



Sun Java™ System

Sun Java Enterprise System 5

UNIX용 업그레이드 설명서

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

부품 번호: 820-0892
2007년 6월

Copyright © 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 모든 권리는 저작권자의 소유입니다.

Sun Microsystems, Inc.는 본 설명서에서 사용하는 기술과 관련한 지적 재산권을 보유합니다. 특히 이러한 지적 재산권에는 <http://www.sun.com/patents>에 나열된 하나 이상의 미국 특허 및 추가 특허 또는 미국 및 기타 국가에서 특허 출원 중인 응용 프로그램이 포함될 수 있습니다.

본 제품에는 SUN MICROSYSTEMS, INC.의 기업 기밀 정보가 포함되어 있습니다. 본 제품의 어떠한 부분도 SUN MICROSYSTEMS, INC.의 사전 서면 승인 없이는 사용, 공개 또는 복제가 불가능합니다.

U.S. 정부 권한 - 상용 소프트웨어. 정부 사용자는 Sun Microsystems, Inc. 표준 사용권 계약과 FAR의 해당 규정 및 추가 사항의 적용을 받습니다.

본 배포판에는 타사에서 개발한 자료가 포함되어 있을 수 있습니다.

본 제품의 일부는 Berkeley BSD 시스템일 수 있으며 University of California로부터 라이선스를 취득했습니다. UNIX는 X/Open Company, Ltd.를 통해 독점 라이선스를 취득한 미국 및 기타 국가의 등록 상표입니다.

Sun, Sun Microsystems, Sun 로고, Java, Solaris, JDK, Java Naming 및 Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, Duke 로고, Java Coffee Cup 로고, Solaris 로고, SunTone Certified 로고 및 Sun ONE 로고 등은 미국 및 기타 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다.

모든 SPARC 상표는 라이선스 하에 사용되며 미국 및 기타 국가에서 SPARC International, Inc.의 상표 또는 등록 상표입니다. SPARC 상표가 부착된 제품은 Sun Microsystems, Inc.가 개발한 아키텍처를 기반으로 합니다.

Legato 및 Legato 로고는 Legato Systems, Inc.의 등록 상표이며 Legato NetWorker는 Legato Systems, Inc.의 상표 또는 등록 상표입니다. Netscape Communications Corp 로고는 Netscape Communications Corporation의 상표 또는 등록 상표입니다.

OPEN LOOK 및 Sun(TM) Graphical User Interface는 Sun Microsystems, Inc.가 해당 사용자 및 라이선스 소유자를 위해 개발했습니다. Sun은 컴퓨터 업계에서 시각적 또는 그래픽 사용자 인터페이스 개념을 연구하고 개발하는 데 있어 Xerox의 선구자적 업적을 인정합니다. Sun은 Xerox Graphical User Interface에 대한 Xerox의 비독점 라이선스를 보유하고 있으며 이 라이선스는 OPEN LOOK GUI를 구현하거나 그 외의 경우 Sun의 서면 라이선스 계약을 준수하는 Sun의 라이선스 소유자에게도 적용됩니다.

본 서비스 설명서에서 다루는 제품과 수록된 정보는 미국 수출 관리법에 의해 규제되며 다른 국가의 수출 또는 수입 관리법의 적용을 받을 수도 있습니다. 본 제품과 정보를 직간접적으로 핵무기, 미사일 또는 생화학 무기에 사용하거나 핵과 관련하여 해상에서 사용하는 것은 엄격하게 금지합니다. 미국 수출 금지 국가 또는 금지된 개인과 특별히 지정된 국민 목록을 포함하여 미국 수출 금지 목록에 지정된 대상으로의 수출이나 재수출은 엄격하게 금지됩니다.

본 설명서는 "있는 그대로" 제공되며 상업성, 특정 목적에 대한 적합성 또는 비침해성에 대한 모든 묵시적 보증을 포함하여 모든 명시적 또는 묵시적 조건, 표현 및 보증에 대해 어떠한 책임도 지지 않습니다. 이러한 보증 부인은 법적으로 허용된 범위 내에서만 적용됩니다.

목차

표	15
머리말	19
본 설명서의 대상	20
본 설명서에 사용된 규약	20
관리 인터페이스	20
활자체 규칙	21
기호	21
셀 프롬프트	22
관련 문서	22
Java ES 설명서 세트	23
Sun 제품 설명서 검색	24
Sun 자원 온라인 액세스	24
Sun 기술 지원 문의	25
타사 웹 사이트 참조	25
Sun은 여러분의 의견을 환영합니다	25
1장 업그레이드 계획	27
Java ES 5 구성 요소	28
릴리스 5 제품 구성 요소	28
릴리스 5 공유 구성 요소	29
Java ES 업그레이드 기술	31
제품 구성 요소 업그레이드	31
제품 구성 요소 업그레이드 방법	32
각 제품 구성 요소에 사용되는 업그레이드 방법	34
공유 구성 요소 업그레이드	35
업그레이드 프로세스	35

업그레이드 계획 시 고려 사항	36
종속성 업그레이드	37
지원되는 업그레이드 경로 및 전략	38
선택적 업그레이드 또는 전체 업그레이드	40
여러 인스턴스 업그레이드	41
운영 체제 고려 사항	42
필수 운영 체제 패치	42
이중 업그레이드: Java ES 및 운영 체제 소프트웨어	43
운영 체제 업그레이드	45
Solaris 10 다중 영역 환경	45
Java ES 구성 요소 종속성	46
공유 구성 요소에 대한 종속성	46
제품 구성 요소에 대한 종속성	50
업그레이드 순서 지침	54
특별 사례	57
선택적 업그레이드: Application Server 업그레이드하지 않음	57
Portal Server 임시 기능 릴리스(IFR) 7.0을 Java ES 5로 업그레이드	58
Java ES 5 업그레이드 및 Solaris 10 영역	58
Java ES 설치 프로그램의 영역 지원	59
제품 구성 요소 업그레이드	59
모든 공유 구성 요소 동기화	60
권장 업그레이드 방법	60
특별 사례 또는 예외	61
제품 구성 요소 특별 사례	61
공유 구성 요소 특별 사례	62
2장 Java ES 공유 구성 요소 업그레이드	63
공유 구성 요소 업그레이드 개요	64
일반적인 고려 사항	64
공유 구성 요소 동기화	64
모든 공유 구성 요소 동기화	65
Solaris 10 영역 고려사항	66
공유 구성 요소를 업그레이드하는 방법	67
공유 구성 요소 업그레이드 절차	68
특별 업그레이드 절차	68
Java SE 업그레이드 절차	69
Java SE 심볼릭 링크 확인	70
현재 Java SE 버전 확인	70
공통 에이전트 컨테이너 업그레이드 절차	71

3장 Sun Cluster 소프트웨어	75
Sun Cluster 소프트웨어 업그레이드 개요	76
릴리스 5 Sun Cluster 소프트웨어 정보	76
Sun Cluster 소프트웨어 업그레이드 로드맵	77
Sun Cluster 데이터	78
Sun Cluster 업그레이드 전략	78
호환성 문제	78
Sun Cluster 종속성	78
이중 업그레이드	79
Java ES 릴리스 5로 Sun Cluster 소프트웨어 업그레이드	80
소개	80
Sun Cluster 업그레이드	81
사전 업그레이드 작업	81
Sun Cluster 소프트웨어 업그레이드	84
업그레이드 확인	85
사후 업그레이드 작업	85
업그레이드 롤백	86
 4장 Sun Cluster Geographic Edition	 87
Sun Cluster Geographic Edition 업그레이드 개요	88
Java ES 릴리스 5 정보	88
Java ES 릴리스 5 업그레이드 로드맵	88
Sun Cluster Geographic Edition 데이터	89
Sun Cluster Geographic Edition 업그레이드 전략	89
호환성 문제	89
종속성	89
이중 업그레이드	90
버전 3.1 8/05에서 Sun Cluster Geographic Edition 업그레이드	91
소개	91
버전 3.1 8/05 Sun Cluster Geographic Edition 업그레이드	92
사전 업그레이드 작업	92
3.1 8/05 Sun Cluster Geographic Edition 업그레이드	94
업그레이드 확인	97
사후 업그레이드 작업	97
업그레이드 롤백	97

5장 Directory Server	99
Directory Server 업그레이드 개요	100
Java ES 릴리스 5 정보	100
Java ES 릴리스 5 업그레이드 로드맵	100
Directory Server 데이터	101
Directory Server 업그레이드 전략	102
호환성 문제	102
종속성	103
이중 업그레이드	103
Java ES 릴리스 4에서 Directory Server 업그레이드	105
소개	105
릴리스 4 Directory Server 업그레이드	106
사전 업그레이드 작업	106
릴리스 4 업그레이드 Directory Server	107
업그레이드 확인	111
사후 업그레이드 작업	112
업그레이드 롤백	112
여러 인스턴스 업그레이드	112
Directory Server 복제의 롤링 업그레이드	112
Directory Server를 데이터 서비스로 업그레이드	112
Java ES 릴리스 3에서 Directory Server 업그레이드	113
Java ES 릴리스 2에서 Directory Server 업그레이드	114
 6장 Directory Proxy Server	 115
Directory Proxy Server 업그레이드 개요	116
Java ES 릴리스 5 정보	116
Java ES 릴리스 5 업그레이드 로드맵	116
Directory Proxy Server 데이터	117
Directory Proxy Server 업그레이드 전략	118
호환성 문제	118
종속성	118
이중 업그레이드	119
Java ES 릴리스 4에서 Directory Proxy Server 업그레이드	120
소개	120
릴리스 4 Directory Proxy Server 업그레이드	121
사전 업그레이드 작업	121
릴리스 4 업그레이드 Directory Proxy Server	122
업그레이드 확인	124
사후 업그레이드 작업	125
업그레이드 롤백	125
여러 인스턴스 업그레이드	125
Java ES 릴리스 3에서 Directory Proxy Server 업그레이드	126
Java ES 릴리스 2에서 Directory Proxy Server 업그레이드	127

7장 Web Server	129
Web Server 업그레이드 개요	130
Java ES 릴리스 5 Web Server 정보	130
Web Server 업그레이드 로드맵	130
Web Server 데이터	131
Web Server 업그레이드 전략	132
호환성 문제	132
Web Server 종속성	132
이중 업그레이드	133
Java ES 릴리스 4에서 Web Server 업그레이드	134
소개	134
릴리스 4 Web Server 업그레이드	135
사전 업그레이드 작업	135
릴리스 4 업그레이드 Web Server	136
업그레이드 확인	146
사후 업그레이드 작업	147
업그레이드 롤백	150
Java ES 릴리스 3에서 Web Server 업그레이드	151
Java ES 릴리스 2에서 Web Server 업그레이드	152
 8장 Java DB	 153
Java DB 업그레이드 개요	154
Java ES 릴리스 5 정보	154
Java ES 릴리스 5 업그레이드 로드맵	154
Java DB 데이터	155
Java DB 업그레이드 전략	155
호환성 문제	155
종속성	155
이중 업그레이드	155
Java ES 릴리스 4에서 Java DB 업그레이드	157
소개	157
릴리스 4 Java DB 업그레이드	158
사전 업그레이드 작업	158
릴리스 4 업그레이드 Java DB	159
업그레이드 확인	160
사후 업그레이드 작업	161
업그레이드 롤백	161
여러 인스턴스 업그레이드	161

9장 고가용성 세션 저장소	163
HADB 업그레이드 개요	164
Java ES 릴리스 5 HADB 정보	164
HADB 업그레이드 로드맵	164
HADB 데이터터	165
HADB 업그레이드 전략	165
호환성 문제	165
HADB 종속성	165
이중 업그레이드	165
Java ES 릴리스 4에서 HADB 업그레이드	166
소개	166
릴리스 4 HADB 업그레이드	167
사전 업그레이드 작업	167
릴리스 4 업그레이드 HADB	169
업그레이드 확인	171
사후 업그레이드 작업	172
업그레이드 롤백	172
Java ES 릴리스 3에서 HADB 업그레이드	173
 10장 Message Queue	 175
Message Queue 업그레이드 개요	176
Java ES 릴리스 5 Message Queue 정보	176
Message Queue 업그레이드 로드맵	177
Message Queue 데이터터	178
Message Queue 업그레이드 전략	179
호환성 문제	179
Message Queue 종속성	179
이중 업그레이드	180
Java ES 릴리스 4에서 Message Queue 업그레이드	181
소개	181
릴리스 4 Message Queue 업그레이드	182
사전 업그레이드 작업	182
릴리스 4 업그레이드 Message Queue	183
Message Queue 업그레이드 확인	184
사후 업그레이드 작업	185
업그레이드 롤백	185
여러 인스턴스 업그레이드	186
Java ES 릴리스 3에서 Message Queue 업그레이드	187

Java ES 릴리스 2로부터 Message Queue 업그레이드	188
소개	188
릴리스 2 호환성 문제	189
프로토콜 호환성	189
브로커 호환성	189
관리 객체 호환성	190
관리 도구 호환성	190
클라이언트 호환성	191
릴리스 2 Message Queue 업그레이드	191
사전 업그레이드 작업	191
릴리스 2 Message Queue 업그레이드(Solaris)	193
릴리스 2 Message Queue 업그레이드(Linux)	194
호환성 패키지 설치(Linux)	196
Message Queue 업그레이드 확인	196
사후 업그레이드 작업	196
업그레이드 롤백	197
여러 인스턴스 업그레이드	197
11장 Application Server	199
Application Server 업그레이드 개요	201
Java ES 릴리스 5 Application Server 정보	201
Application Server 업그레이드 로드맵	201
Application Server 데이터	204
Application Server 업그레이드 전략	204
호환성 문제	205
Application Server 종속성	205
이중 업그레이드	206
Java ES 릴리스 4에서 Application Server 업그레이드	207
소개	207
릴리스 4 Application Server 업그레이드	208
사전 업그레이드 작업	208
릴리스 4 업그레이드 Application Server	210
업그레이드 확인	212
사후 업그레이드 작업	212
업그레이드 롤백	212
Java ES 릴리스 3에서 Application Server 업그레이드	213
Java ES 릴리스 2에서 Application Server 업그레이드	214
소개	214
릴리스 2 Application Server 업그레이드	215
사전 업그레이드 작업	215
릴리스 2 업그레이드 Application Server	218
업그레이드 확인	222
사후 업그레이드 작업	222
업그레이드 롤백	223
Solaris 10 다중 영역 환경에서 Solaris와 함께 번들로 제공되는 Application Server 업그레이드 ...	224

12장 Service Registry	227
Service Registry 업그레이드 개요	228
Java ES 릴리스 5 정보	228
Java ES 릴리스 5 업그레이드 로드맵	228
Service Registry 데이터	229
Service Registry 업그레이드 전략	229
호환성 문제	229
종속성	229
이중 업그레이드	230
Java ES 릴리스 4에서 Service Registry 업그레이드	231
소개	231
릴리스 4 Service Registry 업그레이드	232
사전 업그레이드 작업	232
릴리스 4 업그레이드 Service Registry	234
업그레이드 확인	237
사후 업그레이드 작업	237
업그레이드 롤백	238
여러 인스턴스 업그레이드	238
 13장 Web Proxy Server	 239
Web Proxy Server 업그레이드 개요	240
Java ES 릴리스 5 Web Proxy Server 정보	240
Web Proxy Server 업그레이드 로드맵	240
Web Proxy Server 데이터	241
Web Proxy Server 업그레이드 전략	241
호환성 문제	241
Web Proxy Server 종속성	241
이중 업그레이드	242
Java ES 릴리스 4에서 Web Proxy Server 업그레이드	243
소개	243
릴리스 4 Web Proxy Server 업그레이드	244
사전 업그레이드 작업	244
릴리스 4 Web Proxy Server 업그레이드(Solaris)	245
릴리스 4 Web Proxy Server 업그레이드(Linux)	246
업그레이드 확인	248
사후 업그레이드 작업	248
업그레이드 롤백(Solaris)	248
버전 3.6에서 Web Proxy Server 업그레이드	250
소개	250
버전 3.6 Web Proxy Server 업그레이드	251
사전 업그레이드 작업	251
버전 3.6 Web Proxy Server 업그레이드	252
업그레이드 확인	253
사후 업그레이드 작업	253
버전 3.6 업그레이드 롤백	253

14장 Access Manager	255
Access Manager 업그레이드 개요	256
Java ES 릴리스 5 Access Manager 정보	256
Access Manager 업그레이드 로드맵	257
Access Manager 데이터	258
Access Manager 업그레이드 전략	259
호환성 문제	259
Access Manager 종속성	260
웹 컨테이너 업그레이드 시나리오	261
이중 업그레이드	261
Java ES 릴리스 4에서 Access Manager 업그레이드	262
소개	262
전체 릴리스 4 Access Manager 업그레이드	263
사전 업그레이드 작업	263
릴리스 4 업그레이드 Access Manager	267
Access Manager 업그레이드 확인	277
사후 업그레이드 작업	278
업그레이드 롤백	279
여러 인스턴스 업그레이드	279
릴리스 4 Access Manager SDK 전용 업그레이드	281
사전 업그레이드 작업	282
릴리스 4 Access Manager SDK 업그레이드	283
Access Manager SDK 업그레이드 확인	283
업그레이드 롤백	284
Java ES 릴리스 3에서 Access Manager 업그레이드	285
릴리스 3 Access Manager 업그레이드	285
여러 인스턴스 업그레이드	285
Java ES 릴리스 2에서 Access Manager 업그레이드	289
사전 업그레이드 작업	289
Access Manager 종속성 업그레이드	290
디렉토리 스키마 업그레이드	290
디렉토리 색인 다시 작성	290
릴리스 2 Access Manager 업그레이드	291
릴리스 2 Access Manager: Web Server 웹 컨테이너 업그레이드	291
릴리스 2 Access Manager: Application Server 웹 컨테이너 업그레이드	291
Access Manager 업그레이드 확인	301
사후 업그레이드 작업	302
업그레이드 롤백	302
여러 인스턴스 업그레이드	302

15장 Portal Server	303
Portal Server 업그레이드 개요	305
Java ES 릴리스 5 Portal Server 정보	305
Portal Server 업그레이드 로드맵	305
Portal Server 데이터	306
Portal Server 업그레이드 전략	307
호환성 문제	307
Portal Server 종속성	308
선택적 업그레이드 문제	309
웹 컨테이너 업그레이드 시나리오	309
이중 업그레이드	310
Java ES 릴리스 4에서 Portal Server 업그레이드	311
소개	311
릴리스 4 Portal Server 업그레이드	312
릴리스 4 사전 업그레이드 작업	313
릴리스 4 Portal Server 업그레이드(Solaris)	318
릴리스 4 Portal Server 업그레이드(Linux)	322
업그레이드 확인	325
릴리스 4 사후 업그레이드 작업	326
업그레이드 롤백(Solaris)	334
업그레이드 롤백(Linux)	335
여러 인스턴스 업그레이드	336
Java ES 릴리스 3에서 Portal Server 업그레이드	340
릴리스 3 사전 업그레이드 작업: Portal Server 종속성 업그레이드	340
릴리스 3 업그레이드 Portal Server	341
릴리스 3 사후 업그레이드 작업	341
여러 인스턴스 업그레이드	342
Java ES 릴리스 2에서 Portal Server 업그레이드	343
릴리스 2 사전 업그레이드 작업	343
릴리스 2 업그레이드 Portal Server	344
릴리스 2 Portal Server: Web Server 웹 컨테이너 업그레이드	344
릴리스 2 Portal Server: Application Server 웹 컨테이너 업그레이드	345
릴리스 2 사후 업그레이드 작업	348
단일 사인 온 구성	348
URLScrapper 채널 활성화	352
게이트웨이 서비스 항목 삭제	352
여러 인스턴스 업그레이드	352
임시 기능 릴리스 7.0에서 Portal Server 업그레이드	353
Portal Server IFR 업그레이드 소개	353

Portal Server IFR 7.0 업그레이드	354
IFR 7 사전 업그레이드 작업	354
Portal Server IFR 7.0 업그레이드(Solaris)	356
IFR 7 Portal Server 업그레이드(Linux)	358
업그레이드 확인	361
IFR 7 사후 업그레이드 작업	362
업그레이드 롤백(Solaris)	369
업그레이드 롤백(Linux)	370
여러 인스턴스 업그레이드	371
16장 Portal Server Secure Remote Access	373
Portal Server Secure Remote Access 업그레이드 개요	375
Java ES 릴리스 5 Portal Server Secure Remote Access 정보	375
Portal Server Secure Remote Access 업그레이드 로드맵	376
Portal Server Secure Remote Access 데이터	377
Portal Server Secure Remote Access 업그레이드 전략	377
호환성 문제	377
Portal Server Secure Remote Access 종속성	378
선택적 업그레이드 문제	379
이중 업그레이드	379
Java ES 릴리스 4로부터 Portal Server Secure Remote Access 업그레이드	380
소개	380
릴리스 4 Portal Server Secure Remote Access 업그레이드	381
릴리스 4 사전 업그레이드 작업	382
릴리스 4 Portal Server Secure Remote Access 업그레이드(Solaris)	385
릴리스 4 Portal Server Secure Remote Access 업그레이드(Linux)	388
업그레이드 확인	390
릴리스 4 사후 업그레이드 작업	390
업그레이드 롤백(Solaris)	392
업그레이드 롤백(Linux)	393
여러 인스턴스 업그레이드	393
Java ES 릴리스 3에서 Portal Server Secure Remote Access 업그레이드	399
Portal Server Secure Remote Access 종속성 업그레이드	399
릴리스 3 업그레이드 Portal Server Secure Remote Access	400
여러 인스턴스 업그레이드	400
Java ES 릴리스 2에서 Portal Server Secure Remote Access 업그레이드	401
릴리스 2 사전 업그레이드 작업	401
Portal Server Secure Remote Access 종속성 업그레이드	401
게이트웨이 서비스 항목 삭제	402
릴리스 2 업그레이드 Portal Server Secure Remote Access	402
릴리스 2 사후 업그레이드 작업	403
Proxylet 서비스용 Portal Server 도메인 설정	403
여러 인스턴스 업그레이드	403

임시 기능 릴리스 7.0에서 Portal Server Secure Remote Access 업그레이드	404
소개	404
Portal Server Secure Remote Access IFR 7.0 업그레이드	405
사전 업그레이드 작업	405
Portal Server Secure Remote Access IFR 7.0 업그레이드(Solaris)	405
Portal Server Secure Remote Access IFR 7.0 업그레이드(Linux)	407
업그레이드 확인	409
사후 업그레이드 작업	409
업그레이드 롤백(Solaris)	409
업그레이드 롤백(Linux)	410
여러 인스턴스 업그레이드	410
부록 A Java Enterprise System 릴리스 내용	411
Java ES 2003Q4(릴리스 1)	412
릴리스 1 설치 프로그램-선택 가능한 구성 요소	412
릴리스 1 공유 구성 요소	413
Java ES 2004Q2(릴리스 2)	414
릴리스 2 설치 프로그램-선택 가능한 구성 요소	414
릴리스 2 공유 구성 요소	415
Java ES 2005Q1(릴리스 3)	417
릴리스 3 설치 프로그램에서 선택 가능한 구성 요소	417
릴리스 3 공유 구성 요소	420
Java ES 2005Q4(릴리스 4)	422
릴리스 4 설치 프로그램-선택 가능한 구성 요소	422
릴리스 4 공유 구성 요소	425
Java ES 5(릴리스 5)	427
릴리스 5 설치 프로그램-선택 가능한 구성 요소	427
릴리스 5 공유 구성 요소	430
색인	433

표 1	활자체 규칙	21
표 2	기호 규칙	21
표 3	셀 프롬프트	22
표 4	Java Enterprise System 설명서	23
표 1-1	Java ES 5 제품 구성 요소	28
표 1-2	Java ES 5 공유 구성 요소	29
표 1-3	Java ES 제품 구성 요소 업그레이드 방법	34
표 1-4	업그레이드 프로세스의 단계	36
표 1-5	Java ES 5(릴리스 5)로 업그레이드하는 방법	39
표 1-6	선택적 업그레이드와 전체 업그레이드 비교	41
표 1-7	Java ES/운영 체제 지원 매트릭스	43
표 1-8	Java ES 5 제품 구성 요소 이중 업그레이드 지원	44
표 1-9	Java ES 5(릴리스 5) 제품 구성 요소의 공유 구성 요소 종속성	47
표 1-10	Java ES 제품 구성 요소 종속성	51
표 1-11	Application Server를 릴리스 5로 업그레이드하지 않을 때 필요한 패치	57
표 2-1	Java SE에 대한 플랫폼별 심볼릭 링크	70
표 2-2	Java SE 버전 확인 출력	71
표 2-3	공통 에이전트 컨테이너 디렉토리 경로	71
표 3-1	Java ES 5(릴리스 5) 업그레이드 경로: Sun Cluster 3.1 8/05	77
표 3-2	Sun Cluster 데이터 사용	78
표 3-3	Sun Cluster 버전 확인 출력	82
표 4-1	Sun Cluster Geographic Edition 디렉토리 경로	87
표 4-2	Java ES 5(릴리스 5) 업그레이드 경로: Sun Cluster Geographic Edition 3.1 2006Q4	88
표 4-3	Sun Cluster Geographic Edition 데이터 사용	89

표 4-4	Sun Cluster Geographic Edition 버전 확인 출력	92
표 5-1	Directory Server 디렉토리 경로	99
표 5-2	Java ES 5(릴리스 5) 업그레이드 경로: Directory Server 6.0	100
표 5-3	Directory Server 데이터 사용	102
표 5-4	Directory Server 버전 확인 출력	106
표 6-1	Directory Proxy Server 디렉토리 경로	115
표 6-2	Java ES 5(릴리스 5) 업그레이드 경로: Directory Proxy Server 6.0	116
표 6-3	Directory Proxy Server 데이터 사용	117
표 6-4	Directory Proxy Server 버전 확인 출력	121
표 7-1	Web Server 디렉토리 경로	129
표 7-2	Java ES 5(릴리스 5) 업그레이드 경로: Web Server 7.0	130
표 7-3	Web Server 데이터 사용	131
표 7-4	Web Server 기본 이름	132
표 7-5	Web Server 버전 확인 출력	135
표 7-6	WSInstall.properties 값	140
표 7-7	wadm 공통 명령 옵션	143
표 7-8	wadm migrate-server 명령 옵션 및 피연산자	144
표 7-9	wadm create-instance 명령 옵션 및 피연산자	145
표 7-10	wadm set-virtual-server-prop 명령 옵션 및 피연산자	147
표 7-11	wadm deploy-cofig 명령 옵션 및 피연산자	148
표 7-12	wadm set-search-collection-prop 명령 옵션 및 피연산자	149
표 8-1	Java DB 디렉토리 경로	153
표 8-2	Java ES 5(릴리스 5) 업그레이드 경로: Java DB 10.1.3	154
표 8-3	Java DB 데이터 사용	155
표 8-4	Java DB 버전 확인 출력	158
표 9-1	HADB 디렉토리 경로	163
표 9-2	Java ES 5(릴리스 5) 업그레이드 경로: HADB 4.4.3	164
표 9-3	HADB 데이터 사용	165
표 9-4	HADB 버전 확인 출력	168
표 9-5	Solaris 플랫폼에서 HADB 업그레이드를 위한 패키지 버전	169
표 10-1	Message Queue 디렉토리 경로	175
표 10-2	Java ES 5(릴리스 5) 업그레이드 경로: Message Queue 3.7 UR1	177
표 10-3	Message Queue 데이터 사용(Solaris OS)	178
표 10-4	Message Queue 버전 확인 출력	182

표 10-5	Message Queue 버전 확인 출력	192
표 11-1	Application Server 디렉토리 경로	200
표 11-2	Java ES 5(릴리스 5) 업그레이드 경로: Application Server Enterprise Edition 8.2	202
표 11-3	Solaris OS에 번들로 제공된 Application Server 버전 업그레이드 방법	203
표 11-4	Application Server 데이터 사용	204
표 11-5	Application Server 버전 확인 출력	208
표 12-1	Service Registry 디렉토리 경로	227
표 12-2	Java ES 5(릴리스 5) 업그레이드 경로: Service Registry 3.1	228
표 12-3	Service Registry 데이터 사용	229
표 12-4	Service Registry 버전 확인 출력	233
표 13-1	Web Proxy Server 디렉토리 경로	239
표 13-2	Java ES 5(릴리스 5) 업그레이드 경로: Web Proxy Server 4.0.4	240
표 13-3	Web Proxy Server 데이터 사용	241
표 13-4	Web Proxy Server 버전 확인 출력	244
표 13-5	Solaris에서 Web Proxy Server를 업그레이드하기 위한 패치	245
표 13-6	Linux에서 Web Proxy Server를 업그레이드하기 위한 패치	247
표 13-7	Web Proxy Server 버전 확인 출력	251
표 14-1	Access Manager 디렉토리 경로	255
표 14-2	Java ES 5(릴리스 5) 업그레이드 경로: Access Manager 7.1	257
표 14-3	Access Manager 데이터 사용	258
표 14-4	Access Manager 업그레이드를 위한 웹 컨테이너 업그레이드 시나리오	261
표 14-5	Access Manager 버전 확인 출력	264
표 14-6	Access Manager Mobile Access 소프트웨어를 업그레이드하기 위한 패치	268
표 14-7	Mobile Access 패치 매개 변수	268
표 14-8	Access Manager 구성 매개 변수: ampre71upgrade	269
표 14-9	amconfig에서 지원하는 관리 서버 모드	272
표 14-10	Access Manager 구성 매개 변수: amconfig	273
표 14-11	amconfig 매개 변수: 릴리스 5 Web Server	275
표 14-12	Access Manager 구성 매개 변수: amupgrade	276
표 14-13	Access Manager Mobile Access 소프트웨어를 업그레이드하기 위한 패치	293
표 14-14	Mobile Access 패치 매개 변수	293
표 14-15	Access Manager 구성 매개 변수: ampre71upgrade	294
표 14-16	Access Manager 구성 매개 변수: amconfig	297
표 14-17	Access Manager 구성 매개 변수: amupgrade	300

표 15-1	Portal Server 디렉토리 경로	304
표 15-2	Java ES 5(릴리스 5) 업그레이드 경로: Portal Server 7.1	305
표 15-3	Portal Server 데이터 사용	306
표 15-4	Portal Server 업그레이드를 위한 웹 컨테이너 업그레이드 시나리오	310
표 15-5	Portal Server 버전 확인 출력	313
표 15-6	웹 컨테이너 업그레이드 시나리오마다 psupgrade 스크립트에 필요한 정보	315
표 15-7	JVM 설정의 위치	317
표 15-8	Portal Server IFR을 릴리스 5로 업그레이드하는 패치	353
표 16-1	Portal Server Secure Remote Access 디렉토리 경로	374
표 16-2	Java ES 5(릴리스 5) 업그레이드 경로: Portal Server Secure Remote Access 7.1	376
표 16-3	Portal Server Secure Remote Access 데이터 사용	377
표 16-4	Portal Server Secure Remote Access 버전 확인 출력	382
표 16-5	Portal Server Secure Remote Access IFR을 릴리스 5로 업그레이드하는 패치	404

머리말

Java Enterprise System UNIX 용 업그레이드 설명서에는 Sun Solaris™ 운영 체제 (Solaris OS) 또는 Red Hat Enterprise Linux(RHEL) 운영 체제 환경에서 Sun Java™ Enterprise System(Java ES) 소프트웨어를 업그레이드하는 데 필요한 정보가 포함되어 있습니다. HP-UX 환경에서의 업그레이드는 설명하지 않습니다.

본 설명서에서는 Java ES 2004Q2(릴리스 2), Java ES 2005Q1(릴리스 3) 및 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로의 업그레이드에 대해 설명합니다.

이 머리말의 내용은 다음과 같습니다.

- 20페이지의 "본 설명서의 대상"
- 20페이지의 "본 설명서에 사용된 규약"
- 22페이지의 "관련 문서"
- 24페이지의 "Sun 자원 온라인 액세스"
- 25페이지의 "Sun 기술 지원 문의"
- 25페이지의 "타사 웹 사이트 참조"
- 25페이지의 "Sun은 여러분의 의견을 환영합니다"

본 설명서의 대상

본 설명서는 Java ES 소프트웨어를 업그레이드하려는 시스템 관리자나 소프트웨어 기술자를 위한 것입니다.

본 설명서에서는 사용자가 다음과 같은 작업에 익숙하다고 가정합니다.

- 엔터프라이즈 수준의 소프트웨어 제품 설치
- 현재 사용자 환경에 배포된 Java ES 구성 요소
- 지원되는 Java ES 플랫폼에서 시스템 관리 및 네트워킹
- 클러스터링 모델(클러스터링 소프트웨어를 설치하는 경우)

본 설명서에 사용된 규약

설명서에 사용된 표기 규칙이 이 절에서 표로 나와 있습니다.

관리 인터페이스

본 업그레이드 설명서에서 설명하는 대부분의 업그레이드 절차에는 그래픽 사용자 인터페이스(GUI)와 명령줄 인터페이스의 두 가지 관리 인터페이스가 사용됩니다.

본 업그레이드 설명서에서 Java ES 구성 요소 업그레이드 절차를 설명하는 데는 주로 명령줄 인터페이스가 사용되었습니다. 명령줄 인터페이스를 사용하여 Java ES 배포 업그레이드를 스크립팅하면 필요할 때 손쉽게 절차를 반복할 수 있습니다.

하지만 업그레이드 절차에서 Java ES 설치 프로그램을 사용하는 경우 대화식 텍스트 기반 인터페이스가 아닌 GUI 인터페이스에 대해 설명합니다. Java ES 설치 프로그램 세션은 상태 파일로 저장하여 필요할 때 절차를 반복하는 데 사용할 수 있습니다.

활자체 규칙

다음 표에는 본 설명서에 사용된 활자체 규칙 변경 사항이 나와 있습니다.

표 1 활자체 규칙

서체	의미	예
AaBbCc123 (고정 폭 글꼴)	API 및 언어 요소, HTML 태그, 웹 사이트 URL , 명령 이름, 파일 이름, 디렉토리 경로 이름, 컴퓨터 화면에 출력되는 내용, 샘플 코드입니다.	.login 파일을 편집합니다. ls -a를 사용하여 모든 파일을 나열합니다. % You have mail.
AaBbCc123 (굵은 고정 폭 글꼴)	화면 상의 컴퓨터 출력과 반대로 사용자가 직접 입력하는 내용입니다.	% su Password:
AaBbCc123 (기울임꼴)	실제 이름이나 값으로 바꾸어야 할 명령이나 경로 이름의 자리 표시자입니다.	이것을 <i>클래스</i> 옵션이라고 합니다. 파일을 저장하지 <i>마십시오</i> . 이 파일은 <i>install-dir/bin</i> 디렉토리에 들어 있습니다.
AaBbCc123	책 제목, 장, 절	사용 설명서의 6 장을 참조하십시오.

기호

다음 표에는 본 설명서에 사용된 기호 규칙이 나와 있습니다.

표 2 기호 규칙

기호	설명	예	의미
[]	선택적인 명령 옵션이 포함됩니다.	ls [-l]	-l 옵션은 선택 옵션입니다.
{ }	선택할 수 있는 필수 명령 옵션 집합이 포함됩니다.	-d {y n}	-d 옵션에는 y 인수 또는 n 인수를 사용해야 합니다.
-	동시에 눌러야 하는 여러 키를 결합합니다.	Control-A	A 키를 누른 상태에서 Control 키를 누릅니다.

표 2 기호 규칙 (계속)

기호	설명	예	의미
+	연속적으로 눌러야 하는 여러 키를 결합합니다.	Ctrl+A+N	Control 키를 눌렀다 놓고 나서 나머지 키를 순서대로 누릅니다.
>	그래픽 사용자 인터페이스에서 메뉴 항목 선택을 나타냅니다.	파일 > 새로 만들기 > 템플리트	파일 메뉴에서 새로 만들기를 선택합니다. 새로 만들기 하위 메뉴에서 템플리트를 선택합니다.

셸 프롬프트

다음 표는 본 설명서에 사용된 셸 프롬프트를 설명합니다.

표 3 셸 프롬프트

셸	프롬프트
UNIX 또는 Linux의 C 셸	<i>machine-name%</i>
UNIX 또는 Linux의 C 셸 슈퍼유저	<i>machine-name#</i>
UNIX 또는 Linux의 Bourne 셸 및 Korn 셸	\$
UNIX 또는 Linux의 Bourne 셸 및 Korn 셸 슈퍼유저	#
Windows 명령줄	C:\

관련 문서

<http://docs.sun.com>SM 웹 사이트에서 Sun 기술 설명서를 온라인으로 액세스할 수 있습니다. 아카이브를 찾아보거나 특정한 제목이나 주제를 검색할 수 있습니다.

Java ES 설명서 세트

Java ES 설명서는 PDF 및 HTML 형식의 온라인 파일로 사용할 수 있습니다. 두 형식 모두 장애자용 보조 기술로 판독 가능합니다. Sun™ 설명서는 다음 웹 사이트에서 액세스할 수 있습니다.

<http://docs.sun.com>

Java ES 설명서에는 전체 시스템 정보와 구성 요소 정보가 포함되어 있습니다. 이 설명서는 다음 웹 사이트에서 액세스할 수 있습니다.

<http://docs.sun.com/coll/1397.1>

다음 표에는 Java ES 설명서 세트의 시스템 수준 설명서가 나열되어 있습니다. 왼쪽 열은 각 문서의 이름과 부품 번호 위치를 제공하고 오른쪽 열은 문서의 일반 내용을 설명합니다.

표 4 Java Enterprise System 설명서

문서	목차
Java Enterprise System 5 UNIX 용 릴리스 노트 http://docs.sun.com/doc/820-0454	알려진 문제점을 포함하여 Java ES 에 대한 최신 정보를 포함합니다. 또한, 구성 요소마다 릴리스 노트가 있습니다.
Java Enterprise System 5 기술 개요 http://docs.sun.com/doc/820-0885	Java ES 의 기술적 및 개념적 기초를 소개합니다. 구성 요소, 구조, 과정 및 기능을 설명합니다.
Java Enterprise System 배포 계획 설명서 http://docs.sun.com/doc/819-2326	Java ES 을 기반으로 하는 엔터프라이즈 배포 솔루션의 계획 및 설계를 소개합니다. 배포 계획 및 설계의 기본 개념 및 원칙을 소개하고 솔루션 라이프 사이클을 설명하며 Java ES 기반 솔루션 계획 시 사용할 수 있는 고급 예 및 전략을 제공합니다.
Java Enterprise System 5 설치 계획 설명서 http://docs.sun.com/doc/820-0878	Java ES 배포의 하드웨어, 운영 체제 및 네트워크 측면에 대한 구현 사양을 개발하는 데 도움이 됩니다. 설치 및 구성 계획에서의 구성 요소 종속성과 같은 문제를 설명합니다.
Java Enterprise System 5 UNIX 용 설치 설명서 http://docs.sun.com/doc/820-0102	Solaris 운영 체제 또는 Linux 운영 체제에 Java ES 을 설치하는 과정 전체를 설명합니다. 또한 설치 후 구성 요소를 구성하는 방법과 기능이 올바르게 작동하는지 확인하는 방법도 설명합니다.
Java Enterprise System 5 UNIX 용 설치 참조 설명서 http://docs.sun.com/doc/820-0588	구성 매개 변수에 대한 추가 정보와 구성 계획에 사용할 워크시트를 제공하며 기본 디렉토리 및 포트 번호와 같은 참조 자료를 나열합니다.
Java Enterprise System 5 UNIX 용 업그레이드 설명서 http://docs.sun.com/doc/819-6553	Solaris 운영 체제 또는 Linux 운영 환경에서 Java ES 업그레이드를 위한 지침을 제공합니다.

표 4 Java Enterprise System 설명서 (계속)

문서	목적
Sun Java Enterprise System 5 모니터링 설명서 http://docs.sun.com/doc/820-1591	각 제품 구성 요소에 대해 Monitoring Framework 를 설정하고 Monitoring Console 을 사용하여 실시간 데이터를 보고 경보 임계값을 설정하는 방법에 대한 지침을 제공합니다.
Java Enterprise System 용어집 http://docs.sun.com/doc/819-3875	Java ES 설명서에 사용되는 용어를 정의합니다.

Sun 제품 설명서 검색

docs.sun.com 웹 사이트에서 Sun 제품 설명서를 검색하는 방법 외에도 검색 필드에 다음 구문을 입력하여 검색 엔진을 사용할 수 있습니다.

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

예를 들어 "broker"를 검색하려면 다음을 입력합니다.

`broker site:docs.sun.com`

다른 Sun 웹 사이트(예를 들어, java.sun.com, www.sun.com, developers.sun.com)를 검색하려면 검색 필드에 docs.sun.com 대신 sun.com을 사용합니다.

Sun 자원 온라인 액세스

제품 다운로드, 전문가 서비스, 패치, 지원 및 추가 개발자 정보를 보려면 다음으로 이동하십시오.

- 다운로드 센터
<http://www.sun.com/software/download/>
- 클라이언트 솔루션
<http://www.sun.com/service/sunjavasystem/sjsservicessuite.html>
- Sun 엔터프라이즈 서비스, Solaris 패치 및 지원
<http://sunsolve.sun.com/>
- 개발자 정보
<http://developers.sun.com>

다음 위치의 웹 페이지에는 Java Enterprise System과 해당 구성 요소에 대한 정보가 포함되어 있습니다.

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

Sun 기술 지원 문의

이 제품에 대한 기술적인 의문점이 있는 경우 그 답을 설명서에서 찾을 수 없다면 <http://www.sun.com/service/contacting>에서 문의하십시오.

타사 웹 사이트 참조

Sun은 본 설명서에서 언급된 타사 웹 사이트의 가용성 여부에 대해 책임을 지지 않습니다. 또한 Sun은 해당 사이트나 리소스를 통해 제공되는 내용, 광고, 제품 및 기타 자료에 대해 어떠한 보증도 하지 않으며 그에 대한 책임도 지지 않습니다. 따라서 Sun은 타사 웹 사이트의 내용, 제품 또는 리소스의 사용으로 인해 발생한 실제 또는 주장된 손상이나 피해에 대해서도 책임을 지지 않습니다.

Sun은 여러분의 의견을 환영합니다

Sun은 설명서의 내용 개선에 노력을 기울이고 있으며, 여러분의 의견과 제안을 환영합니다.

<http://docs.sun.com>으로 가서 "의견 보내기"를 눌러 여러분의 의견을 제출하여 주십시오. 해당 필드에 전체 설명서 제목과 부품 번호를 입력해 주십시오. 부품 번호는 해당 설명서의 제목 페이지나 문서 맨 위에 있으며 일반적으로 7자리 또는 9자리 숫자입니다. 사용자 의견을 제출할 때 해당 양식에 영문 설명서 제목과 부품 번호를 입력해야 할 수도 있습니다. 본 설명서의 영문 부품 번호와 제목은 819-6553, Sun Java Enterprise System 5 Upgrade Guide for UNIX입니다.

Sun은 여러분의 의견을 환영합니다

업그레이드 계획

이 장에서는 Sun Solaris™ 운영 체제 또는 Red Hat Enterprise Linux(간단히 Linux) 운영 체제 환경에서 Sun Java™ Enterprise System(Java ES) 소프트웨어를 Java ES 5로 업그레이드하기 위한 계획을 세우는 데 도움이 되는 정보를 제공합니다.

이 장은 다음 내용으로 구성되어 있습니다.

- 28페이지의 "Java ES 5 구성 요소"
- 31페이지의 "Java ES 업그레이드 기술"
- 35페이지의 "업그레이드 프로세스"
- 36페이지의 "업그레이드 계획 시 고려 사항"
- 46페이지의 "Java ES 구성 요소 종속성"
- 54페이지의 "업그레이드 순서 지침"
- 57페이지의 "특별 사례"
- 58페이지의 "Java ES 5 업그레이드 및 Solaris 10 영역"

Java ES 5 구성 요소

Java ES 소프트웨어 업그레이드를 계획하는 시작 단계로, 이 절에서는 Java ES 5(릴리스 5)에 포함된 구성 요소에 대해 검토합니다. 업그레이드 시나리오에 따라 이러한 구성 요소를 하나 이상 릴리스 5 버전으로 업그레이드해야 할 수 있습니다.

Java ES 구성 요소는 Java Enterprise System 5 기술 개요

(<http://docs.sun.com/doc/820-0885>)에 설명된 대로 여러 유형으로 그룹화됩니다.

- **제품 구성 요소.** Java ES 제품 구성 요소는 다음 항목으로 구성됩니다.

- 주 Java ES 인프라 서비스를 제공하는 시스템 서비스 구성 요소
- 시스템 서비스를 향상시키는 서비스 품질 구성 요소

제품 구성 요소는 Java ES 설치 프로그램에서 선택할 수 있습니다.

- **공유 구성 요소.** Java ES 공유 구성 요소는 Java ES 제품 구성 요소가 종속되는 로컬 공유 라이브러리입니다. 공유 구성 요소는 Java ES 설치 프로그램에서 자동으로 설치합니다. 설치된 제품 구성 요소에 따라 설치할 공유 구성 요소가 결정됩니다.

릴리스 5 제품 구성 요소

다음 표에는 릴리스 5 제품 구성 요소가 영문자순으로 정리되어 있으며, 그 다음 표에서 사용할 해당 약어도 표시되어 있습니다. 이 중 서비스 품질 구성 요소의 경우 제공하는 개선된 서비스 유형이 표에 포함되어 있습니다.

표 1-1 Java ES 5 제품 구성 요소

제품 구성 요소	약어	버전	유형
Access Manager	AM	7.1	시스템 서비스 구성 요소
Application Server	AS	8.2	시스템 서비스 구성 요소
Directory Proxy Server	DPS	6.0	서비스 품질: 액세스 구성 요소
Directory Server	DS	6.0	시스템 서비스 구성 요소
고가용성 세션 저장소	HADB	4.4.3	서비스 품질: 가용성 구성 요소
Java DB	JavaDB	10.2	시스템 서비스 구성 요소
Message Queue	MQ	3.7 UR1	시스템 서비스 구성 요소
Monitoring Console	MC	1.0	서비스 품질: 관리 구성 요소
Portal Server	PS	7.1	시스템 서비스 구성 요소

표 1-1 Java ES 5 제품 구성 요소 (계/속)

제품 구성 요소	약어	버전	유형
Portal Server Secure Remote Access	PSRA	7.1	서비스 품질: 액세스 구성 요소
Service Registry	SR	3.1	시스템 서비스 구성 요소
Sun Cluster	SC	3.1 8/05	서비스 품질: 가용성 구성 요소
Sun Cluster Geographic Edition	SCG	2006Q4	서비스 품질: 가용성 구성 요소
Web Proxy Server	WPS	4.0.4	서비스 품질: 액세스 구성 요소
Web Server	WS	7.0	시스템 서비스 구성 요소

릴리스 5 공유 구성 요소

다음 표에는 릴리스 5 공유 구성 요소가 영문자순으로 정리되어 있으며, 다음에 나올 표에서 사용할 해당 약어도 표시되어 있습니다.

표 1-2 Java ES 5 공유 구성 요소

공유 구성 요소	버전	약어
Apache Commons Logging	1.0.3	ACL
Jakarta ANT Java/XML 기반 빌드 도구	1.6.5	ANT
Berkeley 데이터베이스	4.2.52	BDB
공통 에이전트 컨테이너	1.1 및 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™ 런타임	2.0	JavaHelp
JavaMail™ 런타임	1.3.2	JavaMail
Java Architecture for XML Binding 런타임	2.0.3	JAXB
Java API for XML Processing	1.3.1	JAXP
Java API for XML Registries 런타임	1.0.8	JAXR

표 1-2 Java ES 5 공유 구성 요소 (계/속)

공유 구성 요소	버전	약어
Java API for XML-based Remote Procedure Call 런타임	1.1.3_01	JAX-RPC
Java API for Web Services 런타임	2.0	JAXWS
Java Calendar API	1.2	JCAPI
Java Dynamic Management™ Kit 런타임	5.1.2	JDMK
Java Security Services(Network Security Services for Java)	4.2.4 및 3.1.11	JSS 및 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(Solaris 전용)	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 업그레이드 기술

하나의 시스템 유틸리티가 모든 Java ES 구성 요소를 업그레이드하지는 않습니다. 또한 이 절의 아래에서 설명하는 것처럼 제품 구성 요소와 공유 구성 요소 업그레이드에는 서로 다른 특성과 업그레이드 기술이 사용됩니다.

제품 구성 요소 업그레이드

Java ES 제품 구성 요소를 릴리스 5로 업그레이드하는 작업은 업그레이드 설명서에서 설명하는 구성 요소별 업그레이드 절차를 사용하여 구성 요소 및 컴퓨터별로 수행됩니다.

제품 구성 요소 업그레이드는 이전 버전의 구성 요소와 호환되지 않는 주요 기능 업그레이드부터 이전 버전과 완전히 호환되는 버그 수정 업그레이드까지 다양합니다. Java ES 구성 요소 간의 종속성으로 인해 업그레이드의 특성에 따라 다른 구성 요소도 함께 업그레이드해야 할 수 있습니다.

Java ES 제품 구성 요소 업그레이드는 Java ES 제품 구성 요소 초기 설치와 구성의 두 가지 기본 작업으로 이루어집니다.

- **소프트웨어 업그레이드 설치.** 업그레이드된 소프트웨어는 기존 소프트웨어를 개선 또는 수정하거나 기존 소프트웨어를 대체합니다. 소프트웨어 설치의 기존 소프트웨어 패키지에 패치를 적용하거나 기존 패키지를 선택적으로 대체하거나 새 패키지를 설치하거나 구성 요소 소프트웨어를 완전히 다시 설치하는 방법으로 수행할 수 있습니다.
- **재구성.** 재구성에는 업그레이드된 소프트웨어를 지원하는 데 필요한 구성 데이터, 사용자 데이터 또는 동적 응용 프로그램 데이터의 변경 내용이 포함됩니다. 데이터의 변경 작업에는 데이터 추가, 데이터 형식 변경(등록 정보 파일 또는 데이터베이스 스키마) 또는 새 위치로 데이터 마이그레이션 등이 포함될 수 있습니다. 재구성은 사용자가 절차를 수행하는 경우도 있고 자동으로 수행되는 경우도 있습니다. 또한 구성 요소 소프트웨어를 웹 컨테이너에 재배포해야 하는 경우도 있습니다.

Java ES 제품 구성 요소를 업그레이드할 때는 일반적으로 사전 업그레이드 작업이 필요하며 경우에 따라 사후 업그레이드 절차를 수행해야 업그레이드가 제대로 작동하는 경우도 있습니다.

제품 구성 요소 업그레이드 방법

업그레이드된 소프트웨어를 설치하고 구성 요소 재구성을 수행하는 구성 요소별 업그레이드 절차에는 다음과 같이 업그레이드 방법이 있습니다.

- [Java ES 설치 프로그램의 업그레이드 기능 사용](#)
- [제품 구성 요소 새로 설치](#)
- [구성 요소별 업그레이드 유틸리티 실행](#)
- [기존 구성 요소 패키지 패치](#)

Java ES 설치 프로그램의 업그레이드 기능 사용

릴리스 5 설치 프로그램에는 Application Server, Message Queue, HADB 및 Java DB와 같은 몇 가지 특별한 경우에 제품 구성 요소 업그레이드를 수행할 수 있는 업그레이드 기능이 있습니다. Java ES 설치 프로그램이 이러한 제품 구성 요소에 대해 이전에 설치된 릴리스 버전을 검색하면 "업그레이드 가능"으로 표시합니다.

이러한 구성 요소를 업그레이드하기 전에 설치 프로그램이 공유 구성 요소의 현재 버전과 이전 버전을 확인합니다. 선택한 구성 요소에 필요한 공유 구성 요소가 이전 버전이거나 해당 공유 구성 요소가 누락되었음을 설치 프로그램이 감지하면 현재 설치된 모든 공유 구성 요소를 업그레이드하고 선택한 구성 요소에서 필요한 누락된 공유 구성 요소를 설치합니다. 일부 경우(특히 Application Server) 설치 프로그램은 업그레이드 중인 구성 요소가 종속된 제품 구성 요소도 업그레이드합니다.

설치 프로그램은 이전 버전의 패키지를 제거하고 릴리스 5 제품 구성 요소 패키지를 설치하며 필요한 경우 업그레이드하는 제품 구성 요소를 재구성합니다. 하지만 Solaris 9 운영 체제와 함께 번들로 제공된 Application Server의 경우 설치 프로그램이 패키지를 제거하지 않습니다([215페이지의 "릴리스 2 Application Server 업그레이드" 참조](#)).

Solaris 10 운영 체제 영역 기능을 사용하는 경우 특별한 고려 사항이 적용됩니다. [59페이지의 "Java ES 설치 프로그램의 영역 지원"](#)을 참조하십시오.

제품 구성 요소 새로 설치

일부 제품 구성 요소는 Java ES 설치 프로그램을 사용하여 구성 요소를 새로 설치하면 업그레이드됩니다. 먼저 이전 버전의 패키지를 제거하고 같은 경로에 릴리스 5를 설치하거나 병렬 경로에 릴리스 5를 설치하여 이전 버전을 그대로 둡니다.

두 경우 모두 이전 버전의 구성 데이터를 새로운 설치로 마이그레이션하거나 구성을 새로 수행하거나 이 두 가지를 조합하여 제품 구성 요소를 재구성할 수 있습니다. 일부 제품 구성 요소에는 해당 구성 요소의 재구성 또는 구성 데이터 마이그레이션을 위한 유틸리티가 제공됩니다.

