提示符號	22
相關文件	22
Java ES 文件集中的書籍	22
搜尋 Sun 產品文件	24
線上存取 Sun 資源	24
連絡 Sun 技術支援	24
協力廠商網站參照	25
Sun 歡迎您提出寶貴意見	25
第 1 章 升級規劃	27
Java ES 5 元件	28
發行版本 5 產品元件	28
發行版本 5 共用元件	29
Java ES 升級技術	31
產品元件升級	31
產品元件升級方法	31
用於每個產品元件的升級方法	33
共用元件升級	34
升級程序	35

升級規劃注意事項	36
升級相依性	36
支援的升級路徑與策略	37
選擇性升級或整體升級	38
多重實例升級	39
作業系統注意事項	40
必要的作業系統修補程式	40
雙重升級：Java ES 與作業系統軟體	40
作業系統升級	42
Solaris 10 多區域環境	42
Java ES 元件相依性	43
對共用元件的相依性	43
對產品元件的相依性	47
升級順序指導原則	51
特殊情况	53
選擇性升級：未升級 Application Server	53
將 Portal Server 臨時功能發行版本 (IFR) 7.0 升級到 Java ES 5	54
Java ES 5 升級與 Solaris 10 區域	54
Java ES 安裝程式中的區域支援	54
產品元件的升級	55
同步化所有共用元件	55
建議的升級措施	56
特殊情况或例外情況	57
產品元件特殊情况	57
共用元件特殊情况	57
第 2 章 升級 Java ES 共用元件	59
共用元件升級簡介	60
一般注意事項	60
同步化共用元件	60
同步化所有共用元件	61
Solaris 10 區域注意事項	62
升級共用元件的方法	62
共用元件升級程序	63
特殊情况升級程序	63
Java SE 升級程序	63
檢查 Java SE 符號連結	65
驗證目前的 Java SE 版本	65
Common Agent Container 升級程序	66

第 3 章 Sun Cluster 軟體	69
Sun Cluster 軟體升級簡介	70
關於發行版本 5 Sun Cluster 軟體	70
Sun Cluster 軟體升級資訊指南	71
Sun Cluster 資料	72
Sun Cluster 升級策略	72
相容性問題	72
Sun Cluster 相依性	72
雙重升級	73
將 Sun Cluster 軟體升級到 Java ES 發行版本 5	74
簡介	74
Sun Cluster 升級	75
升級前作業	75
升級 Sun Cluster 軟體	78
驗證升級	79
升級後作業	79
回復升級	79
第 4 章 Sun Cluster Geographic Edition	81
Sun Cluster Geographic Edition 升級簡介	82
關於 Java ES 發行版本 5	82
Java ES 發行版本 5 升級資訊指南	82
Sun Cluster Geographic Edition 資料	83
Sun Cluster Geographic Edition 升級策略	83
相容性問題	83
相依性	83
雙重升級	84
從版本 3.1 8/05 升級 Sun Cluster Geographic Edition	85
簡介	85
版本 3.1 8/05 Sun Cluster Geographic Edition 升級	86
升級前作業	86
升級 3.1 8/05 Sun Cluster Geographic Edition	88
驗證升級	90
升級後作業	91
回復升級	91

第 5 章 Directory Server	93
Directory Server 升級簡介	94
關於 Java ES 發行版本 5	94
Java ES 發行版本 5 升級資訊指南	94
Directory Server 資料	95
Directory Server 升級策略	96
相容性問題	96
相依性	97
雙重升級	97
從 Java ES 發行版本 4 升級 Directory Server	99
簡介	99
升級發行版本 4 Directory Server	100
升級前作業	100
升級發行版本 4 Directory Server	101
驗證升級	105
升級後作業	106
回復升級	106
多重實例升級	106
漸進式升級 Directory Server 複製	106
將 Directory Server 做為資料服務進行升級	106
從 Java ES 發行版本 3 升級 Directory Server	107
從 Java ES 發行版本 2 升級 Directory Server	108
第 6 章 Directory Proxy Server	109
Directory Proxy Server 升級簡介	110
關於 Java ES 發行版本 5	110
Java ES 發行版本 5 升級資訊指南	110
Directory Proxy Server 資料	111
Directory Proxy Server 升級策略	112
相容性問題	112
相依性	112
雙重升級	113
從 Java ES 發行版本 4 升級 Directory Proxy Server	114
簡介	114
升級發行版本 4 Directory Proxy Server	115
升級前作業	115
升級發行版本 4 Directory Proxy Server	116
驗證升級	118
升級後作業	118
回復升級	119
多重實例升級	119
從 Java ES 發行版本 3 升級 Directory Proxy Server	120
從 Java ES 發行版本 2 升級 Directory Proxy Server	121

第 7 章 Web Server	123
Web Server 升級簡介	124
關於 Java ES 發行版本 5 Web Server	124
Web Server 升級資訊指南	124
Web Server 資料	125
Web Server 升級策略	126
相容性問題	126
Web Server 相依性	126
雙重升級	127
從 Java ES 發行版本 4 升級 Web Server	128
簡介	128
升級發行版本 4 Web Server	128
升級前作業	129
升級發行版本 4 Web Server	130
驗證升級	138
升級後作業	139
回復升級	142
從 Java ES 發行版本 3 升級 Web Server	143
從 Java ES 發行版本 2 升級 Web Server	144
第 8 章 Java DB	145
Java DB 升級簡介	146
關於 Java ES 發行版本 5	146
Java ES 發行版本 5 升級資訊指南	146
Java DB 資料	147
Java DB 升級策略	147
相容性問題	147
相依性	147
雙重升級	147
從 Java ES 發行版本 4 升級 Java DB	149
簡介	149
升級發行版本 4 Java DB	149
升級前作業	150
升級發行版本 4 Java DB	150
驗證升級	152
升級後作業	152
回復升級	152
多重實例升級	153

第 9 章 High Availability Session Store	155
HADB 升級簡介	156
關於 Java ES 發行版本 5 HADB	156
HADB 升級資訊指南	156
HADB 資料	157
HADB 升級策略	157
相容性問題	157
HADB 相依性	157
雙重升級	157
從 Java ES 發行版本 4 升級 HADB	158
簡介	158
升級發行版本 4 HADB	159
升級前作業	159
升級發行版本 4 HADB	161
驗證升級	163
升級後作業	164
回復升級	164
從 Java ES 發行版本 3 升級 HADB	165
第 10 章 Message Queue	167
Message Queue 升級簡介	168
關於 Java ES 發行版本 5 Message Queue	168
Message Queue 升級資訊指南	169
Message Queue 資料	170
Message Queue 升級策略	171
相容性問題	171
Message Queue 相依性	171
雙重升級	172
從 Java ES 發行版本 4 升級 Message Queue	173
簡介	173
升級發行版本 4 Message Queue	174
升級前作業	174
升級發行版本 4 Message Queue	175
驗證 Message Queue 升級	176
升級後作業	176
回復升級	176
多重實例升級	178
從 Java ES 發行版本 3 升級 Message Queue	179
從 Java ES 發行版本 2 升級 Message Queue	180
簡介	180
發行版本 2 相容性問題	181
協定相容性	181
代理程式相容性	181

管理的物件相容性	182
管理工具相容性	182
用戶端相容性	182
升級發行版本 2 Message Queue	183
升級前作業	183
升級發行版本 2 Message Queue (Solaris)	184
升級發行版本 2 Message Queue (Linux)	185
安裝相容性套裝軟體 (Linux)	187
驗證 Message Queue 升級	187
升級後作業	187
回復升級	188
多重實例升級	188
第 11 章 Application Server	189
Application Server 升級簡介	191
關於 Java ES 發行版本 5 Application Server	191
Application Server 升級資訊指南	191
Application Server 資料	194
Application Server 升級策略	194
相容性問題	195
Application Server 相依性	195
雙重升級	196
從 Java ES 發行版本 4 升級 Application Server	197
簡介	197
升級發行版本 4 Application Server	198
升級前作業	198
升級發行版本 4 Application Server	199
驗證升級	201
升級後作業	201
回復升級	201
從 Java ES 發行版本 3 升級 Application Server	202
從 Java ES 發行版本 2 升級 Application Server	203
簡介	203
升級發行版本 2 Application Server	204
升級前作業	204
升級發行版本 2 Application Server	206
驗證升級	210
升級後作業	211
回復升級	211
在 Solaris 10 多重區域環境中升級 Solaris 隨附的 Application Server	212

第 12 章 Service Registry	215
Service Registry 升級簡介	216
關於 Java ES 發行版本 5	216
Java ES 發行版本 5 升級資訊指南	216
Service Registry 資料	217
Service Registry 升級策略	217
相容性問題	217
相依性	217
雙重升級	218
從 Java ES 發行版本 4 升級 Service Registry	219
簡介	219
升級發行版本 4 Service Registry	219
升級前作業	220
升級發行版本 4 Service Registry	221
驗證升級	224
升級後作業	224
回復升級	224
多重實例升級	225
第 13 章 Web Proxy Server	227
Web Proxy Server 升級簡介	228
關於 Java ES 發行版本 5 Web Proxy Server	228
Web Proxy Server 升級資訊指南	228
Web Proxy Server 資料	229
Web Proxy Server 升級策略	229
相容性問題	229
Web Proxy Server 相依性	229
雙重升級	230
從 Java ES 發行版本 4 升級 Web Proxy Server	231
簡介	231
升級發行版本 4 Web Proxy Server	231
升級前作業	232
升級發行版本 4 Web Proxy Server (Solaris)	233
升級發行版本 4 Web Proxy Server (Linux)	234
驗證升級	235
升級後作業	235
回復升級 (Solaris)	236
從版本 3.6 升級 Web Proxy Server	237
簡介	237
版本 3.6 Web Proxy Server 升級	237
升級前作業	238
升級版本 3.6 Web Proxy Server	238
驗證升級	240
升級後作業	240
回復版本 3.6 升級	240

第 14 章 Access Manager	241
Access Manager 升級簡介	242
關於 Java ES 發行版本 5 Access Manager	242
Access Manager 升級資訊指南	243
Access Manager 資料	244
Access Manager 升級策略	245
相容性問題	245
Access Manager 相依性	246
Web 容器升級方案	246
雙重升級	247
從 Java ES 發行版本 4 升級 Access Manager	248
簡介	248
完整升級發行版本 4 Access Manager	249
升級前作業	249
升級發行版本 4 Access Manager	252
驗證 Access Manager 升級	261
升級後作業	262
回復升級	263
多重實例升級	263
僅升級發行版本 4 Access Manager SDK	265
升級前作業	265
升級發行版本 4 Access Manager SDK	266
驗證 Access Manager SDK 升級	266
升級回復	267
從 Java ES 發行版本 3 升級 Access Manager	268
升級發行版本 3 Access Manager	268
多重實例升級	268
從 Java ES 發行版本 2 升級 Access Manager	271
升級前作業	271
升級 Access Manager 相依元件	272
升級目錄模式	272
為目錄重新編製索引	272
升級發行版本 2 Access Manager	273
升級發行版本 2 Access Manager : Web Server Web 容器	273
升級發行版本 2 Access Manager : Application Server Web 容器	273
驗證 Access Manager 升級	283
升級後作業	283
回復升級	283
多重實例升級	284

第 15 章 Portal Server	285
Portal Server 升級簡介	287
關於 Java ES 發行版本 5 Portal Server	287
Portal Server 升級資訊指南	287
Portal Server 資料	288
Portal Server 升級策略	289
相容性問題	289
Portal Server 相依性	290
選擇性的升級問題	291
Web 容器升級方案	291
雙重升級	292
從 Java ES 發行版本 4 升級 Portal Server	293
簡介	293
升級發行版本 4 Portal Server	294
發行版本 4 升級前作業	294
升級發行版本 4 Portal Server (Solaris)	299
升級發行版本 4 Portal Server (Linux)	303
驗證升級	306
發行版本 4 升級後作業	306
回復升級 (Solaris)	314
回復升級 (Linux)	316
多重實例升級	316
從 Java ES 發行版本 3 升級 Portal Server	320
發行版本 3 升級前作業：升級 Portal Server 相依元件	320
升級發行版本 3 Portal Server	321
發行版本 3 升級後作業	321
多重實例升級	322
從 Java ES 發行版本 2 升級 Portal Server	323
發行版本 2 升級前作業	323
升級發行版本 2 Portal Server	324
升級發行版本 2 Portal Server：Web Server Web 容器	324
升級發行版本 2 Portal Server：Application Server Web 容器	324
發行版本 2 升級後作業	328
單次登入配置	328
啟用 URLScrapper 通道	332
刪除 Gateway 服務項目	332
多重實例升級	332

從臨時功能發行版本 7.0 升級 Portal Server	333
Portal Server IFR 升級簡介	333
Portal Server IFR 7.0 升級	334
IFR 7 升級前作業	334
升級 Portal Server IFR 7.0 (Solaris)	335
升級 IFR 7 Portal Server (Linux)	338
驗證升級	340
IFR 7 升級後作業	341
回復升級 (Solaris)	348
回復升級 (Linux)	349
多重實例升級	349
第 16 章 Portal Server Secure Remote Access	351
Portal Server Secure Remote Access 升級簡介	353
關於 Java ES 發行版本 5 Portal Server Secure Remote Access	353
Portal Server Secure Remote Access 升級資訊指南	354
Portal Server Secure Remote Access 資料	355
Portal Server Secure Remote Access 升級策略	355
相容性問題	355
Portal Server Secure Remote Access 相依性	356
選擇性的升級問題	356
雙重升級	357
從 Java ES 發行版本 4 升級 Portal Server Secure Remote Access	358
簡介	358
升級發行版本 4 Portal Server Secure Remote Access	359
發行版本 4 升級前作業	360
升級發行版本 4 Portal Server Secure Remote Access (Solaris)	363
升級發行版本 4 Portal Server Secure Remote Access (Linux)	365
驗證升級	367
發行版本 4 升級後作業	367
回復升級 (Solaris)	369
回復升級 (Linux)	370
多重實例升級	370
從 Java ES 發行版本 3 升級 Portal Server Secure Remote Access	375
升級 Portal Server Secure Remote Access 相依元件	375
升級發行版本 3 Portal Server Secure Remote Access	376
多重實例升級	376

從 Java ES 發行版本 2 升級 Portal Server Secure Remote Access	377
發行版本 2 升級前作業	377
升級 Portal Server Secure Remote Access 相依元件	377
刪除 Gateway 服務項目	378
升級發行版本 2 Portal Server Secure Remote Access	378
發行版本 2 升級後作業	379
為 Proxylet 服務設定 Portal Server 網域	379
多重實例升級	379
從臨時功能發行版本 7.0 升級 Portal Server Secure Remote Access	380
簡介	380
Portal Server Secure Remote Access IFR 7.0 升級	381
升級前作業	381
升級 Portal Server Secure Remote Access IFR 7.0 (Solaris)	381
升級 Portal Server Secure Remote Access IFR 7.0 (Linux)	383
驗證升級	385
升級後作業	385
回復升級 (Solaris)	385
回復升級 (Linux)	386
多重實例升級	386
附錄 A Java Enterprise System 發行版本內容	387
Java ES 2003Q4 (發行版本 1)	388
發行版本 1 安裝程式 - 可選取的元件	388
發行版本 1 共用元件	389
Java ES 2004Q2 (發行版本 2)	390
發行版本 2 安裝程式 - 可選取的元件	390
發行版本 2 共用元件	391
Java ES 2005Q1 (發行版本 3)	393
發行版本 3 安裝程式 - 可選取的元件	393
發行版本 3 共用元件	396
Java ES 2005Q4 (發行版本 4)	398
發行版本 4 安裝程式 - 可選取的元件	398
發行版本 4 共用元件	401
Java ES 5 (發行版本 5)	403
發行版本 5 安裝程式 - 可選取的元件	403
發行版本 5 共用元件	406
索引	409

表目錄

表 1	印刷排版慣例	21
表 2	符號慣例	21
表 3	Shell 提示符號	22
表 4	Java Enterprise System 文件	23
表 1-1	Java ES 5 產品元件	28
表 1-2	Java ES 5 共用元件	29
表 1-3	Java ES 產品元件升級方法	33
表 1-4	升級程序中的階段	35
表 1-5	至 Java ES 5 (發行版本 5) 的升級路徑	37
表 1-6	選擇性升級與整體升級之比較	39
表 1-7	Java ES/作業系統支援矩陣	40
表 1-8	Java ES 5 產品元件的雙重升級支援	41
表 1-9	Java ES 5 (發行版本 5) 產品元件的共用元件相依性	43
表 1-10	Java ES 產品元件相依性	47
表 1-11	Application Server 未升級到發行版本 5 時所需的修補程式	53
表 2-1	依平台而定的 Java SE 的符號連結	64
表 2-2	Java SE 版本驗證輸出	65
表 2-3	Common Agent Container 目錄路徑	66
表 3-1	至 Java ES 5 (發行版本 5) 的升級路徑： Sun Cluster 3.1 8/05 軟體	71
表 3-2	Sun Cluster 資料用途	72
表 3-3	Sun Cluster 版本驗證輸出	76
表 4-1	Sun Cluster Geographic Edition 目錄路徑	81
表 4-2	至 Java ES 5 (發行版本 5) 的升級路徑： Sun Cluster Geographic Edition 3.1 2006Q4	82

表 4-3	Sun Cluster Geographic Edition 資料用途	83
表 4-4	Sun Cluster Geographic Edition 版本驗證輸出	86
表 5-1	Directory Server 目錄路徑	93
表 5-2	至 Java ES 5 (發行版本 5) 的升級路徑：Directory Server 6.0	94
表 5-3	Directory Server 資料用途	96
表 5-4	Directory Server 版本驗證輸出	100
表 6-1	Directory Proxy Server 目錄路徑	109
表 6-2	至 Java ES 5 (發行版本 5) 的升級路徑：Directory Proxy Server 6.0	110
表 6-3	Directory Proxy Server 資料用途	111
表 6-4	Directory Proxy Server 版本驗證輸出	115
表 7-1	Web Server 目錄路徑	123
表 7-2	至 Java ES 5 (發行版本 5) 的升級路徑：Web Server 7.0	124
表 7-3	Web Server 資料用途	125
表 7-4	Web Server 預設名稱	126
表 7-5	Web Server 版本驗證輸出	129
表 7-6	WSInstall.properties 值	133
表 7-7	wadm 共用指令選項	136
表 7-8	wadm migrate-server 指令選項與運算元	136
表 7-9	wadm create-instance 指令選項與運算元	138
表 7-10	wadm set-virtual-server-prop 指令選項與運算元	139
表 7-11	wadm deploy-cofig 指令選項與運算元	140
表 7-12	wadm set-search-collection-prop 指令選項與運算元	141
表 8-1	Java DB 目錄路徑	145
表 8-2	至 Java ES 5 (發行版本 5) 的升級路徑：Java DB 10.1.3	146
表 8-3	Java DB 資料用途	147
表 8-4	Java DB 版本驗證輸出	150
表 9-1	HADB 目錄路徑	155
表 9-2	至 Java ES 5 (發行版本 5) 的升級路徑：HADB 4.4.3	156
表 9-3	HADB 資料用途	157
表 9-4	HADB 版本驗證輸出	160
表 9-5	用於在 Solaris 平台上升級 HADB 的套裝軟體版本	161
表 10-1	Message Queue 目錄路徑	167
表 10-2	至 Java ES 5 (發行版本 5) 的升級路徑：Message Queue 3.7 UR1	169
表 10-3	Message Queue 資料用途 (Solaris OS)	170

表 10-4	Message Queue 版本驗證輸出	174
表 10-5	Message Queue 版本驗證輸出	183
表 11-1	Application Server 目錄路徑	190
表 11-2	至 Java ES 5 (發行版本 5) 的升級路徑： Application Server Enterprise Edition 8.2	192
表 11-3	Solaris OS 隨附 Application Server 版本的升級路徑	193
表 11-4	Application Server 資料用途	194
表 11-5	Application Server 版本驗證輸出	198
表 12-1	Service Registry 目錄路徑	215
表 12-2	至 Java ES 5 (發行版本 5) 的升級路徑：Service Registry 3.1	216
表 12-3	Service Registry 資料用途	217
表 12-4	Service Registry 版本驗證輸出	220
表 13-1	Web Proxy Server 目錄路徑	227
表 13-2	至 Java ES 5 (發行版本 5) 的升級路徑：Web Proxy Server 4.0.4	228
表 13-3	Web Proxy Server 資料用途	229
表 13-4	Web Proxy Server 版本驗證輸出	232
表 13-5	用於在 Solaris 上升級 Web Proxy Server 的修補程式	233
表 13-6	用於在 Linux 上升級 Web Proxy Server 的修補程式	234
表 13-7	Web Proxy Server 版本驗證輸出	238
表 14-1	Access Manager 目錄路徑	241
表 14-2	至 Java ES 5 (發行版本 5) 的升級路徑：Access Manager 7.1	243
表 14-3	Access Manager 資料用途	244
表 14-4	Access Manager 升級的 Web 容器升級方案	246
表 14-5	Access Manager 版本驗證輸出	249
表 14-6	用於升級 Access Manager Mobile Access 軟體的修補程式	253
表 14-7	Mobile Access 修補程式參數	253
表 14-8	Access Manager 配置參數：ampre71upgrade	254
表 14-9	amconfig 支援的 Administrative Server 模式	256
表 14-10	Access Manager 配置參數：amconfig	258
表 14-11	amconfig 參數：發行版本 5 Web Server	259
表 14-12	Access Manager 配置參數：amupgrade	260
表 14-13	用於升級 Access Manager Mobile Access 軟體的修補程式	275
表 14-14	Mobile Access 修補程式參數	275
表 14-15	Access Manager 配置參數：ampre71upgrade	276

表 14-16	Access Manager 配置參數：amconfig	279
表 14-17	Access Manager 配置參數：amupgrade	281
表 15-1	Portal Server 目錄路徑	286
表 15-2	至 Java ES 5 (發行版本 5) 的升級路徑：Portal Server 7.1	287
表 15-3	Portal Server 資料用途	288
表 15-4	Portal Server 升級的 Web 容器升級方案	291
表 15-5	Portal Server 版本驗證輸出	295
表 15-6	各個 Web 容器升級方案中 psupgrade 程序檔所需的資訊	296
表 15-7	JVM 設定的位置	298
表 15-8	將 Portal Server IFR 升級到發行版本 5 的修補程式	333
表 16-1	Portal Server Secure Remote Access 目錄路徑	352
表 16-2	至 Java ES 5 (發行版本 5) 的升級路徑：Portal Server Secure Remote Access 7.1 ...	354
表 16-3	Portal Server Secure Remote Access 資料用途	355
表 16-4	Portal Server Secure Remote Access 版本驗證輸出	360
表 16-5	將 Portal Server Secure Remote Access IFR 升級到發行版本 5 的修補程式	380

前言

「Java Enterprise System 適用於 UNIX 的升級指南」包含在 Sun Solaris™ 作業系統 (Solaris OS) 或 Red Hat Enterprise Linux (RHEL) 作業系統環境中升級 Sun Java™ Enterprise System (Java ES) 軟體所需的資訊。其中並未涵蓋在 HP-UX 環境中進行的升級。

本指南說明了從 Java ES 2004Q2 (發行版本 2)、Java ES 2005Q1 (發行版本 3) 和 Java ES 2005Q4 (發行版本 4) 到 Java ES 5 (發行版本 5) 的升級。

本前言包含以下各節：

- 第 20 頁的「本書適用對象」
- 第 20 頁的「本書中所使用的慣例」
- 第 22 頁的「相關文件」
- 第 24 頁的「線上存取 Sun 資源」
- 第 24 頁的「連絡 Sun 技術支援」
- 第 25 頁的「協力廠商網站參照」
- 第 25 頁的「Sun 歡迎您提出寶貴意見」

本書適用對象

本書適合系統管理員或要升級 Java ES 軟體的軟體技術人員閱讀。

本書假定您熟悉以下內容：

- 企業級軟體產品的安裝
- 目前部署在您的環境中的 Java ES 元件
- 支援的 Java ES 平台上的系統管理和網路
- 叢集模型 (如果您要安裝叢集軟體)

本書中所使用的慣例

本節的表格描述本書中所使用的慣例。

管理介面

對於本「升級指南」中所說明的大部份升級程序而言，都有兩個管理介面可以使用：圖形化使用者介面 (GUI) 和指令行介面。

大多數情況下，本「升級指南」在說明 Java ES 元件升級程序時，使用的都是指令行介面。指令行介面可用於編寫 Java ES 部署升級的程序檔，這樣便可在必要時重複執行各個程序。

但是，當程序使用 Java ES 安裝程式時，將會介紹 GUI 介面，而非互動式基於文字的介面。可將 Java ES 安裝程式階段作業儲存於狀態檔案中，必要時可以使用該檔案來重複執行各個程序。

印刷排版慣例

下表描述本書中所使用的印刷排版慣例。

表 1 印刷排版慣例

字體	意義	範例
AaBbCc123 (固定間距)	API 和語言元素、HTML 標記、網站 URL、指令名稱、檔案名稱、目錄路徑名稱、螢幕畫面輸出、範例程式碼。	請編輯您的 .login 檔案。 請使用 <code>ls -a</code> 列出所有檔案。 % You have mail.
AaBbCc123 (固定間距粗體)	您所鍵入的內容 (與螢幕畫面輸出相區別)。	% su Password:
AaBbCc123 (固定間距粗體)	新的字彙或術語、要強調的詞。將用實際的名稱或數值取代的指令或路徑名稱變數。	這些被稱為 類別 選項。 不儲存檔案。 該檔案位於 安裝目錄 /bin 目錄中。
<i>AaBbCc123</i> (斜體)	保留未譯的新的字彙或術語、要強調的詞。	應謹慎使用 <i>On Error</i> 指令。
「AaBbCc123」	用於書名及章節名稱。	請參閱「使用者指南」中的第 6 章。

符號

下表描述本書中所使用的符號慣例。

表 2 符號慣例

符號	說明	範例	意義
[]	包含選擇性指令選項。	<code>ls [-l]</code>	-l 選項不是必需的。
{ }	包含所需指令選項的一組選擇。	<code>-d {y n}</code>	-d 選項需要您使用 y 引數或是 n 引數。
-	結合多個同步按鍵。	Control-A	按下 Control 鍵同時按住 A 鍵。
+	結合多個連續按鍵。	Ctrl+A+N	按下 Control 鍵、放掉然後再按下後續的鍵。
>	指出圖形化使用者介面中的功能表項目選項。	[檔案]>[新增]>[範本]	從 [檔案] 功能表選擇 [新增]。 從 [新增] 子功能表選擇 [範本]。

Shell 提示符號

下表描述本書中所使用的 Shell 提示符號。

表 3 Shell 提示符號

Shell	提示符號
UNIX 或 Linux 上的 C shell	電腦名稱 %
UNIX 或 Linux 上的 C shell 超級使用者	電腦名稱 #
UNIX 或 Linux 上的 Bourne shell 與 Korn shell	\$
UNIX 或 Linux 上的 Bourne shell 與 Korn shell 超級使用者	#
Windows 指令行	C:\

相關文件

您可以透過下列網站存取 Sun 線上技術文件：<http://docs.sun.com>SM。您可以瀏覽歸檔或是搜尋特定的書名或主題。

Java ES 文件集中的書籍

Java ES 手冊以線上檔案方式提供，有可攜式文件格式 (PDF) 和超文字標記語言 (HTML) 兩種格式。殘障人士可以使用輔助技術閱讀這兩種格式的檔案。可從下列位置存取 SunTM 文件網站：

<http://docs.sun.com>

Java ES 文件包括系統整體及其元件的相關資訊。可從以下位置存取該文件：

<http://docs.sun.com/coll/1286.2> 與 <http://docs.sun.com/coll/1412.2>

下表列出 Java ES 文件集中的系統級手冊。左欄為每份文件的名稱及文件號碼位置，右欄為文件的一般性內容。

表 4 Java Enterprise System 文件

文件	內容
「適用於 UNIX 的 Java Enterprise System 5 版本說明」 http://docs.sun.com/doc/820-0452	包含有關 Java ES 的最新資訊，包括已知問題。此外，元件均有各自的版本說明。
「Java Enterprise System 5 技術摘要」 http://docs.sun.com/doc/820-0884	介紹 Java ES 的技術及概念基礎。描述元件、架構、程序和功能。
「Java Enterprise System 部署規劃指南」 http://docs.sun.com/doc/819-3451	介紹規劃及設計以 Java ES 為基礎的企業部署解決方案。說明部署規劃及設計的一些基本概念及原則、介紹解決方案生命週期，以及提供在部署以 Java ES 為基礎的解決方案時所使用的高層級範例和策略。
「Java Enterprise System 5 安裝規劃指南」 http://docs.sun.com/doc/820-0877	協助您開發 Java ES 部署之硬體、作業系統和網路層面的實作規格。描述要在安裝和配置規劃中解決的問題（如元件相依性）。
「適用於 UNIX 的 Java Enterprise System 5 安裝指南」 http://docs.sun.com/doc/820-0104	引導您完成在 Solaris 作業系統或 Linux 作業系統上安裝 Java ES 的整個程序。還顯示了如何在安裝後配置元件，以及如何確認它們是否運作正常。
「適用於 UNIX 的 Java Enterprise System 5 安裝參照」 http://docs.sun.com/doc/820-0590	提供有關配置參數的附加資訊、提供要在配置規劃中使用的作業表，並列出了參考材料，如預設目錄和連接埠號。
「適用於 UNIX 的 Java Enterprise System 5 升級指南」 http://docs.sun.com/doc/820-0891	提供在 Solaris 作業系統或 Linux 作業環境中升級 Java ES 的指示。
「Sun Java Enterprise System 5 監視指南」 http://docs.sun.com/doc/820-1593	說明如何為每個產品元件設定 Monitoring Framework 以及如何使用 Monitoring Console 來檢視即時資料和設定臨界值警報。
「Java Enterprise System 字彙表」 http://docs.sun.com/doc/819-4632	定義 Java ES 文件中所使用的術語。

搜尋 Sun 產品文件

除了從 docs.sun.com 網站搜尋 Sun 產品文件外，還可藉由在搜尋欄位中鍵入下列語法來使用搜尋引擎：

```
search-term site:docs.sun.com
```

例如，若要搜尋「broker」，請鍵入下列語法：

```
broker site:docs.sun.com
```

若要將其他 Sun 網站納入您的搜尋中 (例如 java.sun.com、www.sun.com、developers.sun.com)，請在搜尋欄位中用 sun.com 取代 docs.sun.com。

線上存取 Sun 資源

如需產品下載、專業服務、修補程式與支援，以及其他開發者資訊，請連線至下列網址：

- 下載中心
<http://www.sun.com/software/download/>
- 用戶端解決方案
<http://www.sun.com/service/sunjavasystem/sjsservicessuite.html>
- Sun 企業服務、Solaris 修補程式與支援
<http://sunsolve.sun.com/>
- 開發者資訊
<http://developers.sun.com>

以下位置包含有關 Java Enterprise System 及其元件的資訊：

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

連絡 Sun 技術支援

如果您在本文件中找不到所需之本產品相關技術問題的解答，請至：
<http://www.sun.com/service/contacting>。

協力廠商網站參照

Sun 對於本文件中所提及之協力廠商網站的使用不承擔任何責任。Sun 對於此類網站或資源中的 (或透過它們所取得的) 任何內容、廣告、產品或其他材料不做背書，也不承擔任何責任。對於因使用或依靠此類網站或資源中的 (或透過它們所取得的) 任何內容、產品或服務而造成的或連帶產生的實際或名義上之損壞或損失，Sun 概不負責，也不承擔任何責任。

Sun 歡迎您提出寶貴意見

Sun 致力於提高文件品質，因此誠心歡迎您提出意見與建議。

若要提出您的意見，請至 <http://docs.sun.com> 並按一下 [傳送您的意見] (Send Comments)。請在線上表單中提供文件標題與文件號碼。文件號碼位於書本的標題頁或文件的頂部，通常是一組七位或九位數的數字。提出意見時您還需要在表格中輸入此文件的英文標題和文件號碼。例如，本文件的英文文件號碼為 819-6553，完整標題為「Sun Java Enterprise System 5 Upgrade Guide for UNIX」。

Sun 歡迎您提出寶貴意見

升級規劃

本章提供在 Sun Solaris™ 作業系統或 Red Hat Enterprise Linux (簡稱 Linux) 作業系統環境中，規劃 Sun Java™ Enterprise System (Java ES) 軟體至 Java ES 5 的升級所需使用的資訊。

包含下列各節：

- [第 28 頁的「Java ES 5 元件」](#)
- [第 31 頁的「Java ES 升級技術」](#)
- [第 35 頁的「升級程序」](#)
- [第 36 頁的「升級規劃注意事項」](#)
- [第 43 頁的「Java ES 元件相依性」](#)
- [第 51 頁的「升級順序指導原則」](#)
- [第 53 頁的「特殊情況」](#)
- [第 54 頁的「Java ES 5 升級與 Solaris 10 區域」](#)

Java ES 5 元件

做為規劃 Java ES 軟體升級的簡介，本節介紹了 Java ES 5 (發行版本 5) 中包含的元件。依您的升級方案而定，可能需要將這些元件之中的一個或多個元件升級到其發行版本 5 版本。

如「Java Enterprise System 5 技術摘要」(<http://docs.sun.com/doc/820-0884>) 中所述，Java ES 元件分組為不同的類型：

- **產品元件。** Java ES 產品元件包括下列內容：
 - 提供主要 Java ES 基礎架構服務的系統服務元件
 - 增強系統服務的服務品質元件
 您可以在 Java ES 安裝程式中選取產品元件。
- **共用元件。** Java ES 共用元件是 Java ES 產品元件依賴的本機共用程式庫。Java ES 安裝程式會自動安裝共用元件。安裝哪些共用元件取決於安裝了哪些產品元件。

發行版本 5 產品元件

下表中以字母順序列出了發行版本 5 產品元件以及在後續表格中使用的縮寫。對於其中的服務品質元件，表中包含它們提供的服務增強類型。

表 1-1 Java ES 5 產品元件

產品元件	縮寫	版本	類型
Access Manager	AM	7.1	系統服務元件
Application Server	AS	8.2	系統服務元件
Directory Proxy Server	DPS	6.0	服務品質：存取元件
Directory Server	DS	6.0	系統服務元件
High Availability Session Store	HADB	4.4.3	服務品質：可用性元件
Java DB	JavaDB	10.2	系統服務元件
Message Queue	MQ	3.7 UR1	系統服務元件
Monitoring Console	MC	1.0	服務品質：管理元件
Portal Server	PS	7.1	系統服務元件

表 1-1 Java ES 5 產品元件 (續)

產品元件	縮寫	版本	類型
Portal Server Secure Remote Access	PSRA	7.1	服務品質：存取元件
Service Registry	SR	3.1	系統服務元件
Sun Cluster	SC	3.1 8/05	服務品質：可用性元件
Sun Cluster Geographic Edition	SCG	2006Q4	服務品質：可用性元件
Web Proxy Server	WPS	4.0.4	服務品質：存取元件
Web Server	WS	7.0	系統服務元件

發行版本 5 共用元件

下表中以字母順序列出了發行版本 5 共用元件以及在後續表格中使用的縮寫。

表 1-2 Java ES 5 共用元件

共用元件	版本	縮寫
Apache Commons Logging	1.0.3	ACL
Jakarta ANT Java/XML-based build tool	1.6.5	ANT
Berkeley Database	4.2.52	BDB
Common Agent Container	1.1 和 2.0	CAC
FastInfoSet	1.0.2	FIS
International Components for Unicode	3.2	ICU
Instant Messenger SDK	6.2.8	IM-SDK
Java Platform, Standard Edition	5.0 Update 7	Java SE
JavaBeans™ Activation Framework	1.0.3	JAF
Java Studio Web Application Framework	2.1.5	JATO
JavaHelp™ runtime	2.0	JavaHelp
JavaMail™ runtime	1.3.2	JavaMail
Java Architecture for XML Binding runtime	2.0.3	JAXB
Java API for XML Processing	1.3.1	JAXP
Java API for XML Registries runtime	1.0.8	JAXR

表 1-2 Java ES 5 共用元件 (續)

共用元件	版本	縮寫
Java API for XML-based Remote Procedure Call runtime	1.1.3_01	JAX-RPC
Java API for Web Services runtime	2.0	JAXWS
Java Calendar API	1.2	JCAPI
Java Dynamic Management™ Kit runtime	5.1.2	JDMK
Java Security Services (Network Security Services for Java)	4.2.4 與 3.1.11	JSS 與 JSS3
JavaServer Pages™ Standard Tag Library	1.0.6	JSTL
KT Search Engine	1.3.4	KTSE
LDAP C SDK	6.0	LDAP C SDK
LDAP Java SDK	4.19	LDAP J SDK
Mobile Access Core	6.2	MA Core
Netscape Portable Runtime	4.6.4	NSPR
Network Security Services	3.11.4	NSS
SOAP Runtime with Attachments API for Java	1.3	SAAJ
Simple Authentication and Security Layer	2.19	SASL
Sun Explorer Data Collector (僅限於 Solaris)	4.3.1	SEDC
Sun Java Monitoring Framework	2.0	MFWK
Sun Java Web Console	3.0.2	SJWC
Web Services Common Library	2.0	WSCL
XML Web Services Security	2.0	XWSS

Java ES 升級技術

沒有一個單一系統公用程式會升級所有 Java ES 元件。此外，產品元件與共用元件升級具有不同的特性與升級技術，如以下各節所述。

產品元件升級

將 Java ES 產品元件升級到發行版本 5 是使用此「升級指南」中所述的元件特定的升級程序，來逐個元件、逐個電腦地執行升級。

產品元件的升級範圍可能是重大功能升級（可能與先前版本的元件不相容），也可能是與先前版本完全相容的錯誤修正升級。由於 Java ES 元件之間的相依性，升級的本質會影響是否還需要升級其他元件。

Java ES 產品元件升級包含兩個基本作業，它們鏡像 Java ES 產品元件的初始安裝和配置：

- **軟體安裝升級。** 升級的軟體會增強或修正現有軟體，或替代現有軟體。軟體安裝可以透過為現有套裝軟體套用修補程式、選擇性替代現有套裝軟體、安裝新套裝軟體或完全重新安裝元件軟體來達成。
- **重新配置。** 重新配置包含為支援已升級軟體而需要對配置資料、使用者資料或動態應用程式資料進行的任何變更。資料變更可以是額外資料、資料格式變更（無論是在特性檔案還是在資料庫模式中），或將資料遷移到新位置。有時重新配置會要求您執行程序，有時它會自動進行。在某些情況下，重新配置也要求將元件軟體重新部署至 Web 容器。

此外，Java ES 產品元件升級通常包含升級前作業，在某些情況下，還包含升級後程序，然後升級才能正常運作。

產品元件升級方法

用於安裝升級軟體及執行元件重新配置的元件特定升級程序包括以下升級方法：

- 使用 [Java ES 安裝程式的升級功能](#)
- [執行產品元件的全新安裝](#)
- [執行元件特定的升級公用程式](#)
- [修補現有元件套裝軟體](#)

使用 Java ES 安裝程式的升級功能

發行版本 5 安裝程式包括在少數特殊情況下執行產品元件升級的升級功能：

Application Server、Message Queue、HADB 與 Java DB。當 Java ES 安裝程式偵測到這些產品元件的先前安裝的發行版本時，它會將這些元件標記為「可升級」。

在升級這些元件中的任何元件之前，安裝程式會檢查目前及先前版本的共用元件。如果安裝程式偵測到選取元件所需的共用元件是先前版本或已經遺失，安裝程式會升級目前安裝的所有共用元件，並安裝選取元件所需的任何遺失的共用元件。在某些情況下 (尤其是 Application Server)，安裝程式也會升級升級元件依賴的產品元件。

安裝程式會移除先前版本的套裝軟體，安裝發行版本 5 產品元件套裝軟體，並會依需要重新配置升級的產品元件。(但是，使用隨附於 Solaris 9 作業系統的 Application Server 時，安裝程式並不會移除套裝軟體；請參閱第 204 頁的「升級發行版本 2 Application Server」。)

如果您使用 Solaris 10 作業系統區域功能，請注意一些特殊注意事項。請參閱第 54 頁的「Java ES 安裝程式中的區域支援」。

執行產品元件的全新安裝

有些產品元件藉由使用 Java ES 安裝程式執行元件的全新安裝來升級。首先，移除先前版本的套裝軟體，然後將發行版本 5 安裝在同一路徑中，或將發行版本 5 安裝在平行路徑中並維持先前版本不變。

在以上兩種情況下，都要藉由將先前版本的配置資料遷移到新安裝、執行新配置或同時執行兩者，來重新配置產品元件。針對某些產品元件，會提供公用程式來為該元件重新配置或遷移配置資料。

執行元件特定的升級公用程式

有些產品元件會提供升級公用程式或程序檔，藉以將元件自動升級到發行版本 5。公用程式通常會執行套裝軟體的升級和做為升級一部份所需要的任何重新配置。針對那些部署到 Web 容器的元件，公用程式通常會將升級的元件軟體重新部署到 Web 容器。

修補現有元件套裝軟體

針對某些產品元件，可透過手動修補現有套裝軟體來執行升級。雖然 Solaris 與 Linux 平台採用類似的技術，透過套裝軟體登錄來管理安裝的套裝軟體及追蹤對這些套裝軟體的變更，但是平台之間修補技術上的差異會影響升級程序。

- **Solaris 平台。** 透過 Solaris pkgadd 和 pkgrm 指令來安裝及移成套裝軟體。安裝後，可使用透過 patchadd 和 patchrm 指令套用或移除的修補程式來修改套裝軟體內容。Solaris 套裝軟體修補程式透過 SunSolve 網站發行，其網址為：
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

Solaris 修補程式可修補一個或多個套裝軟體。patchadd 指令會儲存要修補之套裝軟體的備份，以便使用 patchrm 指令移除修補程式。藉由修補程式 ID 來識別修補程式，該 ID 由修補程式號碼和後面的修訂版號碼構成，該修訂版號碼會隨修補程式的修改而遞增。

- **Linux 平台。** 可以透過 rpm 指令來安裝或更新 Red Hat Enterprise Linux 套裝軟體 (RPM)。不過，一旦安裝了這些套裝軟體，就無法使用修補程式來修改套裝軟體內容，而需要使用 rpm -U 指令選項來更新 RPM 套裝軟體，它會以較新的套裝軟體替代目前套裝軟體。

方便起見，許多 RPM 套裝軟體升級不僅透過 Java ES 發行版本 5 發行物發行，還透過 SunSolve 網站發行，其網址為：

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

透過 SunSolve 發行時，可將 RPM 套裝軟體封裝成修補程式，並為其指定與 Solaris 修補程式類似的修補程式 ID 和修訂版號碼。這些 Linux 修補程式可能包含一個或多個 RPM 套裝軟體，每個套裝軟體都以唯一的 RPM 名稱、RPM 號碼及隨 RPM 套裝軟體的修改而遞增的修訂版號碼來識別。

用於每個產品元件的升級方法

下表顯示用來將每個產品元件升級到發行版本 5 的升級方法：

表 1-3 Java ES 產品元件升級方法

產品元件	升級軟體的安裝	需要重新
Access Manager	替代套裝軟體：使用套裝軟體 移除程序檔 + 全新安裝	使用 amconfig 與 amupgrade 程序檔 來重新配置並重新部署到 Web 容器
Application Server	替代套裝軟體：使用 Java ES 安裝程式的升級功能	除了從發行版本 2 升級之外，其他的 升級不會使用 postinstall 與 asupgrade 程序檔
Directory Proxy Server	執行全新安裝，但不取代先前的 套裝軟體	手動重新配置
Directory Server	執行全新安裝，但不取代先前的 套裝軟體	使用 dsmig 指令來遷移「目錄」資料
High Availability Session Store	替代套裝軟體：使用 Java ES 安裝程式的升級功能或平行全 新安裝	無
Java DB	替代套裝軟體：使用 Java ES 安裝程式的升級功能	無
Message Queue	替代套裝軟體：使用 Java ES 安裝程式的升級功能或 mqupgrade 程序檔 (從發行版本 2)	除了在 Linux 上從發行版本 2 升級之 外，其他的升級不會使用 mqmigrate 程序檔

表 1-3 Java ES 產品元件升級方法 (續)

產品元件	升級軟體的安裝	需要重新
Portal Server	替代套裝軟體：使用 psupgrade 程序檔	使用 psupgrade 程序檔來重新配置並重新部署到 Web 容器
Portal Server Secure Remote Access	替代套裝軟體：使用 psupgrade 程序檔	需要重新配置 psupgrade 程序檔
Service Registry	執行全新安裝，但不取代先前的套裝軟體	手動重新配置並使用 ant upgrade 指令來重新部署到 Application Server 網域
Sun Cluster	替代套裝軟體：使用 scinstall 程序檔來取代二進位檔	使用 scinstall 程序檔來遷移配置
Sun Cluster Geo	替代套裝軟體：使用 uninstall 程序檔 + 全新安裝	無
Web Proxy Server	修補程式二進位檔	無
Web Server	執行全新安裝，但不取代先前的套裝軟體	使用 wadm migrate-server 指令來遷移伺服器實例配置

共用元件升級

Java ES 共用元件升級是升級依賴它們的產品元件之必要組成部份。

共用元件的升級不需要重新配置元件，也不需要升級前或升級後程序。此外，無法將共用元件升級回復到其先前版本。

大量 (大約 30 個) 的 Java ES 共用元件及共用元件與產品元件之間的複雜互動要求將單一作業系統實例中的所有共用元件都與同一 Java ES 發行版本同步化。作業系統實例表示執行 Solaris 9、Solaris 10 或 Red Hat Enterprise Linux 作業系統的單一電腦，或執行 Solaris 10 作業系統的電腦上的任何虛擬作業系統環境 (區域)。

由於同步化需求的原因，您不應該逐一升級 Java ES 共用元件，而需要同時將共用元件升級到其發行版本 5 版本。

共用元件升級到發行版本 5 的同步化可以透過使用 Java ES 安裝程式來達成。當執行產品元件的升級 (請參閱第 32 頁的「[使用 Java ES 安裝程式的升級功能](#)」) 或產品元件的全新安裝時，安裝程式會同步化共用元件。安裝程式還包含同步化功能，它會升級任何現有的共用元件並安裝任何遺失的共用元件。如需此功能的更完整說明，請參閱第 61 頁的「[同步化所有共用元件](#)」。

升級程序

Java ES 升級程序中包含一些階段，在生產環境中執行這些階段之前，通常會先在中繼環境中執行它們。使用中繼環境可以讓您測試每個階段以及編寫 IT 人員要用來升級複雜 Java ES 部署的程序檔。

當您已經在中繼環境中測試了升級程序，並確信升級正常運作之後，您可以在生產環境中重現該程序。

程序包括下表中顯示的各個階段，這些階段在此「升級指南」中有所記錄。各個階段會整體套用到個別元件升級以及您的 Java ES 部署。

表 1-4 升級程序中的階段

升級階段	說明
規劃	制訂升級規劃。其中，您可以指定要升級的 Java ES 元件及在部署中的不同電腦或作業系統實例上升級這些元件時需要依循的順序。
升級前準備	備份配置與應用程式資料、執行作業系統的任何修補、升級任何必要的相依性，並執行其他作業來為升級任何個別元件做準備。
升級	取得升級所需的全部必要套裝軟體、修補程式和工具。依規定安裝升級軟體並重新配置每個元件，包括將資料遷移到升級系統。
驗證	使用規定的驗證測試來驗證升級已經成功，包括啟動升級軟體元件及測試各種使用方案。
升級後程序	執行其他任何使升級元件能夠正常運作（例如，整合新功能）可能需要的配置、自訂或其他作業。
回復 / 復原	回復升級並驗證回復成功。如果由於某種原因，您必須將生產環境復原至其先前狀態，測試升級回復會很重要。

升級規劃注意事項

在升級規劃中，可以指定將升級到發行版本 5 的 Java ES 元件及在 Java ES 部署中的不同電腦或作業系統實例上升級這些元件時需要依循的順序。

您的規劃取決於升級目標和優先權，以及部署架構的範圍和複雜度。

例如，Java ES 部署架構可能僅由單一電腦上執行的單一 Java ES 元件構成，而升級目標是修正以前軟體發行版本中的某個錯誤。或者，Java ES 部署架構也可能由若干相互依存、部署在若干不同電腦上的 Java ES 元件構成，而升級目標則是希望藉由升級最少數量的元件便獲得某些新功能，並使當機時間減至最低。

一般而言，部署架構中 Java ES 元件和電腦的數量愈大，升級規劃就愈複雜。

但是，除了部署架構的範圍和複雜度外，升級規劃還與若干注意事項有關。這些注意事項包括以下因素：

- [升級相依性](#)
- [支援的升級路徑與策略](#)
- [選擇性升級或整體升級](#)
- [多重實例升級](#)
- [作業系統注意事項](#)

升級相依性

規劃 Java ES 產品元件升級時的主要問題之一，就是瞭解該元件對其他 Java ES 元件的相依性，以及是否還需要升級此類其他元件以支援附屬元件的升級。

有兩種類型的升級相依性：

- **硬性升級相依性。** 產品元件的升級會要求您升級它所依賴的元件。此需求可能是因附屬元件需要新功能、新介面或錯誤修正而產生。對於硬性升級相依性，若不是先升級附屬元件所依賴的元件，將無法成功地升級並使用該附屬元件。
- **非硬性升級相依性。** 產品元件的升級不需要您升級它所依賴的元件。對於非硬性升級相依性，無須升級它所依賴的元件，就可以成功地升級並使用該附屬元件。

升級 Java ES 產品元件時需要升級它具有**硬性**升級相依性的所有元件，但可以不升級它具有**非硬性**升級相依性的那些元件（除了本書中說明的某些例外之外）。當升級中牽涉多個相互依存的元件時，如果只有其中一個升級的 Java ES 元件對特定元件具有硬性升級相依性，則必須升級該特定元件。

在少數特殊情況下，由於引入了不相容性，元件升級會要求您同時升級它支援的元件。本書中說明了這些特殊情況。

支援的升級路徑與策略

升級規劃取決於您要升級到發行版本 5 的 Java ES 發行版本。

儘管將所有以前發行版本的 Java ES 軟體都升級到 Java ES 5 (發行版本 5) 是可行的，但唯一支援的升級路徑是從 Java ES 2005Q4 (發行版本 4)、Java ES 2005Q1 (發行版本 3) 和 Java ES 2004Q2 (發行版本 2) 執行升級。雖然本「升級指南」提供了從 Java ES 2003Q4 (發行版本 1) 與 Java ES 之前的發行版本升級的策略，但並未提供執行此類升級的程序。

下表描述了升級到發行版本 5 的不同升級路徑、它們的特性及執行升級時要使用的升級策略。

由於表格中描述的升級路徑之間的差異，以及因為產品元件升級程序通常取決於要升級的發行版本，因此本「升級指南」中對各產品元件升級做逐一描述的章節皆分為以下區段：每個區段代表不同的升級路徑。

表 1-5 至 Java ES 5 (發行版本 5) 的升級路徑

產品版本	Java ES 發行版本	系統特性	升級策略
2005Q4	發行版本 4	Java ES 5 (發行版本 5) 支援在單一電腦上混合使用發行版本 4 與發行版本 5 產品元件，但需要將共用元件與同一發行版本同步。發行版本 4 和發行版本 5 產品元件之間的互通功能已經經過測試，「適用於 UNIX 的 Java Enterprise System 5 版本說明」(http://docs.sun.com/doc/820-0452) 中有對已知介面不相容問題的說明。	發行版本 4 和發行版本 5 產品元件共存使得在單一電腦上或在包含多台電腦的部署架構內選擇性地將發行版本 4 產品元件升級到發行版本 5 具有了可行性。 如果任何發行版本 5 產品元件都需要發行版本 5 共用元件的支援，則必須將電腦上的所有共用元件都與發行版本 5 同步。
2005Q1	發行版本 3	與以上發行版本 4 升級路徑類似。Java ES 5 (發行版本 5) 支援在單一電腦上混合使用發行版本 3 (還有發行版本 4) 與發行版本 5 產品元件，但需要將共用元件與同一發行版本同步。發行版本 3 和發行版本 5 元件之間的互通功能已經經過測試，「適用於 UNIX 的 Java Enterprise System 5 版本說明」(http://docs.sun.com/doc/820-0452) 中有對已知介面不相容問題的說明。	與以上發行版本 4 升級路徑類似。發行版本 3 和發行版本 5 元件共存使得在單一電腦上或在包含多台電腦的部署架構內選擇性地將發行版本 3 元件升級到發行版本 5 具有了可行性。 如果任何發行版本 5 產品元件都需要發行版本 5 共用元件的支援，則必須將電腦上的所有共用元件都與發行版本 5 同步。

表 1-5 至 Java ES 5 (發行版本 5) 的升級路徑 (續)

產品版本	Java ES 發行版本	系統特性	升級策略
2004Q2	發行版本 2	與以上發行版本 4 及發行版本 3 升級路徑相區別。Java ES 5 (發行版本 5) 不支援在單一電腦上混合使用發行版本 2 與發行版本 5 元件，無論是產品元件還是共用元件都不受支援。發行版本的各版本之間存在已知的介面不相容問題，發行版本 2 與發行版本 5 元件之間的互通功能也未經過認證 (未經測試)。	在單一電腦上將元件從發行版本 2 升級到發行版本 5 時，必須將所有發行版本 2 元件都升級到發行版本 5。不過，有時混合使用位於部署架構內不同電腦上的發行版本 2 和發行版本 5 元件是可行的。
2003Q4 及更早版本	發行版本 1 及 Java ES 之前的發行版本	與以上發行版本 2 升級路徑類似。Java ES 5 (發行版本 5) 不支援在單一電腦上混合使用發行版本 2 與發行版本 5 元件，無論是產品元件還是共用元件都不受支援。發行版本的各版本之間存在已知的介面不相容問題，發行版本 1 或更早發行版本與發行版本 5 元件之間的互通功能也未經過認證 (未經測試)。	<p>Java ES 未認證發行版本 1 或更早發行版本到發行版本 5 的直接升級。</p> <p>但是，在某些情況下，您可以藉由先升級到 Java ES 發行版本 3 (如發行版本 3「Java Enterprise System 升級與遷移指南」(http://docs.sun.com/doc/819-2238) 中所述)，然後再從發行版本 3 升級到發行版本 5，來執行從發行版本 1 的升級。在這些情況下，本「升級指南」中該元件的升級資訊指南指出了這種可能性。</p> <p>在其他情況下，可依從發行版本 2 或發行版本 3 執行至發行版本 5 升級的方法執行從發行版本 1 至發行版本 5 的升級。在這些情況下，本「升級指南」中該元件的升級資訊指南指出了這種可行性。</p>

備註

當產品元件發佈正式 Java ES 發行版本之間的臨時功能發行版本 (IFR) 時，通常使用與上述 Java ES 發行版本相同的程序來執行 IFR 升級。例如，如果 IFR 發生在發行版本 3 與發行版本 4 之間，會使用從發行版本 3 升級到發行版本 5 的程序來升級元件。如果情況並非如此 (例如 Portal Server 與 Portal Server Secure Remote Access)，本「升級指南」還說明了 IFR 特定的升級程序。

選擇性升級或整體升級

硬性與非硬性升級相依性之間的差異使得在升級規劃中，在已部署系統內選擇性地升級 Java ES 產品元件具有了可行性。選擇性升級適用於在單一電腦上從發行版本 3 與發行版本 4 升級到發行版本 5。不支援在單一電腦上執行從發行版本 2 至發行版本 5 的選擇性升級。

一般而言，您可以選擇執行選擇性升級或升級電腦上的所有 Java ES 產品元件：

- **選擇性升級。** 使用此方法從您要升級到發行版本 5 的 Java ES 產品元件開始。確定該元件的硬性升級相依性；這些元件也需要升級。為後續的每個硬性升級相依性重複此程序，直到不再有需要升級的元件。此方法須指定需要升級的所有 Java ES 產品元件。
- **整體升級。** 此方法可使所有部署的 Java ES 產品元件都升級到發行版本 5。在某些情況下，由於部署複雜性的原因，基於業務方面的考量，一次升級整個系統並不可行。

下表對這兩種執行升級的方法做了比較。

表 1-6 選擇性升級與整體升級之比較

升級方法	優點	缺點
選擇性升級	將需要升級的元件數量減至最低	導致已部署系統中所有元件的版本不一致
整體升級	維護已部署系統中所有元件的版本均一致	需要升級的元件數量最多

Java ES 發行版本 4 中也支援選擇性升級。因此，使發行版本 3 產品元件與發行版本 4 產品元件並存在一台電腦上是可能的，可以將二者都選擇性地升級到發行版本 5。

多重實例升級

升級規劃中升級程序的順序依在部署架構中如何使用備援而定。可以使用 Java ES 元件的多重實例來達成高可用性、延展性、服務性或這些服務品質的某種組合。有三種技術可以利用 Java ES 部署架構中的備援元件：負載平衡 (Directory Proxy Server、Web Server、Web Proxy Server、Application Server、Access Manager 與 Portal Server)、高可用性技術 (Sun Cluster 與 High Availability Session Store) 和 Directory Server 複製。

在大多數牽涉備援的情況中，執行升級時不可導致重大當機。這些漸進式升級會嘗試在不影響元件備援實例所提供服務的情況下順序升級這些實例。

備援實例通常部署在多台電腦上。針對升級規劃，您可能需要將複製元件的升級與其他元件的升級相隔離，以將當機時間降至最低水平。先在每台電腦上執行複製元件的所有升級前作業，然後再執行漸進式升級。

每種複製技術均有配置或重新配置程序，這些程序可能會影響 Java ES 元件升級的整體順序。例如，在 Sun Cluster 環境中執行的元件可能需要先升級 Sun Cluster，然後再升級在 Sun Cluster 環境中執行的元件。

描述每個產品元件升級的本「升級指南」中的章節描述了如何執行其各自元件的多重實例升級。

作業系統注意事項

若干作業系統注意事項會影響 Java ES 升級規劃，如下所述。

必要的作業系統修補程式

Java ES 產品元件的成功升級可能需要您先修補作業系統，或者將作業系統更新為 Java ES 5 產品元件所需的層級。但是，通常較好的做法是先將作業系統整體更新為 Java ES 5 所需的層級，然後再執行特定產品元件的升級，而不是套用每種情況下需要的特定修補程式或修正程式。

- **Solaris 平台。**可透過 SunSolve 網站取得修補程式叢集形式的作業系統修補程式，修補程式叢集是可一併套用的作業系統修補程式的集合。可從 <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access> 取得支援適用於 Solaris 9 與 10 的 Java ES 發行版本 5 所需的修補程式叢集。
- **Linux 平台。**可從 <https://www.redhat.com/apps/download/> 取得更新發行版本。但是，不必在執行 Java ES 升級之前更新 Linux 作業系統。

雙重升級：Java ES 與作業系統軟體

當您嘗試將 Java ES 軟體或作業系統軟體升級到不支援的版本時，作業系統與 Java ES 軟體可能會變得不協調。下表顯示了相關的支援矩陣。

表 1-7 Java ES/ 作業系統支援矩陣

Java ES 發行版本	Solaris			RHEL		
	8	9	10	2.1	3.0	4.0
2003Q4 (發行版本 1)	X	X		X		
2004Q2 (發行版本 2)	X	X		X		
2005Q1 (發行版本 3)	X	X	X	X	X	
2005Q4 (發行版本 4)	X	X	X	X	X	
5 (發行版本 5)		X	X		X	X

如果升級 Java ES 軟體或作業系統軟體會導致不支援的配置，則您必須執行雙重升級：也就是 Java ES 與作業系統均進行升級。對於下列情況，可能需要進行雙重升級：

- 將作業系統升級到安裝的 Java ES 軟體不支援的版本。

例如，Solaris 8 和 9 作業系統以及 Red Hat Enterprise Linux (RHEL) 2.1 皆支援 Java ES 2004Q2 (發行版本 2)。如果要將作業系統平台升級到 Solaris 10 或 RHEL 3.0 (它們不受 Java ES 發行版本 2 支援)，則還需要將 Java ES 發行版本 2 升級到支援升級後平台的 Java ES 發行版本。若是這樣，最好是升級到 Java ES 5 (發行版本 5)。

- 將 Java ES 升級到不受現有作業系統軟體支援的版本。

例如，Solaris 8 與 RHEL 2.1 支援 Java ES 2005Q1 (發行版本 3) 與 Java ES 2005Q4 (發行版本 4)。但是，如果您要將 Java ES 升級到發行版本 5 (它在 Solaris 8 或 RHEL 2.1 上不受支援)，您就必須將作業系統升級到 Java ES 5 (發行版本 5) 所支援的版本。若是這樣，最好是升級到 Solaris 10 或 RHEL 4.0。

一般而言，您可以採取兩種方法來執行雙重升級：

- 全新安裝作業系統。**安裝新的作業系統，緊接著進行 Java ES 發行版本 5 的全新安裝，包括遷移較早版本產品元件資料 (如配置資料、執行階段資料、自訂內容等等)。可將作業系統安裝在新系統 (或 Solaris 10 區域) 上，亦可讓其徹底覆蓋現有的檔案系統。若是後一種情況，必須先備份元件資料，然後在作業系統安裝之後進行復原。
- 原地升級作業系統。**執行作業系統升級，原樣保留現有的檔案系統，接著將 Java ES 產品升級到發行版本 5。為了做到這一點，作業系統升級不得對已安裝的 Java ES 產品元件及其資料和所需的共用元件有任何影響。

如果有任何 Java ES 產品元件不支援雙重升級，也就是說，如果這兩種方法都行不通，則您必須在執行作業系統安裝或升級之後，重新安裝並全新配置該元件。

下表顯示各個 Java ES 產品元件所支援的雙重升級方法。

表 1-8 Java ES 5 產品元件的雙重升級支援

產品元件	全新安裝作業系統	原地升級作業系統
Access Manager	不支援	不支援
Application Server	僅在相同電腦上才支援	支援
Directory Proxy Server	支援	支援
Directory Server	支援	支援
High Availability Session Store	在 Application Server 雙重升級環境下執行	在 Application Server 雙重升級環境下執行
Java DB	支援	支援

表 1-8 Java ES 5 產品元件的雙重升級支援 (續)

產品元件	全新安裝作業系統	原地升級作業系統
Message Queue	支援	支援
Portal Server	不支援	支援
Portal Server Secure Remote Access	不支援	支援
Service Registry	支援	支援
Sun Cluster	不支援	支援
Sun Cluster Geographic Edition	不支援	在 Sun Cluster 雙重升級環境下執行
Web Proxy Server	不支援	支援
Web Server	不支援	支援

作業系統升級

在某些情況下，升級 Solaris 作業系統會覆寫較早版本的現有 Java ES 共用元件。在這些情況下，將 Solaris 作業系統隨附的 Message Queue 升級到發行版本 5 可以復原正確的 Java ES 版本。升級 Message Queue 也會強制升級所有常駐共用元件。

Solaris 10 多區域環境

在多區域環境中安裝及升級 Java ES 元件時牽涉到許多問題。如需在 Solaris 10 區域中部署 Java ES 的好處與限制及在多區域環境中升級 Java ES 元件的建議措施的說明，請參閱第 54 頁的「Java ES 5 升級與 Solaris 10 區域」。

Java ES 元件相依性

已部署系統中各種 Java ES 元件之間的相依性是升級規劃的最重要注意事項之一。執行元件升級的順序會受到它們之間的相依性特性的影響。

本節提供關於影響升級規劃的 Java ES 元件相依性的資訊。

- 對共用元件的相依性
- 對產品元件的相依性

對共用元件的相依性

第 43 頁的表 1-9 顯示了 Java ES 5 (發行版本 5) 產品元件對 Java ES 共用元件的相依性。表 1-9 欄標題中的產品元件縮寫取自第 28 頁的表 1-1。第 29 頁的表 1-2 中列出了共用元件的縮寫。

在表 1-9 矩陣中，對於發行版本 3 與發行版本 4 到發行版本 5 的升級，硬性升級相依性以「H」標示，非硬性升級相依性以「S」標示。對於發行版本 2 到發行版本 5 的升級，依定義，所有共用元件相依性皆為硬性升級相依性；必須將所有共用元件從發行版本 2 升級到發行版本 5。

表 1-9 Java ES 5 (發行版本 5) 產品元件的共用元件相依性

共用元件	AM	AS	DPS	DS	DS Console	HADB	JavaDB	MQ	MC	PS	PSRA	SC	SCG	SR	WPS	WS
ANT		S							S	H	H			H		
ACL	S													H		
BDB	S															
CAC	H	S	H	H	H				S	S		S ₁	S ₁			
FIS																
ICU		S	H	H						S					S	S
IM-SDK										S						
Java SE	S	S	H	H	H	S	H	S	S	S	S	S	S	H	S	S
JAF	S	S								S	S			H		
JATO	S	S							S	S		S	S			

表 1-9 Java ES 5 (發行版本 5) 產品元件的共用元件相依性 (續)

共用元件	AM	AS	DPS	DS	DS Console	HADB	JavaDB	MQ	MC	PS	PSRA	SC	SCG	SR	WPS	WS
JavaHelp™	S	S						S	S							S
JavaMail™	S	S								S	S			H		S
JAXB	S	S														S
JAXP	S	S								S	S			H		S
JAXR	S	S												H		S
JAX-RPC	S	S												H		S
JAXWS																S
JCAPI																
JDMK	H	S	H	H	H				S			S	S			S
JSS	S									S	S				S	S
JSTL																
KTSE										S					S	S
LDAP C SDK	H			H											S	S
LDAP J SDK	S															
MA Core	S									H	H					
MFWK	H			H					H							
NSPR	S	S	H	H				H	S	S	S	S	S		S	H
NSS	S	S		H				H	S	S	S	S	S		S	H
SAAJ	S	S								S	S			H		
SASL				H											S	S
SEDC												S	S			
SJWC	S	S			H				H			S	S			
WSCL	S	S												H		S
XWSS														H		

1. 此相依性具體地講是對 Common Agent Container (CAC) 版本 1.1 的相依性。

表 1-9 中顯示的任何產品元件的相依性既包括直接又包括間接共用元件相依性：某個產品元件可能依賴於特定的共用元件（直接相依性），而該共用元件又依賴於一個或多個其他共用元件（間接相依性）。第 46 頁的圖 1-1 說明了共用元件之間的交互相依性。

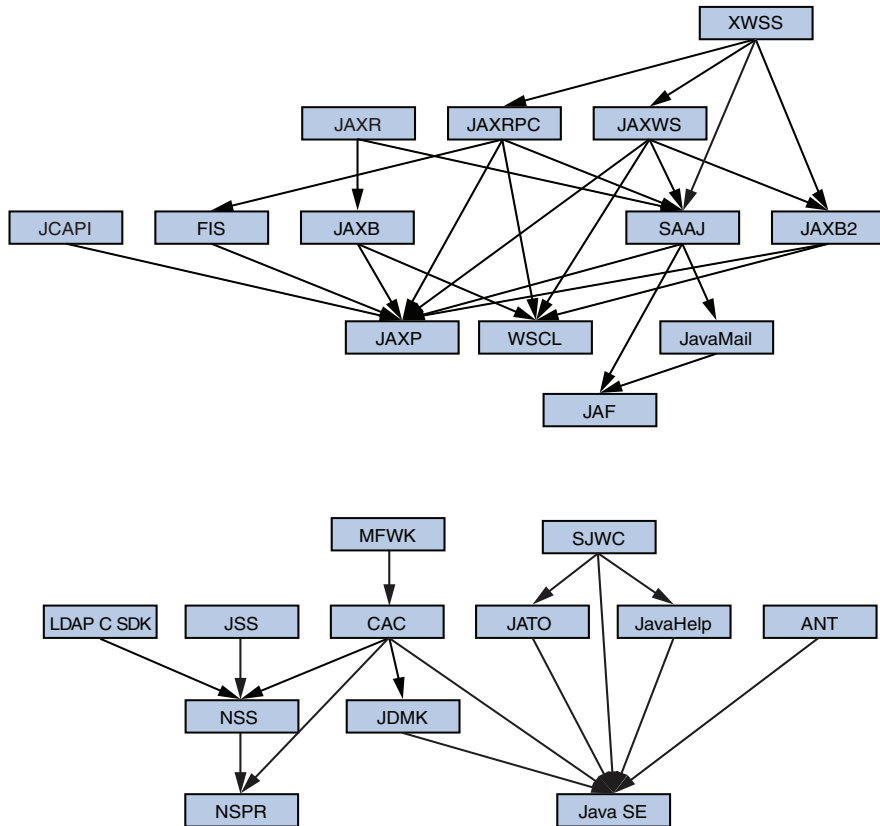
表 1-9 顯示在指定電腦上升級一或多個產品元件時，必須升級的共用元件。

但是，由於必須同步化共用元件（請參閱第 34 頁的「共用元件升級」），因此您無法逐一升級 Java ES 共用元件，而是必須同時將電腦上或作業系統實例中的所有共用元件升級到其發行版本 5 版本。

若不牽涉硬性升級相依性，您就不需要升級共用元件。不過，較好的做法是將底層的 Java ES 共用元件基礎升級到最新版本。實際上，當 Java ES 安裝程式安裝或升級產品元件時，位於主機電腦上的所有共用元件皆會自動與發行版本 5 同步。

若需關於如何手動升級共用元件的資訊，請參閱第 2 章「升級 Java ES 共用元件」。

圖 1-1 共用元件相依性



對產品元件的相依性

對產品元件的相依性分為兩大種類：執行階段相依性和配置相依性。

- **執行階段相依性。** 軟體系統的運作以其已部署元件之間的互動為基礎。「Java Enterprise System 5 技術摘要」中討論了 Java ES 產品元件之間的基礎架構相依性。如果發行版本 5 產品元件對其他產品元件有硬性升級相依性，則僅在升級了附屬元件所依賴的元件的情況下，才能成功升級該附屬元件並依預期使用。
- **配置相依性。** 在某些情況下，必須先安裝、配置並執行某個 Java ES 元件，才能配置另一個元件。例如，Directory Server 使用者 / 群組目錄必須正在執行，才能註冊 Access Manager 服務。元件升級程序通常包括重新配置升級後的元件或遷移配置資料。配置相依性可能會影響升級程序的順序。

針對執行階段相依性，產品元件之間的關係可以是以下三種類型：

- **強制性。** 元件在沒有支援元件的情況下無法執行。
- **選擇性。** 元件可以在沒有支援元件的情況下執行，但是它的功能的子集需要支援元件。
- **互有相依性。** 兩個元件都可以在沒有彼此支援的情況下執行，但是一起使用的元件可以提供某些增強的功能或效能。

下表顯示在 [第 28 頁的表 1-1](#) 中列出的 Java ES 產品元件之間的相依性與相依性關係。可以使用此資訊來確定影響您升級規劃的硬性升級相依性。

第一欄會依字母順序列出發行版本 5 產品元件，第二欄顯示發行版本 5 元件與其有相依性關係的其他 Java ES 元件，第三欄提供支援發行版本 5 相依性的 Java ES 發行版本，第四欄描寫相依性關係的特性，最後一欄指示相依性的特殊特性，例如支援元件是否必須為本機元件 (與遠端相對) 或者其他協力廠商產品是否可以支援相依性。

如果您要升級到發行版本 5 的產品元件對發行版本 5 (與較早版本相對) 的支援元件有相依性，那麼支援元件便代表硬性升級相依性：支援元件也必須升級到發行版本 5。

表 1-10 Java ES 產品元件相依性

發行版本 5 產品元件	相依性 ¹	Java ES 發行版本	相依性的本質	特性
Access Manager	Directory Server	2-5	強制性：儲存配置資料和啓用使用者資料的查找	
	J2EE Web 容器：		強制性：提供 Web 容器執行階段服務	僅本機
	- Application Server	4-5		也受支援；
	- Web Server	4-5		- Weblogic ² - WebSphere ³

表 1-10 Java ES 產品元件相依性 (續)

發行版本 5 產品元件	相依性 ¹	Java ES 發行版本	相依性的本質	特性
Access Manager SDK	Access Manager	3-5	強制性：提供 Access Manager 服務	
Access Manager Distr.Authentication	Access Manager	4-5	強制性：提供 Access Manager 服務	
	J2EE Web 容器：		強制性：提供 Web 容器執行 階段服務	僅本機
	- Application Server	4-5		也受支援；
	- Web Server	4-5		- Weblogic ² - WebSphere ³
Access Manager Session Failover	Access Manager	5	強制性：提供 Access Manager 服務	
	Message Queue	4-5	強制性：提供可靠的非同步郵件 傳送	
Application Server	Message Queue	3-5	強制性：提供可靠的非同步郵件 傳送	僅本機
	High Availability Session Store (HADB)	5	強制性：儲存支援實例之間容錯 移轉所需的階段作業狀態	僅本機
	Java DB	5	強制性：提供預設開發者資料庫 與其他永久儲存	僅本機
	Web Server	3-5	選擇性：提供實例之間的負載 平衡	僅本機
Directory Proxy Server	Directory Server	1-5	互有相依性：產生針對目錄請求 的改進的安全性與效能。將資料 提供給 Directory Proxy Server	
Directory Server	Directory Proxy Server	1-5	互有相依性：產生針對目錄請求 的改進的安全性與效能。分散負 載並快取來自 Directory Server 的資料	
High Availability Session Store (HADB)	無			
Java DB	無			

表 1-10 Java ES 產品元件相依性 (續)