구성 요소별 업그레이드 유틸리티 실행

일부 제품 구성 요소는 구성 요소를 릴리스 5로 자동 업그레이드하는 업그레이드 유틸리티 또는 스크립트를 제공합니다. 일반적으로 유틸리티는 소프트웨어 패키지 업그레이드와 업그레이드에 필요한 재구성을 함께 수행합니다. 웹 컨테이너에 배포된 구성 요소의 경우 일반적으로 유틸리티는 업그레이드된 구성 요소 소프트웨어를 웹 컨테이너로 재배포합니다.

기존 구성 요소 패키지 패치

일부 제품 구성 요소의 경우 기존 소프트웨어 패키지를 수동으로 패치하여 업그레이드를 수행합니다. Solaris와 Linux 플랫폼에서는 설치된 소프트웨어 패키지를 관리하고 패키지 레지스트리를 통해 패키지의 변경 사항을 추적하는 데 비슷한 기술을 사용하지만 플랫폼 간 패치 기술의 차이 때문에 업그레이드 절차가 달라질 수 있습니다.

- **Solaris 플랫폼.** Solaris pkgadd 및 pkgrm 명령을 사용하여 패키지를 설치 및 제거할 수 있습니다. 설치된 패키지 내용은 patchadd 및 patchrm 명령을 통해 패치를 적용 또는 제거하여 수정할 수 있습니다. Solaris 패키지에 대한 패치는 SunSolve 웹 사이트 (<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>)를 통해 배포됩니다.

Solaris 패치는 하나 이상의 패키지에 적용할 수 있습니다. patchadd 명령은 패치되는 패키지의 백업을 저장하여 patchrm 명령을 사용한 패치 제거를 용이하게 합니다. 패치는 패치 아이디로 식별하며 패치 아이디는 패치 번호와 개정판 번호로 이루어져 있습니다. 개정판 번호는 패치가 수정됨에 따라 커집니다.

- **Linux 플랫폼.** Red Hat Enterprise Linux 패키지(RPM)는 rpm 명령을 사용하여 설치 또는 업데이트할 수 있습니다. 그러나 설치된 패키지 내용은 패치를 사용하여 수정할 수 없습니다. 대신 RPM 패키지는 rpm -U 명령 옵션을 사용하여 현재 패키지를 새 패키지로 대체하는 방법으로 업데이트합니다.

편의상 많은 RPM 패키지 업그레이드가 Java ES 릴리스 5 배포본 아니라 SunSolve 웹 사이트

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

SunSolve를 통한 배포의 경우 RPM 패키지를 패치로 캡슐화하여 Solaris 패치와 비슷한 패치 아이디와 개정판 번호를 할당할 수 있습니다. 이러한 Linux 패치에는 하나 이상의 RPM 패키지가 포함되며, 각 패키지는 고유한 RPM 이름, RPM 번호 및 개정판 번호로 식별됩니다. 개정판 번호는 RPM 패키지가 수정됨에 따라 커집니다.

각 제품 구성 요소에 사용되는 업그레이드 방법

다음 표에서는 각 제품 구성 요소를 릴리스 5로 업그레이드하는 방법을 보여 줍니다.

표 1-3 Java ES 제품 구성 요소 업그레이드 방법

제품 구성 요소	업그레이드된 소프트웨어 설치	재구성
Access Manager	패키지 바꾸기: 패키지 제거 스크립트 사용 및 새로 설치	amconfig 및 amupgrade 스크립트를 사용하여 웹 컨테이너에 대한 재구성 및 재배포 수행
Application Server	패키지 바꾸기: Java ES 설치 프로그램의 업그레이드 기능 사용	없음. 단, postinstall 및 asupgrade 스크립트를 사용하여 릴리스 2에서 업그레이드한 경우 제외
Directory Proxy Server	이전 패키지를 바꾸지 않고 새로 설치 수행	수동 재구성
Directory Server	이전 패키지를 바꾸지 않고 새로 설치 수행	dsmig 명령을 사용하여 디렉토리 데이터 마이그레이션
고가용성 세션 저장소	패키지 바꾸기: Java ES 설치 프로그램의 업그레이드 기능 사용 또는 병렬 새로 설치	없음
Java DB	패키지 바꾸기: Java ES 설치 프로그램의 업그레이드 기능 사용	없음
Message Queue	패키지 바꾸기: Java ES 설치 프로그램의 업그레이드 기능 또는 mqupgrade 스크립트 사용 (릴리스 2에서)	없음. Linux의 릴리스 2에서 mmigrate 스크립트를 사용하여 업그레이드한 경우 제외
Portal Server	패키지 바꾸기: psupgrade 스크립트 사용	psupgrade 스크립트를 사용하여 웹 컨테이너에 대한 재구성 및 재배포 수행
Portal Server Secure Remote Access	패키지 바꾸기: psupgrade 스크립트 사용	psupgrade 스크립트를 사용하여 재구성
Service Registry	이전 패키지를 바꾸지 않고 새로 설치 수행	수동으로 재구성하고 ant upgrade 스크립트를 사용하여 Application Server 도메인에 재배포
Sun Cluster	패키지 바꾸기: scinstall 스크립트를 사용하여 바이너리 바꾸기	scinstall 스크립트를 사용하여 구성 마이그레이션
Sun Cluster Geo	패키지 바꾸기: uninstall 스크립트를 사용하고 새로 설치 수행	없음
Web Proxy Server	바이너리 패치	없음
Web Server	이전 패키지를 바꾸지 않고 새로 설치 수행	wadm migrate-server 명령을 사용하여 서버 인스턴스 구성 마이그레이션

공유 구성 요소 업그레이드

Java ES 공유 구성 요소는 해당 공유 구성 요소에 종속되는 제품 구성 요소의 업그레이드에 따라 업그레이드해야 합니다.

공유 구성 요소를 업그레이드할 때는 구성 요소 재구성이나 사전 또는 사후 업그레이드 절차가 필요하지 않습니다. 또한 공유 구성 요소 업그레이드는 이전 버전으로 롤백할 수 없습니다.

많은 Java ES 공유 구성 요소의 개수(약 30개)와 공유 구성 요소 및 제품 구성 요소 간의 복잡한 상호 작용 때문에 단일 운영 체제 인스턴스 내의 모든 공유 구성 요소를 동일한 Java ES 릴리스 버전으로 동기화해야 합니다. 운영 체제 인스턴스는 Solaris 9, Solaris 10 또는 Red Hat Enterprise Linux 운영 체제를 실행하는 단일 컴퓨터 또는 Solaris 10 운영 체제를 실행하는 컴퓨터의 가상 운영 체제 환경(영역)을 의미합니다.

동기화 요구 사항 때문에 Java ES 공유 구성 요소를 하나씩 업그레이드해서는 안 되며 공유 구성 요소를 한꺼번에 해당 릴리스 5 버전으로 업그레이드해야 합니다.

공유 구성 요소를 릴리스 5와 동기화하는 작업은 Java ES 설치 프로그램을 사용하여 수행됩니다. 설치 프로그램은 제품 구성 요소 업그레이드를 수행할 때 (32페이지의 "Java ES 설치 프로그램의 업그레이드 기능 사용" 참조) 또는 제품 구성 요소를 새로 설치할 때 공유 구성 요소를 동기화합니다. 또한 설치 프로그램에는 기존 공유 구성 요소를 업그레이드하고 누락된 공유 구성 요소를 설치하는 동기화 기능이 포함되어 있습니다. 이 기능에 대한 자세한 내용은 65페이지의 "모든 공유 구성 요소 동기화"를 참조하십시오.

업그레이드 프로세스

Java ES 업그레이드 프로세스에는 프로덕션 환경에서 실행되기 전에 스테이징 환경에서 먼저 수행되는 여러 단계가 포함됩니다. 스테이징 환경을 사용하면 각 단계를 테스트하는 것은 물론 IT 담당자가 복잡한 Java ES 배포를 업그레이드하는 데 사용하는 스크립트를 작성할 수 있습니다.

스테이징 환경에서 업그레이드 프로세스를 테스트하고 업그레이드가 올바르게 작동하는 것을 확인했으면 프로덕션 환경에서 해당 프로세스를 수행할 수 있습니다.

이런 프로세스에는 다음 표에 나와 있는 단계가 포함되며, 이러한 단계는 업그레이드 설명서에서 설명합니다. 각 단계는 개별 구성 요소 업그레이드는 물론 배포된 Java ES 전체에도 적용됩니다.

표 1-4 업그레이드 프로세스의 단계

업그레이드 단계	설명
계획	업그레이드 계획을 수립합니다. 이 단계에서는 업그레이드할 Java ES 구성 요소를 지정하고 배포 시 다른 컴퓨터 또는 운영 체제 인스턴스에서 이러한 구성 요소의 업그레이드 순서를 지정합니다.
사전 업그레이드 준비	구성 및 응용 프로그램 데이터를 백업하고 운영 체제 패치를 수행하며 필요한 종속성을 업그레이드한 다음 개별 구성 요소 업그레이드를 준비하는 데 필요한 다른 작업을 수행합니다.
업그레이드	업그레이드에 필요한 모든 패키지, 패치 및 도구를 확보합니다. 업그레이드된 소프트웨어를 설치하고 지정된 대로 업그레이드된 시스템으로 데이터 마이그레이션을 포함하여 각 구성 요소를 재구성합니다.
확인	업그레이드된 소프트웨어 구성 요소 시작 및 다양한 사용 시나리오 테스트를 포함하는 지정된 검증 테스트를 사용하여 업그레이드가 성공적으로 수행되었는지 확인합니다.
사후 업그레이드 절차	새 기능을 통합하는 등과 같이 작업을 업그레이드된 구성 요소가 작동 되도록 하기 위해 필요한 추가 구성, 사용자 정의 또는 다른 작업을 수행합니다.
롤백/복원	업그레이드를 롤백하고 롤백이 성공했는지 확인합니다. 어떤 이유로 프로덕션 환경을 이전 환경으로 복원해야 하는 경우 업그레이드 롤백을 테스트하는 것이 중요합니다.

업그레이드 계획 시 고려 사항

업그레이드 계획에서는 릴리스 5로 업그레이드할 Java ES 구성 요소를 지정하고 Java ES 배포 시 여러 컴퓨터 또는 운영 체제 인스턴스에서 이러한 구성 요소를 업그레이드하는 순서를 지정합니다.

이런 계획은 업그레이드 목적과 우선 순위와 더불어 배포 구조의 범위와 복잡성에 따라 달라집니다.

예를 들어 Java ES 배포 구조가 단일 컴퓨터에서 실행되는 단일 Java ES 구성 요소로 이루어져 있고 업그레이드 목적은 이전 소프트웨어 릴리스의 몇 가지 버그를 수정하는 것일 수 있습니다. 반면 Java ES 배포 구조가 여러 개의 상호 종속적인 Java ES 구성 요소들이 여러 대의 서로 다른 컴퓨터에 걸쳐 배포된 형태로 이루어져 있고, 업그레이드 목적이 일부 새로운 기능을 구현하기 위해 필요한 최소한의 구성 요소를 최소한의 중단 시간으로 업그레이드하는 것일 수 있습니다.

일반적으로 배포 구조 내에 Java ES 구성 요소와 컴퓨터의 수가 많을수록 업그레이드 계획도 복잡해집니다.

하지만 업그레이드 계획은 배포 구조의 범위 및 복잡성 외에도 여러 고려 사항에 따라 달라집니다. 이러한 고려 사항에는 다음과 같은 요인이 포함됩니다.

- 종속성 업그레이드
- 지원되는 업그레이드 경로 및 전략
- 선택적 업그레이드 또는 전체 업그레이드
- 여러 인스턴스 업그레이드
- 운영 체제 고려 사항

종속성 업그레이드

모든 Java ES 제품 구성 요소의 업그레이드 계획에서 중요한 사항 중 하나는 특정 구성 요소의 다른 Java ES 구성 요소에 대한 종속성, 그리고 종속되는 구성 요소의 업그레이드를 지원하기 위해 다른 구성 요소도 업그레이드해야 하는지 여부를 파악하는 것입니다.

업그레이드 종속성에는 두 가지 유형이 있습니다.

- **엄격한 업그레이드 종속성.** 제품 구성 요소를 업그레이드하려면 종속되는 구성 요소를 업그레이드해야 합니다. 이러한 요구 사항은 종속되는 구성 요소에서 필요로 하는 새로운 기능, 새로운 인터페이스 또는 버그 수정으로 인해 발생합니다. 엄격한 업그레이드 종속성이 있는 경우 구성 요소가 종속되는 다른 구성 요소를 먼저 업그레이드해야 해당 구성 요소를 성공적으로 업그레이드하고 사용할 수 있습니다.
- **엄격하지 않은 업그레이드 종속성.** 제품 구성 요소를 업그레이드해도 종속되는 구성 요소를 업그레이드할 필요는 없습니다. 엄격하지 않은 업그레이드 종속성이 있는 경우 구성 요소가 종속되는 다른 구성 요소를 업그레이드하지 않고도 해당 구성 요소를 성공적으로 업그레이드하고 사용할 수 있습니다.

Java ES 구성 요소를 업그레이드하는 경우 해당 구성 요소가 *엄격한* 업그레이드 종속성을 갖는 모든 구성 요소를 업그레이드해야 하지만 *엄격하지 않은* 업그레이드 종속성을 갖는 구성 요소는 여기서 설명하는 몇 가지 예외 상황을 제외하고는 업그레이드하지 않아도 됩니다. 업그레이드에 여러 상호 종속 구성 요소가 포함되어 있는 경우 업그레이드하는 Java ES 구성 요소 중 하나라도 이러한 특정 구성 요소에 대해 엄격한 업그레이드 종속성이 있으면 해당 구성 요소를 업그레이드해야 합니다.

몇몇 특별한 경우 호환성 문제 때문에 구성 요소를 업그레이드하려면 이 구성 요소가 지원하는 구성 요소도 업그레이드해야 합니다. 이런 특별한 경우에 대해서는 이 설명서에 설명되어 있습니다.

지원되는 업그레이드 경로 및 전략

업그레이드 계획은 릴리스 5로 업그레이드하려는 Java ES 릴리스에 따라 다릅니다.

모든 이전 Java ES 소프트웨어 릴리스를 Java ES 5(릴리스 5)로 업그레이드할 수 있지만 지원되는 업그레이드는 Java ES 2005Q4(릴리스 4), Java ES 2005Q1(릴리스 3) 및 Java ES 2004Q2(릴리스 2)에서 수행한 업그레이드뿐입니다. 이 업그레이드 설명서에서는 Java ES 2003Q4(릴리스 1) 및 Java ES 이전 릴리스에서 업그레이드하는 전략을 제공하지만 이러한 업그레이드를 수행하는 절차에 대해서는 설명하지 않습니다.

다음 표에는 릴리스 5로 업그레이드하는 여러 가지 경로, 해당 특성 및 업그레이드를 수행하는 데 사용할 업그레이드 전략이 나와 있습니다.

이 표에서 설명하는 업그레이드 방법에 차이가 있고 제품 구성 요소 업그레이드 절차는 업그레이드 중인 릴리스에 따라 달라질 수 있기 때문에 업그레이드 설명서에서 각 제품 구성 요소의 업그레이드에 대해 설명하는 장은 서로 다른 업그레이드 방법을 설명하는 절로 구분되어 있습니다.

표 1-5 Java ES 5(릴리스 5)로 업그레이드하는 방법

제품 버전	Java ES 릴리스	시스템 특징	업그레이드 전략
2005Q4	릴리스 4	Java ES 5(릴리스 5)는 단일 컴퓨터에서 릴리스 4 및 릴리스 5 제품 구성 요소를 혼합하여 사용하도록 지원하지만 공유 구성 요소를 동일한 릴리스로 동기화해야 합니다. 릴리스 4와 릴리스 5 제품 구성 요소 간의 호환성은 테스트되었으며 알려진 인터페이스 비호환성 문제는 Java Enterprise System 5 UNIX용 릴리스 노트(http://docs.sun.com/doc/820-0454)에 기록되어 있습니다.	릴리스 4와 릴리스 5 제품 구성 요소가 공존 가능하기 때문에 단일 컴퓨터 또는 여러 컴퓨터로 이루어진 배포 구조 내에서 릴리스 4 제품 구성 요소를 선택적으로 릴리스 5로 업그레이드하는 것이 가능합니다. 릴리스 5 제품 구성 요소에서 릴리스 5 공유 구성 요소 지원이 필요한 경우 해당 컴퓨터의 모든 공유 구성 요소를 릴리스 5로 동기화해야 합니다.
2005Q1	릴리스 3	위의 릴리스 4 업그레이드 방법과 비슷합니다. Java ES 5(릴리스 5)는 단일 컴퓨터에서 릴리스 3(릴리스 4 포함) 및 릴리스 5 제품 구성 요소를 혼합하여 사용하도록 지원하지만 공유 구성 요소를 동일한 릴리스로 동기화해야 합니다. 릴리스 3과 릴리스 5 구성 요소 간의 호환성은 테스트되었으며 알려진 인터페이스 비호환성 문제는 Java Enterprise System 5 UNIX용 릴리스 노트(http://docs.sun.com/doc/820-0454)에 기록되어 있습니다.	위의 릴리스 4 업그레이드 방법과 비슷합니다. 릴리스 3과 릴리스 5 구성 요소가 공존 가능하기 때문에 단일 컴퓨터 또는 여러 컴퓨터로 이루어진 배포 구조 내에서 릴리스 3 구성 요소를 선택적으로 릴리스 5로 업그레이드하는 것이 가능합니다. 릴리스 5 제품 구성 요소에서 릴리스 5 공유 구성 요소 지원이 필요한 경우 해당 컴퓨터의 모든 공유 구성 요소를 릴리스 5로 동기화해야 합니다.
2004Q2	릴리스 2	위의 릴리스 4 및 릴리스 3 업그레이드 방법과 대조적입니다. Java ES 5(릴리스 5)는 단일 컴퓨터에서 제품 구성 요소나 공유 구성 요소의 릴리스 2 및 릴리스 5 구성 요소를 혼합하여 사용하도록 지원하지 <i>않습니다</i> . 이 두 릴리스 버전 간에는 알려진 인터페이스 비호환성이 있으며 릴리스 2와 릴리스 5 구성 요소 간의 상호 운용성은 입증되지 않았습니니다(테스트되지 않음).	한 대의 컴퓨터에서 구성 요소를 릴리스 2에서 릴리스 5로 업그레이드하는 경우 모든 릴리스 2 구성 요소를 릴리스 5로 업그레이드해야 합니다. 그러나 여러 컴퓨터로 구성된 배포 구조에서 서로 다른 컴퓨터에 릴리스 2 및 릴리스 5 구성 요소가 상주할 수도 있습니다.

표 1-5 Java ES 5(릴리스 5)로 업그레이드하는 방법 (계속)

제품 버전	Java ES 릴리스	시스템 특징	업그레이드 전략
2003Q4 및 이전 버전	릴리스 1 및 Java ES 이전 버전	위의 릴리스 2 업그레이드 방법과 비슷합니다. Java ES 5(릴리스 5)는 단일 컴퓨터에서 제품 구성 요소나 공유 구성 요소의 릴리스 2 및 릴리스 5 구성 요소를 혼합하여 사용하도록 지원하지 않습니다. 이 두 릴리스 버전 간에는 알려진 인터페이스 비호환성이 있으며 릴리스 1 또는 이전 릴리스 및 릴리스 5 구성 요소 간의 상호 운용성은 인증되지 않습니다(테스트되지 않음).	Java ES는 릴리스 1 또는 그 이전 릴리스에서 릴리스 5로의 직접 업그레이드를 인증하지 않습니다. 그러나 릴리스 3 Java Enterprise System Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)에 설명된 대로 먼저 Java ES 릴리스 3으로 업그레이드한 다음 릴리스 3을 릴리스 5로 업그레이드하여 릴리스 1에서 업그레이드할 수 있는 경우도 있습니다. 이러한 경우 이 업그레이드 설명서의 해당 구성 요소 업그레이드 로드맵에서 이러한 가능성에 대해 설명합니다. 그 외의 경우 릴리스 1에서 릴리스 5로의 업그레이드는 릴리스 2 또는 릴리스 3에서 릴리스 5로 업그레이드하는 경우와 같은 방법으로 수행할 수 있으며 이 경우 업그레이드 설명서의 해당 구성 요소에 대한 업그레이드 로드맵에 이러한 가능성이 언급되어 있습니다.

주	공식 Java ES 릴리스 간에 제품 구성 요소의 임시 기능 릴리스(IFR)가 있는 경우 IFR 업그레이드는 일반적으로 이전 Java ES 릴리스와 동일한 절차를 사용하여 수행할 수 있습니다. 예를 들어 릴리스 3과 릴리스 4 사이에 IFR이 나온 경우 해당 구성 요소는 릴리스 3에서 릴리스 5로 업그레이드하는 절차를 사용하여 업그레이드할 수 있습니다. 그렇지 않은 경우(예: Portal Server 및 Portal Server Secure Remote Access)에는 본 업그레이드 설명서에서 IFR별 업그레이드 절차를 설명합니다.

선택적 업그레이드 또는 전체 업그레이드

엄격한 업그레이드 종속성과 엄격하지 않은 업그레이드 종속성 간의 차이를 이용하면 배포된 시스템 내에서 Java ES 제품 구성 요소를 선택적으로 업그레이드할 수 있습니다. 선택적 업그레이드는 단일 컴퓨터에서 릴리스 3 및 릴리스 4를 릴리스 5로 업그레이드하는 경우에 적용됩니다. 단일 컴퓨터에서 릴리스 2를 릴리스 5로 선택적 업그레이드하는 것은 지원되지 않습니다.

일반적으로 컴퓨터에서 Java ES 제품 구성 요소를 선택적으로 업그레이드하거나 전체 업그레이드할 수 있습니다.

- **선택적 업그레이드.** 이 방식은 릴리스 5로 업그레이드하려는 Java ES 제품 구성 요소부터 시작합니다. 해당 구성 요소에 대한 엄격한 업그레이드 종속성을 확인하여 해당 구성 요소를 업그레이드해야 하는지를 결정합니다. 더 이상 업그레이드할 구성 요소가 없을 때까지 다음에 이어지는 엄격한 업그레이드 종속성에 대해 이 프로세스를 반복합니다. 이 과정을 통해 업그레이드해야 하는 모든 Java ES 제품 구성 요소를 지정할 수 있습니다.
- **전체 업그레이드.** 이 방식은 배포된 모든 Java ES 제품 구성 요소를 릴리스 5로 업그레이드합니다. 배포의 복잡성 때문에 업무상 이유로 전체 시스템을 한 번에 업그레이드할 수 없는 경우도 있습니다.

다음 표에서 업그레이드를 수행하는 두 가지 방법을 비교합니다.

표 1-6 선택적 업그레이드와 전체 업그레이드 비교

업그레이드 방법	이점	단점
선택적 업그레이드	업그레이드할 구성 요소의 수를 최소화	배포된 시스템의 모든 구성 요소 간에 버전이 일관되지 않음
전체 업그레이드	배포된 시스템의 모든 구성 요소를 일관된 버전으로 유지	업그레이드하는 구성 요소의 수가 최대화됨

선택적 업그레이드는 Java ES 릴리스 4에서도 지원됩니다. 따라서 한 컴퓨터에 릴리스 3 제품 구성 요소와 릴리스 4 제품 구성 요소가 함께 있는 경우 두 구성 요소를 릴리스 5로 선택적 업그레이드하는 것이 가능합니다.

여러 인스턴스 업그레이드

업그레이드 계획에서 업그레이드 절차의 순서는 배포 구조에서 중복 기능이 사용되는 방식에 따라 달라집니다. Java ES 구성 요소의 여러 인스턴스가고가용성, 확장성, 서비스 가능성 또는 이러한 서비스 품질의 조합을 위해 사용될 수 있습니다. Java ES 배포 구조에서 중복 구성 요소는 로드 균형 조정(Directory Proxy Server, Web Server, Web Proxy Server, Application Server, Access Manager 및 Portal Server),고가용성 기술(Sun Cluster 및고가용성 세션 저장소) 및 Directory Server 복제의 세 가지 기술에서 사용됩니다.

중복이 사용되는 대부분의 경우 시스템 중단 시간이 생기지 않도록 업그레이드를 수행해야 합니다. 이러한 롤링 업그레이드에서는 구성 요소의 중복 인스턴스를 이들이 제공하는 서비스에 손상을 주지 않으면서 연속적으로 업그레이드하려고 시도합니다.

중복 인스턴스는 일반적으로 여러 컴퓨터에 걸쳐 배포됩니다. 업그레이드 계획에서 시스템 중단 시간을 최소화하기 위해 복제된 구성 요소 업그레이드를 다른 구성 요소 업그레이드로부터 분리해야 할 수 있습니다. 롤링 업그레이드를 수행하기 전에 각 컴퓨터에서 복제된 구성 요소를 위한 모든 사전 업그레이드 작업을 수행합니다.

각 복제 기술에는 전체 Java ES 구성 요소 업그레이드 순서에 영향을 주는 구성 또는 재구성 절차가 있습니다. 예를 들어, Sun Cluster 환경에서 실행되는 구성 요소는 Sun Cluster 환경에서 실행 중인 구성 요소를 업그레이드하기 전에 Sun Cluster의 업그레이드를 요구할 수 있습니다.

본 업그레이드 설명서에서 각 제품 구성 요소 업그레이드를 설명하는 장에서는 해당 구성 요소에 대해 여러 인스턴스 업그레이드를 수행하는 방법을 설명합니다.

운영 체제 고려 사항

아래에 설명된 대로 Java ES 업그레이드 계획에 영향을 주는 몇 가지 운영 체제 고려 사항이 있습니다.

필수 운영 체제 패치

Java ES 제품 구성 요소를 성공적으로 업그레이드하려면 먼저 운영 체제에서 패치를 수행하거나 Java ES 5 제품 구성 요소에 필요한 수준까지 운영 체제를 업데이트해야 할 수 있습니다. 그러나 각각의 경우에 필요한 특정 패치나 수정을 적용하는 것보다 특정 제품 구성 요소를 업그레이드하기 전에 Java ES 5에 필요한 수준으로 운영 체제를 업그레이드하는 것이 좋습니다.

- Solaris 플랫폼.** SunSolve 웹 사이트에서 패치 클러스터 형태로 운영 체제 패치를 다운로드할 수 있습니다. 패치 클러스터는 한꺼번에 적용할 수 있는 운영 체제 패치 모음입니다. Solaris 9 및 10용 Java ES 릴리스 5를 지원하는 데 필요한 패치 클러스터는 <http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 다운로드할 수 있습니다.
- Linux 플랫폼.** <https://www.redhat.com/apps/download/>에서 업데이트 릴리스를 다운로드할 수 있습니다. 그러나 Java ES 업그레이드를 수행하기 전에 Linux 운영 체제를 업그레이드할 필요는 없습니다.

이중 업그레이드: Java ES 및 운영 체제 소프트웨어

Java ES 소프트웨어 또는 운영 체제 소프트웨어를 지원되지 않는 버전으로 업그레이드하는 경우 운영 체제 및 Java ES 소프트웨어가 일치하지 않을 수 있습니다. 관련 지원 매트릭스는 다음 표에 나와 있습니다.

표 1-7 Java ES/운영 체제 지원 매트릭스

Java ES 릴리스	Solaris			RHEL		
	8	9	10	2.1	3.0	4.0
2003Q4(릴리스 1)	X	X		X		
2004Q2(릴리스 2)	X	X		X		
2005Q1(릴리스 3)	X	X	X	X	X	
2005Q5(릴리스 4)	X	X	X	X	X	
5(릴리스 5)		X	X		X	X

Java ES 소프트웨어 또는 운영 체제 소프트웨어를 업그레이드하여 지원되지 않는 구성이 발생하는 경우 Java ES와 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다. 다음 상황에서 이중 업그레이드가 필요할 수 있습니다.

- 설치된 Java ES 소프트웨어에서 지원되지 않는 버전으로 운영 체제를 업그레이드했습니다.

예를 들어, Java ES 2004Q2(릴리스 2)는 Solaris 8 및 9 운영 체제와 Red Hat Enterprise Linux(RHEL) 2.1에서 지원됩니다. Java ES 릴리스 2에서 지원되지 않는 Solaris 10 또는 RHEL 3.0으로 운영 체제 플랫폼을 업그레이드하려면 Java ES 릴리스 2도 업그레이드된 플랫폼을 지원하는 Java ES 릴리스 버전으로 업그레이드해야 합니다. 이 경우 Java ES 5(릴리스 5)로 업그레이드하는 것이 좋습니다.

- 기존 운영 체제 소프트웨어에서 지원하지 않는 버전으로 Java ES를 업그레이드했습니다.

예를 들어 Java ES 2005Q1(릴리스 3) 및 Java ES 2005Q4(릴리스 4)는 Solaris 8 및 RHEL 2.1에서 지원됩니다. Java ES를 Solaris 8 또는 RHEL 2.1에서 지원되지 않는 릴리스 5로 업그레이드하려면 운영 체제를 Java ES 5(릴리스 5)에서 지원되는 버전으로 업그레이드해야 합니다. 이 경우 Solaris 10 또는 RHEL 4.0으로 업그레이드하는 것이 좋습니다.

일반적으로 이중 업그레이드를 수행하는 방법은 두 가지가 있습니다.

- **운영 체제 새로 설치.** 새 운영 체제를 설치한 후 Java ES 릴리스 5를 새로 설치합니다. 여기에는 구성 데이터, 런타임 데이터, 사용자 정의 등과 같은 이전 버전의 제품 구성 요소 데이터에 대한 마이그레이션이 포함됩니다. 운영 체제는 새 시스템(또는 Solaris 10 영역)에 설치하거나 기존 파일 시스템을 덮어쓸 수 있습니다. 후자의 경우 먼저 구성 요소 데이터를 백업하고, 운영 체제를 설치한 후 백업한 데이터를 복원해야 합니다.
- **기존 운영 체제 인플레이스 업그레이드.** 기존 파일 시스템을 그대로 두고 운영 체제 업그레이드를 수행한 다음 Java ES 제품 구성 요소를 릴리스 5로 업그레이드합니다. 이 방법을 사용하려면 운영 체제 업그레이드가 이미 설치되어 있는 Java ES 제품 구성 요소, 해당 데이터 및 필요한 공유 구성 요소의 업그레이드에 영향을 주지 않아야 합니다.

이중 업그레이드가 지원되지 않는 Java ES 제품 구성 요소가 있는 경우 즉, 앞에서 설명한 방법을 모두 사용할 수 없는 경우에는 운영 체제 설치 또는 업그레이드를 수행한 후 해당 구성 요소를 다시 설치하고 새로 구성해야 합니다.

다음 표에는 각 Java ES 제품 구성 요소에서 지원되는 이중 업그레이드 방법이 정리되어 있습니다.

표 1-8 Java ES 5 제품 구성 요소 이중 업그레이드 지원

제품 구성 요소	운영 체제 새로 설치	기존 운영 체제 업그레이드
Access Manager	지원되지 않음	지원되지 않음
Application Server	같은 컴퓨터에서만 지원	지원
Directory Proxy Server	지원	지원
Directory Server	지원	지원
고가용성 세션 저장소	Application Server 이중 업그레이드와 연결하여 수행	Application Server 이중 업그레이드와 연결하여 수행
Java DB	지원	지원
Message Queue	지원	지원
Portal Server	지원되지 않음	지원
Portal Server Secure Remote Access	지원되지 않음	지원
Service Registry	지원	지원
Sun Cluster	지원되지 않음	지원

표 1-8 Java ES 5 제품 구성 요소 이중 업그레이드 지원 (계속)

제품 구성 요소	운영 체제 새로 설치	기존 운영 체제 업그레이드
Sun Cluster Geographic Edition	지원되지 않음	Sun Cluster 이중 업그레이드와 연결하여 수행
Web Proxy Server	지원되지 않음	지원
Web Server	지원되지 않음	지원

운영 체제 업그레이드

Solaris 운영 체제를 업그레이드하면 기존 Java ES 공유 구성 요소를 이전 버전으로 덮어 쓰는 경우가 있습니다. 이러한 경우 Solaris 운영 체제와 함께 번들로 제공되는 Message Queue를 릴리스 5로 업그레이드하여 올바른 Java ES 버전으로 복원할 수 있습니다. Message Queue를 업그레이드하면 모든 기존 공유 구성 요소도 업그레이드됩니다.

Solaris 10 다중 영역 환경

다중 영역 환경에서 Java ES 구성 요소를 설치 및 업그레이드하는 경우 여러 가지 문제가 있을 수 있습니다. Solaris 10 영역에 Java ES를 배포하는 데 따르는 장점 및 제한에 대한 내용과 다중 영역 환경에서 Java ES 구성 요소를 업그레이드하기 위해 권장되는 방법에 대한 자세한 내용은 [58페이지의 "Java ES 5 업그레이드 및 Solaris 10 영역"](#)을 참조하십시오.

Java ES 구성 요소 종속성

업그레이드 계획에서 가장 중요한 고려 사항 중 하나는 배포된 시스템 내의 다양한 Java ES 구성 요소들 간 종속성입니다. 구성 요소 업그레이드를 수행하는 순서는 구성 요소 간의 종속성에 따라 달라집니다.

이 절에서는 업그레이드 계획에 영향을 주는 Java ES 구성 요소 종속성에 대한 정보를 제공합니다.

- [공유 구성 요소에 대한 종속성](#)
- [제품 구성 요소에 대한 종속성](#)

공유 구성 요소에 대한 종속성

[47페이지 표 1-9](#)에서는 Java ES 5(릴리스 5) 제품 구성 요소가 Java ES 공유 구성 요소에 대해 갖는 종속성을 보여 줍니다. [표 1-9](#)의 머리글 옆에 있는 제품 구성 요소 약어는 [28페이지 표 1-1](#)에서 가져온 것입니다. 공유 구성 요소에 대한 약어는 [29페이지 표 1-2](#)에 정리되어 있습니다.

[표 1-9](#)의 행렬에서 릴리스 3 및 릴리스 4에서 릴리스 5로의 엄격한 업그레이드 종속성은 "H"로 표시되어 있으며 엄격하지 않은 업그레이드 종속성은 "S"로 표시되어 있습니다. 릴리스 2에서 릴리스 5로 업그레이드하는 경우 모든 공유 구성 요소 종속성은 엄격한 업그레이드 종속성으로 정의되어 있으므로 모든 공유 구성 요소를 릴리스 2에서 릴리스 5로 업그레이드해야 합니다.

표 1-9 Java ES 5(릴리스 5) 제품 구성 요소의 공유 구성 요소 종속성

공유 구성 요소	AM	AS	DPS	DS	DS 콘솔	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			
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

표 1-9 Java ES 5(릴리스 5) 제품 구성 요소의 공유 구성 요소 종속성 (계속)

공유 구성 요소	AM	AS	DPS	DS	DS 콘솔	HADB	JavaDB	MQ	MC	PS	PSRA	SC	SCG	SR	WPS	WS
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. 이 종속성은 특히 CAC(Common Agent Container) 버전 1.1에 적용됩니다.

제품 구성 요소에 대해 표 1-9에 나와 있는 종속성은 직접 및 간접 공유 구성 요소 종속성을 모두 나타냅니다. 제품 구성 요소는 특정 공유 구성 요소에 종속(직접 종속성)되거나, 하나 이상의 다른 공유 구성 요소에 종속된 공유 구성 요소에 종속(간접 종속성)될 수 있습니다. 49페이지 그림 1-1에서 공유 구성 요소 간의 상호 종속성을 보여 줍니다.

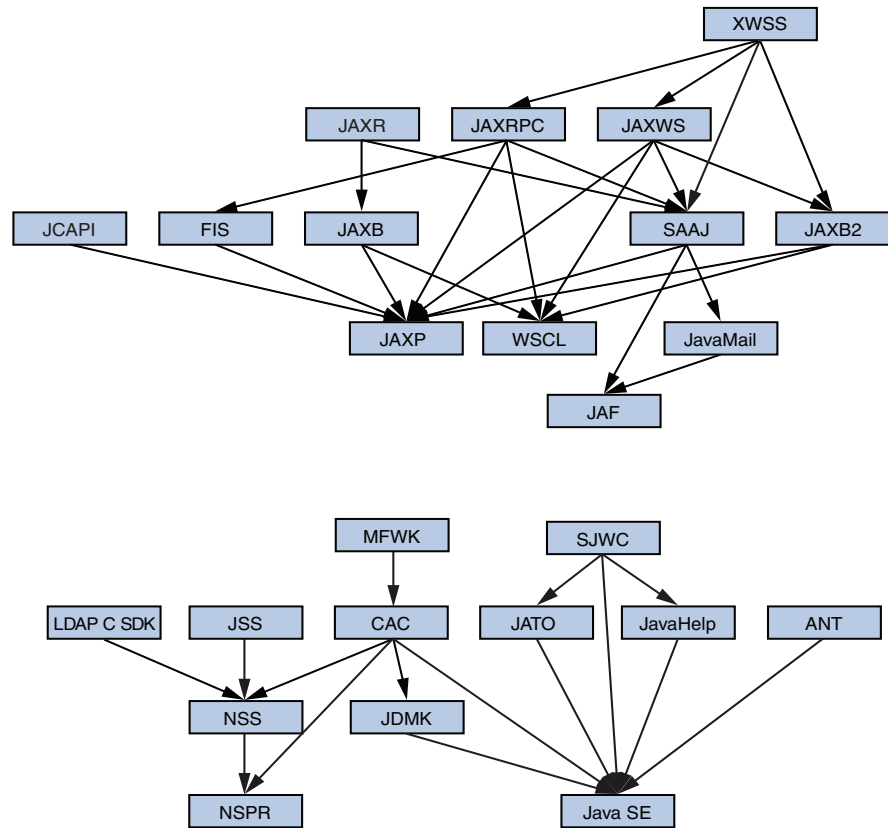
표 1-9에서는 지정된 컴퓨터에서 하나 이상의 제품 구성 요소를 업그레이드하는 경우 해당 공유 구성 요소를 보여 줍니다.

그러나 공유 구성 요소를 동기화해야 하므로(35페이지의 "공유 구성 요소 업그레이드" 참조) Java ES 공유 구성 요소를 하나씩 업그레이드할 수 없으며 컴퓨터 또는 운영 체제 인스턴스에서 모든 공유 구성 요소를 동시에 릴리스 5 버전으로 업그레이드해야 합니다.

엄격한 업그레이드 종속성이 없는 경우에는 공유 구성 요소를 업그레이드하지 않아도 됩니다. 그러나 최신 버전을 기준으로 기본 Java ES 공유 구성 요소를 업그레이드하는 것이 좋습니다. 사실상 Java ES 설치 프로그램에서 제품 구성 요소를 설치 또는 업그레이드할 때는 호스트 컴퓨터에 있는 모든 공유 구성 요소가 릴리스 5로 자동으로 동기화됩니다.

공유 구성 요소를 수동으로 업그레이드 방법에 대한 자세한 내용은 2장, "Java ES 공유 구성 요소 업그레이드"를 참조하십시오.

그림 1-1 공유 구성 요소 상호 종속성



제품 구성 요소에 대한 종속성

제품 구성 요소에 대한 종속성은 일반적으로 런타임 종속성과 구성 종속성의 두 가지 범주로 분류됩니다.

- **런타임 종속성.** 소프트웨어 시스템의 기능은 해당 소프트웨어 시스템의 배포된 구성 요소 간 상호 작용을 바탕으로 합니다. Java ES 제품 구성 요소 간 인프라 종속성은 Java Enterprise System 5 기술 개요에 설명되어 있습니다. 릴리스 5 제품 구성 요소에 다른 제품 구성 요소에 대한 엄격한 업그레이드 종속성이 있는 경우 다른 제품 구성 요소를 먼저 업그레이드한 다음에 해당 구성 요소를 업그레이드 하고 사용할 수 있습니다.
- **구성 종속성.** 다른 구성 요소를 구성하기 위해 특정 Java ES 구성 요소가 설치 및 구성되어 실행 중이어야 하는 경우가 있습니다. 예를 들어 Access Manager 서비스를 등록하려면 Directory Server 사용자/그룹 디렉토리가 실행되고 있어야 합니다. 구성 요소 업그레이드 절차에는 업그레이드된 구성 요소의 재구성 또는 구성 데이터의 마이그레이션까지 포함되는 경우가 많습니다. 구성 종속성은 업그레이드 절차의 순서에 영향을 줄 수 있습니다.

런타임 종속성에서 제품 구성 요소 간의 관계는 다음 세 가지 유형 중 하나일 수 있습니다.

- **필수 종속성.** 지원 구성 요소가 없으면 구성 요소가 작동할 수 없습니다.
- **선택적 종속성.** 지원 구성 요소가 없어도 구성 요소가 작동할 수 있지만 해당 기능 중 일부를 수행하려면 지원 구성 요소가 있어야 합니다.
- **상호 종속성.** 상호 간의 지원이 없어도 두 구성 요소가 작동할 수 있지만 구성 요소를 함께 사용하는 경우 특정 기능 또는 성능이 향상됩니다.

다음 표에서는 [28페이지 표 1-1](#)에 나와 있는 Java ES 제품 구성 요소 간의 종속성 및 종속 관계를 보여줍니다. 이 정보는 업그레이드 계획에 영향을 주는 엄격한 업그레이드 종속성을 판단하는 데 사용할 수 있습니다.

첫 번째 열은 릴리스 5 제품 구성 요소를 영문자순으로 나열하며 두 번째 열은 릴리스 5 구성 요소와 종속 관계가 있는 다른 Java ES 구성 요소를 보여 줍니다. 세 번째 열은 릴리스 5 종속성을 지원하는 Java ES 릴리스 버전을 보여 주며 네 번째 열은 종속 관계가 정리되어 있습니다. 마지막 열은 지원 구성 요소가 원격이 아닌 로컬이어야 하는지 또는 다른 타사 제품이 종속성을 지원할 수 있는지와 같은 종속성의 특별한 특성을 나타냅니다.

릴리스 5로 업그레이드하려는 제품 구성 요소에 이전 릴리스와 반대로 지원 구성 요소의 릴리스 5에 대한 종속성이 있는 경우 해당 지원 구성 요소에 엄격한 업그레이드 종속성이 있는 것이며 지원 구성 요소도 릴리스 5로 업그레이드해야 합니다.

표 1-10 Java ES 제품 구성 요소 종속성

릴리스 5 제품 구성 요소	종속성 ¹	Java ES 릴리스	종속성 특성	특성
Access Manager	Directory Server	2-5	필수 종속성: 구성 데이터 저장 및 사용자 데이터 조회 활성화	
	J2EE 웹 컨테이너: - Application Server	4-5	필수 종속성: 웹 컨테이너 런타임 서비스 제공	로컬 전용
	- Web Server	4-5		다음 항목도 지원합니다. - Weblogic ² - WebSphere ³
Access Manager SDK	Access Manager	3-5	필수 종속성: Access Manager 서비스 제공	
Access Manager 분산 인증	Access Manager	4-5	필수 종속성: Access Manager 서비스 제공	
	J2EE 웹 컨테이너: - Application Server	4-5	필수 종속성: 웹 컨테이너 런타임 서비스 제공	로컬 전용
	- Web Server	4-5		다음 항목도 지원합니다. - Weblogic ² - WebSphere ³
Access Manager 세션 페일오버	Access Manager	5	필수 종속성: Access Manager 서비스 제공	
	Message Queue	4-5	필수 종속성: 신뢰할 수 있는 비동기 메시징 제공	
Application Server	Message Queue	3-5	필수 종속성: 신뢰할 수 있는 비동기 메시징 제공	로컬 전용
	고가용성 세션 저장소 (HADB)	5	필수 종속성: 인스턴스 간 페일오버 지원에 필요한 세션 상태 저장	로컬 전용
	Java DB	5	필수 종속성: 기본 개발자 데이터베이스 및 기타 지속성 저장소 제공	로컬 전용
	Web Server	3-5	선택적 종속성: 인스턴스 간 로드 균형 조정 제공	로컬 전용
Directory Proxy Server	Directory Server	1-5	상호 종속성: 디렉토리 요청에 대한 보안 및 성능이 향상됨. Directory Proxy Server에 데이터 제공	

표 1-10 Java ES 제품 구성 요소 종속성 (계속)

릴리스 5 제품 구성 요소	종속성 ¹	Java ES 릴리스	종속성 특성	특성
Directory Server	Directory Proxy Server	1-5	상호 종속성: 디렉토리 요청에 대한 보안 및 성능이 향상됨. Directory Server의 부하 및 캐시 데이터 분산	
고가용성 세션 저장소 (HADB)	없음			
Java DB	없음			
Message Queue	Directory Server	2-5	선택적 종속성: 관리 객체 및 사용자 데이터 저장	
	J2EE 웹 컨테이너: - Application Server	2-5	선택적 종속성: 클라이언트 및 Message Queue 브로커 간 HTTP 전송 지원	
	- Web Server	2-5		
	Java DB	5	선택적 종속성: 지속성 메시지 저장	로컬 전용
	Sun Cluster	2-5	선택적 종속성: 고가용성 지원	
Monitoring Console	없음			
Portal Server	Directory Server	4-5	필수 종속성: 사용자 프로필 저장 및 조회 활성화	
	J2EE 웹 컨테이너: - Application Server	4-5	필수 종속성: 웹 컨테이너 런타임 서비스 제공	로컬 전용
	- Web Server	4-5		
	Access Manager 또는 Access Manager SDK	4-5	필수 종속성: 인증 및 권한 부여 서비스, 단일 사인온 제공	로컬 전용 (Access Manager가 원격인 경우 Access Manager SDK는 로컬로 사용되어야 함)
	Portal Server Secure Remote Access	5	선택적 종속성: Gateway, Rewriter 프록시 및 Netlet 프록시 구성 요소를 통한 보안 원격 액세스 제공	
	Service Registry 클라이언트	5	필수 종속성: 컴파일에 필요한 라이브러리 제공	
	Java DB	5	필수 종속성: 여러 포틀릿 응용 프로그램을 위한 지원 제공	

표 1-10 Java ES 제품 구성 요소 종속성 (계속)

릴리스 5 제품 구성 요소	종속성 ¹	Java ES 릴리스	종속성 특성	특성
Portal Server Secure Remote Access Gateway	Portal Server	5	필수 종속성: 게이트웨이 기능 지원	
	Access Manager 또는 Access Manager SDK	4-5	필수 종속성: 인증 및 권한 부여 서비스, 단일 사인온 제공	로컬 전용 (Access Manager가 원격인 경우 Access Manager SDK는 로컬로 사용되어야 함)
	Directory Server	4-5	필수 종속성: 사용자 데이터 저장 및 조회 활성화	
Rewriter 프록시	Portal Server	5	필수 종속성: Rewriter 프록시 기능 지원	
Netlet 프록시	Portal Server	5	필수 종속성: Netlet 프록시 기능 지원	
Service Registry 배포	Application Server	5	필수 종속성: 컨테이너 런타임 서비스 제공	로컬 전용
	Java DB	5	필수 종속성: 서비스 및 관련 메타 데이터 저장을 위한 기본 데이터베이스 제공	로컬 전용
	Service Registry 클라이언트	5	필수 종속성: 필수 클라이언트 라이브러리 제공	로컬 전용
클라이언트	없음			
Sun Cluster	없음			
Sun Cluster 에이전트	Sun Cluster	4-5	필수 종속성: Sun Cluster 서비스에 대한 액세스 제공	로컬 전용
Sun Cluster Geographic Edition	Sun Cluster	4-5	필수 종속성: Sun Cluster Geographic Edition 기능 지원	로컬 전용
Web Proxy Server	Directory Server	2-5	선택적 종속성: LDAP 기반 인증 제공	
	Web Server	2-5	상호 종속성: HTTP 요청에 대한 보안 및 성능이 향상됨. Web Proxy Server에 데이터 제공	다음 항목도 지원합니다. - Weblogic ² - WebSphere ³

표 1-10 Java ES 제품 구성 요소 종속성 (계속)

릴리스 5 제품 구성 요소	종속성 ¹	Java ES 릴리스	종속성 특성	특성
Web Server	Directory Server	1-5	선택적 종속성: LDAP 기 반 인증 제공	
	Web Proxy Server	1-5	상호 종속성: HTTP 요청 에 대한 보안 및 성능이 향 상됨. Web Server의 부하 및 캐시 데이터 분산	

1. 각 제품 구성 요소에서 종속성은 일반적으로 업그레이드해야 하는 순서로 나열됩니다.

2. BEA Weblogic Server

3. IBM WebSphere Application Server

업그레이드 순서 지침

선택적 업그레이드 또는 전체 업그레이드 간의 선택, 엄격한 업그레이드 종속성의 영향, 그리고 이전 절에서 설명한 다른 요소들은 업그레이드하려고 계획하는 Java ES 구성 요소뿐만 아니라 이러한 구성 요소를 업그레이드하는 순서에도 영향을 줄 수 있습니다. 그러나 모든 경우에 적용할 수는 없지만 몇 가지 일반적인 순서 지침을 적용할 수 있습니다.

다음 목록은 단일 컴퓨터 또는 배포된 시스템에서 Java ES 구성 요소를 성공적으로 업그레이드할 수 있는 순서를 제공합니다. 업그레이드를 계획할 때는 배포 구조에 포함되지 않는 구성 요소를 생략할 수 있으며 선택적 업그레이드를 수행하는 경우 엄격하지 않은 업그레이드 종속성이 있는 구성 요소를 생략할 수 있습니다.

본 업그레이드 설명서의 장은 다음 목록에 나와 있는 구성 요소 순서에 따라 정리되어 있습니다.

주	Java ES 구성 요소를 업그레이드하기 전에 운영 체제에 필요한 업데이트를 적용했는지 확인하십시오(42페이지의 "필수 운영 체제 패치" 참조). 또한 57페이지의 "특별 사례"를 검토하여 해당 업그레이드 시나리오에 적용되는 사항이 있는지 확인하십시오.

1. 공유 구성 요소 ([63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"](#) 참조)

공유 구성 요소는 이 구성 요소에 종속된 구성 요소보다 먼저 업그레이드해야 합니다. 대부분의 공유 구성 요소 업그레이드는 Java ES 설치 프로그램에서 처리하지만 Web Proxy Server 및 Portal Server의 경우에는 직접 공유 구성 요소를 업그레이드해야 합니다.

2. Sun Cluster 소프트웨어 ([75페이지의 3장, "Sun Cluster 소프트웨어"](#) 참조)

Sun Cluster 환경에서 실행되는 구성 요소가 있고 Sun Cluster 소프트웨어를 업그레이드해야 하는 경우 Sun Cluster 서비스를 사용하는 구성 요소보다 먼저 업그레이드해야 합니다. Sun Cluster 에이전트를 업그레이드하는 경우 Sun Cluster 업그레이드의 일부로 업그레이드되어야 합니다.

3. Sun Cluster Geographic Edition 소프트웨어 ([87페이지의 4장, "Sun Cluster Geographic Edition"](#) 참조)

Sun Cluster Geographic Edition은 종속되는 Sun Cluster 소프트웨어보다 나중에 업그레이드해야 하며 Sun Cluster 서비스를 사용하는 구성 요소 중에 가장 먼저 업그레이드해야 합니다.

4. Directory Server ([99페이지의 5장, "Directory Server"](#) 참조)

많은 구성 요소가 Directory Server에 사용자 데이터 또는 구성 데이터를 저장하므로 Directory Server 업그레이드는 Directory Server에 대해 런타임 또는 구성 종속성을 갖는 구성 요소의 업그레이드보다 일반적으로 먼저 수행해야 합니다.

5. Directory Proxy Server ([115페이지의 6장, "Directory Proxy Server"](#) 참조)

Directory Proxy Server는 Directory Server에 대한 엄격하지 않은 종속성이 있으며 언제든지 업그레이드할 수 있습니다. 일부 구성 요소는 Directory Proxy Server를 통해 Directory Server에 액세스할 수 있습니다. 따라서 Directory Proxy Server를 업그레이드하는 경우 Directory Server를 업그레이드한 직후에 업그레이드해야 합니다.

6. Web Server (129페이지의 7장, "Web Server" 참조)

여러 Java ES 구성 요소에서 웹 컨테이너가 지원되어야 하며, 이러한 웹 컨테이너는 웹 컨테이너 서비스가 필요한 구성 요소보다 먼저 업그레이드해야 합니다. 보통 웹 컨테이너 서비스는 Web Server 또는 Application Server에서 제공하지만 아키텍처에 이 두 가지가 모두 포함된다면 Application Server를 업그레이드하기 전에 Web Server를 먼저 업그레이드합니다.

7. Java DB (153페이지의 8장, "Java DB" 참조)

Java DB는 기본 데이터베이스로 Java DB를 사용해야 하는 Application Server보다 먼저 업그레이드해야 합니다. 하지만 Application Server를 업그레이드하면 Java ES 설치 프로그램이 자동으로 Java DB를 업그레이드합니다.

8. 고가용성 세션 저장소 (163페이지의 9장, "고가용성 세션 저장소" 참조)

고가용성 세션 저장소(HADB)는 고가용성을 위해 고가용성 세션 저장소가 필요한 Application Server보다 먼저 업그레이드해야 합니다. 하지만 Application Server를 업그레이드하면 Java ES 설치 프로그램이 자동으로 HADB를 업그레이드합니다.

9. Message Queue (175페이지의 10장, "Message Queue" 참조)

Message Queue는 Java Enterprise Edition (Java EE) 호환성을 제공하기 위해 Message Queue가 필요한 Application Server보다 먼저 업그레이드해야 합니다. 하지만 Application Server를 업그레이드하면 Java ES 설치 프로그램이 자동으로 Message Queue를 업그레이드합니다.

10. Application Server (199페이지의 11장, "Application Server" 참조)

Application Server는 Message Queue 및 고가용성 세션 저장소에 종속되므로 업그레이드하는 경우 이러한 구성 요소를 업그레이드한 다음에 업그레이드해야 합니다. 또한 Application Server는 로드 균형 조정 플러그인에 대해 Web Server에 종속될 수 있으므로 이 기능을 사용 중인 경우 Application Server는 Web Server 다음에 업그레이드해야 합니다.

11. Service Registry (227페이지의 12장, "Service Registry" 참조)

Service Registry가 런타임 컨테이너 서비스에 대해 Application Server에 종속되므로 Application Server를 업그레이드한 다음에 Service Registry를 업그레이드할 수 있습니다.

12. Web Proxy Server (239페이지의 13장, "Web Proxy Server" 참조)

Web Proxy Server는 언제든지 업그레이드할 수 있지만 프록시 서비스를 제공하는 Web Server 또는 Application Server 구성 요소 다음에 업그레이드하는 것이 일반적입니다. Web Proxy Server는 새로운 Java ES 릴리스 5 구성 요소로, 이전의 비Java ES 릴리스에서 업그레이드할 수 있습니다.

13. Access Manager (255페이지의 14장, "Access Manager"참조)

Access Manager는 단일 사인온을 포함한 인증 및 권한 부여에서 중추적 역할을 하며 이러한 서비스에 대해 Access Manager에 종속되는 구성 요소보다 먼저 업그레이드해야 합니다.

14. Portal Server (303페이지의 15장, "Portal Server" 참조)

Portal Server는 많은 선행 구성 요소(Directory Server, 웹 컨테이너 및 Access Manager)에 종속되며 이러한 구성 요소 다음에 업그레이드해야 합니다.

15. Portal Server Secure Remote Access (373페이지의 16장, "Portal Server Secure Remote Access" 참조)

Portal Server Secure Remote Access는 Portal Server를 업그레이드할 때 업그레이드해야 합니다.

특별 사례

Java ES 구성 요소를 릴리스 5로 업그레이드하려고 계획하는 경우 알아두어야 할 몇 가지 특별 사례가 있으며, 자세한 내용은 아래에 설명되어 있습니다.

선택적 업그레이드: Application Server 업그레이드하지 않음

릴리스 3 또는 릴리스 4 Application Server (8.1)를 실행하는 컴퓨터에서 Java ES 구성 요소를 Java ES 5로 선택적으로 업그레이드하는 경우 Application Server를 릴리스 5로 업그레이드하지 않으면 Application Server가 제대로 동작하도록 하기 위해 해결해야 할 몇 가지 상황이 있습니다.

- **JSP 컴파일 오류.** 선택적 업그레이드를 수행하기 전에 다음 표에 나와 있는 Application Server 패치를 먼저 적용해야 합니다.

표 1-11 Application Server를 릴리스 5로 업그레이드하지 않을 때 필요한 패치¹

설명	패치 아이디: Solaris 9 및 10	패치 아이디: Linux
릴리스 3 및 릴리스 4용 Application Server	119166-17(SPARC) 119166-17(x86)	119168-17

1. 표시된 패치 개정 버전 번호는 필요한 최소 버전입니다. 새로운 개정 버전이 있는 경우 해당 버전을 사용하십시오.

패치를 적용하지 못하면 Application Server에서 JSP 컴파일 오류가 발생합니다. [표 1-11](#)에 나와 있는 패치를 소급 적용하면 문제를 해결할 수 있습니다.

- **Linux에서 ANT 공유 구성 요소 바이너리 재배포.** 릴리스 5 ANT는 이전 버전과 다른 경로에 있습니다. *AppServer8-base/config/asenv.conf* 파일에서 ANT를 가리키는 다음 Application Server 환경 변수를 찾습니다.

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

이 변수를 다음과 같이 변경합니다.

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

그리고 Application Server를 다시 시작해야 합니다.

Portal Server 임시 기능 릴리스(IFR) 7.0을 Java ES 5로 업그레이드

Web Server 환경에서 Portal Server를 임시 기능 릴리스(IFR) 7.0 2005Q4에서 릴리스 5로 업그레이드하는 경우 [54페이지](#)의 "업그레이드 순서 지침"에 나와 있는 지침의 예외 상황에 대해 [353페이지](#)의 "임시 기능 릴리스 7.0에서 Portal Server 업그레이드"를 참조하십시오.

Java ES 5 업그레이드 및 Solaris 10 영역

이 절에서는 Solaris 10 영역에서 Java ES 소프트웨어를 업그레이드할 때 발생하는 문제를 해결하고 이러한 환경에서 권장되는 사항에 대해 설명합니다. 이 절은 Java Enterprise System 5 설치 계획 설명서(<http://docs.sun.com/doc/819-5079>)에서 Java ES 5 및 Solaris 10 영역에 대한 내용을 보충한 것입니다.

이 장은 다음 내용으로 구성되어 있습니다.

- [Java ES 설치 프로그램의 영역 지원](#)
- [권장 업그레이드 방법](#)
- [특별 사례 또는 예외](#)

Java ES 설치 프로그램의 영역 지원

Java ES 5 설치 프로그램에서는 Java ES 제품 구성 요소 업그레이드 및 설치, 그리고 공유 구성 요소 동기화를 위한 정식 영역 지원을 제공합니다. 문제가 발생할 수 있는 업그레이드 시나리오를 방지하기 위해 설치 프로그램에 여러 가지 정책이 구현되었습니다.

제품 구성 요소 업그레이드

32페이지의 ["Java ES 설치 프로그램의 업그레이드 기능 사용"](#)에 설명된 대로 Java ES 설치 프로그램은 제한된 수의 제품 구성 요소와 해당 공유 구성 요소를 업그레이드하는 데 사용할 수 있습니다. 업그레이드 기능은 전역 영역과 모든 비전역 영역에 적용됩니다.

하지만 여기에는 영역과 관련된 세 가지 예외가 있습니다.

- 스파스 루트 영역에서 일부 공유 구성 요소는 읽기 전용 디렉토리에 있기 때문에 설치 또는 업그레이드할 수 없습니다. 이런 경우 제품 구성 요소 업그레이드는 전역 영역에서 해당 공유 구성 요소를 설치 또는 업그레이드할 때까지 중단됩니다. 또한 설치 프로그램에서는 "선택한 구성 요소에 필요한 다음 공유 구성 요소를 스파스 루트 영역에서 설치 또는 업그레이드할 수 없습니다. 이러한 공유 구성 요소를 전역 영역에서 설치 또는 업그레이드하고 진행하십시오. 모든 공유 구성 요소 옵션을 사용하십시오."와 같은 메시지가 표시됩니다.
- Application Server와 Message Queue는 둘 다 Solaris 운영 체제와 함께 번들로 제공됩니다. 이러한 버전은 스파스 루트 영역에서 직접 업그레이드할 수 없습니다. 이러한 두 가지 번들 구성 요소에 대한 자세한 내용은 61페이지의 ["제품 구성 요소 특별 사례"](#)를 참조하십시오.
- 전역 영역에 비전역 영역이 있는 경우 현재 설치된 모든 공유 구성 요소를 업그레이드하고 선택된 구성 요소에 필요한 누락된 모든 공유 구성 요소를 설치하는 대신 설치 프로그램은 특정 제품 구성 요소에서 필요한지 여부에 관계없이 *모든* Java ES 공유 구성 요소를 릴리스 5로 동기화합니다. 이를 통해 모든 릴리스 5 공유 구성 요소를 비전역 영역으로 전파할 수 있으므로 비전역 영역에서 공유 구성 요소 버전이 혼합되지 않도록 할 수 있습니다.

주 비전역 영역에서 제품 구성 요소 설치 또는 업그레이드에 방해가 될 수 있는 몇몇 특별 사례 또는 예외가 있으며 이러한 문제는 61페이지의 ["특별 사례 또는 예외"](#)에서 설명합니다.

모든 공유 구성 요소 동기화

모든 공유 구성 요소를 해당 릴리스 5 버전으로 동기화해야 하는 경우를 위해 릴리스 5에는 공유 구성 요소 동기화 옵션이 포함되어 있습니다. "모든 공유 구성 요소" 옵션을 선택하면 설치 프로그램이 특정 제품 구성 요소에 필요한지 여부에 관계없이 현재 설치된 모든 공유 구성 요소를 업그레이드하고 누락된 공유 구성 요소를 설치합니다. 이 옵션은 전역 영역과 전체 루트 영역에 적용되지만 스파스 루트 영역에는 적용되지 않습니다.

"모든 공유 구성 요소" 옵션에 대한 자세한 내용은 [65페이지의 "모든 공유 구성 요소 동기화"](#)에 자세히 설명되어 있으며 다음 두 가지 영역 기반 업그레이드 시나리오에서 필요합니다.

- **제품 구성 요소 수동 업그레이드.** Java ES 설치 프로그램을 사용하여 업그레이드할 수 없는 제품 구성 요소를 업그레이드하는 데 필요한 공유 구성 요소를 설치하고 업그레이드하는 데는 "모든 공유 구성 요소" 옵션이 필요합니다.
- **스파스 루트 영역에서 업그레이드.** 일부 공유 구성 요소는 기본 스파스 루트 영역에서 설치 또는 업그레이드할 수 없습니다. 따라서 Java ES 설치 프로그램을 사용하여 스파스 루트 영역에서 제품 구성 요소를 업그레이드하는 경우 관련 구성 요소에 따라 전역 영역에서 공유 구성 요소를 먼저 동기화해야 할 수 있습니다. 이 경우 전역 영역에서 "모든 공유 구성 요소" 옵션을 사용하여 필요한 공유 구성 요소 설치 및 업그레이드를 수행하면 됩니다.

공유 구성 요소와 관련된 Java ES 설치 프로그램의 영역 동작에 대한 요약은 Java Enterprise System 5 설치 계획 설명서(<http://docs.sun.com/doc/819-5079>)에서 Java ES 5 및 Solaris 10 영역에 대한 정보를 확인하십시오.

권장 업그레이드 방법

업그레이드 계획을 수립할 때는 Java Enterprise System 5 설치 계획 설명서(<http://docs.sun.com/doc/819-5079>)에서 설명한 영역 설치 및 관리 전략을 염두에 두고 Java ES 소프트웨어의 기존 다중 영역 배포를 검토해 보아야 합니다. 어떤 경우에는 다음의 권장되는 방법을 구현하기 위해 하나 이상의 영역에서 구성 요소를 제거하고 다른 영역에 다시 설치해야 할 수 있습니다.

- 전략을 혼합하지 마십시오. 특히 다음과 같습니다.
 - Java ES 영역 배포 및 관리 전략을 가능하면 단순히 세우십시오. 동일한 컴퓨터에 Java ES 구성 요소의 전체 루트 및 스파스 루트 배포를 혼합하지 마십시오. 스파스 루트 영역 배포를 지원하는 데 필요한 절차와 방법이 전체 루트 영역 배포와 충돌할 수 있습니다.
 - 동일한 Java ES 제품 구성 요소는 버전이 달라도 전역 영역과 비전역 영역에 함께 설치하지 마십시오. 전역 영역 설치를 업그레이드하는 데 필요한 절차로 인해 비전역 영역 설치가 손상될 수 있습니다.
 - 릴리스 4(또는 이전) Java ES 구성 요소를 전체 루트 영역에 설치한 경우 전역 영역에서 Java ES 구성 요소를 릴리스 5로 업그레이드하지 마십시오. 전역 영역에서 업그레이드하면 전체 루트 영역에서 릴리스 4 및 릴리스 5 파일이 섞일 수 있습니다.
- 업그레이드 방법
 - 설치된 모든 릴리스 4 제품 구성 요소를 릴리스 5로 업그레이드하려면 전역 영역에서 모든 Java ES 공유 구성 요소를 동기화한 다음 설치된 영역에서 원하는 제품 구성 요소를 업그레이드하십시오. 릴리스 5 공유 구성 요소는 이전 버전과 호환됩니다.
 - 릴리스 4 또는 릴리스 5 제품 구성 요소가 비전역 환경에 설치되어 있는 경우 해당 환경에 비전역 영역을 추가하고 새로 추가한 비전역 영역에 제품 구성 요소를 설치하려면 전역 영역에서 구성 요소를 제거하고 비전역 영역에서 다시 설치해야 할 수 있습니다.

특별 사례 또는 예외

몇 가지 특별 사례가 있는데 이 중 일부는 Java ES 공유 구성 요소와 Java ES 제품 구성 요소가 Solaris 10과 함께 번들로 제공되는 경우에 발생합니다. 번들의 특성상 이러한 Java ES 구성 요소는 자동으로 전역 영역에 존재하게 되며, 이 때문에 전역 영역에서 만든 비전역 영역에도 존재하게 됩니다.

제품 구성 요소 특별 사례

- **Message Queue.** Message Queue는 Solaris 10과 함께 번들로 제공되므로 전역 영역에서 먼저 Message Queue를 제거하지 않는 경우 비전역 영역을 만들면 자동으로 전파됩니다. Message Queue는 스파스 루트 영역에서 설치하거나 업그레이드할 수 없습니다. Java ES 설치 프로그램을 사용하여 전역 영역에서 설치 또는 업그레이드하는 경우 Message Queue는 다른 제품 구성 요소와 달리 기본적으로 비전역 영역으로 전파됩니다.

- **Application Server.** Application Server는 Solaris 10과 함께 번들로 제공되므로 전역 영역에서 먼저 Application Server를 제거하지 않는 경우 비전역 영역을 만들면 자동으로 전파됩니다. /usr에 설치된 번들 Application Server가 이런 방식으로 전파되면 기본적으로 /usr이 읽기 전용이기 때문에 Java ES 설치 프로그램이 스파스 루트에서 업그레이드할 수 없습니다. 이 문제를 해결하려면 번들로 제공된 Application Server 패키지를 전역 영역에서 수동으로 제거한 다음 스파스 루트 영역에 릴리스 5 Application Server를 설치해야 합니다. [216 페이지의 "Solaris OS만 해당: 운영 체제에 번들로 제공된 Application Server 패키지 수동 제거"](#)를 참조하십시오.
- **Sun Cluster.** Sun Cluster 소프트웨어는 비전역 영역에서 지원되지 않습니다.

공유 구성 요소 특별 사례

- **SJWC(Sun Java Web Console).** Solaris 10(Update 1 및 Update 2)에 번들로 제공된 SJWC 패키지는 Java ES 설치 프로그램을 사용하여 제거할 수 없습니다. 이러한 이전 SJWC 패키지는 SUNW_PKG_ALLZONES 속성이 True로 설정되어 있는데, 이는 모든 영역에서 패키지가 동일해야 하며 전역 관리자만 관리할 수 있음을 나타냅니다. 따라서 이러한 패키지는 전역 영역에서 수동으로 제거한 다음 올바른 패키지로 대체해야 합니다.

Java ES 설치 프로그램이 선택된 제품 구성 요소를 비전역 영역에 설치하려고 하면 SJWC를 업그레이드해야 한다는 메시지가 표시되고 설치 프로그램이 중단됩니다. 이 문제는 Solaris 10, Update 1 및 2에서 설치할 때 발생합니다.

해결 방법으로 전역 영역에서 SJWC의 기존 패키지를 제거한 다음 올바른 영역 전파 속성 값을 가진 릴리스 5 SJWC로 대체하는 특별한 스크립트가 개발되었습니다. 자세한 내용은 Java Enterprise System 5 UNIX용 설치 설명서를 참조하십시오.

- **CAC(공용 에이전트 컨테이너).** 버전 1.1은 Sun Cluster, Sun Cluster Geographic Edition 또는 Sun Cluster 에이전트가 설치된 경우에만 설치됩니다. "모든 공유 구성 요소" 옵션이 선택된 경우에는 설치되지 않습니다. 이 경우에는 버전 2.0만 설치됩니다.

Java ES 공유 구성 요소 업그레이드

이 장에서는 Java ES 공유 구성 요소를 Java ES 5(릴리스 5)로 업그레이드하는 방법을 설명합니다.

각 Java ES 제품 구성 요소는 Java ES 공유 구성 요소라고 하는 하나 이상의 로컬 공유 라이브러리에 종속됩니다. 공유 구성 요소는 제품 구성 요소를 설치하는 과정에서 설치되는 제품 구성 요소에 따라 Java ES 설치 프로그램에서 자동으로 설치합니다. Java ES 제품 구성 요소 배포 중에 명시적으로 선택, 설치 또는 구성되지 않습니다.

비슷하게 Java ES 설치 프로그램을 사용하여 업그레이드할 수 있는 제품 구성 요소의 경우 해당 공유 구성 요소가 자동으로 업그레이드됩니다.

그러나 본 업그레이드 설명서의 여러 장에서 설명된 것처럼 제품 구성 요소를 수동으로 업그레이드하는 경우에는 이 장에서 설명한 절차를 사용하여 공유 구성 요소를 명시적으로 업그레이드해야 할 수 있습니다.

이 장의 내용은 다음과 같습니다.

- [64페이지의 "공유 구성 요소 업그레이드 개요"](#)
- [68페이지의 "공유 구성 요소 업그레이드 절차"](#)
- [68페이지의 "특별 업그레이드 절차"](#)

공유 구성 요소를 업그레이드하기 전에 [68페이지의 "특별 업그레이드 절차"](#)를 읽어야 합니다.

공유 구성 요소 업그레이드 개요

Java ES 5(릴리스 5)로의 공유 구성 요소 업그레이드는 [1장](#), "[업그레이드 계획](#)"에서 설명한 것처럼 큰 업그레이드 계획의 일부로 수행되어야 합니다. 성공적인 업그레이드를 위해서는 1장을 주의하여 읽고 사용자의 요구에 맞는 업그레이드 계획을 준비하십시오.

이 절은 다음 내용으로 구성되어 있습니다.

- [일반적인 고려 사항](#)
- [공유 구성 요소 동기화](#)
- [모든 공유 구성 요소 동기화](#)
- [Solaris 10 영역 고려사항](#)

일반적인 고려 사항

공유 구성 요소를 업그레이드하는 경우 다음 문제를 고려하십시오.

- **운영 체제 문제.** [42페이지의 "운영 체제 고려 사항"](#)에 설명된 대로 운영 체제 업그레이드를 수행하십시오. Solaris 10 운영 체제를 제외한 모든 플랫폼에서는 공유 구성 요소를 업그레이드하기 전에 운영 체제를 업그레이드하십시오.
- **순서 지침.** [54페이지의 "업그레이드 순서 지침"](#)에 설명된 순서 지침을 검토합니다. 일반적으로 공유 구성 요소가 먼저 업그레이드됩니다. 그러나 업그레이드 프로세스를 시작하기 전에 Java ES 릴리스 5로 업그레이드하기 위한 전체 과정을 파악해야 합니다.

공유 구성 요소 동기화

많은 수(약 30개)의 공유 구성 요소를 테스트하고 지원하는 어려움 및 Java ES 공유 구성 요소와 Java ES 제품 구성 요소 간의 복잡한 상호 작용 때문에 단일 운영 체제 인스턴스 내의 모든 공유 구성 요소를 동일한 Java ES 버전으로 동기화해야 합니다. 운영 체제 인스턴스는 Solaris 9 또는 Linux 운영 체제를 실행하는 단일 컴퓨터를 의미하며 Solaris 10 운영 체제의 경우에는 단일 컴퓨터에서 실행되는 가상 운영 체제 환경(영역)을 의미합니다.

다시 말해 한 운영 체제 인스턴스에 설치되는 모든 Java ES 공유 구성 요소는 동일한 버전이어야 합니다. 이러한 동기화 요구 사항은 Java ES 공유 구성 요소를 설치하고 업그레이드하는 방법에 대해 특정 제한을 설정합니다.

- Java ES 공유 구성 요소의 다른 버전은 다른 운영 체제 인스턴스에만 있을 수 있습니다. 예를 들어 Java ES 릴리스 4 공유 구성 요소를 한 운영 체제 인스턴스에 설치하고 Java ES 릴리스 5 공유 구성 요소를 다른 운영 체제 인스턴스에 설치할 수 있지만 이들을 같은 운영 체제 인스턴스에서 조합하여 사용할 수는 없습니다.
- 운영 체제 인스턴스의 공유 구성 요소가 업그레이드되거나 높은 버전의 새 공유 구성 요소가 추가되면 해당 운영 체제 인스턴스의 모든 공유 구성 요소도 함께 업그레이드해야 합니다. 공유 구성 요소는 이전 버전과 호환되어야 하므로 릴리스 4 제품 구성 요소가 릴리스 5 공유 구성 요소와 작동하는 데는 문제가 없습니다.

예를 들어 릴리스 5 제품 구성 요소가 하나 이상의 릴리스 4 제품 구성 요소가 있는 운영 체제 인스턴스에 설치되었다고 가정해 보겠습니다. 릴리스 5 제품 구성 요소에는 몇 가지 릴리스 5 공유 구성 요소가 필요하므로 이 경우 동기화 요구 사항에 따라 릴리스 5 제품 구성 요소를 설치할 때 해당 운영 체제 인스턴스에 있는 모든 릴리스 4 공유 구성 요소를 함께 릴리스 5로 업그레이드해야 합니다. 설치되는 릴리스 5 제품 구성 요소가 이미 설치되어 있는 구성 요소와 다른 공유 구성 요소를 필요로 하는 경우에도 마찬가지입니다.

비슷하게, 릴리스 4 제품 구성 요소를 릴리스 5로 업그레이드하는 경우 종속된 몇몇 공유 구성 요소도 업그레이드해야 하면 업그레이드되는 특정 릴리스 4 제품 구성 요소가 종속성을 가지는지 여부에 관계없이 해당 운영 체제에 설치된 모든 공유 구성 요소를 릴리스 5로 업그레이드해야 합니다.

모든 공유 구성 요소 동기화

Java ES 설치 프로그램에는 모든 공유 구성 요소를 해당 릴리스 5 버전으로 동기화해야 하는 경우를 위해 모든 동기화 구성 요소 동기화 기능이 포함되어 있습니다.

Java ES 설치 프로그램의 구성 요소 선택 페이지에서 "모든 공유 구성 요소"를 선택하면 설치 프로그램은 특정 제품 구성 요소에서 필요하지 여부에 관계없이 기존 공유 구성 요소를 업그레이드하고 누락된 공유 구성 요소를 설치합니다.

모든 공유 구성 요소 동기화 기능은 Java ES 설치 프로그램이 명시적으로 업그레이드하지 않는 제품 구성 요소(Application Server, Message Queue, HADB 및 Java DB를 제외한 모든 Java ES 제품 구성 요소)의 업그레이드를 지원합니다. 예를 들어 모든 공유 구성 요소 동기화 기능은 Web Proxy Server 및 Portal Server를 릴리스 5로 업그레이드하는 데 사용됩니다.

이런 경우 *모든* 공유 구성 요소를 동기화하는 이유는 설치 프로그램이 컴퓨터에서 동기화해야 하는 공유 구성 요소를 정확히 알 수 없기 때문입니다. 따라서 이 기능은 *모든* Java ES 공유 구성 요소를 릴리스 5 버전으로 설치하거나 업그레이드합니다.

또한 모든 공유 구성 요소 동기화 기능은 아래 "[Solaris 10 영역 고려사항](#)"에서 설명한 대로 여러 가지 영역 시나리오를 지원합니다.

Solaris 10 영역 고려사항

공유 구성 요소의 고려 사항에 따라 Solaris 10 다중 영역 환경에 Java ES를 배포하는 방법에 몇 가지 제한이 발생합니다.

이러한 고려 사항의 대부분은 스파스 루트 영역의 읽기 전용 파일 시스템 때문에 많은 공유 구성 요소를 스파스 루트 영역에 설치할 수 없는 문제와 관련이 있습니다. 이러한 제한은 기본 디렉토리가 /usr(기본적으로 전역 영역에서 공유하는 디렉토리)인 공유 구성 요소에 적용됩니다.

여러 Java ES 공유 구성 요소를 스파스 루트 영역에 설치할 수 없다는 것은 이러한 공유 구성 요소에 종속되어 있는 제품 구성 요소를 스파스 루트 영역에 성공적으로 설치하려면 먼저 공유 구성 요소가 비전역 영역으로 전파되는 전역 영역에 해당 구성 요소를 설치 또는 업그레이드해야 함을 의미합니다.

공유 구성 요소와 관련된 동기화 요구 사항([64페이지의 "공유 구성 요소 동기화" 참조](#))과 스파스 루트 영역에서 제품 구성 요소를 설치 또는 업그레이드하려면 전역 영역에 공유 구성 요소를 설치 및 업그레이드해야 한다는 조건 때문에 결국 전역 영역에 있는 모든 공유 구성 요소를 해당 릴리스 5 버전으로 동기화해야 합니다. 따라서 스파스 루트 영역의 특정 제품 구성 요소에서 필요한지 여부에 관계없이 모든 기존 공유 구성 요소가 업데이트되고 누락된 공유 구성 요소가 설치됩니다.

전역 영역에서 공유 구성 요소의 이러한 동기화는 Java ES 설치 프로그램의 구성 요소 선택 페이지에서 "모든 공유 구성 요소"를 선택하여 수행할 수 있습니다.

공유 구성 요소를 설치하고 전역 영역에서 전파하는 경우(예를 들어 전역 영역에서 Java ES 제품 구성 요소 설치) *모든* 영역에서 공유 구성 요소가 동기화되도록 특별한 주의해야 합니다. 그렇지 않으면 비전역 영역에 있는 이전 버전의 공유 구성 요소가 전역 영역에서 전파된 릴리스 5 공유 구성 요소와 혼합될 수 있습니다.

공유 구성 요소를 업그레이드하는 방법

특정 Java ES 제품 구성 요소를 업그레이드하는 데 필요한 공유 구성 요소를 업그레이드하는 방법은 두 가지가 가능합니다. 첫 번째는 필요한 모든 공유 구성 요소를 확인하고 해당 릴리스 5 버전으로 수동 설치하거나 업그레이드하는 것입니다. 두 번째는 Java ES 설치 프로그램의 모든 공유 구성 요소 동기화 기능을 사용하는 것입니다.

하지만 공유 구성 요소와 관련된 동기화 요구 사항([64페이지의 "공유 구성 요소 동기화" 참조](#)) 때문에 현실적으로 사용 가능한 방법은 Java ES 설치 프로그램의 모든 공유 구성 요소 동기화 기능을 사용하는 것입니다.

이는 특정한 경우에 어떤 공유 구성 요소를 설치 또는 업그레이드해야 하는지 확인하기 어렵기 때문입니다. 그리고 필요한 공유 구성 요소를 성공적으로 확인했다라도 이러한 공유 구성 요소를 직접 설치하는 일은 매우 어렵습니다. 일부 공유 구성 요소는 릴리스 5 버전으로 패치할 수 있지만 이전 패키지를 릴리스 5 패키지로 대체해야 하는 구성 요소도 있습니다. 업그레이드 방법에 따라 일부 이전 패키지를 수동으로 제거해야 할 수도 있습니다.

이전에는 수동으로 공유 구성 요소를 업그레이드하는 어려웠기 때문에 제품 구성 요소를 업그레이드하는 데에 있어 큰 장애물로 작용했습니다. 모든 구성 요소 동기화 기능은 불필요한 공유 구성 요소까지 설치하는 단점이 있지만 Java ES 릴리스 3 및 릴리스 4에서 필요한 공유 구성 요소를 수동으로 업그레이드하는 것에 비해 상당히 개선된 것입니다.

공유 구성 요소 업그레이드 절차

공유 구성 요소를 업그레이드하는 절차는 Java ES 설치 프로그램의 모든 공유 구성 요소 기능을 사용하는 것입니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. [68페이지의 "특별 업그레이드 절차"](#)에 설명된 대로 서비스 또는 프로세스를 종료합니다.

3. Java ES 설치 프로그램을 시작합니다.

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

여기서 *os_arch*는 *solaris_sparc*과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

4. 구성 요소 선택 페이지에서 "모든 공유 구성 요소"를 선택합니다.
5. 선택을 확인합니다.

모든 공유 구성 요소가 해당 릴리스 5 버전으로 동기화됩니다.

6. Java ES 설치 프로그램을 종료합니다.

후속 절차에 대해서는 [68페이지의 "특별 업그레이드 절차"](#)를 참조하십시오.

특별 업그레이드 절차

이 절에서는 다음 공유 구성 요소를 업그레이드하는 데 필요한 특별한 절차에 대해 설명합니다.

- [69페이지의 "Java SE 업그레이드 절차"](#)
- [71페이지의 "공통 에이전트 컨테이너 업그레이드 절차"](#)

Java SE 업그레이드 절차

Java ES 릴리스 5는 Java SE 5.0 버전 5.0 Update 9로 식별되는 Java Platform, Standard Edition(Java SE) 버전 5.0용으로 인증되었습니다(Java SE 5.0은 개발자 버전 1.5.0로 불리기도 합니다).

다른 공유 구성 요소와 마찬가지로 Java SE도 Java ES 설치 프로그램을 사용하여 업그레이드합니다. 하지만 설치 프로그램을 사용하면 다른 공유 구성 요소가 설치 또는 업그레이드되는지에 상관없이 Java SE 5.0 Update 9로 업그레이드합니다. Java SE 5.0 Update 9를 설치하기 위해 "모든 공유 구성 요소"를 선택할 필요는 없습니다.

주	Java SE를 업그레이드하는 경우 먼저 현재 설치된 Java SE에 종속되어 있는 서비스를 종료할 수 있습니다. 이것은 이러한 서비스로 인해 문제가 발생하는 것을 방지하기 위해서입니다. Java SE에 종속되어 있는 서비스를 종료하지 않으면 Java SE를 릴리스 5 버전으로 업그레이드한 다음 시스템을 재부팅해야 합니다.

설치 프로그램이 컴퓨터에서 Java SE 패키지의 이전 버전 또는 불완전한 패키지를 발견하면 대화 상자를 표시하여 알려줍니다(완전한 패키지: SUNWj5rt, SUNWj5rtx, SUNWj5dev, SUNWj5dmo, SUNWj5dvx, SUNWj5man, SUNWj5cfg, SUNWj5dmx). 구성 요소 선택 페이지 다음에 표시되는 대화 상자에서 Java SE를 릴리스 5 수준으로 자동 업그레이드할지 또는 자동 업그레이드를 통과하고 수동으로 업그레이드할지 선택할 수 있습니다.

- **수동 업그레이드.** Java SE를 수동 업그레이드하도록 선택한 경우 다음 절차를 사용하십시오.
 - a. 다른 구성 요소를 설치 또는 업그레이드하기 전에 Java ES 설치 프로그램을 종료합니다.
 - b. 해당 Java SE 버전을 수동으로 설치합니다.
 - c. Java ES 설치 프로그램을 다시 실행합니다.

설치 프로그램에서 Java SE의 정확한 버전 또는 완전한 버전을 감지하면 계속 진행할 수 있지만, 그렇지 않으면 이전 대화 상자가 표시됩니다.
- **자동 업그레이드.** Java SE를 자동 업그레이드하도록 선택한 경우 Java ES 설치 프로그램이 Java SE를 Version 5.0 Update 9로 업그레이드하며 업그레이드 동작은 다음과 같습니다.

- 업그레이드해도 다른 응용 프로그램이 해당 버전에 종속되어 있을 수 있으므로 이전에 설치한 주요 Java SE 릴리스(예: Java SE 버전 1.4.2 또는 Java SE 버전 1.6.x)가 제거되지는 *않습니다*. 하지만 업그레이드를 수행하면 릴리스 5 버전을 참조하는 심볼릭 링크가 아래 표와 같이 설정됩니다.

표 2-1 Java SE에 대한 플랫폼별 심볼릭 링크

플랫폼	심볼릭 링크	위치
Solaris	/usr/jdk/entsys-j2se	/usr/jdk/instances/jdk1.5.0
Linux	/usr/jdk/entsys-j2se	/usr/java/jdk1.5.0_09

이전 버전을 사용하는 서비스를 위해 주요 Java SE 릴리스(예: Java SE 1.4.2)에 대한 포인터를 그대로 유지해야 합니다. 이전 버전의 Java SE에 대한 심볼릭 링크를 유지하는 방법에 대한 내용은 해당 제품 구성 요소 설명서를 참조하십시오.

- 업그레이드를 *수행하면* 이전에 설치한 부차적 Java SE 릴리스(예: Java SE 버전 5.0 Update 5)가 제거되며 Java SE 5.0 Update 9로 대체됩니다.

Java SE 심볼릭 링크 확인

Java Enterprise System는 Java ES 서비스에서 사용할 올바른 Java SE 런타임을 찾을 수 있도록 지원되는 Java SE 플랫폼 버전에 대한 심볼릭 링크를 유지 관리합니다.

예를 들어 Solaris 운영 체제에서 다음과 같이 심볼릭 링크를 확인할 수 있습니다.

```
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
```

여기서 /usr/jdk/instances/jdk1.5.0은 기본 위치입니다.

현재 Java SE 버전 확인

Java ES 설치에서 사용 중인 Java SE 버전을 확인하려면 다음 명령을 사용하여 Java SE 심볼릭 링크가 참조하는 Java SE 버전을 확인합니다.

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

출력되는 내용은 다음 표에 정리되어 있습니다.

표 2-2 Java SE 버전 확인 출력

Java ES 릴리스	Java SE 버전 번호	Java SE 버전 문자열
릴리스 2	1.4.2 Update 5	1.4.2_05
릴리스 3	5.0 Update 1	1.5.0_01
릴리스 4	5.0 Update 4	1.5.0_04
릴리스 5	5.0 Update 9	1.5.0_09

공통 에이전트 컨테이너 업그레이드 절차

Sun Cluster 소프트웨어 업그레이드를 위한 준비(버전 1.1 패치)로 공통 에이전트 컨테이너 공유 구성 요소를 업그레이드하는 경우 Sun Cluster Software Installation Guide for Solaris OS(<http://docs.sun.com/doc/819-0420/6n2rlmncr?a=view>)의 "Upgrading Sun Cluster Software"에서 종속성 소프트웨어 업그레이드 절차를 따르십시오.

위 설명서의 업데이트는 Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions(<http://docs.sun.com/doc/819-4351>)에서 "Upgrading to Sun Cluster 3.1 8/05 Software"를 참조하십시오.

다른 Java ES 구성 요소를 업그레이드하기 위한 준비로 공통 에이전트 컨테이너 공유 구성 요소를 업그레이드하는 경우(버전 2.0으로 업그레이드) 경로 이름 변수에 주의하면서 다음 절차를 따르십시오.

표 2-3 공통 에이전트 컨테이너 디렉토리 경로

경로 이름 변수	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. 현재 설치에서 사용자 정의 구성 설정을 사용하는 경우 다음 명령을 사용하여 구성 설정을 캡처합니다.

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

결과는 다음과 비슷합니다.

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

위의 예는 기본값을 보여 줍니다. [72페이지의 단계 4](#)에서 기본값이 아닌 설정을 사용하는 방법을 참조하십시오.

2. 다음 명령을 사용하여 공통 에이전트 컨테이너를 중지합니다.

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

종료 코드가 0이 아니면 강제 중지합니다.

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

3. Java ES 설치 프로그램의 모든 공유 구성 요소 동기화 기능을 사용하여 공통 에이전트 컨테이너를 업그레이드합니다.

[68페이지의 "공유 구성 요소 업그레이드 절차"](#)를 참조하십시오.

4. 앞서 [71페이지의 단계 1](#)에서 확보한 사용자 정의 구성 설정을 적용합니다.

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

5. Java SE를 Java SE 버전 5로 업그레이드한 경우 rebuild-dependencies 유틸리티를 실행합니다.

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

이 명령은 다음과 같은 내용을 출력합니다.

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


6. 공통 에이전트 컨테이너 서비스를 다시 시작합니다.

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

7. 공통 에이전트 컨테이너의 업그레이드 상태를 확인합니다.

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


Sun Cluster 소프트웨어

이 장에서는 Sun Cluster 소프트웨어를 Java ES 5(릴리스 5): Sun Cluster 3.1 8/05 소프트웨어로 업그레이드하는 방법을 설명합니다.

이 장에서는 Sun Cluster 소프트웨어를 릴리스 5로 업그레이드할 때 고려해야 할 사항에 대해 설명합니다.

Sun Cluster 소프트웨어는 Solaris 플랫폼에서만 지원됩니다.

이 장에 설명된 Sun Cluster 소프트웨어의 업그레이드는 Sun Cluster 프레임워크 소프트웨어 및 Sun Cluster 데이터 서비스 소프트웨어 또는 에이전트를 모두 포함합니다.

- [76페이지의 "Sun Cluster 소프트웨어 업그레이드 개요"](#)
- [80페이지의 "Java ES 릴리스 5로 Sun Cluster 소프트웨어 업그레이드"](#)

Sun Cluster 소프트웨어 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Sun Cluster 소프트웨어의 일반 요소를 설명합니다.

- [릴리스 5 Sun Cluster 소프트웨어 정보](#)
- [Sun Cluster 소프트웨어 업그레이드 로드맵](#)
- [Sun Cluster 데이터](#)
- [Sun Cluster 업그레이드 전략](#)

릴리스 5 Sun Cluster 소프트웨어 정보

릴리스 5 Sun Cluster 소프트웨어는 릴리스 4 Sun Cluster 소프트웨어와 비교할 때 부차적 업그레이드입니다. 자세한 내용은 Sun Cluster 릴리스 노트 (<http://docs.sun.com/doc/819-1405/6n3p13hac?a=view>)를 참조하십시오. 릴리스 5에서 Solaris 8 운영 체제를 지원하지 않는다는 점을 제외하고 릴리스 5 Sun Cluster는 근본적으로 4와 동일합니다.

Sun Cluster 3.1 8/05 설명서에서 변경된 사항과 추가된 사항에 대해서는 Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions (<http://docs.sun.com/doc/819-4351>)를 참조하십시오. Sun Cluster 3.1 8/05 설명서의 절차는 릴리스 5 Sun Cluster 소프트웨어에 대해 유효합니다.

-
주	Solaris 8 OS에서 Sun Cluster 3.1 8/05 소프트웨어가 필요한 경우 Java ES 릴리스 4 배포에서 Sun Cluster 3.1 8/05 소프트웨어를 구해야 합니다. 이 소프트웨어는 다음 위치에 있습니다. http://www.sun.com/software/javaenterprisesystem/previous/index.xml Java ES 5 설치 프로그램은 Solaris 8 OS에서 Sun Cluster 3.1 8/05 소프트웨어 설치를 더 이상 지원하지 않습니다. Solaris 8 OS에서 Sun Cluster 3.1 8/05 소프트웨어를 설치하려면 다음 단계를 수행하십시오. 1. Java ES 릴리스 4 배포에서 Sun Cluster 3.1 8/05 소프트웨어를 설치합니다. 2. Sun Cluster 3.1 8/05 소프트웨어에 필요한 모든 패치를 설치합니다.
-

Sun Cluster 소프트웨어 업그레이드 로드맵

표 3-1에서는 지원되는 Sun Cluster를 Java ES 5(릴리스 5)로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 운영 체제에만 적용됩니다.

Sun Cluster 버전은 Java ES 릴리스와 일대일로 매핑되지 않습니다. 이전에는 Sun Cluster 소프트웨어의 임시 기능 릴리스(IFR)가 정식 Java ES 릴리스 중간 단계의 Java ES 릴리스에 통합되었습니다. 이런 이유로 Java ES 릴리스 2, 릴리스 3 및 릴리스 4 Sun Cluster를 릴리스 5 Sun Cluster로 업그레이드하는 데는 표 3-1에 설명된 대로 Sun Cluster 3.1 4/04, Sun Cluster 3.1 9/04 및 Sun Cluster 3.1 8/05 소프트웨어를 릴리스 5로 업그레이드하는 작업이 포함됩니다.

표 3-1 Java ES 5(릴리스 5) 업그레이드 경로: Sun Cluster 3.1 8/05

Java ES 릴리스	Sun Cluster 소프트웨어 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Cluster 3.1 8/05 (2005Q4)	직접 업그레이드: Sun Cluster scinstall 유틸리티를 사용하여 수행됩니다.	없음
릴리스 3	Sun Cluster 3.1 9/04 또는 Sun Cluster 3.1 8/05 (IFR)	직접 업그레이드: Sun Cluster scinstall 유틸리티를 사용하여 수행됩니다.	클러스터 구성이 업그레이드된 버전으로 자동 마이그레이션됩니다. IFR을 릴리스 5로 업그레이드하는 데 재구성은 필요하지 않습니다.
릴리스 2	Sun Cluster 3.1 4/04 또는 Sun Cluster 3.1 9/04 (IFR)	직접 업그레이드: Sun Cluster scinstall 유틸리티를 사용하여 수행됩니다.	클러스터 구성 요소가 업그레이드 버전으로 자동 마이그레이션됩니다.
릴리스 1	Sun Cluster 3.1	직접 업그레이드 인증 안 됨: 그러나 scinstall 유틸리티를 사용하여 수행할 수 있습니다.	클러스터 구성 요소가 업그레이드 버전으로 자동 마이그레이션됩니다.
이전 Java ES 릴리스	Sun Cluster 3.0	직접 업그레이드 인증 안 됨: 그러나 scinstall 유틸리티를 사용하여 수행할 수 있습니다.	클러스터 구성 요소가 업그레이드 버전으로 자동 마이그레이션됩니다.

1. Sun Cluster 소프트웨어에서 사용하는 공유 구성 요소를 릴리스 5로 업그레이드하지 않은 경우 릴리스 4에서 릴리스 5로 업그레이드할 필요가 없습니다.

Sun Cluster 데이터

다음 표는 Sun Cluster 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 3-2 Sun Cluster 데이터 사용

데이터 유형	위치	사용
클러스터 구성 데이터	모든 클러스터 노드에서 복제되고 동기화된 클러스터 구성 저장소(주의: 노드 또는 전체 클러스터가 작동을 중지할 수도 있으므로 CCR 파일은 절대로 수동으로 편집하지 마십시오.)	Sun Cluster 작업의 모든 요소에 대한 구성 정보 저장: 클러스터 노드 구성, 페일오버 메커니즘, 자원 관리 등

Sun Cluster 업그레이드 전략

Sun Cluster 업그레이드를 위한 전략은 일반적으로 1장, "[업그레이드 계획](#)"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Sun Cluster 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Sun Cluster에 대한 일반적인 내용을 설명합니다.

호환성 문제

Java ES 릴리스 5 Sun Cluster 소프트웨어는 새로운 그래픽 관리 인터페이스를 포함하지만 이전 버전의 Sun Cluster 에이전트와 호환됩니다.

Sun Cluster 종속성

다른 Java ES 구성 요소에 대한 Sun Cluster 종속성은 Sun Cluster 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어 Sun Cluster 인터페이스나 기능이 변경되면 Sun Cluster 소프트웨어가 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Sun Cluster는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Sun Cluster 소프트웨어는 특정한 Java ES 공유 구성 요소 (47페이지 표 1-9 참조)에 종속됩니다.
- **데이터 서비스.** Sun Cluster 소프트웨어는 Java ES 제품 구성 요소의 가용성을 향상시킬 수 있는 특정 데이터 서비스(또는 에이전트)를 필요로 합니다. Sun Cluster 환경에서 실행 중인 각 제품 구성 요소에 대해 해당 클러스터 자원을 관리하기 위한 적절한 데이터 서비스가 있어야 합니다. 에이전트 패키지는 보통 Sun Cluster 업그레이드 프로세스의 일부로 업그레이드됩니다.

이중 업그레이드

Sun Cluster 소프트웨어와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"에서 설명한 대로 롤링 업그레이드가 아닌 Sun Cluster 비롤링 업그레이드와 연결하여 수행할 수 있습니다.

자세한 절차는 Sun Cluster Installation Guide

(<http://docs.sun.com/doc/819-0420/6n2rlnnd1?a=view>)의 업그레이드 장에 설명되어 있습니다. 릴리스 5 Sun Cluster에 대한 수정 사항은 Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions(<http://docs.sun.com/doc/819-4351>)에 설명되어 있습니다.

이 절차는 Solaris 운영 체제를 Solaris 8 또는 Solaris 9에서 Solaris 10으로 업그레이드하는 작업에 적용됩니다.

Java ES 릴리스 5로 Sun Cluster 소프트웨어 업그레이드

이 절에는 Sun Cluster 소프트웨어를 Java ES 2005Q4(릴리스 4), Java ES 2005Q1(릴리스 3) 및 Java ES 2004Q2(릴리스 2)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 업그레이드 절차는 Java ES 릴리스의 세 Sun Cluster 버전, 즉 Sun Cluster 3.1 4/04, Sun Cluster 3.1 9/04 및 Sun Cluster 3.1 8/05 소프트웨어에서 동일합니다.

Sun Cluster 3.1 8/05(릴리스 3 또는 릴리스 4) 소프트웨어를 설치한 경우 Sun Cluster 소프트웨어에서 공유 구성 요소를 사용하지 않으면 릴리스 5로 업그레이드할 필요가 없습니다.

이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [Sun Cluster 업그레이드](#)

소개

Sun Cluster 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** scinstall 스크립트를 실행하여 업그레이드를 수행합니다. 이 스크립트는 Sun Cluster 소프트웨어를 업그레이드를 완료한 후 이전 Sun Cluster 구성을 적용합니다. 그러나 클러스터 환경의 모든 노드를 같은 버전으로 업그레이드해야 하며 이를 위해서는 클러스터를 종료한 후 모든 노드를 업그레이드하거나 롤링 업그레이드를 사용하여 클러스터를 종료하지 않고 한 번에 한 개씩 노드를 업그레이드할 수 있습니다.
- **업그레이드 종속성.** Sun Cluster 소프트웨어는 여러 Java ES 공유 구성 요소에 종속되지만([47페이지 표 1-9](#) 참조), 이것은 엄격하지 않은 업그레이드 종속성이며 Java ES 릴리스 5 Sun Cluster 소프트웨어는 이러한 구성 요소의 릴리스 4 버전과 호환됩니다.
- **역방향 호환성.** 릴리스 5 Sun Cluster (프레임워크) 소프트웨어는 이전 버전의 클라이언트 에이전트와 호환되며 릴리스 5 Sun Cluster 에이전트 소프트웨어는 Java ES 릴리스 4 구성 요소와 호환됩니다. 그러나 클러스터의 모든 노드는 같은 버전의 프레임워크 및 에이전트 소프트웨어를 실행해야 합니다.

- **업그레이드 롤백.** Sun Cluster 소프트웨어의 릴리스 5 업그레이드를 이전 버전으로 롤백하는 작업은 지원되지 않습니다.
- **플랫폼 문제.** Sun Cluster 소프트웨어를 업그레이드하는 방법은 모든 Solaris 플랫폼 및 하드웨어 구조에서 동일하지만 Linux 플랫폼에서는 Sun Cluster 소프트웨어가 지원되지 않습니다.

Sun Cluster 업그레이드

이 절에서는 Java ES 릴리스 4에서 Java ES 릴리스 5로 Sun Cluster 소프트웨어 업그레이드를 수행하는 방법에 대한 개요를 제공합니다.

- 사전 업그레이드 작업
- Sun Cluster 소프트웨어 업그레이드
- 업그레이드 확인
- 사후 업그레이드 작업
- 업그레이드 롤백

이 절은 비롤링 Sun Cluster 업그레이드에 대해 다룹니다. 롤링 업그레이드가 약간 다른 경우 해당 클러스터는 종료되지 않습니다. 그러나 아래 설명된 대로 두 경우에 모두 주어진 클러스터 노드에 대해 동일한 일반 절차를 따릅니다. 해당 절차는 Sun Cluster Installation Guide(<http://docs.sun.com/doc/819-0420/6n2rlmncr?a=view>)의 업그레이드 장에 설명되어 있습니다. 릴리스 5 Sun Cluster에 대한 수정 사항은 Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions (<http://docs.sun.com/doc/819-4351>)에 설명되어 있습니다.

사전 업그레이드 작업

Sun Cluster 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- 현재 버전 정보 확인
- Sun Cluster Geographic Edition 인프라 종료
- 업그레이드를 위한 클러스터 노드 준비
- 운영 체제 업그레이드
- Sun Cluster 종속성 업그레이드
- 필수 구성 정보 및 비밀번호 얻기

현재 버전 정보 확인

다음 명령을 입력하여 Sun Cluster 소프트웨어의 현재 버전을 확인할 수 있습니다.

```
scinstall -pv
```

이 명령은 Sun Cluster 버전과 설치된 각 소프트웨어 패키지의 버전을 반환합니다. 이 명령이 3.1 8/05 버전(3.1u4)을 반환하면 다음 표에서 패치 개정 번호를 확인하여 릴리스 4 또는 릴리스 5 소프트웨어가 있는지 확인하십시오.

표 3-3 Sun Cluster 버전 확인 출력

Java ES 릴리스	Sun Cluster 버전 번호	패치 개정 번호
릴리스 1(Sun Cluster 3.1)	3.1	
릴리스 2(Sun Cluster 3.1 4/04)	3.1u2	
릴리스 2 또는 3(Sun Cluster 3.1 9/04)	3.1u3	
릴리스 3 또는 4(Sun Cluster 3.1 8/05)	3.1u4	Solaris 9 sparc: 117949-15 Solaris 8 sparc: 117950-15 Solaris 9 x86: 117909-15
릴리스 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. 패치 개정 번호는 여러 플랫폼에서 Java ES 릴리스 5로 업그레이드하기 위해 필요한 최소한의 요구 사항입니다. 새로운 개정 버전이 있는 경우 해당 버전을 사용하십시오.

Sun Cluster Geographic Edition 인프라 종료

Sun Cluster Geographic Edition 소프트웨어를 실행하는 클러스터를 업그레이드하는 경우 먼저 Sun Cluster Geographic Edition 인프라를 종료하고 업그레이드할 클러스터를 준비하는 다른 단계를 수행해야 합니다. Sun Cluster Geographic Edition Installation Guide(<http://docs.sun.com/doc/819-8004/6n9tmd19d?=view>)의 업그레이드 절차를 따르십시오. 이 절차에는 해당 업그레이드 단계에서 Sun Cluster 소프트웨어를 업그레이드하는 단계가 포함되어 있습니다.

업그레이드를 위한 클러스터 노드 준비

Sun Cluster 소프트웨어를 업그레이드하기 전에 클러스터 환경에서 클러스터 노드를 제거해야 합니다.

- **비클러스터 업그레이드.** 클러스터 환경에서 노드 제거는 오프라인 상태로 자원 그룹 전환 및 비활성화, 해당 환경에서 실행 중인 응용 프로그램의 종료, 공유 데이터 백업, 클러스터 종료, 시스템 디스크 백업 및 비클러스터 모드로의 노드 재부팅과 같은 환경의 종료를 의미합니다.
- **클러스터 업그레이드.** 클러스터 환경에서 노드 제거는 노드에서 모든 자원 그룹 및 장치 그룹 제거, 공유 데이터 및 시스템 디스크 백업 그리고 비클러스터 모드로의 노드의 재부팅을 의미합니다.

이러한 작업 및 특정 상황에서 수행하는 데 필요할 수 있는 기타 작업에 대한 내용은 Sun Cluster Installation Guide(<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>)의 업그레이드 장에 설명되어 있습니다.

운영 체제 업그레이드

사용 중인 운영 체제를 최신 버전으로 업그레이드하고 볼륨 관리자의 버전을 업그레이드하기 위해 업그레이드 중단 시간을 사용할 수도 있습니다.

이 작업에 대한 자세한 내용은 Sun Cluster Installation Guide (<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>)의 업그레이드 장에서 설명되어 있습니다.

Sun Cluster 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 릴리스 5로 업그레이드하는 것이 좋습니다. Sun Cluster 소프트웨어가 종속되어 있는 릴리스 4 공유 구성 요소를 업그레이드하는 것은 선택 사항이며 **2장, "Java ES 공유 구성 요소 업그레이드"**의 설명에 따라 업그레이드할 수 있습니다. 하지만 릴리스 2 공유 구성 요소를 릴리스 5로 업그레이드하는 것은 필수입니다.

주	Java ES 공유 구성 요소를 릴리스 5로 동기화한 경우 Sun Cluster 소프트웨어를 릴리스 5로 업그레이드해야 합니다. 이는 릴리스 4 Sun Cluster가 릴리스 5 Sun Java Web Console 공유 구성 요소와 호환되지 않기 때문입니다. Sun Java Web Console이 릴리스 5(버전 3.0)로 업그레이드된 경우 이 장에 설명된 대로 최신 Sun Cluster 패치를 적용하거나 Sun Cluster를 업그레이드해야 합니다.

필수 구성 정보 및 비밀번호 얻기

현재 설치된 버전에 대한 특수 정보가 필요하지 않습니다. 그러나 업그레이드를 수행하려면 슈퍼유저로 로그인해야 합니다.

Sun Cluster 소프트웨어 업그레이드

이 절에서는 Sun Cluster 소프트웨어의 업그레이드 절차에 영향을 미치는 고려 사항을 논의하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

Sun Cluster 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- Sun Cluster 프레임워크 소프트웨어를 업그레이드할 때는 고가용성 Java ES 구성 요소 및 클러스터 환경에서 실행 중인 기타 응용 프로그램을 관리하는 데 필요한 데이터 서비스를 업그레이드하는 것이 좋습니다.
- Sun Cluster 소프트웨어 업그레이드는 또한 클러스터 환경에서 실행 중인 Java ES 구성 요소 또는 기타 응용 프로그램을 업그레이드할 수 있는 기회를 제공합니다.