發行版本 5 產品元件	相依性 ¹	Java ES 發行版本	相依性的本質	特性
Message Queue	Directory Server	2-5	選擇性：儲存管理的物件與使用者資料	
	J2EE Web 容器：		選擇性：支援用戶端與 Message Queue 代理程式之間的 HTTP 傳輸	
	- Application Server	2-5		
	- Web Server	2-5		
	Java DB	5	選擇性：儲存永久訊息	僅本機
	Sun Cluster	2-5	選擇性：支援高可用性	
Monitoring Cosole	無			
Portal Server	Directory Server	4-5	強制性：儲存及啓用使用者設定檔的查找	
	J2EE Web 容器：		強制性：提供 Web 容器執行階段服務	僅本機
	- Application Server	4-5		
	- Web Server	4-5		
	Access Manager或 Access Manager SDK	4-5	強制性：提供認證和授權服務、單次登入	僅本機 (如果 Access Manager 是遠端的，則必須在本機使用 Access Manager SDK)
	Portal Server Secure Remote Access	5	選擇性：透過 Gateway、Rewriter Proxy 與 Netlet Proxy 元件提供安全遠端存取	
	Service Registry Client	5	強制性：提供編譯所需的程式庫	
	Java DB	5	強制性：為數個 portlet 應用程式提供支援	
Portal Server Secure Remote Access Gateway	Portal Server	5	強制性：支援 Gateway 功能	
	Access Manager或 Access Manager SDK	4-5	強制性：提供認證和授權服務、單次登入	僅本機 (如果 Access Manager 是遠端的，則必須在本機使用 Access Manager SDK)
	Directory Server	4-5	強制性：儲存及啓用使用者資料的查找	

表 1-10 Java ES 產品元件相依性 (續)

發行版本 5 產品元件	相依性 ¹	Java ES 發行版本	相依性的本質	特性
Rewriter Proxy	Portal Server	5	強制性：支援 Rewriter Proxy 功能	
Netlet Proxy	Portal Server	5	強制性：支援 Netlet Proxy 功能	
Service Registry 部署	Application Server	5	強制性：提供容器執行階段服務	僅本機
	Java DB	5	強制性：為儲存服務與相關中介資料提供預設資料庫	僅本機
	Service Registry Client	5	強制性：提供所需用戶端程式庫	僅本機
Client	無			
Sun Cluster	無			
Sun Cluster Agents	Sun Cluster	4-5	強制性：提供對 Sun Cluster 服務的存取	僅本機
Sun Cluster Geographic Edition	Sun Cluster	4-5	強制性：支援 Sun Cluster Geographic Edition 功能	僅本機
Web Proxy Server	Directory Server	2-5	選擇性：提供基於 LDAP 的認證	
	Web Server	2-5	互有相依性：產生針對 HTTP 請求的改進的安全性與效能。將資料提供給 Web Proxy Server	也受支援； - Weblogic ² - WebSphere ³
Web Server	Directory Server	1-5	選擇性：提供基於 LDAP 的認證	
	Web Proxy Server	1-5	互有相依性：產生針對 HTTP 請求的改進的安全性與效能。分散負載並快取來自 Web Server 的資料	

1. 針對每個產品元件，相依性會以通常的升級順序列出。

2. BEA Weblogic Server

3. IBM WebSphere Application Server

升級順序指導原則

前面各節中討論的選擇性升級或整體升級之間的選擇、硬性升級相依性的影響，及其他因素皆可影響規劃升級的 Java ES 元件以及升級它們需要的順序。儘管如此，仍可套用幾個一般升級順序指導原則，雖然它們並非在每種情況下都適用。

以下清單提供了可以在單一電腦上或已部署的系統中成功升級 Java ES 元件的順序。規劃升級時，您可以省略不屬於部署架構組成部份的元件，或者如果執行的是選擇性升級，則可以省略那些代表非硬性升級相依性的元件。

本「升級指南」中的章節是根據元件在以下清單中的顯示順序排列的。

備註 升級 Java ES 元件之前，請務必套用您的作業系統所需要的任何更新（請參閱第 40 頁的「必要的作業系統修補程式」）。
並請檢查第 53 頁的「特殊情況」以查看是否有一些特殊情況適合您的升級方案。

1. 共用元件 (請參閱第 59 頁的第 2 章「升級 Java ES 共用元件」)

應先升級共用元件，然後再升級依賴它們的元件。在大多數情況下，共用元件升級由 Java ES 安裝程式處理，但是就 Web Proxy Server 與 Portal Server 而言，您必須明確升級共用元件。

2. Sun Cluster 軟體 (請參閱第 69 頁的第 3 章「Sun Cluster 軟體」)

如果任何元件在 Sun Cluster 環境中執行，而且需要升級 Sun Cluster 軟體，則應先升級這些元件，然後再升級使用 Sun Cluster 服務的元件。如果要升級 Sun Cluster Agents，應將其做為 Sun Cluster 升級的一部份進行升級。

3. Sun Cluster Geographic Edition 軟體 (請參閱第 81 頁的第 4 章「Sun Cluster Geographic Edition」)

Sun Cluster Geographic Edition 應該在升級它所依賴的 Sun Cluster 軟體之後升級。Sun Cluster 應該在升級任何使用 Sun Cluster 服務的元件之前升級。

4. Directory Server (請參閱第 93 頁的第 5 章「Directory Server」)

許多元件在 Directory Server 中儲存使用者資料或配置資料，因此，一般應先執行對 Directory Server 的升級，然後再升級對 Directory Server 有執行階段或配置相依性的元件。

5. Directory Proxy Server (請參閱第 109 頁的第 6 章「[Directory Proxy Server](#)」)

Directory Proxy Server 對 Directory Server 有非硬性升級相依性，而且可以隨時升級。但某些元件可能會透過 Directory Proxy Server 存取 Directory Server，因此如果升級 Directory Proxy Server，則它應該在 Directory Server 之後升級。

6. Web Server (請參閱第 123 頁的第 7 章「[Web Server](#)」)

若干 Java ES 元件需要 Web 容器的支援，如果要升級 Web 容器，應先對其進行升級，然後再升級需要 Web 容器服務的元件。通常 Web 容器服務由 Web Server 或 Application Server 提供，但若架構包含這兩者，請先升級 Web Server，再升級 Application Server。

7. Java DB (請參閱第 145 頁的第 8 章「[Java DB](#)」)

Java DB 必須在升級 Application Server 之前升級，Application Server 需要 Java DB 做為預設資料庫。但是，當升級 Application Server 時，Java ES 安裝程式會自動升級 Java DB。

8. High Availability Session Store (請參閱第 155 頁的第 9 章「[High Availability Session Store](#)」)

必須在升級 Application Server 之前升級 High Availability Session Store (HADB)，Application Server 需要 High Availability Session Store 來提供高可用性。但是，當升級 Application Server 時，Java ES 安裝程式會自動升級 HADB。

9. Message Queue (請參閱第 167 頁的第 10 章「[Message Queue](#)」)

必須在升級 Application Server 之前升級 Message Queue，Application Server 需要 Message Queue 以與 Java Enterprise Edition (Java EE) 相容。但是，當升級 Application Server 時，Java ES 安裝程式會自動升級 Message Queue。

10. Application Server (請參閱第 189 頁的第 11 章「[Application Server](#)」)

Application Server 依賴 Message Queue 與 High Availability Session Store，若要升級 Application Server，應該在升級這些元件之後升級。Application Server 也可以依賴 Web Server 的負載平衡外掛程式，因此，若要使用該功能，應在升級 Web Server 之後再升級 Application Server。

11. Service Registry (請參閱第 215 頁的第 12 章「[Service Registry](#)」)

Service Registry 可以在升級 Application Server 之後的任何時間升級，因為 Service Registry 依賴 Application Server 的執行階段容器服務。

12. Web Proxy Server (請參閱第 227 頁的第 13 章「Web Proxy Server」)

可隨時升級 Web Proxy Server，雖然一般是在升級 Web Server 或 Application Server 元件 (它為這兩者提供代理服務) 之後再升級它。Web Proxy Server 是一個新的 Java ES 發行版本 5 元件，可從它先前的非 Java ES 發行版本進行升級。

13. Access Manager (請參閱第 241 頁的第 14 章「Access Manager」)

Access Manager 在認證和授權 (包括單次登入) 中發揮著中心作用，如果要升級它，應先對其進行升級，然後再升級依賴它來存取這些服務的元件。

14. Portal Server (請參閱第 285 頁的第 15 章「Portal Server」)

Portal Server 依賴上述元件之中的許多元件 (Directory Server、Web 容器與 Access Manager)，如果要升級它，應在升級這些元件之後進行。

15. Portal Server Secure Remote Access (請參閱第 351 頁的第 16 章「Portal Server Secure Remote Access」)

Portal Server Secure Remote Access 必須在升級 Portal Server 之後才可升級。

特殊情況

當規劃將 Java ES 元件升級到發行版本 5 時，需要注意一些特殊情況。如下所述。

選擇性升級：未升級 Application Server

如果要在執行發行版本 3 或發行版本 4 Application Server (8.1) 的電腦上將任何 Java ES 元件選擇性升級到 Java ES 5，而不將 Application Server 升級到發行版本 5，則必須處理一些情況，Application Server 才能繼續正確運作：

- **JSP 編譯錯誤**。在執行選擇性升級之前，您應該先套用下表中顯示的 Application Server 修補程式。

表 1-11 Application Server 未升級到發行版本 5 時所需的修補程式¹

說明	修補程式 ID：Solaris 9 和 10	修補程式 ID：Linux
修正發行版本 3 與發行版本 4 Application Server	119166-17 (SPARC) 119166-17 (x86)	119168-17

¹ 修補程式的修訂版號碼是最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

如果您套用修補程式失敗，Application Server 會遇到 JSP 編譯錯誤。(也可以追溯地套用表 1-11 中的修補程式以修正問題。)

- **Linux 上 ANT 共用元件二進位檔的重新定位。**發行版本 5 ANT 位於與先前版本不同的路徑中。在 *AppServer8-base/config/asenv.conf* 檔案中指定的指向 ANT 的 Application Server 環境變數必須變更，必須從：

```
AS_ANT_LIB="/opt/sun/lib"
```

變更爲：

```
AS_ANT_LIB="/opt/sun/share/lib"
```

然後必須重新啓動 Application Server。

將 Portal Server 臨時功能發行版本 (IFR) 7.0 升級到 Java ES 5

如果您要在 Web Server 環境中將 Portal Server 從臨時功能發行版本 (IFR) 7.0 2005Q4 升級到發行版本 5，請參閱第 333 頁的「從臨時功能發行版本 7.0 升級 Portal Server」以瞭解第 51 頁的「升級順序指導原則」中指導原則的例外情況。

Java ES 5 升級與 Solaris 10 區域

本節陳述在 Solaris 10 區域中升級 Java ES 軟體的相關問題，並提供此類環境中的建議。該節對「Java Enterprise System 5 安裝規劃指南」(<http://docs.sun.com/doc/820-0877>) 中有關 Java ES 5 與 Solaris 10 區域的資訊做了補充。

包含下列主題：

- [Java ES 安裝程式中的區域支援](#)
- [建議的升級措施](#)
- [特殊情況或例外情況](#)

Java ES 安裝程式中的區域支援

Java ES 5 安裝程式爲 Java ES 產品元件的升級 (與安裝) 及共用元件的同步化提供合格的區域支援。已經在安裝程式中實作策略以幫助防止出現有問題的升級方案。

產品元件的升級

如第 32 頁的「使用 Java ES 安裝程式的升級功能」中所述，Java ES 安裝程式可以用來升級有限數量的產品元件以及其對應的共用元件。升級功能會套用至全域區域及所有非全域區域。

但是，此運作方式有三種與區域相關的例外情況：

- 在稀疏根區域中，由於某些共用元件位於唯讀目錄中，因此無法安裝或升級它們。在此類情況下，會暫停產品元件的升級，直到已在全域區域中安裝或升級此類共用元件為止。安裝程式提供以下訊息：[無法在稀疏根區域中安裝或升級以下您已選取元件所需的共用元件。繼續之前，請先在全域區域中安裝或升級這些共用元件。請使用 [所有共用元件] 選項。]
- Application Server 與 Message Queue 都隨附於 Solaris 作業系統。這兩個版本都無法直接在稀疏根區域中升級。如需有關這兩個隨附元件的詳細資訊，請參閱第 57 頁的「產品元件特殊情況」。
- 在全域區域中，如果存在非全域區域，安裝程式不會升級目前安裝的所有共用元件，也不會安裝選取元件所需的任何遺失的共用元件，而是將**所有** Java ES 共用元件與發行版本 5 同步，無論任何特定產品元件是否需要它們。這就可以讓所有發行版本 5 共用元件傳遞至非全域區域，從而確保非全域區域中沒有混合的共用元件版本。

備註 有一些特殊情況或例外情況，它們可能會干擾非全域區域中的產品元件的安裝或升級。我們將在第 57 頁的「特殊情況或例外情況」中進行介紹。

同步化所有共用元件

發行版本 5 中提供了共用元件同步選項，以符合所有共用元件都必須與它們的發行版本 5 版本同步的情況。當選取了 [所有共用元件] 選項之後，安裝程式將會升級目前安裝的所有共用元件，及安裝任何遺失的共用元件，無論任何特定產品元件是否需要它們。此選項會套用至全域區域及完整根區域 (但不會套用至稀疏根區域)。

以下兩個基於區域的升級方案中需要第 61 頁的「同步化所有共用元件」中詳細描述的 [所有共用元件] 選項：

- **手動升級產品元件。** 當升級**無法**使用 Java ES 安裝程式升級的產品元件時，需要 [所有共用元件] 選項，以執行所需的共用元件安裝及升級。

- **在稀疏根區域中的升級。** 無法在預設稀疏根區域中安裝或升級某些共用元件。因此，當在稀疏根區域中使用 Java ES 安裝程式來升級產品元件時，根據相關的共用元件而定，可能會先要求您同步化全域區域中的共用元件。在這種情況下，您可以在全域區域中使用 [所有共用元件] 選項來執行所需的共用元件安裝及升級。

如需有關共用元件的 Java ES 安裝程式的區域運作方式的摘要，請參閱「Java Enterprise System 5 安裝規劃指南」(<http://docs.sun.com/doc/820-0877>) 中有關 Java ES 5 與 Solaris 10 區域的資訊。

建議的升級措施

在制訂升級規劃的過程中，您應該檢查任何 Java ES 軟體的現有多區域部署，請記住「Java Enterprise System 5 安裝規劃指南」(<http://docs.sun.com/doc/820-0877>) 中敘述的區域安裝與管理策略。在某些情況下，可能會要求您解除安裝一或多個區域中的元件，並在其他區域中重新安裝它們，以實作以下建議的措施：

- 避免混合策略。特別的是：
 - 盡量將您的 Java ES 區域部署與管理策略維持地越簡單越好。請勿在同一台電腦上混合使用 Java ES 元件的完整根與稀疏根部署。(支援稀疏根區域部署所需的程序與措施可能會干擾完整根區域部署。)
 - 請勿在全域區域與非全域區域中安裝相同的 Java ES 產品元件，即使它們是不同的版本。(升級全域區域安裝所需的程序可能會破壞非全域區域安裝。)
 - 當已將發行版本 4 (或較早版本) Java ES 元件安裝在完整根區域中時，請勿在全域區域中將 Java ES 元件升級到發行版本 5。在全域區域中升級可能會導致在完整根區域中混合使用發行版本 4 與發行版本 5 檔案。
- 升級措施：
 - 如果要將所有安裝的發行版本 4 產品元件升級到發行版本 5，請在全域區域中同步化所有 Java ES 共用元件，然後在已安裝所需產品元件的區域中升級它們。(向下相容發行版本 5 共用元件。)
 - 如果已經將發行版本 4 或發行版本 5 產品元件安裝在非區域環境中，而且要將非全域區域新增到此環境中及在新的非全域區域中安裝產品元件，則可能需要解除安裝全域區域中的元件並在非全域區域中重新安裝它們。

特殊情況或例外情況

有一些特殊情況，其中一些情況是由隨附於 Solaris 10 的某些 Java ES 共用元件與某些 Java ES 產品元件所引起的。藉由此隨附方式，這些 Java ES 元件會自動存在於全域區域中，也因此存在於任何從全域區域建立的非全域區域中。

產品元件特殊情況

- **Message Queue**。Message Queue 隨附於 Solaris 10，因此會在建立非全域區域時自動傳遞（除非您已經先從全域區域中移除了 Message Queue）。無法在稀疏根區域中安裝或升級 Message Queue。當透過 Java ES 安裝程式在全域區域中進行安裝或升級時，Message Queue 與其他產品元件不同，依預設它會被傳遞到非全域區域。
- **Application Server**。Application Server 隨附於 Solaris 10，因此會在建立非全域區域時自動傳遞（除非您已經先從全域區域中移除了 Application Server）。當以此方式傳遞時，Java ES 安裝程式無法在稀疏根區域中升級安裝在 /usr 中的隨附的 Application Server（依預設，/usr 是唯讀的）。若要解決此問題，必須先從全域區域中將隨附的 Application Server 套裝軟體手動移除，再將發行版本 5 Application Server 安裝在稀疏根區域中。請參閱第 204 頁的「[僅限於 Solaris OS：手動移除隨附於作業系統的 Application Server 套裝軟體](#)」。
- **Sun Cluster**。Sun Cluster 軟體在非全域區域中不受支援。

共用元件特殊情況

- **Sun Java Web Console (SJWC)**。Java ES 安裝程式無法移除隨附於 Solaris 10 (Update 1 與 Update 2) 的 SJWC 套裝軟體。這些較舊的 SJWC 套裝軟體已經將 SUNW_PKG_ALLZONES 屬性設定為 True，這表示套裝軟體在所有區域中必須是相同的，而且只能由全域管理員管理。因此，必須在全域區域中手動移除這些套裝軟體並由正確的套裝軟體取代。

如果 Java ES 安裝程式嘗試在非全域區域中安裝選取的產品元件，並偵測到需要升級 SJWC，安裝程式將會封鎖。這種情況會於在 Solaris 10 (Update 1 與 2) 上安裝時發生。

已經開發了特殊的程序檔做為解決方法，它可從全域區域中移除 SJWC 的舊套裝軟體，並以擁有正確區域傳遞屬性值的發行版本 5 SJWC 取代它們。有關詳細資訊，請參閱「[適用於 UNIX 的 Java Enterprise System 5 安裝指南](#)」。

- **Common Agent Container (CAC)**。只有在安裝了 Sun Cluster、Sun Cluster Geographic Edition 或 Sun Cluster Agents 之後才可以安裝版本 1.1。選取 [所有共用元件] 選項之後不會安裝它。在此情況下只會安裝版本 2.0。

升級 Java ES 共用元件

本章提供有關將 Java ES 共用元件升級到 Java ES 5 (發行版本 5) 的資訊。

每個 Java ES 產品元件均依賴一個或多個稱為 Java ES 共用元件的本機共用程式庫。依所安裝的產品元件而定，共用元件在產品元件安裝期間由 Java ES 安裝程式自動安裝。在部署 Java ES 產品元件的過程中，不會明確選取、安裝或配置它們。

同樣的，對於可以使用 Java ES 安裝程式升級的產品元件來說，對應的共用元件也會自動升級。

但是，對於手動升級產品元件的情況 (如本「升級指南」的多章中所述)，就可能需要按照本章所述程序明確執行共用元件的升級。

本章包含以下各節：

- [第 60 頁的「共用元件升級簡介」](#)
- [第 63 頁的「共用元件升級程序」](#)
- [第 63 頁的「特殊升級程序」](#)

執行共用元件升級之前，請務必閱讀[第 63 頁的「特殊升級程序」](#)。

共用元件升級簡介

將共用元件升級到 Java ES 5 (發行版本 5) 應做為大型升級規劃的一部份來執行，如第 1 章「升級規劃」中所述。為確保升級成功，請仔細閱讀第 1 章並制訂符合您需求的升級規劃。

本節涵蓋下列主題：

- [一般注意事項](#)
- [同步化共用元件](#)
- [同步化所有共用元件](#)
- [Solaris 10 區域注意事項](#)

一般注意事項

當升級共用元件時，請考慮以下問題：

- **作業系統問題。** 如第 40 頁的「作業系統注意事項」中所述，執行作業系統升級。對於除 Solaris 10 作業系統以外的所有平台，請在升級共用元件之前，先執行作業系統升級。
- **升級順序指導原則。** 檢視第 51 頁的「升級順序指導原則」中所列的升級順序指導原則。通常是先升級共用元件。不過，在開始升級程序前，應瞭解升級到 Java ES 發行版本 5 的整個順序。

同步化共用元件

要測試與支援 Java ES 共用元件與 Java ES 產品元件之間大量 (大約 30 個) 的複雜互動較為困難，其必須將單一作業系統實例中的所有共用元件都同步為同一 Java ES 版本。作業系統實例是指執行 Solaris 9 或 Linux 作業系統的單一電腦，或者在使用 Solaris 10 作業系統的情況下，它是指任何在單一電腦上執行的虛擬作業系統環境 (區域)。

也就是說，安裝在作業系統實例中的所有 Java ES 共用元件都必須是同一版本。此同步化需求會對 Java ES 共用元件的安裝及升級方式設定某些限制：

- 不同版本的 Java ES 共用元件只能位於不同的作業系統實例中。例如，您可以在一個作業系統實例中安裝 Java ES 發行版本 4 共用元件，並在另一個作業系統實例中安裝 Java ES 發行版本 5 共用元件，但不能將它們結合在同一作業系統實例中。

- 如果升級作業系統實例中的任何共用元件或推出新的更高版本的共用元件，那麼也必須同時升級該作業系統實例中的所有共用元件。(共用元件需要向下相容，使發行版本 4 產品元件與發行版本 5 共用元件可以搭配使用，不會出現問題。)

例如，假設將發行版本 5 產品元件安裝在一或多個發行版本 4 產品元件所在的作業系統實例中。由於發行版本 5 產品元件需要一些發行版本 5 共用元件，因此同步化需求是指在安裝發行版本 5 產品元件的同時，也必須將位於該作業系統實例中的所有發行版本 4 共用元件升級到發行版本 5。(即使正在安裝的發行版本 5 產品元件需要與已經安裝的共用元件不同的共用元件，也會發生這種情況。)

同樣的，如果要將發行版本 4 產品元件升級到發行版本 5，且該升級需要升級一些它所依賴的共用元件，則必須將安裝在該作業系統實例中的所有共用元件升級到發行版本 5，無論正在升級的特殊發行版本 4 產品元件是否對所有這些共用元件都有相依性。

同步化所有共用元件

Java ES 安裝程式包括一個同步化所有共用元件的功能，其可用於所有共用元件必須與它們的發行版本 5 版本同步的情況下。

當在 Java ES 安裝程式的元件選取頁面中選取了 [所有共用元件] 時，安裝程式將會升級任何現有的共用元件並安裝任何遺失的共用元件，無論特定產品元件是否需要它們皆是如此。

用於同步化所有共用元件的功能支援 Java ES 安裝程式未明確升級的產品元件 (即除 Application Server、Message Queue、HADB 和 Java DB 以外的所有 Java ES 產品元件) 的升級。例如，用於同步化所有共用元件的功能可用來將 Web Proxy Server 與 Portal Server 升級到發行版本 5。

在這種情況下同步化**所有**共用元件的理由是安裝程式目前對在電腦上需要同步化哪些共用元件一無所知。因此此功能會安裝**所有** Java ES 共用元件或將其升級到發行版本 5 版本。

用於同步化所有共用元件的功能也支援一些區域方案，如下面的「[Solaris 10 區域注意事項](#)」所述。

Solaris 10 區域注意事項

阻礙在 Solaris 10 多重區域環境中部署 Java ES 的一些限制源自共用元件的注意事項。

這些注意事項中最重要的一點是，由於稀疏根區域中的唯讀檔案系統而使許多共用元件無法安裝在稀疏根區域中。此限制適用於其基底目錄為 /usr (依預設由全域區域共用的目錄) 的共用元件。

在稀疏根區域中無法安裝一些 Java ES 共用元件，這表示若要成功地將對此類共用元件有相依性的產品元件安裝或升級到稀疏根區域中，必須先在傳遞共用元件到非全域區域時的起源全域區域中安裝或升級共用元件。

由於與共用元件相關的同步化需求的關係 (請參閱第 60 頁的「同步化共用元件」)，以及在全域區域中安裝或升級共用元件也必須配合在稀疏根區域中安裝或升級的任何產品元件，因此必須將全域區域中的所有共用元件與其發行版本 5 版本同步。這表示要升級任何現有共用元件及安裝任何遺失的共用元件，無論稀疏根區域中的任何特定產品元件是否需要它們。

這個在全域區域中同步化共用元件的動作可藉由在 Java ES 安裝程式的元件選取頁面中選取 [所有共用元件] 來執行。

將共用元件安裝到全域區域並從中傳遞它們時 (例如透過在全域區域中安裝 Java ES 產品元件)，必須特別小心地維持所有區域中的共用元件同步化。否則非全域區域中較早版本的共用元件可能會與已從全域區域中傳遞的發行版本 5 共用元件混合。

升級共用元件的方法

有兩種可能的方法可以升級特殊 Java ES 產品元件升級所需的共用元件。其中一個方法是決定所有需要的共用元件，並將其手動安裝或升級到其發行版本 5 版本。另一個方法是使用 Java ES 安裝程式的同步化 [所有共用元件] 功能。

但是，由於與共用元件有關的同步化需求的關係 (請參閱第 60 頁的「同步化共用元件」)，唯一可行的方法是使用 Java ES 安裝程式的同步化 [所有共用元件] 功能。

這是因為在大多數情況下，要決定在任何特殊情況下需要安裝或升級哪些共用元件很困難，而即使能夠成功完成此動作，手動升級所需共用元件也不是那麼簡單的。有些共用元件可以修補至發行版本 5 版本，但某些則需要用發行版本 5 套裝軟體來替代先前的套裝軟體。視升級路徑而定，可能需要手動移除某些先前的套裝軟體。

過去，手動升級共用元件的困難性為升級產品元件創造了明顯的障礙。儘管有安裝可能不需要的共用元件的缺點，相對於在 Java ES 發行版本 3 和發行版本 4 中需要手動升級共用元件，同步化 [所有共用元件] 功能依然代表一個明顯的改進。

共用元件升級程序

升級共用元件的程序就是使用 Java ES 安裝程式的同步化 [所有共用元件] 功能。

1. 請以超級使用者身份登入或成爲超級使用者。

```
su -
```

2. 如第 63 頁的「特殊升級程序」所述關閉服務或程序。
3. 啓動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch  
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示爲 [可升級]。)

4. 在元件選取頁面中選取 [所有共用元件]。
5. 確認您的選擇。

會將所有共用元件與其發行版本 5 版本同步。

6. 結束 Java ES 安裝程式。

關於任何後續程序，請參閱第 63 頁的「特殊升級程序」。

特殊升級程序

本節提供升級下列共用元件所需的特殊程序：

- 第 63 頁的「Java SE 升級程序」
- 第 66 頁的「Common Agent Container 升級程序」

Java SE 升級程序

Java ES 發行版本 5 已經過認證，可用於 Java Platform, Standard Edition (Java SE) 版本 5.0 Update 9 (在此寫作 Java SE 5.0 Update 9)。(Java SE 5.0 有時稱爲開發者版本 1.5.0)。

如同其他共用元件，Java SE 也是由 Java ES 安裝程式進行升級的。但無論是否安裝或升級其他共用程式，安裝程式都會將它升級到 Java SE 5.0 Update 9。您不需要為要安裝的 Java SE 5.0 Update 9 明確選擇 [所有共用元件]。

備註 升級 Java SE 時，可能要先關閉任何依賴於目前安裝之 Java SE 的服務。這是為了要避免發生任何可能因那些服務所產生的任何問題。如果您不關閉依賴於 Java SE 的服務，應於將 Java SE 升級到發行版本 5 版本之後重新啟動系統。

當安裝程式偵測到您電腦上有較舊版本的 Java SE 套裝軟體或不完整的套裝軟體集時 (完整集合為：SUNWj5rt、SUNWj5rtx、SUNWj5dev、SUNWj5dmo、SUNWj5dvr、SUNWj5man、SUNWj5cfg、SUNWj5dmx)，它會向您顯示一個對話方塊。該對話方塊跟隨在元件選取頁面之後，您可以在此選擇自動將 Java SE 升級到發行版本 5 層級，或略過該自動升級，而選擇以手動方式執行升級。

- **手動升級。**如果您選擇執行 Java SE 的手動升級，請使用下列程序：
 - a. 在安裝或升級其他元件之前結束 Java ES 安裝程式。
 - b. 手動安裝適當版本的 Java SE。
 - c. 重新啟動 Java ES 安裝程式。

如果安裝程式偵測到正確 / 完整的 Java SE 版本，它便會讓您繼續進行，否則會顯示上一個對話方塊。

- **自動升級。**如果您選擇執行 Java SE 的自動升級，Java ES 安裝程式會將 Java SE 升級到版本 5.0 Update 9。升級的運作方式如下：
 - 升級並不會移除先前安裝的主要 Java SE 發行版本 (例如 Java SE 版本 1.4.2 或 Java SE 版本 1.6.x)，因為其他應用程式可能會依賴於該版本。但是，升級會將下表中顯示的符號連結設定為參照發行版本 5 版本。

表 2-1 依平台而定的 Java SE 的符號連結

平台	符號連結	位置
Solaris	/usr/jdk/entsys-j2se	/usr/jdk/instances/jdk1.5.0
Linux	/usr/jdk/entsys-j2se	/usr/java/jdk1.5.0_09

對於需要較早版本的服務，您應該維持指向主要 Java SE 發行版本 (例如 Java SE 1.4.2) 的任何指標。如需有關如何維持對較早版本 Java SE 的符號連結的資訊，請查看相應的產品元件文件。

- 升級會移除先前安裝的次要 Java SE 發行版本 (例如 Java SE 版本 5.0 Update 5) 並用 Java SE 5.0 Update 9 替代它。

檢查 Java SE 符號連結

Java Enterprise System 維持了對於 Java SE 平台的支援版本的符號連結，以確保 Java ES 服務可以找到正確的 Java SE 執行階段來使用。

檢查符號連結 (例如在 Solaris 作業系統上)，如下所示：

```
ls -l /usr/jdk/entsys-j2se
lrwxrwxrwx 1 root  other 7 Jul 7 23:18 /usr/jdk/entsys-j2se ->
  /usr/jdk/instances/jdk1.5.0
```

其中 /usr/jdk/instances/jdk1.5.0 是預設位置。

驗證目前的 Java SE 版本

若要判斷 Java ES 安裝所使用的是哪個版本的 Java SE，請執行下列指令，其可驗證 Java SE 符號連結所參照的 Java SE 版本：

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

下表顯示其輸出。

表 2-2 Java SE 版本驗證輸出

Java ES 發行版本	Java SE 版本編號	Java SE 版本字串
發行版本 2	1.4.2 Update 5	1.4.2_05
發行版本 3	5.0 Update 1	1.5.0_01
發行版本 4	5.0 Update 4	1.5.0_04
發行版本 5	5.0 Update 9	1.5.0_09

Common Agent Container 升級程序

如果您要升級 Common Agent Container 共用元件以爲升級 Sun Cluster 軟體 (即修補版本 1.1) 做準備，請遵循「Sun Cluster 軟體安裝指南 (適用於 Solaris 作業系統)」的「升級 Sun Cluster 軟體」

(<http://docs.sun.com/doc/819-2057/6n4cet06q?a=view>) 中升級相依性軟體之程序。

如需對上述指南的更新，請參閱「Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions」(<http://docs.sun.com/doc/819-4351>) 中的「Upgrading to Sun Cluster 3.1 8/05 Software」。

如果您要升級 Common Agent Container 共用元件以爲升級其他 Java ES 元件做準備 (即升級到版本 2.0)，請使用下列程序，路徑名稱變數如下：

表 2-3 Common Agent Container 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>rel4CACbase-dir</i>	/opt/SUNWcacao	/opt/sun/cacao
<i>rel5CAC-admin-dir</i>	/usr/lib/cacao	/opt/sun/cacao

1. 如果目前安裝使用自訂配置設定，請使用下列指令擷取配置設定：

```
rel4CAC-base-dir/bin/cacaoadm list-params
```

輸出將與以下內容類似：

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

以上範例顯示的是預設值。請注意在 [第 67 頁的步驟 4](#) 中使用的是非預設設定。

2. 使用下列指令停止 Common Agent Container 程序：

```
rel4CAC-base-dir/bin/cacaoadm stop
echo $?
```

若結束代碼不是 0，則強制停止：

```
rel4CAC-base-dir/bin/cacaoadm stop -f
```

3. 使用 Java ES 安裝程式的同步化 [所有共用元件] 功能來升級 Common Agent Container 。

請參閱第 63 頁的「共用元件升級程序」。

4. 套用之前在第 66 頁的步驟 1 中擷取的自訂配置設定。

```
rel5CAC-admin-dir/bin/cacaoadm set-param java-flags=Value
rel5CAC-admin-dir/bin/cacaoadm set-param jmxmp-connector-port=Value
rel5CAC-admin-dir/bin/cacaoadm set-param snmp-adaptor-port=Value
rel5CAC-admin-dir/bin/cacaoadm set-param
    snmp-adaptor-trap-port=Value
rel5CAC-admin-dir/bin/cacaoadm set-param
    commandstream-adaptor-port=Value
rel5CAC-admin-dir/bin/cacaoadm set-param retries=Value
```

5. 若已將 Java SE 升級到 Java SE 版本 5，請執行 rebuild-dependencies 公用程式：

```
rel5CAC-admin-dir/bin/cacaoadm rebuild-dependencies
```

此指令的輸出將是：

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

6. 重新啟動 Common Agent Container 服務：

```
rel5CAC-admin-dir/bin/cacaoadm stop
rel5CAC-admin-dir/bin/cacaoadm start
```

7. 驗證 Common Agent Container 的升級：

```
rel5CAC-admin-dir/bin/cacaoadm status
rel5CAC-admin-dir/bin/cacaoadm verify-configuration
```


Sun Cluster 軟體

本章描述如何將 Sun Cluster 軟體升級到 Java ES 5 (發行版本 5) : Sun Cluster 3.1 8/05 軟體。

本章簡單介紹將 Sun Cluster 軟體升級到發行版本 5 的升級注意事項。

僅 Solaris 平台支援 Sun Cluster 軟體。

本章描述的 Sun Cluster 軟體的升級包括 Sun Cluster 架構軟體和 Sun Cluster 資料服務軟體或代理程式。

- [第 70 頁的「Sun Cluster 軟體升級簡介」](#)
- [第 74 頁的「將 Sun Cluster 軟體升級到 Java ES 發行版本 5」](#)

Sun Cluster 軟體升級簡介

本節描述下列會影響 Sun Cluster 軟體升級到 Java ES 5 (發行版本 5) 的一般層面：

- [關於發行版本 5 Sun Cluster 軟體](#)
- [Sun Cluster 軟體升級資訊指南](#)
- [Sun Cluster 資料](#)
- [Sun Cluster 升級策略](#)

關於發行版本 5 Sun Cluster 軟體

相對於發行版本 4 Sun Cluster 軟體，發行版本 5 Sun Cluster 軟體只有少量的升級變更 (請參閱「Sun Cluster 版本說明」

(<http://docs.sun.com/doc/819-2085/6n4d9jfgc?a=view>)。發行版本 5 Sun Cluster 除了不支援 Solaris 8 作業系統以外，基本上與發行版本 4 相同。

關於 Sun Cluster 3.1 8/05 文件集的變更與新增內容，請參閱「Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions」

(<http://docs.sun.com/doc/819-4351>)。另外，Sun Cluster 3.1 8/05 文件集中的程序對發行版本 5 Sun Cluster 軟體也有效。

備註

如果您需要在 Solaris 8 作業系統中使用 Sun Cluster 3.1 8/05 軟體，您必須從 Java ES 發行版本 4 發行軟體中取得 Sun Cluster 3.1 8/05 軟體，它位於 <http://www.sun.com/software/javaenterprisesystem/previous/index.xml>。

Java ES 5 安裝程式不再支援在 Solaris 8 作業系統上安裝 Sun Cluster 3.1 8/05 軟體。

若要在 Solaris 8 作業系統上安裝 Sun Cluster 3.1 8/05 軟體，請執行下列步驟：

1. 使用 Java ES 發行版本 4 發行軟體安裝 Sun Cluster 3.1 8/05 軟體。
 2. 安裝 Sun Cluster 3.1 8/05 軟體的所有必要修補程式。
-

Sun Cluster 軟體升級資訊指南

表 3-1 顯示了所支援的升級到 Java ES 5 (發行版本 5) 的 Sun Cluster 升級路徑。該表僅適用於 Solaris 作業系統。

Sun Cluster 版本編號並不與 Java ES 發行版本編號相對應。在過去，Sun Cluster 軟體的臨時功能發行版本 (IFR) 在正式發行各個 Java ES 發行版本時已經整合至 Java ES。因此，將 Java ES 發行版本 2、發行版本 3 和發行版本 4 Sun Cluster 升級到發行版本 5 Sun Cluster (如表 3-1 所示) 包括將 Sun Cluster 3.1 4/04、Sun Cluster 3.1 9/04 和 Sun Cluster 3.1 8/05 軟體升級到發行版本 5。

表 3-1 至 Java ES 5 (發行版本 5) 的升級路徑：Sun Cluster 3.1 8/05 軟體

Java ES 發行版本	Sun Cluster 軟體版本	一般方法	需要重新配置
發行版本 4	Sun Cluster 3.1 8/05 (2005Q4)	直接升級 ¹ ： 使用 Sun Cluster <code>scinstall</code> 公用程式來執行。	無。
發行版本 3	Sun Cluster 3.1 9/04 或 Sun Cluster 3.1 8/05 (IFR)	直接升級： 使用 Sun Cluster <code>scinstall</code> 公用程式來執行。	自動遷移到升級後版本的叢集配置。 將 IFR 升級到發行版本 5 時不需要進行重新配置。
發行版本 2	Sun Cluster 3.1 4/04 或 Sun Cluster 3.1 9/04 (IFR)	直接升級： 使用 Sun Cluster <code>scinstall</code> 公用程式來執行。	自動遷移到升級後版本的叢集配置
發行版本 1	Sun Cluster 3.1	尚未認證直接升級： 但可使用 <code>scinstall</code> 公用程式來執行。	自動遷移到升級後版本的叢集配置
Java ES 發行版本之前的發行版本	Sun Cluster 3.0	尚未認證直接升級： 但可使用 <code>scinstall</code> 公用程式來執行。	自動遷移到升級後版本的叢集配置

1. 除非已將 Sun Cluster 軟體使用的共用元件升級到發行版本 5，否則不需要從發行版本 4 升級到發行版本 5。

Sun Cluster 資料

下表顯示可能受升級 Sun Cluster 軟體影響的資料類型。

表 3-2 Sun Cluster 資料用途

資料類型	位置	用途
叢集配置資料	叢集配置儲存庫在所有叢集節點上複製和同步化 (注意：永遠不要手動編輯 CCR 檔案，這會造成一個節點或整個叢集停止運作)	儲存 Sun Cluster 作業所有層面的配置資訊：叢集節點配置、容錯轉移機制、資源管理等等

Sun Cluster 升級策略

Sun Cluster 的升級策略一般取決於在 [第 1 章「升級規劃」](#) 中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Sun Cluster 升級規劃的問題，藉以對 Sun Cluster 進行一般性的介紹。

相容性問題

Java ES 發行版本 5 Sun Cluster 軟體包括新的圖形化管理介面，但向下相容較早發行版本的 Sun Cluster 代理程式。

Sun Cluster 相依性

Sun Cluster 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Sun Cluster 軟體的程序。例如，Sun Cluster 介面或功能上的變更可能需要 Sun Cluster 軟體相依之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Sun Cluster 對下列 Java ES 元件有相依性：

- **共用元件**。 Sun Cluster 軟體對特定 Java ES 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))。
- **資料服務**。 Sun Cluster 軟體需有特定資料服務 (或代理程式) 才能使 Java ES 產品元件具有高可用性。對於在 Sun Cluster 環境中執行的每個產品元件，都必須有對應的資料服務來管理對應的叢集資源。代理程式套裝軟體的升級通常是 Sun Cluster 升級程序的一部份。

雙重升級

在雙重升級中，Sun Cluster 軟體與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可以在 Sun Cluster 非漸進式升級環境中執行，但不能做為漸進式升級的一部份來執行。

有關程序的詳細資訊，可在「[Sun Cluster 安裝指南](#)」關於升級的章節 (<http://docs.sun.com/doc/819-2057/6n4cet070?a=view>) 中找到。在「[Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions](#)」 (<http://docs.sun.com/doc/819-4351>) 中記述了針對發行版本 5 Sun Cluster 所做的修改。

該程序適用於將 Solaris 作業系統從 Solaris 8 或 Solaris 9 升級到 Solaris 10。

將 Sun Cluster 軟體升級到 Java ES 發行版本 5

本節包含將 Sun Cluster 軟體從 Java ES 2005Q4 (發行版本 4)、Java ES 2005Q1 (發行版本 3) 和 Java ES 2004Q2 (發行版本 2) 升級到 Java ES 5 (發行版本 5) 的相關資訊。對於在以下這些 Java ES 發行版本中找到的三個 Sun Cluster 版本，升級程序皆相同：Sun Cluster 3.1 4/04、Sun Cluster 3.1 9/04 和 Sun Cluster 3.1 8/05 軟體。

如果您已經安裝了 Sun Cluster 3.1 8/05 (發行版本 3 或發行版本 4) 軟體，除非已將 Sun Cluster 軟體使用的共用元件升級到發行版本 5，否則您不需要升級到發行版本 5。

本節涵蓋以下主題：

- [簡介](#)
- [Sun Cluster 升級](#)

簡介

將 Sun Cluster 軟體升級到 Java ES 發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 藉由執行 `scinstall` 程序檔執行升級，該程序檔可升級 Sun Cluster 軟體，並在軟體升級完成後套用先前的 Sun Cluster 配置。但是，叢集環境中的所有節點皆必須升級到相同版本，方式可以是藉由關閉叢集再升級所有節點，或是透過漸進式升級同時將節點連續升級而不關閉叢集。
- **升級相依性。** 雖然 Sun Cluster 軟體對一些 Java ES 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))，但這代表非硬性升級相依性：Java ES 發行版本 5 Sun Cluster 軟體與這些元件的發行版本 4 版本相容。
- **向下相容性。** 發行版本 5 Sun Cluster (架構) 軟體向下相容較早的叢集代理程式，且發行版本 5 Sun Cluster 代理程式軟體向下相容 Java ES 發行版本 4 元件。但是叢集中的所有節點必須執行相同版本的架構和代理程式軟體。
- **升級回復。** 不支援將 Sun Cluster 軟體的發行版本 5 升級回復至較早的版本。
- **平台問題。** 升級 Sun Cluster 軟體的方法在所有 Solaris 平台及硬體架構上均相同，但 Linux 平台不支援 Sun Cluster 軟體。

Sun Cluster 升級

本節概要介紹如何將 Sun Cluster 軟體從 Java ES 發行版本 4 升級到 Java ES 發行版本 5：

- 升級前作業
- 升級 Sun Cluster 軟體
- 驗證升級
- 升級後作業
- 回復升級

本節涵蓋 Sun Cluster 非漸進式升級的情況。漸進式升級的情況稍有不同，其叢集不會關閉。然而兩種情況皆牽涉到指定叢集節點之相同的一般升級程序，如下所述。具體程序可在「Sun Cluster 安裝指南」關於升級的一章 (<http://docs.sun.com/doc/819-2057/6n4cet06q?a=view>) 中找到。在「Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions」 (<http://docs.sun.com/doc/819-4351>) 中記述了針對發行版本 5 Sun Cluster 所做的修改。

升級前作業

升級 Sun Cluster 軟體之前，應該執行下列作業：

- 驗證目前版本資訊
- 關閉 Sun Cluster Geographic Edition 基礎架構
- 準備欲升級的叢集節點
- 升級作業系統
- 升級 Sun Cluster 相依元件
- 取得必要的配置資訊和密碼

驗證目前版本資訊

您可輸入下列指令驗證 Sun Cluster 軟體的目前版本：

```
scinstall -pv
```

該指令會傳回 Sun Cluster 版本和每個安裝的套裝軟體的版本。如果此指令傳回 3.1 8/05 版本 3.1u4，則請檢查修補程式的修訂版號碼，看看您擁有的是發行版本 4 還是發行版本 5 軟體，如下表所述。

表 3-3 Sun Cluster 版本驗證輸出

Java ES 發行版本	Sun Cluster 版本編號	修補程式的修訂版號碼
發行版本 1 (Sun Cluster 3.1)	3.1	
發行版本 2 (Sun Cluster 3.1 4/04)	3.1u2	
發行版本 2 或 3 (Sun Cluster 3.1 9/04)	3.1u3	
發行版本 3 或 4 (Sun Cluster 3.1 8/05)	3.1u4	Solaris 9 SPARC : 117949-15 Solaris 8 SPARC : 117950-15 Solaris 9 x86 : 117909-15
發行版本 5 (Sun Cluster 3.1 8/05)	3.1u4	Solaris 10 SPARC : 120500-08 ¹ Solaris 9 SPARC : 117949-23 Solaris 8 SPARC : 117950-23 Solaris 10 x86 : 120501-08 Solaris 9 x86 : 117909-23

1. 修補程式的修訂版號碼是針對不同平台升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

關閉 Sun Cluster Geographic Edition 基礎架構

如果您要對執行 Sun Cluster Geographic Edition 軟體的叢集進行升級，必須先關閉 Sun Cluster Geographic Edition 基礎架構，並執行其他步驟以使叢集做好升級準備。請遵循「Sun Cluster Geographic Edition Installation Guide」(<http://docs.sun.com/doc/819-8004/6n9tmd19d?=view>) 中的升級程序。這些程序包含在適當升級階段升級 Sun Cluster 軟體的步驟。

準備欲升級的叢集節點

必須先移除叢集環境中的叢集節點，才能升級 Sun Cluster 軟體：

- **非漸進式升級。** 移除叢集環境中的節點即表示關閉叢集環境：將資源群組切換成離線，停用它們，關閉在叢集環境中執行的應用程式，備份共用資料，關閉該叢集，備份系統磁碟，然後重新啓動節點，進入非叢集模式。
- **漸進式升級。** 移除叢集環境中的節點表示移除節點上的所有資源群組和裝置群組，備份共用資料和系統磁碟，然後重新啓動節點，進入非叢集模式。

可在「Sun Cluster 安裝指南」關於升級的一章

(<http://docs.sun.com/doc/819-2057/6n4cet06q?a=view>) 中找到這些作業以及在特定情況下可能需要執行的其他作業之詳細資訊。

升級作業系統

您可能想利用升級的停機時間將作業系統升級為最新的版本，還要升級您使用的磁碟區管理器。

「Sun Cluster 安裝指南」關於升級的一章

(<http://docs.sun.com/doc/819-2057/6n4cet06q?a=view>) 中提供了這些作業的詳細資訊。

升級 Sun Cluster 相依元件

一般建議將電腦系統 (和運算環境) 中的所有 Java ES 元件都升級到發行版本 5。升級 Sun Cluster 軟體相依的發行版本 4 共用元件是選擇性的，且可依照第 2 章「升級 Java ES 共用元件」中的說明執行 (將發行版本 2 共用元件升級到發行版本 5 則是必要的動作)。

備註

若已將 Java ES 共用元件與發行版本 5 同步，則 Sun Cluster 軟體也必須升級到發行版本 5。這是因為發行版本 4 Sun Cluster 與發行版本 5 Sun Java Web Console 共用元件不相容。

若 Sun Java Web Console 已經升級到發行版本 5 (版本 3.0)，則應依本章所述套用最新的 Sun Cluster 修補程式或升級 Sun Cluster。

取得必要的配置資訊和密碼

不需要關於目前安裝版本的特殊資訊。不過，您需要以超級使用者的身份登入才能執行升級。

升級 Sun Cluster 軟體

本節討論影響 Sun Cluster 軟體升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Sun Cluster 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 升級 Sun Cluster 架構軟體時，最好是升級管理具備高可用性的 Java ES 元件時所需的資料服務，以及其他在您的叢集環境中執行的應用程式。
- 升級 Sun Cluster 軟體亦會提供一個升級 Java ES 元件或其他在您的叢集環境中執行的應用程式的機會。

升級程序

下列程序適用於升級每個叢集節點上的 Sun Cluster 軟體。這些是概略步驟，「Sun Cluster 安裝指南」關於升級的一章

(<http://docs.sun.com/doc/819-2057/6n4cet06q?a=view>) 中提供了有關如何執行這些步驟的詳細資訊。在「Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions」(<http://docs.sun.com/doc/819-4351>) 中記述了針對發行版本 5 Sun Cluster 所做的修改。

1. 開機進入非叢集模式。
2. 請以超級使用者身份登入或成為超級使用者。
3. 變更為 Java ES 發行版本 5 發行軟體上的下列目錄：

```
su -
```

```
cd /os_arch/Product/sun_cluster/os-ver/Tools
```

其中 *os_arch* 符合您的平台，例如 *Solaris_sparc*，*os-ver* 是 Solaris 9 或 Solaris 10。

4. 執行 `scinstall` 公用程式。
5. 升級 Sun Cluster 架構軟體和任何需要的資料服務。

```
./scinstall
```

會顯示一個主功能表，可用於執行叢集安裝、配置及升級作業。

需要將資料服務對應的資源遷移到升級後的資源類型，以配置升級後的資料服務 (請參閱第 79 頁的「升級後作業」)。

- 將任何必需的修補程式套用到 Sun Cluster 架構軟體和資料服務中。
在「Sun Cluster 3.0-3.1 Release Notes Supplement」
(<http://docs.sun.com/app/docs/doc/816-3381/6m9lratq9?a=view#gcpom>) 中提供了有關存取和套用相關修補程式的資訊。
- 重新啟動節點，進入叢集。

驗證升級

您可按以下所述驗證 Sun Cluster 軟體的升級是否成功：

- 請輸入下列指令。

```
scinstall -pv
```

該指令會傳回 Sun Cluster 版本和每個安裝的套裝軟體的版本。如果此指令傳回 3.1 8/05 版本 3.1u4，則請檢查修補程式的修訂版號碼，看看您擁有的是發行版本 4 還是發行版本 5 軟體，如第 76 頁的表 3-3 所述。

- 檢查資料服務升級記錄檔。

在升級輸出訊息的最後即參照記錄檔。

升級後作業

執行 Sun Cluster 軟體的升級後，可能需要執行一些額外的作業，這要依您執行的是非漸進式升級還是漸進式升級而定。完全復原您的叢集環境所需的作業是：

- 驗證叢集配置的狀態
- 將資源遷移到新的資源類型版本
- 升級在叢集上安裝的其他 Java ES 元件或應用程式

「Sun Cluster 安裝指南」關於升級的一章

(<http://docs.sun.com/doc/819-2057/6n4cet06q?a=view>) 中提供了這些安裝後步驟的詳細資訊。

回復升級

不支援回復 Sun Cluster 軟體。無法輕易復原在升級過程中所做的變更。

將 Sun Cluster 軟體升級到 Java ES 發行版本 5

Sun Cluster Geographic Edition

本章描述如何將 Sun Cluster Geographic Edition 升級到 Java ES 5 (發行版本 5) : Sun Cluster Geographic Edition 3.1 2006Q4。

本章簡要介紹發行版本 5 支援的不同升級路徑之升級考量因素。本章僅涵蓋在 Solaris 作業系統上的升級作業：

- [第 82 頁的「Sun Cluster Geographic Edition 升級簡介」](#)
- [第 85 頁的「從版本 3.1 8/05 升級 Sun Cluster Geographic Edition」](#)

備註 本章中的檔案位置是以稱為 *SunClusterGeo-base* 的目錄路徑為基準指定的。安裝 Sun Cluster Geographic Edition 時可能至少已將此路徑中的一部份指定為安裝目錄。若沒有指定，Geographic Edition 安裝程式會指定一個預設值。

此目錄路徑的預設值顯示在下表中。

表 4-1 Sun Cluster Geographic Edition 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>SunClusterGeo-base</i>	/opt/SUNWscgeo	不適用

Sun Cluster Geographic Edition 升級簡介

本節描述下列會影響 Sun Cluster Geographic Edition 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5](#)
- [Java ES 發行版本 5 升級資訊指南](#)
- [Sun Cluster Geographic Edition 資料](#)
- [Sun Cluster Geographic Edition 升級策略](#)

關於 Java ES 發行版本 5

Java ES 發行版本 5 Sun Cluster Geographic Edition 是做為 Java ES 元件傳送的第一個發行版本；Sun Cluster Geographic Edition 3.1 是最早做為獨立產品發行的。

相對於 Sun Cluster Geographic Edition 3.1 8/05，發行版本 5 Sun Cluster Geographic Edition 只有少量的變更。它是第一個支援 Solaris x86 平台的發行版本。發行版本 5 Sun Cluster Geographic Edition 還包含一些所選錯誤修正程式，適用於各種硬體與軟體元件，且支援其他資料複製產品。

Java ES 發行版本 5 升級資訊指南

[表 4-2](#) 顯示支援的升級到 Java ES 發行版本 5 的 Sun Cluster Geographic Edition 的升級路徑。此表僅適用於 Solaris 作業系統。

表 4-2 至 Java ES 5 (發行版本 5) 的升級路徑：Sun Cluster Geographic Edition 3.1 2006Q4

Java ES 發行版本	Sun Cluster Geographic Edition 版本	一般方法	需要重新配置
Java ES 發行版本之前的發行版本	Sun Cluster Geographic Edition 3.1 8/05	直接升級： 以全新安裝取代 Sun Cluster Geographic Edition 3.1 8/05	無

Sun Cluster Geographic Edition 資料

下表顯示可能受升級 Sun Cluster Geographic Edition 軟體影響的資料類型。

表 4-3 Sun Cluster Geographic Edition 資料用途

資料類型	位置	用途
Sun Cluster Geographic Edition 配置資料	叢集配置儲存庫在所有叢集節點上複製和同步化 (注意：永遠不要手動編輯 CCR 檔案，這會造成一個節點或整個叢集停止運作)	儲存 Sun Cluster Geographic Edition 作業所有層面的配置資訊。

Sun Cluster Geographic Edition 升級策略

Sun Cluster Geographic Edition 的升級策略一般取決於在第 1 章「升級規劃」中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Sun Cluster Geographic Edition 升級規劃的問題，藉以對 Sun Cluster Geographic Edition 進行一般性的介紹。

相容性問題

夥伴關係中的兩個叢集都必須執行相同版本的 Sun Cluster Geographic Edition。因此，發行版本 5 Sun Cluster Geographic Edition 不能與 Sun Cluster Geographic Edition 3.1 8/05 混合為夥伴關係 (沒有向下相容性)。發行版本 5 Sun Cluster Geographic Edition 包含其他無法由 Sun Cluster Geographic Edition 3.1 8/05 讀取的配置資料。

相依性

Sun Cluster Geographic Edition 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Sun Cluster Geographic Edition 軟體的程序。例如，Sun Cluster Geographic Edition 介面或功能上的變更可能需要 Sun Cluster Geographic Edition 相依之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Sun Cluster Geographic Edition 對下列 Java ES 元件有相依性：

- **共用元件。** Sun Cluster Geographic Edition 對特定的 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。
- **Sun Cluster。** Sun Cluster Geographic Edition 對 Sun Cluster 有強制性的相依性，後者提供了基本功能。

雙重升級

在雙重升級中，Sun Cluster Geographic Edition 軟體與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，而這種升級方法不適用於 Sun Cluster Geographic Edition。

由於 Sun Cluster Geographic Edition 對 Sun Cluster 有硬性升級相依性，因此作業系統升級是在 Sun Cluster 升級環境中執行的。如需有關 Sun Cluster 雙重升級的資訊，請參閱第 73 頁的「[雙重升級](#)」。

從版本 3.1 8/05 升級 Sun Cluster Geographic Edition

本節包含將 Sun Cluster Geographic Edition 從版本 3.1 8/05 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [版本 3.1 8/05 Sun Cluster Geographic Edition 升級](#)

簡介

將 Sun Cluster Geographic Edition 發行版本 3.1 8/05 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 升級可藉由解除安裝版本 3.1 8/05 及執行發行版本 5 Sun Cluster Geographic Edition 的全新安裝來達成。會保留配置資料。
- **升級相依性。** Sun Cluster Geographic Edition 對許多 Java ES 共用元件具有相依性 (請參閱第 43 頁的表 1-9)，當您升級 Sun Cluster Geographic Edition 時，Java ES 安裝程式會將所有這些共用元件自動升級到發行版本 5。特別的是，Sun Cluster Geographic Edition 對 Common Agent Container 版本 1.1 有硬性升級相依性。Sun Cluster Geographic Edition 對 Sun Cluster 也有硬性升級相依性：必須將發行版本 4 Sun Cluster (3.1 8/05) 升級到發行版本 5 才能支援發行版本 5 Sun Cluster Geographic Edition。
- **向下相容性。** 發行版本 5 Sun Cluster Geographic Edition 與 Sun Cluster Geographic Edition 3.1 8/05 不相容。夥伴關係中的所有叢集都必須升級到發行版本 5。
- **升級回復。** 不支援將 Sun Cluster Geographic Edition 軟體的發行版本 5 升級回復至版本 3.1 8/05。
- **平台問題。** 升級 Sun Cluster Geographic Edition 軟體的方法在所有 Solaris 平台及硬體架構上均相同，但 Linux 平台不支援 Sun Cluster Geographic Edition 軟體。

版本 3.1 8/05 Sun Cluster Geographic Edition 升級

本節說明如何將 Sun Cluster Geographic Edition 從版本 3.1 8/05 升級到 Java ES 發行版本 5。本節包含下列主題：

- [升級前作業](#)
- [升級 3.1 8/05 Sun Cluster Geographic Edition](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Sun Cluster Geographic Edition 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [為升級準備叢集](#)
- [升級 Sun Cluster Geographic Edition 相依元件](#)
- [備份 Sun Cluster Geographic Edition 資料](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可使用下列指令驗證 Sun Cluster Geographic Edition 的目前版本：

```
/usr/cluster/bin/geoadm --version
```

版權文字後面是版本字串，其顯示在下表中：

表 4-4 Sun Cluster Geographic Edition 版本驗證輸出

Java ES 發行版本	Sun Cluster Geographic Edition 版本編號
版本 3.1 8/05	1.0
發行版本 5	1.1

為升級準備叢集

對與您要升級的叢集有夥伴關係的所有叢集執行下列步驟。程序會從生產環境中移除 Sun Cluster Geographic Edition 層。

1. 確保叢集能夠正常運作。

若要檢視叢集的目前狀態，請在任何節點上執行下列指令。

```
scstat
```

2. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

3. 從保護群組中移除所有應用程式資源群組。

高可用性的應用程式在 Sun Cluster Geographic Edition 升級時不會當機。

```
geopg remove-resource-group resourcegroup protectiongroupname
```

4. 對與此叢集有夥伴關係的所有叢集執行步驟 1 至步驟 3。

5. 停止叢集上的所有使用中保護群組。

```
geopg stop protectiongroupname -e local | global
```

6. 從兩個夥伴叢集上的所有活動訊號中移除 ICRM 外掛程式。

```
geohp remove-plugin icrmpugin heartbeatname
```

升級 Sun Cluster Geographic Edition 相依元件

一般的建議是將電腦系統 (及運算環境中) 的所有 Java ES 元件都升級到 Java ES 發行版本 5。Sun Cluster Geographic Edition 對一些共用元件和 Sun Cluster 有硬性升級相依性。

升級 Sun Cluster Geographic Edition 相依元件時，在升級 Sun Cluster Geographic Edition 之前，應按照下列順序執行 (略過可能已升級過的任何相依元件)。共用元件的升級通常可由 Java ES 安裝程式自動完成。

1. **共用元件。** 第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Sun Cluster Geographic Edition 升級到發行版本 5 時，Sun Cluster Geographic Edition 所需要的所有共用元件會由 Java ES 安裝程式自動升級。

備註 若已將 Java ES 共用元件升級到發行版本 5，則必須將 Sun Cluster Geographic Edition 軟體也升級到發行版本 5。這是因為 Sun Cluster Geographic Edition 3.1 8/05 與發行版本 5 Sun Java Web Console 不相容。

2. **Sun Cluster。** 在第 69 頁的第 3 章「Sun Cluster 軟體」中提供了將 Sun Cluster 升級到發行版本 5 的指示。

備份 Sun Cluster Geographic Edition 資料

Sun Cluster Geographic Edition 會將所有資料儲存在叢集配置儲存庫中，因此不需要備份目前的資料。

取得必要的配置資訊和密碼

不需要關於目前安裝版本的特殊資訊。不過，您需要以超級使用者的身份登入才能執行升級。

升級 3.1 8/05 Sun Cluster Geographic Edition

本節討論影響 Sun Cluster Geographic Edition 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Sun Cluster Geographic Edition 軟體升級到 Java ES 發行版本 5 時應不會干擾應用程式的執行。您可以在執行中的叢集上升級 Sun Cluster Geographic Edition 軟體而不會遭到中斷；叢集會在服務執行中的情況下保留在生產環境中。同樣的，您也可以套用 Sun Cluster Geographic Edition 修補程式而不會當機。在整個升級程序中都會保留配置資料。

在與您要升級的叢集有夥伴關係的全部叢集所在之全部節點上，必須升級 Sun Cluster Geographic Edition 軟體。

如果您要在 Sun Cluster Geographic Edition 軟體升級程序中升級 Solaris 作業系統，必須先移除 Sun Cluster Geographic Edition 套裝軟體，然後才能升級 Solaris 作業系統。

升級程序

下述程序適用於位於執行升級的電腦本機上的 Sun Cluster Geographic Edition 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 關閉 Sun Cluster Geographic Edition 3.1 8/05。

```
/usr/cluster/bin/geoadm stop
```

若要在關閉 Sun Cluster Geographic Edition 期間使底層應用程式仍舊執行，請參閱「Sun Cluster Geographic Edition System Administration Guide」(<http://docs.sun.com/doc/819-8003>)。

3. 解除安裝 Sun Cluster Geographic Edition 3.1 8/05。

```
SunClusterGeo-base/install/uninstall/uninstaller
```

在解除安裝模式下使用 Sun Cluster Geographic Edition 安裝程式，如「Sun Cluster Geographic Edition Installation Guide」(<http://docs.sun.com/doc/819-8004>) 所述。

4. 如果 Sun Cluster 未在執行，請啓動它。

必須執行 Sun Cluster 才能安裝 Sun Cluster Geographic Edition。您可以使用 `scstat` 指令來對此進行檢查。

5. 使用 Java ES 安裝程式執行發行版本 5 Sun Cluster Geographic Edition 的全新安裝。

- a. 在代管發行版本 4 Access Manager 的電腦上啓動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 `os_arch` 與您所在平台相符，例如 `Solaris_sparc`。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 從元件選取頁面中選取 Sun Cluster Geographic Edition。
- c. 選擇 [立即配置] 或 [以後配置]。

您選擇 [立即配置] 或 [以後配置] 並沒有什麼差別，因為並不需要重新配置。

- d. 如有必要，請選取安裝本土化套裝軟體的選項。

- e. 安裝完成時結束 Java ES 安裝程式。

備註 如果您要在 Solaris 8 作業系統上升級 Sun Cluster Geographic Edition 軟體，則必須使用 `pkgadd` 指令從 Java ES 發行版本 5 發行軟體安裝軟體。Java ES 安裝程式不支援在 Solaris 8 作業系統上安裝 Sun Cluster Geographic Edition 軟體。

6. 安裝所有必要的 Sun Cluster Geographic Edition 修補程式。
如需說明，請參閱「Sun Cluster Geographic Edition Installation Guide」(<http://docs.sun.com/doc/819-8004>)。

7. 對與此叢集有夥伴關係的所有叢集執行第 89 頁的步驟 1 至步驟 6。

8. 重新啟動發行版本 5 Sun Cluster Geographic Edition 軟體。

```
/usr/cluster/bin/geoadm start
```

如果叢集處於夥伴關係當中，必須在 Sun Cluster Geographic Edition 軟體啟動前先升級兩個夥伴上的所有節點。

如需更多資訊，請查看上面所參照的「Installation Guide」以及「Sun Cluster Geographic Edition System Administration Guide」(<http://docs.sun.com/doc/819-8003>)。

9. 增加您在為叢集升級做準備時移除的所有應用程式資源群組，如第 87 頁的「為升級準備叢集」所述。

```
geopg add-resource-group resourcegroup protectiongroupname
```

10. 啟動您增加的所有保護群組。

```
geopg start protectiongroupname -e local | global [-n]
```

驗證升級

您可按以下所述驗證 Sun Cluster Geographic Edition 軟體的升級是否成功：

1. 執行以下指令：

```
/usr/cluster/bin/geoadm --version
```

請參閱第 86 頁的表 4-4 中的輸出值。

2. 如果適用，在升級兩個夥伴之後，請在每一個夥伴的一個叢集節點上執行下列指令。

```
/usr/cluster/bin/geoadm start  
/usr/cluster/bin/geoadm show
```

該指令將會顯示 Sun Cluster Geographic Edition 軟體在該節點上是否為使用中狀態。

升級後作業

除了在[第 89 頁](#)的「升級程序」中所述的步驟之外，沒有升級後作業。

回復升級

不支援回復 Sun Cluster Geographic Edition 軟體。

但是，您可以使用 Java ES 安裝程式的解除安裝功能來移除發行版本 5 Sun Cluster Geographic Edition，然後使用版本 3.1 8/05 安裝程式來重新安裝版本 3.1 8/05。由於發行版本 5 Sun Cluster Geographic Edition 會寫入版本 3.1 8/05 無法讀取的資料，因此必須在發行版本 5 啟動之前（也就是您啟動已經升級的 Sun Cluster Geographic Edition 軟體之前）執行此類回復，如[第 89 頁](#)的「升級程序」所述。

Directory Server

本章描述如何將 Directory Server 升級到 Java ES 5 (發行版本 5) : Sun Java System Directory Server 6.0。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 94 頁的「Directory Server 升級簡介」](#)
- [第 99 頁的「從 Java ES 發行版本 4 升級 Directory Server」](#)
- [第 107 頁的「從 Java ES 發行版本 3 升級 Directory Server」](#)
- [第 108 頁的「從 Java ES 發行版本 2 升級 Directory Server」](#)

備註 本章中的檔案位置是以稱為 *serverRoot* (Directory Server 5.x) 和 *DirServer-base* (Directory Server 6.0) 的目錄路徑為基準指定的。安裝 Directory Server 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 5-1 Directory Server 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>serverRoot</i> (Directory Server 5.x)	/var/opt/mps/serverroot	/var/opt/sun/directory-server
<i>DirServer-base</i> (Directory Server 6.0)	/opt/SUNWdsee	/opt/sun

Directory Server 升級簡介

本節描述下列會影響 Directory Server 升級到 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5](#)
- [Java ES 發行版本 5 升級資訊指南](#)
- [Directory Server 資料](#)
- [Directory Server 升級策略](#)

關於 Java ES 發行版本 5

Java ES 發行版本 5 Directory Server 代表一個重大發行版本，包含多種新功能與增強功能。有關詳細資訊，請參閱「Directory Server Enterprise Edition 6 版本說明」(<http://docs.sun.com/doc/820-0290>)。

Java ES 發行版本 5 升級資訊指南

表 5-2 顯示支援的 Directory Server 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 5-2 至 Java ES 5 (發行版本 5) 的升級路徑：Directory Server 6.0

Java ES 發行版本	Directory Server 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Directory Server 5.2 2005Q4 Sun Java System Administration Server 5.2 2005Q4	直接升級： 全新安裝及遷移所有資料。 由 Directory Service Control Center 與 Directory Server EE 指令行公用程式取代 Administration Server 功能。	將配置資料從舊版遷移至新安裝的 Directory Server
發行版本 3	Sun Java System Directory Server 5 2005Q1 Sun Java System Administration Server 5 2005Q1	直接升級： 全新安裝及遷移所有資料。 由 Directory Service Control Center 與 Directory Server EE 指令行公用程式取代 Administration Server 功能。	將配置資料從舊版遷移至新安裝的 Directory Server

表 5-2 至 Java ES 5 (發行版本 5) 的升級路徑：Directory Server 6.0 (續)

Java ES 發行版本	Directory Server 版本	一般方法	需要重新配置
發行版本 2	Sun Java System Directory Server 5.2 2004Q2 Sun Java System Administration Server 5.2 2004Q2	直接升級： 全新安裝及遷移所有資料。 由 Directory Service Control Center 與 Directory Server EE 指令行公用程式取代 Administration Server 功能。	將配置資料從舊版遷移至新安裝的 Directory Server
發行版本 1	Sun ONE Directory Server 5.2 Sun ONE Administration Server 5.2	尚未認證直接升級： 但可以使用從發行版本 2 升級時所使用的方法。	將配置資料從舊版遷移至新安裝的 Directory Server
Java ES 發行版本之前的發行版本	Sun ONE Directory Server 5.2 Sun ONE Administration Server 5.2 Sun ONE Directory Server 5.1 Sun ONE Administration Server 5.1	尚未認證直接升級： 但可以使用從發行版本 2 升級時所使用的方法。 無直接升級： 先升級到發行版本 3。請參閱「Java Enterprise System 2005Q1 升級與遷移指南」 (http://docs.sun.com/doc/819-2238)。 然後從發行版本 3 升級到發行版本 5。	將配置資料從舊版遷移至新安裝的 Directory Server 請參閱「Java Enterprise System 2005Q1 升級與遷移指南」 (http://docs.sun.com/doc/819-2238)

Directory Server 資料

Directory Server 5.x 版本利用 Directory Server 本身來儲存配置資料。資料儲存在目錄內的特定樹狀結構中。代管配置的 Directory Server 實例稱為配置目錄。配置目錄可與其他 Directory Server 實例位於相同的電腦上，不過，在大多數部署架構中，配置目錄對於使用它來儲存配置資訊的其他元件而言是遠端配置。

Directory Server 6.0 不再將配置資料儲存在配置目錄中。配置透過 Directory Service Control Center (或 Directory Server EE 指令行公用程式) 來執行，而且透過此介面進行存取。Directory Service Control Center 將配置資料儲存在它自己的本機 Directory Server 實例中。

下表顯示受 Directory Server 軟體升級到發行版本 5 影響的資料類型。

表 5-3 Directory Server 資料用途

資料類型	位置	用途
Directory Server 配置資料	Directory Server 5.1 與 5.2：配置目錄 Directory Server 6.0：透過 Directory Service Control Center 與 Directory Server EE 指令行公用程式存取	Directory Server 實例的配置
Directory Server 模式		定義目錄中資料的結構與語義
安全性資料	Directory Server 5.1 與 5.2：透過 Directory Server Console 配置的 SSL。 Directory Server 6.0：透過 Directory Service Control Center 與 Directory Server EE 指令行公用程式配置的 SSL。	伺服器憑證
使用者資料	Directory Server	支援含使用者特定配置資料與使用者設定檔的應用程式

Directory Server 升級策略

Directory Server 的升級策略一般取決於在第 1 章「升級規劃」中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Directory Server 升級規劃的問題，藉以對 Directory Server 進行一般性的介紹。

相容性問題

Java ES 發行版本 5 Directory Server 沒有引入新的公共介面，因此向下相容較早版本；它支援發行版本 4 Directory Server 及較早版本支援的所有元件。

但是，發行版本 5 引入了對私人管理介面進行的變更。發行版本 5 的介面與 Directory Server 的較早發行版本相容。需特別指出的是，用來配置較早的 Directory Server 實例的 Administration Server 已由 Directory Service Control Center 與 Directory Server EE 指令行公用程式取代，而且已經消除用於儲存 Directory Server 配置資訊的 `o=NetscapeRoot` 目錄尾碼。「Directory Server Enterprise Edition 6 Migration Guide」(<http://docs.sun.com/doc/819-0994>) 中有詳細資訊。

相依性

一般而言，對其他 Java ES 元件的相依性可能會影響升級 Directory Server 軟體的程序。

Directory Server 對下列 Java ES 元件有相依性：

- **共用元件**。Directory Server 對特定 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。Directory Server 升級可能依賴於這些共用元件的升級版本。
- **Directory Proxy Server**。Directory Server 與 Directory Proxy Server 互有相依性，後者用於提高 LDAP 請求的安全性與效能。

雙重升級

在雙重升級中，Directory Server 與作業系統均會進行升級 (如第 40 頁的「雙重升級：Java ES 與作業系統軟體」中所述)，這種升級可採用下列兩種方式之一來執行：