업그레이드 절차

다음 절차는 각 클러스터 노드에서 Sun Cluster 소프트웨어를 업그레이드하는 데 적용됩니다. 다음은 아주 일반적인 절차이며 각 단계를 수행하는 방법에 대한 자세한 내용은 Sun Cluster Installation Guide

(<http://docs.sun.com/doc/819-0420/6n2rlnncr?a=view>)의 업그레이드 장에 설명되어 있습니다. 또한 릴리스 5 Sun Cluster에 대한 수정 사항은 Sun Cluster 3.1 8/05 With Java Enterprise System 5 Special Instructions(<http://docs.sun.com/doc/819-4351>)에 설명되어 있습니다.

1. 비클러스터 모드로 부팅합니다.
2. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

3. Java ES 릴리스 5 배포의 다음 디렉토리로 변경합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당하며 *os-ver*은 Solaris 9 또는 Solaris 10입니다.

4. scinstall 유틸리티를 실행합니다.

```
./scinstall
```

클러스터 설치, 구성 및 업그레이드 작업을 수행하기 위한 주 메뉴가 표시됩니다.

5. Sun Cluster 프레임워크 소프트웨어 및 원하는 데이터 서비스를 업그레이드합니다.

해당 자원을 업그레이드된 자원 유형으로 마이그레이션하여 업그레이드된 데이터 서비스를 구성해야 합니다(85페이지의 "사후 업그레이드 작업" 참조).

6. Sun Cluster 프레임워크 소프트웨어 및 데이터 서비스에 필요한 패치를 적용합니다.

관련 패치에 액세스 및 적용하는 방법에 대한 자세한 내용은 Sun Cluster 3.0-3.1 Release Notes Supplement

(<http://docs.sun.com/app/docs/doc/816-3381/6m9lratq9?a=view#gcpom>)에 설명되어 있습니다.

7. 노드를 클러스터로 다시 부팅합니다.

업그레이드 확인

다음과 같이 Sun Cluster 소프트웨어의 업그레이드 성공 여부를 확인할 수 있습니다.

1. 다음 명령을 입력합니다.

```
scinstall -pv
```

이 명령은 Sun Cluster 버전과 설치된 각 소프트웨어 패키지의 버전을 반환합니다. 이 명령이 3.1 8/05 버전(3.1u4)을 반환하면 82페이지 표 3-3에서 패치 개정 번호를 확인하여 릴리스 4 또는 릴리스 5 소프트웨어가 있는지 확인하십시오.

2. 데이터 서비스 업그레이드 로그 파일을 확인합니다.

업그레이드 출력 메시지의 끝에서 이 로그 파일을 알려 줍니다.

사후 업그레이드 작업

Sun Cluster 소프트웨어의 업그레이드를 수행한 후에는 비롤링 또는 롤링 업그레이드를 수행했는지에 따라 여러 추가 작업을 수행해야 할 수도 있습니다. 작업 중 클러스터 환경을 완전히 복원해야 하는 것은 다음과 같습니다.

- 클러스터 구성 상태 확인
- 자원을 새 자원 유형 버전으로 마이그레이션
- 클러스터에 설치된 추가 Java ES 구성 요소나 응용 프로그램 업그레이드

이러한 사후 설치 절차에 대한 자세한 내용은 Sun Cluster Installation Guide

(<http://docs.sun.com/doc/819-0420/6n2rlnnncr?a=view>)의 업그레이드 장에 설명되어 있습니다.

업그레이드 롤백

Sun Cluster 소프트웨어 롤백은 지원되지 않습니다. 업그레이드 절차에서 변경한 사항은 쉽게 취소할 수 없습니다.

Sun Cluster Geographic Edition

이 장에서는 Sun Cluster Geographic Edition을 Java ES 5(릴리스 5): Sun Cluster Geographic Edition 3.1 2006Q4로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 방법에 대한 업그레이드 고려 사항의 개요를 제공하며, 이 장에서는 Solaris 운영 체제에서의 업그레이드에 대해서만 설명합니다.

- 88페이지의 "Sun Cluster Geographic Edition 업그레이드 개요"
- 91페이지의 "버전 3.1 8/05에서 Sun Cluster Geographic Edition 업그레이드"

주 이 장에 설명된 파일 위치는 *SunClusterGeo-base*라는 디렉토리 경로에 상대적으로 지정되어 있습니다. Sun Cluster Geographic Edition을 설치할 때 최소한 이 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Geographic Edition 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이 디렉토리 경로의 기본값을 보여 줍니다.

표 4-1 Sun Cluster Geographic Edition 디렉토리 경로

경로 이름 변수	Solaris OS	Linux OS
<i>SunClusterGeo-base</i>	/opt/SUNWscgeo	적용할 수 없음

Sun Cluster Geographic Edition

업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Sun Cluster Geographic Edition의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 정보](#)
- [Java ES 릴리스 5 업그레이드 로드맵](#)
- [Sun Cluster Geographic Edition 데이터](#)
- [Sun Cluster Geographic Edition 업그레이드 전략](#)

Java ES 릴리스 5 정보

Java ES 릴리스 5 Sun Cluster Geographic Edition은 Java ES 구성 요소로 제공되는 첫 번째 릴리스입니다. Sun Cluster Geographic Edition 3.1은 독립 실행형 제품으로 처음 릴리스되었습니다.

릴리스 5 Sun Cluster Geographic Edition은 Sun Cluster Geographic Edition 3.1 8/05와 비교할 때 부차적 릴리스이며 Solaris x86 플랫폼을 지원하는 첫 번째 릴리스입니다. 또한 릴리스 5 Sun Cluster Geographic Edition에는 몇 가지 선택된 버그 수정, 다양한 하드웨어 및 소프트웨어 구성 요소에 대한 지원, 그리고 추가 데이터 복제 제품에 대한 지원이 포함됩니다.

Java ES 릴리스 5 업그레이드 로드맵

[표 4-2](#)는 Sun Cluster Geographic Edition을 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 운영 체제에만 적용됩니다.

표 4-2 Java ES 5(릴리스 5) 업그레이드 경로: Sun Cluster Geographic Edition 3.1 2006Q4

Java ES 릴리스	Sun Cluster Geographic Edition 버전	일반적인 방법	재구성 필요
이전 Java ES 릴리스	Sun Cluster Geographic Edition 3.1 8/05	직접 업그레이드: 새로 설치하여 Sun Cluster Geographic Edition 3.1 8/05 대체	없음

Sun Cluster Geographic Edition 데이터

다음 표는 Sun Cluster Geographic Edition 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 4-3 Sun Cluster Geographic Edition 데이터 사용

데이터 유형	위치	사용
Sun Cluster Geographic Edition 구성 데이터	모든 클러스터 노드에서 복제되고 동기화된 클러스터 구성 저장소(주의: 노드 또는 전체 클러스터가 작동을 중지할 수도 있으므로 CCR 파일은 절대로 수동으로 편집하지 마십시오.)	Sun Cluster Geographic Edition 작업의 모든 측면에 대한 구성 정보를 저장합니다.

Sun Cluster Geographic Edition 업그레이드 전략

Sun Cluster Geographic Edition 업그레이드를 위한 전략은 일반적으로 1장, "[업그레이드 계획](#)"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Sun Cluster Geographic Edition 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Sun Cluster Geographic Edition에 대한 일반적인 내용을 설명합니다.

호환성 문제

파트너 관계가 있는 두 클러스터에서는 동일한 버전의 Sun Cluster Geographic Edition을 실행해야 합니다. 따라서 릴리스 5 Sun Cluster Geographic Edition은 Sun Cluster Geographic Edition 3.1 8/05와 혼합하여 사용할 수 없습니다(역방향 호환성이 제공되지 않음). 릴리스 5 Sun Cluster Geographic Edition에는 Sun Cluster Geographic Edition 3.1 8/05에서 읽을 수 없는 추가 구성 데이터가 포함되어 있습니다.

종속성

다른 Java ES 구성 요소에 대한 Sun Cluster Geographic Edition 종속성은 Sun Cluster Geographic Edition 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어, Sun Cluster Geographic Edition 인터페이스 또는 기능의 변경 사항을 적용하려면 Sun Cluster Geographic Edition이 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Sun Cluster Geographic Edition은 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Sun Cluster Geographic Edition은 특정한 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속됩니다.
- **Sun Cluster.** Sun Cluster Geographic Edition은 기본 기능을 제공하는 Sun Cluster에 대해 필수 종속성이 있습니다.

이중 업그레이드

Sun Cluster Geographic Edition 소프트웨어와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 Sun Cluster Geographic Edition에 적용되지 않습니다.

Sun Cluster Geographic Edition은 Sun Cluster에 대한 엄격한 종속성이 있기 때문에 운영 체제 업그레이드는 Sun Cluster 업그레이드와 연결하여 수행됩니다. Sun Cluster 이중 업그레이드에 대한 자세한 내용은 [79페이지의 "이중 업그레이드"](#)를 참조하십시오.

버전 3.1 8/05에서 Sun Cluster Geographic Edition 업그레이드

이 절에서는 Sun Cluster Geographic Edition을 버전 3.1 8/05에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [버전 3.1 8/05 Sun Cluster Geographic Edition 업그레이드](#)

소개

Sun Cluster Geographic Edition 3.1 8/05를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 버전 3.1 8/05를 제거하고 릴리스 5 Sun Cluster Geographic Edition을 새로 설치하여 업그레이드를 수행합니다. 구성 데이터는 그대로 유지됩니다.
- **업그레이드 종속성.** Sun Cluster Geographic Edition은 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Sun Cluster Geographic Edition을 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다. 특히 Sun Cluster Geographic Edition은 공용 에이전트 컨테이너 버전 1.1에 대해 엄격한 업그레이드 종속성이 있습니다. 또한 Sun Cluster Geographic Edition은 Sun Cluster에 대해서도 엄격한 업그레이드 종속성이 있으며, 릴리스 5 Sun Cluster Geographic Edition을 지원하려면 릴리스 4 Sun Cluster(3.1 8/05)를 릴리스 5로 업그레이드해야 합니다.
- **역방향 호환성.** 릴리스 5 Sun Cluster Geographic Edition은 Sun Cluster Geographic Edition 3.1 8/05와 호환되지 않습니다. 파트너 관계에 있는 모든 클러스터를 릴리스 5로 업그레이드해야 합니다.
- **업그레이드 롤백.** Sun Cluster Geographic Edition 소프트웨어의 릴리스 5 업그레이드를 버전 3.1 8/05로 롤백하는 작업은 지원되지 않습니다.
- **플랫폼 문제.** Sun Cluster Geographic Edition 소프트웨어를 업그레이드하는 방법은 모든 Solaris 플랫폼 및 하드웨어 아키텍처에서 동일하지만 Linux 플랫폼에서는 Sun Cluster Geographic Edition 소프트웨어가 지원되지 않습니다.

버전 3.1 8/05 Sun Cluster Geographic Edition 업그레이드

이 절에서는 버전 3.1 8/05에서 Java ES 릴리스 5로 Sun Cluster Geographic Edition을 업그레이드하는 방법에 대해 설명하며 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [3.1 8/05 Sun Cluster Geographic Edition 업그레이드](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Sun Cluster Geographic Edition 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [업그레이드를 위한 클러스터 준비](#)
- [Sun Cluster Geographic Edition 종속성 업그레이드](#)
- [Sun Cluster Geographic Edition 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

다음 명령을 사용하여 Sun Cluster Geographic Edition의 현재 버전을 확인할 수 있습니다.

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

저작권 텍스트 뒤에 다음 표에 나와 있는 버전 문자열이 표시됩니다.

표 4-4 Sun Cluster Geographic Edition 버전 확인 출력

Java ES 릴리스	Sun Cluster Geographic Edition 버전 번호
버전 3.1 8/05	1.0
릴리스 5	1.1

업그레이드를 위한 클러스터 준비

업그레이드하려는 클러스터와 파트너 관계가 있는 모든 클러스터에서 다음 단계를 수행하십시오. 이 절차는 프로덕션에서 Sun Cluster Geographic Edition 계층을 제거합니다.

1. 클러스가 올바르게 작동하고 있는지 확인합니다.

클러스터의 현재 상태를 보려면 아무 노드에서나 다음 명령을 실행합니다.

```
scstat
```

2. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

3. 보호 그룹에서 모든 응용 프로그램 자원 그룹을 제거합니다.

고가용성 응용 프로그램에서는 Sun Cluster Geographic Edition 업그레이드 중에 시스템 중단 시간이 발생하지 않습니다.

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

4. 이 클러스터와 파트너 관계가 있는 모든 클러스터에서 [단계 1](#)부터 [단계 3](#)까지 수행합니다.

5. 클러스터에서 활성화되어 있는 모든 보호 그룹을 중지합니다.

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

6. 양쪽 파트너 클러스터의 모든 하트비트에서 ICRM 플러그인을 제거합니다.

```
geohp remove-plugin icrmplugin heartbeatname
```

Sun Cluster Geographic Edition 종속성 업그레이드

일반적으로 컴퓨터 시스템 (및 컴퓨팅 환경 내)의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Sun Cluster Geographic Edition은 여러 공유 구성 요소 및 Sun Cluster에 대한 엄격한 업그레이드 종속성이 있습니다.

Sun Cluster Geographic Edition 종속성을 업그레이드할 때는 Sun Cluster Geographic Edition을 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다. 공유 구성 요소 업그레이드는 일반적으로 Java ES 설치 프로그램을 통해 자동으로 수행됩니다.

1. **공유 구성 요소.** 63페이지의 "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Sun Cluster Geographic Edition에 필요한 모든 공유 구성 요소는 Sun Cluster Geographic Edition을 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.

주	Java ES 공유 구성 요소를 릴리스 5로 업그레이드한 경우 Sun Cluster Geographic Edition 소프트웨어도 릴리스 5로 업그레이드해야 합니다. 이는 Sun Cluster Geographic Edition 3.1 8/05가 릴리스 5 Sun Java Web Console과 호환되지 않기 때문입니다.

2. **Sun Cluster.** 75페이지의 3장, "[Sun Cluster 소프트웨어](#)"에서 Sun Cluster를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

Sun Cluster Geographic Edition 데이터 백업

Sun Cluster Geographic Edition은 모든 데이터를 클러스터 구성 저장소에 저장하므로 현재 데이터를 백업할 필요가 없습니다.

필수 구성 정보 및 비밀번호 얻기

현재 설치된 버전에 대한 특수 정보가 필요하지 않습니다. 그러나 업그레이드를 수행하려면 슈퍼유저로 로그인해야 합니다.

3.1 8/05 Sun Cluster Geographic Edition 업그레이드

이 절에서는 Sun Cluster Geographic Edition의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

실행 중인 응용 프로그램을 방해하지 않고 Sun Cluster Geographic Edition 소프트웨어를 Java ES 릴리스 5로 업그레이드할 수 있습니다. 실행 중인 클러스터에서 클러스터를 중단하지 않고 Sun Cluster Geographic Edition 소프트웨어를 업그레이드할 수 있으며 해당 클러스터는 서비스를 실행하면서 프로덕션 상태를 유지할 수 있습니다. 비슷하게 시스템 중단 시간 없이 Sun Cluster Geographic Edition 패치를 적용할 수 있습니다. 구성 데이터는 업그레이드 과정 전체에서 그대로 유지됩니다.

업그레이드하는 클러스터와 파트너 관계가 있는 모든 클러스터의 모든 노드에서 Sun Cluster Geographic Edition 소프트웨어를 업그레이드해야 합니다.

Sun Cluster Geographic Edition 소프트웨어 업그레이드 프로세스 중에 Solaris 운영 체제를 업그레이드하려면 Solaris 운영 체제를 업그레이드하기 전에 Sun Cluster Geographic Edition 패키지를 제거해야 합니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 로컬에 상주하는 Sun Cluster Geographic Edition 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. Sun Cluster Geographic Edition 3.1 8/05를 종료합니다.

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

Sun Cluster Geographic Edition을 종료하는 동안 기본 응용 프로그램을 계속 실행하려면 Sun Cluster Geographic Edition System Administration Guide (<http://docs.sun.com/doc/819-8003>)를 참조하십시오.

3. Sun Cluster Geographic Edition 3.1 8/05를 제거합니다.

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

Sun Cluster Geographic Edition Installation Guide

(<http://docs.sun.com/doc/819-8004>)에 설명된 대로 Sun Cluster Geographic Edition 설치 프로그램을 제거 모드에서 사용합니다.

4. Sun Cluster가 실행되고 있지 않으면 시작합니다.

Sun Cluster Geographic Edition을 설치하려면 Sun Cluster가 실행되고 있어야 합니다. `scstat` 명령을 사용하여 확인할 수 있습니다.

5. Java ES 설치 프로그램을 사용하여 릴리스 5 Sun Cluster Geographic Edition을 새로 설치합니다.

- a. 릴리스 4 Access Manager를 호스팅하는 컴퓨터에서 Java ES 설치 프로그램을 실행합니다.

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

여기서 `os_arch`는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령 줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. 구성 요소 선택 페이지에서 Sun Cluster Geographic Edition을 선택합니다.

- c. "지금 구성" 또는 "나중에 구성"을 선택합니다.
재구성할 필요가 없기 때문에 지금 구성 또는 나중에 구성 중 어떤 것을 선택해도 차이가 없습니다.
- d. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.
- e. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.

주	Sun Cluster Geographic Edition 소프트웨어를 Solaris 8 OS에서 업그레이드하는 경우 pkgadd 명령을 사용하여 Java ES 릴리스 5 배포에서 소프트웨어를 설치해야 합니다. Java ES 설치 프로그램은 Solaris 8 OS에서 Sun Cluster Geographic Edition 소프트웨어 설치를 지원하지 않습니다.

- 6. 필요한 모든 Sun Cluster Geographic Edition 패치를 설치합니다.
자세한 내용은 Sun Cluster Geographic Edition Installation Guide (<http://docs.sun.com/doc/819-8004>)를 참조하십시오.
- 7. 이 클러스터와 파트너 관계가 있는 모든 클러스터에서 [95페이지의 단계 1](#)부터 [단계 6](#)까지 수행합니다.
- 8. 릴리스 5 Sun Cluster Geographic Edition 소프트웨어를 다시 시작합니다.

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


클러스터가 파트너 관계에 있는 경우 두 파트너의 모든 노드를 업그레이드해야 Sun Cluster Geographic Edition 소프트웨어를 시작할 수 있습니다.

자세한 내용은 위에서 참조된 Installation Guide와 Sun Cluster Geographic Edition System Administration Guide(<http://docs.sun.com/doc/819-8003>)를 참조하십시오.
- 9. [93페이지의 "업그레이드를 위한 클러스터 준비"](#)에서 설명한 대로 클러스터 업그레이드를 준비할 때 제거한 모든 응용 프로그램 자원 그룹을 추가합니다.

```
geopg add-resource-group resourcegroup protectiongroupname
```
- 10. 추가한 모든 보호 그룹을 시작합니다.

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


업그레이드 확인

다음과 같이 Sun Cluster Geographic Edition 소프트웨어의 업그레이드 성공 여부를 확인할 수 있습니다.

1. 다음 명령을 실행합니다.

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

출력 값에 대해서는 [92페이지 표 4-4](#)를 참조하십시오.

2. 적용 가능한 경우 두 파트너를 모두 업그레이드하고 다음 명령을 각 파트너의 한 클러스터 노드에서 실행합니다.

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

이 명령은 Sun Cluster Geographic Edition 소프트웨어가 해당 노드에서 활성화되어 있는지를 보여 줍니다.

사후 업그레이드 작업

사후 업그레이드 작업은 [95페이지의 "업그레이드 절차"](#)에 설명된 단계에 모두 포함됩니다.

업그레이드 롤백

Sun Cluster Geographic Edition 소프트웨어 롤백은 지원되지 않습니다.

그러나 Java ES 설치 프로그램의 제거 기능을 사용하여 릴리스 5 Sun Cluster Geographic Edition을 제거한 후 버전 3.1 8/05 설치 프로그램을 사용하여 버전 3.1 8/05를 다시 설치할 수 있습니다. 릴리스 5 Sun Cluster Geographic Edition은 버전 3.1 8/05에서 읽을 수 없는 데이터를 기록하므로 이러한 롤백은 릴리스 5를 시작하기 전, 즉 [95페이지의 "업그레이드 절차"](#)에서 설명한 대로 업그레이드된 Sun Cluster Geographic Edition 소프트웨어를 시작하기 전에 수행해야 합니다.

Directory Server

이 장에서는 Directory Server 를 Java ES 5(릴리스 5): Sun Java System Directory Server 6.0으로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- 100페이지의 "Directory Server 업그레이드 개요"
- 105페이지의 "Java ES 릴리스 4에서 Directory Server 업그레이드"
- 113페이지의 "Java ES 릴리스 3에서 Directory Server 업그레이드"
- 114페이지의 "Java ES 릴리스 2에서 Directory Server 업그레이드"

주 이 장에 설명된 파일 위치는 *serverRoot*(Directory Server 5.x) 및 *DirServer-base*(Directory Server 6.0)라는 디렉토리 경로에 대해 상대적 으로 지정됩니다. Directory Server를 처음 설치할 때 최소한 이 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 5-1 Directory Server 디렉토리 경로

경로 이름 변수	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 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Directory Server의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 정보](#)
- [Java ES 릴리스 5 업그레이드 로드맵](#)
- [Directory Server 데이터](#)
- [Directory Server 업그레이드 전략](#)

Java ES 릴리스 5 정보

Java ES 릴리스 5 Directory Server는 다양한 새 기능과 향상된 다양한 기능을 포함하는 주요 릴리스입니다. 자세한 내용은 Directory Server Enterprise Edition 6 릴리스 노트(<http://docs.sun.com/doc/820-0288>)를 참조하십시오.

Java ES 릴리스 5 업그레이드 로드맵

[표 5-2](#)는 Directory Server를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 5-2 Java ES 5(릴리스 5) 업그레이드 경로: Directory Server 6.0

Java ES 릴리스	Directory Server 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Directory Server 5.2 2005Q4 Sun Java System Administration Server 5.2 2005Q4	직접 업그레이드: 새로 설치하고 모든 데이터를 마이그레이션합니다. Administration Server 기능은 Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티로 대체됩니다.	이전 버전에서 새로 설치된 Directory Server 로 마이그레이션된 구성 데이터
릴리스 3	Sun Java System Directory Server 5 2005Q1 Sun Java System Administration Server 5 2005Q1	직접 업그레이드: 새로 설치하고 모든 데이터를 마이그레이션합니다. Administration Server 기능은 Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티로 대체됩니다.	이전 버전에서 새로 설치된 Directory Server 로 마이그레이션된 구성 데이터

표 5-2 Java ES 5(릴리스 5) 업그레이드 경로: Directory Server 6.0

Java ES 릴리스	Directory Server 버전	일반적인 방법	재구성 필요
릴리스 2	Sun Java System Directory Server 5.2 2004Q2 Sun Java System Administration Server 5.2 2004Q2	직접 업그레이드: 새로 설치하고 모든 데이터를 마이그레이션합니다. Administration Server 기능은 Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티로 대체됩니다.	이전 버전에서 새로 설치된 Directory Server로 마이그레이션된 구성 데이터
릴리스 1	Sun ONE Directory Server 5.2 Sun ONE Administration Server 5.2	직접 업그레이드 인증 안 됨: 그러나 릴리스 2에서 업그레이드할 때와 동일한 방법을 사용할 수 있습니다.	이전 버전에서 새로 설치된 Directory Server로 마이그레이션된 구성 데이터
이전 Java ES 릴리스	Sun ONE Directory Server 5.2 Sun ONE Administration Server 5.2 Sun ONE Directory Server 5.1 Sun ONE Administration Server 5.1	직접 업그레이드 인증 안 됨: 그러나 릴리스 2에서 업그레이드할 때와 동일한 방법을 사용할 수 있습니다. 직접 업그레이드 불가능: 먼저 릴리스 3으로 업그레이드합니다. Java Enterprise System 2005Q1 Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)를 참조하십시오. 그런 다음 릴리스 3에서 릴리스 5로 업그레이드합니다.	이전 버전에서 새로 설치된 Directory Server로 마이그레이션된 구성 데이터 Java Enterprise System 2005Q1 Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)를 참조하십시오.

Directory Server 데이터

Directory Server 5.x 버전은 Directory Server 자체를 사용하여 구성 데이터를 저장합니다. 구성 데이터는 디렉토리의 특정 트리 구조에 저장됩니다. 해당 구성을 호스팅하는 Directory Server 인스턴스를 구성 디렉토리라고 합니다. 이 구성 디렉토리는 다른 Directory Server 인스턴스와 동일한 컴퓨터에 있을 수 있지만 대부분의 배포 구조에서 구성 디렉토리는 이를 사용해 구성 정보를 저장하는 다른 구성 요소와 분리되어 있습니다.

Directory Server 6.0에서는 더 이상 구성 디렉토리에 구성 데이터를 저장하지 않습니다. Directory Service Control Center(또는 Directory Server EE 명령줄 유틸리티)를 사용하여 구성이 수행되며 이 인터페이스를 통해 액세스해야 합니다. Directory Service Control Center는 자체 로컬 Directory Server 인스턴스에 구성 데이터를 저장합니다.

다음 표에서는 Directory Server 소프트웨어의 릴리스 5 업그레이드에 의해 영향을 받는 데이터의 유형을 보여 줍니다.

표 5-3 Directory Server 데이터 사용

데이터 유형	위치	사용
Directory Server 구성 데이터	Directory Server 5.1 및 5.2: 구성 디렉토리 Directory Server 6.0: Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티를 사용하여 액세스	Directory Server 인스턴스 구성
Directory Server 스키마		디렉토리에서 데이터의 구조와 의미를 정의합니다.
보안 데이터	Directory Server 5.1 및 5.2: Directory Server 콘솔을 통해 SSL 구성 Directory Server 6.0: Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티를 통해 SSL 구성	서버 인증서
사용자 데이터	Directory Server	사용자별 구성 데이터 및 사용자 프로필을 사용하여 응용 프로그램 지원

Directory Server 업그레이드 전략

Directory Server 업그레이드를 위한 전략은 일반적으로 [1장, "업그레이드 계획"](#)에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Directory Server 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Directory Server에 대한 일반적인 내용을 설명합니다.

호환성 문제

Java ES 릴리스 5 Directory Server에는 새 공용 인터페이스가 추가되지 않았으므로 이전 버전과 호환되며 릴리스 4 Directory Server 및 이전 버전에서 지원하던 모든 구성 요소를 지원합니다.

그러나 릴리스 5는 전용 관리 인터페이스가 변경되었습니다. 릴리스 5 인터페이스는 이전 Directory Server 릴리스와 호환되지 않습니다. 특히 이전 Directory Server 인스턴스를 구성하는 데 사용된 Administration Server가 Directory Service Control Center 및 Directory Server EE 유틸리티로 대체되었으며 Directory Server 구성 정보를 저장하는 데 사용된 o=NetscapeRoot 디렉토리 접미어는 제거되었습니다. 자세한 내용은 Directory Server Enterprise Edition 6 Migration Guide (<http://docs.sun.com/doc/819-0994>)를 참조하십시오.

종속성

다른 Java ES 구성 요소에 대한 종속성은 일반적으로 Directory Server 소프트웨어를 업그레이드하는 절차에 영향을 줄 수 있습니다.

Directory Server는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Directory Server는 특정 Java ES 공유 구성 요소에 대해 종속성이 있습니다([47페이지 표 1-9](#) 참조). Directory Server 업그레이드는 이러한 공유 구성 요소의 업그레이드된 버전에 종속됩니다.
- **Directory Proxy Server.** Directory Server는 LDAP 요청에 대한 향상된 보안과 성능을 제공하는 Directory Proxy Server에 대해 상호 종속성이 있습니다.

이중 업그레이드

Directory Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 다음 두 가지 방법 중 하나로 수행할 수 있습니다.

- 운영 체제 새로 설치
- 기존 운영 체제 업그레이드

운영 체제 새로 설치

1. 기존 Directory Server 데이터를 백업합니다.

Directory Server 5.x 정보에 대해서는 [107페이지의 "Directory Server 이미지 만들기\(선택 사항\)"](#)를 참조하십시오.

2. 새 운영 체제를 설치합니다.

운영 체제는 새 시스템 (또는 Solaris 10 영역)에 설치하거나 기존 파일 시스템을 덮어쓸 수 있습니다.

3. [단계 1](#)에서 백업한 Directory Server 데이터를 복원합니다.
4. 릴리스 5 Directory Server를 설치합니다.

5. 릴리스 5 Directory Server 인스턴스를 만들고 디렉토리 데이터를 새 인스턴스로 마이그레이션합니다.

[107페이지의 "릴리스 4 업그레이드 Directory Server"](#)의 절차에서 관련 단계를 참조하십시오.

기존 운영 체제 업그레이드

1. 기존 Directory Server 데이터를 백업합니다.

Directory Server 5.x 정보에 대해서는 [107페이지의 "Directory Server 이미지 만들기\(선택 사항\)"](#)를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Directory Server로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 관련 절을 참조하십시오.

Java ES 릴리스 4에서 Directory Server 업그레이드

이 절에는 Directory Server를 Java ES 2005 Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Directory Server 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Directory Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 릴리스 5 Directory Server를 새로 설치한 다음 마이그레이션 도구를 사용하여 이전 Directory Server 인스턴스를 새로운 개별 릴리스 5 Directory Server 인스턴스로 다시 만들어 업그레이드를 수행할 수 있습니다.
- **업그레이드 종속성.** Directory Server는 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Directory Server를 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다.
- **역방향 호환성.** 릴리스 5 Directory Server는 [102페이지의 "호환성 문제"](#)에서 설명한 대로 릴리스 4 버전과 호환되지 않습니다. 그러나 Directory Server 5.x 관리 프레임워크에 종속되는 Directory Server 인스턴스 집합을 계속 유지 관리하는 경우 마이그레이션 도구를 사용하여 o=NescapeRoot 접미어를 마이그레이션할 수 있습니다.
- **업그레이드 롤백.** 릴리스 5로 업그레이드한 다음에도 수정되지 않는 이전 버전으로 되돌리면 릴리스 5 업그레이드를 롤백할 수 있습니다.
- **플랫폼 문제.** Directory Server를 업그레이드하는 일반적인 방법은 Solaris 및 Linux 운영 체제에서 동일합니다.

릴리스 4 Directory Server 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Directory Server를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Directory Server](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Directory Server 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Directory Server 종속성 업그레이드](#)
- [필수 구성 정보 및 비밀번호 얻기](#)
- [Directory Server 이미지 만들기\(선택 사항\)](#)

현재 버전 정보 확인

-v 옵션을 사용하여 Directory Server 데몬을 다시 시작하면 Directory Server의 현재 버전을 확인할 수 있습니다.

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

표 5-4 Directory Server 버전 확인 출력

Java ES 릴리스	Directory Server 버전 번호
릴리스 2	Sun Java(TM) System Directory Server/5.2_Patch_2
릴리스 3	Sun Java(TM) System Directory Server/5.2_Patch_3
릴리스 4	Sun Java(TM) System Directory Server/5.2_Patch_4
릴리스 5	Sun Java(TM) System Directory Server/6.0

Solaris 10 플랫폼에서 ns-slapd 명령이 실패하는 경우 해당 명령을 실행하는 동안 라이브러리 경로를 null로 설정합니다.

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

Directory Server 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. 하지만 Directory Server를 릴리스 5로 업그레이드하면 Directory Server에 필요한 모든 공유 구성 요소가 자동으로 설치됩니다.

필수 구성 정보 및 비밀번호 얻기

현재 설치된 버전에 대한 Directory Server 관리자 아이디와 비밀번호를 알아야 합니다. 다른 구성 정보는 업그레이드 과정에서 보존됩니다.

Directory Server 이미지 만들기(선택 사항)

릴리스 4 버전이 있는 컴퓨터와 다른 컴퓨터에 릴리스 5 Directory Server를 설치하는 경우 릴리스 5 Directory Server를 설치하는 컴퓨터에 릴리스 4 버전의 이미지를 만들어야 합니다. 이 이미지는 새 릴리스 5 Directory Server 인스턴스에 데이터 마이그레이션을 자동화(dsmig 명령 사용)하는 데 필요합니다.

릴리스 4 이미지 파일에는 원본 Directory Server 5.x *serverRoot* 파일 구조와 동일한 레이아웃으로 모든 스키마 파일, 구성 파일, 보안 파일 및 데이터베이스 파일이 포함되어 있습니다. 이미지는 새로운 릴리스 5 Directory Server 인스턴스로 데이터를 마이그레이션하는 데 필요합니다.

릴리스 4 업그레이드 Directory Server

이 절에서는 Directory Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 절차 자체에 대해서도 설명합니다.

업그레이드 고려 사항

Directory Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- Directory Server 인스턴스를 사용하는 모든 Java ES 구성 요소(Access Manager, Portal Server 또는 Sun Java Communications Suite 구성 요소)를 종료하고, 필요한 경우 해당하는 새 릴리스 5 인스턴스에 액세스하도록 다시 구성해야 합니다.
- 한 대의 컴퓨터에서 Directory Server의 여러 인스턴스가 실행되는 배포 구조(동일한 Directory Server 설치 이미지에 해당하는 모든 항목)에서는 Directory Server 이미지를 한 번만 업그레이드해야 하지만 각 인스턴스에 대해서는 별도로 데이터를 마이그레이션해야 합니다.

- 대부분 릴리스 4 Directory Server 배포 구조에서 구성 디렉토리는 별도의 Directory Server 인스턴스입니다. 릴리스 5에서는 구성 디렉토리가 사용되지 않으므로 이러한 인스턴스를 업그레이드할 필요가 없습니다. 하지만 업그레이드를 하게 되면 Directory Server 인스턴스를 원격으로 관리할 별도의 컴퓨터에 릴리스 5 Directory Server 관리 콘솔(Directory Service Control Center)이 배포될 수 있습니다.
- Directory Server에는 스키마, 구성, 보안 및 사용자 데이터의 마이그레이션을 자동화하도록 도와주는 명령줄 도구가 제공됩니다. 마이그레이션 도구를 사용하면 이러한 여러 데이터를 단계별로 마이그레이션할 수 있습니다. 대부분의 업그레이드 시나리오에서는 일부 데이터에 대해서만 마이그레이션을 자동화해도 도움이 됩니다.

업그레이드 절차

아래에서 설명하는 절차는 업그레이드를 수행하는 컴퓨터에 로컬로 상주하는 Directory Server 인스턴스에 적용되거나 대상 컴퓨터에서 실행할 모든 인스턴스를 다른 컴퓨터로 이동하는 경우에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 릴리스 4 Directory Server (5.2) 인스턴스를 종료합니다.

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

오류 로그(`serverRoot/slapd-hostName/logs/errors`)를 참조하여 올바르게 종료되었는지 확인합니다.

```
[23/Jan/2006:15:56:47 +0100] - 모든 데이터베이스 스레드가 중지되었습니다.
```

```
[23/Jan/2006:15:56:50 +0100] - slapd가 중지되었습니다.
```

3. 릴리스 5 Directory Server를 설치할 호스트 컴퓨터에 디스크 공간이 충분한지 확인합니다.

기본적인 계산 방법은 다음과 같습니다.

$2 * (\text{기존 서버의 공간}) + (\text{LDIF 파일에 사용할 공간})$

아쉽게도 내보낸 데이터베이스 파일에서 만들 LDIF 파일의 크기를 예측할 수 있는 도구는 없습니다. 따라서 데이터 항목의 수, 내부 표현 방식, 색인 수 등과 같은 요인에 따라 크기가 달라집니다.

4. 릴리스 5를 원격으로 설치하려면 릴리스 4 이미지를 만들고 이 이미지를 원격 컴퓨터에 전송합니다.

[107페이지의 "Directory Server 이미지 만들기\(선택 사항\)"](#)를 참조하십시오.

5. Directory Server가 엄격한 업그레이드 종속성을 가지는 모든 Java ES 구성 요소를 업그레이드했는지 확인합니다([107페이지의 "Directory Server 종속성 업그레이드"](#) 참조).

6. 릴리스 5 Directory Server를 새로 설치합니다.

다음 단계를 수행합니다.

- a. Java ES 설치 프로그램을 시작합니다.

```
cd Java ES Release 5 distribution/os_arch

./installer
```

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. Directory Server Enterprise Edition의 Directory Server 하위 구성 요소를 선택합니다.

사용하려는 관리 하위 구성 요소(Directory Service 제어판 또는 명령줄 유틸리티)도 설치해야 합니다.

- c. 기존 릴리스 4 Directory Server의 설치 경로와 다른 설치 경로를 지정합니다.

- d. "지금 구성" 또는 "나중에 구성"을 선택합니다.

Directory Server에는 구성이 필요 없기 때문에 "지금 구성"이나 "나중에 구성" 중 어떤 것을 선택해도 차이가 없습니다. 그러나 "지금 구성"을 선택하는 경우에는 새 인스턴스를 만들도록 선택하지 *마십시오*.

- e. 설치 선택 항목을 확인합니다.

Directory Server 패키지가 업그레이드되고 업그레이드 요약이 표시됩니다.

- f. Java ES 설치 프로그램을 종료합니다.

7. Directory Server 인스턴스를 만듭니다.

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

여기서 *instancePath*는 Directory Server 인스턴스의 전체 경로입니다.

Directory Server 인스턴스를 만드는 방법에 대한 자세한 내용은 Directory Server Enterprise Edition 6 관리 설명서(<http://docs.sun.com/doc/820-0292>)를 참조하십시오.

새 인스턴스를 만들지 못하면 dsmig 명령을 사용하여 데이터를 마이그레이션할 때 자동으로 새 인스턴스가 생성됩니다([단계 8](#)).

Solaris 10 플랫폼에서 dsadm 명령이 실패하는 경우 해당 명령을 실행하는 동안 라이브러리 경로를 null로 설정하십시오.

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

8. 릴리스 4 데이터를 릴리스 5 Directory Server 인스턴스로 마이그레이션합니다.

DirServer-base/ds6/bin/dsmig 명령을 사용합니다.

dsmig 명령은 릴리스 4 데이터를 릴리스 5 형식에 맞게 변환하고 적절한 위치에 기록합니다. 예를 들어 Directory Server 인스턴스가 하나 있는 단일 컴퓨터에서 일반적인 마이그레이션 방법은 다음과 같습니다.

```
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
```

주

- Solaris 10 플랫폼에서 `dsmig migrate-config` 명령이 실패하는 경우 해당 명령을 실행하는 동안 라이브러리 경로를 `null`로 설정합니다.

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

- 마이그레이션하는 Directory Server 인스턴스가 Sun Java Communications Suite Messaging Server 구성 요소와 같은 다른 Java ES 구성 요소의 구성 데이터를 저장하는 경우 `o=netscaperoot`라는 이름의 디렉토리 정보 트리의 특정한 부분을 마이그레이션해야 할 수 있습니다. 이 루트 접미어는 기본적으로 마이그레이션되지 않습니다. `o=netscaperoot`를 마이그레이션하려면 `dsmig migrate-config` 및 `dsmig migrate-data` 명령에서 `-N` 옵션을 사용하십시오. 예를 들면 다음과 같습니다.

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

- 32비트 아키텍처에서 64비트 아키텍처로 인스턴스를 마이그레이션하는 경우 `dsmig migrate-data` 명령 (자동 마이그레이션 도구)을 사용할 수 없습니다. 아래에 참조된 Migration Guide의 설명에 따라 수동으로 데이터를 마이그레이션해야 합니다. 그러나 스키마, 구성 및 보안 데이터는 계속 자동으로 마이그레이션할 수 있습니다.
- 디렉토리 데이터를 마이그레이션하고 Directory Server를 시작하면 새로운 릴리스 5 오류 검사 기능이 Directory Server 그룹 항목에서 순환 정의를 검색하는 경우가 있습니다. 이러한 순환 정의가 기능상 큰 문제는 아니지만 오류 파일에 많은 수의 오류가 기록될 수 있습니다.

마이그레이션 과정, `dsmig` 명령 및 수동 마이그레이션에 대한 자세한 내용은 Directory Server Enterprise Edition 6 Migration Guide (<http://docs.sun.com/doc/819-0994>)를 참조하십시오.

업그레이드 확인

다음과 같이 Directory Server가 성공적으로 업그레이드되었는지 확인할 수 있습니다.

1. 새 Directory Server 인스턴스를 시작합니다.

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

출력 값에 대해서는 [106페이지 표 5-4](#)를 참조하십시오.

2. Directory Server 오류 로그에서 시작 메시지를 확인합니다.

instancePath/logs/errors

주	시작 시 릴리스 5 Directory Server가 순환 정의를 감지합니다. 이러한 순환 정의는 기능상 큰 문제가 없지만 이러한 순환 정의가 포함된 이전 버전에서 업그레이드하는 경우 많은 오류가 기록될 수 있습니다.

사후 업그레이드 작업

Directory Server에 종속된 모든 Java ES 구성 요소가 새 Directory Server 인스턴스를 가리키도록 재구성해야 한다는 것을 제외하면 [108페이지의 "업그레이드 절차"](#)에 설명된 단계 이외의 사후 업그레이드 작업은 없습니다.

업그레이드 롤백

릴리스 5로 업그레이드한 다음에도 수정되지 않는 이전 버전으로 되돌리면 릴리스 5 업그레이드를 롤백할 수 있습니다.

여러 인스턴스 업그레이드

[106페이지의 "릴리스 4 Directory Server 업그레이드"](#)에 있는 절차에서는 가용성이나 확장성을 위해 Directory Server가 복제되는 배포 구조를 명시적으로 다루지는 않습니다. 이러한 구조에는 Directory Server 복제 또는 Directory Server 배포를 Sun Cluster 환경에서 데이터 서비스로 포함할 수 있습니다.

Directory Server 복제의 롤링 업그레이드

멀티마스터 복제 배포 구조에서 사용된 것처럼 다른 컴퓨터 시스템의 여러 Directory Server 인스턴스를 순서대로 한 번에 하나씩 업그레이드할 수 있습니다. 모든 Directory Server 마스터를 동기화한 다음 해당 호스트 컴퓨터에서 다른 인스턴스가 실행되는 동안 각 인스턴스를 업그레이드합니다. 이러한 롤링 업그레이드를 사용하면 서비스를 제공하는 개별 Directory Server를 업그레이드하는 동안 디렉토리 서비스를 온라인 상태로 유지할 수 있습니다.

Directory Server를 데이터 서비스로 업그레이드

Sun Cluster 환경에서 Directory Server를 데이터 서비스로 업그레이드 및 롤백하는 방법에 대한 정보는 현재 개발 중에 있습니다.

Java ES 릴리스 3에서 Directory Server 업그레이드

Java ES 2003Q1(릴리스 3) Directory Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Directory Server를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 3 Directory Server를 릴리스 5로 업그레이드하려면 [105페이지의 "Java ES 릴리스 4에서 Directory Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

Java ES 릴리스 2에서 Directory Server 업그레이드

Java ES 2004Q2(릴리스 2) Directory Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Directory Server를 릴리스 5로 업그레이드하는 절차와 동일하지만, 사전 업그레이드 작업에는 모든 공유 구성 요소를 릴리스 5로 업그레이드하는 작업이 포함되어야 합니다([47페이지 표 1-9](#) 참조).

[63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"](#)에서 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

릴리스 2 Directory Server를 릴리스 5로 업그레이드하려면 [105페이지의 "Java ES 릴리스 4에서 Directory Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바꾸어 생각하십시오.

주	Linux 플랫폼에서 릴리스 2 Directory Server를 업그레이드하는 경우 Directory Server <i>와</i> 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Directory Server가 지원되지 않음). 자세한 내용은 103페이지의 "이중 업그레이드" 를 참조하십시오.

Directory Proxy Server

이 장에서는 Directory Proxy Server를 Java ES 5(릴리스 5): Sun Java System Directory Proxy Server 6.0으로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- 116페이지의 "Directory Proxy Server 업그레이드 개요"
- 120페이지의 "Java ES 릴리스 4에서 Directory Proxy Server 업그레이드"
- 126페이지의 "Java ES 릴리스 3에서 Directory Proxy Server 업그레이드"
- 127페이지의 "Java ES 릴리스 2에서 Directory Proxy Server 업그레이드"

주 이 장에 설명된 파일 위치는 *serverRoot*(Directory Proxy Server 5.x) 및 *DirServer-base*(Directory Proxy Server 6.0)라는 디렉토리 경로에 대해 상대적으로 지정됩니다. Directory Proxy Server를 처음 설치할 때 최소한 이 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 6-1 Directory Proxy Server 디렉토리 경로		
경로 이름 변수	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 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Directory Proxy Server의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 정보](#)
- [Java ES 릴리스 5 업그레이드 로드맵](#)
- [Directory Proxy Server 데이터](#)
- [Directory Proxy Server 업그레이드 전략](#)

Java ES 릴리스 5 정보

Java ES 릴리스 5 Directory Proxy Server는 주요 릴리스이며 릴리스 4 Directory Proxy Server 및 모든 이전 릴리스와 다른 새로운 제품입니다.

릴리스 5 Directory Proxy Server는 여전히 LDAP 프록시이지만 새로 확장된 라우팅 기능이 있습니다. 또한 릴리스 5를 사용하면 여러 데이터 보기를 하나의 보기에 표시하는 가상 디렉토리 기능을 사용할 수 있습니다. 이러한 데이터 보기는 LDAP 또는 SQL 액세스 가능 데이터 저장소를 나타낼 수 있습니다.

자세한 내용은 Directory Server Enterprise Edition 6 릴리스 노트를 참조하십시오.

Java ES 릴리스 5 업그레이드 로드맵

[표 6-2](#)는 Directory Proxy Server를 Java ES 릴리스 5로 업그레이드하는 방법을 보여줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 6-2 Java ES 5(릴리스 5) 업그레이드 경로: Directory Proxy Server 6.0

Java ES 릴리스	Directory Proxy Server 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Directory Proxy Server 5.2 2005Q4	직접 업그레이드: 릴리스 5를 새로 설치하고 구성하여 릴리스 4를 대체합니다.	이전 버전과 호환해야 하는 경우에는 이전 구성을 새 구성 등록 정보에 수동으로 매핑합니다.
릴리스 3	Sun Java System Directory Proxy Server 5.2 2005Q1	직접 업그레이드: 릴리스 5를 새로 설치하고 구성하여 릴리스 3를 대체합니다.	이전 버전과 호환해야 하는 경우에는 이전 구성을 새 구성 등록 정보에 수동으로 매핑합니다.
릴리스 2	Sun Java System Directory Proxy Server 5.2 2004Q2	직접 업그레이드: 릴리스 5를 새로 설치하고 구성하여 릴리스 2를 대체합니다.	이전 버전과 호환해야 하는 경우에는 이전 구성을 새 구성 등록 정보에 수동으로 매핑합니다.

표 6-2 Java ES 5(릴리스 5) 업그레이드 경로: Directory Proxy Server 6.0 (계속)

Java ES 릴리스	Directory Proxy Server 버전	일반적인 방법	재구성 필요
릴리스 1	Sun ONE Directory Proxy Server 5.2	직접 업그레이드 인증 안 됨: 그러나 릴리스 2에서 업그레이드할 때와 동일한 방법을 사용할 수 있습니다.	이전 버전과 호환해야 하는 경우에는 이전 구성을 새 구성 등록 정보에 수동으로 매핑합니다.
이전 Java ES 릴리스	Sun ONE Directory Proxy Server 5.2	직접 업그레이드 인증 안 됨: 그러나 릴리스 2에서 업그레이드할 때와 동일한 방법을 사용할 수 있습니다.	이전 버전과 호환해야 하는 경우에는 이전 구성을 새 구성 등록 정보에 수동으로 매핑합니다.
	Sun ONE Directory Access Router 5.0 또는 5.0 SP1	직접 업그레이드 불가능: 먼저 릴리스 3으로 업그레이드합니다. 자세한 내용은 Java Enterprise System 2005Q1 Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)를 참조하십시오. 그런 다음 릴리스 3에서 릴리스 5로 업그레이드합니다.	Java Enterprise System 2005Q1 Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)를 참조하십시오.

Directory Proxy Server 데이터

Directory Proxy Server는 더 이상 Directory Server를 사용하여 구성 데이터를 저장하지 않습니다. 구성은 새로운 Directory Service Control Center 또는 Directory Server EE 명령줄 유틸리티를 사용하여 수행됩니다.

다음 표는 Directory Proxy Server 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 6-3 Directory Proxy Server 데이터 사용

데이터 유형	위치	사용
Directory Proxy Server 구성 데이터	Directory Proxy Server 5.2: 구성 디렉토리 Directory Proxy Server 6.0: Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티를 사용하여 액세스	Directory Proxy Server 구성
보안 데이터	Directory Proxy Server 5.2: Directory Proxy Server 콘솔을 사용하여 구성된 SSL Directory Proxy Server 6.0: Directory Service Control Center 및 Directory Server EE 명령줄 유틸리티를 통해 SSL 구성	서버 인증서

Directory Proxy Server 업그레이드 전략

Directory Proxy Server 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Directory Proxy Server 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Directory Proxy Server에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Directory Proxy Server는 인터페이스가 변경되어 이전 Directory Proxy Server 릴리스와 호환되지 않습니다. 릴리스 5 Directory Proxy Server는 완전히 새로운 Java 기반 구현을 바탕으로 하고 있으며 릴리스 4 Directory Proxy Server 및 이전 릴리스와 근본적으로 구성에 차이가 있습니다.

그러나 릴리스 5 Directory Proxy Server가 이전 버전과 호환되도록, 즉 릴리스 4 Directory Proxy Server 및 이전 릴리스처럼 동작하도록 구성하는 것은 가능합니다. 이러한 구성을 위해서는 이전 구성 속성을 릴리스 5 구성 등록 정보에 수동으로 매핑해야 합니다. 자세한 내용은 Directory Server Enterprise Edition 6 Migration Guide (<http://docs.sun.com/doc/819-0994>)를 참조하십시오.

그러나 릴리스 5 Directory Proxy Server는 이전 버전과 비교하여 기본 동작이 다르기 때문에 LDAP 제어 작업에서 프록시를 통과할 수 없습니다. 이전 버전처럼 동작하도록 하려면 125페이지의 "사후 업그레이드 작업"에 설명된 대로 이러한 제어에 대해 차단을 해제해야 합니다.

종속성

다른 Java ES 구성 요소에 대한 종속성은 Directory Proxy Server 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다.

Directory Proxy Server는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Directory Proxy Server는 특정 Java ES 공유 구성 요소에 대해 종속성이 있습니다(47페이지 표 1-9 참조). Directory Proxy Server 업그레이드는 이러한 공유 구성 요소의 업그레이드된 버전에 종속됩니다.
- **Directory Server.** Directory Proxy Server는 LDAP 요청에 대한 향상된 보안과 성능을 제공하는 Directory Server에 대해 상호 종속성이 있습니다. Directory Proxy Server는 Directory Server에 대한 프런트엔드 액세스를 제공하지만 이러한 기능적 관계 이외에는 Directory Server에 대한 종속성이 없습니다.

이중 업그레이드

Directory Proxy Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 다음 두 가지 방법 중 하나로 수행할 수 있습니다.

- 운영 체제 새로 설치
- 기존 운영 체제 업그레이드

운영 체제 새로 설치

1. 기존 Directory Proxy Server 데이터를 백업합니다.

필수 데이터의 위치는 [117페이지의 "Directory Proxy Server 데이터"](#)를 참조하십시오.

2. 새 운영 체제를 설치합니다.

운영 체제는 새 시스템 (또는 Solaris 10 영역)에 설치하거나 기존 파일 시스템을 덮어쓸 수 있습니다.

3. 릴리스 5 Directory Proxy Server를 설치합니다.

4. 릴리스 5 Directory Proxy Server 인스턴스를 만들고 구성 속성을 릴리스 5 Directory Proxy Server 등록 정보에 매핑합니다.

[122페이지의 "릴리스 4 업그레이드 Directory Proxy Server"](#)의 절차에서 관련 단계를 참조하십시오.

기존 운영 체제 업그레이드

1. 기존 Directory Proxy Server 데이터를 백업합니다.

필수 데이터의 위치는 [117페이지의 "Directory Proxy Server 데이터"](#)를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Directory Proxy Server로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 관련 절을 참조하십시오.

Java ES 릴리스 4에서 Directory Proxy Server 업그레이드

이 절에는 Directory Proxy Server를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Directory Proxy Server 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Directory Proxy Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 릴리스 5 Directory Proxy Server를 새로 설치하고 Directory Service Control Center 또는 Directory Server EE 명령줄 유틸리티를 사용하여 새 Directory Proxy Server 인스턴스를 구성하여 업그레이드를 수행할 수 있습니다.
- **업그레이드 종속성.** Directory Proxy Server는 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Directory Proxy Server를 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다.
- **역방향 호환성.** [118페이지의 "호환성 문제"](#)에서 설명한 것처럼 릴리스 5 Directory Proxy Server가 릴리스 4 버전과 호환되도록 구성할 수 있습니다.
- **업그레이드 롤백.** 릴리스 5로 업그레이드한 다음에도 수정되지 않는 이전 버전으로 되돌리면 릴리스 5 업그레이드를 롤백할 수 있습니다.
- **플랫폼 문제.** Directory Proxy Server를 업그레이드하는 방법은 Solaris와 Linux 운영 체제에서 동일합니다.

릴리스 4 Directory Proxy Server 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Directory Proxy Server를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Directory Proxy Server](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Directory Proxy Server 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Directory Proxy Server 종속성 업그레이드](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

다음 명령을 사용하여 Directory Proxy Server의 현재 버전을 확인할 수 있습니다.

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

출력은 다음 표에 표시되어 있습니다.

표 6-4 Directory Proxy Server 버전 확인 출력

Java ES 릴리스	Directory Proxy Server 버전 번호
릴리스 2	Sun ONE Directory Proxy Server Version 5.2_Patch_2
릴리스 3	Sun ONE Directory Proxy Server Version 5.2_Patch_3
릴리스 4	Sun ONE Directory Proxy Server Version 5.2_Patch_4
릴리스 5 ¹	Sun ONE Directory Proxy Server 버전 6.0

1. ldapfwd 명령을 사용하여 릴리스 5의 버전 번호를 반환할 수 없습니다. [124페이지의 "업그레이드 확인"](#)을 참조하십시오.

Directory Proxy Server 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Directory Proxy Server는 몇몇 공유 구성 요소에 대해서만 엄격한 업그레이드 종속성이 있습니다.

Directory Proxy Server 종속성을 업그레이드할 때는 Directory Proxy Server를 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다. 공유 구성 요소 업그레이드는 일반적으로 Java ES 설치 프로그램을 통해 자동으로 수행됩니다.

1. **공유 구성 요소.** 63페이지의 "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Directory Proxy Server에 필요한 모든 공유 구성 요소는 Directory Proxy Server를 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.
2. **Directory Server(엄격하지 않은 업그레이드 종속성).** 99페이지의 5장, "[Directory Server](#)"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다. 그러나 릴리스 5 Directory Proxy Server는 릴리스 4 Directory Server에서도 지원됩니다.

필수 구성 정보 및 비밀번호 얻기

구성 정보는 업그레이드 과정에서 보존되며 이 정보를 사용하여 릴리스 4 구성 속성을 릴리스 5 구성 등록 정보에 매핑할 수 있습니다. 118페이지의 "[호환성 문제](#)"를 참조하십시오.

릴리스 4 업그레이드 Directory Proxy Server

이 절에서는 Directory Proxy Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

Directory Proxy Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- Directory Proxy Server 인스턴스를 사용하는 모든 Java ES 구성 요소(Access Manager, Communications Express, Messaging Server, Portal Server 등)를 종료하고 해당하는 새 릴리스 5 인스턴스에 액세스하도록 다시 구성해야 합니다.
- 단일 컴퓨터에서 Directory Proxy Server의 여러 인스턴스(동일한 설치 Directory Proxy Server 이미지에 해당하는 모든 항목)가 실행되는 배포 구조에서 Directory Proxy Server 이미지를 업그레이드하려면 새 Directory Proxy Server 인스턴스를 만들어야 합니다.

- Directory Proxy Server가 포함된 릴리스 4 배포 구조에서는 Administration Server를 사용하여 Directory Proxy Server 인스턴스를 구성하고 관리합니다. 릴리스 5에서 Directory Proxy Server를 업그레이드하려면 Directory Proxy Server 인스턴스를 구성하고 관리하는 데 사용하는 Directory Service Control Center를 배포해야 할 수 있습니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 로컬에 상주하는 Directory Proxy Server 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 업그레이드할 Directory Proxy Server 인스턴스에 종속되어 있는 Java ES 구성 요소를 모두 종료하십시오. 이 단계는 Directory Proxy Server가 배포 구조에서 복제되는 방법에 따라 달라집니다.

Java ES 구성 요소 종료 방법에 대한 내용은 해당 관리 설명서를 참조하십시오.

3. 릴리스 5 Directory Proxy Server를 새로 설치합니다.

다음 단계를 수행합니다.

- a. Java ES 설치 프로그램을 시작합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. Directory Server Enterprise Edition의 Directory Proxy Server 하위 구성 요소를 선택합니다.

사용하려는 관리 하위 구성 요소(Directory Service 제어판 또는 명령줄 유틸리티)도 설치해야 합니다.

- c. 기존 릴리스 4 Directory Proxy Server의 설치 경로와 다른 설치 경로를 지정합니다.

- d. "지금 구성" 또는 "나중에 구성"을 선택합니다.

Directory Proxy Server에는 구성이 필요하지 않기 때문에 "지금 구성"이나 "나중에 구성" 중 어떤 것을 선택해도 차이가 없습니다. 그러나 "지금 구성"을 선택하면 새 인스턴스를 만들 것인지 묻는 메시지가 표시됩니다. 구성 요소를 설치한 다음에는 원하는 만큼 Directory Proxy Server 인스턴스를 만들 수 있습니다.

- e. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.
- f. 설치 선택 항목을 확인합니다.

Directory Proxy Server 패키지가 설치되고 설치 요약이 표시됩니다.

- g. Java ES 설치 프로그램을 종료합니다.

새 인스턴스를 만들지 않은 경우 [단계 4](#)로 진행하고, 인스턴스를 만든 경우에는 [단계 5](#)로 진행합니다.

4. Directory Proxy Server 인스턴스를 만듭니다.

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

여기서 *instancePath*는 Directory Proxy Server 인스턴스의 전체 경로입니다.

Directory Proxy Server 인스턴스를 만드는 방법에 대한 자세한 내용은 Directory Server Enterprise Edition 6 관리 설명서 (<http://docs.sun.com/doc/820-0292>)를 참조하십시오.

5. 원하는 경우 릴리스 4 구성 속성을 릴리스 5 Directory Proxy Server 등록 정보에 매핑합니다.

매핑 절차에 대한 자세한 내용은 Directory Server Enterprise Edition 6 Migration Guide(<http://docs.sun.com/doc/819-0994>)를 참조하십시오.

업그레이드 확인

다음과 같이 Directory Proxy Server가 성공적으로 업그레이드되었는지 확인할 수 있습니다.

1. 새 Directory Proxy Server 인스턴스를 시작합니다.

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

2. Directory Proxy Server 버전을 확인합니다.

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

[121페이지 표 6-4](#)에서 결과 값을 보여 줍니다.

사후 업그레이드 작업

Directory Proxy Server에 종속된 모든 Java ES 구성 요소가 새 Directory Proxy Server 인스턴스를 가리키도록 재구성해야 합니다.

또한 이전 버전의 기본 동작을 그대로 사용하려면 LDAP 제어가 프록시를 통과할 수 있도록 명시적으로 허용해야 합니다. 다음과 같이 `allowed-ldap-controls` 등록 정보를 설정하여 이러한 제어를 활성화할 수 있습니다.

```
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
```

업그레이드 롤백

릴리스 5로 업그레이드한 다음에도 수정되지 않는 이전 버전으로 되돌리면 릴리스 5 업그레이드를 롤백할 수 있습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Directory Proxy Server를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Directory Proxy Server 구성 요소를 실행하여 로드를 분산시킬 수 있습니다.

로드 균형 조정된 Directory Proxy Server 인스턴스의 경우 롤링 업그레이드를 수행할 수 없습니다. 로드 밸런서를 종료하고 릴리스 5 인스턴스에 액세스하도록 다시 구성해야 합니다. [121 페이지의 "릴리스 4 Directory Proxy Server 업그레이드"](#)의 설명을 참조하여 각 인스턴스의 업그레이드를 수행하십시오.

Java ES 릴리스 3에서 Directory Proxy Server 업그레이드

Java ES 2005Q1(릴리스 3) Directory Proxy Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Directory Proxy Server를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 3 Directory Proxy Server를 릴리스 5로 업그레이드하려면 [120페이지의 "Java ES 릴리스 4에서 Directory Proxy Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

Java ES 릴리스 2에서 Directory Proxy Server 업그레이드

Java ES 2004Q2(릴리스 2) Directory Proxy Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Directory Proxy Server를 릴리스 5로 업그레이드하는 절차와 동일하지만, 사전 업그레이드 작업에는 모든 공유 구성 요소([47페이지 표 1-9](#) 참조)와 Directory Proxy Server가 종속되는 모든 로컬 상주 제품 구성 요소를 릴리스 5로 동기화하는 작업이 포함됩니다.

[63페이지의 2장](#), "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다.

릴리스 2 Directory Proxy Server를 릴리스 5로 업그레이드하려면 [120페이지의 "Java ES 릴리스 4에서 Directory Proxy Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바뀌어 생각하십시오.

주

Linux 플랫폼에서 릴리스 2 Directory Proxy Server를 업그레이드하는 경우 Directory Proxy Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Directory Proxy Server가 지원되지 않음). 자세한 내용은 [119페이지의 "이중 업그레이드"](#)를 참조하십시오.

Web Server

이 장에서는 Web Server를 Java ES 5(릴리스 5): Sun Java System Web Server 7.0으로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- 130페이지의 "Web Server 업그레이드 개요"
- 134페이지의 "Java ES 릴리스 4에서 Web Server 업그레이드"
- 151페이지의 "Java ES 릴리스 3에서 Web Server 업그레이드"
- 152페이지의 "Java ES 릴리스 2에서 Web Server 업그레이드"

주 이 장에 설명된 파일 위치는 *WebServer6-base*(Web Server 6.x)와 *WebServer7-base* 및 *WebServer7Config-base*(Web Server 7.0) 디렉토리 경로에 대해 상대적으로 지정됩니다. Web Server를 처음 설치할 때 최소한 이 두 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 7-1 Web Server 디렉토리 경로

경로 이름 변수	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 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Web Server의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Web Server 정보](#)
- [Web Server 업그레이드 로드맵](#)
- [Web Server 데이터](#)
- [Web Server 업그레이드 전략](#)

Java ES 릴리스 5 Web Server 정보

Java ES 릴리스 5 Web Server는 릴리스 4와 비교할 때 다양한 새 기능과 향상된 기능이 포함된 주요 릴리스입니다.

릴리스 5 Web Server는 새 관리 도구가 포함된 새로운 관리 인프라를 제공합니다. 관리 인프라에는 많은 수의 Web Server 인스턴스에 대해 구성 정보를 호스팅하는 Administration Server 인스턴스가 포함됩니다. Web Server 인스턴스를 로컬로 또는 원격 컴퓨터에서 만들고 이러한 인스턴스를 구성 및 관리하는 데는 새로운 명령줄 인터페이스(wadm) 및 그래픽 사용자 인터페이스가 사용됩니다. 새 관리 도구를 사용하려면 관리자 이름과 비밀번호가 필요합니다.

새 관리 인프라에 대한 자세한 내용은 Web Server 7.0 관리자 설명서 (<http://docs.sun.com/doc/820-0874>)를 참조하십시오.

Web Server 관리 인터페이스의 이러한 변화는 업그레이드에도 상당한 영향을 줍니다.

Web Server 업그레이드 로드맵

[표 7-2](#)는 Web Server를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 7-2 Java ES 5(릴리스 5) 업그레이드 경로: Web Server 7.0

Java ES 릴리스	Web Server 버전	일반적인 방법	재구성 필수
릴리스 4	Sun Java System Web Server 6.1 SP5 2005Q4	직접 업그레이드: 새로 설치하고 데이터 마이그레이션 수행	새 인스턴스로 인스턴스 구성 마이그레이션

표 7-2 Java ES 5(릴리스 5) 업그레이드 경로: Web Server 7.0

Java ES 릴리스	Web Server 버전	일반적인 방법	재구성 필수
릴리스 3	Sun Java System Web Server 6 2005Q1 Update 1 SP 4	직접 업그레이드: 새로 설치하고 데이터 마이그레이션 수행	새 인스턴스로 인스턴스 구성 마이그레이션
릴리스 2	Sun Java System Web Server 6 2004Q2 Update 1 SP 2 Platform Edition 및 Enterprise Edition	직접 업그레이드: 새로 설치하고 데이터 마이그레이션 수행	새 인스턴스로 인스턴스 구성 마이그레이션
릴리스 1	Sun ONE Web Server 6.1(2003Q4)	직접 업그레이드 인증 안 됨: 하지만 새로 설치하고 데이터를 마이그레이션하여 수행 가능	새 인스턴스로 인스턴스 구성 마이그레이션
이전 Java ES 릴리스		직접 업그레이드할 수 없습니다.	

Web Server 데이터

다음 표는 Web Server 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 7-3 Web Server 데이터 사용

데이터 유형	위치	사용
구성 데이터	Web Server 6.x(Java ES 릴리스 2, 3 및 4): <i>WebServer6-base/https-instanceName/config/</i> Web Server 7.0(Java ES 릴리스 5) Instance Configuration <i>WebServer7Config-base/https-configName/config¹/</i> 중앙 구성 저장소 Web Server 콘솔 및 wadm 명령줄 인터페이스를 통해 액세스됨	Web Server 인스턴스 구성

1. *WebServer7Config-base* 경로는 *WebServer6-base* 경로와 크게 다릅니다.

Web Server 업그레이드 전략

Web Server 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Web Server 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Web Server에 대한 일반적인 내용을 설명합니다.

호환성 문제

Java ES 릴리스 5 Web Server는 공용 인터페이스가 변경되지 않았으며 이 때문에 이전 버전과 호환됩니다. 그러나 새 관리 인터페이스는 이전의 관리 인터페이스와 호환되지 않습니다. 따라서 웹 응용 프로그램(예: Java ES 구성 요소 포함)의 업그레이드와 재배포에 영향을 줍니다.

특히 릴리스 5 Web Server는 다음 표에 나와 있는 것처럼 인스턴스 디렉토리 및 가상 서버 이름에 다른 기본값을 사용합니다.

표 7-4 Web Server 기본 이름

항목	Java ES 릴리스 2, 3 및 4 Web Server 6.x 기본값	Java ES 릴리스 5 Web Server 7.0 기본값
구성 이름		<i>hostName.domainName</i>
인스턴스 디렉토리 경로	<i>WebServer6-base/ https-hostName.domainName</i>	<i>WebServer7Config-base https-hostName.domainName</i>
가상 서버 이름	<i>https-hostName.domainName</i>	<i>hostName.domainName</i>

Web Server 종속성

Web Server는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Web Server는 특정 Java ES 공유 구성 요소에 대해 종속성이 있습니다(47페이지 표 1-9 참조). Web Server 업그레이드는 이러한 공유 구성 요소의 업그레이드된 버전에 종속됩니다.
- **Directory Server.** Web Server는 LDAP 기반 인증을 제공하는 Directory Server에 대해 선택적 종속성이 있습니다.
- **Web Proxy Server.** Web Server는 HTTP 요청에 대한 향상된 보안과 성능을 제공하는 Web Proxy Server에 대해 상호 종속성이 있습니다.

이중 업그레이드

Web Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 기존 운영 체제 업그레이드 방식을 사용하여 수행할 수 있습니다.

1. 기존 Web Server 데이터를 백업합니다.

필수 데이터의 위치는 [131페이지의 "Web Server 데이터"](#)를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Web Server로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 해당 절을 참조하십시오.

Java ES 릴리스 4에서 Web Server 업그레이드

이 절에는 Web Server를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Web Server 업그레이드](#)

소개

Java ES 릴리스 4 Web Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 릴리스 5 Web Server를 새로 설치하고 릴리스 4 Web Server 인스턴스 구성 정보를 릴리스 5 구성으로 마이그레이션한 다음 릴리스 4 인스턴스에 해당하는 릴리스 5 Web Server 인스턴스를 만들어 업그레이드를 수행할 수 있습니다.
- **업그레이드 종속성.** Web Server에는 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Java ES 설치 프로그램이 Web Server 업그레이드를 수행할 때 릴리스 5로 자동 업그레이드됩니다. Web Server는 NSS 및 NSPR 공유 구성 요소에 대해서만 엄격한 업그레이드 종속성이 있습니다.
- **역방향 호환성.** 릴리스 5 Web Server 관리 인터페이스는 릴리스 4 버전과 호환되지 않습니다.
- **업그레이드 롤백.** 변경되지 않고 그대로 유지되어 있는 릴리스 4 설치로 되돌리는 방법으로 릴리스 5 업그레이드를 롤백할 수 있습니다.
- **플랫폼 문제.** Web Server를 업그레이드하는 일반적인 방법은 Solaris와 Linux 운영 체제에서 동일합니다.

릴리스 4 Web Server 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Web Server를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Web Server](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Web Server 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Web Server 종속성 업그레이드](#)
- [Web Server 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

Web Server의 현재 버전은 `-version` 옵션을 사용하여 Web Server를 실행하면 확인할 수 있습니다.

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

표 7-5 Web Server 버전 확인 출력

Java ES 릴리스	Web Server 버전 번호
릴리스 2	6.1SP2
릴리스 3	6.1SP4
릴리스 4	6.1SP5
릴리스 5	7.0

Web Server 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. 하지만 Web Server를 릴리스 5로 업그레이드하면 Web Server에 필요한 모든 공유 구성 요소([47페이지 표 1-9](#) 참조)가 Java ES 설치 프로그램에 의해 자동으로 설치됩니다.

Web Server 데이터 백업

Web Server를 릴리스 4에서 릴리스 5로 업그레이드하는 경우 기존 구성 데이터는 수정되지 않고 그대로 유지됩니다. 따라서 현재 데이터를 백업하지 않아도 됩니다.

필수 구성 정보 및 비밀번호 얻기

업그레이드를 수행하려면 슈퍼유저로 로그인해야 하며 마이그레이션을 수행하는 사용자 계정에는 기존 Web Server 설치 디렉토리에 액세스할 수 있는 권한이 있어야 합니다.

릴리스 4 업그레이드 Web Server

이 절에서는 Web Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

Web Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- **지금 구성 또는 나중에 구성.** 업그레이드를 수행하는 경우 "지금 구성" 또는 "나중에 구성" 옵션 중에서 릴리스 5 Web Server를 설치하는 데 사용할 옵션을 지정해야 할 수 있습니다.
 - "지금 구성"을 선택하면 설치 프로그램이 지정된 대로 Administration Server 또는 Administration Node를 설정하며 기본 구성 및 해당 Web Server 인스턴스를 만듭니다. 이 방법은 한 대의 컴퓨터에서 설치할 때 유용하지만 기본 구성 이름은 업그레이드 중에 기존 인스턴스 구성의 마이그레이션에 영향을 줄 수 있습니다.
 - "나중에 구성"에서는 설치 프로그램이 구성을 수행하지 않습니다. 따라서 사용자가 입력 파일에 등록 정보 값을 제공한 후 configureServer 스크립트를 수동으로 실행해야 합니다. 이 방법은 자동 설치를 수행하는 스크립트를 사용하여 여러 컴퓨터에서 설치를 자동화하려는 경우 유용합니다. 구성 이름을 완전히 제어할 수 있으므로 업그레이드 중에 기존 인스턴스 구성의 마이그레이션과 충돌을 방지할 수 있습니다.

- **기본 릴리스 4 인스턴스 구성 마이그레이션.** 업그레이드를 수행하는 경우 각 릴리스 4 Web Server 인스턴스의 구성 데이터를 Web Server Administration Server에서 관리하는 중앙 구성 저장소로 마이그레이션하게 됩니다. 마이그레이션은 `wadm migrate-server` 명령 또는 릴리스 5 관리 콘솔을 사용하여 수행됩니다.

마이그레이션되는 인스턴스가 기본 릴리스 4 Web Server (6.x) 인스턴스인 경우 "지금 구성" 옵션에서 자동으로 만든 기본 릴리스 5 Web Server (7.0) 구성과 같은 이름(*hostName.domainName*)을 사용하게 됩니다.

기본 릴리스 4 인스턴스 구성을 릴리스 5로 마이그레이션하는 방법에는 세 가지가 있으며 각각 구성 이름이 달라지게 됩니다.

선택하는 방법에 따라 배포된 웹 응용 프로그램의 후속 업그레이드에 영향을 줄 수 있습니다. 예를 들어 배포된 Java ES 구성 요소(예: Access Manager 및 Portal Server)와 Sun Java Communications Suite 구성 요소(예: Communications Express, Instant Messaging 및 Delegated Administrator)를 업그레이드하려면 일반적으로 해당 업그레이드를 수행하는 사용자가 릴리스 4 인스턴스 구성이 어떤 릴리스 5 구성 이름으로 업그레이드되었는지 알아야 합니다.

세 가지 방법은 다음과 같습니다.

- 기본 이름과 다른 새 구성 이름(예: "JavaESapps")을 지정합니다. 그러면 `migrate-server` 명령에서 JavaESapps라는 릴리스 5 구성을 새로 만듭니다.
이어서 업그레이드된 웹 응용 프로그램을 JavaESapps 구성으로 재배포해야 합니다.
- 새 구성 이름을 지정하지 마십시오. 하지만 `migrate-server` 명령을 실행하기 전에 기본 릴리스 5 인스턴스 및 구성(*hostName.domainName*)을 삭제합니다. `migrate-server` 명령은 기본 이름(*hostName.domainName*)을 사용하여 새 구성을 만듭니다. 다음 과정이 이어집니다(자세한 내용은 `wadm help` 참조).

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

이어서 업그레이드된 웹 응용 프로그램을 기본 릴리스 4 인스턴스 이름과 동일한 *hostName.domainName* 구성으로 재배포해야 합니다.

- 새 구성 이름을 지정하지 말고 기본 릴리스 5 구성을 삭제하지 마십시오. migrate-server 명령은 *hostName.domainName-1*이라는 이름을 사용하여 새 구성을 만듭니다.

이어서 업그레이드된 웹 응용 프로그램을 *hostName.domainName-1* 구성으로 재배포해야 합니다.

릴리스 4 인스턴스 구성을 마이그레이션하는 방법이 무엇이든, 릴리스 4 인스턴스 구성을 마이그레이션한 릴리스 5 구성 이름을 해당 인스턴스에 배포한 웹 응용 프로그램의 후속 업그레이드를 수행할 사용자에게 알려야 합니다.

- **구성 데이터의 마이그레이션.** 릴리스 4 인스턴스 구성을 마이그레이션하면 다음 정보가 자동으로 마이그레이션됩니다.

- 릴리스 4 Web Server 인스턴스 디렉토리의 모든 구성 정보:
WebServer6-base/https-instanceName/config. 여기에는 릴리스 4 인스턴스에 배포된 모든 웹 응용 프로그램의 구성 정보가 포함됩니다(예: Access Manager 및 Portal Server와 같은 Java ES 구성 요소).
- *WebServer6-base/httpacl*의 acl 정보
- *WebServer6-base/userdb*의 auth-db 정보
- *WebServer6-base/https-admserv/config*의 스케줄러 정보
- *WebServer6-base/alias*의 인증서 정보
- 마이그레이션을 수행할 때 지정한 대로 컬렉션 정보 및 색인 파일을 검색합니다.

자동 마이그레이션에는 다음 데이터가 포함되지 *않습니다*.

- docroot 콘텐츠. 그 대신 새 구성이 이전 docroot를 가리키고 마이그레이션 로그에 로그 메시지가 기록됩니다.
- Webdav 데이터. Webdav 컬렉션 정보가 마이그레이션됩니다.
- 타사 NSAPI 플러그인은 마이그레이션되지 않습니다. 그 대신 릴리스 4 파일을 가리키고 마이그레이션 로그에 로그 메시지가 기록됩니다.
- 로그 파일
- 검색 컬렉션 docroot에 대한 변경 사항
- 명령줄 스크립트(*startsvr*, *startsvr.bat*, *stopsvr*, *stopsvr.bat*, *restart*, *reconfig*, *reconfig.bat*)

데이터 마이그레이션에 대한 자세한 내용은 Web Server 7.0 Installation and Migration Guide(<http://docs.sun.com/doc/819-2625>)를 참조하십시오.

- **릴리스 5 Web Server 인스턴스 만들기.** 마이그레이션한 다음 해당 마이그레이션된 릴리스 4 인스턴스에 해당하는 릴리스 5 Web Server 인스턴스를 명시적으로 만들어야 합니다. 이 작업은 자동으로 수행되지 않습니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 동일한 설치 Web Server 이미지에 해당하는 모든 Web Server 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 실행 중인 모든 Web Server 및 Administration Server 인스턴스를 중지합니다.

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

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

3. 릴리스 5 Web Server를 새로 설치합니다.

다음 단계를 수행합니다.

- a. Java ES 설치 프로그램을 시작합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령 줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. 구성 요소 선택 페이지에서 Web Server를 선택합니다.
- c. 릴리스 4 Web Server와 다른 설치 디렉토리 경로를 지정합니다.
- d. "지금 구성" 또는 "나중에 구성"을 선택합니다.
 - "지금 구성"을 선택한 경우 [단계 e](#)로 이동합니다.
 - "나중에 구성"을 선택한 경우 [단계 f](#)로 이동합니다.

- e. "지금 구성"을 선택하면 Java ES 설치 프로그램에서 두 가지 중 하나를 선택할 수 있습니다.
- 관리 인스턴스를 Administration Server로 구성
Administration Server를 호스팅할 컴퓨터에서 이 선택을 사용하십시오. 다른 관리 작업 중에도 릴리스 4 인스턴스를 릴리스 5로 마이그레이션하는 데 필요합니다.
 - 관리 인스턴스를 관리 노드로 구성
Administration Server에서 원격으로 Web Server 인스턴스를 호스팅할 컴퓨터에서 이 선택을 사용하십시오. 관리 인스턴스는 Administration Server와 상호 작용하는 노드 에이전트로 구성됩니다.
- I. 필요한 구성 값을 지정합니다.
- 호스트 이름, HTTP 포트, 관리자 이름 및 관리자 비밀번호를 입력해야 합니다.
- II. 설치 선택 항목을 확인합니다.
- Web Server 패키지가 설치되고 설치 요약이 표시됩니다.
- Java ES 설치 프로그램은 *hostName.domainName*이라는 기본 구성과 해당 Web Server 인스턴스를 만듭니다.
- III. Java ES 설치 프로그램을 종료하고 [142페이지의 단계 4](#)로 이동합니다.
- f. "나중에 구성"을 선택하면 Java ES 설치 프로그램이 [142페이지의 단계 IV](#)에서 실행할 `configureServer` 스크립트를 만듭니다.
- I. 설치 선택 항목을 확인합니다.
- Web Server 패키지가 설치되고 설치 요약이 표시됩니다.
- II. Java ES 설치 프로그램을 종료합니다.
- III. `WebServer7-base/setup/WSInstall.properties` 파일에서 값을 설정합니다.
- 다음 표에 나와 있는 모든 필수(선택 사항이 아님) 등록 정보에 값을 지정합니다.

표 7-6 WSInstall.properties 값

등록 정보	설명
WS_DOCROOT	(선택 사항) 웹 콘텐츠 파일을 호스팅할 수 있는 문서 위치
WS_SERVER_NAME	HTTP 요청에 대해 서비스를 제공하는 데 사용할 수 있는 호스트 이름

표 7-6 WSInstall.properties 값 (계속)

등록 정보	설명
WS_SERVER_USER	런타임 UNIX 사용자. 유효한 값은 root, 모든 유효한 UNIX 사용자 또는 webservd(기본값)입니다.
WS_HTTP_PORT	HTTP 요청을 수신하는 데 사용할 수 있는 인스턴스 포트
WS_ADMIN_SSL_PORT	관리 SSL 포트
WS_ADMIN_HOST	관리 서버 작업을 위한 관리 호스트 이름
WS_CONFIG_NAME	이 호스트의 구성 이름. 이 값은 WS_SERVER_NAME에 제공된 값과 같은 값일 수 있습니다.
WS_ADMIN_SERVER_USER	관리 서버 런타임 UNIX 사용자. 유효한 값: 'root' 또는 WS_SERVER_USER와 동일한 사용자
WS_ADMIN_LOGIN_USER	관리 서버 로그인 사용자 이름
WS_ADMIN_LOGIN_PASSWORD	관리 서버 로그인 비밀번호
WS_ADMIN_HTTP_PORT	(선택 사항) 관리 비SSL 포트. 기본값: 8800
WS_START_ON_BOOT	(선택 사항) 부팅 시 시작 기능(true/false). true를 선택하면 시스템 재부팅 후에 서버 인스턴스 및 해당 관리 서버를 자동으로 시작할 수 있습니다. 기본값: false
WS_64BIT_INSTALL	(선택 사항) 서버 런타임 모드(true/false). true를 선택하면 서버가 64비트 모드로 구성됩니다. (Solaris에만 해당) false를 선택하면 서버가 32비트 모드로 구성됩니다. (Solaris에만 해당) 기본값: false
WS_ADMIN_IS_SERVER_MODE	(선택 사항) 관리 구성 모드. (true/false). true를 선택하면 서버가 관리 서버 모드로 구성됩니다. false를 선택하면 서버가 관리 에이전트 모드로 구성됩니다. 기본값: true
WS_REGISTER_ADMIN_AGENT	(선택 사항) 원격 에이전트 등록. (true/false). WS_ADMIN_IS_SERVER_MODE가 false로 설정된 경우에만 필요합니다. true를 선택하면 원격 관리 서버 호스트를 등록해야 합니다. 기본값: true
WS_AGENT_SSL_PORT	(선택 사항) 관리 에이전트 SSL 포트. WS_ADMIN_IS_SERVER_MODE가 false로 설정된 경우에만 필요합니다.
WS_AGENT_HOST=	(선택 사항) 관리 에이전트 호스트 이름. WS_ADMIN_IS_SERVER_MODE가 false로 설정된 경우에만 필요합니다.

IV. configureServer 스크립트를 실행합니다.

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

configureServer 스크립트는 *hostName.domainName*이라는 기본 구성과 해당 Web Server 인스턴스를 만듭니다.

4. Web Server Administration Server 서비스를 시작합니다.

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

5. 릴리스 4 Web Server 인스턴스 구성을 릴리스 5 구성으로 마이그레이션합니다.

명령줄(wadm) 또는 그래픽 사용자 인터페이스 관리 도구(Web Server Admin Server GUI에 로그인)를 사용할 수 있습니다. 이후의 단계는 wadm 명령줄 인터페이스에 따라 달라집니다.

예를 들어, myinstance라는 인스턴스를 새 구성으로 마이그레이션하려면 다음을 수행합니다.

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

전체 명령 구문은 다음과 같습니다.

```
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])
```

위에서 첫 번째 명령 옵션 집합은 모든 wadm 명령에 공통이며 아래의 표 7-7에서 설명합니다. 두 번째 명령 옵션 집합은 migrate-server 명령 전용으로, 표 7-8에서 설명합니다.

첫 번째 명령 옵션 집합만 지정하고 wadm을 호출하면 wadm 명령 셸로 이동합니다. 이 셸에서 명령을 호출하면 공통 옵션을 다시 지정할 필요가 없습니다.

셸 외부에서 전체 wadm 명령을 지정하는 경우에는 최소한 --user 및 --host 옵션을 지정해야 합니다. --password-file 옵션을 생략하면 비밀번호를 묻는 메시지가 표시되며 다른 옵션을 생략하면 기본값이 사용됩니다. 이 장에서 절차를 설명하기 위해 사용된 명령의 경우 설명하기 쉽도록 --user 및 --host 옵션을 생략했습니다.

기본적으로 wadm은 포트 8989에서 SSL 프로토콜을 사용합니다.

wadm 명령 및 옵션에 대한 자세한 내용은 Web Server 7.0 CLI Reference Manual (<http://docs.sun.com/doc/819-3283>)을 참조하십시오.

표 7-7 wadm 공통 명령 옵션

옵션	설명
user	인증된 Web Server 관리자 아이디.
password-file	관리자를 Administration Server에 대해 인증하는 데 사용할 비밀번호가 포함된 파일. 이 비밀번호 파일에는 WADM_PASSWORD=password 줄이 포함되어야 합니다. 명령에서 이 옵션을 지정하지 않으면 비밀번호를 묻는 메시지가 표시됩니다.
host	Administration Server가 실행 중인 컴퓨터 이름. 기본값: localhost
echo	이 옵션을 true로 설정하면 명령을 실행하기 전에 표준 출력으로 명령 줄이 출력됩니다. 기본값: false
interactive	이 옵션을 true로 설정하면 필수 비밀번호 옵션을 표시합니다. 기본값: true
rcfile	wadm의 시작 시 로드하는 데 사용할 시작 파일. 기본값: ~/.wadmrc
no-prompt	이 옵션을 true로 설정하면 해당 명령이 어떤 상황에서도 사용자 입력을 요구하지 않습니다. 예를 들어 매개 변수가 누락된 채 호출한 경우에도 사용자 입력을 요청하고 기다리는 대신 간단히 오류 메시지를 출력합니다. wadm 명령을 셸 스크립트에서 사용하는 경우 사용자 입력을 기다리지 않고 항상 반환되도록 하려면 true로 설정할 수 있습니다. 기본값: false
verbose	true로 설정하면 자세한 목록이 표시됩니다. 기본값: false

표 7-8 wadm migrate-server 명령 옵션 및 피연산자

옵션/피연산자	설명
search-collection-copy-path	검색 컬렉션을 마이그레이션할 때 검색 컬렉션 색인 파일을 복사할 경로를 지정합니다. 다음 마이그레이션 시나리오가 가능합니다. Web Server 6.x 검색 컬렉션 경로가 Web Server 6.x 인스턴스 외부에 있는 경우 마이그레이션된 검색 컬렉션 경로는 Web Server 6.x 검색 컬렉션 경로를 가리키고 이 옵션은 무시됩니다. Web Server 6.x 검색 컬렉션 경로가 Web Server 6.x 인스턴스 내에 있고 이 옵션에 유효한 경로를 지정한 경우 검색 컬렉션 색인 파일은 <code>searchCollectionPath/configName/virtualServerName/collectionName</code> 디렉토리로 복사됩니다. 지정한 경로가 유효하지 않으면 오류 메시지가 기록됩니다. Web Server 6.x 검색 컬렉션 경로가 Web Server 6.x 인스턴스 내에 있지만 이 옵션에 경로를 지정하지 않은 경우에는 검색 컬렉션 색인 파일이 복사되지 않습니다. 마이그레이션 로그에는 사용자가 <code>wadm add-documents</code> 명령을 사용하여 수동으로 검색 컬렉션 색인 파일을 복사할 것인지 확인하는 메시지가 기록됩니다. 이 경우 마이그레이션되는 검색 컬렉션 경로는 <code>WebServer7Config-base/https-configName/config/collections/virtualServerName/collectionName</code>입니다.
log-dir	마이그레이션 로그의 위치. 기본값: <code>WebServer7Config-base/admin-serv/logs</code>
serverroot	Web Server 6.x 버전이 설치되는 설치 위치(디렉토리)이며 <code>WebServer6-base</code>와 같습니다.
all	<code>true</code>로 설정하면 모든 Web Server 6.x 인스턴스 구성이 해당 인스턴스와 동일한 이름의 Web Server 7.0 구성으로 마이그레이션됩니다. 해당 이름이 이미 있으면 <code>instanceName-1</code>이 구성 이름으로 사용됩니다. 기본값: <code>false</code>
instance	인스턴스 구성이 개별적으로 마이그레이션되는 경우(<code>all=false</code>), 마이그레이션할 Web Server 6.x 인스턴스 구성의 이름(형식: <code>https-instanceName</code>). 기본 Web Server 6.x 인스턴스 이름은 <code>hostName.domainName</code>입니다.
config	지정한 Web Server 6.x 인스턴스 구성이 마이그레이션될 구성의 이름. 기본값은 Web Server 6.x 인스턴스 구성의 <code>instanceName</code>을 사용하는 것입니다. 하지만 해당 이름의 구성이 이미 있으면 명령에서 이름 뒤에 정수를 붙입니다. 기본 Web Server 6.x 인스턴스 구성이 마이그레이션되는 경우 이런 상황이 발생할 수 있습니다.

migrate-server 명령을 사용할 때는 다음 고려 사항을 주의하십시오.

- 여러 릴리스 4 인스턴스 구성을 마이그레이션하는 경우 다른 --instance 값 및 해당하는 --config 인수를 사용하여 migrate-server 명령을 여러 번 실행하거나 --all 옵션을 사용하여 한 번에 마이그레이션할 수 있습니다.
- migrate-server 명령을 호출할 때마다 마이그레이션 작업에서 --log-dir 옵션으로 지정한 디렉토리(또는 기본 *WebServer7Config-base/admin-server/logs* 디렉토리)에 다음과 같은 이름의 로그 파일을 만듭니다.

```
MIGRATION_yyyymmddhhmmss.log
```

--all 옵션을 선택하면 로그 파일에는 마이그레이션된 모든 인스턴스에 대한 마이그레이션 정보가 저장됩니다.

- migrate-server 명령으로 마이그레이션되지 않는 데이터([136페이지의 "업그레이드 고려 사항"](#) 참조)에 대해서는 사용자가 직접 마이그레이션을 수행해야 합니다([147페이지의 "사후 업그레이드 작업"](#) 참조).

6. 릴리스 5 Web Server 인스턴스를 만듭니다.

[단계 5](#)에서 마이그레이션된 각 릴리스 4 인스턴스 구성에 대한 릴리스 5 인스턴스를 새로 만들어야 합니다.

- 새 인스턴스를 만들기 전에 마이그레이션 로그를 확인하여 마이그레이션된 구성에서 발생한 문제를 해결합니다.
- create-instance 명령을 실행합니다.

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

공통 명령 옵션은 [143페이지 표 7-7](#)에 정리되어 있습니다. create-instance 명령에만 해당되는 옵션은 다음 표에 정리되어 있습니다.

표 7-9 wadm create-instance 명령 옵션 및 피연산자

옵션/피연산자	설명
config	인스턴스가 가리켜야 하는 릴리스 5 구성의 이름
nodehost	인스턴스가 생성되는 컴퓨터의 이름. <i>hostName.domainName</i> 형식으로 여러 컴퓨터를 공백으로 분리된 목록으로 지정하여 동일한 인스턴스를 여러 개 만들 수 있습니다.

`create-instance` 명령은 지정된 노드의 `WebServer7Config-base/https-configName`에 인스턴스 디렉토리를 만들고 해당 인스턴스 디렉토리로 구성을 배포합니다.

7. 각 릴리스 5 인스턴스를 시작합니다.

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

인스턴스를 만들 때 `startserv` 스크립트가 생성됩니다. 인스턴스가 아무 문제 없이 시작되면 "서버 시작 성공"이라는 메시지와 함께 인스턴스의 기본 URL이 표시됩니다.

업그레이드 확인

다음 단계를 수행하여 Web Server가 릴리스 5로 업그레이드되었는지 확인할 수 있습니다.

1. 새로 만든 마이그레이션 로그 파일에서 `ERROR` 메시지가 있는지 확인합니다.
필요한 경우 수동으로 변경합니다([147페이지의 "사후 업그레이드 작업"](#) 참조).
2. 릴리스 5 Web Server 인스턴스를 확인합니다.
웹 브라우저에서 다음 URL에 액세스하여 시작 페이지가 표시되는지 확인합니다.
`http://hostName.domainName:port`
여기서 정규화된 호스트 이름과 포트는 각 인스턴스에 따라 달라집니다.
3. `-version` 옵션을 사용하여 Web Server 인스턴스를 실행합니다.
`WebServer7Config-base/https-configName/bin/startserv -version`
버전 출력 값에 대해서는 [135페이지 표 7-5](#)를 참조하십시오.

사후 업그레이드 작업

주요 사후 업그레이드 작업은 필요한 경우 특정 릴리스 4 데이터를 수동으로 마이그레이션하는 작업과 관련되어 있습니다. 이러한 데이터는 일반적으로 릴리스 4용으로 구성된 하나 이상의 가상 서버와 연관되어 있으며 `server.xml` 구성 파일에 지정되어 있습니다.

다음 상황을 해결하려면 사후 업그레이드 절차에 주의하십시오.

- [Web Server 6.1 docroot 콘텐츠 마이그레이션](#)
- [webdav 컬렉션 정보 마이그레이션](#)
- [로그 파일 마이그레이션](#)
- [타사 NSAPI 플러그인 마이그레이션](#)
- [검색 컬렉션 문서 루트 변경](#)
- [명령줄 스크립트 사용자 정의](#)

Web Server 6.1 docroot 콘텐츠 마이그레이션

1. Web Server 6.1 docroot 콘텐츠를 원하는 위치로 복사합니다.
2. 다음 명령을 사용하여 새 문서 루트 경로를 업데이트합니다.

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

공통 명령 옵션은 [143페이지 표 7-7](#)에 정리되어 있습니다.

`set-virtual-server-prop` 명령에만 해당되는 옵션은 다음 표에 정리되어 있습니다.

표 7-10 wadm set-virtual-server-prop 명령 옵션 및 피연산자

옵션 / 피연산자	설명
config	새 문서 루트 경로를 설정하려는 릴리스 5 구성의 이름
vs	마이그레이션된 문서 루트에 해당하는 가상 서버의 이름
document-root	새 문서 루트 디렉토리의 경로

3. 관련 Web Server 인스턴스로 구성을 재배포합니다.

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

공통 명령 옵션은 [143페이지 표 7-7](#)에 정리되어 있습니다. deploy-config 명령에만 해당되는 옵션은 다음 표에 정리되어 있습니다.

표 7-11 wadm deploy-cofig 명령 옵션 및 피연산자

옵션/피연산자	설명
force	true로 설명하면 이전 구성 배포 이후 수동으로 수정한 인스턴스 구성을 강제로 덮어 씁니다. 기본값: false
restart	true로 설정하면 배포된 구성에서 구성 설정을 가져오기 위해 실행 중인 인스턴스를 다시 시작합니다. 기본값: false
no-reconfig	true로 설정하면 실행 중인 인스턴스는 다시 시작할 때까지 배포된 구성에서 구성 설정을 가져오지 않습니다. 기본값: false
configName	인스턴스 이름이 구성 이름에 해당하는 Web Server 인스턴스에 배포할 릴리스 5 구성의 이름

webdav 컬렉션 정보 마이그레이션

추가적인 수동 마이그레이션은 필요하지 않으며 docroot 경로를 업데이트하는 것으로 충분합니다.

로그 파일 마이그레이션

이러한 파일을 저장하려면 해당 파일을 다른 위치로 복사합니다. 그렇지 않으면 설치된 릴리스 4를 제거할 때 해당 파일이 삭제됩니다.

타사 NSAPI 플러그인 마이그레이션

1. 해당 릴리스 4 위치에서 WebServer7-base/lib 디렉토리로 라이브러리 파일을 복사합니다.
2. magnus.conf 및 obj.conf 구성 파일을 임시 디렉토리로 내보냅니다.

```
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
```

공통 명령 옵션은 [143페이지 표 7-7](#)에 정리되어 있습니다.

3. 타사 NSAPI 플러그인 설명서에 지정된 대로 magnus.conf 및 obj.conf 파일을 수정합니다.

4. magnus.conf 및 obj.conf 구성 파일을 임시 디렉토리에 가져옵니다.

```
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
```

공통 명령 옵션은 [143페이지 표 7-7](#)에 정리되어 있습니다.

5. 관련 Web Server 인스턴스로 수정된 구성을 재배포합니다.

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

명령 옵션은 [148페이지 표 7-11](#)에 정리되어 있습니다.

검색 컬렉션 문서 루트 변경

migrate-server 명령에는 검색 컬렉션 정보를 마이그레이션하는 옵션이 있지만 다음과 같이 검색 컬렉션 문서 루트를 변경해야 할 수도 있습니다.

1. 검색 컬렉션의 문서 루트가 릴리스 4에서 사용된 것과 다르다면 다음 명령을 사용하여 검색 컬렉션의 문서 루트를 설정합니다.

```
WebServer7-base/bin/wadm set-search-collection-prop
--config=configName --vs=virtualServerName
--collection-name=searchCollectionName document-root=검색 컬렉션의 새
docroot 경로
```

공통 명령 옵션은 [143페이지 표 7-7](#)에 정리되어 있습니다.

set-search-collection-prop 명령에만 해당되는 옵션은 다음 표에 정리되어 있습니다.

표 7-12 wadm set-search-collection-prop 명령 옵션 및 피연산자

옵션/피연산자	설명
config	검색 컬렉션의 문서 루트를 설정하려는 릴리스 5 구성의 이름
vs	검색 컬렉션에 해당하는 가상 서버의 이름
collection-name	새 문서 루트 경로를 설정하려는 검색 컬렉션의 이름
document-root	검색 컬렉션의 새 문서 루트 디렉토리 경로

2. 관련 Web Server 인스턴스로 구성을 재배포합니다.

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

명령 옵션은 [148페이지 표 7-11](#)에 정리되어 있습니다.

명령줄 스크립트 사용자 정의

startsvr, startsvr.bat, stopsvr, stopsvr.bat, restart, reconfig 및 reconfig.bat와 같은 스크립트를 사용자 정의한 경우 *WebServer7-base/bin*에 있는 릴리스 5 기본 스크립트도 동일하게 사용자 정의해야 합니다.

업그레이드 롤백

릴리스 4 Web Server는 릴리스 5를 새로 설치하고 Web Server 인스턴스 구성을 마이그레이션해도 변경되지 않고 그대로 유지됩니다. 따라서 다음 단계에 따라 릴리스 4로 되돌리면 릴리스 5 Web Server를 롤백할 수 있습니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 실행 중인 모든 Web Server 인스턴스를 하나씩 중지합니다.

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

서버가 올바르게 중단되면 "서버가 종료되었습니다."라는 메시지가 표시됩니다.

3. 설치된 릴리스 5 Web Server를 제거합니다.

모든 릴리스 5 인스턴스와 마이그레이션된 구성을 제거해야 합니다.

- a. 모든 릴리스 5 인스턴스를 삭제합니다.

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

- b. 모든 릴리스 5 구성을 삭제합니다.

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

4. [139페이지의 "업그레이드 절차"](#)에 설명된 대로 Web Server 업그레이드 중에 중지한 Web Server 인스턴스를 다시 시작합니다.

Java ES 릴리스 3에서 Web Server 업그레이드

Java ES 2005Q1(릴리스 3) Web Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Web Server를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 3 Web Server를 릴리스 5로 업그레이드하려면 [134페이지의 "Java ES 릴리스 4에서 Web Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

Java ES 릴리스 2에서 Web Server 업그레이드

Java ES 2004Q2(릴리스 2) Web Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Web Server를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 2 Web Server를 릴리스 5로 업그레이드하려면 [134페이지의 "Java ES 릴리스 4에서 Web Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바꾸어 생각하십시오.

주	Linux 플랫폼에서 릴리스 2 Web Server를 업그레이드하는 경우 Web Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Web Server가 지원되지 않음). 자세한 내용은 133페이지의 "이중 업그레이드" 를 참조하십시오.

Java DB

이 장에서는 Java DB를 Java ES 5(릴리스 5): Java DB 10.1.3으로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- [154페이지의 "Java DB 업그레이드 개요"](#)
- [157페이지의 "Java ES 릴리스 4에서 Java DB 업그레이드"](#)

주 이 장에 설명된 파일 위치는 *JavaDB-base*라는 디렉토리 경로와 관련하여 지정되어 있습니다. 이 경로는 Java DB를 설치할 때 Java ES 설치 프로그램이 설정한 경로입니다.

다음 표에서는 이 디렉토리 경로의 값을 보여 줍니다.

표 8-1 Java DB 디렉토리 경로

경로 이름 변수	Solaris OS	Linux OS
<i>JavaDB-base</i>	/opt/SUNWjavadb	/opt/sun/javadb

Java DB 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Java DB의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 정보](#)
- [Java ES 릴리스 5 업그레이드 로드맵](#)
- [Java DB 데이터](#)
- [Java DB 업그레이드 전략](#)

Java ES 릴리스 5 정보

Java ES 릴리스 5 Java DB는 Java ES 제품 구성 요소로 제공되는 첫 번째 릴리스입니다. Java DB는 DerbyDatabase라는 이름으로 Java ES 릴리스 4에 포함된 공유 구성 요소로 처음 릴리스되었습니다.

릴리스 5 Java DB는 릴리스 4 버전과 비교했을 때 부차적 릴리스입니다. 여기에는 몇 가지 향상된 기능, 업데이트된 인터페이스 및 몇 가지 버그 수정이 포함되어 있습니다.

Java ES 릴리스 5 업그레이드 로드맵

[표 8-2](#)는 Java DB를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 8-2 Java ES 5(릴리스 5) 업그레이드 경로: Java DB 10.1.3

Java ES 릴리스	Java DB 버전	일반적인 방법	재구성 필요
릴리스 4	Derby Database 10.0.2	직접 업그레이드: 새로 설치하여 릴리스 4를 대체합니다. 지속성 데이터는 영향을 받지 않습니다.	없음

Java DB 데이터

다음 표는 Java DB 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 8-3 Java DB 데이터 사용

데이터 유형	위치	사용
구성 데이터	인스턴스 구성은 응용 프로그램에 따라 달라지며 Java DB 데이터베이스에 저장됩니다.	Java DB 인스턴스 구성
지속성 데이터	데이터베이스 디렉토리 및 해당 내용은 응용 프로그램에 따라 달라지며 디렉토리 위치는 데이터베이스 연결 URL에서 지정됩니다. jdbc:derby: <i>데이터베이스 전체 경로</i> .	데이터베이스 및 사용자 인증서

Java DB 업그레이드 전략

Java DB 업그레이드를 위한 전략은 일반적으로 [1장, "업그레이드 계획"](#)에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Java DB 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Java DB에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Java DB는 릴리스 4 버전과 호환됩니다.

종속성

Java DB는 J2SE 공유 구성 요소에 대해서만 종속성이 있습니다([47페이지 표 1-9](#) 참조).

이중 업그레이드

Java DB와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 다음 두 가지 방법 중 하나로 수행할 수 있습니다.

- 운영 체제 새로 설치
- 기존 운영 체제 업그레이드

운영 체제 새로 설치

1. 기존 Java DB 데이터를 백업합니다.
필수 데이터의 위치는 [155페이지의 "Java DB 데이터"](#)를 참조하십시오.
2. 새 운영 체제를 설치합니다.
운영 체제는 새 시스템 (또는 Solaris 10 영역)에 설치하거나 기존 파일 시스템을 덮어쓸 수 있습니다.
3. 릴리스 5 Java DB를 설치합니다.
4. [단계 1](#)에서 백업한 Java DB 데이터를 복원합니다.

기존 운영 체제 업그레이드

1. 기존 Java DB 데이터를 백업합니다.
필수 데이터의 위치는 [155페이지의 "Java DB 데이터"](#)를 참조하십시오.
2. 운영 체제를 업그레이드합니다.
업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.
3. 릴리스 5 Java DB로 업그레이드합니다.
[157페이지의 "Java ES 릴리스 4에서 Java DB 업그레이드"](#)를 참조하십시오.
Java DB 데이터는 업그레이드의 영향을 받지 않고 그대로 남아 있습니다.

Java ES 릴리스 4에서 Java DB 업그레이드

이 절에는 Java DB를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Java DB 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Java DB를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 릴리스 5 Java DB를 새로 설치하고 릴리스 4 버전을 대체하여 업그레이드를 수행할 수 있습니다. 릴리스 4 데이터 및 구성은 그대로 유지됩니다.
- **업그레이드 종속성.** Java DB는 J2SE 공유 구성 요소에 대한 엄격한 업그레이드 종속성([47페이지 표 1-9](#) 참조)이 있으며, 이 공유 구성 요소는 Java DB를 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다.
- **역방향 호환성.** 릴리스 5 Java DB는 릴리스 4와 완전히 호환됩니다.
- **업그레이드 롤백.** 백업된 릴리스 4 설치로 되돌리는 방법 이외에는 릴리스 5 업그레이드를 롤백하는 방법이 없습니다.
- **플랫폼 문제.** Java DB를 업그레이드하는 일반적인 방법은 Solaris 및 Linux 운영 체제에서 동일합니다.

릴리스 4 Java DB 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Java DB를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Java DB](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Java DB 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Java DB 종속성 업그레이드](#)
- [Java DB 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

다음 명령을 사용하여 Java DB의 현재 버전을 확인할 수 있습니다.

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

다음 표에는 derby.jar 파일의 버전 정보가 정리되어 있습니다.

표 8-4 Java DB 버전 확인 출력

Java ES 릴리스	Java DB 버전 번호
릴리스 4	10.0.2.1
릴리스 5	10.1.3.1

Java DB 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. 하지만 Java DB를 릴리스 5로 업그레이드하면 Java ES 설치 프로그램에 의해 Java DB에 필요한 모든 공유 구성 요소가 자동으로 설치됩니다.

Java DB 데이터 백업

Java DB를 릴리스 4에서 릴리스 5로 업그레이드하는 경우 구성 데이터 또는 지속성 데이터는 수정되지 않습니다. 그러나 보안을 유지하려면 전체 Java DB 설치 및 데이터를 백업해야 합니다. [155페이지 표 8-3](#)을 참조하십시오.

필수 구성 정보 및 비밀번호 얻기

Java DB 업그레이드에는 구성 정보 또는 비밀번호가 필요하지 않습니다.

릴리스 4 업그레이드 Java DB

이 절에서는 Solaris 및 Linux 플랫폼에서의 업그레이드 절차를 설명합니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 로컬에 상주하는 모든 Java DB 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 릴리스 4 Java DB를 중지합니다.

네트워크 서버가 실행되고 있는 경우에는 다음 명령을 사용합니다.

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

그렇지 않으면 간단히 Java DB를 사용하는 모든 응용 프로그램을 종료합니다.

3. 릴리스 5 Java DB를 새로 설치합니다.

다음 단계를 수행합니다.

- a. 릴리스 4 Java DB를 호스팅하는 컴퓨터에서 Java ES 설치 프로그램을 실행합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령 줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. 구성 요소 선택 페이지에서 Java DB를 선택합니다.

이전 설치를 덮어 쓰게 됩니다.

- c. "나중에 구성" 옵션을 선택합니다.

"지금 구성"은 지원되지 않습니다.

- d. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.

- e. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.

4. 릴리스 5 Java DB를 시작합니다.

네트워크 서버를 사용하고 있는 경우 다음 명령을 사용합니다.

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

그렇지 않으면 Java DB를 사용하는 모든 응용 프로그램을 간단히 포함 모드에서 시작하십시오.

업그레이드 확인

다음 명령을 사용하여 Java DB가 성공적으로 업그레이드되었는지 확인할 수 있습니다.

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

derby.jar 파일 버전에 대한 출력 값에 대해서는 [158페이지 표 8-4](#)를 참조하십시오.