- 全新安裝作業系統
- 原地升級作業系統

全新安裝作業系統

1. 備份現有的 Directory Server 資料。

有關 Directory Server 5.x 資訊，請參閱第 101 頁的「[建立 Directory Server 影像 \(選擇性\)](#)」。

2. 安裝新的作業系統。

可將作業系統安裝在新系統 (或 Solaris 10 區域) 上，亦可讓其徹底覆蓋現有的檔案系統。

3. 復原[步驟 1](#)中所備份的 Directory Server 資料。

4. 安裝發行版本 5 Directory Server。

5. 建立一個發行版本 5 Directory Server 實例並將目錄資料遷移至新實例。

請參閱第 101 頁的「[升級發行版本 4 Directory Server](#)」程序中的相關步驟。

原地升級作業系統

1. 備份現有的 Directory Server 資料。

有關 Directory Server 5.x 資訊，請參閱第 101 頁的「[建立 Directory Server 影像 \(選擇性\)](#)」。

2. 升級作業系統。
升級時會原樣保留現有的檔案系統。
3. 升級到發行版本 5 Directory Server。
請根據升級路徑，參閱本章的相關章節。

從 Java ES 發行版本 4 升級 Directory Server

本節包括將 Directory Server 從 Java ES 2005 Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Directory Server](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Directory Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 若要達成升級，首先執行發行版本 5 Directory Server 的全新安裝，然後使用遷移工具在全新、不同的發行版本 5 Directory Server 實例中重新建立之前的 Directory Server 實例。
- **升級相依性。** Directory Server 對許多 Java ES 共用元件具有相依性 (請參閱 [第 43 頁的表 1-9](#))，當您升級 Directory Server 時，Java ES 安裝程式會將所有這些共用元件自動升級到發行版本 5。
- **向下相容性。** 發行版本 5 Directory Server 不向下相容發行版本 4 版本，如 [第 96 頁的「相容性問題」](#) 中所述。但是，如果您繼續保留依賴 Directory Server 5.x 管理架構的一組 Directory Server 實例，則遷移工具可以遷移 `o=NescapeRoot` 尾碼。
- **升級回復。** 發行版本 5 升級的回復是藉由復原為先前版本來達成的，先前版本在升級到發行版本 5 時保持不變。
- **平台問題。** 升級 Directory Server 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 Directory Server

本節描述如何在 Solaris 及 Linux 平台上將 Directory Server 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 Directory Server](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Directory Server 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Directory Server 相依元件](#)
- [取得必要的配置資訊和密碼](#)
- [建立 Directory Server 影像 \(選擇性\)](#)

驗證目前版本資訊

使用 `-v` 選項重新啓動 Directory Server 常駐程式，即可驗證 Directory Server 的目前版本：

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

表 5-4 Directory Server 版本驗證輸出

Java ES 發行版本	Directory Server 版本編號
發行版本 2	Sun Java (TM) System Directory Server/5.2_Patch_2
發行版本 3	Sun Java (TM) System Directory Server/5.2_Patch_3
發行版本 4	Sun Java (TM) System Directory Server/5.2_Patch_4
發行版本 5	Sun Java (TM) System Directory Server/6.0

如果 `ns-slapd` 指令在 Solaris 10 平台上失敗，請在執行該指令時將程式庫路徑設定為空值：

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

升級 Directory Server 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。但是，當您將 Directory Server 升級到發行版本 5 時，會自動升級 Directory Server 所需要的所有共用元件。

取得必要的配置資訊和密碼

您應該知道 Directory Server 目前安裝版本的管理員使用者 ID 和密碼。升級過程中會保留其他配置資訊。

建立 Directory Server 影像 (選擇性)

在要將發行版本 5 Directory Server 安裝到與發行版本 4 版本所在電腦不同的情況下，應該在要安裝發行版本 5 Directory Server 的電腦上建立發行版本 4 版本的影像。需要影像才能將資料自動遷移 (使用 `dsmig` 指令) 到新的發行版本 5 Directory Server 實例中。

發行版本 4 影像包括與原始 Directory Server 5.x *serverRoot* 檔案結構相同配置的所有模式檔案、配置檔案、安全性檔案與資料庫檔案。需要影像才能將資料遷移至新的發行版本 5 Directory Server 實例中。

升級發行版本 4 Directory Server

本節先討論影響 Directory Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Directory Server 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 應關閉使用 Directory Server 實例的任何 Java ES 元件 (例如 Access Manager 或 Portal Server，或者 Sun Java Communications Suite 元件)，並在必要時對這些元件重新進行配置，以存取對應的新發行版本 5 實例。
- 在有多個 Directory Server 實例在一臺電腦上執行 (全都對應於相同的已安裝 Directory Server 影像) 的部署架構中，您只需升級 Directory Server 影像一次；但是，您必須為每個實例分別遷移資料。

- 在許多發行版本 4 Directory Server 部署架構中，配置目錄是單獨的 Directory Server 實例。這些實例不需要升級，因為發行版本 5 中已停用配置目錄。從另一方面來說，升級可能需要將發行版本 5 Directory Server 管理主控台 (Directory Service Control Center) 部署到單獨的一臺電腦上，您可以透過此電腦遠端管理 Directory Server 實例。
- Directory Server 隨附指令行工具，它可協助將遷移模式、配置、安全性與使用者資料自動化。遷移工具可以逐步遷移這些不同的資料。大多數升級方案都獲益於至少某些資料的自動化遷移。

升級程序

下述程序適用於位於即將進行升級之電腦本機上的 Directory Server 實例，或者在將實例移動至其他電腦的情況下，則適用於將要在目標電腦上執行的所有實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 關閉發行版本 4 Directory Server (5.2) 實例。

```
serverRoot/slapd-instanceName/stop-slapd
```

請檢查錯誤記錄 (*serverRoot/slapd-hostName/logs/errors*) 是否報告已徹底關機：

```
[23/Jan/2006:15:56:47 +0100] - All database threads now stopped  
  
[23/Jan/2006:15:56:50 +0100] - slapd stopped.
```

3. 確定發行版本 5 Directory Server 的主機電腦有足夠的磁碟空間。

基本計算如下：

2 * (現有伺服器的空間) + (LDIF 檔案的空間)

很遺憾，沒有可以預估從匯出的資料庫中建立的 LDIF 檔案大小的工具。其大小將依據資料項目數目、它們的內部表示形式、索引數目等而定。

4. 針對發行版本 5 的遠端安裝，請建立發行版本 4 影像，然後將它傳輸到遠端電腦。

請參閱第 101 頁的「[建立 Directory Server 影像 \(選擇性\)](#)」。

5. 確定已升級 Directory Server 具有硬性升級相依性的所有 Java ES 元件 (請參閱第 101 頁的「[升級 Directory Server 相依元件](#)」)。

6. 執行發行版本 5 Directory Server 的全新安裝。

執行下列步驟：

a. 啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch  
  
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

b. 選取 Directory Server Enterprise Edition 的 Directory Server 子元件。

您還需要安裝您要使用的管理子元件 (Directory Service Control Panel 或指令行公用程式)。

c. 指定與任何現有發行版本 4 Directory Server 不同的安裝路徑。

d. 選擇 [立即配置] 或 [以後配置]。

無論您選擇 [立即配置] 還是 [以後配置] 都沒有關係，因為 Directory Server 實際上並不需要配置。但是，如果您選擇 [立即配置]，請不要選擇建立新實例。

e. 確認您的安裝選擇。

將會升級 Directory Server 套裝軟體，並顯示升級摘要。

f. 結束 Java ES 安裝程式。

7. 建立 Directory Server 實例。

```
DirServer-base/ds6/bin/dsadm create instancePath
```

其中 *instancePath* 是 Directory Server 實例的完整路徑。

如需有關建立 Directory Server 實例的資訊，請參閱「Directory Server Enterprise Edition 6 管理指南」(<http://docs.sun.com/doc/820-0294>)。

如果您建立新實例失敗，當您使用 `dsmig` 指令遷移資料時，會自動為您建立新實例 (步驟 8)。

如果 `dsadm` 指令在 Solaris 10 平台上失敗，請在執行該指令時將程式庫路徑設為空值：

```
LD_LIBRARY_PATH= ./dsadm create instancePath
```

8. 將發行版本 4 資料遷移至發行版本 5 Directory Server 實例。

使用 *DirServer-base*/ds6/bin/`dsmig` 指令。

`dsmig` 指令會將發行版本 4 資料修改為發行版本 5 格式，並將其寫入到適當的位置。例如，具有一個 Directory Server 實例的單一電腦上的典型遷移應該如下所示：

```
dsmig migrate-schema -v old_instancePath new_instancePath  
  
dsmig migrate-config -v old_instancePath new_instancePath  
  
dsmig migrate-security -v old_instancePath new_instancePath  
  
dsmig migrate-data -v old_instancePath new_instancePath
```

備註

- 如果 `dsmig migrate-config` 指令在 Solaris 10 平台上失敗，請在執行該指令時將程式庫路徑設為空值：

```
LD_LIBRARY_PATH= ./dsmig migrate-config ...
```

- 如果您要遷移的 Directory Server 實例儲存著其他 Java ES 元件 (例如 Sun Java Communications Suite Messaging Server 元件) 的配置資料，可能會要求您遷移名為 `o=netscaperoot` 的目錄資訊樹狀結構的特定部分。依預設，不會遷移此根尾碼。若要遷移 `o=netscaperoot`，請使用 `dsmig migrate-config` 與 `dsmig migrate-data` 指令的 `-N` 選項。例如：

```
dsmig migrate-config -v old_instancePath
new_instancePath -N
```

- 如果您要從 32 位元架構上的實例遷移到 64 位元架構上的實例，您不可以使用 `dsmig migrate-data` 指令 (自動遷移工具)。您必須手動遷移資料，如以下「Migration Guide」中所述。但是，您仍然可以執行模式、配置與安全性資料的自動遷移。
 - 在某些情況下，當在遷移目錄資料之後啟動 Directory Server 時，新的發行版本 5 錯誤檢查會偵測 Directory Server 群組項目中的循環定義。這些循環定義具有有利的功能，但是可能會導致在錯誤檔案中記錄大量的錯誤。
-

如需遷移程序、`dsmig` 指令與手動遷移的詳細資訊，請參閱「Directory Server Enterprise Edition 6 Migration Guide」(<http://docs.sun.com/doc/819-0994>)。

驗證升級

您可按以下所述驗證 Directory Server 的升級是否成功。

1. 啟動新 Directory Server 實例：

```
DirServer-base/ds6/bin/dsadm -V
```

請參閱第 100 頁的表 5-4 中的輸出值。

2. 檢查 Directory Server 錯誤記錄中的啟動訊息：

```
instancePath/logs/errors
```

備註

啟動時，發行版本 5 Directory Server 會立即偵測循環定義。這些循環定義具有有利的功能，但是當您從包含此類循環定義的舊版升級時，它可能會導致大量錯誤。

升級後作業

除非需要重新配置所有依賴 Directory Server 的 Java ES 元件以指向新的 Directory Server 實例，否則除了在第 102 頁的「升級程序」中所述的步驟以外，沒有其他的升級後的作業。

回復升級

發行版本 5 升級的回復是藉由復原為先前版本來達成的，先前版本在升級到發行版本 5 時保持不變。

多重實例升級

第 100 頁的「升級發行版本 4 Directory Server」中的程序沒有對一些部署架構進行明確說明，在這些部署架構中，可藉由複製 Directory Server 來獲得可用性或延展性。這些架構可能包括 Directory Server 複製，或 Directory Server 做為資料服務在 Sun Cluster 環境中的部署。

漸進式升級 Directory Server 複製

對於不同電腦系統上的多個 Directory Server 實例，可如多個主伺服器複製部署架構中實例的使用方式那樣，依序一次升級一個實例。第一次同步化所有 Directory Server 主機之後，您可在每個實例各自的主機上升級它們，而其他實例仍繼續執行。這種漸進式升級可以讓目錄服務在提供該服務的個別 Directory Server 實例正在升級時，仍保持線上狀態。

將 Directory Server 做為資料服務進行升級

有關在 Sun Cluster 環境中將 Directory Server 做為資料服務升級與回復的資訊目前正在開發。

從 Java ES 發行版本 3 升級 Directory Server

將 Java ES 2003Q1 (發行版本 3) Directory Server 升級到發行版本 5 的程序與將發行版本 4 Directory Server 升級到發行版本 5 的程序相同。

若要將發行版本 3 Directory Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 99 頁的「從 Java ES 發行版本 4 升級 Directory Server」中的指示。

從 Java ES 發行版本 2 升級 Directory Server

將 Java ES 2004Q2 (發行版本 2) Directory Server 升級到發行版本 5 的程序與將發行版本 4 Directory Server 升級到發行版本 5 的程序幾乎相同，唯一不同之處在於升級前作業應該包括將所有共用元件都升級到發行版本 5 (請參閱第 43 頁的表 1-9)。

在第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了將 Java ES 共用元件升級到發行版本 5 的指示。

若要將發行版本 2 Directory Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他請遵循第 99 頁的「從 Java ES 發行版本 4 升級 Directory Server」中的指示。

備註 如果您要在 Linux 平台上從發行版本 2 Directory Server 進行升級，則必須執行雙重升級，也就是 Directory Server 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Directory Server)。如需更多資訊，請參閱第 97 頁的「雙重升級」。

Directory Proxy Server

本章描述如何將 Directory Proxy Server 升級到 Java ES 5 (發行版本 5) : Sun Java System Directory Proxy Server 6.0。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 110 頁的「Directory Proxy Server 升級簡介」](#)
- [第 114 頁的「從 Java ES 發行版本 4 升級 Directory Proxy Server」](#)
- [第 120 頁的「從 Java ES 發行版本 3 升級 Directory Proxy Server」](#)
- [第 121 頁的「從 Java ES 發行版本 2 升級 Directory Proxy Server」](#)

備註 本章中的檔案位置是以稱為 *serverRoot* (Directory Proxy Server 5.x) 和 *DirServer-base* (Directory Proxy Server 6.0) 的目錄路徑為基準指定的。安裝 Directory Proxy Server 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 6-1 Directory Proxy Server 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>serverRoot</i> (Directory Proxy Server 5.x)	/var/opt/mps/serverroot	/var/opt/sun/directory-server
<i>DirServer-base</i> (Directory Proxy Server 6.0)	/opt/SUNWdsee	/opt/sun

Directory Proxy Server 升級簡介

本節描述下列會影響 Directory Proxy Server 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5](#)
- [Java ES 發行版本 5 升級資訊指南](#)
- [Directory Proxy Server 資料](#)
- [Directory Proxy Server 升級策略](#)

關於 Java ES 發行版本 5

Java ES 發行版本 5 Directory Proxy Server 代表一個重大發行版本，是相對於發行版本 4 Directory Proxy Server 和所有以前發行版本的新產品。

發行版本 5 Directory Proxy Server 仍然是 LDAP 代理，但新增了一些可延伸的路由功能。發行版本 5 也支援「虛擬目錄」功能，即將多個資料檢視整合到單一檢視的功能。這些資料檢視可以表示 LDAP 或 SQL 可存取資料儲存區。

如需詳細資訊，請參閱「Directory Server Enterprise Edition 6 版本說明」。

Java ES 發行版本 5 升級資訊指南

表 6-2 顯示支援的 Directory Proxy Server 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 6-2 至 Java ES 5 (發行版本 5) 的升級路徑：Directory Proxy Server 6.0

Java ES 發行版本	Directory Proxy Server 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Directory Proxy Server 5.2 2005Q4	直接升級： 以發行版本 5 的全新安裝與配置取代發行版本 4。	如果需要向下相容性，請手動將先前的配置對應至新的配置特性。
發行版本 3	Sun Java System Directory Proxy Server 5.2 2005Q1	直接升級： 以發行版本 5 的全新安裝與配置取代發行版本 3。	如果需要向下相容性，請手動將先前的配置對應至新的配置特性。
發行版本 2	Sun Java System Directory Proxy Server 5.2 2004Q2	直接升級： 以發行版本 5 的全新安裝與配置取代發行版本 2。	如果需要向下相容性，請手動將先前的配置對應至新的配置特性。

表 6-2 至 Java ES 5 (發行版本 5) 的升級路徑：Directory Proxy Server 6.0 (續)

Java ES 發行版本	Directory Proxy Server 版本	一般方法	需要重新配置
發行版本 1	Sun ONE Directory Proxy Server 5.2	尚未認證直接升級： 但可以使用從發行版本 2 升級時所使用的方法。	如果需要向下相容性，請手動將先前的配置對應至新的配置特性。
Java ES 發行版本之前的發行版本	Sun ONE Directory Proxy Server 5.2	尚未認證直接升級： 但可以使用從發行版本 2 升級時所使用的方法。	如果需要向下相容性，請手動將先前的配置對應至新的配置特性。
	Sun ONE Directory Access Router 5.0 或 5.0 SP1	無直接升級： 先升級到發行版本 3。請參閱「Java Enterprise System 2005Q1 升級與遷移指南」 (http://docs.sun.com/doc/819-2238)。 然後從發行版本 3 升級到發行版本 5。	請參閱「Java Enterprise System 2005Q1 升級與遷移指南」 (http://docs.sun.com/doc/819-2238)。

Directory Proxy Server 資料

Directory Proxy Server 不再使用 Directory Server 來儲存配置資料。將會使用新的 Directory Service Control Center 或 Directory Server EE 指令行公用程式來執行配置。

下表顯示可能受升級 Directory Proxy Server 軟體影響的資料類型。

表 6-3 Directory Proxy Server 資料用途

資料類型	位置	用途
Directory Proxy Server 配置資料	Directory Proxy Server 5.2：配置目錄 Directory Proxy Server 6.0：透過 Directory Service Control Center 與 Directory Server EE 指令行公用程式存取。	Directory Proxy Server 的配置
安全性資料	Directory Proxy Server 5.2：透過 Directory Proxy Server Console 配置的 SSL。 Directory Proxy Server 6.0：透過 Directory Service Control Center 與 Directory Server EE 指令行公用程式配置的 SSL。	伺服器憑證

Directory Proxy Server 升級策略

Directory Proxy Server 的升級策略一般取決於在第 1 章「升級規劃」中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Directory Proxy Server 升級規劃的問題，藉以對 Directory Proxy Server 進行一般性的介紹。

相容性問題

發行版本 5 Directory Proxy Server 引入了使其與較早 Directory Proxy Server 發行版本不相容的介面變更。發行版本 5 Directory Proxy Server 以全新基於 Java 的實作為基礎，其配置與發行版本 4 Directory Proxy Server 以及較早的發行版本完全不同。

但是可以將發行版本 5 Directory Proxy Server 配置為向下相容，即像發行版本 4 Directory Proxy Server 與較早發行版本一樣運作。此配置需要您手動將先前的配置屬性對應至發行版本 5 的配置特性。「Directory Server Enterprise Edition 6 Migration Guide」(<http://docs.sun.com/doc/819-0994>) 中有詳細資訊。

但是，與先前版本相比，發行版本 5 Directory Proxy Server 具有不同的預設運作方式：它不允許 LDAP 控制項通過代理。若要重現先前版本的運作方式，您可以按照第 118 頁的「升級後作業」中所述的方式解除這些控制的封鎖。

相依性

對其他 Java ES 元件的相依性可能會影響升級和重新配置 Directory Proxy Server 軟體的程序。

Directory Proxy Server 對下列 Java ES 元件有相依性：

- **共用元件**。Directory Proxy Server 對特定 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。Directory Proxy Server 升級可能依賴於這些共用元件的升級版本。
- **Directory Server**。Directory Proxy Server 與 Directory Server 互有相依性，後者用於提高 LDAP 請求的安全性與效能。Directory Proxy Server 提供了對 Directory Server 的前端存取，但是除了這種功能關係之外，它對 Directory Server 沒有相依性。

雙重升級

在雙重升級中，Directory Proxy Server 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用下列兩種方式之一來執行：

- 全新安裝作業系統
- 原地升級作業系統

全新安裝作業系統

1. 備份現有的 Directory Proxy Server 資料。

如需必要資料的位置，請參閱第 111 頁的「[Directory Proxy Server 資料](#)」。

2. 安裝新的作業系統。

可將作業系統安裝在新系統 (或 Solaris 10 區域) 上，亦可讓其徹底覆蓋現有的檔案系統。

3. 安裝發行版本 5 Directory Proxy Server 。

4. 建立一個發行版本 5 Directory Proxy Server 實例，並將配置屬性對映至發行版本 5 Directory Proxy Server 特性。

請參閱第 116 頁的「[升級發行版本 4 Directory Proxy Server](#)」程序中的相關步驟。

原地升級作業系統

1. 備份現有的 Directory Proxy Server 資料。

如需必要資料的位置，請參閱第 111 頁的「[Directory Proxy Server 資料](#)」。

2. 升級作業系統。

升級時會原樣保留現有的檔案系統。

3. 升級到發行版本 5 Directory Proxy Server 。

請根據升級路徑，參閱本章的相關章節。

從 Java ES 發行版本 4 升級 Directory Proxy Server

本節包括將 Directory Proxy Server 從 Java ES 2005Q4 (發行版本 4) 升級至 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Directory Proxy Server](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Directory Proxy Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 若要達成升級，首先執行發行版本 5 Directory Proxy Server 的全新安裝，然後使用 Directory Service Control Center 或 Directory Server EE 指令行公用程式來配置新的 Directory Proxy Server 實例。
- **升級相依性。** Directory Proxy Server 對許多 Java ES 共用元件具有相依性 (請參閱第 43 頁的表 1-9)，當您升級 Directory Proxy Server 時，Java ES 安裝程式會將所有這些共用元件自動升級到發行版本 5。
- **向下相容性。** 發行版本 5 Directory Proxy Server 可以配置為向下相容其發行版本 4 版本，如第 112 頁的「[相容性問題](#)」中所解釋。
- **升級回復。** 發行版本 5 升級的回復是藉由復原為先前版本來達成的，先前版本在升級到發行版本 5 時保持不變。
- **平台問題。** 升級 Directory Proxy Server 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 Directory Proxy Server

本節描述如何在 Solaris 及 Linux 平台上將 Directory Proxy Server 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 Directory Proxy Server](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Directory Proxy Server 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Directory Proxy Server 相依元件](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可使用下列指令驗證 Directory Proxy Server 的目前版本：

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

下表顯示其輸出：

表 6-4 Directory Proxy Server 版本驗證輸出

Java ES 發行版本	Directory Proxy Server 版本編號
發行版本 2	Sun ONE Directory Proxy Server Version 5.2_Patch_2
發行版本 3	Sun ONE Directory Proxy Server Version 5.2_Patch_3
發行版本 4	Sun ONE Directory Proxy Server Version 5.2_Patch_4
發行版本 5 ¹	Sun ONE Directory Proxy Server Version 6.0

1. 不能使用 ldapfwd 指令來傳回發行版本 5 的版本號碼。請參閱第 118 頁的「驗證升級」。

升級 Directory Proxy Server 相依元件

一般的建議是將電腦系統 (及運算環境中) 的所有 Java ES 元件都升級到 Java ES 發行版本 5。Directory Proxy Server 只對幾個共用元件有硬性升級相依性。

升級 Directory Proxy Server 相依元件時，在升級 Directory Proxy Server 之前，應按照下列順序執行 (略過可能已升級過的任何相依元件)。共用元件的升級通常可由 Java ES 安裝程式自動完成。

1. **共用元件。** 在第 59 頁的「[升級 Java ES 共用元件](#)」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Directory Proxy Server 升級到發行版本 5 時，Directory Proxy Server 所需要的所有共用元件會由 Java ES 安裝程式自動升級。
2. **Directory Server (非硬性升級相依性)。** 在第 93 頁的第 5 章「[Directory Server](#)」中提供了將 Directory Server 升級到發行版本 5 的指示。但是，發行版本 4 Directory Server 支援發行版本 5 Directory Proxy Server。

取得必要的配置資訊和密碼

配置資訊在升級過程中會保留下來，可以用來將發行版本 4 的配置屬性對應至發行版本 5 的配置特性。請參閱第 112 頁的「[相容性問題](#)」。

升級發行版本 4 Directory Proxy Server

本節討論影響 Directory Proxy Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Directory Proxy Server 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 應該關閉使用 Directory Proxy Server 實例的任何 Java ES 元件 (如 Access Manager、Communications Express、Messaging Server、Portal Server 等)，並將這些元件重新配置為存取對應的新發行版本 5 實例。
- 在有多個 Directory Proxy Server 實例在一臺電腦上執行 (全都對應於相同的 Directory Proxy Server 安裝影像) 的部署架構中，升級 Directory Proxy Server 影像時將需要您建立新的 Directory Proxy Server 實例。
- 在包含 Directory Proxy Server 的發行版本 4 部署架構中，使用了 Administration Server 來配置及管理 Directory Proxy Server 實例。在發行版本 5 中，Directory Proxy Server 的升級可能需要部署用來配置及管理 Directory Proxy Server 實例的 Directory Service Control Center。

升級程序

下述程序適用於位於執行升級的電腦本機上的 Directory Proxy Server 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 關閉所有依賴要升級的 Directory Proxy Server 實例之 Java ES 元件。此步驟可能取決於如何在部署架構內複製 Directory Proxy Server。

如需關於如何關閉 Java ES 元件的資訊，請參閱其各自的管理指南。

3. 執行發行版本 5 Directory Proxy Server 的全新安裝。

執行下列步驟：

- a. 啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 選取 Directory Server Enterprise Edition 的 Directory Proxy Server 子元件。
您還需要安裝您要使用的管理子元件 (Directory Service Control Panel 或指令行公用程式)。
- c. 指定與任何現有發行版本 4 Directory Proxy Server 不同的安裝路徑。
- d. 選擇 [立即配置] 或 [以後配置]。

無論您選擇立即配置還是以後配置都沒有關係，因為 Directory Proxy Server 實際上並不需要配置。但是，如果您選擇立即配置，系統會詢問您是否要建立新實例。(安裝元件後，您可以建立盡可能多的 Directory Proxy Server 實例。)

- e. 如有必要，請選取安裝本土化套裝軟體的選項。
- f. 確認您的安裝選擇。

將會安裝 Directory Proxy Server 套裝軟體，並顯示安裝摘要。

- g. 結束 Java ES 安裝程式。

如果您尚未建立新實例，請繼續前往步驟 4，否則，請繼續步驟 5。

4. 建立 Directory Proxy Server 實例。

```
DirServer-base/dps6/bin/dpadm create instancePath
```

其中 *instancePath* 是 Directory Proxy Server 實例的完整路徑。

如需有關建立 Directory Proxy Server 實例的資訊，請參閱「Directory Server Enterprise Edition 6 管理指南」(<http://docs.sun.com/doc/820-0294>)。

5. 依需要，將發行版本 4 的配置屬性對應至發行版本 5 Directory Proxy Server 的特性。

如需對應程序的詳細資訊，請參閱「Directory Server Enterprise Edition 6 Migration Guide」(<http://docs.sun.com/doc/819-0994>)。

驗證升級

您可按以下所述驗證 Directory Proxy Server 的升級是否成功。

1. 啟動新 Directory Proxy Server 實例。

```
DirServer-base/dps6/bin/dpadm start instancePath
```

2. 檢查 Directory Proxy Server 版本。

```
DirServer-base/dps6/bin/dpadm --version
```

第 115 頁的表 6-4 中顯示了輸出值。

升級後作業

所有依賴於 Directory Proxy Server 的 Java ES 元件都需要重新配置為指向新的 Directory Proxy Server 實例。

此外，若要重現先前版本的預設運作方式，必須明確允許 LDAP 控制項通過代理。您可以藉由按如下所示設定 `allowed-ldap-controls` 特性來啟用這些控制：

```
cd DirServer-base/dps6/bin

./dpconf set-server-prop
allowed-ldap-controls:auth-request
allowed-ldap-controls:chaining-loop-detection
allowed-ldap-controls:manage-dsa
allowed-ldap-controls:persistent-search
allowed-ldap-controls:proxy-auth-v1
allowed-ldap-controls:proxy-auth-v2
allowed-ldap-controls:real-attributes-only
allowed-ldap-controls:server-side-sorting
```

回復升級

發行版本 5 升級的回復是藉由復原為先前版本來達成的，先前版本在升級到發行版本 5 時保持不變。

多重實例升級

在某些部署架構中，會將 Directory Proxy Server 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Directory Proxy Server 元件在多台電腦上執行，並透過負載平衡器來分散負載。

若 Directory Proxy Server 實例採用負載平衡，則無法執行漸進式升級；您需要關閉負載平衡器並重新配置為存取發行版本 5 實例。需要按照第 115 頁的「[升級發行版本 4 Directory Proxy Server](#)」中所述的方式升級每個實例。

從 Java ES 發行版本 3 升級 Directory Proxy Server

將 Java ES 2005Q1 (發行版本 3) Directory Proxy Server 升級到發行版本 5 的程序與將發行版本 4 Directory Proxy Server 升級到發行版本 5 的程序相同。

若要將發行版本 3 Directory Proxy Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 114 頁的「[從 Java ES 發行版本 4 升級 Directory Proxy Server](#)」中的指示。

從 Java ES 發行版本 2 升級 Directory Proxy Server

將 Java ES 2004Q2 (發行版本 2) Directory Proxy Server 升級到發行版本 5 的程序與將發行版本 4 Directory Proxy Server 升級到發行版本 5 的程序幾乎相同，唯一不同之處在於升級前作業應該包括將所有共用元件 (請參閱第 43 頁的表 1-9) 以及依賴於 Directory Proxy Server 的本機所有產品元件與發行版本 5 同步。

在第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。

若要將發行版本 2 Directory Proxy Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他請遵循第 114 頁的「從 Java ES 發行版本 4 升級 Directory Proxy Server」中的指示。

備註 如果您要在 Linux 平台上從發行版本 2 Directory Proxy Server 進行升級，則必須執行雙重升級，也就是 Directory Proxy Server 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Directory Proxy Server)。如需更多資訊，請參閱第 113 頁的「雙重升級」。

Web Server

本章描述如何將 Web Server 升級到 Java ES 5 (發行版本 5) : Sun Java System Web Server 7.0。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 124 頁的「Web Server 升級簡介」](#)
- [第 128 頁的「從 Java ES 發行版本 4 升級 Web Server」](#)
- [第 143 頁的「從 Java ES 發行版本 3 升級 Web Server」](#)
- [第 144 頁的「從 Java ES 發行版本 2 升級 Web Server」](#)

備註 本章中的檔案位置是以稱為 *WebServer6-base* (Web Server 6.x) 及 *WebServer7-base* 和 *WebServer7Config-base* (Web Server 7.0) 的目錄路徑為基準指定的。初始安裝 Web Server 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 7-1 Web Server 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>WebServer6-base</i>	/opt/SUNWwbsvr	/opt/sun/webserver
<i>WebServer7-base</i>	/opt/SUNWwbsvr7	/opt/sun/webserver7
<i>WebServer7Config-base</i>	/var/opt/SUNWwbsvr7	/var/opt/sun/webserver7

Web Server 升級簡介

本節描述下列會影響 Web Server 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Web Server](#)
- [Web Server 升級資訊指南](#)
- [Web Server 資料](#)
- [Web Server 升級策略](#)

關於 Java ES 發行版本 5 Web Server

Java ES 發行版本 5 Web Server 代表相對於發行版本 4 的一個主要發行版本。它新增了許多功能，並在介面方面明顯增強。

發行版本 5 Web Server 擁有新的管理基礎架構及新的管理工具。其管理基礎架構包括一個代管任意多個 Web Server 實例之配置資訊的 Administration Server 實例。新的命令行介面 (wadm) 與新的圖形化使用者介面可用來在本機電腦或遠端電腦上建立 Web Server 實例，並配置與管理這些實例。新的管理工具需要管理員使用者名稱與密碼。

如需有關新管理基礎架構的更多資訊，請參閱「Web Server 7.0 管理員指南」(<http://docs.sun.com/doc/820-0873>)。

Web Server 管理介面中的這些變化對升級有明顯的影響。

Web Server 升級資訊指南

表 7-2 顯示支援的 Web Server 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 7-2 至 Java ES 5 (發行版本 5) 的升級路徑：Web Server 7.0

Java ES 發行版本	Web Server 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Web Server 6.1 SP5 2005Q4	直接升級： 全新安裝並在之後遷移資料	將實例配置遷移至新實例。
發行版本 3	Sun Java System Web Server 6 2005Q1 Update 1 SP 4	直接升級： 全新安裝並在之後遷移資料	將實例配置遷移至新實例。

表 7-2 至 Java ES 5 (發行版本 5) 的升級路徑：Web Server 7.0 (續)

Java ES 發行版本	Web Server 版本	一般方法	需要重新配置
發行版本 2	Sun Java System Web Server 6 2004Q2 Update 1 SP 2 Platform Edition 和 Enterprise Edition	直接升級： 全新安裝並在之後遷移資料	將實例配置遷移至新 實例。
發行版本 1	Sun ONE Web Server 6.1 (2003Q4)	尚未認證直接升級： 但是可以先執行全新安裝再遷移 資料來完成升級。	將實例配置遷移至新 實例。
Java ES 發行版本 之前的發行版本		無直接升級。	

Web Server 資料

下表顯示可能受升級 Web Server 軟體影響的資料類型。

表 7-3 Web Server 資料用途

資料類型	位置	用途
配置資料	Web Server 6.x (Java ES 發行版本 2、3 和 4)： <i>WebServer6-base/https-instanceName/config/</i> Web Server 7.0 (Java ES 發行版本 5)： 實例配置 <i>WebServer7Config-base/https-configName/config¹/</i> 中央配置存放區 透過 Web Server Console 與 <code>wadm</code> 指令行介面存取。	Web Server 實例的配置

1. 請注意，*WebServer7Config-base* 路徑與 *WebServer6-base* 路徑極為不同。

Web Server 升級策略

Web Server 的升級策略一般取決於在 [第 1 章「升級規劃」](#) 中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Web Server 升級規劃的問題，藉以對 Web Server 進行一般性的介紹。

相容性問題

Java ES 發行版本 5 Web Server 沒有在公用介面中引入任何變更，因此在這方面向下相容較早版本。但是，新管理介面並不向下相容較早的管理介面。這樣會影響 Web 應用程式 (例如包括 Java ES 元件) 的升級與重新部署。

需特別指出的是，發行版本 5 Web Server 針對實例目錄與虛擬伺服器名稱使用不同的預設值，如下表所示。

表 7-4 Web Server 預設名稱

項目	Java ES 發行版本 2、3 和 4 Web Server 6.x 預設值	Java ES 發行版本 5 Web Server 7.0 預設值
配置名稱		<i>hostName.domainName</i>
實例目錄路徑	<i>WebServer6-base/ https-hostName.domainName</i>	<i>WebServer7Config-base https-hostName.domainName</i>
虛擬伺服器名稱	<i>https-hostName.domainName</i>	<i>hostName.domainName</i>

Web Server 相依性

Web Server 對下列 Java ES 元件有相依性：

- **共用元件**。Web Server 對特定 Java ES 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))。Web Server 升級可能依賴於這些共用元件的升級版本。
- **Directory Server**。Web Server 對 Directory Server 有選擇性的相依性，後者用於提供基於 LDAP 的認證。
- **Web Proxy Server**。Web Server 與 Web Proxy Server 互有相依性，後者用於提高 HTTP 請求的安全性與效能。

雙重升級

在雙重升級中，Web Server 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用原地升級作業系統的方法來執行：

1. 備份現有的 Web Server 資料。
如需必要資料的位置，請參閱第 125 頁的「[Web Server 資料](#)」。
2. 升級作業系統。
升級時會原樣保留現有的檔案系統。
3. 升級到發行版本 5 Web Server。
請根據升級路徑，參閱本章的相應小節。

從 Java ES 發行版本 4 升級 Web Server

本節包括將 Web Server 從 Java ES 2005Q4 (發行版本 4) 升級至 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Web Server](#)

簡介

將 Java ES 發行版本 4 Web Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 若要達成升級，需要執行發行版本 5 Web Server 的全新安裝、將發行版本 4 Web Server 實例配置資訊遷移至發行版本 5 配置，然後建立對應於發行版本 4 實例的發行版本 5 Web Server 實例。
- **升級相依性。** Web Server 對許多 Java ES 共用元件具有相依性 (請參閱第 43 頁的表 1-9)，當您升級 Web Server 時，Java ES 安裝程式會將其相依的所有共用元件自動升級至發行版本 5。Web Server 只對 NSS 與 NSPR 共用元件有硬性升級相依性。
- **向下相容性。** 發行版本 5 Web Server 管理介面不向下相容發行版本 4 版本。
- **升級回復。** 發行版本 5 的升級回復藉由復原到發行版本 4 安裝 (保持不變的) 來達成。
- **平台問題。** 升級 Web Server 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 Web Server

本節描述如何在 Solaris 及 Linux 平台上將 Web Server 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 Web Server](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Web Server 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Web Server 相依元件](#)
- [備份 Web Server 資料](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可藉由使用 `-version` 選項執行 Web Server 實例伺服器來驗證 Web Server 的目前版本：

```
WebServer6-base/https-hostName.domainName/start -version
```

表 7-5 Web Server 版本驗證輸出

Java ES 發行版本	Web Server 版本編號
發行版本 2	6.1SP2
發行版本 3	6.1SP4
發行版本 4	6.1SP5
發行版本 5	7.0

升級 Web Server 相依元件

一般的建議是將電腦系統 (和運算環境中) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。但是，當您將 Web Server 升級到發行版本 5 時，Java ES 安裝程式會自動升級 Web Server 所需要的所有共用元件 (請參閱第 43 頁的表 1-9)。

備份 Web Server 資料

Web Server 從發行版本 4 到發行版本 5 的升級不會修改現有配置資料；它會保持不變。不需要備份目前的資料。

取得必要的配置資訊和密碼

您需要以超級使用者的身份登入才能執行升級，而執行遷移的使用者帳號應該具備存取現有 Web Server 安裝目錄的權限。

升級發行版本 4 Web Server

本節討論影響 Web Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Web Server 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- **[立即配置] 或 [以後配置]**。當執行升級時，可以指定是使用 [立即配置] 還是 [以後配置] 選項來安裝發行版本 5 Web Server：
 - [立即配置] 表示安裝程式會依照指定安裝 Administration Server 或 Administration Node，也會建立預設配置及對應的 Web Server 實例。此方法對於單一電腦上的安裝非常有用，但是預設配置名稱可能會在升級期間影響現有實例配置的遷移。
 - [以後配置] 表示安裝程式不會執行配置：在將特性值提供給輸入檔案之後，您必須手動執行 `configureServer` 程序檔。如果想要使用執行無訊息安裝的程序檔自動在多部電腦上安裝，此方法會非常有用。您也可以完全控制配置名稱，可以避免在升級期間與現有實例配置的遷移發生衝突。
- **預設發行版本 4 實例配置的遷移**。當執行升級時，將每個發行版本 4 Web Server 實例的配置資料遷移到 Web Server Administration Server 所維護的中央配置存放區中。遷移是使用 `wadm migrate-server` 指令或發行版本 5 Administration Console 來達成的。

如果要遷移的實例是預設發行版本 4 Web Server (6.x) 實例，則它擁有與預設發行版本 5 Web Server (7.0) 配置相同的名稱 (`hostName.domainName`)，而後者由 [立即配置] 選項自動建立。

在執行預設發行版本 4 實例配置到發行版本 5 的遷移時，有三種方法可以採用，每種方法都會產生不同的配置名稱。

- **您選擇的方法可能會影響已部署 Web 應用程式的後續升級**。例如，已部署的 Java ES 元件 (例如 Access Manager 與 Portal Server) 及 Sun Java Communications Suite 元件 (例如 Communications Express、Instant Messaging 與 Delegated Administrator) 在升級時，一般會要求執行此類升級的人員知道發行版本 4 實例配置遷移後的發行版本 5 配置的名稱。

三種方法如下：

- 指定與預設名稱不同的**新**配置名稱，例如「JavaESapps」。然後，`migrate-server` 指令便會建立名為 JavaESapps 的新發行版本 5 配置。

隨後升級的 Web 應用程式需要重新部署到 JavaESapps 配置。

- 不指定新配置名稱，但要先刪除預設發行版本 5 實例及配置 (*hostName.domainName*)，然後再執行 `migrate-server` 指令。然後，`migrate-server` 指令會以預設名稱 (*hostName.domainName*) 建立新配置。順序如下 (如需詳細資訊，請參閱 `wadm help`)：

```
wadm delete-instance
wadm delete-config
wadm migrate-server
wadm create-instance
```

隨後升級的 Web 應用程式需要重新部署到 *hostName.domainName* 配置，該配置與預設發行版本 4 實例同名。

- 不指定新配置名稱，也不刪除預設發行版本 5 配置。然後，`migrate-server` 指令會以如下名稱建立新配置：
hostName.domainName-1。

隨後升級的 Web 應用程式需要重新部署到 *hostName.domainName-1* 配置。

無論您採用哪種方法遷移發行版本 4 實例配置，都應將發行版本 4 實例配置所遷移到之發行版本 5 配置的名稱傳達給後續對該實例中所部署的 Web 應用程式執行升級的所有人。

- **配置資料的遷移。** 當遷移發行版本 4 實例配置時，會自動遷移以下資訊：
 - 發行版本 4 Web Server 實例目錄中的所有配置資訊：
WebServer6-base/https-instanceName/config。這包括在發行版本 4 實例中部署的所有 Web 應用程式的配置資訊 (例如 Access Manager 與 Portal Server 之類的 Java ES 元件)。
 - *WebServer6-base/httpacl* 中的 acl 資訊
 - *WebServer6-base/userdb* 中的 auth-db 資訊
 - *WebServer6-base/https-admserv/config* 中的排程程式資訊
 - *WebServer6-base/alias* 中的憑證資訊
 - 執行遷移時所指定的搜尋集合資訊與索引檔案。

自動遷移不包括以下資料：

- 文件根內容。相反的，新配置會指向舊的文件根，記錄訊息會記錄在遷移記錄中
- Webdav 資料。將會遷移 Webdav 集合資訊
- 將不會遷移協力廠商 NSAPI 外掛程式。相反的，它們會指向發行版本 4 檔案，記錄訊息會記錄在遷移記錄中

- 記錄檔案
- 對搜尋集合文件根的變更
- 指令行程序檔 (startsvr、startsvr.bat、stopsvr、stopsvr.bat、restart、reconfig、reconfig.bat)。

如需有關資料遷移的詳細資訊，請參閱「Web Server 7.0 Installation and Migration Guide」(<http://docs.sun.com/doc/819-2625>)。

- **發行版本 5 Web Server 實例的建立。** 遷移之後，您必須明確地建立對應於已遷移發行版本 4 實例的發行版本 5 Web Server 實例。此作業不會自動完成。

升級程序

下述程序適用於對應於執行升級的電腦上同一安裝的 Web Server 影像的所有 Web Server 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 停止所有執行中的 Web Server 實例和 Administration Server。

```
WebServer6-base/https-instanceName/stop  
WebServer6-base/https-admserv/stop
```

3. 執行發行版本 5 Web Server 的全新安裝。

執行下列步驟：

- a. 啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch  
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對命令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 在元件選取頁面中選取 Web Server。
- c. 指定與發行版本 4 Web Server 不同的安裝路徑。
- d. 選擇 [立即配置] 或 [以後配置]。
 - 如果您選擇 [立即配置]，請跳至**步驟 e**。
 - 如果您選擇 [以後配置]，請跳至**步驟 f**。

- e. 如果您選擇 [立即配置]，Java ES 安裝程式會提供兩個選擇：
- 將管理實例配置為管理伺服器
在發行版本 4 實例遷移到發行版本 5 (及其他管理作業) 所需的 Administration Server 的代管電腦上使用此選擇。
 - 將管理實例配置為管理節點
在將代管 Administration Server 的遠端 Web Server 實例的電腦上使用此選擇。管理實例會配置為與 Administration Server 互動的節點代理程式。
- I. 指定請求的配置值。
- 系統會要求您提供主機名稱、HTTP 連接埠、管理員使用者名稱與管理員密碼。
- II. 確認您的安裝選擇。
- 將會安裝 Web Server 套裝軟體，並顯示安裝摘要。
- Java ES 安裝程式會建立名為 *hostName.domainName* 的預設配置及對應的 Web Server 實例。
- III. 結束 Java ES 安裝程式並跳至第 135 頁的步驟 4。
- f. 如果您選擇 [以後配置]，Java ES 安裝程式會建立在第 134 頁的步驟 IV 中執行的 `configureServer` 程序檔。
- I. 確認您的安裝選擇。
- 將會安裝 Web Server 套裝軟體，並顯示安裝摘要。
- II. 結束 Java ES 安裝程式。
- III. 設定 `WebServer7-base/setup/WSInstall.properties` 檔案中的值。
- 將值提供給下表中所有必需 (非選擇性) 的特性。

表 7-6 WSInstall.properties 值

特性	說明
WS_DOCROOT	(選擇性) 可以代管 Web 內容檔案的文件位置
WS_SERVER_NAME	可以用來處理 HTTP 請求的主機名稱
WS_SERVER_USER	執行階段 Unix 使用者。有效值可以是 root、任何有效的 UNIX 使用者或 <code>webservd</code> (預設值)
WS_HTTP_PORT	可以用來偵聽 HTTP 請求的實例連接埠
WS_ADMIN_SSL_PORT	管理 SSL 連接埠

表 7-6 WSInstall.properties 值 (續)

特性	說明
WS_ADMIN_HOST	管理伺服器作業的管理主機名稱
WS_CONFIG_NAME	此主機的配置名稱。此值可以與在 WS_SERVER_NAME 中所提供的值相同。
WS_ADMIN_SERVER_USER	管理伺服器執行階段 UNIX 使用者。有效值：'root' 或與 WS_SERVER_USER 相同的使用者
WS_ADMIN_LOGIN_USER	管理伺服器登入使用者名稱
WS_ADMIN_LOGIN_PASSWORD	管理伺服器登入密碼
WS_ADMIN_HTTP_PORT	(選擇性) 管理非 SSL 連接埠。預設值：8800
WS_START_ON_BOOT	(選擇性) 開機時啟動功能 (true/false)。True 將允許伺服器實例與它的管理伺服器在系統重新開機後自動啟動。預設值：false
WS_64BIT_INSTALL	(選擇性) 伺服器執行階段模式 (true/false)。True 將在 64 位元模式下配置伺服器。(只適用於 Solaris)。False 將在 32 位元模式下配置伺服器。(只適用於 Solaris) 預設值：false
WS_ADMIN_IS_SERVER_MODE	(選擇性) 管理配置模式。(true/false)。True 將在管理伺服器模式下配置伺服器。False 將在管理代理程式模式下配置伺服器。預設值：true
WS_REGISTER_ADMIN_AGENT	(選擇性) 移除代理程式註冊。(true/false)。只有在 WS_ADMIN_IS_SERVER_MODE 設定為 false 時才需要此選擇。True 將會要求您提供遠端管理伺服器主機以進行註冊。預設值：true
WS_AGENT_SSL_PORT	(選擇性) 管理代理程式 SSL 連接埠。只有在 WS_ADMIN_IS_SERVER_MODE 設定為 false 時才需要此選擇。
WS_AGENT_HOST=	(選擇性) 管理代理程式主機名稱。只有在 WS_ADMIN_IS_SERVER_MODE 設定為 false 時才需要此選擇。

IV. 執行 configureServer 程序檔。

```
WebServer7-base/setup/configureServer
-inputfile WebServer7-base/setup/WSInstall.properties
-logfile WebServer7-base/setup/WSInstall.log
-verbose
```

configureServer 程序檔會建立名為 *hostName.domainName* 的預設配置及對應的 Web Server 實例。

4. 啓動 Web Server Administration Server 服務。

```
WebServer7Config-base/admin-server/bin/startserv
```

5. 將發行版本 4 Web Server 實例配置遷移到發行版本 5 配置。

您可以使用指令行 (wadm) 或圖形化使用者介面管理工具 (登入 Web Server 管理伺服器 GUI)。之後的步驟以 wadm 指令行介面為基礎。

例如，將名為 myinstance 的實例遷移到新配置：

```
WebServer7-base/bin/wadm migrate-server --user=admin
--host=localhost --server-root=/opt/SUNWwbsvr
--instance=https-myinstance --config=newconfigname
```

完整指令語法如下：

```
WebServer7-base/bin/wadm migrate-server
```

```
--user=admin-user [--password-file=admin-pswd-file] [--host=admin-host]
[--echo] [--rcfile=rcfile] [--no-prompt] [--verbose]
```

```
[--search-collection-copy-path=searchCollectionPath]
[--log-dir=directory] --serverroot=path
([--all] | [--instance=https-instanceName] [--config=newconfigName])
```

上述第一組指令選項由所有 wadm 指令共用，在下面的表 7-7 中將有所說明。第二組指令選項是 migrate-server 指令所特有的，在表 7-8 中將有所說明。

呼叫僅具有第一組指令選項的 wadm 會帶您進入 wadm 指令 shell 中。在此 shell 中呼叫指令不需要再次指定共用選項。

如果您從 shell 外部呼叫完整的 wadm 指令，您至少必須指定 --user 與 --host 選項。(如果您省略 --password-file 選項，系統會提示您輸入密碼，如果您省略其他選項，則會假設為預設值。)但是為簡化起見，對於本章中用來說明程序的指令來說，並不包括 --user 與 --host 選項。

依預設，wadm 會在連接埠 8989 上使用 SSL 協定。

如需有關 wadm 指令與選項的完整資訊，請參閱「Web Server 7.0 CLI Reference Manual」(<http://docs.sun.com/doc/819-3283>)。

表 7-7 wadm 共用指令選項

選項	說明
user	授權的 Web Server 管理使用者 ID。
password-file	包含向 Administration Server 認證管理使用者的密碼的檔案。密碼檔案必須包含一行 WADM_PASSWORD= <i>password</i> 。如果未在指令中指定此選項，系統會提示您輸入密碼。
host	正在執行該 Administration Server 的電腦的名稱。預設值： localhost。
echo	將此選項設定為 true 會在執行指令之前在標準輸出上回應指令行。預設值： false。
interactive	如果將此選項設定為 true，系統會提示輸入所需密碼選項的值。預設值： true。
rcfile	啟動 wadm 時用來載入的啟動檔案。預設值： ~/wadmrc。
no-prompt	如果將此選項設定為 true，指令在任何情況下都不會要求使用者輸入任何值。例如，如果不帶參數呼叫指令，指令只會輸出錯誤訊息，而不會要求及等待使用者輸入。當透過 shell 程序檔使用 wadm 指令時，您可能要將此參數設定為 true，以使指令無需等待使用者輸入值即始終會傳回。預設值： false。
verbose	如果設定為 true，會顯示詳細清單。預設值： false。

表 7-8 wadm migrate-server 指令選項與運算元

選項 / 運算元	說明
search-collection-copy-path	<p>指定在遷移搜尋集合時搜尋集合索引檔案將要複製到的路徑。以下遷移方案是可行的：</p> <p>如果 Web Server 6.x 搜尋集合路徑位於 Web Server 6.x 實例的外部，那麼遷移的搜尋集合路徑將會指向 Web Server 6.x 搜尋集合路徑，會忽略此選項。</p> <p>如果 Web Server 6.x 搜尋集合路徑在 Web Server 6.x 實例中，並為此選項指定了有效的路徑，那麼搜尋集合索引檔案會複製到以下目錄： <i>searchCollectionPath/configName/virtualServerName/collectionName</i>。如果指定的路徑無效，會記錄錯誤訊息。</p> <p>如果 Web Server 6.x 搜尋集合路徑在 Web Server 6.x 實例中，但沒有為此選項指定路徑，則不會複製搜尋集合索引檔案。會有一條訊息寫入遷移記錄中，要求使用者使用 <code>wadm add-documents</code> 指令來手動複製搜尋集合索引檔案。在這種情況下，遷移的搜尋集合路徑如下： <i>WebServer7Config-base/https-configName/config/collections/virtualServerName/collectionName</i>。</p>

表 7-8 wadm migrate-server 指令選項與運算元 (續)

選項 / 運算元	說明
log-dir	遷移記錄的位置。 預設值：WebServer7Config-base/admin-serv/logs
serverroot	安裝 Web Server 6.x 版本的安裝位置 (目錄)：與 WebServer6-base 相同。
all	如果設定為 true，會將所有 Web Server 6.x 實例配置都遷移至與實例同名的 Web Server 7.0 配置。如果已經存在具有該名稱的配置，會將 instanceName-1 做為配置名稱使用。預設值：false。
instance	如果要個別遷移實例配置 (all=false)，則為要遷移的 Web Server 6.x 實例配置的名稱 (形式為：https-instanceName)。預設 Web Server 6.x 實例名稱為 hostName.domainName
config	要將指定 Web Server 6.x 實例配置遷移到的配置的名稱。預設為使用 Web Server 6.x 實例配置的 instanceName。但是，如果已經存在具有該名稱的配置，指令會在名稱後附加一個整數。如果正在遷移預設 Web Server 6.x 實例配置，這是一個適合的方案。

使用 migrate-server 指令時，請記住下列注意事項：

- 如果想要遷移多個發行版本 4 實例配置，您可以使用不同的 --instance 值與對應的 --config 引數來多次執行 migrate-server 指令，或使用 --all 選項來一次全部遷移它們。
- 針對 migrate-server 指令的每個呼叫，遷移都會在 --log-dir 選項指定的目錄 (或在預設 WebServer7Config-base/admin-server/logs 目錄) 中建立具有以下名稱的記錄檔：

```
MIGRATION_yyyymmddhhmmss.log
```

如果您選取 --all 選項，則記錄檔會儲存所有遷移實例的遷移資訊。

- 針對 migrate-server 指令未遷移的資料 (請參閱第 130 頁的「升級注意事項」)，您必須手動執行遷移 (請參閱第 139 頁的「升級後作業」)。

6. 建立發行版本 5 Web Server 實例。

您必須為在步驟 5 中遷移的每個發行版本 4 實例配置建立新的發行版本 5 實例。

- a. 在建立新實例之前，驗證遷移記錄並修正遷移配置中的任何問題。

- b. 執行 create-instance 指令。

```
WebServer7-base/bin/wadm create-instance
--config=configName nodehost1 [nodehost2 ...nodehostN]
```

第 136 頁的表 7-7 中說明了共用的指令選項。下表說明 create-instance 指令特有的選項。

表 7-9 wadm create-instance 指令選項與運算元

選項 / 運算元	說明
config	實例應該指向的發行版本 5 配置的名稱。
nodehost	在上面建立實例的電腦的名稱。您可以空格分隔的 <i>hostName.domainName</i> 清單來指定多台電腦，進而建立多個相同的實例。

create-instance 指令會在指定節點的 *WebServer7Config-base/https-configName* 上建立實例目錄，並將配置部署到對應的實例目錄。

7. 啟動每個發行版本 5 實例。

```
WebServer7Config-base/https-configName/bin/startserv
```

建立實例時會建立 *startserv* 程序檔。如果啟動實例時沒有發生任何問題，您會看到 [伺服器啟動成功] 的訊息。會顯示實例的預設 URL。

驗證升級

您可藉由執行下列步驟來驗證 Web Server 是否升級到發行版本 5：

1. 檢查新建立的遷移記錄檔中是否有任何 ERROR 訊息。
如有必要，進行手動變更 (請參閱第 139 頁的「升級後作業」)。
2. 驗證發行版本 5 Web Server 實例。

從 Web 瀏覽器存取以下 URL 並確定您進入歡迎頁面：

```
http://hostName.domainName:port
```

其中完全合格的主機名稱與連接埠對應於每個實例。

3. 使用 `-version` 選項執行 Web Server 實例：

```
WebServer7Config-base/https-configName/bin/startserv -version
```

請參閱第 129 頁的表 7-5 取得版本輸出值。

升級後作業

如有必要，執行主要升級後作業時還應手動遷移某些發行版本 4 資料。這些資料通常與一或多個為發行版本 4 配置並在 `server.xml` 配置檔案中指定的虛擬伺服器關聯。

請注意因應下列情況的升級後程序：

- 遷移 Web Server 6.1 文件根內容
- 遷移 webdav 集合資訊
- 遷移記錄檔案
- 遷移協力廠商 NSAPI 外掛程式
- 變更搜尋集合文件根
- 自訂指令行程序檔

遷移 Web Server 6.1 文件根內容

1. 將 Web Server 6.1 文件根內容複製到您想要的位置。
2. 使用以下指令更新新的文件根路徑：

```
WebServer7-base/bin/wadm set-virtual-server-prop
--config=configName --vs=virtualServerName
document-root=new docroot path
```

第 136 頁的表 7-7 中說明了共用的指令選項。下表說明 `set-virtual-server-prop` 指令特有的選項。

表 7-10 wadm set-virtual-server-prop 指令選項與運算元

選項 / 運算元	說明
<code>config</code>	要設定新文件根路徑的發行版本 5 配置的名稱。
<code>vs</code>	遷移的文件根所對應的虛擬伺服器名稱。
<code>document-root</code>	新文件根目錄的路徑。

3. 將配置重新部署到相關 Web Server 實例。

```
WebServer7-base/bin/wadm deploy-config
[--force] [--restart] [--no-reconfig]
configName
```

第 136 頁的表 7-7 中說明了共用的指令選項。下表說明 `deploy-config` 指令特有的選項。

表 7-11 wadm deploy-cofig 指令選項與運算元

選項 / 運算元	說明
force	若為 true，會強制覆寫自從前一次配置部署之後已經手動修改的實例配置。預設值：false
restart	若為 true，會重新啟動執行中的實例來在已部署配置中選取配置設定。預設值：false
no-reconfig	若為 true，在重新啟動實例之前，執行中的實例不會在已部署配置中選取配置設定。預設值：false
configName	部署到實例名稱與配置名稱相對應的 Web Server 實例的發行版本 5 配置名稱。

遷移 webdav 集合資訊

不需要額外的手動遷移。只要更新文件根路徑就足夠了。

遷移記錄檔案

如果您想要儲存這些檔案，請將它們複製到已知位置（否則，如果您移除了發行版本 4 安裝，它們會被刪除）。

遷移協力廠商 NSAPI 外掛程式

1. 將程式庫檔案從其發行版本 4 位置複製到 `WebServer7-base/lib` 目錄。
2. 將 `magnus.conf` 與 `obj.conf` 配置檔案匯出至一個暫存目錄。

```
WebServer7-base/bin/wadm get-config-file --config=configName
magnus.conf > /tmp/magnus.conf
```

```
WebServer7-base/bin/wadm get-config-file --config=configName
obj.conf > /tmp/obj.conf
```

第 136 頁的表 7-7 中說明了共用的指令選項。

3. 依協力廠商 NSAPI 外掛程式文件中的指定內容來修改 `magnus.conf` 與 `obj.conf` 檔案。

4. 從暫存目錄匯入 `magnus.conf` 與 `obj.conf` 配置檔案。

```
WebServer7-base/bin/wadm set-config-file --config=configName
--upload-file=/tmp/magnus.conf magnus.conf
```

```
WebServer7-base/bin/wadm set-config-file --config=configName
--upload-file=/tmp/obj.conf obj.conf
```

第 136 頁的表 7-7 中說明了共用的指令選項。

5. 將修改的配置重新部署到相關 Web Server 實例。

```
WebServer7-base/bin/wadm deploy-config
[--force] [--restart] [--no-reconfig]
configName
```

第 140 頁的表 7-11 中說明了共用選項。

變更搜尋集合文件根

`migrate-server` 指令具有遷移搜尋集合資訊的選項，但是，您可能要將搜尋集合文件根做如下變更：

1. 如果搜尋集合的文件根與用於發行版本 4 的文件根不同，請使用以下指令來設定搜尋集合的文件根：

```
WebServer7-base/bin/wadm set-search-collection-prop
--config=configName --vs=virtualServerName
--collection-name=searchCollectionName document-root=new docroot path for
the search collection
```

第 136 頁的表 7-7 中說明了共用的指令選項。下表說明 `set-search-collection-prop` 指令特有的選項。

表 7-12 wadm set-search-collection-prop 指令選項與運算元

選項 / 運算元	說明
<code>config</code>	要設定搜尋集合文件根的發行版本 5 配置的名稱。
<code>vs</code>	搜尋集合對應的虛擬伺服器名稱。
<code>collection-name</code>	要設定新文件根路徑的搜尋集合的名稱。
<code>document-root</code>	搜尋集合新文件根目錄的路徑。

2. 將配置重新部署到相關 Web Server 實例。

```
WebServer7-base/bin/wadm deploy-config  
[--force] [--restart] [--no-reconfig]  
configName
```

第 140 頁的表 7-11 中說明了共用選項。

自訂指令行程序檔

如果已經自訂程序檔 (例如 startsvr、startsvr.bat、stopsvr、stopsvr.bat、restart、reconfig 與 reconfig.bat)，您就必須在位於以下目錄中的發行版本 5 預設程序檔上執行相同的自訂：WebServer7-base/bin。

回復升級

發行版本 5 的全新安裝與 Web Server 實例配置的後續遷移都會使發行版本 4 Web Server 保持不變。因此，發行版本 5 Web Server 的回復包括下列用於回復至發行版本 4 的步驟。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 逐一停止所有正在執行的 Web Server 實例。

```
WebServer7Config-base/https-configName/bin/stopserv
```

如果正確停止了伺服器，那麼您會看到 [伺服器已關機] 的訊息。

3. 移除發行版本 5 Web Server 安裝。

您必須移除所有發行版本 5 實例與遷移的配置：

- a. 刪除所有發行版本 5 實例。

```
WebServer7-base/bin/wadm delete-instance --user  
... --config=configName hostName.domainName
```

- b. 刪除所有發行版本 5 配置。

```
WebServer7-base/bin/wadm delete-config --user ... configName
```

4. 重新啟動升級 Web Server 時停止的 Web Server 實例 (如第 132 頁的「升級程序」中所述)。

從 Java ES 發行版本 3 升級 Web Server

將 Java ES 2005Q1 (發行版本 3) Web Server 升級到發行版本 5 的程序與將發行版本 4 Web Server 升級到發行版本 5 的程序相同。

若要將發行版本 3 Web Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 128 頁的「[從 Java ES 發行版本 4 升級 Web Server](#)」中的指示。

從 Java ES 發行版本 2 升級 Web Server

將 Java ES 2004Q2 (發行版本 2) Web Server 升級到發行版本 5 的程序與將發行版本 4 Web Server 升級到發行版本 5 的程序相同。

若要將發行版本 2 Web Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他請遵循第 128 頁的「從 Java ES 發行版本 4 升級 Web Server」中的指示。

備註 如果您要在 Linux 平台上從發行版本 2 Web Server 進行升級，則必須執行雙重升級，也就是 Web Server 和作業系統都要進行升級（在 RHEL 2.1 上不支援發行版本 5 Web Server）。如需更多資訊，請參閱第 127 頁的「雙重升級」。

Java DB

本章描述如何將 Java DB 升級到 Java ES 5 (發行版本 5)：Java DB 10.1.3。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 146 頁的「Java DB 升級簡介」](#)
- [第 149 頁的「從 Java ES 發行版本 4 升級 Java DB」](#)

備註 本章中的檔案位置是以稱為 *JavaDB-base* 的目錄路徑為基準指定的。
此路徑是在安裝 Java DB 時由 Java ES 安裝程式設定的。
此目錄路徑的值顯示在下表中。

表 8-1 Java DB 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>JavaDB-base</i>	/opt/SUNWjavadb	/opt/sun/javadb

Java DB 升級簡介

本節描述下列會影響 Java DB 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5](#)
- [Java ES 發行版本 5 升級資訊指南](#)
- [Java DB 資料](#)
- [Java DB 升級策略](#)

關於 Java ES 發行版本 5

Java ES 發行版本 5 Java DB 是做為 Java ES 產品元件傳送的第一個發行版本；Java DB 是最早做為名為 DerbyDatabase 的共用元件 (包含在 Java ES 發行版本 4 中) 發行的。

相對於發行版本 4 版本而言，發行版本 5 Java DB 為次要發行版本。它包含一些改進的功能、更新的介面以及所選錯誤修正程式。

Java ES 發行版本 5 升級資訊指南

[表 8-2](#) 顯示支援的 Java DB 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 8-2 至 Java ES 5 (發行版本 5) 的升級路徑：Java DB 10.1.3

Java ES 發行版本	Java DB 版本	一般方法	需要重新配置
發行版本 4	Derby Database 10.0.2	直接升級： 以全新安裝取代發行版本 4。不影響永久性資料。	無

Java DB 資料

下表顯示可能受升級 Java DB 軟體影響的資料類型。

表 8-3 Java DB 資料用途

資料類型	位置	用途
配置資料	實例配置是應用程式特定的，且儲存於 Java DB 資料庫中。	配置 Java DB 實例
永久性資料	資料庫目錄及其內容是應用程式特定的。其位置由資料庫連線 URL (<code>jdbc:derby:full path to database</code>) 所指定。	資料庫和使用者憑證

Java DB 升級策略

Java DB 的升級策略一般取決於在 [第 1 章「升級規劃」](#) 中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Java DB 升級規劃的問題，藉以對 Java DB 進行一般性的介紹。

相容性問題

發行版本 5 Java DB 向下相容發行版本 4 版本。

相依性

Java DB 只對 J2SE 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))。

雙重升級

在雙重升級中，Java DB 與作業系統均會進行升級 (如 [第 40 頁的「雙重升級：Java ES 與作業系統軟體」](#) 中所述)，這種升級可採用下列兩種方式之一來執行：

- 全新安裝作業系統
- 原地升級作業系統

全新安裝作業系統

1. 備份現有的 Java DB 資料。

如需要資料的位置，請參閱 [第 147 頁的「Java DB 資料」](#)。

2. 安裝新的作業系統。

可將作業系統安裝在新系統 (或 Solaris 10 區域) 上，亦可讓其徹底覆蓋現有的檔案系統。

3. 安裝發行版本 5 Java DB。
4. 復原[步驟 1](#) 中所備份的 Java DB 資料。

原地升級作業系統

1. 備份現有的 Java DB 資料。

如需必要資料的位置，請參閱第 147 頁的「[Java DB 資料](#)」。

2. 升級作業系統。

升級時會原樣保留現有的檔案系統。

3. 升級到發行版本 5 Java DB。

請參閱第 149 頁的「[從 Java ES 發行版本 4 升級 Java DB](#)」。

Java DB 資料應當不會受升級影響。

從 Java ES 發行版本 4 升級 Java DB

本節包括將 Java DB 從 Java ES 2005Q4 (發行版本 4) 升級至 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Java DB](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Java DB 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 藉由執行發行版本 5 Java DB 的全新安裝並取代發行版本 4 版本來達成升級。發行版本 4 資料和配置會保持不變。
- **升級相依性。** Java DB 對 J2SE 共用元件有硬性升級相依性 (請參閱第 43 頁的表 1-9)，當您升級 Java DB 時，Java ES 安裝程式會將其自動升級到發行版本 5。
- **向下相容性。** 發行版本 5 Java DB 完全向下相容發行版本 4。
- **升級回復。** 除非復原為所備份的發行版本 4 安裝，否則無法達成發行版本 5 的升級回復。
- **平台問題。** 升級 Java DB 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 Java DB

本節描述如何在 Solaris 及 Linux 平台上將 Java DB 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 Java DB](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Java DB 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Java DB 相依元件](#)
- [備份 Java DB 資料](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可使用下列指令驗證 Java DB 的目前版本：

```
java -cp JavaDB-base/lib/derby.jar org.apache.derby.tools.sysinfo
```

版本資訊 (針對 `derby.jar` 檔案) 顯示在下表中：

表 8-4 Java DB 版本驗證輸出

Java ES 發行版本	Java DB 版本編號
發行版本 4	10.0.2.1
發行版本 5	10.1.3.1

升級 Java DB 相依元件

一般的建議是將電腦系統 (和運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。但是，當您將 Java DB 升級到發行版本 5 時，Java ES 安裝程式會自動升級 Java DB 所需要的所有共用元件。

備份 Java DB 資料

Java DB 從發行版本 4 到發行版本 5 的升級不會修改配置資料或永久性資料。但是，為安全起見，您應對您的整個 Java DB 安裝及資料進行備份。請參閱[第 147 頁的表 8-3](#)。

取得必要的配置資訊和密碼

升級 Java DB 時無須使用配置資訊或密碼。

升級發行版本 4 Java DB

本節描述 Solaris 和 Linux 平台上的升級程序。

升級程序

下述程序適用於位於執行升級的電腦本機上的所有 Java DB 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 停止發行版本 4 Java DB。

如果網路伺服器正在執行，請使用以下指令：

```
java -cp JavaDB-base/lib/derby.jar:JavaDB-base/lib/derbynet.jar
org.apache.derby.drda.NetworkServerControl shutdown
```

否則，只需關閉使用 Java DB 的所有應用程式即可。

3. 執行發行版本 5 Java DB 的全新安裝。

執行下列步驟：

- a. 在代管發行版本 4 Java DB 的電腦上啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 從元件選取頁面中選取 Java DB。
將會覆寫先前的安裝。
- c. 選取 [以後配置] 選項。
不支援 [立即配置]。
- d. 如有必要，請選取安裝本土化套裝軟體的選項。
- e. 安裝完成時結束 Java ES 安裝程式。

4. 啟動發行版本 5 Java DB。

如果您正在使用網路伺服器，請使用以下指令：

```
java -jar <JavaDB-base>/lib/derbynet.jar start
```

否則，只需啟動所有在內嵌模式下使用了 Java DB 的應用程式即可。

驗證升級

您可使用下列指令來驗證 Java DB 的升級是否成功：

```
java -cp JavaDB-base/lib/derby.jar org.apache.derby.tools.sysinfo
```

有關輸出值，請參閱第 150 頁的表 8-4 (針對 *derby.jar* 檔案的版本)。

升級後作業

將 Java DB 從發行版本 4 升級到發行版本 5 時，您必須將資料從 Java DB 10.0 磁碟格式轉換為 10.1 格式。若要執行此轉換，請連接到資料庫，在 JDBC URL 的後面附加 `upgrade=true`。例如：

```
java -cp JavaDB-base/lib/derbytools.java:JavaDB-base/lib/derby.jar  
org.apache.derby.tools.ij
```

```
ij version 10.1
```

```
ij> connect 'jdbc:derby:databasePath;upgrade=true';
```

```
ij> exit;
```

回復升級

除非復原為備份發行版本 4 安裝及其資料，否則無法達成發行版本 5 的升級回復。

多重實例升級

在某些部署架構中，會將 Java DB 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Java DB 實例在多台電腦上執行，並透過負載平衡器來分散負載。

如第 149 頁的「[升級發行版本 4 Java DB](#)」中所述，您要在每台電腦上執行 Java DB 的升級。

High Availability Session Store

本章描述如何將 High Availability Session Store 升級到 Java ES 5 (發行版本 5) : High Availability Session Store (HADB) 4.4.3。

本章提供發行版本 5 支援的不同升級路徑之升級問題和程序的一般簡介。本章涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 156 頁的「HADB 升級簡介」](#)
- [第 158 頁的「從 Java ES 發行版本 4 升級 HADB」](#)
- [第 165 頁的「從 Java ES 發行版本 3 升級 HADB」](#)

備註 本章中的檔案位置是以稱為 *HADB-base* 的目錄路徑為基準指定的。初始安裝 HADB 時可能至少已將此路徑中的一部份指定為安裝目錄。若沒有指定，安裝程式會指定一個預設值。

HADB-base 的預設值不依作業系統平台而定，如下表所示。

表 9-1 HADB 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>HADB-base</i> Java ES 安裝程式	/opt/SUNWhadb/ <i>version_number</i>	/opt/SUNWhadb/ <i>version_number</i>
<i>HADB-base</i> 獨立的 Application Server 8.2 EE 安裝程式	/opt/SUNWappserver/appserver /hadb/ <i>version_number</i>	/opt/SUNWappserver/appserver /hadb/ <i>version_number</i>

HADB 升級簡介

本節描述下列會影響 HADB 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 HADB](#)
- [HADB 升級資訊指南](#)
- [HADB 資料](#)
- [HADB 升級策略](#)

關於 Java ES 發行版本 5 HADB

相對於發行版本 4 HADB 而言，HADB 的 Java ES 發行版本 5 版本只進行了小的使用者增強。

HADB 升級資訊指南

[表 9-2](#) 顯示支援的 HADB 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 9-2 至 Java ES 5 (發行版本 5) 的升級路徑：HADB 4.4.3

Java ES 發行版本	HADB 版本	一般方法	需要重新配置
發行版本 4	HADB 4.4.2 (2005Q4)	直接升級： 線上和離線升級方法皆可。	無
發行版本 3	HADB 4.4.1 (2005Q1)	直接升級： 線上和離線升級方法皆可。	無
發行版本 2	HADB 4.4.0-14 (2004Q2)	不支援升級。	無
發行版本 1	未提供	無升級	無
Java ES 發行版本之前的發行版本	未提供	無升級	無

HADB 資料

下表顯示可能受升級 HADB 軟體影響的資料類型。

表 9-3 HADB 資料用途

資料類型	位置	用途
動態應用程式資料	/var/opt/SUNWhadb	High availability session store
配置資料	/etc/opt/SUNWhadb /etc/init.d/ma-initd	高可用性伺服器配置

HADB 升級策略

HADB 的升級策略一般取決於在 [第 1 章「升級規劃」](#) 中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 HADB 升級規劃的問題，藉以對 HADB 進行一般性的介紹。

相容性問題

發行版本 5 HADB 向下相容 Java ES 發行版本 4 隨附的 HADB。

HADB 相依性

發行版本 5 HADB 只對 J2SE 共用元件有相依性：Java™ 2 Platform, Standard Edition (J2SE™) 版本 1.4 或更高版本。

雙重升級

在雙重升級中，HADB 軟體與作業系統均會進行升級 (如 [第 40 頁的「雙重升級：Java ES 與作業系統軟體」](#) 中所述)，這種升級是在 Application Server 雙重升級環境中執行的。如需有關 Application Server 雙重升級的資訊，請參閱 [第 196 頁的「雙重升級」](#)。

從 Java ES 發行版本 4 升級 HADB

本節包括將 HADB 從 Java ES 2005Q4 (發行版本 4) 升級至 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 HADB](#)

簡介

將 Java ES 發行版本 4 HADB 升級到 Java ES 發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 升級包括移除 Java ES 發行版本 4 HADB 套裝軟體以及加入 Java ES 發行版本 5 套裝軟體。有兩種升級方法：
 - **線上升級。** 使用線上升級可避免 HADB 服務中斷。
 - **離線升級。** 若您可在將 HADB 套裝軟體取代為較新版本時中斷 HADB 服務，請使用離線升級。
- **升級相依性。** HADB 不具有硬性升級相依性。HADB 需要 J2SE 版本 1.4 或更高版本，這表示它對 J2SE 有非硬性升級相依性。
- **向下相容性。** Java ES 發行版本 5 提供的 HADB 向下相容 Java ES 發行版本 4 提供的 HADB。
- **升級回復。** 從 Java ES 發行版本 5 到 Java ES 發行版本 4 的升級回復是藉由復原發行版本 4 版本套裝軟體來達成的。
- **平台問題。** 升級 HADB 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 HADB

本節描述如何在 Solaris 及 Linux 平台上將 HADB 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 HADB](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 HADB 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 HADB 相依元件](#)
- [備份目錄資料和配置檔案](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可使用標準版本檢查公用程式來驗證 HADB 的目前版本。例如：

在 Solaris 上：

```
pkgparam -v SUNWhadba
```

在 Linux 上：

```
rpm -qi sun-hadb-a-4.40.3-5.i386.rpm
```

表 9-4 HADB 版本驗證輸出

Java ES 發行版本	HADB 版本編號
發行版本 2	VERSION=4.4.0,REV=14 SUNW_PRODVERS=4.4.0
發行版本 3	VERSION=4.4.1,REV=7 SUNW_PRODVERS=4.4.1
發行版本 4	VERSION=4.4.2,REV=7 SUNW_PRODVERS=4.4.2
發行版本 5	VERSION=4.4.3,REV=5 SUNW_PRODVERS=4.4.3

升級 HADB 相依元件

一般的建議是將電腦系統 (和運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。但是，當您將 HADB 升級到發行版本 5 時，Java ES 安裝程式會自動升級 HADB (即 J2SE) 所需要的所有共用元件。

備份目錄資料和配置檔案

將 HADB 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5 本身並不會修改 HADB 動態資料。不過，您可備份 Java ES 發行版本 4 套裝軟體，以備需要回復升級時使用。

此外，如果自上一次安裝 HADB 之後您已經對它們進行了修改，請備份以下檔案。

```
/etc/opt/SUNWhadb/mgt.cfg  
/etc/init.d/ma-initd
```

取得必要的配置資訊和密碼

升級 HADB 時您必須知道超級使用者的密碼。

升級發行版本 4 HADB

本節討論影響 HADB 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 HADB 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 根據您的產品需求，需要決定線上升級或離線升級何者較合適。
- 下表顯示適用於 Solaris 和 Linux 平台的 Java ES 發行版本 5 升級套裝軟體。Solaris 套裝軟體依安裝順序列出。

表 9-5 用於在 Solaris 平台上升級 HADB 的套裝軟體版本

Solaris 套裝軟體	Linux 套裝軟體
SUNWhadba	sun-hadb-a-4.4.3-5.i386.rpm
SUNWhadbc	sun-hadb-c-4.4.3-5.i386.rpm
SUNWhadbe	sun-hadb-e-4.4.3-5.i386.rpm
SUNWhadbi	sun-hadb-i-4.4.3-5.i386.rpm
SUNWhadbj	sun-hadb-j-4.4.3-5.i386.rpm
SUNWhadbm	sun-hadb-m-4.4.3-5.i386.rpm
SUNWhadbs	sun-hadb-s-4.4.3-5.i386.rpm
SUNWhadbv	sun-hadb-v-4.4.3-5.i386.rpm
SUNWhadbx	sun-hadb-x-4.4.3-5.i386.rpm

線上升級 HADB

當您執行 HADB 線上升級時，首先要在即將升級之叢集中的每個伺服器上安裝、啟動並驗證發行版本 5 HADB。然後為每個伺服器取消註冊較早安裝的 HADB，並註冊新安裝的 HADB 版本。

如需有關執行線上升級的詳細資訊，請參閱「Sun Java System Application Server Enterprise Edition 8.2 高可用性管理指南」中的以下小節：

<http://docs.sun.com/doc/820-0852/6nciif39h?a=view>。

離線升級 HADB

從 Java ES 發行版本 3 或發行版本 4 升級時，可離線升級 HADB。

若要執行離線升級，請關閉 HADB 服務，並將現有 HADB 套裝軟體替代為可從您的 Java ES 5 (發行版本 5) 發行物中取得的較新版本，如第 161 頁的表 9-5 中所示。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 關閉所有 HADB 服務。

- a. 列出所有正在執行的資料庫。

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

- b. 關閉列出的各個資料庫。

```
HADB-base/bin/hadbm stop databaseName
```

如果資料庫已經處於停止狀態，請忽略此訊息。

- c. 在每個執行管理代理程式的主機上關閉 HADB 管理代理程式：

```
/etc/init.d/ma-initd stop
```

(如果您已經使用獨立的 Application Server 安裝程式而非 Java ES 安裝程式安裝了 HADB，則 ma-initd 程序檔位於 *HADB-base/bin* 中。)

3. 啓動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch  
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

4. 在元件選取頁面中選取 High Availability Session Store 4.4。

如果您已經選取了 Application Server Enterprise Edition 8.2，則會自動選取 HADB。

5. 確認升級選擇。

將會升級 HADB 套裝軟體，並顯示升級摘要。

6. 結束 Java ES 安裝程式。

7. 復原第 160 頁的「備份目錄資料和配置檔案」中所備份的檔案。

8. 驗證符號連結 `/opt/SUNWhadb/4` 現在是否指向 *HADB-base*。

例如，對於預設的 *HADB-base*：

```
ls -l /opt/SUNWhadb/4  
lrwxrwxrwx 1 root other 7 Jul 7 23:18 /opt/SUNWhadb/4 ->  
4.4.3-5/
```

9. 重新啓動在步驟 2 中所關閉的 HADB 管理代理程式：

```
/etc/init.d/ma-initd start
```

(如果您已經使用獨立的 Application Server 安裝程式而非 Java ES 安裝程式安裝了 HADB，則 ma-initd 程序檔位於 *HADB-base/bin* 中。)

驗證升級

完成線上升級後，使用下列程序驗證升級。驗證升級成功後，即可刪除舊版安裝的套裝軟體。

若要驗證執行程序是否使用升級後的 HADB 服務，可執行下列步驟。

1. 針對所有執行中的 HADB 服務，發出下列指令之一：

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

例如，

```
HADB-base/bin/ma -V
Sun Java System High Availability Database 4.4 Database Management
Agent
Version      : 4.4.3.5 [V4-5-3-5 2006-03-31 13:59:50 pakker@astra07]
(SunOS_5.9_sparc)
```

2. 檢查資料庫是否正在執行。

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

例如，針對名為 Example DB 的資料庫，輸入以下指令。

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

```
HADB-base/bin/hadbm status ExampleDB
Database      Status
ExampleDB     FaultTolerant
```

```
HADB-base/bin/hadbm status -n ExampleDB
NodeNo   HostName      Port      NodeRole   NodeState  MirrorNode
0        sungod012    15000    active     running    1
1        sungod012    15020    active     running    0
```

所列節點的所有 HADB 服務都應是「running」狀態。

3. 驗證使用 HADB 的所有產品是否都使用了新的 HADB 路徑。

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

例如，針對名為 Example DB 的資料庫，輸入以下指令。

```
HADB-base/bin/hadbm get PackageName ExampleDB  
Attribute Value  
PackageName V4.4.3.5
```

上述指令顯示 HADB 的目前版本。如需詳細列表，可發出下列指令：

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

升級後作業

除了在第 158 頁的「從 Java ES 發行版本 4 升級 HADB」中所述的步驟之外，沒有升級後作業。

回復升級

若要回復 HADB 的升級，請如第 160 頁的「備份目錄資料和配置檔案」中所述，用您先前備份的版本替代所安裝之 HADB 套裝軟體的較新版本。

從 Java ES 發行版本 3 升級 HADB

將 Java ES 2005Q1 (發行版本 3) HADB 升級到發行版本 5 的程序與將發行版本 4 HADB 升級到發行版本 5 的程序相同。

若要將發行版本 3 HADB 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 158 頁的「從 Java ES 發行版本 4 升級 HADB」中的指示。

Message Queue

本章描述如何將 Message Queue 軟體從先前的 Java ES 版本升級到 Java ES 5 (發行版本 5) : Sun Java System Message Queue 3.7 UR1。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 168 頁的「Message Queue 升級簡介」](#)
- [第 173 頁的「從 Java ES 發行版本 4 升級 Message Queue」](#)
- [第 179 頁的「從 Java ES 發行版本 3 升級 Message Queue」](#)
- [第 180 頁的「從 Java ES 發行版本 2 升級 Message Queue」](#)

備註 本章中的檔案位置是以稱爲 *MessageQueue-base* 的固定目錄路徑爲基準指定的。

MessageQueue-base 的值依作業系統平台而定，如下表所示。

表 10-1 Message Queue 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>MessageQueue-base</i>	/usr/bin	/opt/sun/mq/bin

Message Queue 升級簡介

本節描述下列會影響 Message Queue 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Message Queue](#)
- [Message Queue 升級資訊指南](#)
- [Message Queue 資料](#)
- [Message Queue 升級策略](#)

關於 Java ES 發行版本 5 Message Queue

Java ES 發行版本 5 Message Queue 代表相對於發行版本 4 的次要升級。它包含很大程度上的程式碼修正，並未加入小的功能增強。

以往，Message Queue 軟體包含兩種版本，即 Platform Edition 和 Enterprise Edition，分別對應不同的功能集和授權功能。Enterprise Edition 用於在企業產品環境中部署和執行郵件傳送應用程式。Platform Edition 則主要用於部署、除錯和載入測試郵件傳送應用程式和元件。對於版本 5 Message Queue，Platform Edition 已不再使用，Message Queue 包含所有 Enterprise Edition 功能。將較早的 Java ES 發行版本升級到發行版本 5 時，所有已安裝的 Platform Edition 都轉換為完整的 Message Queue 企業級功能。

Message Queue 升級資訊指南

表 10-2 顯示支援的 Message Queue 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 10-2 至 Java ES 5 (發行版本 5) 的升級路徑：Message Queue 3.7 UR1

Java ES 發行版本	Message Queue 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Message Queue 2005Q4 (3.6 SP3) 僅限 Enterprise Edition	直接升級： 使用 Java ES 安裝程式執行。	自動執行資料轉換。
發行版本 3	Sun Java System Message Queue 2005Q1 (3.6) 僅限 Enterprise Edition	直接升級： 使用 Java ES 安裝程式執行。	自動執行資料轉換。
發行版本 2	Sun Java System Message Queue 2004Q2 (3.5 SP1) Platform Edition 和 Enterprise Edition	直接升級： 使用 mqupgrade 程序檔執行。	在 Solaris 平台上自動執行，在 Linux 平台上亦提供 mqmigrate 程序檔。
發行版本 1	Sun Java System Message Queue 2003Q4 (3.0.1 SP2) Platform Edition 和 Enterprise Edition	尚未認證直接升級： 但可使用 mqupgrade 程序檔執行。 ¹	在 Solaris 平台上自動執行，在 Linux 平台上亦提供 mqmigrate 程序檔。
Java ES 發行版本之前的發行版本	Sun Java System Message Queue 3.0.x 以及較早版本 Platform Edition 和 Enterprise Edition	尚未認證直接升級： 但可使用 Java ES 安裝程式執行。	

1. 在執行 mqupgrade 程序檔之前及之後，可能需要備份然後再復原以下檔案：例如 (在 Solaris 平台中)：將 /etc/imq/passwd 與 /etc/imq/accesscontrol.properties 復原為 /var/imq/instances/*instanceName*/etc/

除了表 10-2 中所示之 Message Queue 的 Java ES 發行版本以外，Solaris OS 軟體亦隨附有 Message Queue。可以使用 Java ES 安裝程式將 Message Queue 的隨附版本升級到發行版本 5。

Message Queue 資料

Message Queue 與其他 Java ES 元件類似，會利用各種資料，對於任何特定升級，可能都需要將這些資料遷移至升級的版本。下表顯示可能受升級 Message Queue 軟體影響的資料類型。

表 10-3 顯示在 Solaris 系統上資料的位置。Linux 系統上的位置與此類似，但要依 Message Queue 的版本而定：

- 發行版本 2：以 `/opt/imq` 替代表中的 `/imq`
- 發行版本 3 及更高版本：以 `/opt/sun/mq` 替代表中的 `/imq`

如需更多資訊，請參閱「Message Queue 3.7 UR1 管理指南」
(<http://docs.sun.com/doc/820-0844/6nciaqtg?a=view>)。

在表 10-3 中，*instanceName* 為與資料關聯之 Message Queue 代理程式實例的名稱。

表 10-3 Message Queue 資料用途 (Solaris OS)

資料種類	位置 (在 Solaris 上)	用途
代理程式實例配置特性	<code>/var/imq/instances/instanceName/props/config.properties</code>	代理程式和相關服務配置
動態應用程式資料的永久性存放區	發行版本 2、發行版本 3 和發行版本 4： <code>/var/imq/instances/instanceName/fs350/</code> 發行版本 5： <code>/var/imq/instances/instanceName/fs370/</code> 或 JDBC 可存取資料存放區	儲存訊息、目標、長期訂閱、作業事件和其他動態資料
管理的物件 (物件存放區)	您選擇的本機目錄 或 LDAP Directory Server	用於配置用戶端/代理程式連線的物件
安全性：使用者儲存庫	<code>/var/imq/instances/instanceName/etc/passwd</code> 或 LDAP 目錄伺服器	儲存用於認證和授權的使用者資料
安全性：存取控制檔案 (預設的位置)	<code>/var/imq/instances/instanceName/etc/accesscontrol.properties</code>	設定用於授權使用者存取目標和相關功能的規則
安全性：密碼檔目錄 (預設位置)	<code>/var/imq/instances/instanceName/etc/</code>	儲存加密密碼資訊
安全性：代理程式的金鑰庫檔案位置	<code>/etc/imq/</code>	儲存用於安全郵件傳送的加密憑證資訊

Message Queue 升級策略

Message Queue 的升級策略一般取決於在 [第 1 章「升級規劃」](#) 中介紹的許多問題：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Message Queue 升級規劃的問題，藉以對 Message Queue 進行一般性的介紹。

相容性問題

發行版本 5 Message Queue 並未相對於發行版本 3 或發行版本 4 引入新的不相容之處。不過，相對於發行版本 2 及較早版本卻存在重要的相容性問題。這些問題將在 [第 181 頁的「發行版本 2 相容性問題」](#) 中進行討論。

此外，通常如果在叢集中將發行版本 4 及較早的 Message Queue 代理程式與發行版本 5 Message Queue 代理程式混合使用，則主代理程式必定是較早的發行版本代理程式，且叢集會以較早發行版本 Message Queue 叢集來執行。

Message Queue 相依性

Message Queue 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Message Queue 軟體的程序。例如，Message Queue 介面或功能上的變更可能需要 Message Queue 相依之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Message Queue 對下列 Java ES 元件有相依性：

- **共用元件。** Message Queue 對特定的 Java ES 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))。
- **Directory Server。** Message Queue 對 Directory Server 有選擇性的相依性：您可以將 Message Queue 配置為將管裡的物件和 / 或使用者資料儲存在 LDAP 目錄 (Directory Server) 而非本機中。
- **Web 容器。** Message Queue 對 Web Server、Application Server 或協力廠商 Web 容器有選擇性的相依性，可支援用戶端與代理程式之間的 HTTP 訊息傳送。
- **資料庫。** Message Queue 對 Java DB (或協力廠商資料庫) 有選擇性的相依性，可為 Message Queue 持續性層級提供 JDBC 可存取資料存放區而非平面檔案訊息存放區。
- **Sun Cluster。** Message Queue 對 Sun Cluster 有選擇性的相依性，後者用於提供高可用性支援。

雙重升級

在雙重升級中，Message Queue 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用下列兩種方式之一來執行：

- 全新安裝作業系統
- 原地升級作業系統

全新安裝作業系統

1. 備份現有的 Message Queue 資料。

如需必要資料的位置，請參閱第 170 頁的「[Message Queue 資料](#)」。

2. 安裝新的作業系統。

可將作業系統安裝在新系統 (或 Solaris 10 區域) 上，亦可讓其徹底覆蓋現有的檔案系統。

3. 安裝發行版本 5 Message Queue。

4. 復原或遷移[步驟 1](#) 中所備份的 Message Queue 資料。

在 Linux 上從發行版本 2 Message Queue 進行升級時，會將資料復原至發行版本 5 位置。

原地升級作業系統

1. 備份現有的 Message Queue 資料。

如需必要資料的位置，請參閱第 170 頁的「[Message Queue 資料](#)」。

2. 升級作業系統。

升級時會原樣保留現有的檔案系統。

3. 升級到發行版本 5 Message Queue。

請根據您的升級路徑，參閱本章的相應小節。升級時，現有的 Message Queue 資料應保持不變。

但是，在 Linux 上從發行版本 2 Message Queue 進行升級時，必須將資料移動至升級版本 5 位置。

從 Java ES 發行版本 4 升級 Message Queue

本節包括將 Message Queue 從 Java ES 2005Q4 (發行版本 4) 升級至 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Message Queue](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Message Queue 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 使用 Java ES 安裝程式來執行升級。安裝程式會自動從發行版本 4 遷移配置資料。此外，當第一次執行 `imqbrokerd` 時，將會自動轉換與發行版本 4 相關的動態應用程式資料。針對基於檔案的存放區，這表示 `fs350` 目錄的內容會複製到新的 `fs370` 目錄中。針對 JDBC 存放區，現有資料庫表格將會發生簡單的版本更新。
- **升級相依性。** Message Queue 對許多 Java ES 共用元件具有相依性 (請參閱 [第 43 頁的表 1-9](#))，當您升級 Message Queue 時，Java ES 安裝程式會將所有這些共用元件自動升級到發行版本 5。

此外，發行版本 5 Message Queue 對 Java ES 產品元件有相依性，如 [第 171 頁的「Message Queue 相依性」](#) 中所述。但是，將 Message Queue 升級到發行版本 5 不需要升級這些元件。

- **向下相容性。** 發行版本 5 Message Queue 與發行版本 4 在協定、代理程式相容性、管理的物件、管理工具及用戶端應用程式方面完全相容。
- **升級回復。** 沒有可回復 Message Queue 到發行版本 4 的升級的回復公用程式。您必須移除升級後的元件，並手動還原先前版本和配置資料。
- **平台問題。** 升級 Message Queue 的方法在 Solaris 與 Linux 作業系統上相同。

升級發行版本 4 Message Queue

本節描述如何執行從 Java ES 發行版本 4 至 Java ES 發行版本 5 的 Message Queue 升級：

- [升級前作業](#)
- [升級發行版本 4 Message Queue](#)
- [驗證 Message Queue 升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Message Queue 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Message Queue 相依元件](#)
- [備份 Message Queue](#)

驗證目前版本資訊

可以使用 `-version` 選項啟動 Message Queue 代理程式，藉此確定系統上安裝的 Message Queue 版本和修訂版號碼：

```
imqbrokerd -version
```

表 10-4 Message Queue 版本驗證輸出

Java ES 發行版本	Message Queue 版本編號
發行版本 2	Sun Java (tm) System Message Queue 3 2004Q2 Version: 3.5
發行版本 3	Sun Java (tm) System Message Queue 3 2005Q1 Version: 3.6
發行版本 4	Sun Java (tm) System Message Queue 3 2005Q4 Version:3.6 SP3
發行版本 5	Sun Java (tm) System Message Queue 3.7 Version:3.7 UR1

升級 Message Queue 相依元件

一般的建議是將電腦系統 (及運算環境中) 的所有 Java ES 元件都升級到 Java ES 發行版本 5。Message Queue 只對幾個共用元件有硬性升級相依性。

升級 Message Queue 相依元件時，在升級 Message Queue 之前，應按照下列順序執行 (略過可能已升級過的任何相依元件)。共用元件的升級可由 Java ES 安裝程式自動完成。

1. **共用元件**。在第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Message Queue 升級到發行版本 5 時，Message Queue 所需要的所有共用元件會由 Java ES 安裝程式自動升級。
2. **Sun Cluster (非硬性升級相依性)**。在第 69 頁的第 3 章「Sun Cluster 軟體」中提供了將 Sun Cluster 升級到發行版本 5 的指示。
3. **Directory Server (非硬性升級相依性)**。在第 93 頁的第 5 章「Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。
4. **Java DB (非硬性升級相依性)**。當升級 Message Queue 時，您需要執行發行版本 5 Java DB 的全新安裝。
5. **Web 容器軟體 (非硬性升級相依性)**。在第 123 頁的第 7 章「Web Server」和第 189 頁的第 11 章「Application Server」中分別提供了升級 Web Server 或 Application Server 的指示。

備份 Message Queue

在生產環境中，在執行升級之前備份應用程式資料永遠都是一個好做法。請注意動態應用程式資料的永久性存放區的位置，如第 170 頁的表 10-3 所述。

升級發行版本 4 Message Queue

升級程序包括下列步驟：

1. 停止任何執行中的 Message Queue 用戶端應用程式。

若 Message Queue 正在 Application Server 環境中使用，同樣要關閉 Application Server。

2. 停止任何執行中的代理程式。系統會提示您輸入管理員使用者名稱和密碼。

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

3. 若您不想保留動態資料、Message Queue 平面檔案使用者儲存庫以及與各代理程式相關的 Message Queue 存取控制檔案，則使用下列指令移除這些資料。

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

否則，將繼續保留動態資料和配置資訊，並用於發行版本 5 Message Queue。

4. 以超級使用者的身份登入。

```
su -
```

5. 啓動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch  
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

6. 在元件選取頁面中選取 Message Queue。

7. 確認升級選擇。

將會升級 Message Queue 套裝軟體，並顯示升級摘要。

8. 結束 Java ES 安裝程式。

驗證 Message Queue 升級

完成升級程序後，使用 `-version` 選項啓動 Message Queue 代理程式，以驗證升級是否成功。

```
imqbrokerd -version
```

該指令會傳回 Java ES 版本編號以及 Message Queue 特定的版本編號。

升級後作業

若您已升級 Web 容器且正在使用 Message Queue HTTP 通道 servlet，可能需要將它重新部署在新的 Web 容器中。在發行版本 4 與發行版本 5 之間，沒有對 HTTP 通道 servlet 做任何變更。如需有關 HTTP 支援的更多資訊，請參閱「Message Queue 3.7 UR1 管理指南」(<http://docs.sun.com/doc/820-0844>)。

此外，如果您確定不需要回復升級，您可以移除 `fs350` 目錄中的發行版本 4 基於檔案的資料存放區 (請參閱第 170 頁的表 10-3)。

回復升級

未提供用於將 Message Queue 回復到其升級前狀態的程序檔。必須依下列步驟手動執行回復：

1. 停止任何執行中的 Message Queue 用戶端應用程式。

2. 停止任何執行中的代理程式。系統會提示您輸入管理員使用者名稱和密碼。

```
imqcmd shutdown bkr [-b hostName:port]
```
3. 若您想刪除動態資料、Message Queue 平面檔案使用者儲存庫以及與各代理程式實例相關的 Message Queue 存取控制檔案，請使用下列指令移除這些資料。

```
imqbrokerd -name instanceName -remove instance
```
4. 請以超級使用者身份登入或成為超級使用者。

```
su -
```
5. 使用下列指令，擷取已安裝之 Message Queue 套裝軟體的清單：

在 Solaris 上：

```
pkginfo | grep -i "message queue"
```

在 Linux 上：

```
rpm -qa | grep mq
```
6. 使用下列指令，移除 Message Queue 套裝軟體：

在 Solaris 上：

```
pkgrm packageName
```

其中 *packageName* 是指任意 Message Queue 套裝軟體。若要移除多個套裝軟體，請以空格分隔套裝軟體名稱。

在 Linux 上：

```
rpm -e --nodeps RPMName
```

其中 *RPMName* 是指任意 Message Queue rpm 元件。若要移除多個元件，請以空格分隔 RPM 名稱。

由於其他產品可能正在使用 Message Queue 套裝軟體，移除時請務必小心。pkgrm 指令會在移除套裝軟體前警告您與該套裝軟體相關的相依性。螢幕上顯示提示時，請鍵入 **y** (是) 以確認您的移除請求。
7. 鍵入「q」退出。
8. 結束 root shell。
9. 重新安裝發行版本 4 Message Queue。
 使用 Java ES 發行版本 4 安裝程式。
10. 復原在[第 175 頁](#)的「[備份 Message Queue](#)」中備份的發行版本 4 Message Queue 資料。
 發行版本 4 Message Queue 將會使用升級到發行版本 5 之前所備份的資料正常工作。

多重實例升級

若要升級 Message Queue 叢集 (其中多個代理程式可進行互動式操作，以提供可延伸的郵件服務)，可執行漸進式升級，在此過程中，當每個 Message Queue 實例從發行版本 4 升級到發行版本 5 時，叢集仍保持在線上。執行叢集升級時要記住兩點：

- 當關閉代理程式以進行升級時，在重新啓動代理程式之前，無法使用它所儲存的永久郵件。
- 主代理程式應最後升級。

除此之外升級程序是頗直接了當的：先關閉代理程式，執行升級，等全部升級執行完畢後，再重新啓動代理程式。

從 Java ES 發行版本 3 升級 Message Queue

將 Java ES 2005Q1 (發行版本 3) Message Queue 升級到發行版本 5 的程序與將發行版本 4 Message Queue 升級到發行版本 5 的程序相同。

若要將發行版本 3 Message Queue 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 173 頁的「[從 Java ES 發行版本 4 升級 Message Queue](#)」中的指示。

從 Java ES 發行版本 2 升級 Message Queue

本節包括將 Message Queue 從 Java ES 2004Q2 (發行版本 2) 升級到 Java ES 發行版本 5 的相關資訊。本節涵蓋下列主題：

- [簡介](#)
- [發行版本 2 相容性問題](#)
- [升級發行版本 2 Message Queue](#)
- [多重實例升級](#)

備註 如果您要在 Linux 平台上從發行版本 2 Message Queue 進行升級，則必須執行雙重升級，也就是 Message Queue 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Message Queue)。如需更多資訊，請參閱第 172 頁的「[雙重升級](#)」。

簡介

將 Java ES 發行版本 2 Message Queue 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 使用 `mqupgrade` 程序檔執行升級，它會將先前的軟體套裝軟體取代為新的套裝軟體，並自動從發行版本 2 遷移配置資料。
- **升級相依性。** 在電腦上從發行版本 2 升級任何 Java ES 元件都需要升級該電腦代管的所有其他 Java ES 元件，不支援選擇性地將 Java ES 元件從發行版本 2 升級到發行版本 5。特別的是，必須升級 Message Queue 所使用的所有 Java ES 共用元件。

此外，發行版本 5 Message Queue 選擇性地對 Directory Server 和 Web Server (或 Application Server) 有相依性，如第 171 頁的「[Message Queue 相依性](#)」中所述。如果它們已安裝在同一台電腦上，也需要將這些元件升級到發行版本 5。

- **向下相容性。** 發行版本 5 Message Queue 不完全向下相容發行版本 2，如以下的「[發行版本 2 相容性問題](#)」所述。
- **升級回復。** 目前不支援從發行版本 5 回復到發行版本 2 (請參閱「[回復升級](#)」)。
- **平台問題。** 升級 Message Queue 的一般方法在 Solaris 與 Linux 作業系統上相同，但在 Linux 上還需要其他一些程序。以下程序指出平台特定的指令、檔案位置或程序 (如適用)。

發行版本 2 相容性問題

發行版本 5 Message Queue 相對於發行版本 2 及較早版本引入了以下一般性的 Message Queue 相容性問題。

協定相容性

Message Queue 對 Web 容器有相依性，以提供 Message Queue 用戶端和代理程式之間的 HTTP 協定支援。由於協定的變更，使用 Sun Java System Web Server 為 Message Queue `mqhttp.war` 應用程式提供 Web 容器時，若未升級 Message Queue，便不能升級 Web Server 元件（請參閱第 176 頁和第 187 頁上的「升級後作業」）。

代理程式相容性

發行版本 5 Message Queue 代理程式將與發行版本 4、發行版本 3 或發行版本 2 代理程式協同作業，但相對於發行版本 2 的代理程式特性和永久存放區模式的變更會影響相容性。

發行版本 5 Message Queue 可使用發行版本 4、發行版本 3 和發行版本 2 的資料，但在 Linux 系統上除外，在 Linux 系統上，發行版本 2 必須先遷移到發行版本 5。

升級到發行版本 5 Message Queue 時，應考慮下列事項：

- 您可使用較早的 Message Queue `config.properties` 檔案。亦可將它們複製到另一個位置，然後在您配置發行版本 5 Message Queue 代理程式時，查看這些檔案包含的特性設定。
- 第一次啟動代理程式時，任何永久性的 Message Queue 資料（郵件、目標、長期訂閱）在需要時皆會自動轉換為發行版本 5 Message Queue 資料。例如，在需要時，任何現有目標皆會轉換為發行版本 5 Message Queue 目標，且保留現有屬性並使用新屬性的預設值。
- 如果在叢集中混合使用 Message Queue 發行版本 2 代理程式和 Message Queue 發行版本 5 代理程式，則主代理程式必定是 Message Queue 發行版本 2 代理程式（取較舊版本），且叢集會以 Message Queue 發行版本 2 的叢集來執行。

管理的物件相容性

發行版本 5 Message Queue 管理的物件與發行版本 3 及發行版本 4 管理的物件相同。不過，相對較早版本而言，部份發行版本 3 管理的物件已被重新命名或增加了新的屬性。因此，從發行版本 2 Message Queue 升級到發行版本 5 時，您應考慮下列事項：

- 您可使用在發行版本 2 中建立的物件存放區和管理的物件；但是，最好是將管理的物件遷移到發行版本 5。在執行更新作業時，管理主控台 (imqadmin) 和 ObjectManager 命令行公用程式 (imqobjmgr) 會將發行版本 2 管理的物件轉換為發行版本 5 管理的物件。
- 發行版本 5 用戶端執行階段將會查找及例項化發行版本 2 的管理的物件，並將其轉換，以供發行版本 5 用戶端使用。但**不會**轉換進行查找時所在之物件存放區內的發行版本 2 管理的物件。
- 現有發行版本 2 用戶端 (應用程式和/或元件) (亦即直接實例化而非查找管理物件的用戶端) 與發行版本 5 相容。不過，如果它們要使用**新的**管理物件屬性 (如需有關管理物件屬性的資訊，請參閱「Message Queue 3.7 UR1 管理指南」的第 16 章 (<http://docs.sun.com/doc/820-0844>))，則需對其進行重寫。(以發行版本 5 重新編譯發行版本 2 用戶端時，會顯示在發行版本 5 中重新命名了哪些 Message Queue 發行版本 2 屬性。舊名稱仍可使用。)
- 用於啟動 Java 用戶端以及使用命令行選項設定管理物件屬性值的程序檔，與發行版本 5 相容。不過，如果它們要使用**新的**管理物件屬性 (如需有關管理物件屬性的資訊，請參閱「Message Queue 3.7 UR1 管理指南」的第 16 章 (<http://docs.sun.com/doc/820-0844>))，則需對其進行重寫。

管理工具相容性

由於在發行版本 3 中增加了新的指令和管理功能，所以發行版本 5 管理工具 (管理主控台和命令行公用程式) 只與發行版本 3、發行版本 4 和發行版本 5 代理程式配合使用。但仍支援發行版本 2 的所有指令和指令選項。

用戶端相容性

發行版本 3 和發行版本 4 用戶端與發行版本 5 Message Queue 完全相容。然而從發行版本 2 升級到發行版本 5 時，針對 Java 用戶端您應考慮下列相容性問題：

- 發行版本 5 代理程式支援發行版本 2 用戶端 (但不具備其他發行版本 5 功能)。
- 發行版本 5 Java 用戶端可連線發行版本 2 代理程式 (但不具備其他發行版本 5 功能)。
- 發行版本 5 和以 Enterprise Edition 授權或 Platform Edition 試用授權執行的發行版本 2、發行版本 3 或發行版本 4 代理程式支援 C 用戶端程式。

升級發行版本 2 Message Queue

本節描述如何執行從 Java ES 發行版本 2 至 Java ES 發行版本 5 的 Message Queue 升級：

- 升級前作業
- 升級發行版本 2 Message Queue (Solaris)
- 升級發行版本 2 Message Queue (Linux)
- 安裝相容性套裝軟體 (Linux)
- 驗證 Message Queue 升級
- 升級後作業
- 回復升級

升級前作業

升級 Message Queue 軟體之前，應該執行下列作業：

- 驗證目前版本資訊
- 升級 Message Queue 相依元件
- 備份 Message Queue

驗證目前版本資訊

可以使用 `-version` 選項啓動 Message Queue 代理程式，藉此確定系統上安裝的 Message Queue 版本和修訂版號碼：

```
imqbrokerd -version
```

表 10-5 Message Queue 版本驗證輸出

Java ES 發行版本	Message Queue 版本編號
發行版本 2	Sun Java (tm) System Message Queue 3 2004Q2 Version: 3.5
發行版本 3	Sun Java (tm) System Message Queue 3 2005Q1 Version: 3.6
發行版本 4	Sun Java (tm) System Message Queue 3 2005Q4 Version:3.6 SP3
發行版本 5	Sun Java (tm) System Message Queue 3.7 Version:3.7 UR1

升級 Message Queue 相依元件

一般的建議是將電腦系統 (及運算環境中) 的所有 Java ES 元件都升級到 Java ES 發行版本 5。Message Queue 只對幾個共用元件有硬性升級相依性。

升級 Message Queue 相依元件時，在升級 Message Queue 之前，應按照下列順序執行 (略過可能已升級過的任何相依元件)。

1. **共用元件**。在第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了將 Java ES 共用元件升級到發行版本 5 的指示。
2. **Sun Cluster (非硬性升級相依性)**。在第 69 頁的第 3 章「Sun Cluster 軟體」中提供了將 Sun Cluster 升級到發行版本 5 的指示。
3. **Directory Server (非硬性升級相依性)**。在第 93 頁的第 5 章「Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。
4. **Java DB (非硬性升級相依性)**。當升級 Message Queue 時，您需要執行發行版本 5 Java DB 的全新安裝。
5. **Web 容器軟體 (非硬性升級相依性)**。在第 123 頁的第 7 章「Web Server」和第 189 頁的第 11 章「Application Server」中分別提供了升級 Web Server 或 Application Server 的指示。

備份 Message Queue

在生產環境中，在執行升級之前備份應用程式資料永遠都是一個好做法。針對 Solaris OS，動態資料會儲存在以下目錄中：

`/var/imq/instances/instanceName`。

對於其他作業系統，請參閱「Message Queue 3.7 UR1 管理指南」(<http://docs.sun.com/doc/820-0844>)。

升級發行版本 2 Message Queue (Solaris)

將 Message Queue 軟體升級到 Java ES 發行版本 5 時會使用 `mqupgrade` 程序檔，它會安裝發行版本 5 套裝軟體。

升級程序包括下列步驟：

1. 停止任何執行中的 Message Queue 用戶端應用程式。
若 Message Queue 正在 Application Server 環境中使用，同樣要關閉 Application Server。
2. 停止任何執行中的代理程式。系統會提示您輸入管理員使用者名稱和密碼。
`imqcmd shutdown bkr [-b hostName:port]`

3. 若您不想保留動態資料、Message Queue 平面檔案使用者儲存庫以及與各代理程式相關的 Message Queue 存取控制檔案，則使用下列指令移除這些資料。

```
imgbrokerd -name instanceName -remove instance
```

否則，將繼續保留動態資料和配置資訊，並用於發行版本 5 Message Queue。

4. 以超級使用者的身份登入。

```
su -
```

5. 將目錄變更為 Java ES 發行版本 5 發行軟體 Tools 目錄的位置。

在 Solaris SPARC 上：

```
cd Solaris_sparc/Product/message_queue/Tools
```

在 Solaris x86 上：

```
cd Solaris_x86/Product/message_queue/Tools
```

6. 執行 mqupgrade 程序檔。

- a. 啟動程序檔：

```
./mqupgrade
```

mqupgrade 程序檔會列出安裝的 Message Queue 元件。

- b. 輸入 y (是) 以升級 Message Queue 元件。

mqupgrade 程序檔會偵測並列出已安裝的本土化檔案。

若不需要升級 Message Queue 元件，則輸入 n (否)。mqupgrade 程序檔會結束且不升級任何 Message Queue 元件。

- c. 如果螢幕上出現提示，請輸入 y (是) 以升級本土化檔案。

mqupgrade 程序檔會將輸出資料傳送到下列位置中的記錄檔：

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

升級發行版本 2 Message Queue (Linux)

由於 RHEL 2.1 上僅支援 Java ES 發行版本 2，而不支援 Java ES 發行版本 5，因此在 Linux 平台上將發行版本 2 Message Queue 升級到發行版本 5 會比較複雜。因此需要雙重升級：作業系統和 Message Queue 都必須升級。請參閱第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」。

基本程序是先升級 Linux OS，再升級所有 Message Queue 相依元件，然後升級 Message Queue。

將 Message Queue 從發行版本 2 升級到發行版本 5 的程序包括在 Solaris 系統上不需要的資料遷移步驟，亦即將代理程式實例資料遷移到適當的發行版本 5 位置。若您想在升級到發行版本 5 時保存發行版本 2 的資料，Message Queue 提供的遷移工具 `mqmigrate` 可執行此遷移。

若要從發行版本 2 升級到發行版本 5，請同樣遵循第 184 頁的「升級發行版本 2 Message Queue (Solaris)」中的指示，只是在執行 `mqupgrade` 程序檔 (第 186 頁的步驟 7) 之前，先要執行 `mqmigrate` 程序檔 (第 186 頁的步驟 6)，詳見下列程序。

1. 停止任何執行中的 Message Queue 用戶端應用程式。
2. 停止任何執行中的代理程式。系統會提示您輸入管理員使用者名稱和密碼。

```
imqcmd shutdown bkr [-b hostName:port]
```
3. 若您不想保留動態資料、Message Queue 平面檔案使用者儲存庫以及與各代理程式相關的 Message Queue 存取控制檔案，則使用下列指令移除這些資料。

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

否則，將繼續保留動態資料和配置資訊，並用於發行版本 5 Message Queue。
4. 請以超級使用者身份登入或成為超級使用者。

```
su -
```
5. 將目錄變更為 Java ES 發行版本 5 發行軟體 Tools 目錄的位置。

```
cd Linux_x86/Product/message_queue/Tools
```
6. 使用下列指令遷移代理程式實例資料：

```
./mqmigrate
```

`mqmigrate` 程序檔會將發行版本 2 代理程式實例配置資料移到適當的發行版本 4 位置。
7. 執行 `mqupgrade` 程序檔。
 - a. 啟動程序檔：

```
./mqupgrade
```

`mqupgrade` 程序檔會列出安裝的 Message Queue 元件。
 - b. 輸入 y (是) 以升級 Message Queue 元件。

`mqupgrade` 程序檔會偵測並列出已安裝的本土化檔案。

若不需要升級 Message Queue 元件，則輸入 n (否)。`mqupgrade` 程序檔會結束且不升級任何 Message Queue 元件。

- c. 如果螢幕上出現提示，請輸入 y (是) 以升級本土化檔案。

mqupgrade 程序檔會將輸出資料傳送到下列位置中的記錄檔：

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

安裝相容性套裝軟體 (Linux)

若您的程序檔或您的發行版本 2 用戶端應用程式包含的程序檔依賴於發行版本 5 安裝檔案的位置，則需要安裝 sun-mq-compat 套裝軟體，它包含從發行版本 2 檔案位置到發行版本 5 檔案位置的對稱連結。

sun-mq-compat 套裝軟體位於下列位置，也就是您解壓縮 Java ES 發行版本 5 發行軟體的位置。

```
Linux_x86/Product/message_queue/Packages
```

執行下列步驟以安裝 sun-mq-compat 套裝軟體：

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 在套裝軟體目錄中，輸入下列指令：

```
rpm -ivh --nodeps sun-mq-compat-3.7-RelNo.i386.rpm
```

驗證 Message Queue 升級

完成升級程序後，使用 `-version` 選項啟動 Message Queue 代理程式，以驗證升級是否成功。

```
imqbrokerd -version
```

該指令會傳回 Java ES 版本編號以及 Message Queue 特定的版本編號。

升級後作業

若您使用 HTTP 通道 servlet 提供 HTTP 連線服務支援，則將 Message Queue 從發行版本 2 升級到發行版本 5 時亦會將 servlet 升級。這就要求您在將 Message Queue 升級到發行版本 5 之後，對其進行重新部署。如需有關 HTTP 支援的更多資訊，請參閱「Message Queue 3.7 UR1 管理指南」(<http://docs.sun.com/doc/820-0844>)。

此外，您還需要使用管理主控台 (imqadmin) 和/或 ObjectManager 命令行公用程式 (imqobjmgr) 將發行版本 2 的管理的物件遷移到其發行版本 5 的版本，以執行更新作業。

回復升級

目前不支援將 Message Queue 從發行版本 2 升級到發行版本 5。此程序通常與從發行版本 5 回復到發行版本 4 類似 (請參閱第 176 頁的「回復升級」)。但是，由於將 Message Queue 從發行版本 2 升級到發行版本 5 並不會更新 Java ES 產品登錄，因此 Java ES 安裝程式無法重新安裝發行版本 2 Message Queue。

如需此問題的解決辦法，請諮詢 Sun 服務中心。

多重實例升級

若要升級 Message Queue 叢集 (其中多個代理程式可進行互動式操作，以提供可延伸的郵件服務)，可執行漸進式升級，在此過程中，當每個 Message Queue 實例從發行版本 2 升級到發行版本 5 時，叢集仍保持在線上。執行叢集升級時要記住兩點：

- 當關閉代理程式以進行升級時，在重新啓動代理程式之前，無法使用它所儲存的永久郵件。
- 主代理程式應最後升級。

除此之外升級程序是頗直接了當的：先關閉代理程式，執行升級，等全部升級執行完畢後，再重新啓動代理程式。

Application Server

本章描述如何將 Application Server 升級到 Java ES 5 (發行版本 5) : Sun Java System Application Server Enterprise Edition 8.2 。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 191 頁的「Application Server 升級簡介」](#)
- [第 197 頁的「從 Java ES 發行版本 4 升級 Application Server」](#)
- [第 202 頁的「從 Java ES 發行版本 3 升級 Application Server」](#)
- [第 203 頁的「從 Java ES 發行版本 2 升級 Application Server」](#)
- [第 212 頁的「在 Solaris 10 多重區域環境中升級 Solaris 隨附的 Application Server」](#)

備註 本章中的檔案位置是以稱為 *AppServer8-base* 和 *AppServer8Config-base* (Application Server 8.x) 以及 *AppServer7-base* 和 *AppServer7Config-base* (Application Server 7.x) 的目錄路徑為基準指定的。安裝 Application Server 時可能至少已將這些路徑中的一部份指定為安裝目錄或網域目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。這些目錄路徑的預設值顯示在下表中。

表 11-1 Application Server 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>AppServer8-base</i>	/opt/SUNWappserver/appserver	/opt/sun/appserver
<i>AppServer8Install-base</i>	/opt/SUNWappserver	/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

備註

發行版本 5 Application Server (8.x) 的預設 Application Server 網域名稱是 domain1。換句話說，domain1 是在本章中使用的 *domainName* 變數的預設值。

此外，預設 Directory Administration Service (DAS) 實例名稱是 server。通常不會將 J2EE 應用程式部署到 DAS 實例，但會部署到其他獨立實例。

如需有關在本章中使用的 `asadmin` 指令的更多資訊，請參閱「Sun Java System Application Server Enterprise Edition 8.2 Reference Manual」(<http://docs.sun.com/doc/819-4736>)。

Application Server 升級簡介

本節描述下列會影響 Application Server 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Application Server](#)
- [Application Server 升級資訊指南](#)
- [Application Server 資料](#)
- [Application Server 升級策略](#)

關於 Java ES 發行版本 5 Application Server

Java ES 發行版本 5 Application Server 相對於發行版本 4 而言，是次要發行版本，僅含所選錯誤修正程式。在功能上，發行版本 5 Application Server 與發行版本 4 是相同的。

Application Server 升級資訊指南

有兩組升級路徑適用於 Application Server 到 Java ES 發行版本 5 的升級：

- [表 11-2](#) 顯示支援的升級到 Java ES 發行版本 5 的 Java ES Application Server 升級路徑。此表適用於 Solaris 和 Linux 作業系統。
- [表 11-3](#) 顯示了所支援之 Solaris 隨附 Application Server 到 Java ES 發行版本 5 的升級路徑。Solaris OS 軟體隨附有 Application Server Platform Edition。使用 Java ES 安裝程式可以將 Application Server 隨附的版本升級到發行版本 5 Enterprise Edition，如[表 11-3](#) 中所述。

表 11-2 至 Java ES 5 (發行版本 5) 的升級路徑：Application Server Enterprise Edition 8.2

Java ES			
發行版本	Application Server 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Application Server Enterprise Edition 8.1 2005Q4	直接升級： 使用 Java ES 安裝程式執行。 ¹	無
發行版本 3	Sun Java System Application Server Enterprise Edition 8.1 2005Q1	直接升級： 使用 Java ES 安裝程式執行。	無
發行版本 2	Sun Java System Application Server 7.0 Update 3 (2004Q2) Platform Edition 和 Enterprise Edition	直接升級： 使用 Java ES 安裝程式來升級套裝 軟體，使用 postInstall 程序檔來 執行重新配置，並使用 asupgrade 公用程式來遷移網域資訊。 ²	環境變數、網域及其他配置資料。 J2EE 元件及應用程式需要遷移到 新的 Application Server 環境並重 新部署。
發行版本 1	Sun ONE Application Server 7.0 Update 1 (2003Q4) Platform Edition 和 Enterprise Edition	尚未認證直接升級： 使用從發行版本 2 Application Server 升級時所使用的方法。	環境變數、網域及其他配置資料。 J2EE 元件及應用程式需要遷移到 新的 Application Server 環境並重 新部署。
Java ES 發行版本 之前的發 行版本		無直接升級： 但可以先使用「Java Enterprise System 2005Q1 升級與遷移指南」 (http://docs.sun.com/doc/819-2238) 中說明的程序升級到發行版本 3。 然後從發行版本 3 升級到發行版本 5。	

- 如果您想要從 Application Server 發行版本 4 升級到發行版本 5 而不保留配置或網域資訊，您可以使用 Java ES 發行版本 4 解除安裝程式來解除安裝發行版本 4 Application Server，然後使用 Java ES 發行版本 5 安裝程式來全新安裝發行版本 5 Application Server。但是，如果已經使用 [以後配置] 選項安裝了發行版本 4 Application Server，那麼在解除安裝發行版本 4 Application Server 之前，您必須先建立 \$HOME/.asadminprefs 檔案 (其中 \$HOME 是安裝及執行 Application Server 的使用者的主目錄)。檔案中有以下兩行：
AS_ADMIN_PASSWORD=password
AS_ADMIN_USER=admin
- 如果您要在 Solaris 平台上從發行版本 2 Application Server 升級，請特別小心，因為 Solaris 隨附的 Application Server 與 Java ES 發行版本 2 Application Server 並存在您的電腦上。請參閱表 11-3 及其後的注意事項。

表 11-3 Solaris OS 隨附 Application Server 版本的升級路徑

Solaris			
Solaris OS 版本	Application Server 版本	一般方法	需要重新配置
Solaris 10	Sun Java System Application Server Platform Edition 8.0.0_01	直接升級： 使用 Java ES 安裝程式執行。	無。
Solaris 9	Sun Java System Application Server 7.0.0_03c	直接升級： 如果使用了 Application Server 並建立了網域，則需要遷移網域資訊。使用從發行版本 2 Application Server 升級中所記錄的方法，使用 Java ES 安裝程式的 [以後配置] 選項、postInstall 程序檔及 asupgrade 公用程式。 如未使用 Application Server，或在未建立網域的情況下使用它，請使用從發行版本 2 Application Server 升級中所記錄的方法，但請勿使用 Java ES 安裝程式的 [立即配置] 選項，並請勿使用 postInstall 程序檔或 asupgrade 公用程式。	環境變數、網域 (如已建立) 及其他配置資料。 J2EE 元件及應用程式需要遷移到新的 Application Server 環境並重新部署。

備註

如果您要在 Solaris 平台上從發行版本 2 Application Server 升級，請特別小心，因為隨附的版本與 Application Server 的 Java ES 版本並存在您的電腦上。因此您必須先解除安裝隨附的版本 (及任何相應的網域) 才能繼續從發行版本 2 版本升級。請參閱第 204 頁的「[僅限於 Solaris OS：手動移除隨附於作業系統的 Application Server 套裝軟體](#)」。

Application Server 資料

下表顯示可能受升級 Application Server 軟體影響的資料類型。

表 11-4 Application Server 資料用途

資料類型	位置	用途
環境變數	<i>AppServer8-base/config/asenv.conf</i>	全域變數
配置資料	發行版本 3、發行版本 4 和發行版本 5： <i>domain.xml</i> 和 <i>server.policy</i> 檔案，位於 <i>AppServer8Config-base/domains/domainName/config</i> 中 發行版本 2： <i>server.xml</i> 和 <i>server.policy</i> 檔案，位於 <i>AppServer7Config-base/domains/domainName/instanceName/config</i>	Application Server 實例的配置
部署資料	發行版本 3、發行版本 4 和發行版本 5： <i>AppServer8Config-base/domains/domainName/applications</i> 發行版本 2： <i>AppServer7Config-base/domains/domainName/instanceName/applications</i>	配置 J2EE 容器，用於特定的 J2EE 元件及應用程式
存取記錄檔	發行版本 3、發行版本 4 和發行版本 5： <i>AppServer8Config-base/domains/domainName/logs/access/</i> 包含兩個檔案： <i>server_acces_log</i> 和 <i>_asadmin_access_log</i> 發行版本 2： <i>AppServer7Config-base/domains/domainName/instanceName/logs/access</i>	存取記錄

Application Server 升級策略

Application Server 的升級策略一般取決於在[第 1 章「升級規劃」](#)中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Application Server 升級規劃的問題，藉以對 Application Server 進行一般性的介紹。

相容性問題

發行版本 5 Application Server 沒有推出任何關係到發行版本 4 或發行版本 3 的介面變更。但是，在發行版本 5 與發行版本 2 之間有很大的介面變更，使得發行版本 5 與發行版本 2 不相容。

但是，發行版本 5 Application Server 並不支援發行版本 4 Service Registry。如果將 Application Server 升級到發行版本 5，也必須將 Service Registry 升級到發行版本 5。

Application Server 相依性

Application Server 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Application Server 軟體的程序。例如，Application Server 介面或功能上的變更可能需要 Application Server 相依之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Application Server 對下列 Java ES 元件有相依性：

- **共用元件。** Application Server 對特定的 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。
- **Message Queue。** Application Server 依靠 Message Queue 來提供 J2EE Java Message Service 相容的非同步傳訊支援。
- **High Availability Session Store。** Application Server 相依於 High Availability Session Store (HADB) 來維持在實例之間支援容錯移轉所需的階段作業狀態資訊。

備註

如果您早先安裝了不需要 HADB 的 Application Server，則必須先安裝 HADB，才能將 Application Server 升級到發行版本 5。例如，必須安裝發行版本 5 HADB，方可升級 Solaris OS 隨附的 Application Server 版本。Java ES 安裝程式將會在這些情況下自動執行 HADB 的安裝。

- **Java DB。** Application Server 依靠 Java DB 做為預設開發者資料庫來儲存範例應用程式資料以及企業 Java Bean 計時器所需的資料。
- **Web 容器 (選擇性相依性)。** Application Server 相依於 Web 容器服務，以取得它的選擇性負載平衡外掛程式。此支援可由 Java ES Web Server 或協力廠商 Web 容器 (像是 Apache Web Server 及 Microsoft IIS) 提供。

雙重升級

在雙重升級中，Application Server 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用下列兩種方式之一來執行：

- 全新安裝作業系統
- 原地升級作業系統

全新安裝作業系統

1. 備份現有的 Application Server 資料。

如需必要資料的位置，請參閱第 194 頁的「[Application Server 資料](#)」。

2. 安裝新的作業系統。

只能在同一台電腦安裝作業系統，它會徹底覆蓋現有的檔案系統。

3. 復原[步驟 1](#)中所備份的 Application Server 資料。

4. 安裝發行版本 5 Application Server。

請根據[步驟 1](#)中所備份的 Application Server 資料版本，使用本章相關升級小節中所述的程序。

原地升級作業系統

1. 備份現有的 Application Server 資料。

如需必要資料的位置，請參閱第 194 頁的「[Application Server 資料](#)」。

2. 升級作業系統。

升級時會原樣保留現有的檔案系統。

3. 升級到發行版本 5 Application Server。

請根據升級路徑，參閱本章的相關章節。

從 Java ES 發行版本 4 升級 Application Server

本節包括將 Application Server 從 Java ES 2005Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Application Server](#)

簡介

將 Java ES 發行版本 4 Application Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 使用 Java ES 安裝程式來執行升級。將 Application Server 從發行版本 4 升級到發行版本 5 不需要重新配置 Application Server 及重新配置或遷移 J2EE 元件。
- **升級相依性。** Application Server 對許多 Java ES 共用元件具有相依性 (請參閱 [第 43 頁的表 1-9](#))，當您升級 Application Server 時，Java ES 安裝程式會將其相依的所有共用元件自動升級到發行版本 5。Application Server 只對 NSS 共用元件有硬性升級相依性。

此外，如 [第 195 頁的「Application Server 相依性」](#) 所述，發行版本 5 Application Server 對 Message Queue、HADB 和 Java DB 有相依性。這些是硬性升級相依性：全部都必須升級到發行版本 5。

此外，Application Server 選擇性地對 Java ES Web Server 或協力廠商 Web 容器有相依性。不過，這些並非硬性的升級相依性；升級 Web 容器對將 Application Server 升級到發行版本 5 而言是選擇性的。

- **向下相容性。** 發行版本 5 Application Server 向下相容發行版本 4 版本。
- **升級回復。** 從發行版本 4 到發行版本 5 的升級無法回復。
- **平台問題。** 升級 Application Server 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 Application Server

本節描述如何在 Solaris 及 Linux 平台上將 Application Server 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 Application Server](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Application Server 軟體之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Application Server 相依元件](#)
- [備份 Application Server 資料](#)
- [取得所需的配置資訊和密碼](#)

驗證目前版本資訊

您可輸入下列指令來驗證 Application Server 的目前版本：

```
AppServer8-base/bin/asadmin version --verbose
```

表 11-5 Application Server 版本驗證輸出

Java ES 發行版本	Application Server 版本編號
發行版本 2	Sun ONE Application Server 7.0.0_03c
發行版本 3	Sun Java Enterprise System Application Server Enterprise Edition 8.1
發行版本 4	Sun Java Enterprise System Application Server Enterprise Edition 8.1_02
發行版本 5	Sun Java Enterprise System Application Server Enterprise Edition 8.2

升級 Application Server 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。Application Server 對 NSS 共用元件及 Message Queue、HADB 與 Java DB 產品元件有硬性升級相依性。

升級 Application Server 相依元件時，在升級 Application Server 之前，應按照下列順序執行 (略過可能已升級過的任何相依元件)。但是，升級 Application Server 時，共用元件、Message Queue、HADB 和 Java DB 的升級通常可由 Java ES 安裝程式自動完成。

1. **共用元件**。在 [第 59 頁](#) 的「[升級 Java ES 共用元件](#)」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Application Server 升級到發行版本 5 時，Application Server 所需要的所有共用元件會由 Java ES 安裝程式自動升級。
2. **Message Queue**。在 [第 167 頁](#) 的 [第 10 章「Message Queue」](#) 中提供了將 Message Queue 升級到發行版本 5 的指示。
3. **High Availability Session Store (HADB)**。在 [第 155 頁](#) 的 [第 9 章「High Availability Session Store」](#) 中提供了升級 HADB 的指示。
4. **Java DB**。在 [第 145 頁](#) 的 [第 8 章「Java DB」](#) 中提供了升級 Java DB 的指示。
5. **Web 容器軟體 (非硬性升級相依性)**。在 [第 123 頁](#) 的 [第 7 章「Web Server」](#) 中提供了升級 Web Server 的指示。

備份 Application Server 資料

Application Server 從發行版本 4 到發行版本 5 的升級不會修改配置資料。因此不需要備份目前的資料。

取得所需的配置資訊和密碼

您應該知道 Application Server 目前安裝版本的管理員使用者 ID 和密碼。

升級發行版本 4 Application Server

本節討論影響 Application Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Application Server 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 應該先關閉在 Application Server 實例中執行的任何 J2EE 元件，然後再升級該實例。但是，如果負載平衡提供高可用性或延展性，則此要求可以放寬。

- 升級已安裝影像時，必須先關閉在一台電腦上執行的所有 Application Server 實例 (全都對應到相同的已安裝 Application Server 影像)。
- 在多重節點部署中，在管控 Application Server 實例的每個節點或電腦上執行升級程序。

升級程序

下述程序適用於位於執行升級的電腦本機上的 Application Server 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 關閉要升級的 Application Server 實例中正在執行的所有 J2EE 元件。
3. 關閉位於要升級之電腦上的所有 Application Server 實例。

- a. 停止所有執行中的節點代理程式。

```
AppServer8-base/bin/asadmin stop-node-agent --user admin_ID  
nodeagentName
```

其中 *nodeagentName* 的形式為 *hostName_domainName*，但簡單預設為 *hostName*。

- b. 停止 Domain Administration Server (DAS)。

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID  
domainName
```

- c. 停止 PointBase 資料庫伺服器 (如果正在使用中)。

```
AppServer8Config-base/appserver/pointbase/tools/stopserver.sh
```

4. 啓動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch  
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 *installer -nodisplay* 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

5. 在元件選取頁面中選取 Application Server。

也會自動選取硬性升級相依性、Message Queue、HADB 和 Java DB 以進行升級。

6. 選擇 [立即配置]。

7. 指定請求的配置值。

您將會看到一些配置面板。

8. 確認升級選擇。

將會升級 Application Server 套裝軟體 (如有必要，也會升級 Message Queue、HADB 和 Java DB 的套裝軟體) 並顯示升級摘要。

9. 結束 Java ES 安裝程式。

10. 重新啓動升級後的 Domain Administration Server (DAS)。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
domainName
```

11. 重新啓動更新的 Application Server 實例。

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
nodeagentName
```

其中 *nodeagentName* 的形式為 *hostName_domainName*，但簡單預設為 *hostName*。

驗證升級

您可使用下列指令來驗證升級是否成功：

```
AppServer8-base/bin/asadmin version --verbose
```

請參閱第 198 頁的表 11-5 中的輸出值。

升級後作業

如果您要繼續使用內嵌式的發行版本 4 PointBase 資料庫而非 Java DB (一個新的發行版本 5 Java ES 產品元件)，則必須手動編輯升級的 *AppServer8-base/config/asenv.conf* 檔案。

從發行版本 4 升級 Application Server 之後，*asenv.conf* 中的 PointBase 設定如下：

```
AS_POINTBASE="%POINTBASE_HOME%"
AS_POINTBASE_SAMPLESDB="%POINTBASE_SAMPLESDB%"
```

將這些設定變更為以下值：

```
AS_POINTBASE="AppServer8-base/pointbase"
AS_POINTBASE_SAMPLESDB="AppServerConfig8-base/var/appserver/pointbase"
```

回復升級

不支援發行版本 5 升級的回復。

從 Java ES 發行版本 3 升級 Application Server

將 Java ES 2005Q1 (發行版本 3) Application Server 升級到發行版本 5 的程序與將發行版本 4 Application Server 升級到發行版本 5 的程序相同。

若要將發行版本 3 Application Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 197 頁的「[從 Java ES 發行版本 4 升級 Application Server](#)」中的指示。

從 Java ES 發行版本 2 升級 Application Server

本節包括將 Application Server 從 Java ES 發行版本 2 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 2 Application Server](#)

備註 如果您要在 Linux 平台上從發行版本 2 Application Server 進行升級，則必須執行雙重升級，也就是 Application Server 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Application Server)。如需更多資訊，請參閱第 196 頁的「[雙重升級](#)」。

簡介

將 Java ES Application Server 從發行版本 2 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 藉由使用 Java ES 安裝程式並選擇 [以後配置] 選項安裝發行版本 5 Application Server 來執行升級。接著使用 `asupgrade` 公用程式完成重新配置。在升級 Application Server 之後，必須將發行版本 2 的 J2EE 元件及應用程式遷移至發行版本 5。
- **升級相依性。** 在電腦上從發行版本 2 升級任何 Java ES 元件都需要升級該電腦代管的所有其他 Java ES 元件，不支援選擇性地將 Java ES 元件從發行版本 2 升級到發行版本 5。

需特別指出的是，必須升級 Application Server 所需要的所有 Java ES 共用元件。如果 Message Queue 位於相同的電腦上，則也必須升級。必須安裝 High Availability Session Store (HADB) 和 Java DB，如果要針對負載平衡使用 Web Server，則它也必須升級。

- **向下相容性。** 發行版本 5 Application Server 不向下相容發行版本 2 版本。需要遷移 J2EE 元件及應用程式才能在發行版本 5 Application Server 環境中執行。
- **升級回復。** 從發行版本 5 到發行版本 2 的升級的回復只需復原到發行版本 2 安裝就可達成 (升級過程並未移除發行版本 2 配置資料)。
- **平台問題。** 在 Solaris OS 上升級 Application Server 的方法不能直接套用到 Linux OS 上：Linux 2.1 上只支援 Java ES 發行版本 2，而 Linux 2.1 上不支援 Java ES 發行版本 5。因此需要雙重升級：作業系統和 Application Server 都必須升級 (請參閱第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」)。

- **資料位置。** 此處有新的發行版本 5 的配置資料、部署資料和存取記錄檔的位置，如第 194 頁的表 11-4 所指定。

升級發行版本 2 Application Server

本節描述如何在 Solaris 及 Linux 平台上將 Application Server 從 Java ES 發行版本 2 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 2 Application Server](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Application Server 之前，應該執行下列作業：

- [僅限於 Solaris OS：手動移除隨附於作業系統的 Application Server 套裝軟體](#)
- [驗證目前版本資訊](#)
- [升級 Application Server 相依性](#)
- [備份 Application Server 資料](#)
- [取得必要的配置資訊和密碼](#)

僅限於 Solaris OS：手動移除隨附於作業系統的 Application Server 套裝軟體

安裝發行版本 2 Application Server 時並沒有移除 Solaris OS 隨附的 Application Server 版本。若要將發行版本 2 Application Server 正確升級到發行版本 5，您必須執行下列步驟。

1. 手動移除 `/usr/appserver` 目錄中的對應 Application Server 套裝軟體以解除安裝隨附的版本：

```
pkgrm SUNWascmnse SUNWaslb SUNWasut ...
```

其中的完整套裝軟體可使用下列指令取得：

```
pkginfo -i|grep -i `application server`
```

結果應會包含例如以下的套裝軟體：

```
SUNWasacee、SUNWascmnse、SUNWasu、SUNWasuee、SUNWasac、SUNWascmn、
SUNWasdb、SUNWasdem、SUNWasdemdb、SUNWasr、SUNWasut、SUNWasman、
SUNWasjdoc
```

也可能包含本土化套裝軟體：

```
SUNWLocaleasacee、SUNWLocaleascmnse、SUNWLocaleasu、SUNWLocaleasuee
```

2. 如果已經使用了隨附的版本，請刪除網域資訊：

```
rm /usr/appserver/domains
```

驗證目前版本資訊

您可輸入下列指令來驗證 Application Server 的目前版本：

```
AppServer7-base/bin/asadmin version --verbose
```

請參閱第 198 頁的表 11-5 取得版本輸出。

升級 Application Server 相依性

升級 Application Server 相依性需要將 Application Server 相依且位於本機上的所有發行版本 2 元件升級到發行版本 5，特別是 Message Queue 和共用元件。但是，在進行升級程序期間，Java ES 安裝程式會自動升級共用元件 (請參閱第 207 頁的「升級程序」)。

此外，升級 Application Server 需要安裝發行版本 5 Application Server 相依的 HADB 和 Java DB。但是，在進行升級程序期間，Java ES 安裝程式會自動選取和安裝這些產品元件 (請參閱第 207 頁的「升級程序」)。

一般而言，當升級 Application Server 相依性時，在您升級 Application Server 之前，應以下列順序 (跳過任何可能已經升級的部分) 執行。但是，升級 Application Server 時，共用元件的升級與 HADB 和 Java DB 的安裝通常可由 Java ES 安裝程式自動執行。

1. **共用元件。** 在第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Application Server 升級到發行版本 5 時，Application Server 所需要的所有共用元件會由 Java ES 安裝程式自動升級。

備註 將共用元件升級到發行版本 5 包括將 J2SE 升級到 JDK 版本 1.5，發行版本 2 Application Server 不支援此升級。因此，在升級共用元件之前，您應該先關閉發行版本 2 Application Server。當使用 Java ES 安裝程式來安裝或升級任何 Java ES 產品元件 (例如 Directory Server、Message Queue、HADB、Java DB 及其他元件) 時，會將共用元件升級到發行版本 5。

2. **Message Queue。** 請參閱第 180 頁的「從 Java ES 發行版本 2 升級 Message Queue」。Message Queue 無法使用 Java ES 安裝程式從發行版本 2 升級。
3. **High Availability Session Store (HADB)。** HADB 的全新安裝會於升級 Application Server 時由 Java ES 安裝程式執行。
4. **Java DB。** Java DB 的全新安裝會於升級 Application Server 時由 Java ES 安裝程式執行。
5. **Web 容器軟體 (非硬性升級相依性)。** 請參閱第 144 頁的「從 Java ES 發行版本 2 升級 Web Server」。

備份 Application Server 資料

Application Server 從發行版本 2 到發行版本 5 的升級不覆寫發行版本 2 的配置資料。但是，為了安全起見，在升級到發行版本 5 之前，應該備份完整的 domains 目錄：

```
AppServer7Config-base/domains
```

取得必要的配置資訊和密碼

您應該知道關於目前安裝版本的下列資訊：

- Application Server 管理員使用者 ID 和密碼
- 發行版本 2 Application Server 基底目錄

升級發行版本 2 Application Server

本節討論影響 Application Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

將 Application Server 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 應該先關閉在 Application Server 實例中執行的任何 J2EE 元件，然後再升級該實例。但是，如果使用負載平衡提供高可用性或延展性，則此要求可以放寬。
- 必須先關閉在一台電腦上執行的所有 Application Server 實例 (全都對應到相同的已安裝 Application Server 影像)，才能升級已安裝的影像。

升級程序

下述程序適用於位於執行升級的電腦本機上的所有 Application Server 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 如果已將 Java ES 共用元件升級到發行版本 5 (特別是將 J2SE 升級到 JDK 1.5)，而發行版本 2 Application Server 仍在執行中，則請編輯 `asenv.conf` 檔案以直接指向 JDK 1.4。

```
AS_JAVA=/usr/java
```

如果 `asenv.conf` 正在參照 JDK 1.5，則無法關閉 Application Server (如下的步驟 3)。

3. 停止所有 Application Server 和相關程序。

```
AppServer7-base/bin/asadmin stop-appserv domainName
```

4. 安裝發行版本 5 Application Server。

執行下列步驟：

- a. 在代管發行版本 2 Application Server 的電腦上啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 `os_arch` 與您所在平台相符，例如 `Solaris_sparc`。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。) Application Server 不會顯示為可從發行版本 2 升級。

- b. 從元件選取頁面中選取 Application Server。

選擇至少安裝前三個子元件，包括「節點代理程式」元件。

- c. 按 [下一步]。

如果尚未將 Message Queue 升級到發行版本 5，會出現一個錯誤訊息視窗要求您升級 Message Queue。在此情況下，按 [確定] 並升級 Message Queue。程序如第 183 頁的「升級發行版本 2 Message Queue」中所述。

- d. 指定與安裝發行版本 2 Application Server 的目錄不同的安裝目錄。
- e. 選取 [以後配置] 選項。

備註

如果您使用的是 Solaris 9 OS 系統隨附的 Application Server 版本，且從未使用過此版本（尚未建立任何網域資訊），則您可以選取 [立即配置] 並省略步驟 5 及後續步驟。

- f. 如有必要，請選取安裝本土化套裝軟體的選項。
 - g. 安裝完成時結束 Java ES 安裝程式。
5. 執行以下安裝後程序：
- a. 找到 Java ES 發行版本 5 發行物的 Application Server 工具目錄中的 postInstall 讀我檔案：

```
Java ES Release 5 distribution/os_arch/Product/application_svr/Tools
```

其中 *os_arch* 與您所在平台相符，例如 Solaris_sparc。
 - b. 請參閱 ReadMe 檔案並執行 postInstall 程序檔。

```
cd Java ES Release 5 distribution/os_arch/Product/application_svr/Tools  
./postInstall AppServer8Install-base AppServer8Config-base
```

這些程序檔會從安裝期間所安裝的範本配置並建立 *AppServer8-base/bin/*shell* 程序檔和一個 *config/asenv.conf* 檔案。(Java ES 安裝程式通常會建立 *bin/*shell* 程序檔，但如果選擇 [以後配置]，則必須按照說明的方式建立它們。)

備註

當您將 Application Server 升級到發行版本 5 時，會將非管理連接埠 8080 (預設) 和 8181 (安全) 分配給 Domain Administration Server (DAS)。這些連接埠的指定可能會與分配給發行版本 2 Application Server 實例的連接埠衝突。如果遇到這種情形，您必須變更 DAS 連接埠以避免衝突。如需說明，請參閱「Sun Java System Application Server Enterprise Edition 8.2 管理指南」(<http://docs.sun.com/doc/820-0848>) 中有關修改 http 偵聽程式屬性的章節。

- c. 若需要，請修改 `AppServer8-base/config/asenv.conf` 檔案中的環境設定。您必須手動編輯這個檔案。

備註

若要配置 Application Server 以使用負載平衡功能，請參閱「Sun Java System Application Server Enterprise Edition 8.2 高可用性管理指南」的「Application Server 高可用性功能」一章中的「為 HTTP 負載平衡配置 Web 伺服器」一節(<http://docs.sun.com/doc/820-0852>)。

6. 執行 `asupgrade` 公用程式。

`asupgrade` 公用程式會建立一個發行版本 5 節點代理程式，用來遷移發行版本 2 Application Server 實例。

該公用程式位於 Application Server 目錄下，例如：

- 升級精靈模式：`AppServer8-base/bin/asupgrade`
- 升級主控台模式：`AppServer8-base/bin/asupgrade -c`

升級精靈或升級主控台將會引導您完成整個升級步驟。

- a. 針對網域資訊的遷移標示目標與來源目錄：
- 發行版本 2：`AppServer7-base`
 - 發行版本 5：`AppServerConfig8-base/domains`

- b. 提供發行版本 2 Application Server 的管理使用者、管理密碼和主密碼。

如果要求您輸入主密碼，請指定 `changeit`，如果您使用的主密碼不是 `changeit` 的金鑰庫，則應將密碼變更為 `changeit`。

```
jdk-home/bin/keytool -storepasswd -new changeit -keystore keystore
-storepass oldpasswd
```

如需有關 Application Server `asupgrade` 公用程式的更多資訊，請參閱「Sun Java System Application Server Enterprise Edition 8.2 Reference Manual」(<http://docs.sun.com/doc/819-4736>) 與「Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide」(<http://docs.sun.com/doc/819-4737/6n6sao3ju?a=view>) 中的使用者指令章節。

7. 如果您已經重新導向第 207 頁的步驟 2 中的 `asenv.conf` 檔案，請將其復原為指向 JDK 1.5。

```
AS_JAVA=/usr/jdk/entsys-j2se
```

如果 `asenv.conf` 正在參照 JDK 1.4，則無法啟動 Application Server (如下的步驟 8 和步驟 9)。

8. 啟動 Domain Administration Server (DAS)。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
domainName
```

9. 重新啟動升級的 Application Server 實例。

執行方式是啟動遷移的已升級 Application Server 實例所屬的節點代理程式。

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
nodeagentName
```

其中 `nodeagentName` 的形式為 `hostName_domainName`，但簡單預設為 `hostName`。

驗證升級

啟動 [管理主控台] 並驗證這些伺服器已啟動。如果有任何一個伺服器停止執行，則請檢查下列記錄檔是否有因為連接埠衝突造成的失效：

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/logs/server.log
```

如果有連接埠衝突造成的失敗，請使用 [管理主控台] 修改連接埠號以消除衝突，然後停止並重新啟動節點代理程式。

您可輸入下列指令驗證 Application Server 是否升級到發行版本 5：

```
AppServer8-base/bin/asadmin version --verbose
```

請參閱第 198 頁的表 11-5 中的輸出值。

升級後作業

除了第 207 頁的「升級程序」中描述的步驟以外，您還必須執行兩個升級後作業：

- [更正 JSS 的參照](#)
- [遷移發行版本 2 J2EE 元件](#)

更正 JSS 的參照

若要使 Application Server 參照正確版本的 JSS 共用元件，您必須用以下位置的 Application Server domain.xml 檔案中的 jss4.jar 取代 jss3.jar：