사후 업그레이드 작업

Java DB를 릴리스 4에서 릴리스 5로 업그레이드하는 경우 Java DB 10.0 디스크 형식에서 10.1 형식으로 데이터를 변환해야 합니다. 이 변환을 수행하려면 JDBC URL에 `upgrade=true`를 추가하여 데이터베이스에 연결하십시오. 예를 들면 다음과 같습니다.

```
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;
```

업그레이드 롤백

릴리스 4 설치 및 해당 데이터의 백업으로 복원하는 방법 이외에는 릴리스 5 업그레이드를 롤백하는 방법이 없습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Java DB를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Java DB 인스턴스를 실행하여 로드를 분산시킬 수 있습니다.

[158페이지의 "릴리스 4 Java DB 업그레이드"](#)에서 설명하는 대로 각 컴퓨터에서 Java DB를 업그레이드하십시오.

고가용성 세션 저장소

이 장에서는 고가용성 세션 저장소를 Java ES 5(릴리스 5): 고가용성 세션 저장소 (HADB) 4.4.3으로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 방법에 대해 다루기 전에 업그레이드 문제 및 절차의 일반적인 개요를 제공합니다. 이 장에서는 Solaris와 Linux 운영 체제에서의 업그레이드에 대해 설명합니다.

- [164페이지의 "HADB 업그레이드 개요"](#)
- [166페이지의 "Java ES 릴리스 4에서 HADB 업그레이드"](#)
- [173페이지의 "Java ES 릴리스 3에서 HADB 업그레이드"](#)

주 이 장에 설명된 파일 위치는 *HADB-base*라는 디렉토리 경로에 대해 상대적으로 지정되어 있습니다. HADB를 처음 설치할 때 최소한 이 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 설치 프로그램에서 기본값을 할당합니다.

다음 표에 나와 있는 것처럼 *HADB-base*의 기본값은 운영 체제 플랫폼에 따라 달라지지 않습니다.

표 9-1 HADB 디렉토리 경로

경로 이름 변수	Solaris OS	Linux OS
<i>HADB-base</i>	<i>/opt/SUNWhadb/version_number</i>	<i>/opt/SUNWhadb/version_number</i>
Java ES 설치 프로그램		
<i>HADB-base</i>	<i>/opt/SUNWappserver/appserver</i>	<i>/opt/SUNWappserver/appserver</i>
독립 실행형 Application	<i>/hadb/version_number</i>	<i>/hadb/version_number</i>
Server 8.2 EE 설치 프로그램		

HADB 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 HADB의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 HADB 정보](#)
- [HADB 업그레이드 로드맵](#)
- [HADB 데이터](#)
- [HADB 업그레이드 전략](#)

Java ES 릴리스 5 HADB 정보

릴리스 4 HADB와 비교할 때 Java ES 릴리스 5 버전의 HADB는 사소한 사용자 기능이 향상되었습니다.

HADB 업그레이드 로드맵

[표 9-2](#)는 HADB를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 9-2 Java ES 5(릴리스 5) 업그레이드 경로: HADB 4.4.3

Java ES 릴리스	HADB 버전	일반적인 방법	재구성 필요
릴리스 4	HADB 4.4.2(2005Q4)	직접 업그레이드: 온라인과 오프라인 업그레이드 방법을 모두 사용할 수 있습니다.	없음
릴리스 3	HADB 4.4.1(2005Q1)	직접 업그레이드: 온라인과 오프라인 업그레이드 방법을 모두 사용할 수 있습니다.	없음
릴리스 2	HADB 4.4.0-14(2004Q2)	업그레이드가 지원되지 않습니다.	없음
릴리스 1	사용할 수 없음	업그레이드 불가능	없음
이전 Java ES 릴리스	사용할 수 없음	업그레이드 불가능	없음

HADB 데이터

다음 표는 HADB 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 9-3 HADB 데이터 사용

데이터 유형	위치	사용
동적 응용 프로그램 데이터	/var/opt/SUNWhadb	고가용성 세션 저장소
구성 데이터	/etc/opt/SUNWhadb /etc/init.d/ma-initd	고가용성 서버 구성

HADB 업그레이드 전략

HADB 업그레이드를 위한 전략은 일반적으로 [1장, "업그레이드 계획"](#)에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 HADB 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 HADB에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 HADB는 Java ES 릴리스 4와 함께 제공되는 HADB 버전과 호환됩니다.

HADB 종속성

릴리스 5 HADB는 Java™ 2 Platform, Standard Edition(J2SE™) 버전 1.4 이상의 J2SE 공유 구성 요소에 대해서만 종속성이 있습니다.

이중 업그레이드

HADB 소프트웨어와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 Application Server 이중 업그레이드와 연결하여 수행됩니다. Application Server 이중 업그레이드에 대한 자세한 내용은 [206페이지의 "이중 업그레이드"](#)를 참조하십시오.

Java ES 릴리스 4에서 HADB 업그레이드

이 절에는 HADB를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 HADB 업그레이드](#)

소개

Java ES 릴리스 4 HADB를 Java ES 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** Java ES 릴리스 4 HADB 패키지를 제거하고 Java ES 릴리스 5 패키지를 추가하여 업그레이드를 수행합니다. 두 가지 업그레이드 방법이 있습니다.
 - **온라인 업그레이드.** HADB 서비스를 중단하지 않으려면 온라인 업그레이드를 사용합니다.
 - **오프라인 업그레이드.** HADB 패키지를 새 버전으로 교체할 때 HADB 서비스를 중단해도 되는 경우에는 오프라인 업그레이드를 사용합니다.
- **업그레이드 종속성.** HADB에는 엄격한 업그레이드 종속성이 없습니다. HADB에는 J2SE 버전 1.4 이상이 필요하며 J2SE에 대한 엄격하지 않은 업그레이드 종속성이 있습니다.
- **역방향 호환성.** Java ES 릴리스 5에서 제공하는 HADB는 Java ES 릴리스 4에서 제공하는 HADB와 호환됩니다.
- **업그레이드 롤백.** Java ES 릴리스 4 버전 패키지를 복원하여 Java ES 릴리스 5 업그레이드를 릴리스 4로 롤백할 수 있습니다.
- **플랫폼 문제.** HADB를 업그레이드하는 일반적인 방법은 Solaris 및 Linux 운영 체제에서 동일합니다.

릴리스 4 HADB 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 HADB를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 HADB](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

HADB 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [HADB 종속성 업그레이드](#)
- [디렉토리 데이터 및 구성 파일 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

표준 버전 확인 유틸리티를 사용하여 HADB의 현재 버전을 확인할 수 있습니다. 예를 들면 다음과 같습니다.

Solaris의 경우

```
pkgparam -v SUNWhadba
```

Linux의 경우

```
rpm -qi sun-hadb-a-4.4.3-5.i386.rpm
```

표 9-4 HADB 버전 확인 출력

Java ES 릴리스	HADB 버전 번호
릴리스 2	VERSION=4.4.0,REV=14 SUNW_PRODVERS=4.4.0
릴리스 3	VERSION=4.4.1,REV=7 SUNW_PRODVERS=4.4.1
릴리스 4	VERSION=4.4.2,REV=7 SUNW_PRODVERS=4.4.2
릴리스 5	VERSION=4.4.3,REV=5 SUNW_PRODVERS=4.4.3

HADB 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. 하지만 HADB를 릴리스 5로 업그레이드하면 Java ES 설치 프로그램에 의해 HADB에 필요한 모든 공유 구성 요소(즉, J2SE)가 자동으로 설치됩니다.

디렉토리 데이터 및 구성 파일 백업

HADB를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드해도 HADB 동적 데이터 자체가 수정되는 것은 아닙니다. 그러나 업그레이드를 롤백해야 하는 경우를 위해 Java ES 릴리스 4 패키지를 백업할 수 있습니다.

또한 이전 HADB를 설치한 후 다음 파일을 수정한 경우에는 해당 파일을 백업해 두십시오.

```
/etc/opt/SUNWhadb/mgt.cfg
/etc/init.d/ma-initd
```

필수 구성 정보 및 비밀번호 얻기

HADB를 업그레이드하려면 슈퍼유저 비밀번호를 알아야 합니다.

릴리스 4 업그레이드 HADB

이 절에서는 HADB의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

HADB 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- 제품 요구 사항에 따라 온라인 업그레이드와 오프라인 업그레이드 중 어떤 것이 더 적절한지를 결정해야 합니다.
- 다음 표에서는 Solaris 및 Linux 플랫폼용 Java ES 릴리스 5 업그레이드 패키지에 대해 보여 줍니다. Solaris 패키지는 설치 순서대로 나열되어 있습니다.

표 9-5 Solaris 플랫폼에서 HADB 업그레이드를 위한 패키지 버전

Solaris 패키지	Linux 패키지
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

HADB 온라인 업그레이드

HADB를 온라인으로 업그레이드하는 경우 먼저 업그레이드하는 클러스터의 각 서버에서 릴리스 5 HADB를 설치, 시작 및 확인합니다. 그런 다음 HADB의 이전 설치에서 각 서버의 등록을 해제하고 새로 설치된 HADB 버전을 등록합니다.

온라인 업그레이드를 수행하는 방법에 대한 자세한 내용은 Sun Java System Application Server Enterprise Edition 8.2 고가용성 관리 설명서 (<http://docs.sun.com/app/docs/doc/820-0853/6ncijdjsu?a=view>)에서 다음 절을 참조하십시오.

HADB 오프라인 업그레이드

Java ES 릴리스 3 또는 릴리스 4에서 업그레이드하는 경우 HADB 오프라인 업그레이드를 사용할 수 있습니다.

오프라인으로 업그레이드하려면 HADB 서비스를 종료하고 기존 HADB 패키지를 [표 9-5 페이지 169](#)에 나와 있는 Java ES 5(릴리스 5) 배포판에서 사용 가능한 새 버전으로 교체합니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 모든 HADB 서비스를 종료합니다.

- a. 실행 중인 모든 데이터베이스를 나열합니다.

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

- b. 표시된 각 데이터베이스를 종료합니다.

```
HADB-base/bin/hadbm stop databaseName
```

데이터베이스가 이미 중지된 상태라는 메시지는 무시하십시오.

- c. 관리 에이전트를 실행하는 모든 호스트에서 HADB 관리 에이전트를 종료합니다.

```
/etc/init.d/ma-initd stop
```

Java ES 설치 프로그램 대신 독립 실행형 Application Server 설치 프로그램을 사용하여 HADB를 설치한 경우 ma-initd 스크립트는 *HADB-base/bin*에 있습니다.

3. Java ES 설치 프로그램을 시작합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

4. 구성 요소 선택 페이지에서 High Availability Session Store 4.4를 선택합니다.

Application Server Enterprise Edition 8.2를 이미 선택한 경우 HADB가 자동으로 선택됩니다.

5. 업그레이드 선택 항목을 확인합니다.

HADB 패키지가 업그레이드되고 업그레이드 요약이 표시됩니다.

6. Java ES 설치 프로그램을 종료합니다.

7. 168페이지의 "디렉토리 데이터 및 구성 파일 백업"에서 백업한 파일을 복원합니다.
8. 심볼릭 링크 /opt/SUNWhadb/4가 *HADB-base*를 가리키는지 확인합니다.

예를 들어 기본 *HADB-base*의 경우 다음과 같습니다.

```
ls -l /opt/SUNWhadb/4
lrwxrwxrwx 1 root other 7 Jul 7 23:18 /opt/SUNWhadb/4 ->
4.4.3-5/
```

9. 단계 2에서 종료된 HADB 관리 에이전트를 다시 시작합니다.

```
/etc/init.d/ma-initd start
```

Java ES 설치 프로그램 대신 독립 실행형 Application Server 설치 프로그램을 사용하여 HADB를 설치한 경우 *ma-initd* 스크립트는 *HADB-base/bin*에 있습니다.

업그레이드 확인

온라인 업그레이드를 완료한 후 다음 절차에 따라 업그레이드를 확인합니다. 성공적으로 업그레이드되었는지 확인한 다음에는 이전 설치 패키지를 삭제할 수 있습니다.

실행 중인 프로세스가 업그레이드된 HADB 서비스를 사용하고 있는지 확인하려면 다음 단계를 수행합니다.

1. 실행 중인 모든 HADB 서비스에 대해 다음 명령 중 하나를 실행합니다.

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

예를 들면 다음과 같습니다.

```
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. 데이터베이스가 실행되고 있는지 확인합니다.

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

예를 들어, 데이터베이스 이름이 Example DB인 경우 다음 명령을 입력합니다.

```
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

표시된 노드에 대한 모든 HADB 서비스는 "실행 중"인 상태여야 합니다.

3. HADB를 사용하는 모든 제품이 새 HADB 경로를 사용하는지 확인합니다.

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

예를 들어, 데이터베이스 이름이 Example DB인 경우 다음 명령을 입력합니다.

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

```
Attribute      Value
```

```
PackageName V4.4.3.5
```

위 명령을 실행하면 HADB의 현재 버전이 표시됩니다. 자세한 목록을 표시하려면 다음 명령을 실행하십시오.

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

사후 업그레이드 작업

사후 업그레이드 작업은 [166페이지](#)의 "Java ES 릴리스 4에서 HADB 업그레이드"에 설명된 단계에 모두 포함됩니다.

업그레이드 롤백

HADB 업그레이드를 롤백하려면 설치한 HADB 패키지의 최신 버전을 [168페이지](#)의 "디렉토리 데이터 및 구성 파일 백업"에서 설명한 대로 이전에 백업한 버전으로 바꾸면 됩니다.

Java ES 릴리스 3에서 HADB 업그레이드

Java ES 2005Q1(릴리스 3) HADB를 릴리스 5로 업그레이드하는 절차는 릴리스 4 HADB를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 3 HADB를 릴리스 5로 업그레이드하려면 [166페이지의 "Java ES 릴리스 4에서 HADB 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

Message Queue

이 장에서는 Message Queue 소프트웨어를 이전 Java ES 버전에서 Java ES 5(릴리스 5): Sun Java System Message Queue 3.7 UR1로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- [176페이지의 "Message Queue 업그레이드 개요"](#)
- [181페이지의 "Java ES 릴리스 4에서 Message Queue 업그레이드"](#)
- [187페이지의 "Java ES 릴리스 3에서 Message Queue 업그레이드"](#)
- [188페이지의 "Java ES 릴리스 2로부터 Message Queue 업그레이드"](#)

주 이 장에 설명된 파일 위치는 *MessageQueue-base*라는 고정 디렉토리 경로에 대해 상대적으로 지정되어 있습니다.

다음 표에 나와 있는 것처럼 *MessageQueue-base*의 값은 운영 체제 플랫폼에 따라 달라집니다.

표 10-1 Message Queue 디렉토리 경로

경로 이름 변수	Solaris OS	Linux OS
<i>MessageQueue-base</i>	/usr/bin	/opt/sun/mq/bin

Message Queue 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Message Queue의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Message Queue 정보](#)
- [Message Queue 업그레이드 로드맵](#)
- [Message Queue 데이터](#)
- [Message Queue 업그레이드 전략](#)

Java ES 릴리스 5 Message Queue 정보

Java ES 릴리스 5 Message Queue는 릴리스 4와 비교했을 때 부차적 릴리스이며 기능 향상보다는 코드 수정이 주로 포함되어 있습니다.

Message Queue 소프트웨어는 지금까지 기능 집합과 라이선스 수에 차이가 있는 Platform Edition과 Enterprise Edition의 두 가지 버전이 있었습니다. Enterprise Edition은 엔터프라이즈 작업 환경에서 메시징 응용 프로그램을 배포 및 실행하는 데 적합합니다. Platform Edition은 주로 메시징 응용 프로그램과 구성 요소를 개발, 디버깅 및 로드 테스트하는 데 사용됩니다. 릴리스 5 Message Queue는 Platform Edition을 더 이상 지원하지 않으며 Message Queue에 모든 Enterprise Edition 기능이 포함됩니다. 이전 Java ES 릴리스 버전에서 릴리스 5로 업그레이드하면 설치된 모든 Platform Edition이 완전한 Message Queue 엔터프라이즈 수준의 기능을 포함하도록 변환됩니다.

Message Queue 업그레이드 로드맵

표 10-2는 Message Queue를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 10-2 Java ES 5(릴리스 5) 업그레이드 경로: Message Queue 3.7 UR1

Java ES 릴리스	Message Queue 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Message Queue 2005Q4(3.6 SP3) Enterprise Edition만 해당	직접 업그레이드: Java ES 설치 프로그램을 사용하여 수행됩니다.	데이터 변환이 자동으로 수행됩니다.
릴리스 3	Sun Java System Message Queue 2005Q1(3.6) Enterprise Edition만 해당	직접 업그레이드: Java ES 설치 프로그램을 사용하여 수행됩니다.	데이터 변환이 자동으로 수행됩니다.
릴리스 2	Sun Java System Message Queue 2004Q2(3.5 SP1) Platform Edition 및 Enterprise Edition	직접 업그레이드: mqupgrade 스크립트를 사용하여 수행합니다.	Solaris 플랫폼에서는 자동으로 수행되며 Linux 플랫폼에서는 mmigrate 스크립트를 사용할 수 있습니다.
릴리스 1	Sun Java System Message Queue 2003Q4(3.0.1 SP2) Platform Edition 및 Enterprise Edition	직접 업그레이드 인증 안 됨: 그러나 mqupgrade 스크립트를 사용하여 수행할 수 있습니다. ¹	Solaris 플랫폼에서는 자동으로 수행되며 Linux 플랫폼에서는 mmigrate 스크립트를 사용할 수 있습니다.
이전 Java ES 릴리스	Sun Java System Message Queue 3.0.x 및 이전 버전 Platform Edition 및 Enterprise Edition	직접 업그레이드 인증 안 됨: 그러나 Java ES 설치 프로그램을 사용하여 수행할 수 있습니다.	

1. mqupgrade 스크립트를 실행하기 전과 후에 다음 파일을 백업하고 복원해야 할 수 있습니다. 예를 들어, Solaris OS에서는 /etc/imq/passwd 및 /etc/imq/accesscontrol.properties를 /var/imq/instances/instanceName/etc/에 복원해야 합니다.

표 10-2에 나와 있는 Message Queue의 Java ES 릴리스와 더불어 Message Queue도 Solaris OS 소프트웨어와 함께 번들로 제공됩니다. Message Queue의 번들 버전은 Java ES 설치 프로그램을 사용하여 릴리스 5로 업그레이드할 수 있습니다.

Message Queue 데이터

다른 Java ES 구성 요소처럼, Message Queue는 해당 업그레이드된 버전으로의 마이그레이션이 필요할 수 있는 다양한 데이터를 사용합니다. 다음 표는 Message Queue 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 10-3에서는 Solaris 시스템에서의 데이터 위치를 보여 줍니다. Linux 시스템의 경우도 비슷한 위치이지만 Message Queue 버전에 따라 달라집니다.

- 릴리스 2: 표의 /img를 /opt/img로 바꿉니다.
- 릴리스 3 이상: 표의 /img를 /opt/sun/mq로 바꿉니다.

자세한 내용은 Message Queue 3.7 UR1 관리 설명서

(<http://docs.sun.com/doc/819-4467/6n6k98br1?a=view>)를 참조하십시오.

표 10-3에서 *instanceName* 은 데이터가 연관된 Message Queue 브로커 인스턴스를 나타냅니다.

표 10-3 Message Queue 데이터 사용(Solaris OS)

데이터 범주	위치(Solaris)	사용
브로커 인스턴스 구성 등록 정보	/var/img/instances/ <i>instanceName</i> /props/config.properties	브로커 및 관련 서비스 구성
동적 응용 프로그램 데이터용 영구 저장소	릴리스 2, 릴리스 3 및 릴리스 4: /var/img/instances/ <i>instanceName</i> /fs350/ 릴리스 5: /var/img/instances/ <i>instanceName</i> /fs370/ 또는 JDBC 액세스 가능 데이터 저장소	메시지, 대상, 영구 가입, 트랜잭션 및 기타 동적 데이터 저장
관리 객체 (객체 저장소)	사용자가 선택한 로컬 디렉토리 또는 LDAP Directory Server	클라이언트/브로커 연결 구성에 사용되는 객체
보안: 사용자 저장소	/var/img/instances/ <i>instanceName</i> /etc/passwd 또는 LDAP 디렉토리 서버	인증 및 권한 부여에 사용되는 사용자 데이터 저장
보안: 액세스 제어 파일 (기본 위치)	/var/img/instances/ <i>instanceName</i> /etc/accesscontrol.properties	대상 및 관련 기능에 대한 사용자 액세스 권한을 부여하는 규칙 설정
보안: passfile 디렉토리 (기본 위치)	/var/img/instances/ <i>instanceName</i> /etc/	암호화된 비밀번호 정보 저장
보안: 브로커의 키 저장소 파일 위치	/etc/img/	보안 메시지를 위한 암호화된 인증서 정보 저장

Message Queue 업그레이드 전략

Message Queue 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 문제에 따라 달라집니다.

이 절에서는 Message Queue 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Message Queue에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Message Queue에는 릴리스 3 또는 릴리스 4에 대한 새로운 호환성 문제가 없지만 릴리스 2 및 이전 버전에 대해서는 많은 호환성 문제가 있습니다. 이 문제에 대해서는 189페이지의 "릴리스 2 호환성 문제"를 참조하십시오.

또한 일반적으로 릴리스 4 및 이전 Message Queue 브로커와 릴리스 5 Message Queue 브로커를 한 클러스터에서 함께 사용하는 경우 마스터 브로커는 이전 릴리스의 브로커가 되어야 하며, 이 경우 클러스터는 이전 릴리스의 Message Queue 클러스터로 실행됩니다.

Message Queue 종속성

다른 Java ES 구성 요소에 대한 Message Queue 종속성은 Message Queue 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어, Message Queue 인터페이스 또는 기능의 변경 사항을 적용하려면 Message Queue이 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Message Queue는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Message Queue은 특정한 Java ES 공유 구성 요소(47페이지 표 1-9 참조)에 종속됩니다.
- **Directory Server.** Message Queue는 Directory Server에 대해 선택적 종속성이 있으며, 관리 객체 또는 사용자 데이터를 로컬이 아닌 LDAP 디렉토리(Directory Server)에 저장하도록 Message Queue를 구성할 수 있습니다.
- **웹 컨테이너.** Message Queue는 클라이언트 및 브로커 간의 HTTP 메시징을 지원하는 Web Server, Application Server 또는 타사 웹 컨테이너에 선택적 종속성이 있습니다.
- **데이터베이스.** Message Queue는 Message Queue 지속성 계층에 플랫폼 파일 메시지 저장소가 아닌 JDBC 액세스 가능 데이터 저장소를 제공하는 Java DB(또는 타사 데이터베이스)에 선택적 종속성이 있습니다.
- **Sun Cluster.** Message Queue는고가용성 지원을 제공하는 Sun Cluster에 대해 선택적 종속성이 있습니다.

이중 업그레이드

Message Queue와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 다음 두 가지 방법 중 하나로 수행할 수 있습니다.

- 운영 체제 새로 설치
- 기존 운영 체제 업그레이드

운영 체제 새로 설치

1. 기존 Message Queue 데이터를 백업합니다.

필수 데이터의 위치는 [178페이지의 "Message Queue 데이터"](#)를 참조하십시오.

2. 새 운영 체제를 설치합니다.

운영 체제는 새 시스템 (또는 Solaris 10 영역)에 설치하거나 기존 파일 시스템을 덮어쓸 수 있습니다.

3. 릴리스 5 Message Queue를 설치합니다.

4. [단계 1](#)에서 백업한 Message Queue 데이터를 복원하거나 마이그레이션합니다.

Liunix에서 릴리스 2 Message Queue를 업그레이드하는 경우 해당 데이터는 릴리스 5 위치에 복원됩니다.

기존 운영 체제 업그레이드

1. 기존 Message Queue 데이터를 백업합니다.

필수 데이터의 위치는 [178페이지의 "Message Queue 데이터"](#)를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Message Queue로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 해당 절을 참조하십시오. 업그레이드해도 기존 Message Queue 데이터는 그대로 남아 있습니다.

하지만 Linux에서 릴리스 2 Message Queue를 업그레이드하는 경우에는 해당 데이터를 릴리스 5 위치로 이동해야 합니다.

Java ES 릴리스 4에서 Message Queue 업그레이드

이 절에는 Message Queue를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Message Queue 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Message Queue를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** Java ES 설치 프로그램을 사용하여 업그레이드를 수행합니다. 설치 프로그램은 릴리스 4에서 자동으로 구성 데이터를 마이그레이션합니다. 또한 릴리스 4와 연결된 모든 동적 응용 프로그램 데이터는 `imqbrokerd`가 처음 실행될 때 자동으로 변환됩니다. 파일 기반 저장소의 경우 `fs350` 디렉토리의 내용이 새 `fs370` 디렉토리로 복사됩니다. JDBC 저장소의 경우에는 기존 데이터베이스 테이블에 대해 간단한 버전 업데이트가 수행됩니다.
- **업그레이드 종속성.** Message Queue은 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Message Queue을 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다.

또한 릴리스 5 Message Queue는 [179페이지의 "Message Queue 종속성"](#)에서 설명한 대로 Java ES 제품 구성 요소에 대한 종속성이 있습니다. 그러나 Message Queue를 릴리스 5로 업그레이드하기 위해 이러한 구성 요소를 업그레이드해야 하는 것은 아닙니다.

- **역방향 호환성.** 릴리스 5 Message Queue는 프로토콜, 브로커 호환성, 관리 객체, 관리 도구 및 클라이언트 응용 프로그램 측면에서 릴리스 4와 완전하게 호환됩니다.
- **업그레이드 롤백.** 릴리스 4 Message Queue로 업그레이드를 롤백하는 유틸리티는 없습니다. 업그레이드된 구성 요소를 제거하고 이전 버전 및 구성 데이터를 수동으로 복원해야 합니다.
- **플랫폼 문제.** Message Queue를 업그레이드하는 방법은 Solaris와 Linux 운영 체제에서 동일합니다.

릴리스 4 Message Queue 업그레이드

이 절에서는 Message Queue를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드 하는 방법을 설명합니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Message Queue](#)
- [Message Queue 업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Message Queue 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Message Queue 종속성 업그레이드](#)
- [백업 Message Queue](#)

현재 버전 정보 확인

다음과 같이 -version 옵션을 사용하여 Message Queue 브로커를 시작하면 시스템에 설치된 Message Queue 버전 정보를 확인할 수 있습니다.

```
imqbrokerd -version
```

표 10-4 Message Queue 버전 확인 출력

Java ES 릴리스	Message Queue 버전 번호
릴리스 2	Sun Java (tm) System Message Queue 3 2004Q2 버전: 3.5
릴리스 3	Sun Java (tm) System Message Queue 3 2005Q1 버전: 3.6
릴리스 4	Sun Java (tm) System Message Queue 3 2005Q4 버전: 3.6 SP3
릴리스 5	Sun Java (tm) System Message Queue 3.7 버전: 3.7 UR1

Message Queue 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Message Queue는 몇몇 공유 구성 요소에 대해서만 엄격한 업그레이드 종속성이 있습니다.

Message Queue 종속성을 업그레이드할 때는 Message Queue을 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다. 공유 구성 요소 업그레이드는 Java ES 설치 프로그램을 통해 자동으로 수행됩니다.

1. **공유 구성 요소.** 63페이지의 "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Message Queue에 필요한 모든 공유 구성 요소는 Message Queue을 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.
2. **Sun Cluster(엄격하지 않은 업그레이드 종속성).** 75페이지의 3장, "Sun Cluster 소프트웨어"에서 Sun Cluster를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **Directory Server(엄격하지 않은 업그레이드 종속성).** 99페이지의 5장, "Directory Server"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
4. **Java DB(엄격하지 않은 업그레이드 종속성).** Message Queue를 업그레이드할 때 릴리스 5 Java DB를 새로 설치해야 합니다.
5. **웹 컨테이너 소프트웨어(엄격하지 않은 업그레이드 종속성).** Web Server 또는 Application Server의 업그레이드 지침은 129페이지의 7장, "Web Server" 및 199페이지의 11장, "Application Server"에서 각각 제공됩니다.

백업 Message Queue

업그레이드를 수행하기 전에 항상 프로덕션 환경의 응용 프로그램 데이터를 백업하는 것이 좋습니다. 178페이지 표 10-3에 나와 있는 동적 응용 프로그램 데이터의 영구 저장소 위치를 확인하십시오.

릴리스 4 업그레이드 Message Queue

업그레이드 절차는 다음 단계로 구성됩니다.

1. 실행 중인 모든 Message Queue 클라이언트 응용 프로그램을 중지합니다.
Application Server 환경에서 Message Queue를 사용 중인 경우 Application Server도 종료합니다.

2. 실행 중인 브로커를 중지합니다. 관리자 이름과 비밀번호를 묻는 메시지가 표시 됩니다.

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

3. 각 브로커 인스턴스와 관련된 동적 데이터, Message Queue 플랫폼 파일 사용자 저장소 및 Message Queue 액세스 제어 파일을 유지하지 않으려면 다음 명령을 사용하여 해당 데이터를 제거합니다.

```
imgbrokerd -name instanceName -remove instance
```

그렇지 않으면 동적 데이터 및 구성 정보는 유지되며 릴리스 5 Message Queue에서 사용됩니다.

4. 루트로 로그인합니다.

```
su -
```

5. Java ES 설치 프로그램을 시작합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

6. 구성 요소 선택 페이지에서 Message Queue를 선택합니다.
7. 업그레이드 선택 항목을 확인합니다.

Message Queue 패키지가 업그레이드되고 업그레이드 요약이 표시됩니다.

8. Java ES 설치 프로그램을 종료합니다.

Message Queue 업그레이드 확인

업그레이드 절차를 마친 뒤 작업이 성공했는지를 확인하려면 `-version` 옵션을 지정하고 Message Queue 브로커를 시작하면 됩니다.

```
imgbrokerd -version
```

이 명령은 Java ES 버전 번호와 Message Queue별 버전 번호를 반환합니다.

사후 업그레이드 작업

웹 컨테이너를 업그레이드했고 Message Queue HTTP 터널링 서블릿을 사용 중인 경우 이를 새 웹 컨테이너에 다시 배포해야 할 수 있습니다. HTTP 터널링 서블릿은 릴리스 4와 릴리스 5 간에 변경된 사항이 없습니다. HTTP 지원에 대한 자세한 내용은 Message Queue 3.7 UR1 관리 설명서(<http://docs.sun.com/doc/820-0845>)를 참조하십시오.

또한 업그레이드를 롤백할 필요가 없다고 생각하는 경우 fs350 디렉토리에 있는 릴리스 4 파일 기반 데이터 저장소를 제거할 수 있습니다(178페이지 표 10-3 참조).

업그레이드 롤백

Message Queue를 업그레이드 전 상태로 롤백할 수 있는 스크립트는 없습니다. 롤백 프로세스는 다음 단계에 따라 수동으로 수행해야 합니다.

1. 실행 중인 모든 Message Queue 클라이언트 응용 프로그램을 중지합니다.
2. 실행 중인 브로커를 중지합니다. 관리자 이름과 비밀번호를 묻는 메시지가 표시됩니다.

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

3. 각 브로커 인스턴스와 관련된 동적 데이터, Message Queue 플랫 파일 사용자 저장소 및 Message Queue 액세스 제어 파일을 삭제하려면 다음 명령을 사용하여 해당 데이터를 제거합니다.

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

4. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

5. 다음 명령을 사용하여 설치된 Message Queue 패키지 목록을 검색합니다.

Solaris의 경우

```
pkginfo | grep -i "message queue"
```

Linux의 경우

```
rpm -qa | grep mq
```

6. 다음 명령을 사용하여 Message Queue 패키지를 제거합니다.

Solaris의 경우

`pkgrm packageName`

여기서 *packageName*은 Message Queue 패키지 중 하나입니다. 여러 패키지를 제거하려면 패키지 이름을 공백으로 구분합니다.

Linux의 경우

`rpm -e --nodeps RPMName`

여기서 *RPMName*은 Message Queue RPM 구성 요소 중 하나입니다. 여러 구성 요소를 제거하려면 RPM 이름을 공백으로 구분합니다.

다른 제품에서 Message Queue 패키지를 사용할 수 있기 때문에 제거하는 데 신중해야 합니다. 패키지를 제거하기 전에 `pkgrm` 명령에서 패키지의 종속성을 경고합니다. 메시지가 표시되면 **y**(예)를 입력하여 제거 요청을 확인합니다.

7. 끝내려면 "q"를 입력합니다.
8. 루트 셸을 종료합니다.
9. 릴리스 4 Message Queue를 다시 설치합니다.

Java ES 릴리스 4 설치 프로그램을 사용합니다.

10. [183페이지의 "백업 Message Queue"](#)에서 백업한 릴리스 4 Message Queue 데이터를 복원합니다.

릴리스 4 Message Queue는 릴리스 5로 업그레이드하기 전에 백업한 데이터를 사용하여 문제 없이 작업할 수 있습니다.

여러 인스턴스 업그레이드

여러 브로커가 상호 작용하여 확장 가능한 메시지 서비스를 제공하는 Message Queue 클러스터를 업그레이드하려면 각 Message Queue 인스턴스가 릴리스 4에서 릴리스 5로 업그레이드되는 동안 클러스터가 온라인 상태로 유지되는 롤링 업그레이드를 수행할 수 있습니다. 클러스터를 업그레이드할 때는 다음과 같은 두 가지 조건을 고려해야 합니다.

- 지속성 메시지는 업그레이드를 위해 브로커가 종료된 동안 저장되며 브로커가 다시 시작될 때까지 사용할 수 없습니다.
- 마스터 브로커는 마지막으로 업그레이드해야 합니다.

그렇지 않은 경우 절차는 간단합니다. 즉 브로커가 모두 업그레이드될 때까지 한 번에 하나씩 브로커를 종료하고 업그레이드한 다음 재시작합니다.

Java ES 릴리스 3에서 Message Queue 업그레이드

Java ES 2005Q1(릴리스 3) Message Queue를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Message Queue를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 3 Message Queue를 릴리스 5로 업그레이드하려면 [181페이지의 "Java ES 릴리스 4에서 Message Queue 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

Java ES 릴리스 2로부터 Message Queue 업그레이드

이 절에는 Message Queue를 Java ES 2004Q2(릴리스 2)에서 Java ES 릴리스 5로 업그레이드하기 위해 필요한 정보가 포함되어 있으며 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 2 호환성 문제](#)
- [릴리스 2 Message Queue 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

주 Linux 플랫폼에서 릴리스 2 Message Queue를 업그레이드하는 경우 Message Queue *와* 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Message Queue가 지원되지 않음). 자세한 내용은 [180페이지의 "이중 업그레이드"](#)를 참조하십시오.

소개

Java ES 릴리스 2 Message Queue를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 업그레이드를 수행하는 데는 소프트웨어 패키지를 새로운 패키지로 교체하고 릴리스 2에서 구성 데이터를 자동으로 마이그레이션하는 mqupgrade 스크립트가 사용됩니다.
- **업그레이드 종속성.** 릴리스 2에서는 한 컴퓨터에서 임의의 Java ES 구성 요소를 업그레이드하는 경우 해당 컴퓨터가 호스팅하는 다른 모든 Java ES 구성 요소를 업그레이드해야 합니다. Java ES 구성 요소를 릴리스 2에서 릴리스 5로 선택적으로 업그레이드할 수는 없습니다. 특히 Message Queue에 사용되는 모든 Java ES 공유 구성 요소를 업그레이드해야 합니다.

또한 릴리스 5 Message Queue는 [179페이지의 "Message Queue 종속성"](#)에 설명한 대로 Directory Server 및 Web Server(또는 Application Server)에 선택적으로 종속됩니다. 이러한 구성 요소가 같은 컴퓨터에 설치된 경우 이러한 구성 요소도 릴리스 5로 업그레이드해야 합니다.

- **역방향 호환성.** 릴리스 5 Message Queue는 "[릴리스 2 호환성 문제](#)"에서 설명한 대로 릴리스 2와 완전히 호환되지 않습니다.

- **업그레이드 롤백.** 릴리스 5에서 릴리스 2로 롤백은 현재 지원되지 않습니다(["업그레이드 롤백"](#) 참조).
- **플랫폼 문제.** Message Queue를 업그레이드하는 일반적인 방법은 Solaris와 Linux 운영 체제에서 동일하지만 Linux에서는 약간의 추가 절차가 필요합니다. 다음 절차에 플랫폼별 명령과 적절한 파일 위치 또는 절차가 나와 있습니다.

릴리스 2 호환성 문제

릴리스 5 Message Queue에는 릴리스 2 및 이전 버전과 관련하여 다음과 같은 일반적인 Message Queue 호환성 문제가 있습니다.

프로토콜 호환성

Message Queue에는 Message Queue 클라이언트와 브로커 간의 HTTP 프로토콜 지원을 제공하는 웹 컨테이너에 대해 종속성이 있습니다. 프로토콜 변경으로 인해 Sun Java System Web Server를 사용하여 Message Queue `mqhttp.war` 응용 프로그램에 대한 웹 컨테이너를 제공하는 경우 Message Queue를 업그레이드해야 Web Server 구성 요소를 업그레이드할 수 있습니다([185페이지](#) 및 [196페이지](#)의 ["사후 업그레이드 작업"](#) 참조).

브로커 호환성

릴리스 5 Message Queue 브로커는 릴리스 4, 릴리스 3, 또는 릴리스 2 브로커와 상호 운영되지만 릴리스 2와 관련된 브로커 등록 정보와 영구 저장소 스키마의 변경 사항으로 인해 호환성에 문제가 있을 수 있습니다.

릴리스 5 Message Queue는 릴리스 4, 릴리스 3 및 릴리스 2 데이터를 사용할 수 있지만 Linux 시스템의 경우 먼저 릴리스 2 데이터를 릴리스 5로 마이그레이션해야 합니다.

릴리스 5 Message Queue 업데이트 시에는 다음을 고려해야 합니다.

- 이전 Message Queue `config.properties` 파일을 사용할 수 있습니다. 또한 파일을 다른 위치에 복사하고 릴리스 5 Message Queue 브로커를 구성할 때 이 파일에 포함된 등록 정보 설정을 참조할 수 있습니다.
- 브로커를 처음 시작할 때 영구 Message Queue 데이터(메시지, 대상, 영구 가입)는 필요한 경우 릴리스 5 Message Queue 데이터로 자동 변환됩니다. 예를 들어, 기존 대상은 필요한 경우 기존 속성을 유지하고 새 속성의 기본값을 사용하여 릴리스 5 Message Queue 대상으로 변환됩니다.

- 한 클러스터에 Message Queue 릴리스 2 브로커와 Message Queue 릴리스 5 브로커를 함께 사용하는 경우 마스터 브로커는 더 오래된 Message Queue 릴리스 2 브로커여야 하며, 이 클러스터는 Message Queue 릴리스 2 클러스터로 실행됩니다.

관리 객체 호환성

릴리스 5 Message Queue 관리 객체는 릴리스 3 및 릴리스 4 관리 객체와 동일합니다. 그러나 일부 릴리스 3 관리 객체는 이름이 변경되었거나 이전 버전과 관련된 새로운 속성으로 향상되었습니다. 따라서 릴리스 2 Message Queue에서 릴리스 5로 업그레이드하는 경우 다음을 고려해야 합니다.

- 릴리스 2에서 생성한 객체 저장소와 관리 객체를 그대로 사용할 수 있지만 관리 객체는 릴리스 5로 마이그레이션하는 것이 좋습니다. 업그레이드를 수행할 때 관리 콘솔(imqadmin) 및 ObjectManager 명령줄 유틸리티(imqobjmgr)는 릴리스 2 관리 객체를 릴리스 5 관리 객체로 변환합니다.
- 릴리스 5 클라이언트 런타임은 릴리스 2 관리 객체를 검색하고 인스턴스화하며 릴리스 5 클라이언트에서 사용하도록 변환합니다. 그러나, 검색이 이루어진 객체 저장소에 있는 릴리스 2 관리 객체는 변환되지 *않습니다*.
- 기존 릴리스 2 클라이언트(응용 프로그램 및 구성 요소), 즉 관리 객체를 검색하지 않고 직접 인스턴스화하는 클라이언트는 릴리스 5와 호환됩니다. 그러나 *새* 관리 객체 속성을 사용하려면 해당 클라이언트를 다시 작성해야 합니다. 관리 객체 속성에 대한 자세한 내용은 Message Queue 3.7 UR1 관리 설명서 (<http://docs.sun.com/doc/820-0845>)의 16장을 참조하십시오. 릴리스 2 클라이언트를 릴리스 5로 다시 컴파일하면 Message Queue 릴리스 5에서 이름이 바뀐 릴리스 2 속성이 표시됩니다. 이전 이름도 그대로 작동합니다.
- Java 클라이언트를 시작하고 명령줄 옵션을 사용하여 관리 객체 속성 값을 설정하는 스크립트는 릴리스 5와 호환됩니다. 그러나 *새* 관리 객체 속성을 사용하려면 해당 스크립트를 다시 작성해야 합니다. 관리 객체 속성에 대한 자세한 내용은 Message Queue 3.7 UR1 관리 설명서 (<http://docs.sun.com/doc/820-0845>)의 16장을 참조하십시오.

관리 도구 호환성

릴리스 3에 새 명령 및 관리 기능이 추가되었기 때문에 릴리스 5 관리 도구(관리 콘솔 및 명령줄 유틸리티)는 릴리스 3, 릴리스 4 및 릴리스 5 브로커와 함께만 작동합니다. 그러나 릴리스 2 명령 및 명령 옵션은 모두 계속 지원됩니다.

클라이언트 호환성

릴리스 3 및 릴리스 4 클라이언트는 릴리스 5 Message Queue와 완전히 호환됩니다. 그러나 릴리스 2에서 릴리스 5로 업그레이드하는 경우 Java 클라이언트와 관련된 다음 호환성 문제를 고려해야 합니다.

- 릴리스 5 브로커는 릴리스 2 클라이언트를 지원하지만 추가 릴리스 5 기능은 제공하지 않습니다.
- 릴리스 5 Java 클라이언트는 릴리스 2 브로커에 연결할 수 있지만 추가 릴리스 5 기능은 제공하지 않습니다.
- C 클라이언트 프로그램은 릴리스 5에서 지원되며 Platform Edition 체험판 사용권 또는 Enterprise Edition 사용권으로 실행하는 릴리스 2, 릴리스 3 또는 릴리스 4 브로커에서 지원됩니다.

릴리스 2 Message Queue 업그레이드

이 절에서는 Message Queue를 Java ES 릴리스 2에서 Java ES 릴리스 5로 업그레이드하는 방법을 설명합니다.

- [사전 업그레이드 작업](#)
- [릴리스 2 Message Queue 업그레이드\(Solaris\)](#)
- [릴리스 2 Message Queue 업그레이드\(Linux\)](#)
- [호환성 패키지 설치\(Linux\)](#)
- [Message Queue 업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Message Queue 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Message Queue 종속성 업그레이드](#)
- [백업 Message Queue](#)

현재 버전 정보 확인

다음과 같이 `-version` 옵션을 사용하여 Message Queue 브로커를 시작하면 시스템에 설치된 Message Queue 버전 정보를 확인할 수 있습니다.

```
imqbrokerd -version
```

표 10-5 Message Queue 버전 확인 출력

Java ES 릴리스	Message Queue 버전 번호
릴리스 2	Sun Java (tm) System Message Queue 3 2004Q2 버전: 3.5
릴리스 3	Sun Java (tm) System Message Queue 3 2005Q1 버전: 3.6
릴리스 4	Sun Java (tm) System Message Queue 3 2005Q4 버전: 3.6 SP3
릴리스 5	Sun Java (tm) System Message Queue 3.7 버전: 3.7 UR1

Message Queue 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Message Queue는 몇몇 공유 구성 요소에 대해서만 엄격한 업그레이드 종속성이 있습니다.

Message Queue 종속성을 업그레이드할 때는 Message Queue를 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다.

1. **공유 구성 요소.** 63페이지의 2장, "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
2. **Sun Cluster(엄격하지 않은 업그레이드 종속성).** 75페이지의 3장, "[Sun Cluster 소프트웨어](#)"에서 Sun Cluster를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **Directory Server(엄격하지 않은 업그레이드 종속성).** 99페이지의 5장, "[Directory Server](#)"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
4. **Java DB(엄격하지 않은 업그레이드 종속성).** Message Queue를 업그레이드할 때 릴리스 5 Java DB를 새로 설치해야 합니다.

5. **웹 컨테이너 소프트웨어(엄격하지 않은 업그레이드 종속성).** Web Server 또는 Application Server의 업그레이드 지침은 [129페이지의 7장, "Web Server"](#) 및 [199페이지의 11장, "Application Server"](#)에서 각각 제공됩니다.

백업 Message Queue

업그레이드를 수행하기 전에 항상 프로덕션 환경의 응용 프로그램 데이터를 백업하는 것이 좋습니다. Solaris OS의 경우 다음 디렉토리에 동적 데이터가 저장됩니다.
`/var/imq/instances/instanceName.`

다른 운영 체제의 경우 Message Queue 3.7 UR1 관리 설명서 (<http://docs.sun.com/doc/820-0845>)를 참조하십시오.

릴리스 2 Message Queue 업그레이드(Solaris)

Message Queue 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 릴리스 5 패키지를 설치하는 mqupgrade 스크립트가 사용됩니다.

업그레이드 절차는 다음 단계로 구성됩니다.

1. 실행 중인 모든 Message Queue 클라이언트 응용 프로그램을 중지합니다.
Application Server 환경에서 Message Queue를 사용 중인 경우 Application Server도 종료합니다.
2. 실행 중인 브로커를 중지합니다. 관리자 이름과 비밀번호를 묻는 메시지가 표시됩니다.

```
imqcmd shutdown bkr [-b hostName:port]
```
3. 각 브로커 인스턴스와 관련된 동적 데이터, Message Queue 플랫폼 파일 사용자 저장소 및 Message Queue 액세스 제어 파일을 유지하지 않으려면 다음 명령을 사용하여 해당 데이터를 제거합니다.

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


그렇지 않으면 동적 데이터 및 구성 정보는 유지되며 릴리스 5 Message Queue에서 사용됩니다.
4. 루트로 로그인합니다.

```
su -
```

5. 디렉토리를 Java ES 릴리스 5 배포의 Tools 디렉토리로 변경합니다.

Solaris SPARC의 경우

```
cd Solaris_sparc/Product/message_queue/Tools
```

Solaris x86의 경우

```
cd Solaris_x86/Product/message_queue/Tools
```

6. mqupgrade 스크립트를 실행합니다.

- a. 스크립트를 시작합니다.

```
./mqupgrade
```

mqupgrade 스크립트는 설치된 Message Queue 구성 요소를 나열합니다.

- b. y(예)를 입력하여 Message Queue 구성 요소를 업그레이드합니다.

mqupgrade 스크립트는 설치된 현지화 파일을 검색하여 나열합니다.

Message Queue 구성 요소를 업그레이드하지 않으려면 n(아니요)을 입력합니다. mqupgrade 스크립트는 Message Queue 구성 요소를 업그레이드하지 않고 종료합니다.

- c. 메시지가 표시되면 y(예)를 입력하여 현지화 파일을 업그레이드합니다.

mqupgrade 스크립트는 다음 위치의 로그 파일에 출력을 전송합니다.

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

릴리스 2 Message Queue 업그레이드(Linux)

RHEL 2.1에서만 Java ES 릴리스 2가 지원되고 Java ES 릴리스 5는 지원되지 않기 때문에 Linux 플랫폼에서 Message Queue 릴리스 2를 릴리스 5로 업그레이드하는 작업은 다소 까다롭습니다. 따라서 운영 체제와 Message Queue를 함께 업그레이드하는 이중 업그레이드를 해야 합니다. [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)를 참조하십시오.

기본적인 절차는 Linux OS를 먼저 업그레이드하고 모든 Message Queue 종속성을 업그레이드한 다음 Message Queue를 업그레이드하는 것입니다.

릴리스 2 Message Queue를 릴리스 5로 업그레이드할 때는 Solaris 시스템에서 필요하지 않은 데이터 마이그레이션 단계, 즉 브로커 인스턴스 데이터를 해당 릴리스 5 위치로 마이그레이션하는 작업이 필요합니다. 릴리스 5로 업그레이드하면서 릴리스 2 데이터를 유지하려는 경우 Message Queue에서 제공하는 마이그레이션 도구인 mqmigrate를 사용하여 마이그레이션을 수행할 수 있습니다.

릴리스 2에서 릴리스 5로 업그레이드하려면 [193페이지의 "릴리스 2 Message Queue 업그레이드\(Solaris\)"](#)에서 사용된 지침과 동일한 지침을 사용합니다. 단, 다음 절차에서 설명한 것처럼 mqupgrade 스크립트를 실행([195페이지의 단계 7](#))하기 전에 mqmigrate 스크립트를 실행([195페이지의 단계 6](#))해야 합니다.

1. 실행 중인 모든 Message Queue 클라이언트 응용 프로그램을 중지합니다.
2. 실행 중인 브로커를 중지합니다. 관리자 이름과 비밀번호를 묻는 메시지가 표시됩니다.

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

3. 각 브로커 인스턴스와 관련된 동적 데이터, Message Queue 플랫폼 파일 사용자 저장소 및 Message Queue 액세스 제어 파일을 유지하지 않으려면 다음 명령을 사용하여 해당 데이터를 제거합니다.

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

그렇지 않으면 동적 데이터 및 구성 정보는 유지되며 릴리스 5 Message Queue에서 사용됩니다.

4. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

5. 디렉토리를 Java ES 릴리스 5 배포의 Tools 디렉토리로 변경합니다.

```
cd Linux_x86/Product/message_queue/Tools
```

6. 다음 명령을 사용하여 브로커 인스턴스 데이터를 마이그레이션합니다.

```
./mqmigrate
```

mqmigrate 스크립트는 릴리스 2 브로커 인스턴스 구성 데이터를 릴리스 4의 적절한 위치로 옮깁니다.

7. mqupgrade 스크립트를 실행합니다.

- a. 스크립트를 시작합니다.

```
./mqupgrade
```

mqupgrade 스크립트는 설치된 Message Queue 구성 요소를 나열합니다.

- b. y(예)를 입력하여 Message Queue 구성 요소를 업그레이드합니다.

mqupgrade 스크립트는 설치된 현지화 파일을 검색하여 나열합니다.

Message Queue 구성 요소를 업그레이드하지 않으려면 n(아니오)을 입력합니다. mqupgrade 스크립트는 Message Queue 구성 요소를 업그레이드하지 않고 종료합니다.

- c. 메시지가 표시되면 y(예)를 입력하여 현지화 파일을 업그레이드합니다.

mqupgrade 스크립트는 다음 위치의 로그 파일에 출력을 전송합니다.

```
/var/sadm/install/logs/Message_Queue_upgrade_'date'.log
```

호환성 패키지 설치(Linux)

릴리스 2 클라이언트 응용 프로그램에 릴리스 5 파일이 설치된 위치에 종속되는 스크립트가 포함되거나 사용자에게 이러한 스크립트가 있는 경우 sun-mq-compat 패키지를 설치하여 릴리스 2와 릴리스 5 파일 위치를 심볼릭 링크로 연결하도록 합니다.

sun-mq-compat 패키지는 Java ES 릴리스 5 배포를 압축 해제한 다음 위치에 있습니다.

```
Linux_x86/Product/message_queue/Packages
```

다음 단계를 수행하여 sun-mq-compat 패키지를 설치합니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 패키지 디렉토리에서 다음 명령을 입력합니다.

```
rpm -ivh --nodeps sun-mq-compat-3.7-RelNo.i386.rpm
```

Message Queue 업그레이드 확인

업그레이드 절차를 마친 뒤 작업이 성공했는지를 확인하려면 -version 옵션을 지정하고 Message Queue 브로커를 시작하면 됩니다.

```
imqbrokerd -version
```

이 명령은 Java ES 버전 번호와 Message Queue별 버전 번호를 반환합니다.

사후 업그레이드 작업

HTTP 터널링 서블릿을 사용하여 HTTP 연결 서비스 지원을 제공하는 경우 Message Queue를 릴리스 2에서 릴리스 5로 업그레이드하면 서블릿도 업그레이드됩니다. 따라서 Message Queue를 릴리스 5로 업그레이드한 다음 서블릿을 다시 배포해야 합니다.

HTTP 지원에 대한 자세한 내용은 Message Queue 3.7 UR1 관리 설명서

(<http://docs.sun.com/doc/820-0845>)를 참조하십시오.

또한 업데이트 작업을 수행하려면 관리 콘솔(imqadmin) 또는 ObjectManager 명령줄 유틸리티(imqobjmgr)를 사용하여 릴리스 2 관리 객체를 릴리스 5 버전으로 마이그레이션해야 합니다.

업그레이드 롤백

Message Queue를 릴리스 2에서 릴리스 5로 업그레이드한 경우 현재는 롤백이 지원되지 않습니다. 일반적으로 절차는 릴리스 5를 릴리스 4로 롤백하는 절차와 비슷합니다 (185페이지의 "업그레이드 롤백" 참조). 그러나 릴리스 2에서 릴리스 5로 Message Queue를 업그레이드해도 Java ES 제품 레지스트리가 업데이트되지 않기 때문에 Java ES 설치 프로그램이 릴리스 2 Message Queue를 다시 설치할 수 없습니다.

이 문제를 해결하려면 Sun Services에 문의하십시오.

여러 인스턴스 업그레이드

여러 브로커가 상호 작용하여 확장 가능한 메시지 서비스를 제공하는 Message Queue 클러스터를 업그레이드하려면 각 Message Queue 인스턴스가 릴리스 2에서 릴리스 5로 업그레이드되는 동안 클러스터가 온라인 상태로 유지되는 롤링 업그레이드를 수행할 수 있습니다. 클러스터를 업그레이드할 때는 다음과 같은 두 가지 조건을 고려해야 합니다.