AppServer8Config-base/domains/domainName/config/domain.xml

遷移發行版本 2 J2EE 元件

您必須遷移發行版本 2 J2EE 元件及應用程式，以在發行版本 5 Application Server 環境中執行，並將它們重新部署到適當的 Application Server 實例。如需關於遷移 J2EE 元件及應用程式的更多資訊，請參閱「Application Server Enterprise Edition 8.2 Upgrade and Migration Guide」中的第 4 章 (<http://docs.sun.com/doc/819-4737>)。

回復升級

發行版本 5 Application Server 的回覆升級程序只是將發行版本 5 復原為 Application Server 的發行版本 2 版本，升級到發行版本 4 時未移除發行版本 2 版本。

在 Solaris 10 多重區域環境中升級 Solaris 隨附的 Application Server

Solaris 10 OS 隨附有 Application Server (以及 Application Server 所依存的 Message Queue)。除非從全域區域中移除，否則當建立此類區域時，會將隨附的 Application Server 傳遞至非全域區域。在所有區域中存在的隨附的 Application Server 會影響 Application Server 後續升級到發行版本 5：

- 在全域區域中升級 Application Server 會移除全域區域中的隨附版本，並從所有非全域區域中自動移除隨附的版本。
- 在全域區域中將 Application Server 升級到發行版本 5 並不會在非全域區域中升級 Application Server，因為不會傳遞發行版本 5 Application Server 套裝軟體。

提供了以下範例來記錄在 Solaris 10 多重區域環境中升級 Application Server 時所牽涉的一些細微之處。(如需有關 Java ES 和 Solaris 10 區域的更詳細的介紹，請參閱第 54 頁的「Java ES 5 升級與 Solaris 10 區域」。)

範例中的目標是在 Solaris 10 稀疏根區域中升級到 Solaris 隨附的發行版本 5 Application Server (版本 8.0.0_01)。

您無法在稀疏根區域中直接升級 Application Server，因為 Solaris 隨附的 Application Server 已安裝到從全域區域中掛載的唯讀目錄中。因此，若要將稀疏根區域中的 Application Server 升級到發行版本 5，您必須先移除全域區域中的隨附版本。

此外，Message Queue 會安裝到全域區域中，這與實際有所偏離，因為一般情況下只有共用元件 (非產品元件) 才會安裝到全域區域中。這是因為由於唯讀目錄的關係，在稀疏根區域中無法安裝或升級 Message Queue。

在 Solaris 10 稀疏根區域中將 Solaris 隨附的 Application Server (版本 8.0.0_01) 升級到發行版本 5 的程序如下：

1. 驗證系統的初始狀態。

此範例假設全域管理員已經配置、安裝及啟動了含稀疏根區域的 Solaris 10 的版本。

稀疏根區域包含已安裝於全域區域中的所有 Java ES 元件，亦即隨附於 Solaris 10 的 Message Queue 和 Application Server 的版本。

此外，範例假設使用者之前已在稀疏根區域中使用了隨附的 Application Server，且已建立需要保留的管理網域資訊。

2. 在全域區域中升級隨附的 Application Server 版本。

此作業將移除隨附的 Application Server 套裝軟體並用發行版本 5 套裝軟體來取代它們。移除的隨附套裝軟體會傳遞到稀疏根區域，在稀疏根區域中有效解除安裝 Application Server 套裝軟體，但發行版本 5 套裝軟體不會傳遞到非全域區域中。

- a. 在全域區域中執行 Java ES 安裝程式。
- b. 在元件選取頁面中選取 Application Server。

將會自動選取 Message Queue、HADB 和 Java DB，並將 Application Server 和 Message Queue 標記為可升級。

- c. 完成升級。

發行版本 5 Message Queue 會傳遞到稀疏根區域，但 Application Server、HADB 或 Java DB 不會傳遞。此外，所有共用元件會與發行版本 5 同步，並傳遞到稀疏根區域。

3. 在稀疏根區域中安裝 Application Server。

- a. 在稀疏根區域中執行 Java ES 安裝程式。
- b. 在元件選取頁面中選取 Application Server。

如果自動選取了 Message Queue，請取消選取它，如果沒有自動選取 HADB 和 Java DB，請選取它們。

- c. 完成 Application Server 的安裝。

選擇 [以後配置] 以使安裝程序不會覆寫網域資訊。

在沒有建立任何網域的情況下，您可以選擇 [立即配置]。

在 Solaris 10 多重區域環境中升級 Solaris 隨附的 Application Server

Service Registry

本章描述如何將 Service Registry 升級到 Java ES 5 (發行版本 5) : Service Registry 3.1。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 216 頁的「Service Registry 升級簡介」](#)
- [第 219 頁的「從 Java ES 發行版本 4 升級 Service Registry」](#)

備註 本章中的檔案位置是相對於名為 *ServiceRegistryR4-base* 和 *RegistryDomainR4-base* (Java ES 發行版本 4 Service Registry) 與 *ServiceRegistryR5-base* 和 *RegistryDomainR5-base* (Java ES 發行版本 5 Service Registry) 的目錄路徑指定的。安裝 Service Registry 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 12-1 Service Registry 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>ServiceRegistryR4-base</i>	/opt/SUNWsoar	/opt/sun/SUNWsoar
<i>RegistryDomainR4-base</i>	/var/opt/SUNWsoar	/var/opt/sun/SUNWsoar
<i>ServiceRegistryR5-base</i>	/opt/SUNWsrcv-registry	/opt/sun/srcv-registry
<i>RegistryDomainR5-base</i>	/var/opt/SUNWsrcv-registry	/var/opt/sun/srcv-registry

Service Registry 升級簡介

本節描述下列會影響 Service Registry 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5](#)
- [Java ES 發行版本 5 升級資訊指南](#)
- [Service Registry 資料](#)
- [Service Registry 升級策略](#)

關於 Java ES 發行版本 5

相對於發行版本 4 Service Registry 而言，Java ES 發行版本 5 Service Registry 為次要發行版本。它包含一些改進的功能、更新的介面以及所選錯誤修正程式。

Java ES 發行版本 5 升級資訊指南

[表 12-2](#) 顯示支援的 Service Registry 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 12-2 至 Java ES 5 (發行版本 5) 的升級路徑：Service Registry 3.1

Java ES 發行版本	Service Registry 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Service Registry 3.0 2005Q4	直接升級： 以發行版本 5 的全新安裝取代發行版本 4，並將登錄資料傳輸至發行版本 5。	無

Service Registry 資料

下表顯示可能受升級 Service Registry 軟體影響的資料類型。

表 12-3 Service Registry 資料用途

資料類型	位置	用途
安裝參數	<i>ServiceRegistryR4-base/install/install.properties</i>	Service Registry 的配置
可信任的憑證	<i>ServiceRegistryR4-base/install/cacerts</i>	Service Registry 信任的且不屬於 Application Server 安裝一部份的憑證
配置資料	<i>RegistryDomainR4-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes/*.properties</i>	配置 Service Registry 實例
登錄 / 儲存庫資料	<i>RegistryDomainR4-base/3.0/data</i>	資料庫和使用者憑證
Web 介面配置	<i>RegistryDomainR4-base/3.0/jaxr-ebxml</i>	Web 介面配置

Service Registry 升級策略

Service Registry 的升級策略一般取決於在第 1 章「升級規劃」中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Service Registry 升級規劃的問題，藉以對 Service Registry 進行一般性的介紹。

相容性問題

發行版本 5 Service Registry 向下相容發行版本 4 Service Registry。

相依性

Service Registry 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Service Registry 軟體的程序。例如，Service Registry 介面或功能上的變更可能需要 Service Registry 相依之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Service Registry 對下列 Java ES 元件有相依性：

- **共用元件。** Service Registry 對特定的 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。

- **Application Server**。 Service Registry 對 Application Server 有強制性的相依性，後者可為 Service Registry 應用程式提供容器，且在 Java ES 發行版本 5 中還可管理與網路登錄/儲存庫資料庫的連線。
- **Java DB**。 Service Registry 對 Java DB 有強制性的相依性，後者做為預設資料庫來儲存服務以及描述它們的中介資料。

雙重升級

在雙重升級中，Service Registry 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用下列兩種方式之一來執行：

- 全新安裝作業系統
- 原地升級作業系統

全新安裝作業系統

1. 備份現有的 Service Registry 資料。

如需必要資料的位置，請參閱第 217 頁的「[Service Registry 資料](#)」。

2. 安裝新的作業系統。

可將作業系統安裝在新系統 (或 Solaris 10 區域) 上，亦可讓其徹底覆蓋現有的檔案系統。

3. 復原[步驟 1](#)中所備份的 Service Registry 資料。
4. 安裝發行版本 5 Service Registry。

原地升級作業系統

1. 備份現有的 Service Registry 資料。

如需必要資料的位置，請參閱第 217 頁的「[Service Registry 資料](#)」。

2. 升級作業系統。

升級時會原樣保留現有的檔案系統。

3. 升級到發行版本 5 Service Registry。

請參閱第 219 頁的「[從 Java ES 發行版本 4 升級 Service Registry](#)」。

從 Java ES 發行版本 4 升級 Service Registry

本節包括將 Service Registry 從 Java ES 2005Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Service Registry](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Service Registry 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 若要達成升級，需要執行發行版本 5 Service Registry 的全新安裝，並將發行版本 4 資料和配置遷移到發行版本 5，然後移除發行版本 4 以保留磁碟空間。
- **升級相依性。** Service Registry 對許多 Java ES 共用元件具有相依性 (請參閱 [第 43 頁的表 1-9](#))，當您升級 Service Registry 時，Java ES 安裝程式會將所有這些共用元件自動升級到發行版本 5。

Service Registry 對 Application Server 和 Java DB 有硬性升級相依性。

- **向下相容性。** 發行版本 5 Service Registry 完全向下相容發行版本 4。
- **升級回復。** 發行版本 5 的升級回復藉由復原儲存的資料庫和配置資料，然後再復原至發行版本 4 來達成。
- **平台問題。** 升級 Service Registry 的一般方法在 Solaris 及 Linux 作業系統上均相同。

升級發行版本 4 Service Registry

本節描述如何在 Solaris 及 Linux 平台上將 Service Registry 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級發行版本 4 Service Registry](#)
- [驗證升級](#)

- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Service Registry 之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Service Registry 相依元件](#)
- [備份 Service Registry 資料](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可藉由注意 Web Console 使用者介面的特性來驗證 Service Registry 的目前版本：

```
http://localhost:6060/soar
```

同時，您還可檢查 Service Registry 套裝軟體名稱。例如：

在 Solaris 上：

```
pkginfo -l|grep srvc
```

在 Linux 上：

```
rpm -qa|grep srvc
```

下表顯示了不同的特性和套裝軟體名稱：

表 12-4 Service Registry 版本驗證輸出

Java ES 發行版本	Service Registry 版本編號	不同的特性
發行版本 4	3.0	Web Console：左窗格中的工具部份 套裝軟體名稱包含字串：soar
發行版本 5	3.1	Web Console：左窗格中的三個標籤 套裝軟體名稱包含字串：srvc-registry

升級 Service Registry 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。Service Registry 對一些共用元件、Application Server 和 Java DB 有硬性升級相依性。

升級 Service Registry 相依元件時，在升級 Service Registry 之前，應按照下列順序執行 (略過可能已升級過的任何相依元件)。共用元件的升級通常可由 Java ES 安裝程式自動完成。

1. **共用元件**。在第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Service Registry 升級到發行版本 5 時，Service Registry 所需要的所有共用元件會由 Java ES 安裝程式自動升級。
2. **Java DB**。在第 145 頁的第 8 章「Java DB」中提供了將 Java DB 升級到發行版本 5 的指示。
3. **Application Server**。在第 189 頁的第 11 章「Application Server」中提供了將 Application Server 升級到發行版本 5 的指示。

修改 HTTP 連接埠號

編輯檔案 `ServiceRegistryR4-base/install/install.properties`，將 HTTP 連接埠從 6060 變更為 6480 (6060 是保留的連接埠)。如需有關設定此特性的資訊，請參閱「Service Registry 3.1 管理指南」(<http://docs.sun.com/doc/820-0538>)。

備份 Service Registry 資料

將 Service Registry 從發行版本 4 升級到發行版本 5 不會修改配置資料或登錄 / 儲存庫資料庫。不需要備份目前的資料。

取得必要的配置資訊和密碼

您需要知道發行版本 4 Service Registry 的使用者 ID、密碼、網域名稱和連接埠號。

升級發行版本 4 Service Registry

本節描述 Solaris 和 Linux 平台上的升級程序。

升級程序 (Solaris)

下述程序適用於位於執行升級的電腦本機上的 Service Registry 實例。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 確定 Jakarta ANT Java/基於 XML 的建立工具 (ANT 共用元件) 參照了正確版本的 J2SE。

(在之後的步驟中使用了 ant 指令。)

```
PATH=/usr/jdk/entsys-j2se/bin:$PATH
```

```
export PATH
```

3. 停止發行版本 4 Service Registry (Application Server) 網域。

```
cd ServiceRegistryR4-base/install
/usr/sfw/bin/ant -f build-install.xml appserver.domain.stop
```

網域與 Service Registry 實例相關聯。

4. 執行發行版本 5 Service Registry 的全新安裝。

執行下列步驟：

- a. 在代管發行版本 4 Service Registry 的電腦上啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 從元件選取頁面中選取 Service Registry。

- c. 指定與發行版本 4 不同的安裝目錄路徑。

依預設，發行版本 5 的安裝路徑 (*ServiceRegistryR5-base*) 與發行版本 4 的安裝路徑 (*ServiceRegistryR4-base*) 不同。

- d. 選取 [以後配置] 選項。

不支援 [立即配置]。

- e. 如有必要，請選取安裝本土化套裝軟體的選項。

- f. 安裝完成時結束 Java ES 安裝程式。

5. 升級和配置發行版本 5 Service Registry 實例。

```
cd ServiceRegistryR5-base/install
/usr/sfw/bin/ant -f build-install.xml
-Dinstall.properties=ServiceRegistryR4-base/install/install.properties
upgrade
```

做為指向發行版本 4 `install.properties` 檔案的另一種選擇方法，您可以修改預設發行版本 5 `install.properties` 檔案以重現任何發行版本 4 特性值。如需有關設定這些特性的資訊，請參閱「Service Registry 3.1 管理指南」(<http://docs.sun.com/doc/820-0538>)。

如果您要使用自訂特性值，而不是將它們放在 `install.properties` 中，那麼，您需要在 `Ant` 指令行上指定這些特性值 (全部在一行中)，如下所示：

```
/usr/sfw/bin/ant -f build-install.xml
-Dregistry.install.RegistryServerKeystorePassword=passwd1
-Dregistry.install.AdministratorPassword=passwd2
-Dregistry.install.ApplicationServerKeystorePassword=passwd3
upgrade
```

不過，建議將這些自訂特性值包含在 `install.properties` 檔案中，並使用有限的權限以避免使用可由未經授權人員檢視的指令行設定。如需更多資訊，請參閱「[Service Registry 管理指南](#)」。

升級公用程式會建立新 `Application Server` 網域、啟動網域以及在網域中部署 `Service Registry` 實例。每個 `Service Registry` 實例皆與其自身的 `Application Server` 網域相關聯。

6. 如果已經修改了發行版本 4 `Service Registry` 的伺服器特性檔案，則您可以按以下方式對發行版本 5 `Service Registry` 配置進行對應的變更：

- a. 停止發行版本 5 `Service Registry (Application Server)` 網域。

(網域由步驟 5 中的 `upgrade` 指令自動啟動。)

```
cd ServiceRegistryR5-base/install
/usr/sfw/bin/ant -f build-install.xml appserver.domain.stop
```

- b. 將發行版本 4 `Service Registry` 實例配置傳輸到發行版本 5。

將所做的任何修改增加至發行版本 4 `Service Registry` 實例配置：

```
RegistryDomainR4-base/domains/registry/applications/j2ee-modules/
soar/WEB-INF/classes/*.properties
```

將所做的任何修改增加至對應的發行版本 5 配置：

```
RegistryDomainR5-base/domains/registry/applications/j2ee-modules/
soar/WEB-INF/classes/*.properties
```

7. 啟動發行版本 5 `Service Registry (Application Server)` 網域。

```
cd ServiceRegistryR5-base/install
/usr/sfw/bin/ant -f build-install.xml appserver.domain.start
```

升級程序 (Linux)

除了 Linux 平台上 `ant` 指令的位置 (在升級程序的各個步驟中使用) 與 Solaris 平台上的位置不同以外，在 Linux 上與在 Solaris 上升級 `Service Registry` 完全相同 (請參閱第 221 頁的「[升級程序 \(Solaris\)](#)」)：

```
/opt/sun/share/bin/ant
```

驗證升級

您可藉由注意 Web Console 使用者界面的特性來驗證 Service Registry 是否已成功升級：

```
http://localhost:6480/soar
```

同時，您還可檢查 Service Registry 套裝軟體名稱。例如：

在 Solaris 上：

```
pkginfo -l|grep soar
```

在 Linux 上：

```
rpm -qa|grep soar
```

第 220 頁的表 12-4 顯示了不同的特性和套裝軟體名稱。

升級後作業

只有您確定您不想回復至發行版本 4 的升級時，才可執行以下有關如何移除發行版本 4 Service Registry 的步驟。

1. 刪除發行版本 4 Service Registry (Application Server) 網域：

```
cd ServiceRegistryR4-base/install
```

在 Solaris 上：

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.delete
```

在 Linux 上：

```
/opt/sun/bin/ant -f build-install.xml appserver.domain.delete
```

2. 刪除包含發行版本 4 Service Registry 網域檔案的目錄。

```
rm -rf RegistryDomainR4-base
```

3. 刪除包含發行版本 4 Service Registry 安裝檔案的目錄。

```
rm -rf ServiceRegistryR4-base
```

回復升級

發行版本 5 升級的回復是藉由復原為先前版本來達成的，先前版本在升級到發行版本 5 時保持不變。

1. 停止和刪除發行版本 5 Service Registry (Application Server) 網域：

```
cd ServiceRegistryR4-base/install
```

在 Solaris 上：

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.delete
```

在 Linux 上：

```
/opt/sun/bin/ant -f build-install.xml appserver.domain.delete
```

2. 執行 Java ES 發行版本 5 解除安裝程式，以解除安裝發行版本 5 Service Registry。
3. 啟動發行版本 4 Service Registry 網域。

```
cd ServiceRegistryR4-base/install
```

在 Solaris 上：

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.start
```

在 Linux 上：

```
/opt/sun/bin/ant -f build-install.xml appserver.domain.start
```

4. 存取發行版本 4 Service Registry Web Console。

```
http://localhost:6480/soar
```

5. 確認主控台顯示了第 220 頁的「[Service Registry 版本驗證輸出](#)」中所介紹的發行版本 4 特性。

多重實例升級

在某些部署架構中，會將 Service Registry 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Service Registry 實例在多台電腦上執行，並透過負載平衡器來分散負載。

在這些架構中，登錄主要是唯讀的，並可藉由存取一般資料庫來回應大量的查詢。

如第 219 頁的「[升級發行版本 4 Service Registry](#)」中所述，您要在每台電腦上執行 Service Registry 的升級。

Web Proxy Server

本章描述如何將 Web Proxy Server 升級到 Java ES 5 (發行版本 5) : Sun Java System Web Proxy Server 4.0.4。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 228 頁的「Web Proxy Server 升級簡介」](#)
- [第 231 頁的「從 Java ES 發行版本 4 升級 Web Proxy Server」](#)
- [第 237 頁的「從版本 3.6 升級 Web Proxy Server」](#)

備註 本章中的檔案位置是以稱為 *WebProxyServer-base* 的目錄路徑為基準指定的。初始安裝 Web Proxy Server 時可能至少已將此路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

WebProxyServer-base 的值依作業系統平台而定，如下表所示。

表 13-1 Web Proxy Server 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>WebProxyServer-base</i>	/opt/SUNWproxy	/opt/sun/webproxyserver

Web Proxy Server 升級簡介

本節描述下列會影響 Web Proxy Server 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Web Proxy Server](#)
- [Web Proxy Server 升級資訊指南](#)
- [Web Proxy Server 資料](#)
- [Web Proxy Server 升級策略](#)

關於 Java ES 發行版本 5 Web Proxy Server

相對於發行版本 4 而言，Java ES 發行版本 5 Web Proxy Server 為次要錯誤修正發行版本。

不過，與納入 Java Enterprise System 之前的 Sun One Web Proxy Server 3.6 相比，發行版本 5 Web Proxy Server 效能更佳、架構延伸性更大、標準遵循性更好，並採用了新的管理介面。

Web Proxy Server 升級資訊指南

表 13-2 顯示 Web Proxy Server 至 Java ES 發行版本 5 的升級路徑。先前的 Java ES 發行版本中不包括 Web Proxy Server。該表格僅適用於 Solaris OS，因為之前在 Linux OS 上並不支援 Web Proxy Server。

表 13-2 至 Java ES 5 (發行版本 5) 的升級路徑：Web Proxy Server 4.0.4

Java ES 發行版本	Web Proxy Server 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Web Proxy Server 4.0.1 2005Q4	直接升級： 使用修補程式執行	無
Java ES 發行版本之前的發行版本 (僅限於 Solaris OS)	Sun ONE Web Proxy Server 3.6 (後文稱為版本 3.6)	直接升級： 藉由使用 Java ES 安裝程式在新位置執行安裝，然後使用管理工具遷移配置資料來執行	必須將配置資訊遷移到新位置。

Web Proxy Server 資料

下表顯示可能受升級 Web Proxy Server 軟體影響的資料類型。

表 13-3 Web Proxy Server 資料用途

資料類型	位置	用途
配置資料	<p><i>WebProxyServer-base/proxy-serverid/</i> config 目錄</p> <p>包含檔案，如：<i>server.xml</i>、<i>magnus.conf</i>、<i>obj.conf</i> 等</p>	儲存伺服器、快取、篩選器、路由及 Web Proxy Server 其他功能層面的配置資訊

Web Proxy Server 升級策略

Web Proxy Server 的升級策略一般取決於在 [第 1 章「升級規劃」](#) 中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Web Proxy Server 升級規劃的問題，藉以對 Web Proxy Server 進行一般性的介紹。

相容性問題

發行版本 5 Web Proxy Server 未推出任何新的公共介面，它向下相容發行版本 4 Web Proxy Server。發行版本 5 Web Proxy Server 還與版本 3.6 相容，但使用版本 3.6 支援的 NSAPI 介面開發的外掛程式除外，必須使用發行版本 5 支援的 NSAPI 介面將它們重新編譯。

Web Proxy Server 相依性

Web Proxy Server 對下列 Java ES 元件有相依性：

- **共用元件**。 Web Proxy Server 對特定的 Java ES 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))。
- **Directory Server**。 Web Proxy Server 對 Directory Server 有選擇性的相依性，後者用於提供基於 LDAP 的認證。
- **Web Server**。 Web Proxy Server 與 Web Server 互有相依性，後者用於提高 HTTP 請求的安全性與效能。

雙重升級

在雙重升級中，Web Proxy Server 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用原地升級作業系統的方法來執行：

1. 備份現有的 Web Proxy Server 資料。
如需必要資料的位置，請參閱第 229 頁的「[Web Proxy Server 資料](#)」。
2. 升級作業系統。
升級時會原樣保留現有的檔案系統。
3. 升級到發行版本 5 Web Proxy Server。
請根據升級路徑，參閱本章的相應小節。

從 Java ES 發行版本 4 升級 Web Proxy Server

本節包括將 Web Proxy Server 從 Java ES 2005Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Web Proxy Server](#)

簡介

將 Web Proxy Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 使用修補程式來執行升級。不需要其他重新配置。
- **升級相依性。** Web Proxy Server 對許多 Java ES 共用元件有相依性 (請參閱 [第 43 頁的表 1-9](#))，但是，Web Proxy Server 只對 NSS 和 NSPR 共用元件有硬性升級相依性。
- **向下相容性。** 發行版本 5 Web Proxy Server 向下相容發行版本 4。
- **升級回復。** Web Proxy Server 之發行版本 5 的升級回復是藉由移除升級修補程式來達成的。
- **平台問題。** 升級 Web Proxy Server 的一般方法在 Solaris 及 Linux 作業系統上相同，但修補技術不同。因此，升級過程包括平台特定的程序。

升級發行版本 4 Web Proxy Server

本節提供如何將 Web Proxy Server 升級到 Java ES 發行版本 5 的簡介。本節涵蓋下列主題：

- [升級前作業](#)
- [升級發行版本 4 Web Proxy Server \(Solaris\)](#)
- [升級發行版本 4 Web Proxy Server \(Linux\)](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級 \(Solaris\)](#)

升級前作業

升級 Web Proxy Server 之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Web Proxy Server 相依元件](#)
- [備份 Web Proxy Server 資料](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可輸入下列指令來驗證 Web Proxy Server 的目前版本：

```
WebProxyServer-base/proxy-admserv/start -version
```

表 13-4 Web Proxy Server 版本驗證輸出

Java ES 發行版本	Web Proxy Server 版本編號
非 Java ES 發行版本 3.6 版	3.6
發行版本 4	4.0.1
發行版本 5	4.0.4

升級 Web Proxy Server 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。但是，將 Web Proxy Server 升級到發行版本 5 只需要升級 NSS 和 NSPR 共用元件。如果尚未升級這些共用元件，您應該使用 [同步化共用元件] 選項將所有共用元件都與其發行版本 5 版本同步。[第 59 頁的第 2 章「升級 Java ES 共用元件」](#)中提供了指示。

備份 Web Proxy Server 資料

將 Web Proxy Server 升級到發行版本 5 不會修改發行版本 4 配置資料。不需要備份目前的資料。

取得必要的配置資訊和密碼

不需要關於目前安裝版本的特殊資訊。不過，您需要以超級使用者的身份登入才能執行升級。

升級發行版本 4 Web Proxy Server (Solaris)

本節討論影響 Web Proxy Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項 (Solaris)

將 Web Proxy Server 軟體升級到 Java ES 發行版本 5 時要考慮下列注意事項：

- 同時升級對應於同一安裝的 Web Proxy Server 影像的所有 Web Proxy Server 實例。在對安裝的影像套用修補程式時，應關閉所有此類實例。
- 下表顯示適用於 Solaris OS 的發行版本 5 Web Proxy Server 升級修補程式：

表 13-5 用於在 Solaris 上升級 Web Proxy Server 的修補程式¹

說明	修補程式 ID：SPARC Solaris 9 和 10	修補程式 ID：X86 Solaris 9 和 10
Web Proxy Server 核心	120981-10	120982-10
Web Proxy Server 本土化版本	122963-01	122964-01

1. 修補程式的修訂版號碼是升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

升級程序 (Solaris)

下述程序適用於位於執行升級的電腦上的 Web Proxy Server。

1. 請以超級使用者身份登入或成為超級使用者。
su -
2. 停止所有執行中的 Web Proxy Server 實例和 Administration Server。
WebProxyServer-base/proxy-instanceName/stop
WebProxyServer-base/proxy-admserv/stop
3. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。
[第 59 頁的第 2 章「升級 Java ES 共用元件」](#)中提供了指示。
4. 根據表 13-5 取得所需的修補程式。

可從下列位置將修補程式下載到 /tmp：

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

5. 請套用適當的 Web Proxy Server 核心，如有必要，也請以該順序套用表 13-5 中的本土化修補程式。

```
patchadd /tmp/patch_ID
```

6. 確認修補程式升級成功：

```
showrev -p | grep proxy
```

輸出應傳回步驟 5 中套用的修補程式 ID 版本。

7. 重新啟動在步驟 2 中停止的 Web Proxy Server 實例。

```
WebProxyServer-base/proxy-instanceName/start
```

升級發行版本 4 Web Proxy Server (Linux)

本節討論影響 Web Proxy Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項 (Linux)

除了 Linux 發行版本 5 升級修補程式與 Solaris 修補程式不同以外，其餘在 Linux 平台上將 Web Proxy Server 軟體升級到 Java ES 發行版本 5 所要考慮的注意事項與在 Solaris 平台上升級時要考慮的注意事項相同 (請參閱第 239 頁的「升級注意事項」)。

下表顯示適用於 Linux OS 的發行版本 5 Web Proxy Server 升級修補程式：

表 13-6 用於在 Linux 上升級 Web Proxy Server 的修補程式¹

說明	修補程式 ID 和 RPM 名稱
Web Proxy Server 核心	120983-10 <ul style="list-style-type: none"> • sun-proxyserver-4.0-6.5.i386.rpm
Web Proxy Server 本土化版本	122965-01 <ul style="list-style-type: none"> • sun-proxyserver-Locale-4.0.4.i386.rpm

¹ 修補程式的修訂版號碼是升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

升級程序 (Linux)

下述程序適用於位於執行升級的電腦上的 Web Proxy Server。

注意 在 Linux 上執行的從 Java ES 發行版本 4 到 Java ES 發行版本 5 的升級無法回復。

1. 請以超級使用者身份登入或成爲超級使用者。
`su -`
2. 停止所有執行中的 Web Proxy Server 實例和 Administration Server。
`WebProxyServer-base/proxy-instanceName/stop`
`WebProxyServer-base/proxy-admserv/stop`
3. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。
 請參閱第 232 頁的「升級 Web Proxy Server 相依元件」。
4. 使用表 13-6 中的修補程式號碼和 RPM 名稱取得所需的修補程式。使用此資訊以取得 RPM 的版本號碼。
 可從下列位置將修補程式下載到 /tmp：
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>
5. 以表 13-6 中的順序套用該表中 Web Proxy Server 的核心 RPM 以及本土化 RPM (如果需要)。
`rpm -Fvh sun-proxyserver-version.i386.rpm`
6. 確認修補程式升級成功：
`rpm -qa | grep sun-proxyserver`
 應傳回 RPM 的新版本編號。
7. 重新啓動在步驟 2 中停止的 Web Proxy Server 實例。
`WebProxyServer-base/proxy-instanceName/start`

驗證升級

您可藉由使用 `-version` 選項啓動 Web Proxy Server 實例來驗證 Web Proxy Server 到發行版本 5 的升級：

```
WebProxyServer-base/proxy-admserv/start -version
```

請參閱第 232 頁的表 13-4 中的輸出值。

升級後作業

除了在第 233 頁的「升級程序 (Solaris)」和第 234 頁的「升級程序 (Linux)」中所述的步驟之外，沒有升級後的作業。

回復升級 (Solaris)

本節描述影響 Web Proxy Server 升級回復程序的注意事項，接著提供程序本身。

回復注意事項 (Solaris)

發行版本 5 Web Proxy Server 的升級回復程序基本上就是升級到發行版本 5 程序的逆過程。

回復程序 (Solaris)

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 停止所有執行中的 Web Proxy Server 實例和 Administration Server。

```
WebProxyServer-base/proxy-instanceName/stop
```

```
WebProxyServer-base/proxy-admserv/stop
```

3. 移除第 233 頁的表 13-5 中的修補程式。

```
patchrm patch_ID
```

4. 重新啟動在步驟 2 中停止的 Web Proxy Server 實例。

```
WebProxyServer-base/proxy-instanceName/start
```

從版本 3.6 升級 Web Proxy Server

本節包括將 Web Proxy Server 從版本 3.6 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [版本 3.6 Web Proxy Server 升級](#)

簡介

將 Web Proxy Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 藉由使用 Java ES 安裝程式在不同於版本 3.6 所在目錄的目錄中安裝發行版本 5 Web Proxy Server 來執行升級。然後使用 Web Proxy Server 管理伺服器將配置設定 (但不包括快取內容) 從版本 3.6 遷移到發行版本 5。
- **升級相依性。** 將 Web Proxy Server 升級到發行版本 5 時，Java ES 安裝程式會自動執行共用元件的安裝。
- **向下相容性。** 發行版本 5 Web Proxy Server 向下相容版本 3.6，但使用版本 3.6 支援的 NSAPI 介面開發的外掛程式除外，必須使用發行版本 5 支援的 NSAPI 介面將它們重新編譯。
- **升級回復。** Web Proxy Server 的發行版本 5 升級回復是藉由復原至版本 3.6 (升級時將其保持不變) 來達成的。
- **平台問題。** 升級 Web Proxy Server 的方法在所有 Solaris 平台上均相同，但 Linux 平台不支援版本 3.6。

版本 3.6 Web Proxy Server 升級

本節提供如何將 Web Proxy Server 升級到 Java ES 發行版本 5 的簡介。以前 Linux 平台並不支援 Web Proxy Server。因此，Web Proxy Server 至 Java ES 發行版本 5 的升級僅可在 Solaris 平台上執行。本節涵蓋以下主題：

- [升級前作業](#)
- [升級版本 3.6 Web Proxy Server](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復版本 3.6 升級](#)

升級前作業

升級 Web Proxy Server 之前，應該執行下列作業：

- 驗證目前版本資訊
- 升級 Web Proxy Server 相依元件
- 備份 Web Proxy Server 資料
- 取得必要的配置資訊和密碼

驗證目前版本資訊

您可輸入下列指令來驗證 Web Proxy Server 的目前版本：

```
WebProxyServer-base/proxy-serverid/start -version
```

表 13-7 Web Proxy Server 版本驗證輸出

Java ES 發行版本	Web Proxy Server 版本編號
非 Java ES 發行版本 3.6 版	3.6
發行版本 4	4.0.1
發行版本 5	4.0.4

升級 Web Proxy Server 相依元件

一般的建議是將電腦系統 (及運算環境中) 的所有 Java ES 元件都升級到 Java ES 發行版本 5。不過，用於將 Web Proxy Server 升級到發行版本 5 的 Java ES 安裝程式會自動升級 Web Proxy Server 依賴的所有共用元件 (請參閱第 43 頁的表 1-9)。

備份 Web Proxy Server 資料

將 Web Proxy Server 升級到發行版本 5 不會修改 3.6 版的配置資料。不過，執行升級前必須儲存使用管理介面對 3.6 版配置資料所做的未儲存變更。

取得必要的配置資訊和密碼

若要從 3.6 版升級，則需要知道該安裝版本的安裝目錄路徑。

升級版本 3.6 Web Proxy Server

本節討論影響 Web Proxy Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項

可升級對應於同一安裝的 Web Proxy Server 影像的所有 Web Proxy Server 實例。不過，每個實例的配置資料遷移皆須分開執行。執行遷移時，應關閉所有此類實例，以確定啟動遷移的實例時不會發生連接埠衝突。

升級程序

下述程序適用於位於執行升級的電腦上的 Web Proxy Server 軟體。

1. 請以超級使用者身份登入或成為超級使用者。
 - su -
2. 安裝 Web Proxy Server 發行版本 5。
 - a. 從 Java ES 發行版本 5 發行物中執行 Java ES 安裝程式。
 - b. 從元件選取頁面中選取 Web Proxy Server。
 - c. 選取 [立即配置] 選項。
 - d. 安裝完成時退出 Java ES 安裝程式。

3. 將配置設定遷移到新安裝的版本。

必須為每個 Web Proxy Server 實例單獨執行此作業。

- a. 啟動 Web Proxy Server Administration Server。
 - `WebProxyServer-base/proxy-admserv/start`
- b. 登入圖形化管理介面。
- c. 按一下 [Server] 標籤，然後按一下 [Migrate Server]。
- d. 指定 3.6 版的安裝目錄路徑。
- e. 選取要遷移的實例。
- f. 按一下 [Migrate] 按鈕。

遷移成功後，遷移螢幕會顯示必須手動執行的額外配置的清單。它提供需要增加的資料和對應的配置檔案。

如需關於遷移配置設定的更多資訊，請參閱「Sun Java System Web Proxy Server 4.0.4 Installation and Migration Guide」
(<http://docs.sun.com/doc/819-5492>)。

4. 進行步驟 f 中指定的任何額外配置變更。

如需更多資訊，請參閱「Sun Java System Web Proxy Server 4.0.4 Configuration File Reference」(<http://docs.sun.com/doc/819-5494>)。

驗證升級

您可藉由使用 `-version` 選項啓動 Web Proxy Server 實例來驗證 Web Proxy Server 到發行版本 5 的升級：

```
WebProxyServer-base/proxy-serverid/start -version
```

請參閱第 238 頁的表 13-7 中的輸出值。

升級後作業

除了在第 239 頁的「升級程序」中所述的步驟之外，沒有升級後作業。

回復版本 3.6 升級

無法回復 Web Proxy Server 到發行版本 5 的升級 (第 238 頁的「升級版本 3.6 Web Proxy Server」中所述)。不過，可以復原到版本 3.6，發行版本 5 升級程序未對其做任何變更。

Access Manager

本章描述如何將 Access Manager 軟體從先前的 Java ES 版本升級到 Java ES 5 (發行版本 5) : Sun Java System Access Manager 7.1。

本章提供發行版本 5 所支援不同升級路徑之 Access Manager 升級問題和程序的一般簡介。本章涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 242 頁的「Access Manager 升級簡介」](#)
- [第 248 頁的「從 Java ES 發行版本 4 升級 Access Manager」](#)
- [第 268 頁的「從 Java ES 發行版本 3 升級 Access Manager」](#)
- [第 271 頁的「從 Java ES 發行版本 2 升級 Access Manager」](#)

備註 本章中的檔案位置是以兩個稱為 *AccessManager-base* 和 *AccessManagerConfig-base* 的目錄路徑為基準指定的。初始安裝 Access Manager 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 14-1 Access Manager 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>AccessManager-base</i>	/opt/SUNWam	/opt/sun/identity
<i>AccessManagerConfig-base</i>	/etc/opt/SUNWam	/etc/opt/sun/identity

Access Manager 升級簡介

本節描述下列會影響 Access Manager 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Access Manager](#)
- [Access Manager 升級資訊指南](#)
- [Access Manager 資料](#)
- [Access Manager 升級策略](#)

備註 在 Java ES 發行版本 3 之前，Access Manager 版本的名稱是 Identity Server。因此，本章中提到的 Identity Server 是指 Java ES Access Manager 元件的早期版本。

關於 Java ES 發行版本 5 Access Manager

Java ES 發行版本 5 Access Manager 代表次要發行版本。它包含對於 Java ES 發行版本 4 Access Manager 的一些錯誤修正與功能增強，發行版本 4 是一個重大發行版本。在發行版本 5 的增強功能中，有一種以 Java ES 監視架構為基礎的新增監視功能。如需發行版本 5 增強功能的更多資訊，請參閱「Sun Java System Access Manager 7.1 版本說明」(<http://docs.sun.com/doc/820-0366>)。

與發行版本 4 類似，發行版本 5 Access Manager 支援多個身份儲存庫或使用資料存放區。因此 Access Manager 發行版本 5 不僅支援 LDAP 目錄 (如 Directory Server)，亦支援其他資料儲存協定和格式。

在前端，使用 Access Manager 主控台來配置新的 Access Manager 服務和身份儲存庫。

為提供與其他 Java ES 元件的向下相容性，發行版本 5 可以在舊有模式下執行，該模式支援依賴發行版本 3 Access Manager 服務的 Java ES 元件 (如需更多資訊，請參閱第 245 頁的「[相容性問題](#)」)。

Access Manager 升級資訊指南

表 14-2 顯示支援的 Access Manager 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 14-2 至 Java ES 5 (發行版本 5) 的升級路徑：Access Manager 7.1

Java ES 發行版本	Access Manager 版本	一般方法	需要重新配置
發行版本 4	Sun Java System Access Manager 7.0 2005Q4	直接升級： 執行時，首先使用預升級程序檔來移除發行版本 4，然後完整安裝並重新配置發行版本 5。	配置資料 用於 Access Manager 主控台和認證 UI 的自訂 JSP 目錄模式
發行版本 3	Sun Java System Access Manager 6.3 2005Q1	直接升級： 執行時，首先使用預升級程序檔來移除發行版本 3，然後完整安裝並重新配置發行版本 5。	配置資料 用於 Access Manager 主控台和認證 UI 的自訂 JSP 目錄模式
發行版本 2	Sun Java System Identity Server 6.2 2004Q2 以及 6.2 SP1	直接升級： 執行時，首先使用預升級程序檔來移除發行版本 2，然後完整安裝並重新配置發行版本 5。	配置資料 用於 Access Manager 主控台和認證 UI 的自訂 JSP 目錄模式
發行版本 1	Sun ONE Identity Server 6.1	無直接升級： 但可以先使用「Java Enterprise System 2005Q1 升級與遷移指南」(http://docs.sun.com/doc/819-2238) 中說明的程序升級到發行版本 3。 然後從發行版本 3 升級到發行版本 5。	配置資料 用於 Access Manager 主控台和認證 UI 的自訂 JSP 目錄模式
Java ES 發行版本之前的發行版本	Sun ONE Identity Server 6.0 或 6.0 SP 1 或 iPlanet Directory Server Access Management Edition (DSAME) 5.1	無直接升級。	

Access Manager 資料

Access Manager 與其他 Java ES 元件類似，會利用各種資料，對於任何特定升級，可能都需要將這些資料遷移至升級的版本。下表顯示可能受 Access Manager 軟體升級影響的資料類型。

表 14-3 Access Manager 資料用途

資料類型	位置	用途
配置資料	<i>AccessManagerConfig-base/config/AMConfig.properties</i> <i>AccessManagerConfig-base/config/serverconfig.xml</i> 認證與自訂模組的 JAR 檔案 <i>AccessManager-base/lib</i>	配置 Access Manager 及其與後端資料存放區的整合。
Web 容器存取控制和配置檔案	Web Server 7.0 (Java ES 發行版本 5) server.policy 和 server.xml 檔案，位於 <i>WebServer7Config-base/https-configName/config</i> 中 Web Server 6.x (Java ES 發行版本 2、3 和 4) server.policy 和 server.xml 檔案，位於 <i>WebServer6-base/https-hostname/config</i> 中 Application Server 8.x (Java ES 發行版本 3、4 和 5) : server.policy 和 domain.xml 檔案，位於 <i>AppServer8Config-base/domains/domainName/config</i> 中 Application Server 7.x (Java ES 發行版本 2) : server.policy 和 server.xml 檔案，位於 <i>AppServer7Config-base/domains/domainName/config</i> 中 WebSphere 和 WebLogic : 將 Access Manager 配置為與這些 Web 容器配合使用時，將修改它們各自的策略和配置檔案。	配置 Access Manager Web 容器實例。
自訂資料 (Web 容器自訂 JSP 檔案)	管理主控台：(Java ES 發行版本 2 和 3) : <i>AccessManager-base/web-src/applications</i> 管理主控台：(Java ES 發行版本 4 和 5) : <i>AccessManager-base/web-src/services</i> 認證 UI : <i>AccessManager-base/web-src/services</i>	配置 Access Manager 管理介面。
目錄模式 服務配置 使用者資料	Directory Server	Access Manager 根據儲存在目錄中的服務配置、使用者及策略資料為一般使用者提供認證和授權服務。
動態應用程式資料	無	Access Manager 不會永久性地儲存應用程式資料 (如階段作業狀態)。

Access Manager 升級策略

Access Manager 的升級策略一般取決於在第 1 章「升級規劃」中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Access Manager 升級規劃的問題，藉以對 Access Manager 進行一般性的介紹。

相容性問題

發行版本 5 Access Manager 向下相容發行版本 4 Access Manager，但發行版本 4 Access Manager 是一個重大發行版本，除了配置為在舊有模式下執行時，其他情況下均不與更早版本相容。同樣地，除非配置為在舊有模式下執行，否則發行版本 5 Access Manager 並不向下相容發行版本 3 Access Manager (或在舊有模式下執行的發行版本 4 Access Manager)。

此外，發行版本 5 Access Manager 不向下相容任何模式下的發行版本 2 Access Manager；發行版本 5 Access Manager 無法與發行版本 2 Access Manager SDK 互通，反之亦然。

發行版本 5 Access Manager 配置為在較新的範圍模式下執行時，會支援多個身份儲存庫和資料儲存協定。必須將目錄資料遷移到新結構中才能支援範圍模式作業。此外，範圍模式並不支援其他 Java ES 元件 (例如 Portal Server) 或 Sun Java Communications Suite 元件 (例如 Communications Express、Messaging Server 和其他元件)。

但當發行版本 5 Access Manager 配置為在舊有模式下執行時，會向下相容發行版本 3 Access Manager 和對應的目錄資料，只有很少一些例外情況 (請參閱「Sun Java System Access Manager 7.1 版本說明」<http://docs.sun.com/doc/820-0366>)。

其他 Java ES 元件及舊版本的 Access Manager 策略代理程式無法在範圍模式下與 Access Manager 互通，支援它們需要使用舊有模式。這種不相容性是一個重要的升級考量，並意味著在大部份 Java ES 部署中，應將 Access Manager 升級到發行版本 5 舊有模式。

但是，即使配置為在舊有模式下執行，發行版本 5 Access Manager 也不與發行版本 3 或更早的 Sun Java Communications Suite 元件相容。如果 Access Manager 升級到發行版本 5，則發行版本 3 或更早的 Delegated Administrator 也必須升級到發行版本 5，才能針對 Messaging Server 和 Calendar Server 佈建使用者。不過，並非一定要將 Messaging Server 和 Calendar Server 本身也升級到發行版本 5。

與發行版本 4 主控台一樣，發行版本 5 Access Manager 主控台也支援範圍和舊有這兩種模式。但是，如果您已將 Access Manager 配置為在舊有模式下執行，則仍可使用在發行版本 2 和發行版本 3 中發行的僅舊有主控台。

Access Manager 相依性

Access Manager 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Access Manager 軟體的程序。例如，Access Manager 介面或功能上的變更可能需要 Access Manager 依賴之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Access Manager 對下列 Java ES 元件有相依性：

- **共用元件。** Access Manager 對特定 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。Access Manager 升級可能依賴於這些共用元件的升級版本。
- **Web 容器。** Access Manager 對 Web 容器服務具有強制性的相依性，這些服務可由 Java ES Web Server、Java ES Application Server 或協力廠商 Web 容器 (來自 Weblogic 和 WebSphere) 提供。升級 Access Manager 可能需要將用於 Access Manager 主控台或認證 UI 的自訂 JSP 遷移到升級後的 Access Manager 環境。
- **Directory Server。** Access Manager 對 Directory Server 有強制性的相依性，後者用於儲存配置資料和使用資料。因此，Access Manager 升級可能需要延伸目錄模式。

Web 容器升級方案

Access Manager 可部署於 Web Server 或 Application Server 提供的 Web 容器中。這樣，其所在的 Web 容器可能也需要升級到發行版本 5，使得 Access Manager 的升級更加複雜。就這一點而言，有一些可能的 Web 容器升級方案，列舉於下表中。

表 14-4 Access Manager 升級的 Web 容器升級方案

方案	最初部署 Access Manager 的 Web 容器	升級之後部署 Access Manager 的 Web 容器	適用 Access Manager 升級路徑：升級始於
方案 1	Web Server 6.x	Web Server 6.x	發行版本 2 發行版本 3 發行版本 4
方案 2	Web Server 6.x	Web Server 7.0	發行版本 2 發行版本 3 發行版本 4
方案 3	Application Server 8.1	Application Server 8.1	發行版本 3 發行版本 4

表 14-4 Access Manager 升級的 Web 容器升級方案

方案	最初部署 Access Manager 的 Web 容器	升級之後部署 Access Manager 的 Web 容器	適用 Access Manager 升級路徑：升級始於
方案 4	Application Server 8.1	Application Server 8.2	發行版本 3 發行版本 4
方案 5	Application Server 7x	Application Server 8.2	發行版本 2

升級 Access Manager (例如使用 amconfig 程序檔時) 來為表 14-4 中的合適升級方案提供適當值時，特別是要進行 Web 容器的重大版本升級時，必須很小心。

雙重升級

在雙重升級中，Access Manager 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，而這種升級方法對於 Access Manager 不受支援。

因此，如果您遇到需要進行雙重升級的情況，必須先執行作業系統安裝或升級，然後再重新安裝並全新配置 Access Manager。

從 Java ES 發行版本 4 升級 Access Manager

本節包括將 Access Manager 從 Java ES 2005Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。本節涵蓋以下主題：

- [簡介](#)
- [完整升級發行版本 4 Access Manager](#)
- [多重實例升級](#)
- [僅升級發行版本 4 Access Manager SDK](#)

簡介

將 Java ES 發行版本 4 Access Manager 升級至發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 執行升級時，首先移除先前的二進位檔案版本，然後全新安裝發行版本 5。提供了一個 `ampre71upgrade` 程序檔來移除發行版本 4 版本，然後使用 Java ES 安裝程式來安裝發行版本 5。接著使用 `amconfig` 程序檔來重新配置 Access Manager，並使用 `amupgrade` 程序檔來遷移目錄模式。
- **升級相依性。** Access Manager 對許多 Java ES 共用元件具有相依性 (請參閱 [第 43 頁的表 1-9](#))，當您升級 Access Manager 時，Java ES 安裝程式會將所有這些共用元件自動升級到發行版本 5。這包括對於新 Java ES 監視架構的支援，這一架構需要發行版本 4 Access Manager 所不需要的一些共用元件。

此外，發行版本 5 Access Manager 還依賴 Directory Server 和 Web Server (或 Application Server 或協力廠商 Web 容器)，如 [第 246 頁的「Access Manager 相依性」](#) 中所述。不過，這些並非硬性的升級相依性；對將 Access Manager 升級至發行版本 5 而言，升級這些元件是選擇性的。

- **向下相容性。** 發行版本 5 Access Manager 與發行版本 4 相容，但與更早的 Access Manager 發行版本不相容 (請參閱 [第 245 頁的「相容性問題」](#))。
- **升級回復。** 沒有公用程式可以用來回復 Access Manager 升級。事實上，由於將 Access Manager 回復至其原始狀態所需的重新配置次數過多，此類回復並不可行。最佳的回復方式是使用備份的配置檔案建立平行安裝，並於執行升級之前對此平行安裝進行測試。這樣，您就可以在必要的情況下復原為該平行安裝。
- **平台問題。** 升級 Access Manager 的一般方法在 Solaris 及 Linux 作業系統上均相同。以下程序指出平台特定的指令或檔案位置 (如適用)。

完整升級發行版本 4 Access Manager

本節描述如何執行從 Java ES 發行版本 4 至 Java ES 發行版本 5 的 Access Manager 完整升級：

- [升級前作業](#)
- [升級發行版本 4 Access Manager](#)
- [驗證 Access Manager 升級](#)
- [升級後作業](#)
- [回復升級](#)

升級前作業

升級 Access Manager 之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Access Manager 相依元件](#)
- [備份 Directory Server 資料](#)
- [備份 Access Manager 發行版本 4 配置資訊](#)
- [備份 Web 容器自訂檔案](#)
- [備份發行版本 4 Access Manager 的記錄和除錯檔案](#)
- [備份自訂本土化檔案](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可使用下列指令驗證 Access Manager 的目前版本：

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

表 14-5 Access Manager 版本驗證輸出

Java ES 發行版本	Access Manager 版本編號
發行版本 2	6.2
發行版本 3	6 2005Q1
發行版本 4	7 2005Q4
發行版本 5	7.1

升級 Access Manager 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。Access Manager 對一些共用元件有硬性的升級相依性 (請參閱第 43 頁的表 1-9)。

如果您選擇升級 Access Manager 產品元件相依元件，應在您升級 Access Manager 之前按下列順序 (跳過任何可能已經升級的部份) 執行。共用元件的升級通常可由 Java ES 安裝程式自動完成。

1. **共用元件。** 在第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您將 Access Manager 升級到發行版本 5 時，Access Manager 所需要的所有共用元件會由 Java ES 安裝程式自動升級。
2. **Directory Server (非硬性升級相依性)。** 在第 93 頁的第 5 章「Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。
3. **Web 容器軟體 (非硬性升級相依性)。** 在第 123 頁的第 7 章「Web Server」和第 189 頁的第 11 章「Application Server」中分別提供了升級 Web Server 或 Application Server 的指示。

如果未在升級 Access Manager 之前升級 Web 容器軟體，升級程序 (使用 amconfig 程序檔) 會將 Access Manager 配置並重新部署到現有的 Web 容器。

備份 Directory Server 資料

Access Manager 升級程序會使用修改 Directory Server 模式的程序檔。因此，在升級 Access Manager 之前，請先使用 Directory Server Console 或命令行公用程式 (如 db2bak) 備份 Directory Server 資料。可以使用 db2ldif 來備份 Access Manager 模式和目錄資訊樹狀結構 (DIT)。

如需關於備份 Directory Server 的更多資訊，請參閱「Sun Java System Directory Server Enterprise Edition 6.0 管理指南」(<http://docs.sun.com/doc/820-0294>)。

備份 Access Manager 發行版本 4 配置資訊

因為重新配置發行版本 5 Access Manager 軟體需要重新配置發行版本 4 版本，所以將配置檔案備份到已知位置很重要。應該備份下列檔案：

- AMConfig.properties 檔案
AccessManagerConfig-base/config/AMConfig.properties
- serverconfig.xml 檔案
AccessManagerConfig-base/config/serverconfig.xml

- Web 容器配置檔案：
 - 對於 Web Server：請參閱第 244 頁的表 14-3 中的 `server.policy` 和 `server.xml` 檔案的位置
 - 對於 Application Server：請參閱第 244 頁的表 14-3 中的 `server.policy` 和 `domain.xml` 檔案的位置
 - 對於協力廠商 Web 容器：相應的配置檔案
- 認證與自訂模組的 JAR 檔案。
AccessManager-base/lib

備份 Web 容器自訂檔案

如果有為 Access Manager 所參照的任何 Web 容器自訂檔案，則應該備份它們。這些自訂項可能包括以下內容：

- 自訂的 Access Manager 主控台 JSP 頁面：
 - 範圍/舊有主控台 (隨 Java ES 發行版本 4 發行)
AccessManager-base/web-src/services/
 - 僅舊有主控台 (隨 Java ES 發行版本 2 和 3 發行)
AccessManager-base/web-src/applications/
- 自訂的認證 UI JSP 頁面。
AccessManager-base/web-src/services/
- 自訂的 XML 檔案。
AccessManagerConfig-base/config/xml/

提示 請記下您的自訂，以便可以在升級 Access Manager 之後使用備份的程式碼再次套用它們。

備份發行版本 4 Access Manager 的記錄和除錯檔案

為了分析系統狀態資訊，備份記錄和除錯檔案以免遺失是一種好的做法。這些檔案位於下列位置：

- 除錯檔案
/var/AccessManager-base/debug
- 記錄檔案
/var/AccessManager-base/logs

備份自訂本土化檔案

如果您對 Java ES 安裝程式所安裝的本土化檔案進行了任何自訂，或新增了 Java ES 安裝程式所未安裝的新語言本土化，則應該備份這些自訂項。這些自訂項可能包括以下內容：

- 自訂的 Access Manager 使用者介面本土化
*AccessManager-base/locale/*_Locale.properties*
- 自訂的認證 UI JSP 頁面
AccessManager-base/web-src/services/config/auth/default_Locale
- 自訂的線上說明翻譯
AccessManager-base/web-src/services/html/Locale

取得必要的配置資訊和密碼

若要升級 Access Manager，必須提供特定的配置資訊，包括：

- Access Manager 管理員使用者 ID 和密碼
- LDAP 使用者 ID 和密碼
- Access Manager 使用之 Directory Server 實例的目錄管理員名稱和密碼

升級發行版本 4 Access Manager

將 Access Manager 軟體升級到 Java ES 發行版本 5 包括重新配置 Access Manager 和遷移 Access Manager 資料的程序。

升級摘要

升級 Access Manager 的程序包括下列步驟：

1. 升級 [Access Manager Mobile Access](#) 軟體。
2. 移除 [Java ES 發行版本 4 版本的 Access Manager](#)。使用 `ampre71upgrade` 程序檔。
3. 如果發行版本 5 的升級需要本土化，請移除發行版本 4 本土化套裝軟體。此步驟必須手動執行。
4. 安裝 [Java ES 發行版本 5 版本的 Access Manager](#)。使用 Java ES 安裝程式並選擇 [以後配置] 選項。
5. 重新自訂用於 Access Manager 的 JSP。
6. 取消部署 Access Manager，重新配置，再重新部署到 Web 容器中。使用 `amconfig` 程序檔。
7. 更新目錄結構和模式。使用 `amupgrade` 程序檔。

下列程序中對這些步驟逐一進行說明。

升級程序

1. 升級 Access Manager Mobile Access 軟體。

需要藉由為發行版本 4 版本安裝修補程式來升級 Access Manager Mobile Access 軟體。下表顯示了所需的修補程式：

表 14-6 用於升級 Access Manager Mobile Access 軟體的修補程式¹

說明	修補程式 ID : Solaris 9 和 10	修補程式 ID : Linux
Mobile Access 軟體	119530-05 (SPARC) 119531-05 (x86)	119532-05 <ul style="list-style-type: none"> • sun-identity-mobileaccess-6.2-25.3.i386.rpm • sun-identity-mobileaccess-config-6.2-25.3.i386.rpm

1. 修補程式的修訂版號碼是升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

a. 使用表 14-6 中的修補程式號碼取得所需的修補程式。

可從下列位置將修補程式下載到 /tmp：

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

b. 執行修補程式 README 檔案中所指示的任何修補前程序。

c. 取得修補程式將會請求的下列參數的值：

表 14-7 Mobile Access 修補程式參數

參數	值
目錄管理員 DN	預設值：cn=Directory Manager
目錄管理員密碼	

d. 套用表 14-6 中的修補程式。

在 Solaris 上：

```
patchadd /tmp/patch_ID
```

在 Linux 上：

```
./update
```

執行修補程式 README 檔案中所指示的任何修補後程序。

2. 移除 Java ES 發行版本 4 版本的 Access Manager。
 - a. 以超級使用者身份登入到代管發行版本 4 Access Manager 的電腦，或成爲超級使用者。

```
su -
```

- b. 將目錄變更為 Java ES 發行版本 5 發行物中的 `os_arch/Product/identity_svr/Tools` 目錄，其中 `os_arch` 應與您所在平台相符，例如 `Solaris_sparc`。
- c. 取得 `ampre71upgrade` 程序檔將會請求的下列參數的值：

表 14-8 Access Manager 配置參數：ampre71upgrade

參數	值
Directory Server 主機	設定完全合格的名稱： <code>hostname.domain</code>
Directory Server 連接埠	指定非 SSL 連接埠號 ¹ 預設值：389
頂層管理員 DN	預設值： <code>uid=amadmin,ou=People,default_org_DN</code>
頂層管理員密碼	
儲存備份檔案的目錄	預設值： <code>AccessManager-base</code>

1. 如果指定 Directory Server SSL 連接埠 (如預設 SSL 值 636)，升級前程序將無法順利完成。

- d. 確定 Directory Server 正在執行，如果未執行，請啓動它。
- e. 執行 `ampre71upgrade` 程序檔。

```
./ampre71upgrade
```

該程序檔會備份 Access Manager 配置檔案，並移除發行版本 4 基本套裝軟體 (本土化套裝軟體必須依照如下的步驟 3 手動移除)。

3. 如果發行版本 5 的升級需要本土化，請移除發行版本 4 本土化套裝軟體。

在上面步驟 2 中執行的 `ampre71upgrade` 程序檔並不會移除本土化套裝軟體，因此您必須按如下方法手動移除它們。

在 Solaris 上：

- a. 檢查本土化套裝軟體。

```
pkginfo | grep SUNWaml
pkginfo | grep SUNWamclnt
pkginfo | grep SUNWamdistauth
```

- b. 移除在上面步驟 a 中找到的任何本土化套裝軟體。

```
pkgrm SUNWam1Locale
pkgrm SUNWamclntLocale
pkgrm SUNWamdistauthLocale
```

在 Linux 上：

- a. 檢查本土化 RPM。

```
rpm -qa | grep sun-identity-sdk-*
rpm -qa | grep sun-identity-clientsdk-*
rpm -qa | grep sun-identity-distauth-*
```

- b. 移除在上面步驟 a 中找到的任何本土化 RPM。

```
rpm -e sun-identity-sdk-Locale-*
rpm -e sun-identity-clientsdk-Locale-*
rpm -e sun-identity-distauth-Locale-*
```

4. 安裝 Java ES 發行版本 5 版本的 Access Manager。

執行下列步驟：

- a. 在代管發行版本 4 Access Manager 的電腦上啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 從元件選取頁面中選取 Access Manager。
- c. 指定安裝發行版本 4 的那個安裝目錄。
- d. 選取 [以後配置] 選項。
- e. 如有必要，請選取安裝本土化套裝軟體的選項。
- f. 安裝完成時結束 Java ES 安裝程式。

5. 重新自訂用於 Access Manager 的 JSP。

重新套用在第 251 頁的「備份 Web 容器自訂檔案」中儲存的、用於 Access Manager 主控台和認證使用者介面 (UI) 的發行版本 4 自訂 JSP 設定。

然後將自訂 JSP 檔案複製到正確的目錄：

- 範圍/舊有 Access Manager 主控台
AccessManager-base/web-src/services/console
- 僅舊有 Access Manager 主控台
AccessManager-base/web-src/applications/console
- 認證 UI：*AccessManager-base/web-src/services/config/auth/default* 或 *AccessManager-base/web-src/services/config/auth/default_Locale* (其中 *Locale* 指示語言環境，如 *ja*)

如需更多資訊，請參閱「Sun Java System Access Manager 7.1 Developer's Guide」(<http://docs.sun.com/doc/819-4675>)。

6. 取消部署 Access Manager，重新配置，再重新部署到 Web 容器中。

藉由執行 `amconfig` 程序檔配置 Access Manager，以與特定 Web 容器配合使用。`amconfig` 程序檔 (和相關的 `amsamplesilent` 範本輸入檔案) 位於以下目錄：

AccessManager-base/bin

如需關於 `amconfig` 程序檔和 `amsamplesilent` 範本檔案的資訊，請參閱「Sun Java System Access Manager 7.1 管理指南」(<http://docs.sun.com/doc/820-0840>)。

執行下列步驟，重新配置 Access Manager 並將其重新部署到 Web 容器：

- a. 如果選擇升級 Web 容器軟體 (如第 250 頁的「升級 Access Manager 相依元件」中所述)，請確定已完成升級。
- b. 確定 Web 容器的管理實例正在執行中，且處於 `amconfig` 程序檔所支援的模式下，如下表所指示：

表 14-9 `amconfig` 支援的 Administrative Server 模式

Web 容器	支援的模式	預設連接埠號
Application Server (8.x) : Java ES 發行版本 3、4 和 5	SSL (安全) 非 SSL	4849
Web Server (7.0): Java ES 發行版本 5	SSL (安全)	8989
Web Server (6.x) : Java ES 發行版本 2、3 和 4	非 SSL	8888

- c. 如果 Web 容器在 SSL 模式下執行，請確定容器的 SSL 憑證尚未過期，仍然有效。
- d. 如果將 Access Manager 部署於發行版本 5 Web Server 中，請停用與 Access Manager 執行於相同實例中、依賴於 Access Manager 的所有 Java ES 元件。

這些元件可能是例如 Portal Server 的元件或例如 Communications Express、Instant Messaging 或 Delegated Administrator 的 Sun Java Communications Suite 元件。

程序如下：

- I. 以管理員身份登入到 `https://host:8989`。
- II. 請至 [編輯虛擬伺服器]。
- III. 選取 [Web 應用程式] 標籤。
- IV. 核取所有依賴於 Access Manager 的應用程式。
- V. 按一下 [停用]。
- VI. 按一下 [儲存]。
- VII. 按一下 [部署申請中 | 部署配置]。

配置變更會傳遞至 Web Server 實例。

- e. 檢查 Directory Server 及相應的 Web 容器是否正在執行。
- f. 建立一個基於 `amsamplesilent` 範本輸入檔案的 `amconfig` 輸入檔案：

```
cp amsamplesilent config-file
```

(在後續步驟中，假設 `config-file` 位於與 `amsamplesilent` 相同的目錄中。)

- g. 在 `config-file` 中設定配置參數。

需要正確設定所有參數。一些值可以自 `AMConfig.properties` 檔案遷移，相較之下，其他值則是升級程序的專用值，如下表所示。

表 14-10 Access Manager 配置參數：amconfig

參數	值
升級參數	
DEPLOY_LEVEL	設定為 26 (表示取消部署) 或 設定為 1 (表示重新配置並部署)
DIRECTORY_MODE	設定為 5
AM_REALM ¹	如果啓用了舊有模式，則設定為 disabled 如果啓用了範圍模式，則設定為 enabled 預設值：enabled
JAVA_HOME	設定為 JDK 發行版本 5 目錄： /usr/java/jdk1.5.0_04/
WEB_CONTAINER	針對 Web Server 7.x 設定為 WS 針對 Web Server 6.x 設定為 WS6 針對 Application Server 8.x 設定為 AS8 針對 IBM WebSphere 5.x 設定為 WAS5 針對 BEA WebLogic 8.x 設定為 WL8 且只填寫對應的 <i>config-file</i> 區段。
WS_INSTANCE (若使用 Web Server 7.x 做為 Web 容器)	設定為大小寫相符的實例配置目錄名稱： <i>https-configName1</i> 此目錄位於以下路徑中： <i>WebServer7Config-base/https-configName1</i>
WS61_INSTANCE (若使用 Web Server 6.x 做為 Web 容器)	設定為大小寫相符的實例配置目錄名稱： <i>https-instanceName</i> 此目錄位於以下路徑中： <i>WebServer6-base/https-instanceName1</i>
AS81_INSTANCE (使用 Application Server 8.x 做 為 Web 容器)	設定為 Application Server 8.x <i>instanceName</i> 預設值：server
AS81_INSTANCE_DIR (使用 Application Server 8.x 做 為 Web 容器)	設定為實例的 Application Server 8.x 網域目錄，預設 值為 <i>AppServer8Config-base/domains/domain1</i>
AS81_DOCS_DIR (使用 Application Server 8.x 做 為 Web 容器)	設定為實例的 Application Server 8.x 文件根目錄，預 設值為 <i>AppServer8Config-base/domains/domain1/docroot</i>
遷移自 AMConfig.properties	
SERVER_PROTOCOL	com.ipplanet.am.server.protocol
SERVER_PORT	com.ipplanet.am.server.port
SERVER_HOST	com.ipplanet.am.server.host

表 14-10 Access Manager 配置參數：amconfig（續）

參數	值
DS_HOST	com.iplanet.am.directory.host
DS_PORT	com.iplanet.am.directory.port
ROOT_SUFFIX ²	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 ²	am.encrypted.pwd ³

1. 如需有關範圍和舊有模式的更多資訊，請參閱第 245 頁的「相容性問題」。
2. 此參數的值應與之前版本的 Access Manager 相同。
3. 如果同時部署了 Access Manager 和 Access Manager SDK，則對於 Access Manager 實例及其相關的 Access Manager SDK 實例，此特性的值必須相同。

若為其他參數，除非要變更 Web 容器或密碼，否則請提供要升級的發行版本 4 配置中使用的那些值。例如，如果您已經將 Web Server 升級到發行版本 5，請提供下列值：

表 14-11 amconfig 參數：發行版本 5 Web Server

參數	值
WS_CONFIG	Web Server 配置的名稱： <i>configName</i>
WS_INSTANCE	<i>https-configName</i>
WS_HOME	<i>WebServer7Config-base</i>
WS_PROTOCOL	http 或 https
WS_HOST	Web Server 實例偵聽連線時使用的完全合格的主機名稱
WS_PORT	Web Server 實例偵聽連線時所在的連接埠
WS_ADMINPORT	Web Server 管理實例偵聽連線時所在的連接埠
WS_ADMIN	Web Server 管理員使用者 ID
WS_ADMINPASSWD	Web Server 管理員密碼

h. 執行 `amconfig` 以取消部署 Access Manager

將 `config-file` 中的 `DEPLOY_LEVEL` 值設定為 26。

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

i. 檢查以確定 Common Agent Container 正在執行中。

```
netstat -an | grep 11163
```

如果並不在執行中，請啟動它。

```
/usr/sbin/cacoadm start
```

j. 執行 `amconfig` 以重新配置 Access Manager 並將它部署到 Web 容器中。

將 `config-file` 中的 `DEPLOY_LEVEL` 值設定為 1。

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

7. 更新目錄結構和模式。

發行版本 5 Access Manager 與發行版本 4 的目錄結構並存，但必須修改其結構才能支援發行版本 5 的功能。藉由執行 `amupgrade` 程序檔，將 Access Manager 目錄結構和模式更新到發行版本 5，該程序檔安裝在下列目錄中：

- 在 Solaris 上：
`AccessManager-base/upgrade/scripts`
- 在 Linux 上：
`AccessManager_base/identity/upgrade/scripts`
- a.** 取得 `amupgrade` 程序檔將會請求的下列參數的值：

表 14-12 Access Manager 配置參數：amupgrade

參數	值
Directory Server 完全合格的主機名稱	設定完全合格的名稱： <code>hostname.domain</code>
Directory Server 連接埠	指定非 SSL 連接埠號 ¹ 預設值：389
目錄管理員 DN	預設值： <code>cn=Directory Manager</code>
目錄管理員密碼	
頂層管理員 DN	預設值： <code>uid=amadmin,ou=People,default_org_DN</code>
頂層管理員密碼	

表 14-12 Access Manager 配置參數：amupgrade（續）

參數	值
啓用範圍模式 (當從發行版本 4 範圍模式升級時不會請求此參數值。)	Y/N：是表示啓用範圍模式，並將服務資料遷移至新的範圍樹狀結構 ² 。否 (預設值) 表示服務資料保留在舊有模式中。

1. 必須指定與預設 SSL 值 636 不同的 Directory Server SSL 連接埠。
2. 請參閱第 262 頁的「遷移至範圍模式」。

- b. 執行 amupgrade 程序檔。

```
cd AccessManager-base/upgrade/scripts
./amupgrade
```

如果升級成功，程序檔會顯示 [升級已完成]。

- c. 檢查下列升級記錄檔案，以瞭解關於目錄模式延伸的資訊：

在 Solaris 上：

```
/var/sadm/install/logs/
    Sun_Java_System_Access_Manager_upgrade_dit_log.mmdhmm
```

在 Linux 上：

```
/var/log/Sun_Java_System_Access_Manager_upgrade_dit_log.mmdhmm
```

8. 啓用在第 257 頁的步驟 d 中停用的任何元件。
9. 重新啓動部署了 Access Manager 的 Web 容器。
10. 啓動 Access Manager。
重新啓動部署了 Access Manager 的 Web 容器。

驗證 Access Manager 升級

完成升級程序之後，請按如下方式驗證升級是否成功：

1. 使用下列指令檢查 Access Manager 套裝軟體的升級：

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

請參閱第 249 頁的表 14-5 中的輸出值。

2. 藉由檢查 /var/sadm/install/logs 目錄中的下列安裝程式記錄檔來檢視升級的狀態：
 - o Java_Shared_Component_Install.timestamp
 - o Java_Enterprise_System_install.Atimestamp

- `Java_Enterprise_System_install.Btimestamp`
 - `Java_Enterprise_System_Summary_Report_install.timestamp`
3. 藉由在執行 `amupgrade` 程序檔時檢查終端機視窗中的錯誤來檢視 Access Manager 遷移的狀態。

同時，請檢查 `/var/sadm/install/logs` 目錄中的下列記錄檔：

`Sun_Java_System_Access_Manager_upgrade_dit_log.timestamp`

4. 檢視 Access Manager 疑難排解檔案中是否有錯誤。

檔案位於在 `AMConfig.properties` 檔案的 `com.ipplanet.services.debug.directory` 特性中指定的位置。預設值為：

在 Solaris 上：

`/var/opt/SUNWam/debug`

在 Linux 上：

`/var/opt/sun/identity/debug`

升級後作業

請注意處理以下情況所需要的升級後程序：

- [遷移至範圍模式](#)
- [安全宣示標記語言](#)

遷移至範圍模式

如果在將 Access Manager 升級到發行版本 5 時已經遷移到範圍模式，(亦即，當使用 `amupgrade` 來更新目錄結構和模式時，啟用範圍模式參數的值為 `Yes`)，則執行以下步驟：

1. 開啓 `AccessManagerConfig-base/config/AMConfig.properties` 檔案。
2. 檢查以下特性的值：
`com.sun.identity.sm.ldap.enableProxy`
3. 如果特性未設定為 `false`，則手動將它設定為 `false`。

安全宣示標記語言

如果正在使用安全宣示標記語言 (SAML) 服務，則必須使用 Access Manager 主控台來增加和啟用 SAML 認證模組。如需有關建立 SAML 認證模組實例的資訊，請參閱「Sun Java System Access Manager 7.1 管理指南」(<http://docs.sun.com/doc/820-0840>)。

回復升級

未提供用於將 Access Manager 回復到其升級前狀態的程序檔。該程序必須使用作為升級前作業之一而備份的 Access Manager 資料手動執行 (請參閱第 251 頁的「備份發行版本 4 Access Manager 的記錄和除錯檔案」)。取消升級是很難實現的。

一種回復方法是重新安裝發行版本 4 並將所有備份的配置檔案遷移到它們的正確位置。另一種方法是在升級前使用備份的配置檔案建立平行系統，並於嘗試升級前對此平行系統進行測試。

多重實例升級

在某些部署架構中，會將 Access Manager 部署在多個電腦系統上，以提供高可用性和延展性。

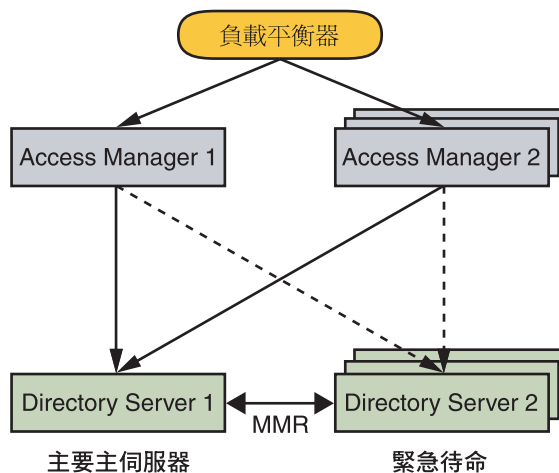
能夠循序升級 Access Manager 實例而不中斷服務往往是人們所希望的。本節討論從發行版本 4 Access Manager 到發行版本 5 執行此類漸進式升級的程序。

備註

目前發行版本不支援升級在相同主機系統上安裝的 Access Manager 的多重實例。如果您在相同主機上具有多重實例，在您升級主要實例後，您必須重新建立其他的實例。

將使用下圖所示的部署架構來說明漸進式升級的程序。

圖 14-1 多重 Access Manager 實例的部署架構範例



在此架構中，透過負載平衡器存取多重 Access Manager 實例，而這些實例進一步存取為多個主伺服器複製 (MMR) 設定的目錄。雖然其他 Directory Server 複製方案也是允許的，但 MMR 是高可用性及延展性目錄服務的代表。在圖 14-1 中，Access Manager 和 Directory Server 的多重實例已組成群組，以便於對升級程序進行說明。例如，Access Manager 2 就代表 Access Manager 的第二個到第 n 個實例。

執行從發行版本 4 Access Manager 到發行版本 5 漸進式升級的程序以下列互通性為基礎：發行版本 5 Access Manager 和發行版本 4 Access Manager 實例可以並存，並可針對同一目錄同時執行，但前提是目錄模式尚未更新至發行版本 5。

因此，針對指向單一 Directory Server 實例的 Access Manager 實例，您可以將目錄模式的更新延遲到所有 Access Manager 實例皆已升級為止，以執行漸進式升級。

您可以使用以下程序執行從發行版本 4 Access Manager 到發行版本 5 的漸進式升級：

1. 在所有 Access Manager 實例上備份發行版本 4 的配置資訊。

請參閱第 244 頁的表 14-3。

2. 升級 Access Manager 1。

- a. 在負載平衡器中停用 Access Manager 1。

請求不會再被路由至 Access Manager 1。

- b. 部份升級 Access Manager 1。

如第 252 頁的「升級發行版本 4 Access Manager」中所述升級 Access Manager，但不更新目錄結構與模式 (第 260 頁的步驟 7)。

- c. 在負載平衡器中啟用 Access Manager 1。

3. 升級 Access Manager 2 至 Access Manager n。

為了簡潔起見，在後續步驟中，「Access Manager 2」將表示 Access Manager 2 至 Access Manager n。

- a. 在負載平衡器中停用 Access Manager 2。

請求不會再被路由至 Access Manager 2。

- b. 部份升級 Access Manager 2。

如第 252 頁的「升級發行版本 4 Access Manager」中所述升級每一個 Access Manager 實例，但不更新目錄結構與模式 (第 260 頁的步驟 7)。

- c. 在負載平衡器中啟用 Access Manager 2。

請求會重新路由至 Access Manager 2。

4. 更新 Directory Server 1 的目錄結構和模式。

如第 260 頁的步驟 7 中所述使用 `amupgrade` 程序檔。Access Manager 1 至 n 在更新完 Directory Server 1 的模式後，將會繼續運作。

僅升級發行版本 4 Access Manager SDK

在某些部署架構中，會將 Access Manager SDK 元件安裝在一個或更多個電腦系統上，而不在這些電腦上安裝其他 Access Manager 元件。Access Manager SDK 充當 Access Manager 的遠端介面，因此必須針對 Access Manager 的作業模式 (舊有或範圍) 將其重新配置。

Access Manager SDK 和將它做為遠端介面使用的完整 Access Manager 皆應升級到發行版本 5。但是，發行版本 5 Access Manager 向下相容發行版本 4 Access Manager SDK，因此通常應首先升級 Access Manager，然後再在其他電腦上升級 Access Manager SDK。

做為 Access Manager 的遠端介面，不需要配置 SDK 就能存取 Directory Server。如果正在使用 Access Manager SDK 來支援 Web 元件 (如 Portal Server，它依賴 Web 容器服務)，則必須針對相應的 Web 容器來配置 Access Manager SDK。不過，Access Manager SDK 也可以支援非 Web 元件，且不需要 Web 容器。

基於上述特性，升級 Access Manager SDK 的程序是 Access Manager 完整升級程序的子集。

本節描述如何執行從 Java ES 發行版本 4 到 Java ES 發行版本 5 的僅 Access Manager SDK 升級：

- [升級前作業](#)
- [升級發行版本 4 Access Manager SDK](#)
- [驗證 Access Manager SDK 升級](#)
- [升級回復](#)

升級前作業

除了與 Directory Server 和 Access Manager 管理工具 JSP 自訂有關的作業外，Access Manager SDK 的升級前作業與 Access Manager 完整升級的升級前作業相同 (請參閱第 249 頁的「升級前作業」)。Access Manager SDK 所需的升級前作業如下：

- [第 250 頁的「升級 Access Manager 相依元件」](#)

不過，Access Manager SDK 對 Directory Server 沒有相依性，且唯有當 Access Manager SDK 在 Web 容器中執行時，才會對 Web 容器軟體有相依性。

- [第 250 頁的「備份 Access Manager 發行版本 4 配置資訊」](#)

不過，對於 Access Manager SDK，唯有當 Access Manager SDK 在 Web 容器中執行時，才需要備份 Web 容器配置檔案。

- [第 251 頁的「備份發行版本 4 Access Manager 的記錄和除錯檔案」](#)

還需要取得用於存取這些檔案的管理使用者名稱和密碼。

升級發行版本 4 Access Manager SDK

除了與本土化、Access Manager 管理工具 JSP 自訂和遷移目錄模式有關的程序外，Access Manager SDK 的升級程序與 Access Manager 完整升級的升級程序相同。

1. 移除 Java ES 發行版本 4 版本的 Access Manager SDK。

除了只移除 Access Manager SDK 之外，其他動作請遵循[第 254 頁的「移除 Java ES 發行版本 4 版本的 Access Manager。」](#)中的指示。

2. 安裝 Java ES 發行版本 5 版本的 Access Manager SDK。

除了只安裝 Access Manager SDK 之外，其他動作請遵循[第 255 頁的「安裝 Java ES 發行版本 5 版本的 Access Manager。」](#)中的指示。

3. 重新配置 Access Manager SDK。

除了按以下所述設定 `DEPLOY_LEVEL` 參數以外，其他動作請遵循[第 256 頁的「取消部署 Access Manager，重新配置，再重新部署到 Web 容器中。」](#)中的指示：

- 如果針對 Web 容器配置了 Access Manager SDK：
`DEPLOY_LEVEL=4` (升級 SDK 並配置 Web 容器)
- 如果未針對 Web 容器配置 Access Manager SDK：
`DEPLOY_LEVEL=3` (僅升級 SDK)

驗證 Access Manager SDK 升級

有三種方法可以驗證 Access Manager SDK 升級是否成功：

- 執行 Portal Server 或其他使用 Access Manager SDK 來與 Access Manager 對接的元件，並檢查認證是否正常。
- 執行下列位置中提供的 Access Manager SDK 範例：

AccessManager-base/samples/sdk

- 檢查 `com.iplanet.am.version` 特性的值，它位於 `AMConfig.properties` 檔案中：

`AccessManagerConfig-base/config/AMConfig.properties`

升級回復

未提供用於將 Access Manager 回復到其升級前狀態的程序檔。該程序必須使用作為升級前作業之一而備份的 Access Manager 資料手動執行（請參閱第 251 頁的「備份發行版本 4 Access Manager 的記錄和除錯檔案」）。取消升級是很難實現的。

一種回復方法是重新安裝 R4 並將所有備份的配置檔案遷移到它們的正確位置。另一種方法是在升級前使用備份的配置檔案建立平行系統，並於嘗試升級前對此平行系統進行測試。

從 Java ES 發行版本 3 升級 Access Manager

除了執行多重實例升級的方式以外，將 Java ES 2003Q1 (發行版本 3) Access Manager 或 Access Manager SDK 升級到發行版本 5 的程序與將發行版本 4 Access Manager 或 Access Manager SDK 升級到發行版本 5 的程序相似。

- [升級發行版本 3 Access Manager](#)
- [多重實例升級](#)

備註 發行版本 5 Access Manager 與某些發行版本 3 Sun Java Communications Suite 元件不相容。如果 Access Manager 升級到發行版本 5，則發行版本 3 或更早的 Delegated Administrator 也必須升級到發行版本 5，才能針對 Messaging Server 和 Calendar Server 佈建使用者。不過，並非一定要將 Messaging Server 和 Calendar Server 本身也升級到發行版本 5。

升級發行版本 3 Access Manager

若要將發行版本 3 Access Manager 或 Access Manager SDK 升級到發行版本 5，除了將參照發行版本 4 的地方取代為發行版本 3 外，其他請遵循第 248 頁的「[從 Java ES 發行版本 4 升級 Access Manager](#)」中的指示。

多重實例升級

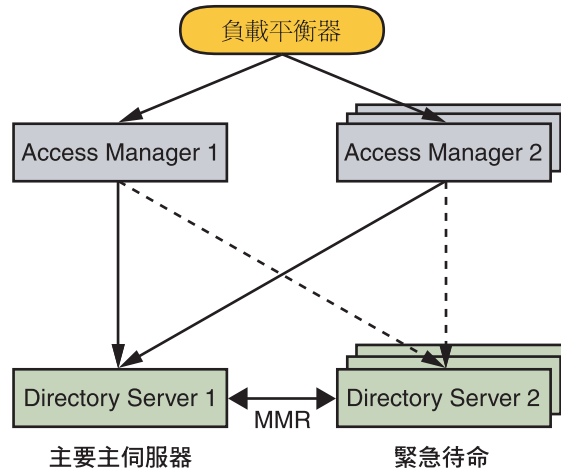
在某些部署架構中，會將 Access Manager 部署在多個電腦系統上，以提供高可用性和延展性。

能夠循序升級多重 Access Manager 實例而不中斷服務往往是人們所希望的。本節討論從發行版本 3 Access Manager 到發行版本 5 執行此類漸進式升級的程序。

備註 目前發行版本不支援升級在相同主機系統上安裝的 Access Manager 的多重實例。如果您在相同主機上具有多重實例，在您升級主要實例後，您必須重新建立其他的實例。

將使用下圖所示的部署架構來說明漸進式升級的程序。

圖 14-2 多重 Access Manager 實例的部署架構範例



在此架構中，透過負載平衡器存取多重 Access Manager 實例，而這些實例進一步存取為多個主伺服器複製 (MMR) 設定的目錄。雖然其他 Directory Server 複製方案也是允許的，但 MMR 是高可用性及延展性目錄服務的代表。在圖 14-2 中，Access Manager 和 Directory Server 的多重實例已組成群組，以便於對升級程序進行說明。例如，Access Manager 2 就代表 Access Manager 的第二個到第 n 個實例。

執行從發行版本 3 Access Manager 到發行版本 5 漸進式升級的程序以下列限制為基礎：發行版本 5 Access Manager 不可以與發行版本 3 的目錄結構並存。但是，如果複製了 Directory Server 實例，如圖 14-2 所示，那麼您可以使用以下程序執行漸進式升級：

1. 在所有 Access Manager 實例上備份發行版本 3 的配置資訊。
請參閱第 244 頁的表 14-3。
2. 修改 Access Manager 1 的配置。
 - a. 將 Access Manager 1 配置為指向 Directory Server 2，而非指向 Directory Server 1。
 - b. 重新啟動 Access Manager 1。

Access Manager 1 將會繼續處理請求，同時將對 Access Manager 2 至 Access Manager n 進行升級。

3. 升級 Access Manager 2 至 Access Manager n。

爲了簡潔起見，在後續步驟中，「Access Manager 2」將表示 Access Manager 2 至 Access Manager n。

- a. 在負載平衡器中停用 Access Manager 2。

請求不會再被路由至 Access Manager 2。

- b. 部份升級 Access Manager 2。

如第 252 頁的「[升級發行版本 4 Access Manager](#)」中所述升級每一個 Access Manager 實例，但不更新目錄結構與模式 (第 260 頁的步驟 7)。

- c. 停用 Directory Server MMR。

- d. 更新 Directory Server 1 的目錄結構和模式。

如第 260 頁的步驟 7 中所述使用 amupgrade 程序檔。Access Manager 1 將會繼續正常運作，因爲 Directory Server 2 的模式不會更新。

- e. 重新啓動 Access Manager 2。

- f. 在負載平衡器中啓用 Access Manager 2。

請求會重新路由至 Access Manager 2。

4. 升級 Access Manager 1。

- a. 在負載平衡器中停用 Access Manager 1。

請求不會再被路由至 Access Manager 1。

- b. 部份升級 Access Manager 1。

如第 252 頁的「[升級發行版本 4 Access Manager](#)」中所述升級 Access Manager，但不更新目錄結構與模式 (第 260 頁的步驟 7)。

- c. 啓用 Directory Server MMR。

現在已經更新了 Directory Server 2 的模式 (和資料)。

- d. 將 Access Manager 1 的配置復原爲指向 Directory Server 1。

- e. 重新啓動 Access Manager 1。

- f. 在負載平衡器中啓用 Access Manager 1。

請求會重新被路由至 Access Manager 1 以及其他所有升級的 Access Manager 實例。

從 Java ES 發行版本 2 升級 Access Manager

將 Java ES 2004Q2 (發行版本 2) Access Manager 升級到發行版本 5 的程序與將發行版本 4 Access Manager 升級到發行版本 5 的程序相似，只有幾處不同，如以下幾節所述：

- [升級前作業](#)
- [升級發行版本 2 Access Manager](#)
- [多重實例升級](#)

此外，將 Java ES 2004Q2 (發行版本 2) Access Manager SDK 升級到發行版本 5 的程序也與將發行版本 4 Access Manager SDK 升級到發行版本 5 (請參閱第 265 頁的「[僅升級發行版本 4 Access Manager SDK](#)」) 的程序相似，例外情況也相似。如果僅升級 Access Manager SDK，則不包含與本土化、Access Manager 管理工具 JSP 自訂和遷移目錄模式相關的程序。

發行版本 2 Access Manager SDK 和將它做為遠端介面使用的完整發行版本 2 Access Manager 都必須升級到發行版本 5。不支援混合使用發行版本 2 和發行版本 5 元件。因此，所有電腦上的所有發行版本 2 Access Manager 和發行版本 2 Access Manager SDK 實例都必須升級到發行版本 5。

備註 如果您要在 Linux 平台上從發行版本 2 Access Manager 進行升級，則必須執行雙重升級，也就是 Access Manager 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Access Manager)。如需更多資訊，請參閱第 247 頁的「[雙重升級](#)」。

升級前作業

升級 Access Manager 之前，請執行第 249 頁的「[升級前作業](#)」中描述的程序，下面是一些例外情況和補充步驟。

- [升級 Access Manager 相依元件](#)
- [升級目錄模式](#)
- [為目錄重新編製索引](#)

升級 Access Manager 相依元件

相較於從發行版本 4 升級，將發行版本 2 升級到發行版本 5 的升級前作業需要將所有共用元件 (請參閱第 43 頁的表 1-9) 和 Access Manager 依賴的本機所有產品元件升級到發行版本 5。

升級 Access Manager 相依元件時，應該按照下列順序升級它們，且都須在升級 Access Manager 之前進行。可以略過可能已升級過的任何相依性。

1. **共用元件。** 在第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。但是，當您執行發行版本 5 Access Manager 的全新安裝時，Java ES 共用元件會由安裝程式自動升級。
2. **Directory Server。** Directory Server 很少與 Access Manager 位於同一電腦上，但第 108 頁的「從 Java ES 發行版本 2 升級 Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。
3. **Web 容器軟體。** 在第 143 頁的「從 Java ES 發行版本 3 升級 Web Server」和第 203 頁的「從 Java ES 發行版本 2 升級 Application Server」中分別提供了升級 Web Server 和 Application Server 的指示。

升級目錄模式

如果使用 Sun Java Communications Suite 的 Directory Preparation Tool (`comm_dssetup.pl`) 將 Directory Server 配置為支援 Communication Suite 元件 (例如 Messaging Server 和 Calendar Server)，則必須在升級 Access Manager 之前先使用 Directory Preparation Tool 6.4 升級目錄模式 (請參閱「Sun Java Communications Suite 5 Upgrade Guide」(<http://docs.sun.com/doc/819-7561>))。請在升級 Access Manager 相依元件之後執行此升級前作業。

為目錄重新編製索引

為了避免在升級目錄模式後執行 Access Manager 升級時的複雜性 (請參閱上面的「升級目錄模式」)，您需要為 Access Manager 目錄根尾碼手動重新編製索引，如下所示：

發行版本 2 至發行版本 4 Directory Server：

1. `cd serverRoot/slapd-`hostname``
2. `./db2index.pl -D "cn=directory manager" -w passwordFile -n databaseName`

其中預設的 `databaseName` 是 `userRoot`。

發行版本 5 Directory Server :

1. `cd DirServer-base/ds6/bin`
2. `./dsconf reindex -D "cn=Directory Manager" -e -w passwordFile suffix`

其中

-e 代表不安全的連線

-D 是目錄管理員

-w 是僅包含密碼的密碼檔案

suffix 是 Access Manager 目錄根尾碼。

視目錄中的項目數目而定，可能需要花大量時間才能完成索引的重新編製。

升級發行版本 2 Access Manager

將 Access Manager 從發行版本 2 升級到發行版本 5 的程序取決於部署 Access Manager 軟體的 Web 容器。

升級發行版本 2 Access Manager : Web Server Web 容器

若要將發行版本 2 Access Manager 升級到發行版本 5，在部署到 Web Server Web 容器中時，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他動作請遵循第 252 頁的「[升級發行版本 4 Access Manager](#)」中的指示。

升級發行版本 2 Access Manager : Application Server Web 容器

若要將發行版本 2 Access Manager 升級到發行版本 5，部署到 Application Server Web 容器中時會有兩種情況：

- **已全新安裝發行版本 5 Application Server**。在此情況下，若要將發行版本 2 Access Manager 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他請遵循第 252 頁的「[升級發行版本 4 Access Manager](#)」中的指示。
- **已將發行版本 2 Application Server 升級到發行版本 5**。在此情況下，升級到發行版本 5 時，原來部署了 Access Manager 的發行版本 2 Application Server 實例 (*instanceName*) 已遷移到由升級程序建立的節點代理程式下。在這個已升級的 Application Server 實例中升級 Access Manager 需要執行下列幾節中所述的步驟。

升級摘要

升級 Access Manager 的程序包括下列步驟：

1. 升級 Access Manager Mobile Access 軟體。
2. 移除 Java ES 發行版本 2 版本的 Access Manager。使用 `ampre71upgrade` 程序檔。
3. 如果發行版本 5 的升級需要本土化，請移除發行版本 2 本土化套裝軟體。此步驟必須手動執行。
4. 安裝 Java ES 發行版本 5 版本的 Access Manager。使用 Java ES 安裝程式並選擇 [以後配置] 選項。
5. 重新自訂用於 Access Manager 的 JSP。
6. 檢查 Directory Server 是否正在執行。
7. 啓動下列 Application Server 實例：部署 Access Manager 的 Domain Administration Server (DAS)、節點代理程式和伺服器實例。
8. 取消部署 Access Manager，重新配置，然後重新部署到 Application Server 實例中。使用 `amconfig` 程序檔。
9. 驗證 Access Manager `classpath-suffix` 和 `server-classpath` 資訊已遷移至發行版本 5 Application Server `domain.xml` 檔案中。
10. 停止 Domain Administration Server (DAS)、節點代理程式和伺服器實例。
11. 重新啓動 Domain Administration Server (DAS)、節點代理程式和伺服器實例。
12. 更新目錄結構和模式。使用 `amupgrade` 程序檔。

下列程序中將對這些步驟進行說明。

升級程序

1. 升級 Access Manager Mobile Access 軟體。

需要藉由為發行版本 2 版本安裝修補程式來升級 Access Manager Mobile Access 軟體。下表顯示了所需的修補程式。

表 14-13 用於升級 Access Manager Mobile Access 軟體的修補程式¹

說明	修補程式 ID : Solaris 9 和 10	
	修補程式 ID : Solaris 9 和 10	修補程式 ID : Linux
Mobile Access 軟體	119530-05 (SPARC) 119531-05 (x86)	119532-05 <ul style="list-style-type: none"> sun-identity-mobileaccess-6.2-25.3.i386.rpm sun-identity-mobileaccess-config-6.2-25.3.i386.rpm

1. 修補程式的修訂版號碼是升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

a. 使用表 14-6 中的修補程式號碼取得所需的修補程式。

可從下列位置將修補程式下載到 /tmp：

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

b. 執行修補程式 README 檔案中所指示的任何修補前程序。

c. 取得修補程式將會請求的下列參數的值：

表 14-14 Mobile Access 修補程式參數

參數	值
目錄管理員 DN	預設值：cn=Directory Manager
目錄管理員密碼	

d. 套用表 14-6 中的修補程式。

在 Solaris 上：
patchadd *patch_ID*

在 Linux 上：
./update

執行修補程式 README 檔案中所指示的任何修補後程序。

2. 移除 Java ES 發行版本 2 版本的 Access Manager。
 - a. 以超級使用者身份登入到代管發行版本 4 Access Manager 的電腦，或成爲超級使用者。

```
su -
```

- b. 將目錄變更改爲 Java ES 發行版本 5 發行物中的 `os_arch/Product/identity_svr/Tools` 目錄，其中 `os_arch` 應與您所在平台相符，例如 `Solaris_sparc`。

- c. 取得 `ampre71upgrade` 程序檔將要請求的下列參數值：

表 14-15 Access Manager 配置參數：ampre71upgrade

參數	值
Directory Server 主機	設定完全合格的名稱： <code>hostname.domain</code>
Directory Server 連接埠	指定非 SSL 連接埠號 ¹ 預設值：389
頂層管理員 DN	預設值： <code>uid=amadmin,ou=People,default_org_DN</code>
頂層管理員密碼	
儲存備份檔案的目錄	預設值： <code>AccessManager-base</code>

1. 必須指定與預設 SSL 值 636 不同的 Directory Server SSL 連接埠。

- d. 確定 Directory Server 正在執行，如果未執行，請啓動它。
- e. 執行 `ampre71upgrade` 程序檔。

```
./ampre71upgrade
```

該程序檔會備份 Access Manager 配置檔案，並移除發行版本 4 基本套裝軟體 (本土化套裝軟體必須依照如下的步驟 3 手動移除)。

3. 如果發行版本 5 的升級需要本土化，請移除發行版本 2 本土化套裝軟體。

在上面步驟 2 中執行的 `ampre71upgrade` 程序檔並不會移除本土化套裝軟體，因此您必須按如下方法手動移除它們。

在 Solaris 上：

- a. 檢查本土化套裝軟體。

```
pkginfo | grep SUNWaml
pkginfo | grep SUNWamclnt
pkginfo | grep SUNWamdistauth
```

- b. 移除在上面步驟 a 中找到的任何本土化套裝軟體。

```
pkgrm SUNWam1Locale
pkgrm SUNWamclntLocale
pkgrm SUNWamdistauthLocale
```

在 Linux 上：

- a. 檢查本土化 RPM。

```
rpm -qa | grep sun-identity-sdk-*
rpm -qa | grep sun-identity-clientsdk-*
rpm -qa | grep sun-identity-distauth-*
```

- b. 移除在上面步驟 a 中找到的任何本土化 RPM。

```
rpm -e sun-identity-sdk-Locale-*
rpm -e sun-identity-clientsdk-Locale-*
rpm -e sun-identity-distauth-Locale-*
```

4. 安裝 Java ES 發行版本 5 版本的 Access Manager。

執行下列步驟：

- a. 在代管發行版本 2 Access Manager 的電腦上啟動 Java ES 安裝程式。

```
cd Java ES Release 5 distribution/os_arch
./installer
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。(針對指令行介面使用 `installer -nodisplay` 選項。)

顯示出 [歡迎] 和 [授權合約] 頁面之後，您將會看到元件選取頁面。(當偵測到可由 Java ES 安裝程式直接升級的安裝元件時，它們的狀態會顯示為 [可升級]。)

- b. 從元件選取頁面中選取 Access Manager。
- c. 指定安裝發行版本 2 的那個安裝目錄。
- d. 選取 [以後配置] 選項。
- e. 如有必要，請選取安裝本土化套裝軟體的選項。
- f. 安裝完成時結束 Java ES 安裝程式。

5. 重新自訂用於 Access Manager 的 JSP。

重新套用在第 251 頁的「備份 Web 容器自訂檔案」中儲存的、用於 Access Manager 主控台和認證使用者介面 (UI) 的發行版本 2 自訂 JSP 設定。

然後將自訂 JSP 檔案複製到正確的目錄：

- 僅舊有 Access Manager 主控台
AccessManager-base/web-src/applications/console
- 認證 UI：*AccessManager-base/web-src/services/config/auth/default* 或 *AccessManager-base/web-src/services/config/auth/default_Locale* (其中 *Locale* 指示語言環境，如 *ja*)

如需更多資訊，請參閱「Sun Java System Access Manager 7.1 Developer's Guide」(<http://docs.sun.com/doc/819-4675>)。

6. 檢查 Directory Server 是否正在執行。

7. 啟動下列 Application Server 實例：

以下指令及後續步驟中使用了下列慣例：

- 其中 *nodeagentName* 的形式為 *hostName_domainName*，但其預設形式為簡單的 *hostName*
- 預設 *domainName* 是 *domain1*
- 預設 *instanceName* 是 *server1*

備註 如下所示，在啟動伺服器實例之前，務必要先使用 `startinstances=false` 選項單獨啟動節點代理程式。

a. 啟動 Domain Administration Server (DAS)

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
domainName
```

b. 啟動所升級 Application Server 實例遷移到的節點代理程式

```
AppServer8-base/bin/asadmin start-node-agent  
--startinstances=false --user admin_ID nodeagentName
```

c. 若部署 Access Manager 的伺服器實例 (*instanceName*) 尚未執行，請啟動它

```
AppServer8-base/bin/asadmin start-instance --user admin_ID  
instanceName
```

8. 取消部署 Access Manager，重新配置，然後重新部署到 Application Server 實例中。
 - a. 如果 Web 容器在 SSL 模式下執行，請確定容器的 SSL 憑證尚未過期，仍然有效。
 - b. 建立一個基於 `amsamplesilent` 範本輸入檔案的 `amconfig` 輸入檔案：


```
cp amsamplesilent config-file
```

(在後續步驟中，假設 `config-file` 位於與 `amsamplesilent` 相同的目錄中。)
 - c. 在 `config-file` 中設定配置參數。
需要正確設定所有參數。一些值可以自 `AMConfig.properties` 檔案遷移，相較之下，其他值則是升級程序的專用值，如下表所示。

表 14-16 Access Manager 配置參數：amconfig

參數	值
升級參數	
DEPLOY_LEVEL	設定為 26 (表示取消部署) 或 設定為 1 (表示重新配置並部署)
DIRECTORY_MODE	設定為 5
AM_REALM ¹	如果啓用了舊有模式，則設定為 <code>disabled</code> 如果啓用了範圍模式，則設定為 <code>enabled</code> 預設值： <code>enabled</code>
JAVA_HOME	設定為 JDK 發行版本 5 目錄： <code>/usr/java/jdk1.5.0_04/</code>
WEB_CONTAINER	針對 Application Server 8.x 設定為 <code>AS8</code> 且只填寫對應的 <code>config-file</code> 區段。
AS81_INSTANCE (使用從 Application Server 7.x 升級的 Application Server 8.x 做為 Web 容器)	設定為 Application Server 7.x <code>instanceName</code> ，預設值為 <code>server1</code>
AS81_INSTANCE_DIR (使用 Application Server 8.x 做 為 Web 容器)	設定為實例的 Application Server 8.x 網域目錄，預設 值為 <code>AppServer8Config-base/domains/domain1</code>
AS81_DOCS_DIR (使用 Application Server 8.x 做 為 Web 容器)	設定為實例的 Application Server 8.x 文件根目錄，預 設值為 <code>AppServer8Config-base/domains/domain1/docroot</code>

表 14-16 Access Manager 配置參數：amconfig（續）

參數	值
遷移自 AMConfig.properties	
SERVER_PROTOCOL	com.iplanet.am.server.protocol
SERVER_PORT	com.iplanet.am.server.port
SERVER_HOST	com.iplanet.am.server.host
DS_HOST	com.iplanet.am.directory.host
DS_PORT	com.iplanet.am.directory.port
ROOT_SUFFIX ²	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 ²	am.encryption.pwd ³

1. 如需有關範圍和舊有模式的更多資訊，請參閱第 245 頁的「相容性問題」。
2. 此參數的值應與之前版本的 Access Manager 相同。
3. 如果同時部署了 Access Manager 和 Access Manager SDK，則對於 Access Manager 實例及其相關的 Access Manager SDK 實例，此特性的值必須相同。

若為其他參數，除非要變更 Web 容器或密碼，否則請提供要升級的發行版本 2 配置中使用的那些值。

- d. 執行 amconfig 以取消部署 Access Manager。

將 *config-file* 中的 DEPLOY_LEVEL 值設定為 26。

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

- e. 檢查以確定 Common Agent Container 正在執行中。

```
netstat -an | grep 11163
```

如果並不在執行中，請啟動它。

```
/usr/sbin/cacoadm start
```

- f. 執行 amconfig 以重新配置 Access Manager 並將它部署到 Web 容器中。

將 *config-file* 中的 DEPLOY_LEVEL 值設定為 1。

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


9. 驗證 Access Manager classpath-suffix 和 server-classpath 資訊已遷移至發行版本 5 Application Server domain.xml 檔案中。

- a. 記錄 Access Manager classpath-suffix 和 server-classpath 資訊，它們位於原來部署 Access Manager 的發行版本 2 Application Server 實例的 server.xml 檔案中：

```
AppServer7Config-base/domains/domainName/instanceName/config/server.xml
```

- b. 檢查 classpath-suffix 和 server-classpath 項目皆已附加到部署 Access Manager 的升級 Application Server 實例的 domain.xml 檔案後面：

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/config/domain.xml
```

應將該類別路徑資訊增加到發行版本 5 Application Server domain.xml 檔案的 instanceName-config 區段中。此區段以下面這行開始：

```
<config dynamic-reconfiguration-enabled="true"
name="instanceName-config">
```

10. 更新目錄結構和模式。

發行版本 5 Access Manager 與發行版本 4 的目錄結構並存，但必須修改其結構才能支援發行版本 5 的功能。藉由執行 amupgrade 程序檔，將 Access Manager 目錄結構和模式更新到發行版本 5，該程序檔安裝在下列目錄中：

- 在 Solaris 上：
AccessManager-base/upgrade/scripts
 - 在 Linux 上：
AccessManager_base/identity/upgrade/scripts
- a. 取得 amupgrade 程序檔將會請求的下列參數的值：

表 14-17 Access Manager 配置參數：amupgrade

參數	值
Directory Server 主機	設定完全合格的名稱： <i>hostname.domain</i>
Directory Server 連接埠	指定非 SSL 連接埠號 ¹ 預設值：389
目錄管理員 DN	預設值：cn=Directory Manager
目錄管理員密碼	
頂層管理員 DN	預設值： uid=amadmin,ou=People,default_org_DN
頂層管理員密碼	

表 14-17 Access Manager 配置參數：amupgrade (續)

參數	值
啓用範圍模式 (當從發行版本 4 範圍模式升級時不會請求 此參數值。)	Y/N：是表示啓用範圍模式，並將服務資 料遷移至新的範圍樹狀結構。否 (預設值) 表示服務資料保留在舊有模式中。

1. 如果指定 Directory Server SSL 連接埠 (如預設 SSL 值 636)，升級程序將無法順利完成。

- b.** 執行 amupgrade 程序檔。

```
cd AccessManager-base/upgrade/scripts
./amupgrade
```

如果升級成功，程序檔會顯示 [升級已完成]。

- c.** 檢查下列升級記錄檔案，以瞭解關於目錄模式延伸的資訊：

在 Solaris 上：

```
/var/sadm/install/logs/
    Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm
```

在 Linux 上：

```
/var/log/Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm
```

- 11.** 停止 Domain Administration Server (DAS)、節點代理程式和伺服器實例。

這些都是在第 278 頁的步驟 7 中啓動的實例。

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID
    domainName
```

```
AppServer8-base/bin/asadmin stop-node-agent --user admin_ID
    nodeagentName
```

12. 重新啓動 Domain Administration Server (DAS)、節點代理程式和伺服器實例。

備註 啓動伺服器實例之前，請務必使用 `startinstances=false` 選項單獨啓動節點代理程式，如下所示。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
domainName
```

```
AppServer8-base/bin/asadmin start-node-agent --port DASportNumber
--startinstances=false --user admin_ID --password password
nodeagentName
```

```
AppServer8-base/bin/asadmin start-instance --port DASportNumber
--user admin_ID --password password instanceName
```

`DASportNumber` 的預設值爲 4848。

驗證 Access Manager 升級

完成升級程序之後，請按第 261 頁的「驗證 Access Manager 升級」所述驗證升級是否成功。

升級後作業

如果正在使用安全宣示標記語言 (SAML) 服務，則必須使用 Access Manager 主控台來增加和啓用 SAML 認證模組。如需有關建立 SAML 認證模組實例的資訊，請參閱「Sun Java System Access Manager 7.1 管理指南」(<http://docs.sun.com/doc/820-0840>)。

回復升級

未提供用於將 Access Manager 回復到其升級前狀態的程序檔。該程序必須使用作爲升級前作業之一而備份的 Access Manager 資料手動執行 (請參閱第 251 頁的「備份發行版本 4 Access Manager 的記錄和除錯檔案」)。回復太過困難，因此並不可行。

多重實例升級

在某些部署架構中，會將 Access Manager 部署在多個電腦系統上，以提供高可用性和延展性。

能夠循序升級 Access Manager 實例而不中斷服務往往是人們所希望的。執行從發行版本 2 Access Manager 到發行版本 5 漸進式升級的程序以下列限制為基礎：發行版本 5 Access Manager 不可以與發行版本 2 的目錄結構並存。但是，如果複製了 Directory Server 實例，如圖 14-2 所示，那麼您可以按第 268 頁的「[多重實例升級](#)」中的指示執行漸進式升級。

Portal Server

本章描述如何將 Portal Server 升級到 Java ES 5 (發行版本 5) : Sun Java System Portal Server 7.1。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 287 頁的「Portal Server 升級簡介」](#)
- [第 293 頁的「從 Java ES 發行版本 4 升級 Portal Server」](#)
- [第 320 頁的「從 Java ES 發行版本 3 升級 Portal Server」](#)
- [第 323 頁的「從 Java ES 發行版本 2 升級 Portal Server」](#)
- [第 333 頁的「從臨時功能發行版本 7.0 升級 Portal Server」](#)

備註 本章中的檔案位置是相對於名為 *PortalServer6-base* 和 *PortalServer6Config-base* (Portal Server 6.x) 以及 *PortalServer7-base* 和 *PortalServer7Config-base* (Portal Server 7.x) 的目錄路徑指定的。初始安裝 Portal Server 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 15-1 Portal Server 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>PortalServer6-base</i>	/opt/SUNWps	/opt/sun/portal
<i>PortalServer6Config-base</i>	/etc/opt/SUNWps	/etc/opt/sun/portal
<i>PortalServer7-base</i>	/opt/SUNWportal	/opt/sun/portal
<i>PortalServer7Config-base</i>	/etc/opt/SUNWportal	/etc/opt/sun/portal
<i>PortalServer7Data-base</i>	/var/opt/SUNWportal	/var/opt/sun/portal

Portal Server 升級簡介

本節描述下列會影響 Portal Server 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Portal Server](#)
- [Portal Server 升級資訊指南](#)
- [Portal Server 資料](#)
- [Portal Server 升級策略](#)

關於 Java ES 發行版本 5 Portal Server

Java ES 發行版本 5 Portal Server 相對於發行版本 4 而言為主要發行版本，包含許多新功能與增強功能。許多變更都是在發行版本 4 隨後的「臨時功能發行版本」(IFR) 中進行的。相對於 IFR 而言，發行版本 5 僅為次要的功能變更。有關 IFR 增強功能及新功能的資訊，請參閱「Sun Java System Portal Server 7.1 版本說明」(<http://docs.sun.com/app/docs/doc/820-0898/6ncjuat45?a=view>)。特別是，發行版本 4 命令行管理介面已由 psadmin 指令所取代。

Portal Server 升級資訊指南

表 15-2 顯示支援的 Portal Server 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 15-2 至 Java ES 5 (發行版本 5) 的升級路徑：Portal Server 7.1

Java ES 發行版本	Portal Server 版本	一般方法	需要重新配置
臨時功能發行版本 (IFR)	Sun Java System Portal Server IFR 7.0 2005Q4	直接升級： 藉由套用修補程式然後使用升級程序檔執行。	需要手動重新套用自訂。
發行版本 4	Sun Java System Portal Server 6.3.1 2005Q4	直接升級： 使用升級程序檔執行。	需要手動重新套用自訂。
發行版本 3	Sun Java System Portal Server 6.3.1 2005Q1	直接升級： 使用升級程序檔執行。	需要手動重新套用自訂。
發行版本 2	Sun Java System Portal Server 6.3 2004Q2	直接升級： 使用升級程序檔執行。	需要手動重新套用自訂。

表 15-2 至 Java ES 5 (發行版本 5) 的升級路徑：Portal Server 7.1 (續)

Java ES 發行版本	Portal Server 版本	一般方法	需要重新配置
發行版本 1	Sun ONE Portal Server 6.2 (2003Q4)	無直接升級： 但可以藉由先升級到發行版本 3，然後再從發行版本 3 升級到發行版本 5 來執行。	配置資料
Java ES 發行版本之前的發行版本		無直接升級。	

Portal Server 資料

下表顯示可能受 Portal Server 軟體升級影響的資料類型。

表 15-3 Portal Server 資料用途

資料類型	位置	用途
配置資料	<i>PortalServer6Config-base/</i>	Portal Server 的配置。
Web 容器存取控制和配置檔案	<p>Web Server 7.0 (Java ES 發行版本 5) server.policy 和 server.xml 檔案，位於 <i>WebServer7Config-base/https-configName/config</i> 中</p> <p>Web Server 6.x (Java ES 發行版本 2、3 和 4) server.policy 和 server.xml 檔案，位於 <i>WebServer6-base/https-hostname/config</i> 中</p> <p>Application Server 8.x (Java ES 發行版本 3、4 和 5)： server.policy 和 domain.xml 檔案，位於 <i>AppServer8Config-base/domains/domainName/config</i> 中</p> <p>Application Server 7.x (Java ES 發行版本 2)： server.policy 和 server.xml 檔案，位於 <i>AppServer7Config-base/domains/domainName/config</i> 中</p>	Portal Server Web 容器實例的配置。
自訂資料	<i>PortalServer6Config-base/desktop</i>	<p>自訂模組的 JAR 檔案</p> <p>自訂的範例 Portal Server 桌面</p>
目錄模式 服務配置 使用者資料	Directory Server	Portal Server 依賴服務配置 (如入口桌面) 和儲存在目錄中的使用者設定檔資料。

表 15-3 Portal Server 資料用途 (續)

資料類型	位置	用途
動態應用程式資料	無	Portal Server 不會永久性地儲存應用程式資料 (如階段作業狀態)。

Portal Server 升級策略

Portal Server 的升級策略一般取決於在第 1 章「升級規劃」中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Portal Server 升級規劃的問題，藉以對 Portal Server 進行一般性的介紹。

相容性問題

發行版本 5 Portal Server 介紹了用來管理 Portal Server 與 Portal Server Secure Remote Access 元件之 psadmin 指令中的公共介面變更。請參閱「Sun Java System Portal Server 7.1 Command-Line Reference」(<http://docs.sun.com/doc/819-5030>)。

因此，發行版本 5 Portal Server 並不向下相容較早版本或 Portal Server Secure Remote Access 元件 (包括 SRA Gateway、Rewriter Proxy 和 Netlet Proxy) 的較早版本，但多重實例部署正在執行漸進式升級的過渡期除外。在 Java ES 發行版本 5 中，所有 Portal Server 實例以及 Portal Server Secure Remote Access 元件實例都需要同步。

另外，個別 Portal Server 元件 (包括行動存取元件) 並不向下相容較早的版本；所有元件都需要與 Java ES 發行版本 5 同步。

此外，發行版本 5 Portal Server 與較早 Portal Server 版本所使用的 Directory Server 資料架構之間存在不相容性。此不相容性會影響使用相同 Directory Server 資料之多重 Portal Server 實例執行的漸進式升級。

Portal Server 相依性

Portal Server 對其他 Java ES 元件的相依性可能會影響升級和重新配置 Portal Server 軟體的程序。例如，Portal Server 介面或功能上的變更可能需要 Portal Server 依賴之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Portal Server 對下列 Java ES 元件有相依性：

- **共用元件。** Portal Server 對特定的 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。
- **Web 容器。** Portal Server 對 Web 容器服務具有強制性的相依性，它們可由 Java ES Web Server、Java ES Application Server 或來自 Weblogic 和 WebSphere 的協力廠商 Web 容器提供。

備註	對於協力廠商 Web 容器中的部署，不支援將 Portal Server 升級到發行版本 5。對於來自 Weblogic 和 WebSphere 的 Web 容器中的部署，您必須對發行版本 5 Portal Server 執行全新安裝。
-----------	--

- **Access Manager (或 Access Manager SDK)。** Portal Server 對 Access Manager 有強制性的相依性，可為一般使用者提供認證和授權服務，包括單次登入。如果 Access Manager 是在遠端電腦上執行，則 Access Manager SDK 在本機上必須可用。
- **Directory Server。** Portal Server 對 Directory Server 有強制性的相依性，後者用於儲存藉由 Access Manager 存取的使用者資料。因此，Portal Server 升級可能需要延伸目錄模式。
- **Portal Server Secure Remote Access。** Portal Server 對 Portal Server Secure Remote Access 有選擇性的相依性，後者可透過 Gateway、Rewriter Proxy 和 Netlet Proxy 元件提供安全遠端存取。
- **Java DB。** Portal Server 對 Java DB 有選擇性的相依性，後者用於提供對多個 portlet 應用程式的支援。
- **Service Registry。** Portal Server 對 Service Registry 有強制性的相依性，後者用於提供編譯所需的程式庫。
- **Communications Express。** Portal Server 對 Communications Express (Sun Java Communications Suite 元件) 有選擇性的相依性，後者用於向一般使用者提供傳訊及行事曆通道。Communications Express 不再是 Java ES 產品元件。

選擇性的升級問題

一般而言，雖然 Java ES 發行版本 5 支援在電腦上對於所有元件的選擇性升級，但由於 Portal Server 對如此多的其他 Java ES 元件都具有相依性，因此要認證各種 Java ES 發行版本之間元件的任意組合是非常困難的。

因此，Portal Server 支援與 Access Manager 及 Web 容器相關的一組有限制的升級方案。

- **如果您要從 Java ES 發行版本 4 升級 Portal Server。** 您可以在升級 Portal Server 之前先將 Directory Server、Access Manager 及 Web 容器 (Web Server 或 Application Server) 升級到發行版本 5，或者您也可以只將 Portal Server 升級到發行版本 5 (讓其他元件保留在其發行版本 4 層級)，但您無法使一些相依元件保留在發行版本 4 而將其他相依元件升級到發行版本 5。
- **如果您要從 Java ES 發行版本 3 升級 Portal Server。** 在升級 Portal Server 之前，您必須將 Directory Server、Access Manager 和 Web 容器 (Web Server 或 Application Server) 升級到發行版本 4 或發行版本 5，但您無法使任何相依元件保留在發行版本 3，也無法將一些相依元件升級到發行版本 4 而將其他相依元件升級到發行版本 5。
- **如果您要從 Java ES 發行版本 2 升級 Portal Server。** 在升級 Portal Server 之前，必須先將 Directory Server、Access Manager 以及 Web 容器 (Web Server 或 Application Server) 升級到發行版本 4 或發行版本 5。既不能留下發行版本 2 上的任何相依元件，也不能將一些相依元件升級到發行版本 4，而將其他相依元件升級到發行版本 5。

Web 容器升級方案

Portal Server 可部署於 Web Server 或 Application Server 提供的 Web 容器中。這樣，其所在的 Web 容器可能也需要升級到發行版本 5，使得 Portal Server 的升級更加複雜。就這一點而言，有一些可能的 Web 容器升級方案，列舉於下表中。

表 15-4 Portal Server 升級的 Web 容器升級方案

方案	最初部署 Portal Server 的 Web 容器	升級之後部署 Portal Server 的 Web 容器	適用 Portal Server 升級路徑：升級始於
方案 1	Web Server 6.x	Web Server 6.x	發行版本 2 發行版本 3 發行版本 4 IFR 7.0
方案 2	Web Server 6.x	Web Server 7.0	發行版本 2 發行版本 3 發行版本 4

表 15-4 Portal Server 升級的 Web 容器升級方案 (續)

方案	最初部署 Portal Server 的 Web 容器	升級之後部署 Portal Server 的 Web 容器	適用 Portal Server 升級路徑：升級始於
方案 3	Application Server 8.1	Application Server 8.1	發行版本 3 發行版本 4 IFR 7.0
方案 4	Application Server 8.1	Application Server 8.2	發行版本 3 發行版本 4 IFR 7.0
方案 5	Application Server 7x	Application Server 8.2	發行版本 2

升級 Portal Server (例如使用 `psupgrade` 程序檔時) 來為表 15-4 中的合適升級方案提供適當值時，特別是要進行 Web 容器的重大版本升級時，必須很小心。

雙重升級

在雙重升級中，Portal Server 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用原地升級作業系統的方法來執行：

1. 備份現有的 Portal Server 資料。
如需必要資料的位置，請參閱第 288 頁的「[Portal Server 資料](#)」。
2. 升級作業系統。
升級時會原樣保留現有的檔案系統。
3. 升級到發行版本 5 Portal Server。
請根據升級路徑，參閱本章的相應小節。

從 Java ES 發行版本 4 升級 Portal Server

本節包括將 Portal Server 從 Java ES 2005Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。

備註 本節不適用於以下特殊情況：Portal Server 部署在 Application Server Web 容器中，且是先從發行版本 2 升級到發行版本 3 或 4，然後再升級到發行版本 5。目前尚不支援前述的升級路徑。

本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Portal Server](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Portal Server 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 使用升級程序檔 psupgrade 來執行升級。此程序檔會安裝新的套裝軟體、在必要時遷移配置資料、更新本土化檔案及將 Portal Server Web 應用程式重新部署至 Web 容器。
- **升級相依性。** Portal Server 對一些 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。雖然發行版本 5 Portal Server 與這些共用元件的發行版本 4 版本相容，但共用元件的升級仍然是必要的，這是因為用來升級 Portal Server 的 psupgrade 程序檔需要使用 ANT 共用元件的發行版本 5 版本。

發行版本 5 Portal Server 還依賴 Web 容器、Access Manager 和 Directory Server，如第 290 頁的「Portal Server 相依性」中所述。支援升級這些相依性的兩種方法 (請參閱第 291 頁的「選擇性的升級問題」)：

- 發行版本 4 元件符合的所有相依性 (沒有任何一項會升級到發行版本 5)。
- 發行版本 5 元件符合的所有相依性 (所有項目都會升級到發行版本 5)。
- **向下相容性。** 發行版本 5 Portal Server 不向下相容發行版本 4 版本。
- **升級回復。** 將 Portal Server 的發行版本 5 升級回復到發行版本 4 包括復原發行版本 4 套裝軟體、復原發行版本 4 目錄資料以及將 Portal Server Web 應用程式重新部署至 Web 容器。

- **平台問題。** 升級 Portal Server 的一般方法在 Solaris 及 Linux 作業系統上均相同，但在 Solaris OS 上，發行版本 5 Portal Server 安裝在新路徑中，而在 Linux OS 上的安裝路徑則與發行版本 4 相同。

升級發行版本 4 Portal Server

本節描述如何在 Solaris 及 Linux 平台上將 Portal Server 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [發行版本 4 升級前作業](#)
- [升級發行版本 4 Portal Server \(Solaris\)](#)
- [升級發行版本 4 Portal Server \(Linux\)](#)
- [驗證升級](#)
- [發行版本 4 升級後作業](#)
- [回復升級 \(Solaris\)](#)
- [回復升級 \(Linux\)](#)

發行版本 4 升級前作業

升級 Portal Server 之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Portal Server 相依元件](#)
- [取得必要的配置資訊和密碼](#)
- [備份 Portal Server 發行版本 4 配置資訊](#)
- [記錄 Java 虛擬機器 \(JVM\) 設定](#)
- [移除負載平衡器的配置](#)
- [移除 Directory Proxy Server 配置](#)

驗證目前版本資訊

您可使用下列指令驗證 Portal Server 的目前版本：

```
PortalServer6-base/bin/version
```

表 15-5 Portal Server 版本驗證輸出

Java ES 發行版本	Portal Server 版本編號
發行版本 2	6.3
發行版本 3	6.3.1
發行版本 4	6.3.1 ¹
IFR 發行版本	7.0
發行版本 5	7.1

1. 發行版本 3 與發行版本 4 的唯一差別在於修補程式。您可使用 `Solaris showrev -p | grep patch_ID` 指令和 `Linux rpm -qa sun-portal-core` 指令，並將版本與「Java ES 發行版本 4 升級指南」中列出的版本進行比較，來檢查發行版本 4 修補程式。

升級 Portal Server 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。

雖然發行版本 5 Portal Server 與 Java ES 共用元件的發行版本 4 版本相容，但共用元件的升級仍然是必要的，這是因為用來升級 Portal Server 的 `psupgrade` 程序檔需要使用 ANT 共用元件的發行版本 5 版本。

如果您選擇將任何 Portal Server 產品元件相依元件升級到發行版本 5，則它們**全部**都需要升級 (請參閱第 291 頁的「選擇性的升級問題」)。在升級 Portal Server 之前，您應該以下列順序升級相依元件 (略過可能已經升級過的任何相依元件)。

1. **共用元件**。在第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。
2. **Directory Server**。在第 93 頁的第 5 章「Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。

3. **Web 容器軟體**。在第 123 頁的第 7 章「Web Server」和第 189 頁的第 11 章「Application Server」中分別提供了升級 Web Server 和 Application Server 的指示。

備註 升級協力廠商 Web 容器 (如來自 Weblogic 和 WebSphere 的容器) 可能導致 Portal Server 發生故障，因為容器升級會覆寫為支援 Portal Server 而對這些容器進行的自訂。

 在這些情況下，必須為升級後的 Web 容器環境重新安裝和重新配置 Portal Server。

4. **Access Manager (Access Manager SDK)**。在第 241 頁的第 14 章「Access Manager」中提供了將 Access Manager 升級到發行版本 5 的指示。
5. **Portal Server Secure Remote Access**。在第 351 頁的第 16 章「Portal Server Secure Remote Access」中提供了將 Portal Server Secure Remote Access 升級到發行版本 5 的指示。
6. **Java DB**。在第 145 頁的第 8 章「Java DB」中提供了將 Java DB 升級到發行版本 5 的指示。
7. **Service Registry**。在第 215 頁的第 12 章「Service Registry」中提供了將 Service Registry 升級到發行版本 5 的指示。
8. **Communications Express**。在「Sun Java Communications Suite Upgrade Guide」(<http://docs.sun.com/doc/819-7561>) 中提供了將 Communications Express 升級到發行版本 5 的指示。

取得必要的配置資訊和密碼

視 Web 容器升級方案而定 (請參閱第 291 頁的表 15-4)，psupgrade 程序檔需要您輸入密碼及其他 Web 容器配置資料相關資訊。不同 Web 容器升級方案需要的資訊如表 15-6 中所述。在開始升級 Portal Server 之前，請務必將相關資訊加以組合。

表 15-6 各個 Web 容器升級方案中 psupgrade 程序檔所需的資訊

資訊	升級方案 ³	Web Server 7.x 範例值： 方案 2	Application Server 8.x 範例值：方案 5 ²
在 Web Server 7.0 上升級 Portal Server (是/否)	2	是	不適用
Web 容器安裝目錄	2 和 5	WebServer7-base	AppServer8Install-base
Web 容器虛擬伺服器實例名稱	2	https-configName ⁴	不適用
Web 容器實例名稱	5	不適用	server1

表 15-6 各個 Web 容器升級方案中 psupgrade 程序檔所需的資訊 (續)

資訊	升級方案 ³	Web Server 7.x 範例值： 方案 2	Application Server 8.x 範例值：方案 5 ²
Web 容器實例目錄	2	<i>WebServer7Config-base/ https-configName⁴ /</i>	不適用
Portal 實例部署目錄	5	不適用	<i>AppServer8Config-base/ domains/domainName</i>
Web 容器實例連接埠	2 和 5	80	80
Web 容器實例協定	2 和 5	http	http
Web 容器配置名稱	2	<i>configName⁴</i>	不適用
Web 容器網域名稱	5	不適用	domain1
Web 容器文件根目錄	2 和 5	<i>WebServer7Config-base/ https-configName⁴ / docs /</i>	<i>AppServer8Config-base/ domains/domainName/ docroot</i>
Web 容器管理主機名稱	2 和 5	localhost	localhost
Web 容器管理連接埠	2 和 5	8989	4848
Web 容器管理協定	2 和 5	https	https
Web 容器管理使用者 ID	2 和 5	admin	admin
Web 容器管理密碼	2 到 5		
Web 容器主密碼	3 到 5	不適用	
目錄管理員 (cn=Directory manager) 密碼	1 到 5		
SRA 記錄使用者密碼 ¹	1 到 5		
Access Manager 管理員密碼	1 到 5		
Directory Server ldapuser 密碼	1 到 5		
入口網站實例 ID ²	1 到 5		

1. 當 Portal Server Secure Remote Access 元件隨 Portal Server 安裝時，需要此資訊來對這些元件進行配置。

2. 必須為此參數提供唯一的非空值。值必須由字母數字組成，而且可以包含連字號 (-)。

3. Web 容器升級方案 5 適用於從發行版本 2 升級 Portal Server。

4. *configName* 的預設值是 *hostName.domainName*。

備份 Portal Server 發行版本 4 配置資訊

將 Portal Server 升級到發行版本 5 不需要重新配置 Portal Server 軟體。不過，為安全起見，psupgrade 程序檔將會備份下列儲存配置資訊的目錄：

PortalServer6Config-base/

備註 建議備份 Portal Server 用來儲存使用者設定檔及其他資料的 Directory Server 實例。如果沒有此類資料，便無法對目標為發行版本 5 Portal Server 的升級進行回復。

記錄 Java 虛擬機器 (JVM) 設定

在升級 Portal Server 之前，如果下列 Web 容器 JVM 設定與預設值不同，請記錄下來。

```
<jvm-options>-XX:MaxPermSize=256m</jvm-options>
<jvm-options>-XX:+CMSPermGenSweepingEnabled</jvm-options>
<jvm-options>-XX:+CMSClassUnloadingEnabled</jvm-options>
```

JVM 設定的位置視 Web 容器而定，如下表中所述。

表 15-7 JVM 設定的位置

Web 容器	配置檔案
發行版本 2、3、4 Web Server (6.x)	<i>WebServer6-base/https-instanceName/config/server.xml</i>
發行版本 5 Web Server (7.0)	<i>WebServer7Config-base/https-configName/config/server.xml</i>
發行版本 3、4、5 Application Server (8.x)	<i>AppServer8Config-base/domains/domainName/config/domain.xml</i>
由節點代理程式管理的 Application Server 實例	<i>AppServer8Config-base/nodeagents/nodeagentName/config/domain.xml</i>

稍後，您將需要檢查這些 JVM 設定是否因為 Portal Server 升級程序而變更。

移除負載平衡器的配置

如果遇到透過負載平衡器存取 Portal Server 實例的情況，用來配置此類存取的 `LOAD_BALANCER_URL` 特性的值會干擾 Portal Server 升級。因此在執行升級之前，必須修改此設定。若要修改 `LOAD_BALANCER_URL` 特性設定：

1. 請注意下列哪些配置檔案位於本機上 (其中一些支援可以在本機安裝的 Portal Server Secure Remote Access 元件)：

PortalServer6Config-base/PSConfig.properties

PortalServer6Config-base/GWConfig.properties (如果 Gateway 在本機)

PortalServer6Config-base/RWPConfig.properties (如果 Rewriter Proxy 在本機)

PortalServer6Config-base/NLPConfig.properties (如果 Netlet Proxy 在本機)

2. 將 `LOAD_BALANCER_URL` 特性的目前值記錄在這些配置檔案中。
3. 將 `LOAD_BALANCER_URL` 特性的值修改為指向相關 Portal Server 實例：

`LOAD_BALANCER_URL=portalHostName:port/portal`

移除 Directory Proxy Server 配置

如果遇到 Portal Server 實例透過 Directory Proxy Server 實例存取 Directory Server 的情況，則在執行升級之前必須修改 Directory Proxy Server 主機與連接埠號設定，然後再在升級完成之後復原為原始值。

修改適當設定：

1. 將 `DS_HOST` 和 `DS_PORT` 特性的目前值記錄在下列 Access Manager 配置檔案中：

AccessManagerConfig-base/config/AMConfig.properties

2. 修改 `DS_HOST` 和 `DS_PORT` 特性的值，以直接指向相關 Directory Server 實例。

升級發行版本 4 Portal Server (Solaris)

本節討論影響 Portal Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項 (Solaris)

將 Portal Server 軟體升級到發行版本 5 時要考慮下列注意事項：

- 同時升級對應於同一安裝的 Portal Server 影像的所有 Portal Server 實例。
- Portal Server 軟體包括執行一些不同角色的子元件，但是全部都一起升級：
 - **Portal-base**。包括管理 Mbeans 及隨附的管理軟體、登入架構及與監視相關的軟體，它們全部都套裝在一起。

- **Portal Server Web 應用程式**。包括一些在 Web 容器中部署的 Web 應用程式。至少這其中的一些 Web 應用程式需要來自 Access Manager 的支援，以及接下來是來自 Directory Server 的支援。
- **Secure Remote Access core**。支援 Portal Server Secure Remote Access 的軟體：內嵌在 Jar 檔案中的一些 servlet 和 applet 以及無法在 Web 容器中部署的一些支援檔案。
- `psupgrade` 程序檔會自動偵測主機上安裝了哪些 Portal Server 子元件以及哪些 Web 容器相依元件。例如，程序檔會查詢系統以偵測您將在其中部署 Portal Server Web 應用程式的 Application Server 或 Web Server 的版本，並會視它可以偵測的資訊來修改它請求的資訊。

升級程序 (Solaris)

下述程序適用於位於執行升級的電腦上的 Portal Server。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。

[第 59 頁的第 2 章「升級 Java ES 共用元件」](#)中提供了指示。

對於在[第 301 頁的步驟 8](#)中執行 `psupgrade` 程序檔來說，此步驟是必要的先決條件。

3. 停止可能在本地執行的 Portal Server Secure Remote Access Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer6-base/bin gateway stop  
PortalServer6-base/bin netletd stop  
PortalServer6-base/bin rwproxyd stop
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443  
Rewriter Proxy : netstat -an | grep 10443  
Netlet Proxy : netstat -an | grep 10555
```

4. 如果 Access Manager 與 Portal Server 分別部署在不同的 Web 容器中，請確定 Access Manager 正在執行。

5. 如果未在執行，請啟動 Portal Server，這可藉由啟動其部署到的 Web 容器來實現。

Web Server 6.x :
`WebServer-base/https-instanceName/start`

Web Server 7.0 :
 Admin Server--
`WebServer7Config-base/admin-server/bin/startserv`
 Instance Server--
`WebServer7Config-base/https-configName/bin/startserv`

Application Server 8.x :
`AppServer8-base/bin/asadmin start-domain --user admin_ID
 --password password domainName`

6. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 和 JAVA_HOME)。例如，

```
export ANT_HOME=/usr/sfw
export JAVA_HOME=/usr/jdk/entsys-j2se
```

7. 請確定您電腦上有足夠的交換空間。

做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。

8. 從 Java ES 發行版本 5 發行物中執行 psupgrade 程序檔。

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

其中 `os_arch` 與您所在平台相符，例如 `Solaris_sparc`。

備註

如果您不慎在錯誤的 `os_arch` 目錄中執行 psupgrade，您需要取消下述程序：

- a. 變更為正確的 `os_arch` 目錄。
 - b. 回復對於 Portal Server 資料的變更。
`./psupgrade rollback`
 提供要求的參數與密碼。
 - c. 再次執行 psupgrade。
-

psupgrade 程序檔會偵測已安裝的 Portal Server 元件及本土化套裝軟體，呼叫 Java ES 安裝程式以安裝新的套裝軟體，並查詢系統以偵測關於您將在其中部署 Portal Server Web 應用程式的 Web 容器之位置與連接埠號及其他資訊。視 Web 容器升級方案而定 (請參閱第 291 頁的表 15-4)，程序檔會要求您輸入將 Portal Server 部署到適當 Web 容器中所需要的其他資訊。

第 296 頁的表 15-6 顯示了表 15-4 中不同 Web 容器升級方案所要求的資訊。

備註 請務必為 `psupgrade` 參數輸入正確的值，因為您無法返回及變更它們，而且也很難回復由 `psupgrade` 程序檔所做的變更。若要回復對於 Portal Server 資料的變更，您必須執行

```
./psupgrade rollback
```

然後嘗試再次執行 `psupgrade`。

9. 如有必要，請復原 Web 容器 JVM 設定。

若要確定 JVM 設定支援發行版本 5 Portal Server，請執行以下步驟：

- a. 檢查您在升級前記錄的 Portal Server Web 容器 JVM 設定是否因升級程序而變更。

請參閱第 298 頁的「記錄 Java 虛擬機器 (JVM) 設定」。

- b. 如果設定已經變更，請將它們回復為您在升級前記錄的值。

請確定包括以下 JVM 設定 (即使之前未設定)：

```
<jvm-options>-XX:MaxPermSize=256m</jvm-options>  
<jvm-options>-XX:+CMSPermGenSweepingEnabled</jvm-options>  
<jvm-options>-XX:+CMSClassUnloadingEnabled</jvm-options>
```

10. 停止 Web 容器，然後將其重新啟動。

儘管並不需要在所有情況下都重新啟動 Web 容器，但這樣做可確保 Portal Server 在未修改的狀態下啟動。

- a. 停止 Web 容器，如下所示：

Web Server 6.x：
`WebServer-base/https-instanceName/stop`

Web Server 7.0：
Admin Server--
`WebServer7Config-base/admin-server/bin/stopserv`
Instance Server--
`WebServer7Config-base/https-configName/bin/stopserv`

Application Server 8.x：
`AppServer8-base/bin/asadmin stop-domain --user admin_ID
--password password domainName`

- b. 使用第 301 頁的步驟 5 中的指令重新啟動 Web 容器。

升級發行版本 4 Portal Server (Linux)

本節討論影響 Portal Server 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項 (Linux)

在 Linux OS 上，發行版本 5 Portal Server 的安裝路徑與發行版本 4 相同，除了這一點外，在 Linux 平台和 Solaris 平台上將 Portal Server 軟體升級到發行版本 5 所考慮的注意事項相同 (請參閱第 299 頁的「升級注意事項 (Solaris)」)。因此，當安裝發行版本 5 RPM 時，psupgrade 程序檔會移除之前的 RPM。

升級程序 (Linux)

下述程序適用於位於執行升級的電腦上的 Portal Server。

注意 在 Linux 上執行的從 Java ES 發行版本 4 至發行版本 5 的升級無法回復。在執行下列程序之前，請務必先備份您的系統。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。

第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了指示。

對於在第 304 頁的步驟 8 中執行 psupgrade 程序檔來說，此步驟是必要的先決條件。

3. 停止可能在本地執行的 Portal Server Secure Remote Access Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxyd stop
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
Rewriter Proxy : netstat -an | grep 10443
Netlet Proxy : netstat -an | grep 10555
```

4. 如果 Access Manager 與 Portal Server 分別部署在不同的 Web 容器中，請確定 Access Manager 正在執行。

5. 如果未在執行，請啓動 Portal Server，這可藉由啓動其部署到的 Web 容器來實現。

Web Server 6.x :
`WebServer-base/https-instanceName/start`

Web Server 7.0 :
Admin Server--
`WebServer7Config-base/admin-server/bin/startserv`
Instance Server--
`WebServer7Config-base/https-configName/bin/startserv`

Application Server 8.x :
`AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName`

6. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 和 JAVA_HOME)。例如，

```
export ANT_HOME=/opt/sun  
export JAVA_HOME=/usr/jdk/entsys-j2se
```

7. 請確定您電腦上有足夠的交換空間。

做爲指導原則，應將交換空間設定爲實體 RAM 容量的兩倍。

8. 從 Java ES 發行版本 5 發行物中執行 psupgrade 程序檔。

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin  
./psupgrade
```

其中 `os_arch` 與您所在平台相符，例如 `Linux_x86`。

psupgrade 程序檔會偵測已安裝的 Portal Server 元件及本土化套裝軟體，呼叫 Java ES 安裝程式以安裝新的套裝軟體，並查詢系統以偵測關於您將在其中部署 Portal Server Web 應用程式的 Web 容器之位置與連接埠號及其他資訊。視 Web 容器升級方案而定 (請參閱第 291 頁的表 15-4)，程序檔會要求您輸入將 Portal Server 部署到適當 Web 容器中所需要的其他資訊。

第 296 頁的表 15-6 顯示了表 15-4 中不同 Web 容器升級方案所要求的資訊。

備註 請務必爲 psupgrade 參數輸入正確的值，因爲您無法返回及變更它們，而且也很難回復由 psupgrade 程序檔所做的變更。提示：在執行 psupgrade 程序檔之前，請備份您的系統。

9. 修改 *PortalServer7Config-base/platform.conf.default* 配置檔案。

從下面的檔案中複製包含 `gateway.logging.password` 的一行，該檔案是由 `psupgrade` 備份的：

```
PortalServer6Config-base.bak/platform.conf.default
```

並將該行置於 *PortalServer7Config-base/platform.conf.default* 中。

10. 如有必要，請復原 Web 容器 JVM 設定。

若要確定 JVM 設定支援發行版本 5 Portal Server，請執行以下步驟：

- a. 檢查您在升級前記錄的 Portal Server Web 容器 JVM 設定是否因升級程序而變更。

請參閱第 298 頁的「記錄 Java 虛擬機器 (JVM) 設定」。

- b. 如果設定已經變更，請將它們回復為您在升級前記錄的值。

請確定包括以下 JVM 設定 (即使之前未設定)：

```
<jvm-options>-XX:MaxPermSize=256m</jvm-options>
<jvm-options>-XX:+CMSPermGenSweepingEnabled</jvm-options>
<jvm-options>-XX:+CMSClassUnloadingEnabled</jvm-options>
```

11. 停止 Web 容器，然後將其重新啟動。

儘管並不需要在所有情況下都重新啟動 Web 容器，但這樣做可確保 Portal Server 在未修改的狀態下啟動。

- a. 停止 Web 容器，如下所示：

Web Server 6.x :

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

Web Server 7.0 :

Admin Server--

```
WebServer7Config-base/admin-server/bin/stopserv
```

Instance Server--

```
WebServer7Config-base/https-configName/bin/stopserv
```

Application Server 8.x :

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID
  --password password domainName
```

- b. 使用第 304 頁的步驟 5 中的指令重新啟動 Web 容器。

驗證升級

您可使用下列指令驗證發行版本 5 套裝軟體的安裝：

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID  
-f adminpasswordfile.
```

請參閱第 295 頁的表 15-5 中的輸出值。

若要驗證完整升級，請確認 Portal Desktop 會顯示出來，且 psadmin 管理公用程式也可以如文件所述地使用。

您也可以檢查以下升級記錄檔：

```
/var/sadm/install/logs/Sun_Java_System_Portal_Server_upgrade.log*  
PortalServer7Data-base/logs/admin/  
PortalServer7Data-base/logs/config/
```

發行版本 4 升級後作業

請注意處理以下情況所需要的升級後程序：

- [遷移自訂 web-src 資料](#)
- [重新部署自訂 Portlet 應用程式](#)
- [遷移自訂的 Portlet 應用程式](#)
- [更正書籤及應用程式通道中的連結](#)
- [正確存取搜尋伺服器](#)
- [復原 Directory Proxy Server 配置](#)
- [手動註冊 Portal Server Secure Remote Access 元件](#)
- [啟用 URLScrapper 通道](#)
- [登出頁面中的變更](#)

遷移自訂 web-src 資料

如果您已經增加了自訂資料 (例如影像、javascript 檔案或其他任何檔案) 用來將 portal.war 建構到以下目錄中：

```
PortalServer6-base/web-src
```

您必須將這些額外的檔案複製到發行版本 5 Portal Server 中的對應目錄中：

```
PortalServer7-base/web-src
```

重新部署自訂 Portlet 應用程式

如果您已經建立並部署了自訂 portlet 應用程式，則必須在升級到發行版本 5 Portal Server 之後手動重新部署這些 portlet。即使顯示設定檔項目將會存在且通道名稱將會顯示，也只能在重新部署完自訂 portlet 之後才會看到內容。

使用下列指令重新部署 portlet：

```
PortalServer7-base/bin/psadmin deploy-portlet
```

您可以在以下位置尋找對應 .war 及 XML 檔案來確認重新部署：

```
PortalServer7Data-base/portals/Upgraded/war
```

遷移自訂的 Portlet 應用程式

以 Sun Java Web Console (SJWC) 所提供之使用者介面架構為基礎的 portlet 應用程式需要手動遷移至發行版本 5 並重新部署。

特別的是，此需求適用於四個 Web 應用程式，這四個應用程式做為要自訂及安裝到入口網站的範例 portlet 應用程式隨 Portal Server 一起分佈，它們分別是：

filesharing、surveys、wiki、rssportlet。在升級期間，會將這些範例 portlet 應用程式修正錯誤後的版本置於 portlet 應用程式區域中的磁碟上。如果您已經自訂了這些 portlet 應用程式供自己使用，則需要手動將它們遷移至發行版本 5 並重新進行部署；Portal Server 升級程序不會處理它們。

依預設，會部署其中一些 portlet 應用程式 (filesharing 和 surveys)，而且當建立社群時，它們可供使用者使用。

您可以使用下列程序升級、自訂及重新部署基於 SJWC 的 portlet 應用程式。filesharing portlet 應用程式做為範例使用：

1. 解壓縮發行版本 5 SJWC jar 檔案。
 - a. mkdir /tmp/lh
 - b. cd /tmp/lh
 - c. /usr/jdk/entsys-j2se/bin/jar xvf


```
PortalServer7-base/portlet/communityportlets.war
WEB-INF/lib/commons-beanutils.jar
WEB-INF/lib/commons-collections-3.1.jar
WEB-INF/lib/commons-digester.jar
WEB-INF/lib/commons-logging.jar
WEB-INF/lib/dataprovider.jar WEB-INF/lib/jsf-api.jar
WEB-INF/lib/jsf-impl.jar WEB-INF/lib/webui.jar
```

- d. 重新命名其中一個檔案。

```
mv WEB-INF/lib/commons-collections-3.1.jar  
WEB-INF/lib/commons-collections.jar
```

2. 找到 filesharing portlet 應用程式。

```
cd PortalServer7Config-base/portals/portall/portletapps/filesharing
```

3. 加入經過更新的 SJWC 程式庫。

```
jar uvf src/filesharing.war.tokenized -C /tmp/lh WEB-INF
```

4. 自訂 filesharing portlet 應用程式。

```
ant customize
```

5. 重新部署 filesharing portlet 應用程式。

- a. `PortalServer7-base/bin/psadmin undeploy-portlet -u amadmin -f passwordfile -p portal_id -i instance_id -g filesharing`

- b. `ant deploy`

- c. 請至下列目錄 (視 Web 容器而定)：

Application Server 8.x :

```
AppServer8Config-base/domains/domain1/applications/j2ee-modules/  
communityportlets/WEB-INF
```

Web Server 6.x :

```
WebServer6-base/https-instanceName/webapps/https-instanceName/  
communityportlets/WEB-INF
```

Web Server 7.x :

```
WebServer7Config-base/https-configName/web-app/https-configName/  
communityportlets/WEB-INF
```

- d. 開啓 sun-web.xml 檔案，並在最後一行之前 (也就是在 sun-web-app 結束標記之前) 加上以下這一行：