- 지속성 메시지는 업그레이드를 위해 브로커가 종료된 동안 저장되며 브로커가 다시 시작될 때까지 사용할 수 없습니다.
- 마스터 브로커는 마지막으로 업그레이드해야 합니다.

그렇지 않은 경우 절차는 간단합니다. 즉 브로커가 모두 업그레이드될 때까지 한 번에 하나씩 브로커를 종료하고 업그레이드한 다음 재시작합니다.

Application Server

이 장에서는 Application Server를 Java ES 5(릴리스 5): Sun Java System Application Server Enterprise Edition 8.2로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- [201페이지의 "Application Server 업그레이드 개요"](#)
- [207페이지의 "Java ES 릴리스 4에서 Application Server 업그레이드"](#)
- [213페이지의 "Java ES 릴리스 3에서 Application Server 업그레이드"](#)
- [214페이지의 "Java ES 릴리스 2에서 Application Server 업그레이드"](#)
- [224페이지의 "Solaris 10 다중 영역 환경에서 Solaris와 함께 번들로 제공되는 Application Server 업그레이드"](#)

주

이 장에 설명된 파일의 위치는 *AppServer8-base* 및 *AppServer8Config-base*(Application Server 8.x), *AppServer7-base* 및 *AppServer7Config-base*(Application Server 7.x) 디렉토리 경로에 대해 상대적으로 지정되어 있습니다. Application Server를 설치할 때 최소한 이 경로의 일부를 설치 디렉토리 또는 도메인 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다. 다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 11-1 Application Server 디렉토리 경로

경로 이름 변수	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

주 릴리스 5 Application Server (8.x) 의 기본 Application Server 도메인 이름은 domain1입니다. 즉, domain1은 이 장에서 사용된 *domainName* 변수의 기본값입니다.

또한 기본 DAS(Directory Administration Service) 인스턴스 이름은 server입니다. J2EE 응용 프로그램은 일반적으로 DAS 인스턴스가 아닌 다른 독립 실행형 인스턴스에 배포됩니다.

이 장에서 사용된 asadmin 명령에 대한 자세한 내용은 Sun Java System Application Server Enterprise Edition 8.2 Reference Manual (<http://docs.sun.com/doc/819-4736>)을 참조하십시오.

Application Server 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Application Server의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Application Server 정보](#)
- [Application Server 업그레이드 로드맵](#)
- [Application Server 데이터](#)
- [Application Server 업그레이드 전략](#)

Java ES 릴리스 5 Application Server 정보

Java ES 릴리스 5 Application Server는 릴리스 4와 관련하여 선택된 버그 수정만 포함된 부차적 릴리스입니다. 기능 측면에서 릴리스 5 Application Server는 릴리스 4와 동일합니다.

Application Server 업그레이드 로드맵

Application Server를 Java ES 릴리스 5로 업그레이드할 때 적용할 수 있는 두 가지 업그레이드 경로 집합이 있습니다.

- [표 11-2](#)는 Java ES Application Server를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 주며, 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.
- [표 11-3](#)은 Solaris에 번들로 제공된 Application Server를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. Application Server Platform Edition은 Solaris OS 소프트웨어와 함께 번들로 제공됩니다. Application Server의 번들 버전은 [표 11-3](#)에 나와 있는 대로 Java ES 설치 프로그램을 사용하여 릴리스 5 Enterprise Edition으로 업그레이드할 수 있습니다.

표 11-2 Java ES 5(릴리스 5) 업그레이드 경로: Application Server Enterprise Edition 8.2

Java ES 릴리스	Application Server 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Application Server Enterprise Edition 8.1 2005Q4	직접 업그레이드: Java ES 설치 프로그램을 사용하여 수행됩니다. ¹	없음
릴리스 3	Sun Java System Application Server Enterprise Edition 8.1 2005Q1	직접 업그레이드: Java ES 설치 프로그램을 사용하여 수행됩니다.	없음
릴리스 2	Sun Java System Application Server 7.0 Update 3(2004Q2) Platform Edition 및 Enterprise Edition	직접 업그레이드: Java ES 설치 프로그램을 사용하여 패키지를 업그레이드하고, postInstall 스크립트를 사용하여 재구성을 수행하고, asupgrade 유틸리티를 사용하여 도메인 정보를 마이그레이션합니다. ²	환경 변수, 도메인 및 기타 구성 데이터 J2EE 구성 요소 및 응용 프로그램은 새로운 Application Server 환경으로 마이그레이션하고 재배포해야 합니다.
릴리스 1	Sun ONE Application Server 7.0 Update 1(2003Q4) Platform Edition 및 Enterprise Edition	직접 업그레이드 인증 안 됨: 릴리스 2 Application Server에서 업그레이드할 때와 같은 방법을 사용합니다.	환경 변수, 도메인 및 기타 구성 데이터 J2EE 구성 요소 및 응용 프로그램은 새로운 Application Server 환경으로 마이그레이션하고 재배포해야 합니다.
이전 Java ES 릴리스		직접 업그레이드 불가능: 그러나 Java Enterprise System 2005Q1 Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)의 절차를 사용하여 릴리스 3으로 처음 업그레이드하는 것은 가능합니다. 그런 다음 릴리스 3에서 릴리스 5로 업그레이드합니다.	

1. 구성이나 도메인 정보를 저장하지 않고 Application Server를 릴리스 4에서 릴리스 5로 업그레이드하려는 경우 Java ES 릴리스 4 제거 프로그램을 사용하여 릴리스 4 Application Server를 제거한 다음 Java ES 릴리스 5 설치 프로그램을 사용하여 릴리스 5 Application Server를 새로 설치할 수 있습니다. 그러나 나중에 구성 옵션을 사용하여 릴리스 4 Application Server를 설치한 경우에는 릴리스 4 Application Server를 제거하기 전에 먼저 `$HOME/asadminprefs` 파일을 만들어야 합니다. 여기서 `$HOME`은 Application Server를 설치하고 실행하는 사용자의 홈 디렉토리입니다. 이 파일에는 다음 두 줄이 포함되어 있습니다.

```
AS_ADMIN_PASSWORD=password
AS_ADMIN_USER=admin
```

2. Solaris 플랫폼의 경우 릴리스 2 Application Server에서 업그레이드하면 컴퓨터에 Solaris에 번들로 제공되는 Application Server와 Java ES 릴리스 2 Application Server가 공존하게 되므로 주의해야 합니다. 표 11-3 및 이후 내용을 참조하십시오.

표 11-3 Solaris OS에 번들로 제공된 Application Server 버전 업그레이드 방법

Solaris OS 버전	Application Server 버전	일반적인 방법	재구성 필요
Solaris 10	Sun Java System Application Server Platform Edition 8.0.0_01	직접 업그레이드: Java ES 설치 프로그램을 사용하여 수행됩니다.	없음
Solaris 9	Sun Java System Application Server 7.0.0_03c	직접 업그레이드: Application Server를 사용하고 도메인을 만든 적이 있는 경우에는 도메인 정보를 마이그레이션해야 합니다. Java ES 설치 프로그램의 나중에 구성 옵션, postInstall 스크립트 및 asupgrade 유틸리티 등을 사용하여 릴리스 2 Application Server에서 업그레이드하는 방법에 대한 설명을 참조하십시오. Application Server를 사용한 적이 <i>없거나</i> 도메인을 만들지 않고 사용한 경우 릴리스 2 Application Server에서 업그레이드하는 방법에 대한 설명을 참조하십시오. 단, Java ES 설치 프로그램의 "지금 구성" 옵션을 사용하고 postInstall 스크립트 또는 asupgrade 유틸리티를 사용하지 않아야 합니다.	환경 변수, 도메인(만든 경우) 및 기타 구성 데이터 J2EE 구성 요소 및 응용 프로그램은 새로운 Application Server 환경으로 마이그레이션하고 재배포해야 합니다.

주 Solaris 플랫폼의 경우 릴리스 2 Application Server에서 업그레이드하면 컴퓨터에 번들 버전과 Java ES 버전의 Application Server가 공존하게 되므로 주의해야 합니다. 따라서 릴리스 2 버전에서 업그레이드를 진행하기 전에 번들 버전 및 해당 도메인을 제거해야 합니다. [216페이지](#)의 "Solaris OS만 해당: 운영 체제에 번들로 제공된 Application Server 패키지 수동 제거"를 참조하십시오.

Application Server 데이터

다음 표는 Application Server 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 11-4 Application Server 데이터 사용

데이터 유형	위치	사용
환경 변수	<i>AppServer8-base/config/asenv.conf</i>	전역 변수
구성 데이터	릴리스 3, 릴리스 4 및 릴리스 5: <i>domain.xml</i> 및 <i>server.policy</i> 파일의 경우 <i>AppServer8Config-base/domains/domainName/config</i> 릴리스 2: <i>server.xml</i> 및 <i>server.policy</i> 파일의 경우 <i>AppServer7Config-base/domains/domainName/instanceName/config</i>	Application Server 인스턴스 구성
배포 데이터	릴리스 3, 릴리스 4 및 릴리스 5: <i>AppServer8Config-base/domains/domainName/applications</i> 릴리스 2: <i>AppServer7Config-base/domains/domainName/instanceName/applications</i>	특정 J2EE 구성 요소 및 응용 프로그램용 J2EE 컨테이너 구성
액세스 로그 파일	릴리스 3, 릴리스 4 및 릴리스 5: <i>AppServer8Config-base/domains/domainName/logs/access/</i> 다음 두 파일 포함 <i>server_access_log</i> 및 <i>_asadmin_access_log</i> 릴리스 2: <i>AppServer7Config-base/domains/domainName/instanceName/logs/access</i>	액세스 로깅

Application Server 업그레이드 전략

Application Server 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Application Server 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Application Server에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Application Server는 릴리스 4 또는 릴리스 3의 인터페이스를 변경하지 않고 그대로 사용합니다. 그러나 릴리스 5와 릴리스 2 간에는 인터페이스가 많이 변경되었기 때문에 릴리스 5와 릴리스 2는 호환되지 않습니다.

그러나 릴리스 5 Application Server가 릴리스 4 Service Registry를 지원하지는 않습니다. Application Server를 릴리스 5로 업그레이드하면 Service Registry도 릴리스 5로 업그레이드해야 합니다.

Application Server 종속성

다른 Java ES 구성 요소에 대한 Application Server 종속성은 Application Server 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어 Application Server 인터페이스나 기능이 변경되면 Application Server가 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Application Server는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Application Server는 특정한 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속됩니다.
- **Message Queue.** Application Server는 J2EE Java Message Service 호환 비동기 메시징 지원을 제공하기 위해 Message Queue에 종속됩니다.
- **고가용성 세션 저장소.** Application Server는 인스턴스 간의 페일오버 지원에 필요한 세션 상태 정보를 유지 관리하기 위해 고가용성 세션 저장소(HADB)에 종속됩니다.

주	HADB를 사용하지 않는 이전 Application Server가 설치되어 있는 경우에는 HADB를 설치해야 Application Server를 릴리스 5로 업그레이드할 수 있습니다. 예를 들어 Solaris OS에 번들로 제공된 Application Server 버전을 업그레이드하려면 릴리스 5 HADB를 설치해야 합니다. 이러한 경우에는 Java ES 설치 프로그램이 자동으로 HADB 설치를 수행합니다.

- **Java DB.** Application Server는 기본 개발자 데이터베이스로 Java DB를 사용하여 샘플 응용 프로그램 데이터와 Enterprise Java Beans 타이머에 필요한 데이터를 저장합니다.
- **웹 컨테이너(선택적 종속성).** Application Server는 선택 사항인 로드 균형 조정 플러그 인의 사용을 위해 웹 컨테이너를 사용합니다. Java ES Web Server 또는 타사 웹 컨테이너(예: Apache Web Server, Microsoft IIS)를 통해서도 이를 지원할 수 있습니다.

이중 업그레이드

Application Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지](#)의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"에서 설명한 대로 다음 두 가지 방법 중 하나로 수행할 수 있습니다.

- 운영 체제 새로 설치
- 기존 운영 체제 업그레이드

운영 체제 새로 설치

1. 기존 Application Server 데이터를 백업합니다.

필수 데이터의 위치는 [204페이지](#)의 "Application Server 데이터"를 참조하십시오.

2. 새 운영 체제를 설치합니다.

운영 체제는 같은 컴퓨터에만 설치할 수 있으며 기존 파일 시스템은 제거됩니다.

3. [단계 1](#)에서 백업한 Application Server 데이터를 복원합니다.

4. 릴리스 5 Application Server를 설치합니다.

[단계 1](#)에서 백업한 Application Server 데이터의 버전에 따라 이 장의 해당 업그레이드 절에 설명된 절차를 사용하십시오.

기존 운영 체제 업그레이드

1. 기존 Application Server 데이터를 백업합니다.

필수 데이터의 위치는 [204페이지](#)의 "Application Server 데이터"를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Application Server로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 해당 절을 참조하십시오.

Java ES 릴리스 4에서 Application Server 업그레이드

이 절에는 Application Server를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Application Server 업그레이드](#)

소개

Java ES 릴리스 4 Application Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** Java ES 설치 프로그램을 사용하여 업그레이드를 수행합니다. 릴리스 4 Application Server를 릴리스 5로 업그레이드하는 데는 Application Server의 재구성 또는 J2EE 구성 요소 마이그레이션은 필요하지 않습니다.
- **업그레이드 종속성.** Application Server는 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Java ES 설치 프로그램이 Application Server 업그레이드를 수행할 때 자동으로 릴리스 5로 업그레이드됩니다. Application Server는 NSS 공유 구성 요소에 대해서만 엄격한 업그레이드 종속성이 있습니다.

또한 [205페이지의 "Application Server 종속성"](#)에서 설명한 대로 릴리스 5 Application Server는 Message Queue, HADB 및 Java DB에 종속됩니다. 이러한 종속성은 엄격한 업그레이드 종속성이며 모두 릴리스 5로 업그레이드해야 합니다.

또한 Application Server는 Java ES Web Server 또는 타사 웹 컨테이너에 선택적으로 종속됩니다. 그러나 이는 엄격하지 않은 업그레이드 종속성입니다. 따라서 Application Server를 릴리스 5로 업그레이드하는 경우 웹 컨테이너는 선택적으로 업그레이드할 수 있습니다.

- **역방향 호환성.** 릴리스 5 Application Server는 릴리스 4 버전과 호환됩니다.
- **업그레이드 롤백.** 릴리스 5로 업그레이드한 경우 릴리스 4로 롤백할 수 없습니다.
- **플랫폼 문제.** Application Server 업그레이드를 위한 방법은 Solaris 및 Linux 운영 체제에서 동일합니다.

릴리스 4 Application Server 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Application Server를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Application Server](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Application Server 소프트웨어를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Application Server 종속성 업그레이드](#)
- [Application Server 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

다음 명령을 입력하여 Application Server의 현재 버전을 확인할 수 있습니다.

```
AppServer8-base/bin/asadmin version --verbose
```

표 11-5 Application Server 버전 확인 출력

Java ES 릴리스	Application Server 버전 번호
릴리스 2	Sun ONE Application Server 7.0.0_03c
릴리스 3	Sun Java Enterprise System Application Server Enterprise Edition 8.1
릴리스 4	Sun Java Enterprise System Application Server Enterprise Edition 8.1_02
릴리스 5	Sun Java Enterprise System Application Server Enterprise Edition 8.2

Application Server 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Application Server는 NSS 공유 구성 요소 및 Message Queue, HADB, Java DB 제품 구성 요소에 대한 엄격한 업그레이드 종속성이 있습니다.

Application Server 종속성을 업그레이드할 때는 Application Server를 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다. 그러나 공유 구성 요소를 비롯하여 Message Queue, HADB 및 Java DB에 대한 업그레이드는 Application Server 업그레이드 시 Java ES 설치 프로그램에서 자동으로 수행합니다.

1. **공유 구성 요소.** 63페이지의 "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Application Server에 필요한 모든 공유 구성 요소는 Application Server를 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.
2. **Message Queue.** 175페이지의 10장, "Message Queue"에서 Message Queue를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **고가용성 세션 저장소(HADB).** 163페이지의 9장, "고가용성 세션 저장소"에서 HADB를 업그레이드하는 방법에 대한 지침을 제공합니다.
4. **Java DB.** 153페이지의 8장, "Java DB"에서 Java DB를 업그레이드하는 방법에 대한 지침을 제공합니다.
5. **웹 컨테이너 소프트웨어(엄격하지 않은 업그레이드 종속성).** 129페이지의 7장, "Web Server"에서 Web Server를 업그레이드하는 방법에 대한 지침을 제공합니다.

Application Server 데이터 백업

Application Server를 릴리스 4에서 릴리스 5로 업그레이드하는 경우 구성 데이터를 수정하지 않습니다. 따라서 현재 데이터를 백업하지 않아도 됩니다.

필수 구성 정보 및 비밀번호 얻기

현재 설치된 버전에 대한 Application Server 관리자 아이디와 비밀번호를 알아야 합니다.

릴리스 4 업그레이드 Application Server

이 절에서는 Application Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

Application Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- Application Server 인스턴스를 업그레이드하려면 해당 인스턴스에서 실행 중인 J2EE 구성 요소를 종료해야 합니다. 그러나 고가용성 또는 확장성을 위해 로드 균형 조정기가 제공되는 경우 이 요구 사항은 완화됩니다.
- 설치된 이미지가 업그레이드되는 동안 단일 컴퓨터에서 실행 중인 모든 Application Server 인스턴스(동일한 Application Server 설치 이미지에 해당하는 모든 인스턴스)를 종료해야 합니다.
- 다중 노드 배포의 경우 각 노드 또는 Application Server 인스턴스를 호스팅하는 컴퓨터에서 업그레이드 절차를 수행합니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 로컬에 상주하는 Application Server 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 업그레이드할 Application Server 인스턴스에서 실행 중인 모든 J2EE 구성 요소를 종료하십시오.
3. 업그레이드할 컴퓨터에 있는 모든 Application Server 인스턴스를 종료합니다.
 - a. 실행 중인 모든 노드 에이전트를 중지합니다.

```
AppServer8-base/bin/asadmin stop-node-agent --user admin_ID  
nodeagentName
```

여기서 *nodeagentName*은 *hostName_domainName* 형식이지만 기본값은 간단히 *hostName*입니다.

- b. DAS(Domain Administration Server)를 중지합니다.

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID  
domainName
```

- c. PointBase 데이터베이스 서버를 중지합니다(사용 중인 경우).

```
AppServer8Config-base/appserver/pointbase/tools/stopserver.sh
```

4. Java ES 설치 프로그램을 시작합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

5. 구성 요소 선택 페이지에서 Application Server를 선택합니다.

엄격한 업그레이드 종속성 때문에 Message Queue, HADB 및 Java DB는 업그레이드되도록 자동으로 선택됩니다.

6. "지금 구성"을 선택합니다.

7. 필요한 구성 값을 지정합니다.

여러 개의 구성 패널이 표시됩니다.

8. 업그레이드 선택 항목을 확인합니다.

Application Server 패키지(필요한 경우 Message Queue, HADB 및 Java DB용 패키지 포함)가 업그레이드되고 업그레이드 요약이 표시됩니다.

9. Java ES 설치 프로그램을 종료합니다.

10. 업그레이드한 DAS(Domain Administration Server)를 다시 시작합니다.

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
domainName
```

11. 업그레이드된 Application Server 인스턴스를 다시 시작합니다.

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
nodeagentName
```

여기서 *nodeagentName*은 *hostName_domainName* 형식이지만 기본값은 간단히 *hostName*입니다.

업그레이드 확인

다음 명령을 사용하여 성공적으로 업그레이드되었는지 확인할 수 있습니다.

```
AppServer8-base/bin/asadmin version --verbose
```

출력 값에 대해서는 [208페이지 표 11-5](#)를 참조하십시오.

사후 업그레이드 작업

새로운 릴리스 5 Java ES 제품 구성 요소인 Java DB 대신 기본 제공되는 릴리스 4 PointBase 데이터베이스를 계속 사용하려면 업그레이드된

AppServer8-base/config/asenv.conf 파일을 수동으로 편집해야 합니다.

Application Server를 릴리스 4에서 업그레이드한 후 *asenv.conf*의 PointBase 설정은 다음과 같습니다.

```
AS_POINTBASE="%POINTBASE_HOME%"  
AS_POINTBASE_SAMPLESDB="%POINTBASE_SAMPLESDB%"
```

이러한 설정을 다음 값으로 변경합니다.

```
AS_POINTBASE="AppServer8-base/pointbase"  
AS_POINTBASE_SAMPLESDB="AppServerConfig8-base/var/appserver/pointbase"
```

업그레이드 롤백

릴리스 5 업그레이드 롤백은 지원되지 않습니다.

Java ES 릴리스 3에서 Application Server 업그레이드

Java ES 2005Q1(릴리스 3) Application Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Application Server를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 3 Application Server를 릴리스 5로 업그레이드하려면 [207 페이지의 "Java ES 릴리스 4에서 Application Server 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

Java ES 릴리스 2에서 Application Server 업그레이드

이 절에는 Application Server를 Java ES 릴리스 2에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 2 Application Server 업그레이드](#)

주 Linux 플랫폼에서 릴리스 2 Application Server를 업그레이드하는 경우 Application Server *와* 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Application Server가 지원되지 않음). 자세한 내용은 [206페이지의 "이중 업그레이드"](#)를 참조하십시오.

소개

Java ES 릴리스 2 Application Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** Java ES 설치 프로그램을 사용하고 나중에 구성 옵션을 선택하여 릴리스 5 Application Server를 설치하는 방법으로 업그레이드를 수행합니다. 재구성 작업은 `asupgrade` 유틸리티를 사용하여 나중에 수행합니다. Application Server를 업그레이드한 다음 릴리스 2 J2EE 구성 요소 및 응용 프로그램을 릴리스 5로 마이그레이션해야 합니다.
- **업그레이드 종속성.** 릴리스 2에서는 한 컴퓨터에서 임의의 Java ES 구성 요소를 업그레이드하는 경우 해당 컴퓨터가 호스팅하는 다른 모든 Java ES 구성 요소를 업그레이드해야 합니다. Java ES 구성 요소를 릴리스 2에서 릴리스 5로 선택적으로 업그레이드할 수는 없습니다.

특히 Application Server에 필요한 모든 Java ES 공유 구성 요소를 업그레이드해야 합니다. 또한 Message Queue가 같은 컴퓨터에 있는 경우에도 업그레이드해야 합니다.고가용성 세션 저장소(HADB) 및 Java DB를 설치해야 하며 로드 균형 조정에는 Web Server를 사용 중인 경우에도 업그레이드해야 합니다.

- **역방향 호환성.** 릴리스 5 Application Server는 릴리스 2 버전과 호환되지 않습니다. J2EE 구성 요소 및 응용 프로그램은 릴리스 5 Application Server 환경에서 마이그레이션하여 실행해야 합니다.

- **업그레이드 롤백.** 릴리스 5 업그레이드를 릴리스 2로 롤백하려면 릴리스 2를 다시 설치하면 됩니다(업그레이드 프로세스를 수행해도 릴리스 2 구성 데이터는 제거되지 않습니다).
- **플랫폼 문제.** Solaris OS에서 Application Server를 업그레이드하는 방법은 Linux OS에 곧바로 적용할 수 없습니다. Linux2.1에서는 Java ES 릴리스 2만 지원되며 Java ES 릴리스 5는 지원되지 않습니다. 따라서 운영 체제와 Application Server를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다([43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어" 참조](#)).
- **데이터 위치.** [204페이지 표 11-4](#)에 정리되어 있는 것처럼 릴리스 5에서는 구성 데이터, 배포 데이터, 액세스 로그 파일을 새로운 위치에 저장합니다.

릴리스 2 Application Server 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Application Server를 Java ES 릴리스 2에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 2 업그레이드 Application Server](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Application Server를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [Solaris OS만 해당: 운영 체제에 번들로 제공된 Application Server 패키지 수동 제거](#)
- [현재 버전 정보 확인](#)
- [Application Server 종속성 업그레이드](#)
- [Application Server 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

Solaris OS만 해당: 운영 체제에 번들로 제공된 Application Server 패키지 수동 제거

릴리스 2 Application Server를 설치해도 Solaris OS에 번들로 제공된 Application Server 버전이 제거되지는 않습니다. 릴리스 2 Application Server를 릴리스 5로 올바르게 업그레이드하려면 다음 단계를 수행해야 합니다.

1. /usr/appserver 디렉토리에 있는 해당 Application Server 패키지를 수동으로 제거하여 번들 버전을 제거합니다.

```
pkgrm SUNWascmnse SUNWaslb SUNWasut ...
```

여기에서 다음 명령을 사용하여 전체 패키지 집합을 얻을 수 있습니다.

```
pkginfo -i|grep -i "application server"
```

결과에는 다음과 같은 패키지가 포함될 수 있습니다.

```
SUNWasacee, SUNWascmnse, SUNWasu, SUNWasuee, SUNWasac, SUNWascmn,
SUNWasdb, SUNWasdem, SUNWasdemdb, SUNWasr, SUNWasut, SUNWasman,
SUNWasjdoc
```

또한 다음 현지화 패키지도 포함될 수 있습니다.

```
SUNWLocaleasacee, SUNWLocaleascmnse, SUNWLocaleasu, SUNWLocaleasuee
```

2. 번들로 제공된 버전을 사용하지 않는 경우 해당 도메인 정보를 삭제합니다.

```
rm /usr/appserver/domains
```

현재 버전 정보 확인

다음 명령을 입력하여 Application Server의 현재 버전을 확인할 수 있습니다.

```
AppServer7-base/bin/asadmin version --verbose
```

버전 출력에 대해서는 [208페이지 표 11-5](#)를 참조하십시오.

Application Server 종속성 업그레이드

Application Server 종속성을 업그레이드하려면 Application Server가 종속된 모든 로컬 상주 릴리스 2 구성 요소, 특히 Message Queue와 공유 구성 요소를 릴리스 5로 업그레이드해야 합니다. 그러나 공유 구성 요소는 업그레이드 절차의 일부로 Java ES 설치 프로그램에 의해 자동으로 업그레이드됩니다([218페이지의 "업그레이드 절차"](#) 참조).

또한 Application Server를 업그레이드하려면 릴리스 5 Application Server가 종속된 HADB 및 Java DB도 설치해야 합니다. 그러나 이러한 제품 구성 요소는 업그레이드 절차의 일부로 Java ES 설치 프로그램에서 자동으로 선택하여 설치합니다([218페이지의 "업그레이드 절차"](#) 참조).

일반적으로 Application Server 종속성을 업그레이드하는 경우 Application Server를 업그레이드하기 전에 다음 순서 대로 업그레이드해야 하며, 이미 업그레이드한 항목은 건너뛸 수 있습니다. 그러나 공유 구성 요소 업그레이드를 비롯하여 HADB 및 Java DB 설치 Application Server 업그레이드 시 Java ES 설치 프로그램에서 자동으로 수행합니다.

1. **공유 구성 요소.** 63페이지의 "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Application Server에 필요한 모든 공유 구성 요소는 Application Server를 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.

주	공유 구성 요소를 릴리스 5로 업그레이드하는 절차에는 Application Server에서 지원하지 않는 J2SE를 JDK 버전 1.5로 업그레이드하는 절차가 포함됩니다. 따라서 공유 구성 요소를 업그레이드하기 전에 릴리스 2 Application Server를 종료해야 합니다. 공유 구성 요소는 Java ES 설치 프로그램을 사용하여 Java ES 제공 구성 요소(Directory Server, Message Queue, HADB, Java DB 및 기타 다른 구성 요소)를 설치 또는 업그레이드할 때마다 업그레이드됩니다.

2. **Message Queue.** 188페이지의 "[Java ES 릴리스 2로부터 Message Queue 업그레이드](#)"를 참조하십시오. Message Queue는 Java ES 설치 프로그램을 사용하여 릴리스 2에서 업그레이드할 수 없습니다.
3. **고가용성 세션 저장소(HADB).** Application Server 업그레이드 시 Java ES 설치 프로그램이 HADB를 새로 설치합니다.
4. **Java DB.** Application Server 업그레이드 시 Java ES 설치 프로그램이 Java DB를 새로 설치합니다.
5. **웹 컨테이너 소프트웨어(엄격하지 않은 업그레이드 종속성).** 152페이지의 "[Java ES 릴리스 2에서 Web Server 업그레이드](#)"를 참조하십시오.

Application Server 데이터 백업

Application Server를 릴리스 2에서 릴리스 5로 업그레이드하더라도 구성 데이터를 덮어쓰지는 않습니다. 그러나 안전하게 작업을 수행하려면 릴리스 5로 업그레이드하기 전에 전체 domains 디렉토리를 백업해야 합니다.

AppServer7Config-base/domains

필수 구성 정보 및 비밀번호 얻기

현재 설치 버전에 대한 다음 정보를 알아야 합니다.

- Application Server 관리자 아이디와 비밀번호
- 릴리스 2 Application Server 기본 디렉토리

릴리스 2 업그레이드 Application Server

이 절에서는 Application Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

Application Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- Application Server 인스턴스를 업그레이드하려면 해당 인스턴스에서 실행 중인 J2EE 구성 요소를 종료해야 합니다. 그러나고가용성 또는 확장성을 제공하기 위해 로드 균형 조정을 사용하는 경우 이 요구 사항은 완화될 수 있습니다.
- 설치된 이미지가 업그레이드되는 동안 단일 컴퓨터에서 실행 중인 모든 Application Server 인스턴스(모두 설치된 동일한 Application Server 이미지에 해당)를 종료해야 합니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 로컬에 상주하는 모든 Application Server 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 릴리스 2 Application Server가 아직 실행 중인 동안 Java ES 공유 구성 요소를 릴리스 5로 업그레이드한 경우, 특히 J2SE를 JDK 1.5로 업그레이드한 경우에는 JDK 1.4를 직접 가리키도록 asenv.conf 파일을 편집합니다.

```
AS_JAVA=/usr/java
```

asenv.conf 파일이 JDK 1.5를 참조하는 경우에는 Application Server를 종료할 수 없습니다(아래 [단계 3](#) 참조).

3. 모든 Application Server 및 관련 프로세스를 중지합니다.

```
AppServer7-base/bin/asadmin stop-appserv domainName
```

4. 릴리스 5 Application Server를 설치합니다.

다음 단계를 수행합니다.

- a. 릴리스 2 Application Server를 호스팅하는 컴퓨터에서 Java ES 설치 프로그램을 실행합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령 줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우 상태가 "업그레이드 가능"으로 표시됩니다. Application Server는 릴리스 2에서 업그레이드 가능한 것으로 표시되지 *않습니다*.

- b. 구성 요소 선택 페이지에서 Application Server를 선택합니다.

노드 에이전트 구성 요소를 포함하여 최소한 처음 세 가지 하위 구성 요소를 설치하도록 선택합니다.

- c. "다음"을 누릅니다.

Message Queue를 릴리스 5로 업그레이드하지 않은 경우 Message Queue를 업그레이드할 것인지 묻는 오류 메시지 창이 표시됩니다. 이 경우 "확인"을 클릭하여 Message Queue를 업그레이드합니다. 절차에 대해서는 [191페이지의 "릴리스 2 Message Queue 업그레이드"](#)에 나와 있습니다.

- d. 릴리스 2 Application Server가 설치된 디렉토리나 다른 설치 디렉토리를 지정합니다.
- e. "나중에 구성" 옵션을 선택합니다.

주	Solaris 9 OS에 번들로 제공된 Application Server가 있고 이 버전을 사용하지 않는 경우(도메인 정보를 만들지 않은 경우) "지금 구성"을 선택한 다음 단계 5 를 생략하고 다음 단계로 진행할 수 있습니다.

- f. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.
- g. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.

5. 다음 사후 설치 절차를 수행합니다.

- a. Java ES 릴리스 5 배포의 Application Server 도구 디렉토리에서 postInstall readme 파일을 찾습니다.

Java ES Release 5 distribution/os_arch/Product/application_svr/Tools

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다.

- b. ReadMe 파일을 참조하여 postInstall 스크립트를 실행합니다.

```
cd Java ES Release 5 distribution/os_arch/Product/application_svr/Tools
./postInstall AppServer8Install-base AppServer8Config-base
```

이 스크립트는 설치 중에 설치된 템플릿에서 *AppServer8-base/bin/** 셸 스크립트와 config/asenv.conf 파일을 구성하고 만듭니다. 일반적으로 Java ES 설치 프로그램에서 bin/* 셸 스크립트를 만들지만 "나중에 구성"을 선택한 경우 설명한 대로 셸 스크립트를 만들어야 합니다.

주 Application Server를 릴리스 5로 업그레이드하면 관리되지 않는 포트 8080(기본값) 및 8181(보안)이 DAS(Domain Administration Server)에 할당됩니다. 이러한 포트 할당은 릴리스 2 Application Server 인스턴스에 할당된 포트와 충돌을 일으킬 수 있습니다. 이런 경우 충돌을 피할 수 있도록 DAS 포트를 변경해야 합니다. 여기에 대한 지침은 Sun Java System Application Server Enterprise 8.2 Edition 관리 설명서 (<http://docs.sun.com/doc/819-4733>)에서 http-listener 속성을 수정하는 방법에 대한 절을 참조하십시오.

- c. 필요한 경우 *AppServer8-base/config/asenv.conf* 파일에서 환경 설정을 수정합니다.

파일을 수동으로 편집해야 합니다.

주 로드 균형 조정을 위해 Application Server를 구성하려면 Sun Java System Application Server Enterprise 8.2 Edition 고가용성 관리 설명서(<http://docs.sun.com/doc/820-0853>)에서 "Application Server 고가용성 기능" 장의 "HTTP 로드 균형 조정을 위한 웹 서버 구성" 절을 참조하십시오.

6. asupgrade 유틸리티를 실행합니다.

asupgrade 유틸리티는 릴리스 5 노드 에이전트를 생성하고 여기에서 릴리스 2 Application Server 인스턴스를 마이그레이션합니다.

이 유틸리티는 Application Server 디렉토리 아래에 있습니다. 예를 들면 다음과 같습니다.

- 업그레이드 마법사 모드: *AppServer8-base/bin/asupgrade*
- 업그레이드 콘솔 모드: *AppServer8-base/bin/asupgrade -c*

업그레이드 마법사 또는 업그레이드 콘솔에서 업그레이드 단계를 안내할 것입니다.

a. 도메인 정보 마이그레이션을 위한 대상 및 원본 디렉토리를 확인합니다.

- 릴리스 2: *AppServer7-base*
- 릴리스 5: *AppServerConfig8-base/domains*

b. 릴리스 2 Application Server의 관리자 아이디, 관리자 비밀번호 및 마스터 비밀번호를 지정합니다.

마스터 비밀번호를 물어보면 *changeit*를 지정하고, *changeit*가 아닌 다른 마스터 비밀번호가 있는 키 저장소를 사용하는 경우에는 해당 비밀번호를 *changeit*로 변경해야 합니다.

```
jdk-home/bin/keytool -storepasswd -new changeit -keystore keystore
-storepass oldpasswd
```

Application Server asupgrade 유틸리티에 대한 자세한 내용은 Sun Java System Application Server Enterprise Edition 8.2 Reference Manual

(<http://docs.sun.com/doc/819-4736>)의 사용자 명령 절과 Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide (<http://docs.sun.com/doc/819-4737/6n6sao3ju?a=view>)를 참조하십시오.

7. 218페이지의 단계 2에서 asenv.conf 파일을 리디렉션한 경우에는 JDK 1.5를 가리키도록 복원합니다.

```
AS_JAVA=/usr/jdk/entsys-j2se
```

asenv.conf 파일이 JDK 1.4를 참조하는 경우 Application Server를 시작(아래 단계 8 및 단계 9 참조)할 수 없습니다.

8. DAS(Domain Administration Server)를 시작합니다.

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
domainName
```

9. 업그레이드된 Application Server 인스턴스를 다시 시작합니다.

다음과 같이 업그레이드된 Application Server 인스턴스가 마이그레이션된 노드 에이전트를 시작하여 이 작업을 수행합니다.

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID  
nodeagentName
```

여기서 *nodeagentName*은 *hostName_domainName* 형식이지만 기본값은 간단히 *hostName*입니다.

업그레이드 확인

관리 콘솔을 시작하고 서버가 시작되었는지 확인합니다. 실행되지 않는 서버가 있는 경우 다음 로그 파일에서 포트 충돌로 인한 실패가 발생하지 않았는지 확인합니다.

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/logs/server.log
```

포트 충돌로 인해 실패가 발생한 경우 관리 콘솔에서 포트 번호를 수정하여 포트 충돌을 제거한 다음 노드 에이전트를 중지하고 다시 시작합니다.

다음 명령을 사용하여 Application Server가 릴리스 5로 업그레이드되었는지 확인할 수 있습니다.

```
AppServer8-base/bin/asadmin version --verbose
```

출력 값에 대해서는 [208페이지 표 11-5](#)를 참조하십시오.

사후 업그레이드 작업

[218페이지의 "업그레이드 절차"](#)에서 설명된 단계 이외에도 수행해야 하는 사후 업그레이드 작업이 두 가지 더 있습니다.

- [JSS에 대한 정확한 참조](#)
- [릴리스 2 J2EE 구성 요소 마이그레이션](#)

JSS에 대한 정확한 참조

Application Server가 정확한 버전의 JSS 공유 구성 요소를 참조하도록 하려면 다음 위치의 Application Server domain.xml 파일에서 jss3.jar을 jss4.jar로 바꿔야 합니다.

```
AppServer8Config-base/domains/domainName/config/domain.xml
```

릴리스 2 J2EE 구성 요소 마이그레이션

릴리스 5 Application Server 환경에서 실행할 수 있도록 릴리스 2 J2EE 구성 요소 및 응용 프로그램을 마이그레이션하고 적절한 Application Server 인스턴스에 재배포해야 합니다. J2EE 구성 요소 및 응용 프로그램 마이그레이션에 대한 자세한 내용은 Application Server Enterprise Edition 8.2 Upgrade and Migration Guide (<http://docs.sun.com/doc/819-4737>)의 4장을 참조하십시오.

업그레이드 롤백

Application Server의 릴리스 5 업그레이드 롤백 절차는 Application Server의 릴리스 2 버전으로 되돌리는 것이며 릴리스 5로 업그레이드해도 릴리스 2 버전이 제거되는 것은 아닙니다.

Solaris 10 다중 영역 환경에서 Solaris와 함께 번들로 제공되는 Application Server 업그레이드

Solaris 10 OS에는 Application Server(Application Server에서 사용하는 Message Queue 포함)가 번들로 제공됩니다. 전역 영역에서 제거하지 않으면 번들로 제공된 Application Server는 이러한 영역이 생성될 때 비전역 영역으로 전파됩니다. 모든 영역에 번들 버전의 Application Server가 있으면 이후에 Application Server를 릴리스 5로 업그레이드하는 작업에 영향을 줄 수 있습니다.

- 전역 영역에서 Application Server를 업그레이드하면 전역 영역의 번들 버전이 제거되며 자동으로 모든 비전역 영역에서 번들 버전이 제거됩니다.
- 릴리스 5 Application Server 패키지는 전파되지 않으므로 전역 영역에서 Application Server를 릴리스 5로 업그레이드해도 비전역 영역의 Application Server는 업그레이드되지 않습니다.

다음 예에서는 Solaris 10 다중 영역 환경에서 Application Server 업그레이드와 관련된 몇 가지 사소한 사항을 설명하기 위해 제공됩니다. Java ES 및 Solaris 10 영역에 대한 자세한 내용은 [58페이지의 "Java ES 5 업그레이드 및 Solaris 10 영역"](#)을 참조하십시오.

이 예의 목표는 Solaris 10 스파스 루트 영역에서 Solaris와 함께 번들로 제공된 Application Server(버전 8.0.0_01)를 릴리스 5로 업그레이드하는 것입니다.

스파스 루트 영역에는 전역 영역에서 직접 마운트된 읽기 전용 디렉토리에 Solaris의 번들 Application Server가 설치되므로 간단히 Application Server를 업그레이드할 수 없습니다. 따라서 스파스 루트 영역에서 Application Server를 릴리스 5로 업그레이드하려면 먼저 전역 영역에서 번들 버전을 제거해야 합니다.

또한 Message Queue는 전역 영역에 설치되는데, 이는 공유 구성 요소(제품 구성 요소 아님)만 전역 영역에 설치되던 기존의 방식과 달라진 점입니다. 이는 스파스 루트 영역에서는 읽기 전용 디렉토리 때문에 Message Queue를 설치 또는 업그레이드할 수 없기 때문입니다.

Solaris 10 스파스 루트 영역에서 Solaris의 번들 Application Server(버전 8.0.0_01)를 릴리스 5로 업그레이드하는 절차는 다음과 같습니다.

1. 시스템의 초기 상태를 확인합니다.

이 예에서는 전역 관리자가 스파스 루트 영역에서 Solaris 10 버전을 구성, 설치 및 부팅했다고 가정합니다.

스파스 루트 영역에는 전역 영역에 이미 설치되어 있는 모든 Java ES 구성 요소, 즉 Solaris 10에 번들로 제공된 Message Queue 및 Application Server 버전이 포함되어 있습니다.

또한 이 예에서는 사용자가 스파스 루트 영역에서 번들로 제공된 Application Server를 사용하여 보존해야 하는 관리 도메인 정보를 만들었다고 가정합니다.

2. 번들로 제공된 Application Server 버전을 전역 영역에서 업그레이드합니다.

이 작업을 수행하면 번들로 제공된 Application Server 패키지가 제거되고 릴리스 5 패키지로 대체됩니다. 번들 패키지 제거는 스파스 루트 영역으로 전파되어, 결과적으로 스파스 루트 영역에서 Application Server 패키지를 제거할 수 있지만 릴리스 5 패키지가 비전역 영역으로 전파되지는 않습니다.

- a. 전역 영역에서 Java ES 설치 프로그램을 실행합니다.
- b. 구성 요소 선택 페이지에서 Application Server를 선택합니다.

Message Queue, HADB 및 Java DB는 자동으로 선택되며 Application Server 및 Message Queue는 업그레이드 가능한 것으로 표시됩니다.

- c. 업그레이드를 완료합니다.

릴리스 5 Message Queue는 스파스 루트 영역으로 전파되지만 Application Server, HADB 및 Java DB는 전파되지 않습니다. 또한 모든 공유 구성 요소가 릴리스 5와 동기화되며 스파스 루트 영역으로 전파됩니다.

3. 스파스 루트 영역에 Application Server를 설치합니다.

- a. 스파스 루트 영역에서 Java ES 설치 프로그램을 실행합니다.
- b. 구성 요소 선택 페이지에서 Application Server를 선택합니다.

Message Queue가 자동으로 선택된 경우 선택을 취소하고, HADB 및 Java DB는 자동으로 선택되어 있지 않은 경우 선택합니다.

- c. Application Server 설치를 완료합니다.

설치 과정에서 도메인 정보를 덮어 쓰지 않도록 "나중에 구성"을 선택합니다.

만든 도메인이 없는 경우 "지금 구성"을 선택할 수 있습니다.

Solaris 10 다중 영역 환경에서 Solaris와 함께 번들로 제공되는 Application Server 업그레이드

Service Registry

이 장에서는 Service Registry를 Java ES 5(릴리스 5): Service Registry 3.1로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- [228페이지의 "Service Registry 업그레이드 개요"](#)
- [231페이지의 "Java ES 릴리스 4에서 Service Registry 업그레이드"](#)

주	이 장에 설명된 파일 위치는 <i>ServiceRegistryR4-base</i>와 <i>RegistryDomainR4-base</i>(Java ES 릴리스 4 Service Registry) 및 <i>ServiceRegistryR5-base</i>와 <i>RegistryDomainR5-base</i>(Java ES 릴리스 5 Service Registry)라는 디렉토리 경로에 상대적으로 지정됩니다. Service Registry를 처음 설치할 때 최소한 이 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다. 다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 12-1 Service Registry 디렉토리 경로

경로 이름 변수	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/SUNWsrvc-registry	/opt/sun/srvc-registry
<i>RegistryDomainR5-base</i>	/var/opt/SUNWsrvc-registry	/var/opt/sun/srvc-registry

Service Registry 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Service Registry의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 정보](#)
- [Java ES 릴리스 5 업그레이드 로드맵](#)
- [Service Registry 데이터](#)
- [Service Registry 업그레이드 전략](#)

Java ES 릴리스 5 정보

Java ES 릴리스 5 Service Registry는 릴리스 4 Service Registry와 비교할 때 부차적 릴리스입니다. 여기에는 몇 가지 향상된 기능, 업데이트된 인터페이스 및 몇 가지 버그 수정이 포함되어 있습니다.

Java ES 릴리스 5 업그레이드 로드맵

[표 12-2](#)는 Service Registry를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 12-2 Java ES 5(릴리스 5) 업그레이드 경로: Service Registry 3.1

Java ES 릴리스	Service Registry 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Service Registry 3.0 2005Q4	직접 업그레이드: 릴리스 5를 새로 설치하고 레지스트리 데이터를 릴리스 5로 전송하여 릴리스 4를 대체합니다.	없음

Service Registry 데이터

다음 표는 Service Registry 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 12-3 Service Registry 데이터 사용

데이터 유형	위치	사용
설치 매개 변수	<i>ServiceRegistryR4-base/install/install.properties</i>	Service Registry 구성
신뢰할 수 있는 인증서	<i>ServiceRegistryR4-base/install/cacerts</i>	Application Server 설치에 포함되지 않은 Service Registry에서 신뢰되는 인증서
구성 데이터	<i>RegistryDomainR4-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes/*.properties</i>	Service Registry 인스턴스 구성
레지스트리/저장소 데이터	<i>RegistryDomainR4-base/3.0/data</i>	데이터베이스 및 사용자 인증서
웹 인터페이스 구성	<i>RegistryDomainR4-base/3.0/jaxr-ebxml</i>	웹 인터페이스 구성

Service Registry 업그레이드 전략

Service Registry 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Service Registry 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Service Registry에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Service Registry는 릴리스 4 Service Registry와 호환됩니다.

종속성

다른 Java ES 구성 요소에 대한 Service Registry 종속성은 Service Registry 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어, Service Registry 인터페이스 또는 기능의 변경 사항을 적용하려면 Service Registry이 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Service Registry는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Service Registry는 특정한 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속됩니다.
- **Application Server.** Service Registry는 Service Registry 응용 프로그램에 컨테이너를 제공하고 Java ES 릴리스 5에서 네트워크로 연결된 레지스트리 / 저장소 데이터베이스에 대한 연결을 관리하는 Application Server에 대해 필수 종속성이 있습니다.
- **Java DB.** Service Registry는 서비스 및 이를 설명하는 메타 데이터를 저장하는 기본 데이터베이스로 사용되는 Java DB에 대해 필수 종속성이 있습니다.

이중 업그레이드

Service Registry와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지](#)의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"에서 설명한 대로 다음 두 가지 방법 중 하나로 수행할 수 있습니다.

- 운영 체제 새로 설치
- 기존 운영 체제 업그레이드

운영 체제 새로 설치

1. 기존 Service Registry 데이터를 백업합니다.
필수 데이터의 위치는 [229페이지](#)의 "Service Registry 데이터"를 참조하십시오.
2. 새 운영 체제를 설치합니다.
운영 체제는 새 시스템(또는 Solaris 10 영역)에 설치하거나 기존 파일 시스템을 덮어쓸 수 있습니다.
3. [단계 1](#)에서 백업한 Service Registry 데이터를 복원합니다.
4. 릴리스 5 Service Registry를 설치합니다.

기존 운영 체제 업그레이드

1. 기존 Service Registry 데이터를 백업합니다.
필수 데이터의 위치는 [229페이지](#)의 "Service Registry 데이터"를 참조하십시오.
2. 운영 체제를 업그레이드합니다.
업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.
3. 릴리스 5 Service Registry로 업그레이드합니다.
[231페이지](#)의 "Java ES 릴리스 4에서 Service Registry 업그레이드"를 참조하십시오.

Java ES 릴리스 4에서 Service Registry 업그레이드

이 절에는 Service Registry를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Service Registry 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Service Registry를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 릴리스 5 Service Registry를 새로 설치하고 릴리스 4 데이터 및 구성을 릴리스 5로 마이그레이션한 다음 릴리스 4를 제거하여 디스크 공간을 확보하면 업그레이드를 수행할 수 있습니다.
- **업그레이드 종속성.** Service Registry는 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Service Registry를 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다.

Service Registry는 Application Server 및 Java DB에 대한 엄격한 업그레이드 종속성이 있습니다.

- **역방향 호환성.** 릴리스 5 Service Registry는 릴리스 4와 완전히 호환됩니다.
- **업그레이드 롤백.** 저장된 데이터베이스와 구성 데이터를 복원한 다음 릴리스 4로 되돌리면 릴리스 5 업그레이드를 롤백할 수 있습니다.
- **플랫폼 문제.** Service Registry 업그레이드를 위한 방법은 Solaris 및 Linux 운영 체제에서 동일합니다.

릴리스 4 Service Registry 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Service Registry를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 업그레이드 Service Registry](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Service Registry를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Service Registry 종속성 업그레이드](#)
- [Service Registry 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

웹 콘솔 사용자 인터페이스의 특성을 살펴보면 Service Registry의 현재 버전을 확인할 수 있습니다.

```
http://localhost:6060/soar
```

또한 Service Registry 패키지 이름을 확인할 수 있습니다. 예를 들면 다음과 같습니다.

Solaris의 경우

```
pkginfo -l|grep srvc
```

Linux의 경우

```
rpm -qa|grep srvc
```

구분 특성 및 패키지 이름은 다음 표에 정리되어 있습니다.

표 12-4 Service Registry 버전 확인 출력

Java ES 릴리스	Service Registry 버전 번호	구분 특성
릴리스 4	3.0	웹 콘솔: 왼쪽 창의 도구 섹션 패키지 이름에 포함되는 문자열: soar
릴리스 5	3.1	웹 콘솔: 왼쪽 창에 있는 세 개의 탭 패키지 이름에 포함되는 문자열: srvc-registry

Service Registry 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Service Registry는 여러 공유 구성 요소, Application Server 및 Java DB에 대해 엄격한 업그레이드 종속성이 있습니다.

Service Registry 종속성을 업그레이드할 때는 Service Registry를 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다. 공유 구성 요소 업그레이드는 일반적으로 Java ES 설치 프로그램을 통해 자동으로 수행됩니다.

1. **공유 구성 요소.** 63페이지의 "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Service Registry에 필요한 모든 공유 구성 요소는 Service Registry를 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.
2. **Java DB.** 153페이지의 8장, "Java DB"에서 Java DB를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **Application Server.** 199페이지의 11장, "Application Server"에서 Application Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

HTTP 포트 번호 수정

ServiceRegistryR4-base/install/install.properties 파일을 수정하여 HTTP 포트를 6060에서 6480으로 변경합니다(6060은 예약된 포트입니다). 이 등록 정보 설정에 대한 자세한 내용은 Service Registry 3.1 관리 설명서(<http://docs.sun.com/doc/820-0536>)를 참조하십시오.

Service Registry 데이터 백업

Service Registry를 릴리스 4에서 릴리스 5로 업그레이드하는 경우 구성 데이터 또는 레지스트리/저장소 데이터베이스는 수정되지 않습니다. 따라서 현재 데이터를 백업하지 않아도 됩니다.

필수 구성 정보 및 비밀번호 얻기

릴리스 4 Service Registry의 사용자 아이디, 비밀번호, 도메인 이름 및 포트 번호를 알아야 합니다.

릴리스 4 업그레이드 Service Registry

이 절에서는 Solaris 및 Linux 플랫폼에서의 업그레이드 절차를 설명합니다.

업그레이드 절차(Solaris)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 로컬에 상주하는 Service Registry 인스턴스에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. Jakarta ANT Java/XML 기반 빌드 도구(ANT 공유 구성 요소)가 올바른 J2SE 버전을 참조하는지 확인합니다.

(이어지는 단계에서 ant 명령이 사용됩니다.)

```
PATH=/usr/jdk/entsys-j2se/bin:$PATH
```

```
export PATH
```

3. 릴리스 4 Service Registry(Application Server) 도메인을 중지합니다.

```
cd ServiceRegistryR4-base/install
```

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.stop
```

도메인은 Service Registry 인스턴스와 연관됩니다.

4. 릴리스 5 Service Registry를 새로 설치합니다.

다음 단계를 수행합니다.

- a. 릴리스 4 Service Registry를 호스팅하는 컴퓨터에서 Java ES 설치 프로그램을 실행합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다. 명령줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. 구성 요소 선택 페이지에서 Service Registry를 선택합니다.

- c. 릴리스 4와 다른 설치 디렉토리 경로를 지정합니다.

기본적으로 릴리스 5 설치 경로(*ServiceRegistryR5-base*)는 릴리스 4 설치 경로(*ServiceRegistryR4-base*)와 다릅니다.

- d. "나중에 구성" 옵션을 선택합니다.

"지금 구성"은 지원되지 않습니다.

- e. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.

- f. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.

5. 릴리스 5 Service Registry 인스턴스를 업그레이드 및 구성합니다.

```
cd ServiceRegistryR5-base/install
/usr/sfw/bin/ant -f build-install.xml
-Dinstall.properties=ServiceRegistryR4-base/install/install.properties
upgrade
```

릴리스 4 `install.properties` 파일을 가리키는 또 다른 방법으로 기본 릴리스 5 `install.properties` 파일을 편집하면 모든 릴리스 4 등록 정보 값을 그대로 사용할 수 있습니다. 이러한 등록 정보 설정에 대한 자세한 내용은 Service Registry 3.1 관리 설명서(<http://docs.sun.com/doc/820-0536>)를 참조하십시오.

사용자 정의 등록 정보 값을 사용 중이지만 이를 `install.properties`에 추가하지 않은 경우 다음과 같이 Ant 명령줄에 이러한 등록 정보 값을 한 줄에 모두 지정해야 합니다.