```
<class-loader delegate="false"/>
```

- e. 針對 filesharing/WEB-INF 底下的 sun-web.xml 檔案重複步驟 c 和步驟 d。

6. 針對 surveys 以及基於 SJWC 架構的其他任何自訂 portlet 應用程式重複步驟 2 到步驟 5。

7. 重新啓動 Web 容器。

更正書籤及應用程式通道中的連結

將發行版本 4 Portal Server 升級到發行版本 5 時，書籤及應用程式通道含有重複和虛假的連結。若要修正這些連結，請執行下列程序。

1. 登入 PSConsole。
2. 從 [一般作業] 標籤中，按一下 [管理通道與容器]。
3. 選擇 [DeveloperSample [Org]] 做為 DN，然後按一下 [確定]。
4. 選取 [JSPTabContainer [預設]] 做為 [檢視類型]。
5. 在 [MyFrontPageTabPanelContainer] 下，按一下 [應用程式通道]。

將會在右側的框架上顯示應用程式通道特性。

若要檢視特定語言環境的特性，請按一下 [表格喜好設定]，然後提供 [語言環境] 值：de、fr、es、ja、ko、zh、zh_CN、zh_TW。

6. 編輯 [userApps] 特性。
 - a. 在 [userApps] 特性中按一下 [編輯值...] 連結。
將會出現含有現有應用程式的快顯式視窗。
 - b. 從清單中移除下列應用程式：
 - NetMail Lite
 - NetMail
 - c. 向清單增加下列應用程式：
 - NetFile
 - d. 按一下 [儲存]，然後按 [關閉]。

7. 編輯 [target] 特性。
 - a. 在 [target] 特性中按一下 [編輯值...] 連結。
將會出現含有現有目標的快顯式視窗。
 - b. 從清單中移除下列目標：
 - NetMailLite|
 - NetMailServlet?nsid=newHTMLSessionNetMailLite|
 - NetMailServlet?nsid=newHTMLSession
 - NetMail|NetMailServlet?nsid=newAppletSession

- c. 移除下列重複出現的目標：

```
Instant Messenger (Java WebStart) |  
IMLaunch?provider=IMChannel&launch=jnlp&last=false
```

```
Instant Messenger (Browser) |  
IMLaunch?provider=IMChannel&launch=plugin&last=false
```

- d. 向清單增加下列目標：

```
NetFile|/portal/NetFileApplet?Refer=java2
```

- e. 按一下 [儲存]，然後按 [關閉]。

正確存取搜尋伺服器

當將發行版本 4 Portal Server 升級到發行版本 5 時，搜尋伺服器會與 Portal Server 分隔開來，因此會變更對於搜尋伺服器的 URL 存取。

- 在發行版本 4 中，URL 是 `http://hostName:port/portal/search`
- 在發行版本 5 中，URL 是 `http://hostName:port/UpgradedSearch/search`

因此，您必須手動修改實作了 SearchProvider 或 DiscussionProvider 介面的所有入口網站通道 (例如 Search、DiscussionLite、Discussions 以及 Instant Messaging 通道) 的「顯示設定檔」。特別的是，您必須修改這些通道的 searchServer 特性 (無論在何組織或角色層級上，它們都可能在「顯示設定檔」中發生) 以正確參照搜尋伺服器。修改 searchServer 值，如下所示：

```
value="http://hostName:port/UpgradedSearch/search"
```

同時，針對 Instant Messaging 通道，Instant Messaging Server 配置特性 (iim_arch.portal.search) 也必須以新的搜尋伺服器 URL 更新。

復原 Directory Proxy Server 配置

如果 Portal Server 實例已經透過 Directory Proxy Server 實例存取了 Directory Server，則必須在升級前將 Directory Proxy Server 主機與連接埠號設定復原為其原始值。請參閱第 299 頁的「[移除 Directory Proxy Server 配置](#)」，在此過程中修改了這些特性的值以備升級之用。

手動註冊 Portal Server Secure Remote Access 元件

如果您在升級程序結束時遇到報告至標準輸出的空指針異常問題，它表示 Portal Server Secure Remote Access 元件的註冊 (若有的話) 失敗。

在這種情況下，您可以透過執行下列指令來手動註冊 (啟用) Portal Server Secure Remote Access 元件：

```
PortalServer7-base/bin/psadmin provision-sra -u amadminUser -f passwordFile
    -p Portal_ID --gateway-profile profileName --enable
```

啟用 URLScrapper 通道

當從發行版本 4 升級到發行版本 5 時，您必須啟用 URLScrapper 通道。請執行下列程序：

1. 登入 Portal Server 主控台

按一下 [Portal] 標籤，然後按一下經過升級的入口網站。

2. 從 [選取 DN] 下拉式功能表中選取 [TopLevel] (全域) 並按一下 [下載顯示設定檔] 連結。

將下載的檔案儲存在某個暫時位置

3. 找到 com.sun.portal.providers.urlscrapper.URLScrapperProvider。

4. 找到 XML 部份，它應以下列內容開頭：

```
<Provider advanced="false"
class="com.sun.portal.providers.urlscrapper.URLScrapperProvider"
```

並以下列內容結尾：

```
</Provider>
```

5. 用以下內容取代步驟 4 中的 XML 部份：

```
<Provider advanced="false"
class="com.sun.portal.providers.urlscrapper.URLScrapperProvider"
container="false" lock="false" merge="fuse"
name="URLScrapperProvider" version="2">

<Properties advanced="false" lock="false" merge="fuse"
name="_properties" propagate="true">

<String advanced="false" lock="false" merge="replace" name="title"
propagate="true" value="UrlScrapper Channel"/>
```

```
<String advanced="false" lock="false" merge="replace"
name="description" propagate="true" value="This is a test for
urlscrapper"/>

<Boolean advanced="true" lock="false" merge="replace"
name="isEditable" propagate="true" value="false"/>

<Boolean advanced="true" lock="false" merge="replace"
name="isTopLevel" propagate="true" value="false"/>

<String advanced="true" lock="false" merge="replace" name="editType"
propagate="true" value="edit_subset"/>

<Boolean advanced="true" lock="false" merge="replace"
name="enableUBT" propagate="true" value="false"/>

<String advanced="false" lock="false" merge="replace"
name="urlScrapperRulesetID" propagate="true"
value="default_ruleset"/>

<String advanced="false" lock="false" merge="replace" name="width"
propagate="true" value="thick"/>

<String advanced="true" lock="false" merge="replace"
name="refreshTime" propagate="true" value="0"/>

<String advanced="true" lock="false" merge="replace" name="helpURL"
propagate="true" value="en/desktop/urlscrpr.htm"/>

<String advanced="false" lock="false" merge="replace" name="url"
propagate="true" value=""/>

<String advanced="false" lock="false" merge="replace"
name="fontFacel" propagate="true" value="Sans-serif"/>

<String advanced="false" lock="false" merge="replace"
name="productName" propagate="true" value="Sun Java™ System Portal
Server 7"/>

<Boolean advanced="false" lock="false" merge="replace"
name="cookiesToForwardAll" propagate="true" value="true"/>

<String advanced="false" lock="false" merge="replace"
name="inputEncoding" propagate="true" value="UTF-8"/>

<Collection advanced="false" lock="false" merge="fuse"
name="cookiesToForwardList" propagate="true"/>
```



```
<Integer advanced="false" lock="false" merge="replace"
name="timeout" propagate="true" value="100"/>

<String advanced="true" lock="false" merge="replace" name="formData"
propagate="true" value=""/>

<Boolean advanced="true" lock="false" merge="replace"
name="isHttpAuth" propagate="true" value="false"/>

<String advanced="true" lock="false" merge="replace" name="loginUrl"
propagate="true" value=""/>

<String advanced="true" lock="false" merge="replace"
name="loginFormData" propagate="true" value=""/>

<String advanced="true" lock="false" merge="replace" name="uid"
propagate="true" value=""/>

<String advanced="true" lock="false" merge="replace" name="password"
propagate="true" value=""/>

<ConditionalProperties advanced="false" condition="client"
lock="false" merge="fuse" propagate="true" value="HTML">

<ConditionalProperties advanced="false" condition="locale"
lock="false" merge="fuse" propagate="true" value="en">

<String advanced="true" lock="false" merge="replace" name="helpURL"
propagate="true" value="en/desktop/urlscrpr.htm"/>

<String advanced="false" lock="false" merge="replace" name="url"
propagate="true" value=""/>

</ConditionalProperties>

<String advanced="true" lock="false" merge="replace" name="helpURL"
propagate="true" value="en/desktop/urlscrpr.htm"/>

<String advanced="false" lock="false" merge="replace" name="url"
propagate="true" value=""/>

</ConditionalProperties>

<ConditionalProperties advanced="false" condition="locale"
lock="false" merge="fuse" propagate="true" value="en">

<String advanced="false" lock="false" merge="replace" name="title"
propagate="true" value="UrlScrapper Channel"/>
```

```
<String advanced="false" lock="false" merge="replace"
name="description" propagate="true" value="This is a test for
urlscrapper"/>

</ConditionalProperties>

</Properties>

</Provider>
```

6. 儲存並上傳經過修改的檔案。

登出頁面中的變更

發行版本 5 Portal Server 登出頁面與之前的 Access Manager 登出頁面已有不同。請注意，此變更並不表示軟體存在缺陷。

回復升級 (Solaris)

本節描述影響 Portal Server 升級回復程序的注意事項，接著提供程序本身。

回復注意事項 (Solaris)

回復至發行版本 5 的升級的程序包括在 *PortalServer6-base* 復原至發行版本 4 安裝以及重新部署發行版本 4 Web 應用程式。

回復程序 (Solaris)

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 將 Directory Server 回復為它在升級前的狀態。

使用 Directory Server backup/restore 指令行及 GUI 公用程式。請參閱「Sun Java System Directory Server Enterprise Edition 6.0 管理指南」(<http://docs.sun.com/doc/820-0294>) 的「Directory Server 備份與復原」一章。

3. 停止 Portal Server，方法是停止它的 Web 容器。

Web Server 6.x :

WebServer-base/https-instanceName/stop

Web Server 7.0 :

Admin Server--

WebServer7Config-base/admin-server/bin/stopserv

Instance Server--

WebServer7Config-base/https-configName/bin/stopserv

Application Server 8.x :

*AppServer8-base/bin/asadmin stop-domain --user admin_ID
--password password domainName*

4. 移除發行版本 5 Portal Server 套裝軟體。

- a. 啟動 Java ES 解除安裝程式。

/var/sadm/prod/SUNWentsys5/uninstall

- b. 選取所有已經安裝的 Portal Server 元件。

- c. 確認解除安裝選擇。

- d. 結束 Java ES 解除安裝程式。

5. 重新啟動 Portal Server，方法是重新啟動它的 Web 容器。

Web Server 6.x :

WebServer-base/https-instanceName/start

Web Server 7.0 :

Admin Server--

WebServer7Config-base/admin-server/bin/startserv

Instance Server--

WebServer7Config-base/https-configName/bin/startserv

Application Server 8.x :

*AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName*

6. 透過 Java ES 發行版本 5 發行軟體使用下列指令來重新部署發行版本 4 Portal Server Web 應用程式：

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade rollback
```

其中 `os_arch` 與您所在平台相符，例如 `Solaris_sparc`。

`psupgrade rollback` 指令會取消對發行版本 5 Portal Server Web 應用程式的部署，並重新部署發行版本 4 Portal Server Web 應用程式。

此指令會將 `PortalServer6-base/web-src` 中的內容重新部署到 `/var/PortalServer6-base/https-hostName/deploy-dir/web-apps`。因此，若要對 Portal Server Web 應用程式進行任何自訂，應先對 `/web-src` 進行這些自訂，然後將它們重新部署到 `/web-apps`。執行 `psupgrade rollback` 指令之前，應在 `/web-src` 中複製您可能會在 `/web-apps` 下進行的任何變更。否則，此類變更將被覆寫。

7. 停止 Web 容器，然後將其重新啟動。

儘管並不需要在所有情況下都重新啟動 Web 容器，但這樣做可確保 Portal Server 在未修改的狀態下啟動。

回復升級 (Linux)

由於升級到發行版本 5 需要移除發行版本 4 二進位檔，因此在 Linux 上很難回復升級。

回復的一個方法是在升級及測試系統之前建立一個平行系統，然後再嘗試進行升級。如果您需要回復升級，您可以回復至該平行系統。

多重實例升級

在某些部署架構中，會將 Portal Server 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Portal Server 實例在多台電腦上執行，並透過負載平衡器來分散負載。

若 Portal Server 實例已達負載平衡，則可以執行漸進式升級，即依序升級 Portal Server 實例而不中斷服務，如下所述。此程序應考慮以下限制：發行版本 4 Portal Server 無法與發行版本 5 Portal Server 目錄資料搭配使用。

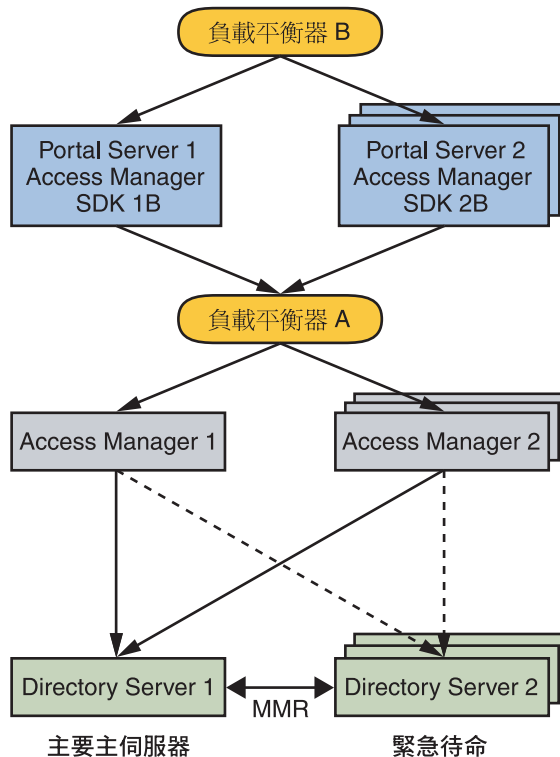
第 317 頁的圖 15-1 中顯示的部署架構將用來說明將發行版本 4 Portal Server 實例串聯升級到發行版本 5 的程序。

備註 如需包含 Portal Server Secure Remote Access 元件的架構的資訊，請參閱第 370 頁的「多重實例升級」。

在圖 15-1 的架構中，透過負載平衡器存取了多個 Portal Server 實例以提供可用性及延展性。相反，Portal Server 實例可透過負載平衡器來存取 Access Manager 實例。Access Manager 和 Access Manager SDK 實例會存取已針對多個主伺服器複製 (MMR) 設定的目錄。雖然其他 Directory Server 複製方案也是允許的，但 MMR 是高可用性及延展性目錄服務的代表。

在圖 15-1 中，已將 Portal Server、Access Manager 和 Directory Server 的多重實例組成群組，以便於對升級程序進行說明。例如，Portal Server 2 就代表 Portal Server 的第二個到第 n 個實例。

圖 15-1 多重 Portal Server 實例的部署架構範例



發行版本 4 Portal Server 到發行版本 5 的漸進式升級的執行方式如下：

1. 如果您要將發行版本 4 Access Manager 升級到發行版本 5，請執行漸進式升級，如第 263 頁的「多重實例升級」所述。請注意，在將發行版本 4 Portal Server 升級到發行版本 5 的過程中，您無須將發行版本 4 Access Manager 升級到發行版本 5。
2. 將 Portal Server 2 配置為指向 Directory Server 2，而非指向 Directory Server 1。

為了簡潔起見，在此步驟及後續步驟中，「Portal Server 2」將表示 Portal Server 2 至 Portal Server n。

3. 升級 Portal Server 1。
 - a. 停用 Load Balancer B 中的 Portal Server 1。

請求不會再被路由至 Portal Server 1。

- b. 停用 Directory Server MMR。

Directory Server 2 將不再與 Directory Server 1 同步。

- c. 將 Access Manager SDK 1B 升級到發行版本 5。

請使用第 265 頁的「僅升級發行版本 4 Access Manager SDK」中的程序。

- d. 將 Portal Server 1 升級到發行版本 5。

執行 Portal Server 實例的升級，如第 294 頁的「升級發行版本 4 Portal Server」所述，並請注意下列項目：

- 請特別注意下列升級前作業：第 299 頁的「移除負載平衡器的配置」。
- 在執行升級之前，請確認在本機 Access Manager SDK 中及與其相關的遠端 Access Manager 實例中，*AccessManagerConfig-base/config/AMConfig.properties* 檔案內 *am.encryption.pwd* 的值是否相同。
- 確定為要升級的每一個 Portal Server 實例的 *psupgrade* 所請求的 Portal Instance ID 參數提供了非空、唯一值。

將 Directory Server 1 的 Portal Server 資料升級到發行版本 5。

- e. 啟用 Load Balancer B 中的 Portal Server 1。

請求會重新路由至 Portal Server 1。

4. 升級 Portal Server 2。
 - a. 停用 Load Balancer B 中的 Portal Server 2。
請求不會再被路由至 Portal Server 2。
 - b. 將 Portal Server 2 的配置復原為指向 Directory Server 1。
 - c. 將 Access Manager SDK 2B 升級到發行版本 5。
使用第 318 頁的步驟 c 中描述的程序。
 - d. 將 Portal Server 2 升級到發行版本 5。
使用第 318 頁的步驟 d 中描述的程序。
 - e. 啓用 Load Balancer B 中的 Portal Server 2。
請求會重新路由至 Portal Server 2。
5. 啓用 Directory Server MMR。
現在，Directory Server 2 的 Portal Server 資料便與 Directory Server 1 同步。

從 Java ES 發行版本 3 升級 Portal Server

將 Java ES 2005Q1 (發行版本 3) Portal Server 升級到發行版本 5 的程序與將發行版本 4 Portal Server 升級到發行版本 5 的程序相同，但有以下例外：

- [發行版本 3 升級前作業：升級 Portal Server 相依元件](#)
- [升級發行版本 3 Portal Server](#)
- [多重實例升級](#)

發行版本 3 升級前作業：升級 Portal Server 相依元件

但是，當從發行版本 3 升級 Portal Server 時，您必須在升級 Portal Server 之前，將 Access Manager 和 Web 容器 (Web Server 或 Application Server) 都升級到發行版本 4 或發行版本 5，但是您不能在發行版本 3 上保留任何相依元件，也不能將某些相依元件升級到發行版本 4，而將其他相依元件升級到發行版本 5。如需更多資訊，請參閱第 291 頁的「選擇性的升級問題」。

下列相依元件必須按以下顯示的順序升級。

1. **共用元件。** 在第 59 頁的第 2 章「[升級 Java ES 共用元件](#)」中提供了將 Java ES 共用元件升級到發行版本 5 的指示。
2. **Directory Server。** 在第 93 頁的第 5 章「[Directory Server](#)」中提供了將 Directory Server 升級到發行版本 5 的指示。
3. **Web 容器軟體。** 在第 123 頁的第 7 章「[Web Server](#)」和第 189 頁的第 11 章「[Application Server](#)」中分別提供了升級 Web Server 和 Application Server 的指示。

備註

升級協力廠商 Web 容器 (如來自 Weblogic 和 WebSphere 的容器) 可能導致 Portal Server 發生故障，因為容器升級會覆寫為支援 Portal Server 而對這些容器進行的自訂。

在這些情況下，必須為升級後的 Web 容器環境重新安裝和重新配置 Portal Server。

4. **Access Manager (Access Manager SDK)。** 在第 241 頁的第 14 章「[Access Manager](#)」中提供了將 Access Manager 升級到發行版本 5 的指示。

升級發行版本 3 Portal Server

若要將發行版本 3 Portal Server 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 293 頁的「從 Java ES 發行版本 4 升級 Portal Server」中的指示。

發行版本 3 升級後作業

將 Portal Server 從發行版本 3 升級至發行版本 5 時，除了第 306 頁的「發行版本 4 升級後作業」中說明的升級後程序之外，您還必須執行處理以下情況所需的升級後程序：

- [訂閱討論](#)

訂閱討論

除非您先編輯全域顯示設定檔最上層特性以增加以下 String 特性，否則在社群中訂閱討論將不會成功：

```
helpURL=en/desktop/usedesk.htm
```

請執行下列程序：

1. 建立顯示設定檔 XML snippet 檔案 helpUrl.xml：

```
<?xml version="1.0" encoding="utf-8" ?>
  <!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
  <Properties>
    <String name="helpURL" value="en/desktop/usedesk.htm" />
  </Properties>
```

2. 使用下列指令執行全域顯示設定檔特性：

```
./psadmin modify-dp -u amadminUser -f /tmp/passwordFile -p portal_ID
-m -g helpUrl.xml
```

其中的 -m 選項為必要選項，可避免覆寫整個全域顯示設定檔。

多重實例升級

在某些部署架構中，會將 Portal Server 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Portal Server 實例在多台電腦上執行，並透過負載平衡器來分散負載。

若 Portal Server 實例已達負載平衡，則可以執行漸進式升級，即依序升級 Portal Server 實例而不中斷服務，如下所述。此程序應考慮以下限制：發行版本 3 Portal Server 無法與發行版本 5 Portal Server 目錄資料搭配使用。

若要執行從發行版本 3 Portal Server 至發行版本 5 的漸進式升級，除了將參照的發行版本 4 取代為發行版本 3 之外，其他請遵循如[第 316 頁](#)的「多重實例升級」中所述的程序。此外，您還必須升級 Access Manager，如[第 318 頁](#)的步驟 1 所述。

從 Java ES 發行版本 2 升級 Portal Server

本節包含將 Java ES 2004Q2 (發行版本 2) Portal Server 升級到發行版本 5 的相關資訊。除了如以下幾節所述的一些變更以外，升級程序與將發行版本 4 Portal Server 升級到發行版本 5 類似：

- [發行版本 2 升級前作業](#)
- [升級發行版本 2 Portal Server](#)
- [發行版本 2 升級後作業](#)
- [多重實例升級](#)

備註 如果您要在 Linux 平台上從發行版本 2 Portal Server 進行升級，則必須執行雙重升級，也就是 Portal Server 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Portal Server)。如需更多資訊，請參閱第 292 頁的「雙重升級」。

發行版本 2 升級前作業

除了升級 Portal Server 相依元件以外，從發行版本 2 升級 Portal Server 的升級前作業與第 294 頁的「發行版本 4 升級前作業」中所述的內容相同。

當從發行版本 2 升級 Portal Server 時，您必須在升級 Portal Server 之前，將 Access Manager 和 Web 容器 (Web Server 或 Application Server) 都升級到發行版本 4 或發行版本 5，但是您不能在發行版本 2 上保留任何相依元件，也不能將某些相依元件升級到發行版本 4，而將其他相依元件升級到發行版本 5。如需更多資訊，請參閱第 291 頁的「選擇性的升級問題」。

特別的是，Web 容器軟體必須從發行版本 2 升級，這表示當執行 psupgrade 程序檔時，只支援第 291 頁的表 15-4 中的方案 2 與方案 5。

下列相依元件必須按以下顯示的順序升級。

1. **共用元件。** 在第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了將 Java ES 共用元件升級到發行版本 5 的指示。
2. **Directory Server。** 在第 108 頁的「從 Java ES 發行版本 2 升級 Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。

3. **Web 容器軟體**。在第 123 頁的第 7 章「Web Server」和第 189 頁的第 11 章「Application Server」中分別提供了升級 Web Server 和 Application Server 的指示。

備註 升級協力廠商 Web 容器 (如來自 Weblogic 和 WebSphere 的容器) 可能導致 Portal Server 發生故障，因為容器升級會覆寫為支援 Portal Server 而對這些容器進行的自訂。在這些情況下，必須為升級後的 Web 容器環境重新安裝和重新配置 Portal Server。

4. **Access Manager (Access Manager SDK)**。在第 241 頁的第 14 章「Access Manager」中提供了將 Access Manager 升級到發行版本 5 的指示。

升級發行版本 2 Portal Server

將 Portal Server 從發行版本 2 升級到發行版本 5 的程序取決於部署 Portal Server 軟體的 Web 容器，如以下幾節所述。

升級發行版本 2 Portal Server : Web Server Web 容器

若要將發行版本 2 Portal Server 升級到發行版本 5，在部署到 Web Server Web 容器中時，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他動作請遵循第 293 頁的「從 Java ES 發行版本 4 升級 Portal Server」中的指示。

但是，如果將 Portal Server 部署到發行版本 5 Web Server (Web Server 7.0)，您必須先執行以下步驟才能升級發行版本 2 Web Server：

1. 登入 Web Server 管理主控台。
2. 按一下 [編輯虛擬伺服器] > [Web 應用程式]。
3. 移除所有 URI 中包含 /portal 或 /portalsamples 的已部署 Web 應用程式。
4. 按一下 [儲存]。
5. 按一下 [部署擱置]。

升級發行版本 2 Portal Server : Application Server Web 容器

如果將發行版本 2 Portal Server 升級到發行版本 5，則當部署到 Application Server Web 容器時，Application Server 便已經從發行版本 2 升級到了發行版本 5。

升級到發行版本 5 時，最初於其中部署了 Portal Server 的發行版本 2 Application Server 實例 (*instanceName*) 遷移到由 Application Server 升級程序所建立的節點代理程式下。在這個已升級的 Application Server 實例中升級 Portal Server 需要執行下列步驟：

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。

[第 59 頁的第 2 章「升級 Java ES 共用元件」](#)中提供了指示。

對於在[第 326 頁的步驟 9](#)中執行 `psupgrade` 程序檔來說，此步驟是必要的先決條件。

3. 停止可能在本地執行的 Portal Server Secure Remote Access Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxyd stop
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
Rewriter Proxy : netstat -an | grep 10443
Netlet Proxy : netstat -an | grep 10555
```

4. 如果 Access Manager 與 Portal Server 分別部署在不同的 Web 容器中，請確定 Access Manager 正在執行。
5. 如果未在執行，請啟動 Portal Server，這可藉由啟動其部署到的 Web 容器來實現。

- a. 啟動 Domain Administration Server (DAS) (如果它尚未啟動)。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

- b. 若部署了 Portal Server 的已升級 Application Server 實例 (*instanceName*) 尚未執行，請啟動它。

執行方式是啟動遷移的已升級 Application Server 實例所屬的節點代理程式：

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

以上指令及後續步驟中使用了下列慣例：

- 其中 *nodeagentName* 的形式為 *hostName_domainName*，但簡單預設為 *hostName*。
- 預設 *domainName* 是 *domain1*
- 預設 *instanceName* 是 *server1*

6. 取消 Portal Server 元件的部署。

```
AppServer8-base/bin/asadmin undeploy --user admin_ID  
--password password --target instanceName portal
```

```
AppServer8-base/bin/asadmin undeploy --user admin_ID  
--password password --target instanceName portletsamples
```

7. 設定 *psupgrade* 程序檔所需的兩個環境變數 (*ANT_HOME* 和 *JAVA_HOME*)。例如，

Solaris OS：

```
export ANT_HOME=/usr/sfw  
export JAVA_HOME=/usr/jdk/entsys-j2se
```

Linux OS：

```
export ANT_HOME=/opt/sun  
export JAVA_HOME=/usr/jdk/entsys-j2se
```

8. 請確定您電腦上有足夠的交換空間。

做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。

9. 從 Java ES 發行版本 5 發行物中執行 *psupgrade* 程序檔。

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin  
./psupgrade
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。

備註 如果您不慎在錯誤的 *os_arch* 目錄中執行 `psupgrade`，您需要取消下述程序：

- a. 變更爲正確的 *os_arch* 目錄。
 - b. 回復對於 Portal Server 資料的變更。

```
./psupgrade rollback
```

提供要求的參數與密碼。
 - c. 再次執行 `psupgrade`。
-

`psupgrade` 程序檔會呼叫 Java ES 安裝程式以安裝新的套裝軟體，並查詢系統以偵測您將在其中部署 Portal Server Web 應用程式的 Web 容器之位置與連接埠號及其他相關資訊。視 Web 容器升級方案而定 (請參閱第 291 頁的表 15-4)，在此方案 5 中，程序檔會要求您輸入將 Portal Server 部署到適當 Web 容器中所需要的資訊。

第 296 頁的表 15-6 顯示當發行版本 2 Application Server 已經升級到發行版本 5 (方案 5) 時所需要的資訊。

備註 請務必爲 `psupgrade` 參數輸入正確的值，因爲您無法返回及變更它們，而且也很難回復由 `psupgrade` 程序檔所做的變更。若要回復對於 Portal Server 資料的變更，您必須執行

```
./psupgrade rollback
```

然後嘗試再次執行 `psupgrade`。

10. 停止 Domain Administration Server (DAS) 與在第 325 頁的步驟 5 中啓動的節點代理程式。

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID
--password password domainName
```

```
AppServer8-base/bin/asadmin stop-node-agent --user admin_ID
--password password nodeagentName
```

11. 重新啓動 Domain Administration Server (DAS)、節點代理程式和在步驟 10 中停止的伺服器實例。

備註 啓動伺服器實例之前，請務必使用 `startinstances=false` 選項單獨啓動節點代理程式，如下所示。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

```
AppServer8-base/bin/asadmin start-node-agent --port DASportNumber  
--startinstances=false --user admin_ID --password password  
nodeagentName
```

```
AppServer8-base/bin/asadmin start-instance --port DASportNumber  
--user admin_ID --password password instanceName
```

`DASportNumber` 的預設值爲 4848。

發行版本 2 升級後作業

將 Portal Server 從發行版本 2 升級至發行版本 5 時，除了第 306 頁的「發行版本 4 升級後作業」中說明的升級後程序之外，您還必須執行處理以下情況所需的升級後程序：

- 單次登入配置
- 啓用 URLScrapper 通道
- 刪除 Gateway 服務項目

單次登入配置

將 Portal Server 從發行版本 2 升級之後，使用 `ssoadapter` 中介範本來存取後端伺服器的 Portal Desktop 通訊通道 (例如，郵件、行事曆與通訊錄) 都會失敗。

例如，如果您已經使用 Messaging Server 的特定設定修改了發行版本 2 郵件 `ssoadapter` 中介範本 `SUN-UWC-MAIL`，則升級到發行版本 5 之後，將存在兩個 `SUN-UWC-MAIL` `ssoadapter` 中介範本：一個是尚未變更的發行版本 2 版本，另一個是新的發行版本 5 版本。您會在 Portal Server 主控台與 `psadmin` 指令行介面中發現重複的 `ssoadapter` 中介範本，二者具有相同的名稱。

使用 `ssoadapter` 中介範本的通道將無法與後端伺服器建立連線並擷取資料。

若要修正此問題，您必須擷取 ssoadapter 中介範本資料、重新命名重複項目，然後取代已經修改的資料。請執行下列程序：

1. 匯出 ssoadapter 中介範本資料。

使用 amadmin 公用程式來匯出 Access Manager 服務資料，如下所示：

a. 建立 amadmin 請求檔案 /tmp/ssoadapter-template-gets.xml。

公用程式會使用此檔案來擷取 ssoadapter 中介範本資料：

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Requests
PUBLIC "-//iPlanet//iDSAME 5.0 Admin CLI DTD//EN"
"jar://com/iplanet/am/admin/cli/amAdmin.dtd"
>
<Requests>
  <SchemaRequests serviceName="SunSSOAdapterService"
  SchemaType="global">
    <GetServiceDefaultValues>
      <Attribute name="sunConfigurationTemplates"/>
    </GetServiceDefaultValues>
  </SchemaRequests>
</Requests>
```

b. 執行下列 asadmin 指令：

```
AccessManager-base/bin/amadmin -u amadminUser -w password
-t ssoadapter-templates-get.xml > /tmp/ssoadapter-templates.xml
```

指令的輸出會儲存到 /tmp/ssoadapter-templates.xml 中。

/tmp/ssoadapter-templates.xml 檔案的格式如下：

```
sunConfigurationTemplates=
[<ssoadapter meta-template1>, <ssoadapter meta-template2>, ...]
```

而且每個 <ssoadapter meta-template> 的語法如下：

```
default|imap:/?configName=SUN-UWC-MAIL
&proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D&subType=sun-one
&enableProxyAuth=false ...
```

2. 修改 /tmp/ssoadapter-templates.xml 檔案以重新命名重複的 ssoadapter 中介範本。

a. 尋找 /tmp/ssoadapter-templates.xml 檔案中的每個範本。

尋找 default|imap:/?configName= 字串。

- b. 以唯一值取代重複的 `ssoadapter` 中介範本名稱。

例如，如果有兩個 `SUN-UWC-MAIL ssoadapter` 中介範本，以 `SUN-UWC-MAIL2` 取代其中一個的 `configName` 值會產生兩個唯一命名的範本：

```
default |imap:/?configName=SUN-UWC-MAIL ...
default |imap:/?configName=SUN-UWC-MAIL2 ...
```

- 3. 建立將匯入已修改 `ssoadapter` 中介範本的 `amadmin` 請求檔案，從而覆寫原始資料。

- a. 將 `/tmp/ssoadapter-templates.xml` 複製到 `/tmp/ssoadapter-new-templates.xml` 中

- b. 在 `/tmp/ssoadapter-new-templates.xml` 中，將以下字串：

```
sunConfigurationTemplates=[
```

替代為：

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Requests
  PUBLIC "-//iPlanet//iDSAME 5.0 Admin CLI DTD//EN"
  "jar://com/iplanet/am/admin/cli/amAdmin.dtd"
>
<Requests>
  <SchemaRequests serviceName="SunSSOAdapterService"
    SchemaType="Global">
    <ModifyDefaultValues>
      <AttributeValuePair>
        <Attribute name="sunConfigurationTemplates"/>
      </AttributeValuePair>
    </ModifyDefaultValues>
  </SchemaRequests>
</Requests>
```

- c. 以「`&`」取代所有連字號（「`&`」）。

例如，以下行：

```
default |imap:/?configName=SUN-UWC-MAIL
&proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&subType=sun-one&enableProxyAuth=false...
```

會變成：

```
default |imap:/?configName=SUN-UWC-MAIL
&amp;proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&amp;subType=sun-one&amp;enableProxyAuth=false ...
```

- d. 移除每個 `ssoadapter` 中介範本末尾的逗號（「`,`」）。

- e. 以起始 `<Value>` 標記與結束 `</Value>` 標記包住每個 `ssoadapter` 中介範本。

例如：

```
<Value>default|imap:/?configName=SUN-UWC-MAIL
&amp;proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&amp;subType=sun-one&amp;enableProxyAuth=false ...</Value>
```

- f. 從上一個 `ssoadapter` 中介範本中移除右括號 (「`]`」)。
- g. 在檔案末尾處加上以下幾行：

```
</AttributeValuePair>
</ModifyDefaultValues>
</SchemaRequests>
</Requests>
```

以上步驟中使用的單一範本的最終 `ssoadapter-new-templates.xml` 檔案應該如下所示：

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Requests
  PUBLIC "-//iPlanet//iDSAME 5.0 Admin CLI DTD//EN"
  "jar://com/iplanet/am/admin/cli/amAdmin.dtd"
>
<Requests>
  <SchemaRequests serviceName="SunSSOAdapterService"
    SchemaType="Global">
    <ModifyDefaultValues>
      <AttributeValuePair>
        <Attribute name="sunConfigurationTemplates"/>
        <Value>default|imap:/?configName=SUN-UWC-MAIL
&amp;proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&amp;subType=sun-one&amp;enableProxyAuth=false ...</Value>
      </AttributeValuePair>
    </ModifyDefaultValues>
  </SchemaRequests>
</Requests>
```

4. 匯入新的 `ssoadapter-new-templates.xml` 檔案。

```
AccessManager-base/bin/amadmin -u amadminUser -w password -v
-t ssoadapter-new-templates.xml
```

此時，您可以存取 Portal Server 主控台家中的 `ssoadapter` 標籤，以查看更新後的 `ssoadapter`。

啓用 URLScrapper 通道

當從發行版本 3 升級到發行版本 5 時，您必須啓用 URLScrapper 通道。請參閱第 311 頁的「啓用 URLScrapper 通道」。

刪除 Gateway 服務項目

當從發行版本 2 升級 Portal Server 時，必須手動刪除 amService-srapGateway 使用者項目，否則如果使用了 Portal Server Secure Remote Access Gateway 元件，將無法在升級之後啓動。執行下列步驟：

1. 登入 Access Manager 主控台。
2. 列出機構 DN 中的所有使用者。
3. 刪除 amService-srapGateway 使用者。

多重實例升級

從發行版本 2 Portal Server 升級到發行版本 5 時，不支援多重實例漸進式升級 (請參閱第 316 頁的「多重實例升級」)。

從臨時功能發行版本 7.0 升級 Portal Server

本節包含將 Portal Server 從臨時功能發行版本 (IFR) 7.0 2005Q4 升級到 Java ES 5 (發行版本 5) 的相關資訊。

本節涵蓋以下主題：

- [Portal Server IFR 升級簡介](#)
- [Portal Server IFR 7.0 升級](#)
- [多重實例升級](#)

Portal Server IFR 升級簡介

將 Portal Server IFR 7.0 升級到發行版本 5 Portal Server 時，請考慮升級程序的下列層面：

- Application Server 7.x 上不支援 Portal Server IFR，因此第 291 頁的表 15-4 中的升級方案 5 不適用。
- 在從發行版本 4 升級時，用於將 Portal Server IFR 升級到發行版本 5 的 `psupgrade` 程序檔不會安裝新的套裝軟體。升級程序會要求您套用以下修補程式：

表 15-8 將 Portal Server IFR 升級到發行版本 5 的修補程式¹

說明	修補程式 ID：Solaris 9 和 10	修補程式 ID：Linux
Portal Server 7.1	121465-28 (SPARC) 121466-28 (x86)	121467-28
Portal Server 7.1 本土化版本	123254-02 (SPARC) 124590-02 (x86)	123255-02

1. 修補程式的修訂版號碼是升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

- `psupgrade` 程序檔不支援將 Portal Server IFR 升級到發行版本 5 Web Server。此限制有兩種含意：
 - 如果 Web Server 已經升級到發行版本 5，則您無法升級在較早版本 Web Server 容器中部署的 Portal Server IFR 軟體。因此不支援第 291 頁的表 15-4 中的 Web 容器升級方案 2。請注意第 53 頁的「特殊情況」中的警告。
 - 如果 Web Server 尚未升級到發行版本 5，您可先將較早 Web Server (6.x) 容器中的 Portal Server IFR 軟體升級到發行版本 5 (如下面「Portal Server IFR 7.0 升級」中所述)，然後依序將 Web Server (必要時再加上 Access Manager) 升級到發行版本 5。如果您的升級方案就是如此，另請參閱第 343 頁的「發行版本 5 Web Server (7.0) Web 容器中的升級」以取得其他的升級後指示。

Portal Server IFR 7.0 升級

本節描述如何在 Solaris 及 Linux 平台上將 Portal Server 從 IFR 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [IFR 7 升級前作業](#)
- [升級 Portal Server IFR 7.0 \(Solaris\)](#)
- [升級 IFR 7 Portal Server \(Linux\)](#)
- [驗證升級](#)
- [IFR 7 升級後作業](#)
- [回復升級 \(Solaris\)](#)
- [回復升級 \(Linux\)](#)

IFR 7 升級前作業

除了下列幾點，IFR 升級與發行版本 4 升級的升級前作業相同 (請參閱第 294 頁的「發行版本 4 升級前作業」)：

- [取得所需配置資訊](#)
- [Common Agent Container 的配置](#)

取得所需配置資訊

`psupgrade` 程序檔所需的資訊 (詳見第 296 頁的「取得必要的配置資訊和密碼」) 不完全適用於從 Portal Server IFR 進行的升級。由於 Application Server 7.x 上不支援 Portal Server IFR，因此第 296 頁的表 15-6 中的 Web 容器升級方案 5 不適用。

Common Agent Container 的配置

Common Agent Container 是一個共用元件，可為 Java ES 監視及管理代理程式提供容器服務。Portal Server 管理工具（例如 Portal Server 主控台及 psadmin 命令行介面）會使用一組皆部署於 Common Agent Container 中的監視及管理代理程式（它們一起稱為 Portal Administration Server）。

如果在執行 Portal Server IFR 7.0 的升級之前，已將 Java ES 共用元件升級到發行版本 5，則需要執行以下額外步驟才能登入發行版本 5 Portal Server 主控台及使用 psadmin 命令行介面。（如果在您執行 Portal Server IFR 7.0 的升級之前，Java ES 共用元件尚未升級到發行版本 5，則請忽略以下額外步驟。）

1. 重新配置 Common Agent Container。

```
PortalServer7-base/bin/psconfig --config
PortalServer7-base/samples/example2.xml
```

example2.xml 檔案提供了重新配置資訊。您必須在執行 psconfig 指令之前先編輯 example2.xml 檔案以提供必要的密碼。如果您使用非預設 Portal Server 位置，您還必須提供正確的目錄。

2. 編輯 Web 容器的類別路徑以參照 Common Agent Container。

Web 容器的類別路徑將包含對於先前 Common Agent Container 發行版本的位置的參照：*(rel4CAC-base-dir/lib/cacao_cacao.jar)*。

以發行版本 5 位置取代此參照：*(rel5CAC-admin-dir/lib/cacao_cacao.jar)*。

3. 檢查是否已在 *PortalServer7Config-base* 目錄中以下列特性建立了特性檔案 `pasconnect.properties`：

```
pas.host=
```

該特性值可為空值、localhost 或實際的 Portal Server 主機名稱。

4. 重新啟動 Common Agent Container。

```
rel5CAC-admin-dir/bin/cacaoadm start
```

升級 Portal Server IFR 7.0 (Solaris)

本節先討論影響 Portal Server IFR 升級程序的注意事項，接著提供對程序本身的說明。

IFR 7 升級注意事項 (Solaris)

將 Portal Server IFR 升級到 Java ES 發行版本 5 與發行版本 4 升級要考慮的注意事項相同（請參閱第 299 頁的「升級注意事項 (Solaris)」）。

此外，請參閱第 333 頁的「Portal Server IFR 升級簡介」中所提出的問題。

IFR 7 升級程序 (Solaris)

下述程序適用於位於執行升級的電腦上的 Portal Server。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 停止可能在本地執行的 Portal Server Secure Remote Access Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t gateway -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t rwproxy -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t nlproxy -N gatewayProfileName
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
```

```
Rewriter Proxy : netstat -an | grep 10443
```

```
Netlet Proxy : netstat -an | grep 10555
```

3. 如果 Access Manager 與 Portal Server 分別部署在不同的 Web 容器中，請確定 Access Manager 正在執行。
4. 如果未在執行，請啟動 Portal Server，這可藉由啟動其部署到的 Web 容器來實現。

Web Server 6.x :

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

Application Server 8.x :

```
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

5. 根據第 333 頁的表 15-8 取得所需的修補程式。

除非指示您使用特定修訂版，請一律使用最新的修補程式修訂版。

可從下列位置將修補程式下載到 /tmp :

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

6. 請套用適當的 Portal Server 修補程式，如有必要，也請套用表 15-8 中的本土化修補程式。

```
patchadd /tmp/patch_ID
```


7. 確認修補程式升級成功：

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

輸出應傳回步驟 6 中套用的修補程式 ID 版本。

8. 若本土化套裝軟體已經在步驟 6 中升級，請將 Portal Server Console JVM 的語言環境設定為 UTF-8。

```
export LC_ALL=ja_JP.UTF-8
export LANG=ja_JP.UTF-8
```

9. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 與 JAVA_HOME)：

```
export ANT_HOME=/usr/sfw
export JAVA_HOME=/usr/jdk/entsys-j2se
```

10. 請確定您電腦上有足夠的交換空間。

做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。

11. 執行 psupgrade 程序檔。

```
cd PortalServer7-base/bin
./psupgrade
```

psupgrade 程序檔不會從 Java ES 發行版本 5 發行者中執行，而且不會呼叫 Java ES 安裝程式 (套裝軟體已經過修補)。

程序檔會查詢系統以偵測您要在其中部署 Portal Server Web 應用程式的 Web 容器之位置、連接埠號及其他相關資訊。視 Web 容器升級方案而定 (請參閱第 291 頁的表 15-4)，程序檔會要求您輸入將 Portal Server 部署到適當 Web 容器中所需要的其他資訊。

第 296 頁的表 15-6 顯示了第 291 頁的表 15-4 中不同 Web 容器升級方案所要求的資訊。

備註

請務必為 psupgrade 參數輸入正確的值，因為您無法返回及變更它們，而且也很難回復由 psupgrade 程序檔所做的變更。

12. 停止 Web 容器，然後將其重新啟動。

儘管並不需要在所有情況下都重新啟動 Web 容器，但這樣做可確保 Portal Server 在未修改的狀態下啟動。

升級 IFR 7 Portal Server (Linux)

本節討論影響 Portal Server 升級程序的注意事項，接著提供對程序本身的說明。

IFR 7 升級注意事項 (Linux)

在 Linux OS 上安裝發行版本 5 修補程式會移除先前的 RPM，除了這一點外，在 Linux 平台上將 Portal Server IFR 軟體升級到發行版本 5 與在 Solaris OS 上升級時所要考慮的注意事項相同 (請參閱第 335 頁的「IFR 7 升級注意事項 (Solaris)」)。

IFR 7 升級程序 (Linux)

下述程序適用於位於執行升級的電腦上的 Portal Server。

注意 在 Linux 上執行的從 Portal Server IFR 到發行版本 5 的升級無法回復。在執行下列程序之前，請務必先備份您的系統。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 停止可能在本地執行的 Portal Server Secure Remote Access Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t gateway -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t rwproxy -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t nlproxy -N gatewayProfileName
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
```

```
Rewriter Proxy : netstat -an | grep 10443
```

```
Netlet Proxy : netstat -an | grep 10555
```

3. 如果 Access Manager 與 Portal Server 分別部署在不同的 Web 容器中，請確定 Access Manager 正在執行。

4. 如果未在執行，請啟動 Portal Server，這可藉由啟動其部署到的 Web 容器來實現。

Web Server 6.x :
`WebServer-base/https-instanceName/start`

Application Server 8.x :
`AppServer8-base/bin/asadmin start-domain --user admin_ID
 --password password domainName`

5. 使用第 333 頁的表 15-8 中的修補程式號碼和 RPM 名稱來取得所需的修補程式。除非指示您使用特定修訂版，請一律使用最新的修補程式修訂版。

可從下列位置將修補程式下載到 /tmp :
<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>

6. 以第 333 頁的表 15-8 中的順序套用該表中的 Portal Server 修補程式以及 Portal Server 本土化 RPM (需要時)。

請參閱 Portal Server 修補程式的讀我檔案，其中說明了如何使用程序檔來套用修補程式的 RPM：

```
cd /tmp
```

其中 /tmp 是將修補程式下載到的目錄。

```
./upgradeportalrpm
```

更新程序檔會安裝 RPM 的部份。

針對本土化修補程式，請使用以下指令安裝每一個 RPM：

```
rpm -Fvh patchName-version.rpm
```

7. 確認修補程式升級成功：

```
rpm -qa | grep sun-portal
```

應傳回 RPM 的升級修訂版號碼。

8. 若本土化套裝軟體已經在步驟 6 中升級，請將 Portal Server Console JVM 的語言環境設定為 UTF-8。

```
export LC_ALL=ja_JP.UTF-8  

export LANG=ja_JP.UTF-8
```

9. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 與 JAVA_HOME)：

```
export ANT_HOME=/opt/sun  

export JAVA_HOME=/usr/jdk/entsys-j2se
```

10. 請確定您電腦上有足夠的交換空間。

做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。

11. 執行 `psupgrade` 程序檔。

```
cd PortalServer7-base/bin
./psupgrade
```

`psupgrade` 程序檔不會從 Java ES 發行版本 5 發行物中執行，而且不會呼叫 Java ES 安裝程式 (套裝軟體已經過修補)。

程序檔會查詢系統以偵測您要在其中部署 Portal Server Web 應用程式的 Web 容器之位置、連接埠號及其他相關資訊。視 Web 容器升級方案而定 (請參閱第 291 頁的表 15-4)，程序檔會要求您輸入將 Portal Server 部署到適當 Web 容器中所需要的其他資訊。

第 296 頁的表 15-6 顯示了第 291 頁的表 15-4 中不同 Web 容器升級方案所要求的資訊。

備註 請務必為 `psupgrade` 參數輸入正確的值，因為您無法返回及變更它們，而且也很難回復由 `psupgrade` 程序檔所做的變更。

12. 停止 Web 容器，然後將其重新啟動。

儘管並不需要在所有情況下都重新啟動 Web 容器，但這樣做可確保 Portal Server 在未修改的狀態下啟動。

驗證升級

您可使用下列指令驗證是否將 Portal Server 套裝軟體修補為發行版本 5：

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID
-f adminpasswordfile.
```

請參閱第 295 頁的表 15-5 中的輸出值。

若要驗證完整升級，請確認 Portal Desktop 會顯示出來，且 `psadmin` 管理公用程式也可以如文件所述地使用。

您也可以檢查 `/var/sadm/install/logs` 中的下列升級記錄檔：

- `Sun_Java_System_Portal_Server_upgrade.log`
- `Sun_Java_System_Portal_Server_upgrade.log_ant_xxx.log`

其中 `xxx` 可以是 `preupgrade`、`upgrade` 或 `postupgrade`。

IFR 7 升級後作業

將 Portal Server 從 IFR 7 升級到發行版本 5 時，您必須執行處理以下情況所需要的升級後程序：

- 啟用 Java ES Monitoring Framework
- Application Server Web 容器中的升級
- 發行版本 5 Web Server (7.0) Web 容器中的升級
- 重新部署自訂 Portlet 應用程式
- 遷移自訂的 Portlet 應用程式
- 更正書籤及應用程式通道中的連結
- 手動遷移基於 Struts 的 Portlet

啟用 Java ES Monitoring Framework

啟用 Java ES Monitoring Framework (MFWK 共用元件) 可讓 Portal Server 管理工具 (例如 Portal Server 主控台及 psadmin 命令行介面) 報告像是訪問者人數與他們常訪問的入口網站等統計資料。若要啟用 MFWK：

1. 找到下列兩個檔案：

```
MFWK-base/template/jesmf/desktopmfwk.properties
MFWK-base/template/jesmf/com.sun.cmm.ps.xml
```

其中 *MFWK-base* 是以下路徑：

```
/opt/SUNWmfwk      (Solaris)
/opt/sun/mfwk      (Linux)
```

2. 將這兩個檔案複製到以下目錄：

```
PortalServer7Data-base/portals/portal_ID/config/Portal_Instance/
```

3. 在 `desktopmfwk.properties` 檔案中，將

```
com.sun.portal.ProductCollectionId=%PS_DIR%
```

替代為

```
com.sun.portal.ProductCollectionId=Portal_Installed_Location
```

4. 將下列兩個 jar 檔案：

```
MFWK-base/lib/mfwk_instrum_tk.jar
MFWK-base/lib/mfwk_agent.jar
```

增加到適當的 Web 容器類別路徑 (Web Server 為 server.xml 檔案，Application Server 為 domain.xml) 中。

5. 重新啟動對應的 Web 容器。

Application Server Web 容器中的升級

如果 Portal Server 部署在 Application Server Web 容器中，則您必須執行以下額外程序才能成功重新部署 Portal Server：

1. 在 *AppServer8Config-base/domains/domainName/config/server.policy* 檔案中找到定義 `psconsole` 設定的區段：
2. 在區段末尾處加上以下這一行：

```
permission java.lang.RuntimePermission "getProtectionDomain"
```

3. 重新啟動 Application Server 實例。

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password passworddomainName
```

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

其中 *nodeagentName* 的形式為 *hostName_domainName*，但簡單預設為 *hostName*。

Application Server Web 容器中連線當機

當已升級的 Portal Server 部署在 Application Server Web 容器中時，入口網站應用程式可能會當機，一直等待取得 Java DB 連線。若要解決此問題，請執行下列步驟：

1. 對於下列 2 個參數，移除 *PortalServer7Data-base/derby/derby.properties* 中的設定：

```
derby.drda.maxThreads
derby.drda.timeslice
```

2. 重新啟動 Java DB。

```
ANT_HOME/bin/ant
-DPS_CONFIG=PortalServer7Config-base/PSConfig.properties
-buildfile PortalServer7-base/lib/derby.xml
[stop-instance|start-instance]
```

其中 *ANT_HOME* 在 Solaris 上是 */usr/sfw*，在 Linux 上是 */opt/sun*。

3. 變更 Application Server 的 Java DB 配置設定。

使用 Application Server 主控台，變更下列連線池資源的屬性值：
communitymcPool、FileSharingDBPool、PointBasePool、SurveyDBPool。

變更下列屬性值，如下：

```
將 Idle Timeout 變更爲 300 或更高
將 Resource Type 變更爲 javax.sql.ConnectionPoolDataSource
將 Datasource classname 變更爲
    org.apache.derby.jdbc.ClientConnectionPoolDataSource
```

4. 重新啓動部署了 Portal Server 的 Application Server 實例。

發行版本 5 Web Server (7.0) Web 容器中的升級

如果因爲 psupgrade 程序檔不支援在發行版本 5 Web Server 上升級 Portal Server IFR，而使升級後的 Portal Server IFR 位於尚未升級到發行版本 5 的 Web Server 中 (請參閱第 333 頁的「Portal Server IFR 升級簡介」)，您必須執行下列的額外升級後步驟：

1. 將 Web Server (必要時再加上 Access Manager) 升級到發行版本 5。
2. 重新配置 Portal Server 主控台及 psadmin 命令行介面所需要的 Web Server 容器值。

- a. 開啓 LDAP 瀏覽器。

配置值儲存在 Directory Server 中。

- b. 在 DN 下，尋找：

```
sunPortalAdminPortalDomainID=defaultDomain
->sunPortalAdminPortalDomainPortalID=portal1
->sunPortalAdminPortalDomainPortalServerInstanceIn=host-port
```

- c. 依下列指示，執行編輯作業。

請注意，所有項目均以下列字串開頭：

```
sunPortalAdminPortalDomainPortalServerInstance
```

- 刪除 WebContainerInstanceDir 對應的項目。
- 爲 WebContainerDomainName 增加一個項目，並爲其指定第 296 頁的表 15-6 中的 Web 容器配置名稱值。
- 編輯 InstallDir、WebContainerType、DocRoot 等項目，以及第 296 頁的表 15-6 中所示的其他參數，使其與發行版本 5 Web Server (7.0) 值相一致。

3. 建立發行版本 5 Portal Server 實例。

```
PortalServer7-base/bin/psadmin create-instance newInstance_ID
```

如果 *newInstance_ID* 的值已經存在，將會丟出錯誤，因此在以下的步驟 4 之前執行此步驟是有利的。

4. 刪除 Portal Server IFR 實例。

```
PortalServer7-base/bin/psadmin delete-instance oldInstance_ID
```

重新部署自訂 Portlet 應用程式

如果您已經建立並部署了自訂 portlet 應用程式，則必須在升級到發行版本 5 Portal Server 之後手動重新部署這些 portlet。即使顯示設定檔項目將會存在且通道名稱將會顯示，也只能在重新部署完自訂 portlet 之後才會看到內容。

使用下列指令重新部署 portlet：

```
PortalServer7-base/bin/psadmin deploy-portlet
```

您可以在以下位置尋找對應 .war 及 XML 檔案來確認重新部署：

```
PortalServer7Data-base/portals/Upgraded/war
```

遷移自訂的 Portlet 應用程式

以 Sun Java Web Console (SJWC) 所提供之使用者介面架構為基礎的 portlet 應用程式需要手動遷移至發行版本 5 並重新部署。

特別是，此需求適用於四個 Web 應用程式，這四個應用程式做為要自訂及安裝到入口網站的範例 portlet 應用程式隨 Portal Server 一起分佈，它們分別是：filesharing、surveys、wiki、rssportlet。在升級期間，會將這些範例 portlet 應用程式修正錯誤後的版本置於 portlet 應用程式區域中的磁碟上。如果您已經自訂了這些 portlet 應用程式供自己使用，則需要手動將它們遷移至發行版本 5 並重新進行部署；Portal Server 升級程序不會處理它們。

依預設，會部署其中一些 portlet 應用程式 (filesharing 和 surveys)，而且當建立社群時，它們可供使用者使用。

您可以使用下列程序升級、自訂及重新部署基於 SJWC 的 portlet 應用程式。filesharing portlet 應用程式做為範例使用：

1. 解壓縮發行版本 5 SJWC jar 檔案。
 - a. mkdir /tmp/lh
 - b. cd /tmp/lh

- c. `/usr/jdk/entsys-j2se/bin/jar xvf`
`PortalServer7-base/portlet/communityportlets.war`
`WEB-INF/lib/commons-beanutils.jar`
`WEB-INF/lib/commons-collections-3.1.jar`
`WEB-INF/lib/commons-digester.jar`
`WEB-INF/lib/commons-logging.jar`
`WEB-INF/lib/dataprovider.jar` `WEB-INF/lib/jsf-api.jar`
`WEB-INF/lib/jsf-impl.jar` `WEB-INF/lib/webui.jar`

如果 `PortalServer7-base/portlet/communityportlets.war`

未找到，請使用

`PortalServer7-base/portlet/core/communityportlets.war`。

- d. 重新命名其中一個檔案。

```
mv WEB-INF/lib/commons-collections-3.1.jar
WEB-INF/lib/commons-collections.jar
```

2. 找到 `filesharing portlet` 應用程式。


```
cd PortalServer7Config-base/portals/portal1/portletapps/filesharing
```
3. 加入經過更新的 SJWC 程式庫。


```
jar uvf src/filesharing.war.tokenized -C /tmp/lh WEB-INF
```
4. 自訂 `filesharing portlet` 應用程式。


```
ant customize
```
5. 重新部署 `filesharing portlet` 應用程式。
 - a. `PortalServer7-base/bin/psadmin undeploy-portlet -u amadmin`
`-f passwordfile -p portal_id -i instance_id -g filesharing`
 - b. `ant deploy`

- c. 轉至下列目錄 (視 Web 容器而定)：

Application Server 8.x :

*AppServer8Config-base/domains/domain1/applications/j2ee-modules/
communityportlets/WEB-INF*

Web Server 6.x :

*WebServer6-base/https-instanceName/webapps/https-instanceName/
communityportlets/WEB-INF*

Web Server 7.x :

*WebServer7Config-base/https-configName/web-app/https-configName/
communityportlets/WEB-INF*

- d. 開啓 sun-web.xml 檔案，並在最後一行之前 (也就是在 sun-web-app 結束標記之前) 加上以下這一行：

```
<class-loader delegate="false"/>
```

- e. 針對 filesharing/WEB-INF 底下的 sun-web.xml 檔案重複步驟 c 和步驟 d。

6. 針對 surveys 以及基於 SJWC 架構的其他任何自訂 portlet 應用程式重複步驟 2 到步驟 5。
7. 重新啓動 Web 容器。

更正書籤及應用程式通道中的連結

將發行版本 4 Portal Server 升級到發行版本 5 時，書籤及應用程式通道含有重複和虛假的連結。若要修正這些連結，請執行下列程序。

1. 登入 PSConsole。
2. 從 [一般作業] 標籤中，按一下 [管理通道與容器]。
3. 選擇 [DeveloperSample [Org]] 做為 DN，然後按一下 [確定]。
4. 選取 [JSPTabContainer [預設]] 做為 [檢視類型]。
5. 在 [MyFrontPageTabPanelContainer] 下，按一下 [應用程式通道]。

將會在右側的框架上顯示應用程式通道特性。

若要檢視特定語言環境的特性，請按一下 [表格喜好設定]，然後提供 [語言環境] 值：de、fr、es、ja、ko、zh、zh_CN、zh_TW。

6. 編輯 [userApps] 特性。
 - a. 在 [userApps] 特性中按一下 [編輯值...] 連結。
將會出現含有現有應用程式的快顯式視窗。
 - b. 向清單增加下列應用程式：
NetFile
 - c. 按一下 [儲存]，然後按 [關閉]。
7. 編輯 [target] 特性。
 - a. 在 [target] 特性中按一下 [編輯值...] 連結。
將會出現含有現有目標的快顯式視窗。
 - b. 向清單增加下列目標：
NetFile|/portal/NetFileApplet?Refer=java2
 - c. 按一下 [儲存]，然後按 [關閉]。

手動遷移基於 Struts 的 Portlet

如果您所擁有的自訂 portlet 使用以 Struts 架構為基礎的編碼，則需要手動更新這些 portlet 以使用發行版本 5 Portal Server 中包含的 struts.jar 檔案。請執行下列程序：

1. 取消部署基於 struts 的 portlet 應用程式。
`PortalServer7-base/bin psadmin undeploy-portlet`
2. 以正確版本的 struts.jar 檔案更新 .war 檔案。
將 `PortalServer7-base/lib/struts.jar` 複製到
`strutsbasedPortlet/WEB-INF/lib/struts.jar`
其中 `strutsbasedPortlet` 是基於 struts 的 portlet 檔案所在的目錄。
3. 建立 `strutsbasedPortlet` 目錄的 .war 歸檔。
4. 重新部署 portlet 應用程式。
`PortalServer7-base/bin psadmin deploy-portlet`

回復升級 (Solaris)

本節描述影響 Portal Server 升級回復程序的注意事項，接著提供程序本身。

回復注意事項 (Solaris)

回復到發行版本 5 的升級的程序包括在 *PortalServer7-base* 復原至 IFR 安裝以及重新部署 IFR Web 應用程式。

回復程序 (Solaris)

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 將 Directory Server 回復為它在升級前的狀態。

使用 Directory Server backup/restore 指令行及 GUI 公用程式。請參閱「Sun Java System Directory Server Enterprise Edition 6.0 管理指南」(<http://docs.sun.com/doc/820-0294>) 的「Directory Server 備份與復原」一章。

3. 取消部署在升級到發行版本 5 期間重新部署的發行版本 5 Portal Server Web 應用程式。

使用 Web 容器的管理公用程式 (指令行或主控台) 來取消下列套裝軟體的部署：

```
portal  
psconsole  
search1  
wsssoportlet  
guessnumber  
portletsamples
```

4. 停止 Portal Server，方法是停止它的 Web 容器。

Web Server 6.x :
WebServer-base/https-instanceName/stop

Web Server 7.0 :
Admin Server--
WebServer7Config-base/admin-server/bin/stopserv
Instance Server--
WebServer7Config-base/https-configName/bin/stopserv

Application Server 8.x :
AppServer8-base/bin/asadmin stop-domain --user admin_ID
--password password domainName

5. 撤銷表 15-8 中的 Portal Server 7.1 修補程式。

```
patchrm patch_ID
```

6. 重新啓動 Portal Server，方法是重新啓動它的 Web 容器。

```
Web Server 6.x :  
WebServer-base/https-instanceName/start
```

```
Web Server 7.0 :  
Admin Server--  
WebServer7Config-base/admin-server/bin/startserv  
Instance Server--  
WebServer7Config-base/https-configName/bin/startserv
```

```
Application Server 8.x :  
AppServer8-base/bin/asadmin start-domain --user admin_ID  
--password password domainName
```

7. 部署第 348 頁的步驟 3 中取消部署的發行版本 5 Portal Server Web 應用程式。

使用 Web 容器的管理公用程式 (指令行或主控台) 部署套裝軟體。

8. 停止 Web 容器，然後將其重新啓動。

儘管並不需要在所有情況下都重新啓動 Web 容器，但這樣做可確保 Portal Server 在未修改的狀態下啓動。

回復升級 (Linux)

無法在 Linux 上執行升級回復。

然而，您可以在嘗試進行升級之前，先建立一個平行系統，**然後**對該系統進行升級和測試。如果您需要回復升級，您可以回復至該平行系統。

多重實例升級

在某些部署架構中，會將 Portal Server 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Portal Server 元件在多台電腦上執行，並透過負載平衡器來分散負載。

若 Portal Server 實例已達負載平衡，則可以執行串聯式升級，即依序升級 Portal Server 實例而不中斷服務。升級 Portal Server 的每個實例時其他實例仍可繼續執行。若要執行從 IFR 到發行版本 5 的漸進式升級，除了將參照的發行版本 4 取代為 IFR 之外，其他請遵循如第 316 頁的「多重實例升級」中所述的程序。

Portal Server Secure Remote Access

本章描述如何將 Portal Server Secure Remote Access 升級到 Java ES 5 (發行版本 5) : Sun Java System Portal Server Secure Remote Access 7.1。

本章簡要介紹發行版本 5 所支援的不同升級路徑之升級注意事項，涵蓋在 Solaris 和 Linux 作業系統上的升級作業：

- [第 353 頁的「Portal Server Secure Remote Access 升級簡介」](#)
- [第 358 頁的「從 Java ES 發行版本 4 升級 Portal Server Secure Remote Access」](#)
- [第 375 頁的「從 Java ES 發行版本 3 升級 Portal Server Secure Remote Access」](#)
- [第 377 頁的「從 Java ES 發行版本 2 升級 Portal Server Secure Remote Access」](#)
- [第 380 頁的「從臨時功能發行版本 7.0 升級 Portal Server Secure Remote Access」](#)

備註 本章中的檔案位置是相對於名為 *PortalServer6-base* 和 *PortalServer6Config-base* (Portal Server 6.x) 以及 *PortalServer7-base* 和 *PortalServer7Config-base* (Portal Server 7.x) 的目錄路徑指定的。初始安裝 Portal Server 時可能至少已將這些路徑中的一部份指定為安裝目錄。若沒有指定，Java ES 安裝程式會指定一個預設值。

這些目錄路徑的預設值顯示在下表中。

表 16-1 Portal Server Secure Remote Access 目錄路徑

路徑名稱變數	Solaris OS	Linux OS
<i>PortalServer6-base</i>	/opt/SUNWps	/opt/sun/portal
<i>PortalServer6Config-base</i>	/etc/opt/SUNWps	/etc/opt/sun/portal
<i>PortalServer6Data-base</i>	/var/opt/SUNWps	/var/opt/sun/portal
<i>ortalServer7-base</i>	/opt/SUNWportal	/opt/sun/portal
<i>PortalServer7Config-base</i>	/etc/opt/SUNWportal	/etc/opt/sun/portal
<i>PortalServer7Data-base</i>	/var/opt/SUNWportal	/var/opt/sun/portal

Portal Server Secure Remote Access 升級簡介

本節描述下列會影響 Portal Server Secure Remote Access 升級至 Java ES 5 (發行版本 5) 的一般層面：

- [關於 Java ES 發行版本 5 Portal Server Secure Remote Access](#)
- [Portal Server Secure Remote Access 升級資訊指南](#)
- [Portal Server Secure Remote Access 資料](#)
- [Portal Server Secure Remote Access 升級策略](#)

關於 Java ES 發行版本 5 Portal Server Secure Remote Access

Portal Server Secure Remote Access (包括 Gateway、Rewriter Proxy、Netlet Proxy 元件) 與 Portal Server 緊密關聯，儘管通常會將其部署到與代管 Portal Server 的電腦不同的電腦上。Portal Server Secure Remote Access 元件使用與 Portal Server 相同的管理基礎架構，並與位於代管 Portal Server 的電腦上的 servlet 和 applet 互動。

Java ES 發行版本 5 Portal Server Secure Remote Access 相對於發行版本 4 而言為主要發行版本，包含許多新功能與增強功能。許多變更都是在發行版本 4 隨後的「臨時功能發行版本」(IFR) 中進行的。相對於 IFR 而言，發行版本 5 僅為次要的功能變更。如需有關 IFR 增強功能及新功能的資訊，請參閱「Sun Java System Portal Server 7.1 版本說明」

(<http://docs.sun.com/app/docs/doc/820-0898/6ncjuat45?a=view>)。特別是，發行版本 4 命令行管理介面已由 psadmin 指令所取代。

Portal Server Secure Remote Access 升級資訊 指南

表 16-2 顯示支援的 Portal Server Secure Remote Access 至 Java ES 發行版本 5 的升級路徑。該表適用於 Solaris 和 Linux 作業系統。

表 16-2 至 Java ES 5 (發行版本 5) 的升級路徑：Portal Server Secure Remote Access 7.1

Java ES 發行版本	Portal Server Secure Remote Access 版本	一般方法	需要重新配置
臨時功能 發行版本 (IFR)	Sun Java System Portal Server Secure Remote Access IFR 7.0 2005Q4	直接升級： 藉由套用修補程式然後使用升級 程序檔執行。	無
發行版本 4	Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4	直接升級： 使用升級程序檔執行。	無
發行版本 3	Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q1	直接升級： 使用升級程序檔執行。	無
發行版本 2	Sun Java System Portal Server Secure Remote Access 6.3 2004Q2	直接升級： 使用升級程序檔執行。	無
發行版本 1	Sun ONE Portal Server Secure Remote Access 6.2 (2003Q4)	無直接升級： 但可以藉由先升級到發行版本 3，然後再從發行版本 3 升級到 發行版本 5 來執行。	配置資料
Java ES 發行版本 之前的發 行版本		無直接升級。	

Portal Server Secure Remote Access 資料

下表顯示可能受升級 Portal Server Secure Remote Access 軟體影響的資料類型。

表 16-3 Portal Server Secure Remote Access 資料用途

資料類型	位置	用途
配置資料	<i>PortalServer6Config-base/</i>	Portal Server Secure Remote Access 的配置。
目錄模式 服務配置 使用者資料	Directory Server	Portal Server Secure Remote Access 依賴服務配置 (如入口桌面) 和儲存在目錄中的使用者設定檔資料。
動態應用程式資料	無	Portal Server Secure Remote Access 不會永久性地儲存應用程式資料 (像是階段作業狀態)。

Portal Server Secure Remote Access 升級策略

Portal Server Secure Remote Access 的升級策略一般取決於在 **第 1 章「升級規劃」** 中介紹的許多考量因素：升級路徑、Java ES 元件之間的相依性、選擇性升級與整體升級之差異、多重實例部署等。

本節將會提出可能影響 Portal Server Secure Remote Access 升級規劃的問題，藉以對 Portal Server Secure Remote Access 進行一般性的介紹。

相容性問題

發行版本 5 Portal Server Secure Remote Access 介紹了用來啟動和停止 Gateway、Rewriter Proxy 與 Netlet Proxy 元件之 psadmin 指令中的公共介面變更。請參閱「Sun Java System Portal Server 7.1 Command-Line Reference」(<http://docs.sun.com/doc/819-5030>)。

個別 Portal Server Secure Remote Access 元件 (包括 Gateway、Rewriter Proxy 與 Netlet Proxy) 並不向下相容較早版本；在 Java ES 發行版本 5 中，所有元件都需要與 Portal Server 本身同步。此需求適用於本機及分散式 Portal Server Secure Remote Access 元件。

此外，發行版本 5 Portal Server 與較早 Portal Server 版本所使用的 Directory Server 資料架構之間存在不相容性。此不相容性會影響使用相同 Directory Server 資料之多重 Portal Server 實例執行的漸進式升級。

Portal Server Secure Remote Access 相依性

視配備有 Portal Server 且與 Portal Server 一樣在相同電腦上執行的軟體而定，Portal Server Secure Remote Access 會與 Portal Server 緊密關聯。

但是，Portal Server Secure Remote Access 還將依賴於其他 Java ES 元件。這些相依性可能會影響升級和重新配置 Portal Server Secure Remote Access 軟體的程序。例如，Portal Server Secure Remote Access 介面或功能上的變更可能需要 Portal Server Secure Remote Access 相依之元件的升級版本。升級此類元件的需求取決於具體的升級路徑。

Portal Server Secure Remote Access 元件對下列 Java ES 元件有相依性：

- **共用元件。** Portal Server Secure Remote Access 元件對特定的 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。
- **Portal Server。** Portal Server Secure Remote Access 元件對 Portal Server 有強制性的相依性，它包含支援 Portal Server Secure Remote Access 功能所需的本機元件。
- **Access Manager (或 Access Manager SDK)。** Portal Server Secure Remote Access 元件對 Access Manager 有強制性的相依性，可為一般使用者提供認證和授權服務，包括單次登入。如果 Access Manager 是在遠端電腦上執行，則 Access Manager SDK 在本機上必須可用。
- **Directory Server。** Portal Server Secure Remote Access 對 Directory Server 有強制性的相依性，後者用於儲存使用者資料。因此，Portal Server Secure Remote Access 升級可能需要延伸目錄模式。

選擇性的升級問題

一般而言，儘管 Java ES 發行版本 5 支援電腦上所有元件的選擇性升級，而事實上，Portal Server Secure Remote Access 與 Portal Server 緊密連結在一起，這表示如果升級了 Portal Server，也必須升級 Portal Server Secure Remote Access。同樣地，升級 Portal Server Secure Remote Access 需要同時升級 Portal Server。

因此，升級 Portal Server Secure Remote Access 與升級 Portal Server 的限制相同 (請參閱 Portal Server 第 291 頁的「選擇性的升級問題」)：您可以將 Portal Server Secure Remote Access 及其**所有**產品元件相依性都升級到發行版本 5，或只將 Portal Server Secure Remote Access 和 Portal Server 升級到發行版本 5，而將其他產品元件相依性保留為發行版本 4。

雙重升級

在雙重升級中，Portal Server Secure Remote Access 與作業系統均會進行升級 (如第 40 頁的「[雙重升級：Java ES 與作業系統軟體](#)」中所述)，這種升級可採用原地升級作業系統的方法來執行：

1. 備份現有的 Portal Server Secure Remote Access 資料。

如需必要資料的位置，請參閱第 355 頁的「[Portal Server Secure Remote Access 資料](#)」。

2. 升級作業系統。

升級時會原樣保留現有的檔案系統。

3. 升級到發行版本 5 Portal Server Secure Remote Access。

請根據升級路徑，參閱本章的相應小節。

從 Java ES 發行版本 4 升級 Portal Server Secure Remote Access

本節包括將 Portal Server Secure Remote Access 從 Java ES 2005Q4 (發行版本 4) 升級到 Java ES 5 (發行版本 5) 的相關資訊。

本節涵蓋以下主題：

- [簡介](#)
- [升級發行版本 4 Portal Server Secure Remote Access](#)
- [多重實例升級](#)

簡介

將 Java ES 發行版本 4 Portal Server Secure Remote Access 升級到發行版本 5 時，請考慮升級程序的下列層面：

- **一般升級方法。** 使用升級程序檔 `psupgrade` 來執行升級。此程序檔會移除舊套裝軟體、安裝新套裝軟體，必要時還將遷移配置資料。
- **升級相依性。** Portal Server Secure Remote Access 對一些 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)。雖然發行版本 5 Portal Server Secure Remote Access 與這些共用元件的發行版本 4 版本相容，但共用元件的升級仍然是必要的，這是因為用來升級 Portal Server Secure Remote Access 的 `psupgrade` 程序檔需要使用 ANT 共用元件的發行版本 5 版本。

發行版本 5 Portal Server Secure Remote Access 還對 Portal Server、Access Manager 和 Directory Server 有相依性，如第 356 頁的「[Portal Server Secure Remote Access 相依性](#)」中所述。支援升級這些相依性的兩種方法 (請參閱第 356 頁的「[選擇性的升級問題](#)」)。

- **升級相依性。** 儘管 Portal Server Secure Remote Access 對許多 Java ES 共用元件有相依性 (請參閱第 43 頁的表 1-9)，但是，發行版本 5 Portal Server Secure Remote Access 卻與這些元件的發行版本 4 版本均相容。因此，對將 Portal Server Secure Remote Access 升級到發行版本 5 而言，升級這些共用元件是選擇性的。

但是，發行版本 5 Portal Server Secure Remote Access 只對 Portal Server 有硬性升級相依性。發行版本 5 Portal Server Secure Remote Access 還對 Access Manager 和 Directory Server 有非硬性升級相依性，如第 356 頁的「[Portal Server Secure Remote Access 相依性](#)」中所述。

支援升級這些產品元件相依性的兩種方法 (請參閱第 356 頁的「選擇性的升級問題」)：

- 所有相依性均符合發行版本 4 元件 (除了 Portal Server 之外沒有任何一項會升級到發行版本 5)
- 所有相依性均符合發行版本 5 元件 (所有項目都會升級到發行版本 5)。

Portal Server Secure Remote Access 採用的方法必須與 Portal Server 採用的方法相同。

- **向下相容性。** 發行版本 5 Portal Server Secure Remote Access 向下相容發行版本 4 版本。
- **升級回復。** 將 Portal Server Secure Remote Access 的發行版本 5 升級回復到發行版本 4 包括復原發行版本 4 套裝軟體和復原發行版本 4 目錄資料。
- **平台問題。** 升級 Portal Server Secure Remote Access 的一般方法在 Solaris 及 Linux 作業系統上均相同，但在 Solaris OS 上，發行版本 5 Portal Server Secure Remote Access 安裝在新路徑中，而在 Linux OS 上的安裝路徑則與發行版本 4 相同。

升級發行版本 4 Portal Server Secure Remote Access

本節描述如何在 Solaris 及 Linux 平台上將 Portal Server Secure Remote Access 從 Java ES 發行版本 4 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [發行版本 4 升級前作業](#)
- [升級發行版本 4 Portal Server Secure Remote Access \(Solaris\)](#)
- [升級發行版本 4 Portal Server Secure Remote Access \(Linux\)](#)
- [驗證升級](#)
- [發行版本 4 升級後作業](#)
- [回復升級 \(Solaris\)](#)
- [回復升級 \(Linux\)](#)

發行版本 4 升級前作業

升級 Portal Server Secure Remote Access 之前，應該執行下列作業：

- [驗證目前版本資訊](#)
- [升級 Portal Server Secure Remote Access 相依元件](#)
- [備份 Portal Server Secure Remote Access 發行版本 4 配置資訊](#)
- [移除負載平衡器的配置](#)
- [移除 Directory Proxy Server 配置](#)
- [取得必要的配置資訊和密碼](#)

驗證目前版本資訊

您可使用下列指令驗證 Portal Server Secure Remote Access 的目前版本：

```
PortalServer6-base/bin/version
```

表 16-4 Portal Server Secure Remote Access 版本驗證輸出

Java ES 發行版本	Portal Server Secure Remote Access 版本編號
發行版本 2	6.3
發行版本 3	6.3.1
發行版本 4	6.3.1 ¹
IFR 發行版本	7.0
發行版本 5	7.1

1. 發行版本 3 與發行版本 4 的唯一差別在於修補程式。您可使用 `Solaris showrev -p | grep patch_ID` 指令和 `Linux rpm -qa sun-portal-core` 指令，並將版本與「Java ES 發行版本 4 升級指南」中列出的版本進行比較，來檢查發行版本 4 修補程式。

升級 Portal Server Secure Remote Access 相依元件

一般的建議是將電腦系統 (及運算環境) 中的所有 Java ES 元件都升級到 Java ES 發行版本 5。

雖然發行版本 5 Portal Server Secure Remote Access 與 Java ES 共用元件的發行版本 4 版本相容，但共用元件的升級仍然是必要的，這是因為用來升級 Portal Server Secure Remote Access 的 `psupgrade` 程序檔需要使用 ANT 共用元件的發行版本 5 版本。

此外，Portal Server Secure Remote Access 要求升級 Portal Server。但是，它不需要升級它所依賴的其他 Java ES 發行版本 4 產品元件。

事實上，相依性升級方法與針對 Portal Server 所採用的方法相同：如果要將任何相依性升級到發行版本 5，需要將它們全部升級 (請參閱第 291 頁的「選擇性的升級問題」)。但是，由於 Portal Server Secure Remote Access 對 Portal Server 的相依性，除了共用元件之外，Portal Server 的升級還要考慮 Portal Server Secure Remote Access 相依性。

當將 Portal Server Secure Remote Access 相依性升級到發行版本 5 時，您應在升級 Portal Server Secure Remote Access 之前按下列順序升級相依性 (跳過任何可能已經升級的部份)。

1. **共用元件。** 在第 59 頁的「升級 Java ES 共用元件」中提供了將 Java ES 共用元件與發行版本 5 同步的指示。
2. **Portal Server。** 在第 285 頁的第 15 章「Portal Server」中提供了升級 Portal Server 的指示。

備份 Portal Server Secure Remote Access 發行版本 4 配置資訊

將 Portal Server Secure Remote Access 升級到發行版本 5 不需要重新配置 Portal Server Secure Remote Access 軟體。不過，為安全起見，psupgrade 程序檔將會備份下列儲存配置資訊的目錄：

PortalServer6Config-base/

移除負載平衡器的配置

如果遇到透過負載平衡器存取 Portal Server Secure Remote Access 實例的情況，用來配置此類存取的 LOAD_BALANCER_URL 特性的值會干擾 Portal Server Secure Remote Access 升級。因此，在執行任何 Portal Server Secure Remote Access 元件的升級之前，必須修改此設定。若要修改 LOAD_BALANCER_URL 特性設定：

1. 請注意下列哪些配置檔案位於本機上 (其中一些支援可以在本機安裝的 Portal Server 元件)：

PortalServer6Config-base/PSConfig.properties (如果 Portal Server 在本機)
PortalServer6Config-base/GWConfig.properties (如果 Gateway 在本機)
PortalServer6Config-base/RWPConfig.properties (如果 Rewriter Proxy 在本機)
PortalServer6Config-base/NLPConfig.properties (如果 Netlet Proxy 在本機)

2. 將 LOAD_BALANCER_URL 特性的目前值記錄在這些配置檔案中。
3. 修改 LOAD_BALANCER_URL 特性的值，以指向要升級的對應 Portal Server Secure Remote Access 實例：

`LOAD_BALANCER_URL=hostName:port/portal`

4. 確定下列配置特性 (如果存在) 參照相關的 Portal Server Secure Remote Access 元件 (而不是負載平衡器)，如下所示：

在 *PortalServer6Config-base/platform.conf.default* 檔案中：

```
gateway.host=Gateway_hostName
```

在 *PortalServer6Config-base/GWConfig.properties* 與
PortalServer6Config-base/GWConfig-default.properties 檔案中：

```
GW_HOST=Gateway_hostName  
GW_IP=Gateway_hostIP
```

在 *PortalServer6Config-base/RWPConfig.properties* 與
PortalServer6Config-base/RWPConfig-default.properties 檔案中：

```
RWP_HOST=RewriterProxy_hostName  
RWP_IP=RewriterProxy_hostIP
```

在 *PortalServer6Config-base/NLPConfig.properties* 與
PortalServer6Config-base/NLPConfig-default.properties 檔案中：

```
NLP_HOST=NetLetProxy_hostName  
NLP_IP=NetLetProxy_hostIP
```

移除 Directory Proxy Server 配置

如果遇到 Portal Server Secure Remote Access 實例透過 Directory Proxy Server 實例存取 Directory Server 的情況，則在執行升級之前必須修改 Directory Proxy Server 主機與連接埠號設定，然後再在升級完成之後復原為原始值。

修改適當設定：

1. 將 DS_HOST 和 DS_PORT 特性的目前值記錄在下列 Access Manager 配置檔案中：
AccessManagerConfig-base/config/AMConfig.properties
2. 修改 DS_HOST 和 DS_PORT 特性的值，以直接指向相關 Directory Server 實例。

取得必要的配置資訊和密碼

視升級方案而定，psupgrade 程序檔會要求您輸入下列管理帳號的相關資訊：

- Directory Server 管理 ID 和密碼
- Access Manager 管理 ID 和密碼
- Directory Server amldapuser ID 和密碼

升級發行版本 4 Portal Server Secure Remote Access (Solaris)

本節討論影響 Portal Server Secure Remote Access 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項 (Solaris)

將 Portal Server Secure Remote Access 軟體升級到發行版本 5 時要考慮下列注意事項：

- Portal Server Secure Remote Access 軟體包括執行一些不同角色的子元件，但是必須全部一起升級到發行版本 5：
 - **Portal-base**。包括管理 Mbeans 及隨附的管理軟體、登入架構及與監視相關的軟體，它們全部都套裝在 SUNWportal-base 套裝軟體中。
 - **Secure Remote Access 應用程式**。包括 Gateway、Rewriter Proxy 和 Netlet Proxy。通常會將這些應用程式部署到與代管 Portal Server 的電腦不同的一台或多台電腦上。Secure Remote Access 應用程式不需要 Web 容器。
- 如果未將 Gateway、Rewriter Proxy 與 Netlet Proxy 部署到同一台電腦上，則應該在升級 Gateway 之前先升級 Rewriter Proxy 和 Netlet Proxy。
- 所有 Portal Server Secure Remote Access 子元件都對應於同一安裝的 Portal Server Secure Remote Access 影像，如果存在於升級的電腦上，會同時升級它們。
- psupgrade 程序檔會自動偵測主機上安裝了哪些 Portal Server Secure Remote Access 子元件，並升級這些元件。

升級程序 (Solaris)

下述程序適用於位於執行升級的電腦上的 Portal Server Secure Remote Access 元件。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。

[第 59 頁的第 2 章「升級 Java ES 共用元件」](#)中提供了指示。

對於在[第 364 頁的步驟 8](#)中執行 psupgrade 程序檔來說，此步驟是必要的先決條件。

3. 停止正在本地執行的 Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxyd stop
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
Rewriter Proxy : netstat -an | grep 10443
Netlet Proxy : netstat -an | grep 10555
```

4. 確定 Access Manager 正在執行。
5. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 和 JAVA_HOME)。例如，

```
export ANT_HOME=/usr/sfw
export JAVA_HOME=/usr/jdk/entsys-j2se
```

6. 請確定您電腦上有足夠的交換空間。
做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。
7. 如果正在升級的 Portal Server Secure Remote Access 元件在 Portal Server 的遠端，請將 dpadmin 可執行檔從代管 Portal Server 的電腦複製到代管 Portal Server Secure Remote Access 元件的電腦。

您可以在以下位置找到 dpadmin 可執行檔：

```
PortalServer7-base/SUNWps.bak/bin/dpadmin，如果已升級 Portal Server。
PortalServer6-base/bin/dpadmin，如果尚未升級 Portal Server。
```

8. 從 Java ES 發行版本 5 發行物中執行 psupgrade 程序檔。

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

其中 *os_arch* 與您所在平台相符，例如 *Solaris_sparc*。

psupgrade 程序檔會呼叫 Java ES 安裝程式來安裝新的套裝軟體，並請求以下資訊：

- Directory Server 管理 ID 和密碼
- Access Manager 管理 ID 和密碼
- Directory Server amldapuser ID 和密碼

9. 啟動在[第 364 頁的步驟 3](#)中停止的 Gateway、Rewriter Proxy 或 Netlet Proxy 的實例。

```
PortalServer7-base/bin/psadmin start-sra-instance -u amadminUser
-f passwordFile --name default --type gateway
```

```
PortalServer7-base/bin/psadmin start-sra-instance -u amadminUser
-f passwordFile --name default --type nlproxy
```

```
PortalServer7-base/bin/psadmin start-sra-instance -u amadminUser
-f passwordFile --name default --type rwproxy
```

如果上述指令失敗，您必須先註冊 (啟用) Portal Server Secure Remote Access 元件：

```
PortalServer7-base/bin/psadmin provision-sra -u amadminUser
-f passwordFile -p Portal_ID --gateway-profile profileName --enable
```

升級發行版本 4 Portal Server Secure Remote Access (Linux)

本節討論影響 Portal Server Secure Remote Access 升級程序的注意事項，接著提供對程序本身的說明。

升級注意事項 (Linux)

在 Linux OS 上，發行版本 5 Portal Server Secure Remote Access 的安裝路徑與發行版本 4 相同，除了這一點外，在 Linux 平台和 Solaris 平台上將 Portal Server Secure Remote Access 軟體升級到發行版本 5 所考慮的注意事項相同 (請參閱[第 363 頁的「升級注意事項 \(Solaris\)」](#))。因此，當安裝發行版本 5 RPM 時，psupgrade 程序檔會移除之前的 RPM。

升級程序 (Linux)

下述程序適用於位於執行升級的電腦上的 Portal Server Secure Remote Access。

注意 在 Linux 上執行的從 Java ES 發行版本 4 至發行版本 5 的升級無法回復。在執行下列程序**之前**，請務必先備份您的系統。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 如果尚未執行此動作，請將所有共用元件與發行版本 5 同步。

第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了指示。

對於在第 366 頁的步驟 8 中執行 `psupgrade` 程序檔來說，此步驟是必要的先決條件。

3. 停止正在本地執行的 Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxyd stop
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
Rewriter Proxy : netstat -an | grep 10443
Netlet Proxy : netstat -an | grep 10555
```

4. 確定 Access Manager 正在執行。
5. 設定 `psupgrade` 程序檔所需的兩個環境變數 (`ANT_HOME` 和 `JAVA_HOME`)。例如，

```
export ANT_HOME=/opt/sun
export JAVA_HOME=/usr/jdk/entsys-j2se
```

6. 請確定您電腦上有足夠的交換空間。
做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。
7. 如果正在升級的 Portal Server Secure Remote Access 元件在 Portal Server 的遠端，請將 `dpadmin` 可執行檔從代管 Portal Server 的電腦複製到代管 Portal Server Secure Remote Access 元件的電腦。

您可以在以下位置找到 `dpadmin` 可執行檔：

PortalServer7-base/SUNWps.bak/bin/dpadmin，如果已升級 Portal Server。

PortalServer6-base/bin/dpadmin，如果尚未升級 Portal Server。

8. 從 Java ES 發行版本 5 發行物中執行 `psupgrade` 程序檔。

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

其中 `os_arch` 與您所在平台相符，例如 `Solaris_sparc`。

`psupgrade` 程序檔會呼叫 Java ES 安裝程式來安裝新的套裝軟體，並請求以下資訊：

- Directory Server 管理 ID 和密碼

- Access Manager 管理 ID 和密碼
- Directory Server aldapuser ID 和密碼

驗證升級

如果您正在升級的 Portal Server Secure Remote Access 元件在 Portal Server 的遠端，則可透過檢查以下檔案中的版本資訊來驗證發行版本 5 套裝軟體的安裝：

```
PortalServer7-base/lib/PSversion.properties
```

不過，如果您正在升級的 Portal Server Secure Remote Access 元件與 Portal Server 位於同一電腦上，則可以使用以下指令來驗證升級：

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID
-f adminpasswordfile.
```

請參閱第 360 頁的表 16-4 中的輸出值。

您也可以檢查位於以下位置的升級記錄檔：

```
/var/sadm/install/logs/Sun_Java_System_Portal_Server_upagrede.log
```

發行版本 4 升級後作業

除了以下情況外，將 Portal Server Secure Remote Access 升級到發行版本 5 時不需要升級後作業：

- 復原負載平衡器的配置
- 復原 Directory Proxy Server 配置
- 刪除發行版本 4 本土化提供者

復原負載平衡器的配置

如果已經透過負載平衡器存取了 Portal Server Secure Remote Access 實例，則需要在升級之後執行以下步驟，以復原負載平衡器配置：

1. 在 `PortalServer7Config-base/platform.conf.default` 檔案中設定以下參數：

```
gateway.virtualhost=loadBalancer_hostName loadBalancer_hostIP
gateway.external.ip=loadBalancer_hostIP
gateway.dsame.agent=http\://loadBalancer_hostName\ :
80/portal/RemoteConfigServlet
```

2. 在 `PortalServer7Config-base/GWConfig-default.properties` 檔案中設定以下參數。

```
gateway.ipaddress=Gateway_hostIP
```

3. 當將這些實例部署到 Portal Server 主機的遠端電腦上時，請針對 Rewriter Proxy 和 Netlet Proxy 設定對應於步驟 1 與步驟 2 的參數。
4. 重新啓動 Portal Server 和負載平衡的 Portal Server Secure Remote Access 實例。

復原 Directory Proxy Server 配置

如果 Portal Server Secure Remote Access 實例已經透過 Directory Proxy Server 實例存取了 Directory Server，則必須在升級前將 Directory Proxy Server 主機與連接埠號設定復原爲其原始值。請參閱第 362 頁的「[移除 Directory Proxy Server 配置](#)」，在此過程中修改了這些特性的值以備升級之用。

刪除發行版本 4 本土化提供者

本土化 Proxylet 服務只有在您刪除發行版本 4 本土化提供者之後才會載入，如下所示：

1. 轉至 `PortalServer7Data-base/portals/Upgraded/desktop` 目錄。
2. 從 `default_Locale` 中刪除除以下內容以外的所有目錄與檔案：
 - 您已經建立的檔案目錄 (不是 Portal Server Secure Remote Access 隨附的檔案目錄)
 - `message.properties` 檔案
 - 以下目錄：

- `AddressBookProvider`
- `BookmarkProvider`
- `CalendarProvider`
- `LoginProvider`
- `LotusNotesAddressBookProvider`
- `LotusNotesCalendarProvider`
- `LotusNotesMailProvider`
- `MSExchangeAddressBookProvider`
- `MSExchangeCalendarProvider`
- `MSExchangeMailProvidervMailProvider`
- `NotesProvider`
- `PersonalNoteProvider`
- `Register`
- `SampleRSS`
- `SampleURLScrapper`
- `SampleXML`
- `TemplateEditContainerProvider`
- `TemplateTabContainerProvider`
- `URLScrapperProvider`


```
UWCAddressBookProvider
UserInfo
UserInfoProvider
XMLProvider
error
```

3. 重新啟動部署 Portal Server 的 Web 容器。

回復升級 (Solaris)

本節描述影響 Portal Server Secure Remote Access 升級回復程序的注意事項，接著提供程序本身。

回復注意事項 (Solaris)

回復至發行版本 5 的升級程序包括復原至 *PortalServer6-base* 的發行版本 4 安裝。

回復程序 (Solaris)

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 將 Directory Server 回復為它在升級前的狀態。

使用 Directory Server backup/restore 指令行及 GUI 公用程式。請參閱「Sun Java System Directory Server Enterprise Edition 6.0 管理指南」(<http://docs.sun.com/doc/820-0294>) 的「Directory Server 備份與復原」一章。

3. 移除發行版本 5 Portal Server Secure Remote Access 套裝軟體。
 - a. 啟動 Java ES 解除安裝程式。


```
/var/sadm/prod/SUNWentsys5/uninstall
```
 - b. 選取所有已經安裝的 Portal Server Secure Remote Access 元件。
 - c. 確認解除安裝選擇。
 - d. 結束 Java ES 解除安裝程式。
4. 將 *PortalServer6-base* 與 *PortalServer6Config-base* 目錄復原至它們的原始位置。升級期間，它們會移動至副檔名為 *.bak* 的目錄中。

回復升級 (Linux)

由於升級到發行版本 5 需要移除發行版本 4 二進位檔，因此在 Linux 上很難回復升級。

回復的一個方法是在升級及測試系統之前建立一個平行系統，然後再嘗試進行升級。如果您需要回復升級，您可以回復至該平行系統。

多重實例升級

在某些部署架構中，會將諸如 Gateway 等 Portal Server Secure Remote Access 元件部署在多個電腦系統上，以提供安全性與延展性，並提高可用性。例如，可以讓 Gateway 元件在多台電腦上執行，並透過負載平衡器來分散負載。

若 Gateway 實例已達負載平衡，則可以執行漸進式升級，即依序升級 Gateway 實例而不中斷服務，如下所述。此程序應考慮以下限制：個別 Portal Server Secure Remote Access 元件不向下相容較早版本；在 Java ES 發行版本 5 中，所有項目都需要與 Portal Server 同步。但是，在漸進式升級期間，發行版本 4 Portal Server Secure Remote Access 實例可以在升級 Portal Server 實例時保持繼續執行。

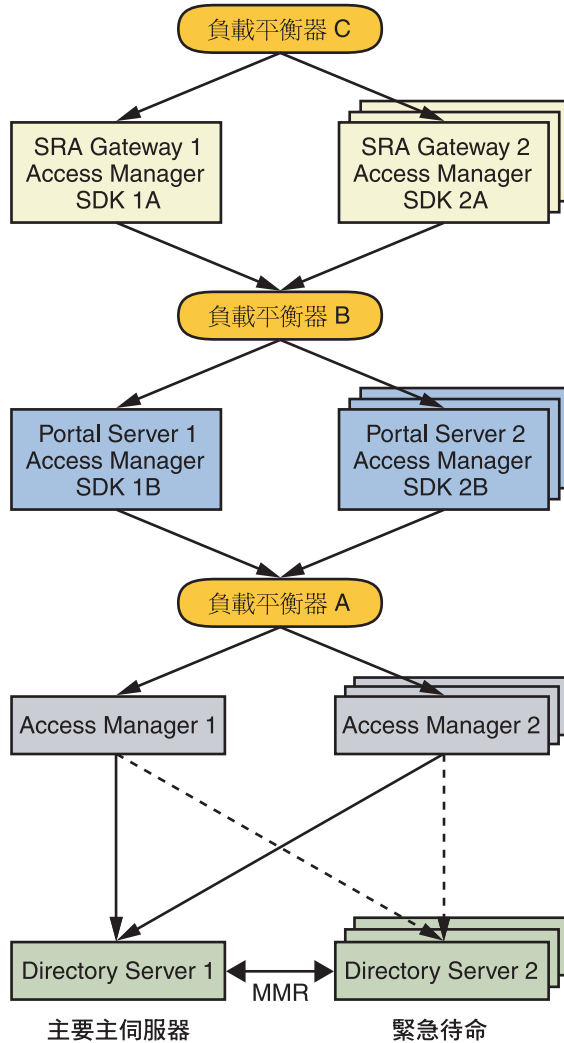
將使用第 371 頁的圖 16-1 中所示的部署架構來說明漸進式升級的程序。

在此架構中，可藉由 Portal Server Secure Remote Access Gateway 實例來存取多個 Portal Server 實例。可使 Portal Server 實例與 Gateway 實例達到負載平衡，以提供可用性與延展性。

相反，Portal Server 實例可透過負載平衡器來存取 Access Manager 實例。Access Manager 和 Access Manager SDK 實例會存取已針對多個主伺服器複製 (MMR) 設定的目錄。雖然其他 Directory Server 複製方案也是允許的，但 MMR 是高可用性與延展性目錄服務的代表。

在圖 16-1 中，已將 Gateway、Portal Server、Access Manager 和 Directory Server 的多重實例組成群組，以便於對升級程序進行說明。例如，Portal Server 2 就代表 Portal Server 的第二個到第 n 個實例。

圖 16-1 多重 Portal Server 實例的部署架構範例



發行版本 4 Gateway (和 Portal Server) 到發行版本 5 的漸進式升級的執行方式如下：

1. 如果您要將發行版本 4 Access Manager 升級到發行版本 5，請執行漸進式升級，如第 263 頁的「多重實例升級」所述。請注意，在將發行版本 4 Gateway 或發行版本 4 Portal Server 升級到發行版本 5 的過程中，您無須將發行版本 4 Access Manager 升級到發行版本 5。

2. 按以下方式修改 Portal Server 與 Gateway 實例的配置。
 - a. 將 Portal Server 2 配置為指向 Directory Server 2，而非指向 Directory Server 1。

為了簡潔起見，在此步驟及後續步驟中，「Portal Server 2」將表示 Portal Server 2 至 Portal Server n。
 - b. 將 Gateway 2 配置為指向 Directory Server 2，而非指向 Directory Server 1。

為了簡潔起見，在此步驟及後續步驟中，「Gateway 2」將表示 Gateway 2 至 Gateway n。
3. 升級 Portal Server 1。
 - a. 停用 Load Balancer B 中的 Portal Server 1。

請求不會再被路由至 Portal Server 1。
 - b. 停用 Directory Server MMR。

Directory Server 2 將不再與 Directory Server 1 同步。
 - c. 將 Access Manager SDK 1B 升級到發行版本 5。

請使用第 265 頁的「[僅升級發行版本 4 Access Manager SDK](#)」中的程序。
 - d. 將 Portal Server 1 升級到發行版本 5。

執行 Portal Server 實例的升級，如第 359 頁的「[升級發行版本 4 Portal Server Secure Remote Access](#)」所述，並請注意下列項目：

- 請特別注意下列升級前作業：[第 361 頁的「\[移除負載平衡器的配置\]\(#\)」](#)。
- 執行升級前，請確認 `AccessManagerConfig-base/config/AMConfig.properties` 檔案中的 `am.encrypted.pwd` 的值對於本機 Access Manager SDK 和與其相關的遠端 Access Manager 實例而言是相同的。
- 確定為要升級的每一個 Portal Server 實例的 `psupgrade` 所請求的 Portal Instance ID 參數提供了非空、唯一值。

將 Directory Server 1 的 Portal Server 資料升級到發行版本 5。

4. 升級 Gateway 1。
 - a. 停用 Load Balancer C 中的 Gateway 1。
請求不會再被路由至 Gateway 1。
 - b. 將 Access Manager SDK 1A 升級到發行版本 5。
請使用第 265 頁的「僅升級發行版本 4 Access Manager SDK」中的程序。
 - c. 將 Gateway 1 升級到發行版本 5。
按第 359 頁的「升級發行版本 4 Portal Server Secure Remote Access」中所述執行 Gateway 的升級，並注意下列項目：
 - 請特別注意下列升級前作業：第 361 頁的「移除負載平衡器的配置」。
 - 執行升級前，請確認 `AccessManagerConfig-base/config/AMConfig.properties` 檔案中的 `am.encrypted.pwd` 的值對於本機 Access Manager SDK 和與其相關的遠端 Access Manager 實例而言是相同的。
5. 按如下方式在各自的負載平衡器中啓用之前停用的 Portal Server 1 與 Gateway 1：
 - a. 啓用 Load Balancer B 中的 Portal Server 1。
請求會重新路由至 Portal Server 1。
 - b. 啓用 Load Balancer C 中的 Gateway 1。
請求會重新路由至 Gateway 1。
6. 按如下方式在各自的負載平衡器中停用 Portal Server 2 與 Gateway 2：
 - a. 停用 Load Balancer B 中的 Portal Server 2。
請求不會再被路由至 Portal Server 2。
 - b. 停用 Load Balancer C 中的 Gateway 2。
請求不會再被路由至 Gateway 2。
7. 升級 Portal Server 2。
 - a. 將 Portal Server 2 的配置復原為指向 Directory Server 1。
 - b. 將 Access Manager SDK 2B 升級到發行版本 5。
請使用第 265 頁的「僅升級發行版本 4 Access Manager SDK」中的程序。

- c. 將 Portal Server 2 升級到發行版本 5。
與升級 Portal Server 1 時 (第 372 頁的步驟 d) 使用相同的程序。
 - d. 啓用 Load Balancer B 中的 Portal Server 2。
請求會重新路由至 Portal Server 2。
8. 升級 Gateway 2。
- a. 將 Gateway 2 的配置復原為指向 Directory Server 1。
 - b. 將 Access Manager SDK 2A 升級到發行版本 5。
請使用第 265 頁的「僅升級發行版本 4 Access Manager SDK」中的程序。
 - c. 將 Gateway 2 升級到發行版本 5。
與升級 Gateway 1 時 (第 373 頁的步驟 c) 使用相同的程序。
 - d. 啓用 Load Balancer C 中的 Gateway 2。
請求會重新路由至 Gateway 2。
9. 啓用 Directory Server MMR。

現在，Directory Server 2 的 Portal Server 資料便與 Directory Server 1 同步。

備註 在漸進式升級方案中，Portal Server 實例將升級到發行版本 5，而 Gateway 元件的較早版本仍在使用中 (上述程序與此情況不同)，且將透過負載平衡器來存取 Gateway 實例，您應對所有 Gateway 實例進行檢查，確定 *PortalServer6Config-base/GWConfig.properties* 檔案和 *GWConfig-default.properties* 檔案中的以下配置特性參照的是 Gateway 而不是負載平衡器：

```
GW_IP=Gateway_hostIP
GW_HOST=Gateway_hostName
```

如果這些特性指向負載平衡器，Gateway 將不再存取已升級的 Portal Server 實例。

從 Java ES 發行版本 3 升級 Portal Server Secure Remote Access

將 Java ES 2005Q1 (發行版本 3) Portal Server Secure Remote Access 升級到發行版本 5 的程序與將發行版本 4 Portal Server Secure Remote Access 升級到發行版本 5 的程序相同，但有以下例外：

- 升級 [Portal Server Secure Remote Access](#) 相依元件
- 升級發行版本 3 [Portal Server Secure Remote Access](#)
- 多重實例升級

升級 Portal Server Secure Remote Access 相依元件

但是，當從發行版本 3 升級 Portal Server Secure Remote Access 時，您必須在升級 Portal Server Secure Remote Access 之前將 Access Manager 升級到發行版本 4 或發行版本 5，而且您不能在發行版本 3 上保留任何其他相依元件，也不能將某些相依元件升級到發行版本 4，並將其他相依元件升級到發行版本 5。如需更多資訊，請參閱第 356 頁的「選擇性的升級問題」。

下列相依元件必須按以下顯示的順序升級。

1. **共用元件**。 在第 59 頁的第 2 章「[升級 Java ES 共用元件](#)」中提供了將 Java ES 共用元件升級到發行版本 5 的指示。
2. **Directory Server**。 在第 108 頁的「[從 Java ES 發行版本 2 升級 Directory Server](#)」中提供了將 Directory Server 升級到發行版本 5 的指示。
3. **Access Manager (Access Manager SDK)**。 在第 241 頁的第 14 章「[Access Manager](#)」中提供了將 Access Manager 升級到發行版本 5 的指示。
4. **Portal Server**。 在第 285 頁的第 15 章「[Portal Server](#)」中提供了升級 Portal Server 的指示。

升級發行版本 3 Portal Server Secure Remote Access

若要將發行版本 3 Portal Server Secure Remote Access 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 3 以外，其他請遵循第 358 頁的「[從 Java ES 發行版本 4 升級 Portal Server Secure Remote Access](#)」中的指示。

多重實例升級

在某些部署架構中，會將諸如 Gateway 等 Portal Server Secure Remote Access 元件部署在多個電腦系統上，以提供安全性與延展性，並提高可用性。例如，可以讓 Gateway 元件在多台電腦上執行，並透過負載平衡器來分散負載。

當從發行版本 3 Portal Server Secure Remote Access 執行多重實例升級時，除了將參照的「發行版本 4」取代為「發行版本 3」之外，其他請遵循第 370 頁的「[多重實例升級](#)」中所述的程序。您還必須升級 Access Manager，如第 371 頁的步驟 1 中所述。

從 Java ES 發行版本 2 升級 Portal Server Secure Remote Access

本節包含將 Java ES 2004Q2 (發行版本 2) Portal Server Secure Remote Access 升級到發行版本 5 的相關資訊。除了如以下幾節所述的一些變更以外，升級程序與將發行版本 4 Portal Server Secure Remote Access 升級到發行版本 5 類似：

- [發行版本 2 升級前作業](#)
- [升級發行版本 2 Portal Server Secure Remote Access](#)
- [發行版本 2 升級後作業](#)
- [多重實例升級](#)

備註 如果您要在 Linux 平台上從發行版本 2 Portal Server Secure Remote Access 進行升級，則必須執行雙重升級，也就是 Portal Server Secure Remote Access 和作業系統都要進行升級 (在 RHEL 2.1 上不支援發行版本 5 Portal Server Secure Remote Access)。如需更多資訊，請參閱第 357 頁的「雙重升級」。

發行版本 2 升級前作業

除了以下作業以外，從發行版本 2 升級 Portal Server Secure Remote Access 的升級前作業與第 360 頁的「[發行版本 4 升級前作業](#)」中所述的內容相同：

- [升級 Portal Server Secure Remote Access 相依元件](#)
- [刪除 Gateway 服務項目](#)

升級 Portal Server Secure Remote Access 相依元件

當從發行版本 2 升級 Portal Server Secure Remote Access 時，您必須在升級 Portal Server Secure Remote Access 之前將 Access Manager 升級到發行版本 4 或發行版本 5，而且您不能在發行版本 2 上保留任何其他相依元件，也不能將某些相依元件升級到發行版本 4，並將其他相依元件升級到發行版本 5。如需更多資訊，請參閱第 356 頁的「[選擇性的升級問題](#)」。

下列相依元件必須按以下顯示的順序升級。

1. **共用元件。** 在第 59 頁的第 2 章「升級 Java ES 共用元件」中提供了將 Java ES 共用元件升級到發行版本 5 的指示。但是，如果尚未升級共用元件，psupgrade 程序檔會自動升級它們。
2. **Directory Server。** 在第 108 頁的「從 Java ES 發行版本 2 升級 Directory Server」中提供了將 Directory Server 升級到發行版本 5 的指示。
3. **Access Manager (Access Manager SDK)。** 在第 241 頁的第 14 章「Access Manager」中提供了將 Access Manager 升級到發行版本 5 的指示。
4. **Portal Server。** 在第 285 頁的第 15 章「Portal Server」中提供了升級 Portal Server 的指示。

若要將發行版本 2 Portal Server Secure Remote Access 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他請遵循第 358 頁的「從 Java ES 發行版本 4 升級 Portal Server Secure Remote Access」中的指示。

刪除 Gateway 服務項目

當從發行版本 2 升級 Portal Server 時，必須手動刪除 amService-srapGateway 使用者項目，否則 Gateway 將無法在升級之後啟動。執行下列步驟：

1. 登入 Access Manager 主控台。
2. 列出機構 DN 中的所有使用者。
3. 刪除 amService-srapGateway 使用者。

升級發行版本 2 Portal Server Secure Remote Access

將 Java ES 2004Q2 (發行版本 2) Portal Server Secure Remote Access 升級到發行版本 5 的程序與將發行版本 4 Portal Server Secure Remote Access 升級到發行版本 5 的程序相同。

若要將發行版本 2 Portal Server Secure Remote Access 升級到發行版本 5，除了將參照的所有發行版本 4 取代為發行版本 2 以外，其他請遵循第 358 頁的「從 Java ES 發行版本 4 升級 Portal Server Secure Remote Access」中的指示。

發行版本 2 升級後作業

除了以下作業以外，從發行版本 2 進行升級的升級後作業與第 367 頁的「發行版本 4 升級後作業」中所述的內容相同：

- 為 [Proxylet 服務設定 Portal Server 網域](#)

為 Proxylet 服務設定 Portal Server 網域

在將發行版本 2 Portal Server Secure Remote Access 升級到發行版本 5 後，您必須設定正確的 Portal Server 網域值。

1. 登入 Portal Server 主控台，然後瀏覽至 Secure Remote Access 下的 Proxylet 標籤。
2. 選取可在其中找到 Proxylet 服務的機構的辨別名稱 (DN)。
3. 在 Proxylet 規則的 [網域] 欄位下，以安裝了 Portal Server 的網域名稱取代 SERVER_DOMAIN。
4. 針對可在其中找到 Proxylet 服務的所有機構重複以上步驟。

多重實例升級

將發行版本 2 Portal Server Secure Remote Access 元件 (或 Portal Server) 升級到發行版本 5 不支援多重實例漸進式升級 (請參閱第 370 頁的「多重實例升級」)。

從臨時功能發行版本 7.0 升級 Portal Server Secure Remote Access

本節包含將 Portal Server Secure Remote Access 從臨時功能發行版本 (IFR) 7.0 2005Q4 升級到 Java ES 5 (發行版本 5) 的相關資訊。

本節涵蓋以下主題：

- [簡介](#)
- [Portal Server Secure Remote Access IFR 7.0 升級](#)
- [多重實例升級](#)

簡介

將 Portal Server Secure Remote Access IFR 7.0 升級到發行版本 5 時，請考慮升級程序的下列層面：

在從發行版本 4 升級時，用於將 Portal Server Secure Remote Access IFR 升級到發行版本 5 的 `psupgrade` 程序檔不會安裝新的套裝軟體。升級程序會要求您套用以下修補程式：

表 16-5 將 Portal Server Secure Remote Access IFR 升級到發行版本 5 的修補程式¹

說明	修補程式 ID：Solaris 9 和 10	修補程式 ID：Linux
Portal Server 7.1	121465-10 (SPARC) 121466-10 (x86)	121467-10
Portal Server 7.1 本土化版本	123254-02 (SPARC) 124590-02 (x86)	123255-02

¹ 修補程式的修訂版號碼是升級到 Java ES 發行版本 5 的最低需求。如果有更新的修訂版可用，請使用更新的修訂版，而不要使用表中的修訂版。

Portal Server Secure Remote Access IFR 7.0 升級

本節描述如何在 Solaris 及 Linux 平台上將 Portal Server Secure Remote Access 從 IFR 升級到 Java ES 發行版本 5。當某個主題依賴平台特定的程序時，該主題將指出其適用的作業系統。本節涵蓋以下主題：

- [升級前作業](#)
- [升級 Portal Server Secure Remote Access IFR 7.0 \(Solaris\)](#)
- [升級 Portal Server Secure Remote Access IFR 7.0 \(Linux\)](#)
- [驗證升級](#)
- [升級後作業](#)
- [回復升級 \(Solaris\)](#)
- [回復升級 \(Linux\)](#)

升級前作業

IFR 升級與發行版本 4 升級的升級前作業相同 (請參閱第 360 頁的「[發行版本 4 升級前作業](#)」)。

升級 Portal Server Secure Remote Access IFR 7.0 (Solaris)

本節討論影響 Portal Server Secure Remote Access 升級程序的注意事項，接著提供對程序本身的說明。

IFR 7 升級注意事項 (Solaris)

將 Portal Server Secure Remote Access IFR 升級到發行版本 5 與發行版本 4 升級要考慮的注意事項相同 (請參閱第 363 頁的「[升級注意事項 \(Solaris\)](#)」)。

IFR 7 升級程序 (Solaris)

下述程序適用於位於執行升級的電腦上的 Portal Server Secure Remote Access。

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 停止正在本地執行的 Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t gateway -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t rwproxy -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser  
-f passwordFile -t nlproxy -N gatewayProfileName
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443  
Rewriter Proxy : netstat -an | grep 10443  
Netlet Proxy : netstat -an | grep 10555
```

3. 確定 Access Manager 正在執行。
4. 根據第 380 頁的表 16-5 取得所需的修補程式。

除非指示您使用特定修訂版，請一律使用最新的修補程式修訂版。

可從下列位置將修補程式下載到 /tmp：

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

5. 請套用適當的 Portal Server 修補程式，如有必要，也請套用表 16-5 中的本土化修補程式。

```
patchadd patch_ID
```

6. 確認修補程式升級成功：

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

輸出應傳回步驟 5 中套用的修補程式 ID 版本。

7. 若本土化套裝軟體已經在步驟 5 中升級，請將 Portal Server Console JVM 的語言環境設定為 UTF-8。

```
export LC_ALL=ja_JP.UTF-8  
export LANG=ja_JP.UTF-8
```

8. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 與 JAVA_HOME)：

```
export ANT_HOME=/usr/sfw  
export JAVA_HOME=/usr/jdk/entsys-j2se
```

9. 請確定您電腦上有足夠的交換空間。

做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。

- 執行 `psupgrade` 程序檔。

```
cd PortalServer7-base/bin
./psupgrade
```

`psupgrade` 程序檔不會從 Java ES 發行版本 5 發行物中執行，而且不會呼叫 Java ES 安裝程式 (套裝軟體已經過修補)。

升級 Portal Server Secure Remote Access IFR 7.0 (Linux)

本節討論影響 Portal Server Secure Remote Access 升級程序的注意事項，接著提供對程序本身的說明。

IFR 7 升級注意事項 (Linux)

在 Linux OS 上安裝發行版本 5 修補程式會移除先前的 RPM，除了這一點外，在 Linux 平台上將 Portal Server Secure Remote Access 軟體升級到發行版本 5 與在 Solaris 平台上升級時所要考慮的注意事項相同 (請參閱第 363 頁的「升級注意事項 (Solaris)」)。

IFR 7 升級程序 (Linux)

下述程序適用於位於執行升級的電腦上的 Portal Server Secure Remote Access。

注意 在 Linux 上執行的從 Portal Server Secure Remote Access IFR 到發行版本 5 的升級無法回復。在執行下列程序之前，請務必先備份您的系統。

- 請以超級使用者身份登入或成為超級使用者。

```
su -
```

- 停止正在本地執行的 Gateway、Rewriter Proxy 或 Netlet Proxy 的任何實例。

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser
-f passwordFile -t gateway -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser
-f passwordFile -t rwproxy -N gatewayProfileName
```

```
PortalServer7-base/bin/psadmin stop-sra-instance -u amadminUser
-f passwordFile -t nlproxy -N gatewayProfileName
```

檢查程序是否已經停止：

```
Gateway : netstat -an | grep 443
```

```
Rewriter Proxy : netstat -an | grep 10443
```

```
Netlet Proxy : netstat -an | grep 10555
```

3. 確定 Access Manager 正在執行。
4. 使用第 380 頁的表 16-5 中的修補程式號碼和 RPM 名稱來取得所需的修補程式。
除非指示您使用特定修訂版，請一律使用最新的修補程式修訂版。

可從下列位置將修補程式下載到 /tmp：

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

5. 套用 Portal Server 修補程式，如有必要，也以該順序套用表 16-5 中的 Portal Server 本土化 RPM。

請參閱 Portal Server 修補程式的讀我檔案，其中說明了如何使用程序檔來套用修補程式的 RPM：

```
cd /tmp
```

其中 /tmp 是將修補程式下載到的目錄。

```
./update
```

更新程序檔會安裝 RPM 的部份。

針對本土化修補程式，請使用以下指令安裝每一個 RPM：

```
rpm -Fvh patchName-version.rpm
```

6. 確認修補程式升級成功：

```
rpm -qa | grep sun-portal-core
```

應傳回 RPM 的升級修訂版號碼。

7. 若本土化套裝軟體已經在步驟 5 中升級，請將 Portal Server Console JVM 的語言環境設定為 UTF-8。

```
export LC_ALL=ja_JP.UTF-8
```

```
export LANG=ja_JP.UTF-8
```

8. 設定 psupgrade 程序檔所需的兩個環境變數 (ANT_HOME 與 JAVA_HOME)：

```
export ANT_HOME=/opt/sun
```

```
export JAVA_HOME=/usr/jdk/entsys-j2se
```

9. 請確定您電腦上有足夠的交換空間。

做為指導原則，應將交換空間設定為實體 RAM 容量的兩倍。

10. 執行 `psupgrade` 程序檔。

```
cd PortalServer7-base/bin
./psupgrade
```

`psupgrade` 程序檔不會從 Java ES 發行版本 5 發行物中執行，而且不會呼叫 Java ES 安裝程式 (套裝軟體已經過修補)。

驗證升級

您可使用下列指令驗證是否將 Portal Server Secure Remote Access 套裝軟體修補為發行版本 5：

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID
-f adminpasswordfile.
```

請參閱第 360 頁的表 16-4 中的輸出值。

您也可以檢查位於以下位置的升級記錄檔：

```
/var/sadm/install/logs/Sun_Java_System_Portal_Server_upagrede.log
```

升級後作業

將 Portal Server Secure Remote Access 升級到發行版本 5 時不需要升級後作業。

回復升級 (Solaris)

本節描述影響 Portal Server Secure Remote Access 升級回復程序的注意事項，接著提供程序本身。

回復注意事項 (Solaris)

回復到發行版本 5 的升級程序包括復原到 `PortalServer7-base` 的 IFR 安裝。

回復程序 (Solaris)

1. 請以超級使用者身份登入或成為超級使用者。

```
su -
```

2. 將 Directory Server 回復為它在升級前的狀態。

使用 `Directory Server backup/restore` 指令行及 GUI 公用程式。請參閱「Sun Java System Directory Server Enterprise Edition 6.0 管理指南」(<http://docs.sun.com/doc/820-0294>) 的「Directory Server 備份與復原」一章。

3. 取消第 380 頁的表 16-5 中的 Portal Server 7.1 修補程式。

```
patchrm patch_ID
```

回復升級 (Linux)

沒有在 Linux 平台上回復升級的程序。

多重實例升級

在某些部署架構中，會將 Portal Server Secure Remote Access 部署在多個電腦系統上，以提供延展性並提高可用性。例如，可以讓 Portal Server Secure Remote Access 元件在多台電腦上執行，並透過負載平衡器來分散負載。

若 Portal Server Secure Remote Access 實例已達負載平衡，則可以執行串聯式升級，即依序升級 Portal Server Secure Remote Access 實例而不中斷服務。升級 Portal Server Secure Remote Access 的每個實例時其他實例仍可繼續執行。需要按照第 381 頁的「Portal Server Secure Remote Access IFR 7.0 升級」中所述的方式升級每個實例。

當從 IFR Portal Server Secure Remote Access 執行多重實例升級時，除了將參照的發行版本 4 取代為「IFR」之外，其他請遵循第 370 頁的「多重實例升級」中所述的程序。您還必須升級 Access Manager，如第 371 頁的步驟 1 中所述。

Java Enterprise System 發行版本 內容

本附錄列出各種 Java Enterprise System 發行版本內容。包含下列各節：

- 第 388 頁的「Java ES 2003Q4 (發行版本 1)」
- 第 390 頁的「Java ES 2004Q2 (發行版本 2)」
- 第 393 頁的「Java ES 2005Q1 (發行版本 3)」
- 第 398 頁的「Java ES 2005Q4 (發行版本 4)」
- 第 403 頁的「Java ES 5 (發行版本 5)」

Java ES 2003Q4 (發行版本 1)

本節列出 Java Enterprise System 2003Q4 的內容。

發行版本 1 安裝程式 - 可選取的元件

Sun Open Network Environment (Sun ONE) 和 Sun Cluster 元件產品提供所需的基礎架構服務，以支援已發行的企業應用程式。這些元件產品包括：

- Sun Cluster 3.1 和 Sun Cluster Agents for Sun ONE
- Sun ONE Administration Server 5.2
- Sun ONE Application Server 7, Update 1
- Sun ONE Calendar Server 6.0
- Sun ONE Directory Server 5.2
- Sun ONE Directory Proxy Server 5.2
- Sun ONE Identity Server 6.1
- Sun ONE Instant Messaging 6.1
- Sun ONE Message Queue 3.0.1 Service Pack 2
- Sun ONE Messaging Server 6.0
- Sun ONE Portal Server 6.2
- Sun ONE Portal Server, Secure Remote Access 6.2
- Sun ONE Web Server 6.1

發行版本 1 共用元件

共用元件提供元件產品所依賴的本機服務和技術支援。當您安裝元件產品時，如果尚未安裝所需的共用元件，Java ES 安裝程式會自動安裝這些共用元件。

Java Enterprise System 包括以下共用元件：

- ANT (Jakarta ANT Java/XML-based build tool)
- Apache Commons Logging
- ICU (International Components for Unicode)
- J2SE™ platform 1.4.1_06 (Java 2 Platform, Standard Edition)
- JAF (JavaBeans™ Activation Framework)
- JATO (Sun ONE Application Framework)
- JavaHelp™ Runtime
- JAXM (Java API for XML Messaging) Client Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries)
- JAX-RPC (Java APIs for XML-based Remote Procedure Call)
- JSS (Java Security Services)
- KT search engine
- LDAP C Language SDK
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- SAAJ (SOAP with Attachments API for Java)
- SASL (Simple Authentication and Security Layer)
- XML C Library (libxml)

備註 對於 Application Server 和 Directory Server，系統上還需要 Perl，但是它不像 Java ES 共用元件那樣會自動安裝。

Java ES 2004Q2 (發行版本 2)

本節列出 Java Enterprise System 2004Q2 的內容。

發行版本 2 安裝程式 - 可選取的元件

元件產品提供所需的基礎架構服務，以支援發行的企業應用程式。當您在特定主機上安裝 Java Enterprise System 時，必須根據整體的配置架構來選擇安裝在主機上的元件產品。

Java Enterprise System 2004Q2 包含下列元件產品：

通訊服務和協作服務

- 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 與應用程式服務

- Sun Java System Application Server 7.0 Update 3 (Standard 及 Platform Edition)
- Sun Java System Web Server 6 2004Q2 Update 1 Service Pack 2
- Sun Java System Message Queue 3.5 SP1 (Platform 及 Enterprise Edition)

目錄服務和身份識別服務

- Sun Java System Identity Server 6.2 2004Q2，包括
Sun Java System Communications Services 6 2004Q2 User Management Utility
- Sun Java System Directory Server 5 2004Q2
- Sun Java System Directory Proxy Server 5 2004Q2

可用性服務

- Sun Cluster 3.1 4/04 和 Sun Cluster Agents for Sun Java System

管理服務

- Sun Java System Administration Server 5 2004Q2
- Sun Remote Services Net Connect 3.5

請注意，Sun Cluster、Sun Cluster Agents 和 Sun Remote Services Net Connect 在 Linux OS 上不可用。

發行版本 2 共用元件

共用元件提供元件產品所依賴的本機服務和技術支援。當您安裝元件產品時，如果尚未安裝所需的共用元件，Java ES 安裝程式會自動安裝這些共用元件。

Java Enterprise System 2004Q2 包括以下共用元件：

- Ant (Jakarta ANT Java/XML-based build tool)
- Apache Commons Logging
- Apache SOAP (Simple Object Access Protocol)
- ICU (International Components for Unicode)
- J2SE™ platform 1.4.2_04 (Java 2 Platform, Standard Edition)
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Application Framework)
- JavaHelp™ Runtime
- JAXB (Java Architecture for XML Binding)
- JAXM (Java API for XML Messaging) Client Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries)
- JAX-RPC (Java APIs for XML-based Remote Procedure Call)
- JCAPI (Java Calendar API)
- JSS (Java Security Services)
- KT search engine

- LDAP C Language SDK
- LDAP Java SDK
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- Perl LDAP , 包括 NSPERL
- SAAJ (SOAP with Attachments API for Java)
- SAML (Security Assertions Markup Language)
- SASL (Simple Authentication and Security Layer)
- SNMP (Simple Network Management Protocol) Peer
- Sun Explorer Data Collector
- XML C Library (libxml)

Java ES 2005Q1 (發行版本 3)

本節會列出 Java Enterprise System 2005Q1 的內容。

發行版本 3 安裝程式 - 可選取的元件

在 Java ES 安裝程式的元件選取頁面上，可選取的元件根據其提供的服務進行分組。下列清單也會顯示與每個元件一起安裝的子元件。

通訊服務和協作服務

- Sun Java System Messaging Server 6 2005Q1
- Sun Java System Calendar Server 6 2005Q1
- Sun Java System Instant Messaging 7 2005Q1
 - Instant Messaging Server Core，包含伺服器及多重訊號組合器軟體
 - Instant Messaging Resources
 - Access Manager Instant Messaging Service
- Sun Java System Portal Server 6 2005Q1
- Sun Java System Portal Server Secure Remote Access 6 2005Q1
 - Secure Remote Access Core
 - Gateway
 - Netlet Proxy
 - Rewriter Proxy
- Sun Java System Communications Express 2005Q1
- Sun Java System Directory Preparation Tool

Web 與應用程式服務

- Sun Java System Application Server Enterprise Edition 8.1 2005Q1
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin

可以與 Web Server 或 Apache Web Server 配合使用，可在配置時選取。預設為 Web Server。

 - PointBase
 - Sample Applications
- Sun Java System Web Server 6 2005Q1 Update 1 Service Pack 4
- Sun Java System Message Queue 3 2005Q1

目錄服務和身份識別服務

- Sun Java System Access Manager 6.3 2005Q1

Portal Server 和 Messaging Server 的 Delegated Administrator 佈建工具會和 Access Manager 一起自動安裝。

 - Identity Management and Policy Services Core (包括 Delegated Administrator Utility)
 - Access Manager Administration Console
 - Common Domain Services for Federation Management
 - Access Manager SDK
- Sun Java System Directory Server 5 2005Q1
- Sun Java System Directory Proxy Server 5 2005Q1

可用性服務

- Sun Cluster 3.1 9/04
 - Sun Cluster Core
- Sun Cluster Agents for Sun Java System
 - HA/Scalable Sun Java System Web Server
 - HA Sun Java System Message Queue
 - HA Sun Java System Portal Server
 - HA Sun Java System Administration Server
 - HA Sun Java System Directory Server
 - HA Sun Java System Messaging Server
- HADB (用於高可用性階段作業儲存)

管理服務

- Sun Java System Administration Server 5 2005Q1
- SunSM Remote Services Net Connect 3.1.1

備註

Solaris 10 或 Linux 作業系統上不提供 Sun Cluster、Sun Cluster Agents 和 Sun Remote Services Net Connect。

Sun Remote Services Net Connect 在 Solaris x86 平台上不可用。

發行版本 3 共用元件

共用元件為可選取的元件提供本機服務和技術支援。當您安裝 Java ES 元件時，如果尚未安裝所需的共用元件，安裝程式會自動安裝這些共用元件。

此 Java ES 發行版本包含下列共用元件：

- Ant (Jakarta ANT Java/XML-based build tool)
- Apache SOAP (Simple Object Access Protocol) Runtime
- Berkeley Database
- Common agent container
- ICU (International Components for Unicode)
- J2SE™ (Java 2 Platform, Standard Edition) platform 5.0
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio Web Application Framework)
- JavaHelp™ Runtime
- JavaMail™ Runtime
- JAXB (Java Architecture for XML Binding) Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries) Runtime
- JAX-RPC (Java API for XML-based Remote Procedure Call) Runtime
- JCAPI (Java Calendar API)
- JDMK (Java Dynamic Management™ Kit) Runtime
- JSS (Java Security Services)
- KTSE (KT Search Engine)
- LDAP C SDK
- LDAP Java SDK
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- Perl LDAP，包括 NSPERL

- SAAJ (SOAP with Attachments API for Java)
- SAML (Security Assertions Markup Language)
- SASL (Simple Authentication and Security Layer)
- SNMP (Simple Network Management Protocol) Peer
- Sun Explorer Data Collector (僅限於 Solaris)
- Sun Java Monitoring Framework
- Sun Java Web Console
- Tomcat Servlet JSP Container
- XML C Library (libxml)
- WSCL (Web services Common Library)

Java ES 2005Q4 (發行版本 4)

本節會列出 Java Enterprise System 2005Q4 的內容。

發行版本 4 安裝程式 - 可選取的元件

在 Java ES 安裝程式的元件選取頁面上，可選取的元件根據其提供的服務進行分組。下列清單也會顯示與每個元件一起安裝的子元件。

通訊服務和協作服務

- Sun Java System Messaging Server 6.2 2005Q4
- Sun Java System Calendar Server 6.2 2005Q4
- Sun Java System Instant Messaging 7.0.1 2005Q4
 - Instant Messaging Server Core，包含伺服器及多重訊號組合器軟體
 - Instant Messaging Resources
 - Access Manager Instant Messaging Service
- Sun Java System Portal Server 6.3.1 2005Q4
- Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4
 - Secure Remote Access Core
 - Gateway
 - Netlet Proxy
 - Rewriter Proxy
- Sun Java System Communications Express 6.2 2005Q4
- Sun Java System Directory Preparation Tool 6.3 2005Q4
- Sun Java System Communications Services Delegated Administrator 6.3 2005Q4
 - Delegated Administrator Console and Utility
 - Delegated Administrator Server

Web 與應用程式服務

- Sun Java System Application Server Enterprise Edition 8.1 2005Q4
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin

可以與 Web Server 或 Apache Web Server 配合使用，可在配置時選取。預設為 Web Server。

- PointBase Database
- Sample Applications
- Sun Java System Web Server 6.1 Service Pack 5 2005Q4
- Sun Java Web Proxy Server 4.0.1 2005Q4
- Sun Java System Message Queue Enterprise Edition 3.6 SP3 2005Q4
- Sun Java Service Registry 3.0

目錄服務和身份識別服務

- Sun Java System Access Manager 7.0 2005Q4
 - Identity Management and Policy Services Core
 - Access Manager Administration Console
 - Common Domain Services for Federation Management
 - Access Manager SDK
- Sun Java System Directory Server 5.2 2005Q4
- Sun Java System Directory Proxy Server 5.2 2005Q4

可用性服務

- Sun Cluster 3.1 8/05
 - Sun Cluster Core
 - Sun Cluster Agents for Sun Java System
 - HA Sun Java System Directory Server
 - HA Sun Java System Administration Server
 - HA/Scalable Sun Java System Web Server
 - HA Sun Java System Message Queue
 - HA Sun Java System Application Server
 - HA Sun Java System Messaging Server
 - HA Sun Java System Calendar Server
 - HA Sun Java System Instant Messaging
- High Availability Session Store (HADB) 4.4.2

管理服務

- Sun Java System Administration Server 5.2 2005Q4

備註 Sun Cluster 和 Sun Cluster Agents 在 Solaris OS 上受支援，但在 Linux OS 上則不受支援。

發行版本 4 共用元件

共用元件為可選取的元件提供本機服務和技術支援。當您安裝 Java ES 元件時，如果尚未安裝所需的共用元件，安裝程式會自動安裝這些共用元件。

此 Java ES 發行版本包含下列共用元件：

- ANT (Jakarta ANT Java/XML-based build tool)
- ACL (Apache Commons Logging)
- BDB (Berkeley Database)
- CAC (Common agent container)
- Derby Database
- ICU (International Components for Unicode)
- IM-SDK (Instant Messenger SDK)
- J2SE™ (Java 2 Platform, Standard Edition) platform 5.0
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio Web Application Framework)
- JavaHelp™ Runtime
- JavaMail™ Runtime
- JAXB (Java Architecture for XML Binding) Runtime
- JAXP (Java API for XML Processing)
- JAXR (Java API for XML Registries) Runtime
- JAX-RPC (Java API for XML-based Remote Procedure Call) Runtime
- JCAPI (Java Calendar API)
- JDMK (Java Dynamic Management™ Kit) Runtime
- JSS (Java Security Services)
- KTSE (KT Search Engine)
- LDAP C SDK
- LDAP Java SDK
- MA (Mobile Access) Core

- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- SAAJ (SOAP runtime with Attachments API for Java)
- SASL (Simple Authentication and Security Layer)
- SEDC (Sun Explorer Data Collector , 僅限於 Solaris)
- MFWK (Java ES Monitoring Framework)
- SJWC (Sun Java Web Console)
- WSCL (Web services Common Library)

Java ES 5 (發行版本 5)

本節列出 Java Enterprise System 5 的內容。

發行版本 5 安裝程式 - 可選取的元件

在 Java ES 安裝程式的元件選取頁面上，可選取的元件根據其提供的服務進行分組。下列清單也會顯示與每個元件一起安裝的子元件。請注意，Java ES 安裝程式不再支援 Communication Services 元件。

協同作業服務

- Sun Java System Portal Server 7.1
- Sun Java System Portal Server Secure Remote Access 7.1
 - Gateway
 - Netlet Proxy
 - Rewriter Proxy

Web 與應用程式服務

- Sun Java System Application Server Enterprise Edition 8.2
 - Domain Administration Server
 - Application Server Node Agent
 - Command Line Administration Tool
 - Load Balancing Plugin
 - 可以與 Web Server 或 Apache Web Server 配合使用，可在配置時選取。預設為 Web Server。
 - Sample Applications
- Sun Java System Web Server 7.0
 - Web Server 7.0 CLI
 - Web Server 7.0 Core
 - Web Server 7.0 Samples
- Sun Java Web Proxy Server 4.0.4
- Sun Java System Message Queue 3.7UR1

- Service Registry 3.1
 - Service Registry Client Support
 - Service Registry Deployment Support

目錄服務和身份識別服務

- Sun Java System Access Manager 7.1
 - Identity Management and Policy Services Core
 - Access Manager Administration Console
 - Common Domain Services for Federation Management
 - Access Manager SDK
 - Distributed Authentication
 - Client SDK
 - Session Failover Client
- Sun Java System Directory Server Enterprise Edition 6
 - Java Enterprise System Directory Server 6 Core Server
 - Java Enterprise System Directory Service Control Center
 - Sun Java System Directory Server Enterprise Edition 6 Command-Line Utilities
 - Sun Java System Directory Proxy Server 6 Core Server

可用性服務

- Sun Cluster 3.1
 - Sun Cluster Core
- Sun Cluster Geographic Edition 3.1 2006Q4
- Sun Cluster Agents 3.1
 - HA Sun Java System Application Server
 - HA Sun Java System Message Queue
 - HA Sun Java System Directory Server
 - HA Sun Java System Messaging Server
 - HA Sun Java System Application Server EE (HADB)

- HA/Scalable Sun Java System Web Server
- HA Instant Messaging
- HA Sun Java System Calendar Server
- ...
- Sun Java System High Availability Session Store 4.4.3

共用服務

- 所有共用元件
請參閱第 406 頁的「發行版本 5 共用元件」
- Sun Java System Monitoring Console 1.0
- Java DB
 - Java DB Client
 - Java DB Server

備註 Sun Cluster 和 Sun Cluster Agents 在 Solaris OS 上受支援，但在 Linux OS 上則不受支援。

發行版本 5 共用元件

共用元件為可選取的元件提供本機服務和技術支援。當您安裝 Java ES 元件時，如果尚未安裝所需的共用元件，安裝程式會自動安裝這些共用元件。

此 Java ES 發行版本包含下列共用元件：

- ANT (Jakarta ANT Java/XML-based build tool)
- ACL (Apache Commons Logging)
- BDB (Berkeley Database)
- CAC (Common agent container) (僅適用於 Sun Cluster)
- CAC (Common agent container)
- FIS (FastInfoSet)
- ICU (International Components for Unicode)
- IM-SDK (Instant Messenger SDK)
- J2SE™ (Java 2 Platform, Standard Edition) platform 5.0
- JAF (JavaBeans™ Activation Framework)
- JATO (Java Studio 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
- JAXWS (Java API for Web Services) Runtime
- JDMK (Java Dynamic Management™ Kit) Runtime
- JSS (Java Security Services)
- JSTL (JSP Standard Library Template)
- KTSE (KT Search Engine)
- LDAP C SDK

- LDAP Java SDK
- MA (Mobile Access) Core
- NSPR (Netscape Portable Runtime)
- NSS (Network Security Services)
- SAAJ (SOAP runtime with Attachments API for Java)
- SASL (Simple Authentication and Security Layer)
- SEDC (Sun Explorer Data Collector · 僅限於 Solaris)
- MFWK (Java ES Monitoring Framework)
- SJWC (Sun Java Web Console)
- WSCL (Web services Common Library)
- XWSS (XML Web Services Security)

索引

A

Access Manager

子元件 399, 404

升級方法 33

共用元件相依性 43

產品元件相依性 47

縮寫 28, 41

AccessManager-base 路徑 241

AccessManagerConfig-base 路徑 241

ACL 共用元件

在相依性表格中 43

完整名稱 29

amconfig 程序檔 (Access Manager) 256

ampre71upgrade 程序檔 (Access Manager) 254, 276

amupgrade 程序檔 (Access Manager) 260, 281

ANT 共用元件

在相依性表格中 43

完整名稱 29

ANT_HOME 環境變數 301, 304, 326, 337, 339, 364, 366, 382, 384

Apache Commons Logging，請參閱 ACL

Application Server

子元件 399, 403

升級方法 33

共用元件相依性 43

安裝後配置 208

產品元件相依性 48

縮寫 28, 41

AppServer7-base 路徑 190

AppServer7Config-base 路徑 190

AppServer8-base 路徑 190

AppServer8Config-base 路徑 190

AppServer8Install-base 路徑 190

B

BDB 共用元件

在相依性表格中 43

完整名稱 29

Berkeley Database，請參閱 BDB

C

CAC 共用元件

在相依性表格中 43

完整名稱 29

Common agent container，請參閱 CAC

D

db2bak 公用程式 250

Derby Database 共用元件 401

E

Directory Proxy Server

- 升級方法 33
- 共用元件相依性 43
- 產品元件相依性 48
- 縮寫 28, 41

Directory Server

- 升級方法 33
- 共用元件相依性 43
- 產品元件相依性 48
- 縮寫 28, 41

DirServer-base 路徑 (Directory Server 6.0) 93, 109

E

Emphasis>PortalServer6-base 路徑 286

F

FastInfoSet，請參閱 FIS

FIS 共用元件

- 在相依性表格中 43
- 完整名稱 29

H

HADB-base 路徑 155

High Availability Session Store (HADB)

- 升級方法 33
- 共用元件相依性 43
- 產品元件相依性 48
- 縮寫 28, 41

I

ICU 共用元件

- 在相依性表格中 43
- 完整名稱 29

IM-SDK 共用元件

- 在相依性表格中 43
- 完整名稱 29

Instant Messenger SDK，請參閱 IM-SDK

International Components for Unicode，請參閱 ICU

J

JAF 共用元件

- 在相依性表格中 43
- 完整名稱 29

Jakarta ANT Java/XML-based build tool，請參閱 ANT

JATO 共用元件

- 在相依性表格中 43
- 完整名稱 29

Java 2 Platform, Standard Edition，請參閱 J2SE

Java 2 Platform, Standard Edition，請參閱 Java SE

Java API for XML Processing，請參閱 JAXP

Java API for XML Registries，請參閱 JAXR

Java API for XML-based Remote Procedure Call，請參閱 JAX-RPC

Java Architecture for XML Binding，請參閱 JAXB

Java Calendar API，請參閱 JCAPI

Java DB

- 升級方法 33
- 共用元件相依性 43
- 產品元件相依性 48
- 縮寫 28, 41

Java Dynamic Management Kit，請參閱 JDMK

Java ES 2003Q4 (發行版本 1)

- 共用元件 389
- 產品元件 388

Java ES 2004Q2 (發行版本 2)
 共用元件 391
 產品元件 390

Java ES 2005Q1 (發行版本 3)
 共用元件 396
 產品元件 393

Java ES 2005Q4 (發行版本 4)
 共用元件 401
 產品元件 398

Java ES 5 (發行版本 5)
 共用元件 406
 產品元件 403

Java ES Monitoring Framework，請參閱 MFWK

Java SE 共用元件
 在相依性表格中 43
 完整名稱 29

Java Security Services，請參閱 JSS

Java Studio Web Application Framework，請參閱 JATO

JAVA_HOME 326, 337, 339, 364, 366, 382, 384

JAVA_HOME 環境變數 258, 279, 301, 304

JavaBeans Activation Framework，請參閱 JAF

JavaHelp 共用元件
 在相依性表格中 44
 完整名稱 29

JavaMail 共用元件
 在相依性表格中 44
 完整名稱 29

JAXB 共用元件
 在相依性表格中 44
 完整名稱 29

JAXP 共用元件
 在相依性表格中 44
 完整名稱 29

JAXR 共用元件
 在相依性表格中 44
 完整名稱 29

JAX-RPC 共用元件
 在相依性表格中 44
 完整名稱 30

JAXWS 共用元件
 在相依性表格中 44
 完整名稱 30

JCAPI 共用元件
 在相依性表格中 44
 完整名稱 30

JDK 共用元件
 在相依性表格中 44
 完整名稱 30

JSP Standard Library Template，請參閱 JSTL

JSP 檔案，自訂 244, 251, 256, 278

JSS 共用元件
 在相依性表格中 44
 完整名稱 30

JSTL 共用元件
 在相依性表格中 44
 完整名稱 30

K

KT Search Engine，請參閱 KTSE

KTSE 共用元件
 在相依性表格中 44
 完整名稱 30

L

LDAP C Language SDK 共用元件 406

LDAP C SDK 共用元件
 在相依性表格中 44
 完整名稱 30

LDAP J SDK 共用元件 407
 在相依性表格中 44
 完整名稱 30

M

M

MA Core 共用元件

在相依性表格中 44

完整名稱 30

Message Queue

升級方法 33

共用元件相依性 43

產品元件相依性 49

縮寫 28, 42

MessageQueue-base 路徑 167

MFWK 共用元件

在相依性表格中 44

完整名稱 30

Mobile Access Core，請參閱 MA core

Monitoring Console

共用元件相依性 43

產品元件相依性 49

縮寫 28

mqugrade 程序檔 (Message Queue) 185, 186

N

Netscape Portable Runtime，請參閱 NSPR

Network Security Services，請參閱 NSS

NSPR 共用元件

在相依性表格中 44

完整名稱 30

NSS 共用元件

在相依性表格中 44

完整名稱 30

P

Portal Server

升級方法 34

共用元件相依性 43

產品元件相依性 49

縮寫 28, 42

Portal Server Secure Remote Access

升級方法 34

共用元件相依性 43

產品元件相依性 49

縮寫 29

Portal Server Secure Remote Access縮寫 42

PortalServer6-base 路徑 352

PortalServer6Config-base 路徑 286, 352

PortalServer7-base 路徑 286, 352

PortalServer7Config-base 路徑 286, 352

PortalServer7Data-base 路徑 286, 352

postInstall 程序檔 (Application Server) 208

R

RegistryDomainR4-base 路徑 215

RegistryDomainR5-base 路徑 215

rel4CAC-base-dir 路徑 66

rel5CAC-admin-dir 路徑 66

S

SAAJ 共用元件

在相依性表格中 44

完整名稱 30

SAML 262, 283

SASL 共用元件

在相依性表格中 44

完整名稱 30

SEDC 共用元件

在相依性表格中 44

完整名稱 30

serverRoot 路徑 (Directory Server 5.x) 93, 109

Service Registry

升級方法 34

共用元件相依性 43

產品元件相依性 50

縮寫 29, 42

ServiceRegistryR4-base 路徑 145, 215
ServiceRegistryR5-base 路徑 215
 Simple Authentication and Security Layer, **請參閱** SASL
 SJWC 共用元件
 在相依性表格中 44
 完整名稱 30
 SML Web Services Security, **請參閱** XWSS
 SOAP runtime with Attachments API for Java, **請參閱** SAAJ
 Sun Cluster
 升級方法 34
 共用元件相依性 43
 產品元件相依性 50
 縮寫 29, 42
 Sun Cluster Geographic Edition
 升級方法 34
 共用元件相依性 43
 產品元件相依性 50
 縮寫 29, 42
 Sun Explorer Data Collector, **請參閱** SEDC
 Sun Java Web Console, **請參閱** SJWC
SunClusterGeo-base 路徑 81

W

Web Proxy Server
 升級方法 34
 共用元件相依性 43
 產品元件相依性 50
 縮寫 29, 42
 Web Server
 升級方法 34
 共用元件相依性 43
 產品元件相依性 50
 縮寫 29, 42
 Web services Common Library, **請參閱** WSCL
WebProxyServer-base 路徑 227
WebServer6-base 路徑 123

WebServer7-base 路徑 123
WebServer7Config-base 路徑 123
 WSCL 共用元件
 在相依性表格中 44
 完整名稱 30

X

XWSS 共用元件
 在相依性表格中 44
 完整名稱 30

六畫

共用元件
 Java ES 2003Q4 (發行版本 1) 389
 Java ES 2004Q2 (發行版本 2) 391
 Java ES 2005Q1 (發行版本 3) 396
 Java ES 2005Q4 (發行版本 4) 401
 Java ES 5 (發行版本 5) 406
 附屬產品元件 43
 相依性 46
 印刷排版慣例 21
 安全宣示標記語言, **請參閱** SAML

七畫

作業系統
 支援矩陣 40
 修補程式 40
 規劃注意事項 40
 雙重升級 40

九畫

相依性

- 共用元件之間 46
- 產品元件, 對共用元件 43
- 產品元件之間 47

十一畫

產品元件

- Java ES 2003Q4 (發行版本 1) 388
- Java ES 2004Q2 (發行版本 2) 390
- Java ES 2005Q1 (發行版本 3) 393
- Java ES 2005Q4 (發行版本 4) 398
- Java ES 5 (發行版本 5) 403
- 相依性 47
- 對共用元件的相依性 43
- 符號慣例 21

十二畫

程序檔

- amconfig (Access Manager) 256
- ampre71upgrade (Access Manager) 254, 276
- amupgrade (Access Manager) 260, 281
- mqmigrate (Message Queue) 186
- mqupgrade (Message Queue) 185
- postInstall (Application Server) 208

十四畫

慣例

- 印刷排版 21
- 符號 21

十八畫

雙重升級

- 元件支援摘要 41
- 方法 41
- 定義 41