```
/usr/sfw/bin/ant -f build-install.xml
-Dregistry.install.RegistryServerKeystorePassword=password1
-Dregistry.install.AdministratorPassword=password2
-Dregistry.install.ApplicationServerKeystorePassword=password3
upgrade
```

그러나 이러한 사용자 정의 등록 정보 값은 제한된 권한으로 `install.properties` 파일에 추가하여 인증되지 않은 사용자가 볼 수 있는 명령줄 설정 사용을 방지하는 것이 좋습니다. 자세한 내용은 Service Registry 관리 설명서를 참조하십시오.

업그레이드 유틸리티는 새 Application Server 도메인을 만들고 도메인을 시작하며 도메인에 Service Registry 인스턴스를 배포합니다. 각 Service Registry 인스턴스는 자체 Application Server 도메인과 연관됩니다.

6. 릴리스 4 Service Registry의 서버 등록 정보 파일이 수정된 경우 다음과 같이 릴리스 5 Service Registry 구성 파일에 대해 해당 변경 작업을 수행할 수 있습니다.

- a. 릴리스 5 Service Registry(Application Server) 도메인을 중지합니다.

이 도메인은 [단계 5](#)의 `upgrade` 명령을 사용하여 자동으로 시작되었습니다.

```
cd ServiceRegistryR5-base/install
/usr/sfw/bin/ant -f build-install.xml appserver.domain.stop
```

- b. 릴리스 4 Service Registry 인스턴스 구성을 릴리스 5로 전송합니다.

릴리스 4 Service Registry 인스턴스 구성에 대한 변경 사항을 추가합니다.

```
RegistryDomainR4-base/domains/registry/applications/j2ee-modules/
soar/WEB-INF/classes/*.properties
```

변경 사항을 해당하는 릴리스 5 구성에 추가합니다.

```
RegistryDomainR5-base/domains/registry/applications/j2ee-modules/
soar/WEB-INF/classes/*.properties
```

7. 릴리스 5 Service Registry(Application Server) 도메인을 시작합니다.

```
cd ServiceRegistryR5-base/install
/usr/sfw/bin/ant -f build-install.xml appserver.domain.start
```

업그레이드 절차(Linux)

Linux에서 Service Registry를 업그레이드하는 절차는 Solaris의 절차([234페이지의 "업그레이드 절차\(Solaris\)"](#) 참조)와 동일하지만, Linux 플랫폼에서는 업그레이드 절차의 다양한 단계에서 사용되는 `ant` 명령의 위치가 Solaris 플랫폼과 다릅니다.

```
/opt/sun/share/bin/ant
```

업그레이드 확인

웹 콘솔 사용자 인터페이스의 특성을 살펴보면 Service Registry가 성공적으로 업그레이드되었는지 확인할 수 있습니다.

```
http://localhost:6480/soar
```

또한 Service Registry 패키지 이름을 확인할 수 있습니다. 예를 들면 다음과 같습니다.

Solaris의 경우

```
pkginfo -l|grep soar
```

Linux의 경우

```
rpm -qa|grep soar
```

구분 특성 및 패키지 이름은 [233페이지 표 12-4](#)에 정리되어 있습니다.

사후 업그레이드 작업

릴리스 4 Service Registry를 제거하는 방법을 설명하는 다음 단계는 릴리스 4로 업그레이드 롤백을 원하지 않는 경우에만 수행하십시오.

1. 릴리스 4 Service Registry(Application Server) 도메인을 삭제합니다.

```
cd ServiceRegistryR4-base/install
```

Solaris의 경우

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.delete
```

Linux의 경우

```
/opt/sun/bin/ant -f build-install.xml appserver.domain.delete
```

2. 릴리스 4 Service Registry 도메인 파일이 포함된 디렉토리를 삭제합니다.

```
rm -rf RegistryDomainR4-base
```

3. 릴리스 4 Service Registry 설치 파일이 포함된 디렉토리를 삭제합니다.

```
rm -rf ServiceRegistryR4-base
```

업그레이드 롤백

릴리스 5로 업그레이드한 다음에도 수정되지 않는 이전 버전으로 되돌리면 릴리스 5 업그레이드를 롤백할 수 있습니다.

1. 릴리스 5 Service Registry(Application Server) 도메인을 중지하고 삭제합니다.

```
cd ServiceRegistryR4-base/install
```

Solaris의 경우

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.delete
```

Linux의 경우

```
/opt/sun/bin/ant -f build-install.xml appserver.domain.delete
```

2. Java ES 릴리스 5 제거 프로그램을 실행하여 릴리스 5 Service Registry를 제거합니다.

3. 릴리스 4 Service Registry 도메인을 시작합니다.

```
cd ServiceRegistryR4-base/install
```

Solaris의 경우

```
/usr/sfw/bin/ant -f build-install.xml appserver.domain.start
```

Linux의 경우

```
/opt/sun/bin/ant -f build-install.xml appserver.domain.start
```

4. 릴리스 4 Service Registry 웹 콘솔에 액세스합니다.

```
http://localhost:6480/soar
```

5. [233페이지의 "Service Registry 버전 확인 출력"](#)에 나와 있는 릴리스 4 특징이 콘솔에 표시되는지 확인합니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Service Registry를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Service Registry 인스턴스를 실행하여 로드를 분산시킬 수 있습니다.

이러한 구조에서 레지스트리는 거의 대부분 읽기 전용이며 공통 데이터베이스에 액세스하여 과중한 쿼리 로드 응답합니다.

[232페이지의 "릴리스 4 Service Registry 업그레이드"](#)에서 설명하는 대로 각 컴퓨터에서 Service Registry를 업그레이드하십시오.

Web Proxy Server

이 장에서는 Web Proxy Server를 Java ES 5(릴리스 5): Sun Java System Web Proxy Server 4.0.4로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- [240페이지의 "Web Proxy Server 업그레이드 개요"](#)
- [243페이지의 "Java ES 릴리스 4에서 Web Proxy Server 업그레이드"](#)
- [250페이지의 "버전 3.6에서 Web Proxy Server 업그레이드"](#)

주 이 장에 설명된 파일 위치는 *WebProxyServer-base*라는 디렉토리 경로에 상대적으로 지정되어 있습니다. Web Proxy Server를 처음 설치할 때 최소한 이 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에 나와 있는 것처럼 *WebProxyServer-base*의 기본값은 운영 체제 플랫폼에 따라 달라집니다.

표 13-1 Web Proxy Server 디렉토리 경로

경로 이름 변수	Solaris OS	Linux OS
<i>WebProxyServer-base</i>	/opt/SUNWproxy	/opt/sun/webproxyserver

Web Proxy Server 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Web Proxy Server의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Web Proxy Server 정보](#)
- [Web Proxy Server 업그레이드 로드맵](#)
- [Web Proxy Server 데이터](#)
- [Web Proxy Server 업그레이드 전략](#)

Java ES 릴리스 5 Web Proxy Server 정보

Java ES 릴리스 5 Web Proxy Server는 릴리스 4와 비교할 때 버그 수정 릴리스입니다.

그러나 릴리스 5 Web Proxy Server는 이전에 Java Enterprise System에 포함되었던 Sun One Web Proxy Server 3.6에 비해 더 나은 성능, 더 확장된 구조, 표준 호환성 및 새로운 관리 인터페이스를 포함합니다.

Web Proxy Server 업그레이드 로드맵

[표 13-2](#)는 Web Proxy Server를 Java ES 릴리스 5로 업그레이드하는 경로를 보여 줍니다. Web Proxy Server는 이전 Java ES 릴리스에는 포함되어 있지 않습니다. Web Proxy Server는 이전에 Linux OS에서는 지원되지 않았으므로 이 표는 Solaris OS에만 적용됩니다.

표 13-2 Java ES 5(릴리스 5) 업그레이드 경로: Web Proxy Server 4.0.4

Java ES 릴리스	Web Proxy Server 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Web Proxy Server 4.0.1 2005Q4	직접 업그레이드: 패치를 사용하여 수행합니다.	없음
이전 Java ES 릴리스 (Solaris OS 전용)	Sun ONE Web Proxy Server 3.6 (이하, 버전 3.6)	직접 업그레이드: Java ES 설치 프로그램을 사용하여 새 위치에 설치한 다음 관리 도구를 사용하여 구성 데이터를 마이그레이션하여 수행할 수 있습니다.	구성 정보를 반드시 새 위치로 마이그레이션해야 합니다.

Web Proxy Server 데이터

다음 표는 Web Proxy Server 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 13-3 Web Proxy Server 데이터 사용

데이터 유형	위치	사용
구성 데이터	<i>WebProxyServer-base/proxy-serverid/</i> config 디렉토리 <i>server.xml, magnus.conf, obj.conf</i> 등의 파일 포함	서버, 캐시, 필터, 라우팅 및 Web Proxy Server 의 기타 기능적인 측면에 대한 구성 정보 저장

Web Proxy Server 업그레이드 전략

Web Proxy Server 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Web Proxy Server 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Web Proxy Server에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Web Proxy Server에는 새로운 공용 인터페이스가 추가되지 않았으므로 릴리스 4 Web Proxy Server와 호환됩니다. 또한 릴리스 5 Web Proxy Server는 버전 3.6 과도 호환됩니다. 단, 버전 3.6에서 지원하는 NSAPI 인터페이스를 사용해 개발한 플러그인은 릴리스 5에서 지원하는 NSAPI 인터페이스로 다시 컴파일해야 합니다.

Web Proxy Server 종속성

Web Proxy Server는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Web Proxy Server는 특정한 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속됩니다.
- **Directory Server.** Web Proxy Server는 LDAP 기반 인증을 제공하는 Directory Server에 대해 선택적 종속성이 있습니다.
- **Web Server.** Web Proxy Server는 HTTP 요청에 대한 향상된 보안과 성능을 제공하는 Web Server에 대해 상호 종속성이 있습니다.

이중 업그레이드

Web Proxy Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지](#)의 "[이중 업그레이드: Java ES 및 운영 체제 소프트웨어](#)"에서 설명한 대로 기존 운영 체제 업그레이드 방식을 사용하여 수행할 수 있습니다.

1. 기존 Web Proxy Server 데이터를 백업합니다.
필수 데이터의 위치는 [241페이지](#)의 "[Web Proxy Server 데이터](#)"를 참조하십시오.
2. 운영 체제를 업그레이드합니다.
업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.
3. 릴리스 5 Web Proxy Server로 업그레이드합니다.
업그레이드 방법에 따라 이 장에서 해당 절을 참조하십시오.

Java ES 릴리스 4에서 Web Proxy Server 업그레이드

이 절에는 Web Proxy Server를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Web Proxy Server 업그레이드](#)

소개

Web Proxy Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 패치를 사용하여 업그레이드를 수행합니다. 추가 재구성은 필요하지 않습니다.
- **업그레이드 종속성.** Web Proxy Server는 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되지만 Web Proxy Server는 NSS 및 NSPR 공유 구성 요소에 대해서만 엄격한 업그레이드 종속성이 있습니다.
- **역방향 호환성.** 릴리스 5 Web Proxy Server는 릴리스 4와 호환됩니다.
- **업그레이드 롤백.** 업그레이드 패치를 제거하여 Web Proxy Server의 릴리스 5 업그레이드를 롤백할 수 있습니다.
- **플랫폼 문제.** Web Proxy Server 업그레이드를 위한 일반적인 방법은 Solaris 및 Linux 운영 체제에서 동일하지만 패치 기술에 차이가 있습니다. 그러므로 업그레이드 프로세스에는 플랫폼별 절차가 포함됩니다.

릴리스 4 Web Proxy Server 업그레이드

이 절에서는 Web Proxy Server를 Java ES 릴리스 5로 업그레이드하는 방법에 대한 개요를 제공하며 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 Web Proxy Server 업그레이드\(Solaris\)](#)
- [릴리스 4 Web Proxy Server 업그레이드\(Linux\)](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백\(Solaris\)](#)

사전 업그레이드 작업

Web Proxy Server를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Web Proxy Server 종속성 업그레이드](#)
- [Web Proxy Server 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

다음 명령을 입력하여 Web Proxy Server의 현재 버전을 확인할 수 있습니다.

```
WebProxyServer-base/proxy-admserv/start -version
```

표 13-4 Web Proxy Server 버전 확인 출력

Java ES 릴리스	Web Proxy Server 버전 번호
비Java ES 릴리스 버전 3.6	3.6
릴리스 4	4.0.1
릴리스 5	4.0.4

Web Proxy Server 종속성 업그레이드

일반적으로 컴퓨터 시스템 및 컴퓨팅 환경 내의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. 그러나 Web Proxy Server는 NSS 및 NSPR 공유 구성 요소가 업그레이드될 때만 릴리스 5로 업그레이드해야 합니다. 이러한 공유 구성 요소가 업그레이드되지 않은 경우 "공유 구성 요소 동기화" 옵션을 사용하여 모든 공유 구성 요소를 해당 릴리스 5 버전으로 동기화해야 합니다. 자세한 내용은 [63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"](#)를 참조하십시오.

Web Proxy Server 데이터 백업

Web Proxy Server를 릴리스 5로 업그레이드하더라도 릴리스 4 구성 데이터가 수정되는 것은 아닙니다. 따라서 현재 데이터를 백업하지 않아도 됩니다.

필수 구성 정보 및 비밀번호 얻기

현재 설치된 버전에 대한 특수 정보가 필요하지 않습니다. 그러나 업그레이드를 수행하려면 슈퍼유저로 로그인해야 합니다.

릴리스 4 Web Proxy Server 업그레이드(Solaris)

이 절에서는 Web Proxy Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항(Solaris)

Web Proxy Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- 동일한 설치 Web Proxy Server 이미지에 해당하는 모든 Web Proxy Server 인스턴스가 동시에 업그레이드됩니다. 설치 이미지에 패치를 적용 중인 경우 그러한 인스턴스를 모두 종료해야 합니다.
- 다음 표에 Solaris OS용 릴리스 5 Web Proxy Server 업그레이드 패치가 표시되어 있습니다.

표 13-5 Solaris에서 Web Proxy Server를 업그레이드하기 위한 패치¹

설명	패치 아이디: SPARC Solaris 9 및 10	패치 아이디: X86 Solaris 9 및 10
Web Proxy Server Core	120981-10	120982-10
Web Proxy Server 현지화	122963-01	122964-01

1. 패치 버전 번호는 Java ES 릴리스 5로 업그레이드하는 데 필요한 최소 버전입니다. 보다 최신 버전이 있으면 표에 표시된 버전 대신 최신 버전을 사용하십시오.

업그레이드 절차(Solaris)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Web Proxy Server에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 실행 중인 모든 Web Proxy Server 및 Administration Server 인스턴스를 중지합니다.

```
WebProxyServer-base/proxy-instanceName/stop
```

```
WebProxyServer-base/proxy-admserv/stop
```

3. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

자세한 내용은 63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"를 참조하십시오.

4. 표 13-5를 기준으로 필수 패치를 확인합니다.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

5. 적합한 Web Proxy Server Core를 적용하고 필요한 경우 표 13-5의 현지화 패치를 순서에 맞게 적용합니다.

```
patchadd /tmp/patch_ID
```

6. 패치가 성공적으로 업그레이드되었는지 확인합니다.

```
showrev -p | grep proxy
```

단계 5에서 적용된 패치 아이디의 버전이 출력으로 반환되어야 합니다.

7. 단계 2에서 중지된 Web Proxy Server 인스턴스를 다시 시작합니다.

```
WebProxyServer-base/proxy-instanceName/start
```

릴리스 4 Web Proxy Server 업그레이드(Linux)

이 절에서는 Web Proxy Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항(Linux)

Linux 플랫폼에서 Web Proxy Server 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 경우 Solaris 플랫폼에서 업그레이드할 때의 고려 사항을 참조하십시오(252페이지의 "업그레이드 고려 사항" 참조). Linux 릴리스 5 업그레이드 패치가 Solaris 패치와 다른 경우는 제외됩니다.

다음 표에 Linux OS용 릴리스 5 Web Proxy Server 업그레이드 패치가 표시되어 있습니다.

표 13-6 Linux에서 Web Proxy Server를 업그레이드하기 위한 패치¹

설명	패치 아이디 및 RPM 이름
Web Proxy Server Core	120983-10 sun-proxyserver-4.0-6.5.i386.rpm
Web Proxy Server 현지화	122965-01 sun-proxyserver-<i>Locale</i>-4.0.4.i386.rpm

1. 패치 버전 번호는 Java ES 릴리스 5로 업그레이드하는 데 필요한 최소 버전입니다. 보다 최신 버전이 있으면 표에 표시된 버전 대신 최신 버전을 사용하십시오.

업그레이드 절차(Linux)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Web Proxy Server에 적용됩니다.

주의	Linux에서 Java ES 릴리스 4를 Java ES 릴리스 5로 업그레이드하는 작업은 롤백할 수 없습니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 실행 중인 모든 Web Proxy Server 및 Administration Server 인스턴스를 중지합니다.

```
WebProxyServer-base/proxy-instanceName/stop
```

```
WebProxyServer-base/proxy-admserv/stop
```

3. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

245페이지의 "Web Proxy Server 종속성 업그레이드"를 참조하십시오.

4. 표 13-6의 패치 번호와 RPM 이름을 이용하여 필요한 패치를 확인합니다. RPM의 버전 번호를 확인할 때도 이 정보를 사용합니다.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

5. 코어를 적용하고, 필요한 경우 표 13-6의 Web Proxy Server 현지화 RPM을 순서에 맞게 적용합니다.

```
rpm -Fvh sun-proxyserver-version.i386.rpm
```

6. 패치가 성공적으로 업그레이드되었는지 확인합니다.

```
rpm -qa | grep sun-proxyserver
```

RPM의 새로운 버전 번호가 반환되어야 합니다.

7. 단계 2에서 중지된 Web Proxy Server 인스턴스를 다시 시작합니다.

```
WebProxyServer-base/proxy-instanceName/start
```

업그레이드 확인

-version 옵션으로 Web Proxy Server 인스턴스를 시작하여 Web Proxy Server 릴리스 5 업그레이드를 확인할 수 있습니다.

```
WebProxyServer-base/proxy-admserv/start -version
```

출력 값에 대해서는 244페이지 표 13-4를 참조하십시오.

사후 업그레이드 작업

사후 업그레이드 작업은 246페이지의 "업그레이드 절차(Solaris)" 및 247페이지의 "업그레이드 절차(Linux)"에 설명된 단계에 모두 포함됩니다.

업그레이드 롤백(Solaris)

이 절에서는 Web Proxy Server의 업그레이드 롤백 절차에 영향을 미치는 고려 사항에 대해 설명하고 뒤이어 절차 자체에 대해 설명합니다.

롤백 고려 사항(Solaris)

Web Proxy Server 릴리스 5로의 업그레이드를 롤백하는 절차는 릴리스 5로 업그레이드하는 순서를 역으로 수행하는 것과 매우 비슷합니다.

롤백 절차(Solaris)

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 실행 중인 모든 Web Proxy Server 및 Administration Server 인스턴스를 중지합니다.

```
WebProxyServer-base/proxy-instanceName/stop
```

```
WebProxyServer-base/proxy-admserv/stop
```


3. [245페이지 표 13-5](#)의 패치를 제거합니다.

```
patchrm patch_ID
```

4. [단계 2](#)에서 중지된 Web Proxy Server 인스턴스를 다시 시작합니다.

```
WebProxyServer-base/proxy-instanceName/start
```

버전 3.6에서 Web Proxy Server 업그레이드

이 절에서는 Web Proxy Server를 버전 3.6에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [버전 3.6 Web Proxy Server 업그레이드](#)

소개

Web Proxy Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** Java ES 설치 프로그램을 사용하여 릴리스 5 Web Proxy Server를 버전 3.6과 다른 디렉토리에 설치하여 업그레이드를 수행합니다. 그런 다음 Web Proxy Server Administration Server를 사용하여 구성 설정(캐시 내용 제외)을 버전 3.6에서 릴리스 5로 마이그레이션합니다.
- **업그레이드 종속성.** 공유 구성 요소는 Web Proxy Server를 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램에서 자동으로 설치합니다.
- **역방향 호환성.** 릴리스 5 Web Proxy Server는 버전 3.6과 호환됩니다. 단, 버전 3.6에서 지원하는 NSAPI 인터페이스를 사용해 개발한 플러그인은 릴리스 5에서 지원하는 NSAPI 인터페이스로 다시 컴파일해야 합니다.
- **업그레이드 롤백.** Web Proxy Server의 릴리스 5 업그레이드 롤백은 업그레이드에 의해 변경되지 않은 버전 3.6으로 되돌리는 방법으로 수행됩니다.
- **플랫폼 문제.** Web Proxy Server를 업그레이드하기 위한 방법은 모든 Solaris 플랫폼에서 동일하지만 Linux 플랫폼에서는 버전 3.6이 지원되지 않습니다.

버전 3.6 Web Proxy Server 업그레이드

이 절에서는 Web Proxy Server를 Java ES 릴리스 5로 업그레이드하는 방법에 대한 개요를 설명합니다. Web Proxy Server는 이전에 Linux 플랫폼에서 지원되지 않았습니니다. 따라서 Java ES 릴리스 5로의 Web Proxy Server 업그레이드는 Solaris 플랫폼에서만 수행됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [버전 3.6 Web Proxy Server 업그레이드](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [버전 3.6 업그레이드 롤백](#)

사전 업그레이드 작업

Web Proxy Server를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- [현재 버전 정보 확인](#)
- [Web Proxy Server 종속성 업그레이드](#)
- [Web Proxy Server 데이터 백업](#)
- [필수 구성 정보 및 비밀번호 얻기](#)

현재 버전 정보 확인

다음 명령을 입력하여 Web Proxy Server의 현재 버전을 확인할 수 있습니다.

```
WebProxyServer-base/proxy-serverid/start -version
```

표 13-7 Web Proxy Server 버전 확인 출력

Java ES 릴리스	Web Proxy Server 버전 번호
비Java ES 릴리스 버전 3.6	3.6
릴리스 4	4.0.1
릴리스 5	4.0.4

Web Proxy Server 종속성 업그레이드

일반적으로 컴퓨터 시스템(및 컴퓨팅 환경)의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. 그러나 Web Proxy Server를 릴리스 5로 업그레이드하는 Java ES 설치 프로그램이 Web Proxy Server가 종속되어 있는 모든 공유 구성 요소를 자동으로 업그레이드합니다([47페이지 표 1-9](#) 참조).

Web Proxy Server 데이터 백업

Web Proxy Server를 릴리스 5로 업그레이드하더라도 버전 3.6의 구성 데이터가 수정되는 것은 아닙니다. 그러나 관리 인터페이스를 사용하여 버전 3.6의 구성 데이터에 변경한 내용이 있으면 업그레이드 전에 저장해야 합니다.

필수 구성 정보 및 비밀번호 얻기

버전 3.6에서 업그레이드하려면 설치된 버전에 대한 설치 디렉토리 경로를 알아야 합니다.

버전 3.6 Web Proxy Server 업그레이드

이 절에서는 Web Proxy Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항

동일한 설치 Web Proxy Server 이미지에 해당하는 모든 Web Proxy Server 인스턴스를 업그레이드할 수 있습니다. 그러나 구성 데이터의 마이그레이션은 각 인스턴스에 대해 별도로 수행해야 합니다. 마이그레이션된 인스턴스가 시작될 때 포트 충돌이 발생하지 않도록 마이그레이션을 수행할 때 이러한 모든 인스턴스를 종료해야 합니다.

업그레이드 절차

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Web Proxy Server 소프트웨어에 적용됩니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

su -

2. Web Proxy Server 릴리스 5를 설치합니다.
 - a. Java ES 릴리스 5 배포에서 Java ES 설치 프로그램을 실행합니다.
 - b. 구성 요소 선택 페이지에서 Web Proxy Server를 선택합니다.
 - c. "지금 구성" 옵션을 선택합니다.
 - d. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.

3. 구성 설정을 새로 설치된 버전으로 마이그레이션합니다.

이 작업은 각 Web Proxy Server 인스턴스에 대해 별도로 수행해야 합니다.

a. Web Proxy Server Administration Server를 시작합니다.

`WebProxyServer-base/proxy-admserv/start`

b. 그래픽 관리 인터페이스에 로그인합니다.

c. "서버" 탭을 누른 다음 "서버 마이그레이션"을 누릅니다.

d. 버전 3.6 설치 디렉토리 경로를 지정합니다.

e. 마이그레이션할 인스턴스를 선택합니다.

f. "Migrate" 버튼을 누릅니다.

마이그레이션을 완료한 후 수동으로 수행해야 하는 추가 구성 목록이 마이그레이션 화면에 표시됩니다. 추가해야 하는 데이터와 해당 구성 파일도 제공됩니다.

구성 설정 마이그레이션에 대한 자세한 내용은 Sun Java System Web Proxy Server 4.0.4 Installation and Migration Guide

(<http://docs.sun.com/doc/819-5492>)를 참조하십시오.

4. 단계 f에서 지정한 추가 구성 변경을 수행합니다.

자세한 내용은 Sun Java System Web Proxy Server 4.0.4 Configuration File Reference(<http://docs.sun.com/doc/819-5494>)를 참조하십시오.

업그레이드 확인

-version 옵션으로 Web Proxy Server 인스턴스를 시작하여 Web Proxy Server 릴리스 5 업그레이드를 확인할 수 있습니다.

`WebProxyServer-base/proxy-serverid/start -version`

출력 값에 대해서는 [251페이지 표 13-7](#)을 참조하십시오.

사후 업그레이드 작업

사후 업그레이드 작업은 [252페이지의 "업그레이드 절차"](#)에 설명된 단계에 모두 포함됩니다.

버전 3.6 업그레이드 롤백

[252페이지의 "버전 3.6 Web Proxy Server 업그레이드"](#)에서 설명한 릴리스 5로의 Web Proxy Server 업그레이드는 롤백할 수 없습니다. 그러나 릴리스 5 업그레이드 절차로 수정되지 않는 버전 3.6으로 되돌리는 것은 가능합니다.

버전 3.6에서 Web Proxy Server 업그레이드

Access Manager

이 장에서는 Access Manager 소프트웨어를 이전 Java ES 버전에서 Java ES 5(릴리스 5): Sun Java System Access Manager 7.1로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 Access Manager 업그레이드 문제 및 절차의 일반 개요를 제공합니다. 이 장에서는 Solaris와 Linux 운영 체제에서의 업그레이드에 대해 설명합니다.

- [256페이지의 "Access Manager 업그레이드 개요"](#)
- [262페이지의 "Java ES 릴리스 4에서 Access Manager 업그레이드"](#)
- [285페이지의 "Java ES 릴리스 3에서 Access Manager 업그레이드"](#)
- [289페이지의 "Java ES 릴리스 2에서 Access Manager 업그레이드"](#)

주 이 장에 설명된 파일 위치는 *AccessManager-base* 및 *AccessManagerConfig-base*라는 두 개의 디렉토리 경로에 상대적으로 지정되어 있습니다. Access Manager를 처음 설치할 때 최소한 이 두 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 14-1 Access Manager 디렉토리 경로

경로 이름 변수	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 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Access Manager의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Access Manager 정보](#)
- [Access Manager 업그레이드 로드맵](#)
- [Access Manager 데이터](#)
- [Access Manager 업그레이드 전략](#)

주 Java ES 릴리스 3 이전에 출시된 Access Manager의 버전은 Identity Server라고 불립니다. 따라서 이 장에서 Identity Server는 Java ES Access Manager 구성 요소의 이전 버전을 의미합니다.

Java ES 릴리스 5 Access Manager 정보

Java ES 릴리스 5 Access Manager는 부차적 릴리스이며, 주요 릴리스인 Java ES 릴리스 4 Access Manager에서 향상된 기능과 여러 가지 버그가 수정된 기능이 포함되어 있습니다. 릴리스 5에서 향상된 여러 기능 중에는 Java ES 모니터링 프레임워크를 바탕으로 하는 새로운 모니터링 기능이 있습니다. 릴리스 5에서 향상된 기능에 대한 자세한 내용은 Sun Java System Access Manager 7.1 릴리스 노트

(<http://docs.sun.com/doc/820-0364>)를 참조하십시오.

릴리스 4와 유사하게 릴리스 5 Access Manager는 여러 Identity 저장소 또는 사용자 데이터 저장소를 지원합니다. 따라서 릴리스 5 Access Manager는 Directory Server와 같은 LDAP 디렉토리 뿐만 아니라 다른 데이터 저장소 프로토콜 및 형식도 지원합니다.

프런트 엔드에서는 Access Manager 콘솔을 사용하여 새 Access Manager 서비스와 Identity 저장소를 구성합니다.

이전 버전의 다른 Java ES 구성 요소와 호환될 수 있도록 릴리스 3 Access Manager 서비스에 종속된 Java ES 구성 요소를 지원하는 레거시 모드로 릴리스 5를 실행할 수 있습니다. 자세한 내용은 [259페이지의 "호환성 문제"](#)를 참조하십시오.

Access Manager 업그레이드 로드맵

표 14-2는 Access Manager를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 14-2 Java ES 5(릴리스 5) 업그레이드 경로: Access Manager 7.1

Java ES 릴리스	Access Manager 버전	일반적인 방법	재구성 필요
릴리스 4	Sun Java System Access Manager 7.0 2005Q4	직접 업그레이드: 사전 업그레이드 스크립트를 사용하여 릴리스 4 버전을 제거한 다음 릴리스 5를 전체 설치 및 재구성하여 수행합니다.	구성 데이터 Access Manager 콘솔 및 인증 UI용 사용자 정의 JSP 디렉토리 스키마
릴리스 3	Sun Java System Access Manager 6.3 2005Q1	직접 업그레이드: 사전 업그레이드 스크립트를 사용하여 릴리스 3 버전을 제거한 다음 릴리스 5를 전체 설치 및 재구성하여 수행합니다.	구성 데이터 Access Manager 콘솔 및 인증 UI용 사용자 정의 JSP 디렉토리 스키마
릴리스 2	Sun Java System Identity Server 6.2 2004Q2 및 6.2 SP1	직접 업그레이드: 사전 업그레이드 스크립트를 사용하여 릴리스 2 버전을 제거한 다음 릴리스 5를 전체 설치 및 재구성하여 수행합니다.	구성 데이터 Access Manager 콘솔 및 인증 UI용 사용자 정의 JSP 디렉토리 스키마
릴리스 1	Sun ONE Identity Server 6.1	직접 업그레이드 불가능: 그러나 Java Enterprise System 2005Q1 Upgrade and Migration Guide (http://docs.sun.com/doc/819-0062)의 절차를 사용하여 릴리스 3으로 처음 업그레이드하는 것은 가능합니다. 그런 다음 릴리스 3에서 릴리스 5로 업그레이드합니다.	구성 데이터 Access Manager 콘솔 및 인증 UI용 사용자 정의 JSP 디렉토리 스키마
이전 Java ES 릴리스	Sun ONE Identity Server 6.0 또는 6.0 SP 1 또는 iPlanet Directory Server Access Management Edition(DSAME) 5.1	직접 업그레이드할 수 없습니다.	

Access Manager 데이터

다른 Java ES 구성 요소처럼, Access Manager는 해당 업그레이드된 버전으로의 마이그레이션이 필요할 수 있는 다양한 데이터를 사용합니다. 다음 표는 Access Manager 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 14-3 Access Manager 데이터 사용

데이터 유형	위치	사용
구성 데이터	<i>AccessManagerConfig-base/config/AMConfig.properties</i> <i>AccessManagerConfig-base/config/serverconfig.xml</i> 인증 및 사용자 정의된 모듈용 JAR 파일 <i>AccessManager-base/lib</i>	Access Manager의 구성 및 백엔드 데이터 저장소와의 통합
웹 컨테이너 액세스 제어 및 구성 파일	Web Server 7.0(Java ES 릴리스 5) server.policy 및 server.xml 파일의 경우 <i>WebServer7Config-base/https-configName/config</i> Web Server 6.x(Java ES 릴리스 2, 3 및 4) server.policy 및 server.xml 파일의 경우 <i>WebServer6-base/https-hostname/config</i> Application Server 8.x(Java ES 릴리스 3, 4 및 5) server.policy 및 domain.xml 파일의 경우 <i>AppServer8Config-base/domains/domainName/config</i> Application Server 7.x(Java ES 릴리스 2): server.policy 및 server.xml 파일의 경우 <i>AppServer7Config-base/domains/domainName/config</i> WebSphere 및 WebLogic: Access Manager가 이러한 웹 컨테이너에 대해 구성되면 해당 정책 및 구성 파일이 수정됩니다.	Access Manager 웹 컨테이너 인스턴스의 구성
사용자 정의 데이터 (웹 컨테이너 사용자 정의 JSP 파일)	관리 콘솔: (Java ES 릴리스 2 및 3): <i>AccessManager-base/web-src/applications</i> 관리 콘솔: (Java ES 릴리스 4 및 5): <i>AccessManager-base/web-src/services</i> 인증 UI: <i>AccessManager-base/web-src/services</i>	Access Manager 관리 인터페이스의 구성
디렉토리 스키마 서비스 구성 사용자 데이터	Directory Server	Access Manager는 서비스 구성, 사용자 및 디렉토리에 저장되어 있는 정책 데이터를 기반으로 최종 사용자에게 인증 및 권한 부여 서비스를 제공합니다.
동적 응용 프로그램 데이터	없음	Access Manager는 세션 상태와 같은 응용 프로그램 데이터를 지속적으로 저장하지 않습니다.

Access Manager 업그레이드 전략

Access Manager 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Access Manager 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Access Manager에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Access Manager는 릴리스 4 Access Manager와 호환되지만 릴리스 4 Access Manager는 레거시 모드로 실행하도록 구성할 때를 제외하고 이전 릴리스와의 호환성이 제공되지 않는 주요 릴리스입니다. 비슷하게 릴리스 5 Access Manager는 레거시 모드에서 실행하도록 구성하지 않으면 릴리스 3 Access Manager(또는 레거시 모드에서 실행되는 릴리스 4 Access Manager)와 호환되지 않습니다.

또한 릴리스 5 Access Manager는 릴리스 2 Access Manager와 어떤 모드에서도 호환되지 않습니다. 릴리스 5 Access Manager는 릴리스 2 Access Manager SDK와 상호 작용할 수 없으며 *반대*로 상호 작용되지도 않습니다.

새로운 영역 모드에서 실행하도록 구성하는 경우 릴리스 5 Access Manager는 여러 Identity 저장소 및 데이터 저장소 프로토콜을 지원합니다. 영역 모드 작업을 지원하려면 디렉토리 데이터를 새 구조로 마이그레이션해야 합니다. 또한 영역 모드에서는 Portal Server와 같은 다른 Java ES 구성 요소 또는 Communications Express, Messaging Server 등과 같은 Sun Java Communications Suite 구성 요소가 지원되지 않습니다.

릴리스 5 Access Manager는 레거시 모드에서 실행하도록 구성하는 경우 사소한 예외(Sun Java System Access Manager 7.1 릴리스 노트 (<http://docs.sun.com/doc/820-0364>) 참조)가 있지만 릴리스 3 Access Manager 및 관련 디렉토리 데이터와 호환됩니다.

영역 모드에서 Access Manager와 상호 작용할 수 없는 Access Manager 정책 에이전트의 이전 버전 뿐 아니라 다른 Java ES 구성 요소를 지원하려면 레거시 모드가 필요 합니다. 이러한 비호환성은 중요한 업그레이드 고려 사항으로, Java ES를 배포하는 대부분의 경우 Access Manager를 릴리스 5 레거시 모드로 업그레이드해야 한다는 것을 의미합니다.

그러나 릴리스 5 Access Manager는 레거시 모드에서 실행하도록 구성해도 릴리스 3 이하 버전의 Sun Java Communications Suite 구성 요소와 호환되지 않습니다. 또한 Access Manager를 릴리스 5로 업그레이드한 경우 Messaging Server 및 Calendar Server 사용자를 지정할 수 있도록 릴리스 3 이전의 Delegated Administrator를 릴리스 5로 업그레이드해야 합니다. 그러나 Messaging Server 및 Calendar Server 자체를 릴리스 5로 업그레이드할 필요는 없습니다.

릴리스 5 Access Manager 콘솔은 릴리스 4 콘솔과 마찬가지로 영역 모드와 레거시 모드를 모두 지원합니다. 하지만 레거시 모드에서 실행하도록 Access Manager를 구성한 경우에는 릴리스 2 및 릴리스 3에 배포된 레거시 전용 콘솔을 사용할 수 있습니다.

Access Manager 종속성

다른 Java ES 구성 요소에 대한 Access Manager 종속성은 Access Manager 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어, Access Manager 인터페이스 또는 기능의 변경 사항을 적용하려면 Access Manager이 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Access Manager는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Access Manager는 특정 Java ES 공유 구성 요소에 대해 종속성이 있습니다([47페이지 표 1-9](#) 참조). Access Manager 업그레이드는 이러한 공유 구성 요소의 업그레이드된 버전에 종속됩니다.
- **웹 컨테이너.** Access Manager에는 Java ES Web Server, Java ES Application Server 또는 타사 웹 컨테이너(Weblogic 및 WebSphere)에서 제공하는 웹 컨테이너 서비스가 반드시 필요합니다. Access Manager를 업그레이드하려면 Access Manager 콘솔 또는 업그레이드한 Access Manager 환경에 맞게 마이그레이션한 인증 UI에 대해 JSP를 사용자 정의해야 합니다.
- **Directory Server.** Access Manager에는 구성 데이터 및 사용자 데이터를 저장하는 데 사용할 Directory Server가 반드시 필요합니다. 따라서, Access Manager를 업그레이드하려면 디렉토리 스키마를 확장해야 합니다.

웹 컨테이너 업그레이드 시나리오

Access Manager는 Web Server 또는 Application Server에서 제공하는 웹 컨테이너에 배포할 수 있습니다. 따라서 배포되는 웹 컨테이너를 릴리스 5로 업그레이드해야 할 수 있기 때문에 릴리스 5로 Access Manager를 업그레이드하는 작업은 복잡해질 수 있습니다. 이와 관련하여 가능한 여러 가지 웹 컨테이너 업그레이드 시나리오가 다음 표에 정리되어 있습니다.

표 14-4 Access Manager 업그레이드를 위한 웹 컨테이너 업그레이드 시나리오

시나리오	원래 Access Manager가 배포된 웹 컨테이너	업그레이드 후 Access Manager를 배포할 웹 컨테이너	적용 가능 Access Manager 업그레이드 경로: 업그레이드 원본
시나리오 1	Web Server 6.x	Web Server 6.x	릴리스 2 릴리스 3 릴리스 4
시나리오 2	Web Server 6.x	Web Server 7.0	릴리스 2 릴리스 3 릴리스 4
시나리오 3	Application Server 8.1	Application Server 8.1	릴리스 3 릴리스 4
시나리오 4	Application Server 8.1	Application Server 8.2	릴리스 3 릴리스 4
시나리오 5	Application Server 7x	Application Server 8.2	릴리스 2

Access Manager를 업그레이드할 때(예를 들어 amconfig 스크립트를 사용하는 경우) 특히, 웹 컨테이너에 대한 주요 버전 업그레이드를 수행하는 경우 해당하는 표 14-4의 업그레이드 시나리오에 대해 적절한 값을 제공해야 합니다.

이중 업그레이드

Access Manager와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 [43페이지의 "이중 업그레이드: Java ES 및 운영 체제 소프트웨어"](#)에서 설명한 대로 Access Manager에서 지원되지 않습니다.

따라서 이중 업그레이드를 수행해야 하는 경우 운영 체제를 설치하거나 업그레이드한 다음 Access Manager를 다시 설치하고 새로 구성해야 합니다.

Java ES 릴리스 4에서 Access Manager 업그레이드

이 절에는 Access Manager를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [전체 릴리스 4 Access Manager 업그레이드](#)
- [여러 인스턴스 업그레이드](#)
- [릴리스 4 Access Manager SDK 전용 업그레이드](#)

소개

Java ES 릴리스 4 Access Manager를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 이전 버전의 바이너리를 제거하고 릴리스 5를 새로 설치하여 업그레이드합니다. ampre71upgrade 스크립트를 사용하여 릴리스 4 버전을 제거하고 Java ES 설치 프로그램을 사용하여 릴리스 5를 설치합니다. 그런 다음 amconfig 스크립트를 사용하여 Access Manager를 재구성하고 amupgrade 스크립트를 사용하여 디렉토리 스키마를 마이그레이션합니다.
- **업그레이드 종속성.** Access Manager은 여러 Java ES 공유 구성 요소([47페이지 표 1-9](#) 참조)에 종속되며, 이러한 구성 요소는 Access Manager을 업그레이드할 때 Java ES 설치 프로그램이 릴리스 5로 자동으로 업그레이드합니다. 여기에는 릴리스 4 Access Manager에는 필요하지 않았던 여러 구성 요소가 필요한 새로운 Java ES 모니터링 프레임워크에 대한 지원이 포함됩니다.

또한 릴리스 5 Access Manager는 [260페이지의 "Access Manager 종속성"](#)에 설명한 대로 Directory Server 및 Web Server(또는 Application Server 또는 타사 웹 컨테이너)에 종속됩니다. 그러나 이는 엄격하지 않은 업그레이드 종속성입니다. 따라서 Access Manager를 릴리스 5로 업그레이드하는 경우 이러한 구성 요소는 선택적으로 업그레이드할 수 있습니다.

- **역방향 호환성.** 릴리스 5 Access Manager는 릴리스 4와 호환되지만 그보다 이전 Access Manager 릴리스와는 호환되지 않습니다([259페이지의 "호환성 문제"](#) 참조).

- **업그레이드 롤백.** Access Manager 업그레이드 롤백을 위한 유틸리티는 없습니다. 사실 Access Manager를 원래 상태로 롤백하는 데는 많은 수의 재구성이 필요하기 때문에 현실적이지 않습니다. 롤백을 수행하는 가장 좋은 방법은 백업된 구성 파일을 사용하여 병렬 설치를 수행하고 업그레이드를 수행하기 전에 이 병렬 설치를 테스트하는 것입니다. 이렇게 하면 필요한 경우 병렬 설치로 되돌릴 수 있습니다.
- **플랫폼 문제.** Access Manager를 업그레이드하기 위한 일반적인 방법은 Solaris 및 Linux 운영 체제에도 적용됩니다. 다음 절차에 플랫폼별 명령과 적절한 파일 위치가 표시되어 있습니다.

전체 릴리스 4 Access Manager 업그레이드

이 절에서는 전체 Access Manager를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법을 설명합니다.

- 사전 업그레이드 작업
- 릴리스 4 업그레이드 Access Manager
- Access Manager 업그레이드 확인
- 사후 업그레이드 작업
- 업그레이드 롤백

사전 업그레이드 작업

Access Manager를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- 현재 버전 정보 확인
- Access Manager 종속성 업그레이드
- Directory Server 데이터 백업
- 릴리스 4 Access Manager 구성 정보 백업
- 웹 컨테이너 사용자 정의 파일 백업
- 릴리스 4 Access Manager 로그 및 디버그 파일 백업
- 사용자 정의 현지화 파일 백업
- 필수 구성 정보 및 비밀번호 얻기

현재 버전 정보 확인

다음 명령을 사용하여 Access Manager의 현재 버전을 확인할 수 있습니다.

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

표 14-5 Access Manager 버전 확인 출력

Java ES 릴리스	Access Manager 버전 번호
릴리스 2	6.2
릴리스 3	6 2005Q1
릴리스 4	7 2005Q4
릴리스 5	7.1

Access Manager 종속성 업그레이드

일반적으로 컴퓨터 시스템(및 컴퓨팅 환경 내)의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다. Access Manager는 여러 공유 구성 요소에 대한 엄격한 업그레이드 종속성이 있습니다([47페이지 표 1-9](#) 참조).

Access Manager 제품 구성 요소 종속성을 업그레이드하는 경우 Access Manager를 업그레이드하기 전에 다음 순서 대로 업그레이드해야 하며, 이미 업그레이드한 항목은 건너뛸 수 있습니다. 공유 구성 요소 업그레이드는 일반적으로 Java ES 설치 프로그램을 통해 자동으로 수행됩니다.

1. **공유 구성 요소.** [63페이지의 "Java ES 공유 구성 요소 업그레이드"](#)에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 Access Manager에 필요한 모든 공유 구성 요소는 Access Manager를 릴리스 5로 업그레이드할 때 Java ES 설치 프로그램이 자동으로 업그레이드합니다.
2. **Directory Server(엄격하지 않은 업그레이드 종속성).** [99페이지의 5장, "Directory Server"](#)에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **웹 컨테이너 소프트웨어(엄격하지 않은 업그레이드 종속성).** Web Server 또는 Application Server의 업그레이드 지침은 [129페이지의 7장, "Web Server"](#) 및 [199페이지의 11장, "Application Server"](#)에서 각각 제공됩니다.

Access Manager를 업그레이드하기 전에 웹 컨테이너 소프트웨어를 업그레이드하지 않은 경우 업그레이드 절차(amconfig 스크립트 사용)를 수행하면 Access Manager가 기존 웹 컨테이너에 구성되고 재배포됩니다.

Directory Server 데이터 백업

Access Manager 업그레이드 프로세스는 Directory Server 스키마를 수정하는 스크립트를 사용합니다. 따라서 Access Manager를 업그레이드하기 전에 Directory Server 콘솔 또는 db2bak와 같은 명령줄 유틸리티를 사용하여 Directory Server 데이터를 백업하십시오. db2ldif를 사용하면 Access Manager 스키마 및 디렉토리 정보 트리(DIT)를 백업할 수 있습니다.

Directory Server 백업에 대한 자세한 내용은 Sun Java System Directory Server Enterprise Edition 6.0 관리 설명서(<http://docs.sun.com/doc/820-0292>)를 참조하십시오.

릴리스 4 Access Manager 구성 정보 백업

릴리스 5 Access Manager 소프트웨어를 재구성하려면 릴리스 4 버전을 재구성해야 하므로 구성 파일을 잘 알고 있는 위치에 백업하는 것이 중요합니다. 다음 파일을 백업해야 합니다.

- AMConfig.properties 파일
AccessManagerConfig-base/config/AMConfig.properties
- serverconfig.xml 파일
AccessManagerConfig-base/config/serverconfig.xml
- 웹 컨테이너 구성 파일:
 - Web Server의 경우: 258페이지 표 14-3에서 server.policy 및 server.xml 파일의 위치를 참조하십시오.
 - Application Server의 경우: 258페이지 표 14-3에서 server.policy 및 domain.xml 파일의 위치를 참조하십시오.
 - 타사 웹 컨테이너의 경우, 적절한 구성 파일
- 인증 및 사용자 정의 모듈용 JAR 파일
AccessManager-base/lib

웹 컨테이너 사용자 정의 파일 백업

Access Manager가 참조하는 웹 컨테이너 사용자 정의 파일이 있는 경우 해당 파일을 백업해야 합니다. 다음과 같은 사용자 정의 파일이 포함됩니다.

- 사용자 정의된 Access Manager 콘솔 JSP 페이지:
 - 영역/레거시 콘솔(Java ES 릴리스 4와 함께 배포)
AccessManager-base/web-src/services/
 - 레거시 전용 콘솔(Java ES 릴리스 2 및 3과 함께 배포)
AccessManager-base/web-src/applications/
- 사용자 정의된 인증 UI JSP 페이지
AccessManager-base/web-src/services/
- 사용자 정의된 XML 파일
AccessManagerConfig-base/config/xml/

팁

Access Manager를 업그레이드한 후 백업된 코드를 사용하여 다시 적용할 수 있도록 사용자 정의 파일을 기록해 두십시오.

릴리스 4 Access Manager 로그 및 디버그 파일 백업

시스템 상태 정보 분석을 위하여 로그 및 디버그 파일이 손실되지 않도록 백업하는 것이 좋습니다. 이러한 파일은 다음 위치에 있습니다.

- 디버그 파일
/var/AccessManager-base/debug
- 로그 파일
/var/AccessManager-base/logs

사용자 정의 현지화 파일 백업

Java ES 설치 프로그램이 설치한 현지화된 파일을 사용자 정의했거나 Java ES 설치 프로그램이 설치하지 않은 현지화 언어를 새로 추가한 경우 이러한 사용자 정의를 백업해야 합니다. 사용자 정의에는 다음과 같은 내용이 포함됩니다.

- 사용자 정의된 Access Manager 사용자 인터페이스 현지화
*AccessManager-base/locale/*_Locale.properties*
- 사용자 정의된 인증 UI JSP 페이지
AccessManager-base/web-src/services/config/auth/default_Locale
- 사용자 정의된 온라인 도움말 번역
AccessManager-base/web-src/services/html/Locale

필수 구성 정보 및 비밀번호 얻기

Access Manager를 업그레이드하려면 다음과 같은 특정 구성 정보를 제공해야 합니다.

- Access Manager 관리자 아이디와 비밀번호
- LDAP 사용자 아이디와 비밀번호
- Access Manager가 사용하는 Directory Server 인스턴스를 위한 디렉토리 관리자 이름과 비밀번호

릴리스 4 업그레이드 Access Manager

Access Manager 소프트웨어를 Java ES 릴리스 5로 업그레이드하는 절차에는 Access Manager를 재구성하고 Access Manager 데이터를 마이그레이션하는 절차가 포함되어 있습니다.

업그레이드 요약

Access Manager 업그레이드 절차는 다음 단계로 구성됩니다.

1. [Access Manager Mobile Access 소프트웨어를 업그레이드합니다.](#)
2. [Access Manager의 Java ES 릴리스 4 버전을 제거합니다.](#) ampre71upgrade 스크립트를 사용합니다.
3. [릴리스 5 업그레이드를 현지화해야 하는 경우 릴리스 4 현지화 패키지를 제거합니다.](#) 이 단계는 수동으로 수행해야 합니다.
4. [Access Manager의 Java ES 릴리스 5 버전을 설치합니다.](#) Java ES 설치 프로그램에서 나중에 구성 옵션을 사용합니다.
5. [Access Manager용 JSP를 다시 사용자 정의합니다.](#)
6. [Access Manager를 배포 해제하고 웹 컨테이너로 재구성 및 재배포합니다.](#) amconfig 스크립트를 사용합니다.
7. [디렉토리 구조와 스키마를 업데이트합니다.](#) amupgrade 스크립트를 사용합니다.

이러한 단계는 다음 절차에 각각 나와 있습니다.

업그레이드 절차

1. Access Manager Mobile Access 소프트웨어를 업그레이드합니다.

릴리스 4 버전에 패치를 사용하여 Access Manager Mobile Access 소프트웨어를 업그레이드해야 합니다. 다음 표에 필요한 패치가 나와 있습니다.

표 14-6 Access Manager Mobile Access 소프트웨어를 업그레이드하기 위한 패치¹

설명	패치 아이디: Solaris 9 및 10	패치 아이디: Linux
Mobile Access 소프트웨어	119530-05(SPARC) 119531-05(x86)	119532-05 - sun-identity-mobileaccess-6.2-25.3.i386.rpm - sun-identity-mobileaccess-config-6.2-25.3.i386.rpm

1. 패치 버전 번호는 Java ES 릴리스 5로 업그레이드하는 데 필요한 최소 버전입니다. 보다 최신 버전이 있으면 표에 표시된 버전 대신 최신 버전을 사용하십시오.

- a. 표 14-6의 패치 번호를 사용하여 필요한 패치를 구합니다.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

- b. 패치의 README 파일에 나와 있는 사전 패치 절차를 수행합니다.
- c. 패치에서 필요한 다음 매개 변수의 값을 구합니다.

표 14-7 Mobile Access 패치 매개 변수

매개 변수	값
디렉토리 관리자 DN	기본값: cn=Directory Manager
디렉토리 관리자 비밀번호	

- d. 표 14-6에 있는 패치를 적용합니다.

Solaris의 경우
`patchadd /tmp/patch_ID`

Linux의 경우
`./update`

패치의 README 파일에 나와 있는 사후 패치 절차를 수행합니다.

2. Access Manager의 Java ES 릴리스 4 버전을 제거합니다.

- a. 릴리스 4 Access Manager를 호스팅하는 컴퓨터에 루트로 로그인하거나 슈퍼유저가 됩니다.

`su -`

- b. 디렉토리를 Java ES 릴리스 5 배포의 `os_arch/Product/identity_svr/Tools`로 변경합니다. 여기서 `os_arch`는 `Solaris_sparc`과 같은 사용자의 플랫폼에 해당합니다.
- c. `ampre71upgrade` 스크립트에 필요한 다음 매개 변수의 값을 구합니다.

표 14-8 Access Manager 구성 매개 변수: `ampre71upgrade`

매개 변수	값
Directory Server 호스트	정규화된 이름 설정: <code>hostname.domain</code>
Directory Server 포트	SSL이 아닌 포트 번호 ¹ 를 지정합니다. 기본값: 389
최상위 관리자 DN	기본값: <code>uid=amadmin,ou=People,default_org_DN</code>
최상위 관리자 비밀번호	
백업 파일을 저장할 디렉토리	기본값: <code>AccessManager-base</code>

1. 636과 같은 기본 SSL 값을 갖는 Directory Server SSL 포트를 지정하면 사전 업그레이드 프로세스가 성공적으로 완료되지 않습니다.

- d. Directory Server가 실행되고 있는지 확인하십시오. 그렇지 않은 경우 Directory Server를 시작하십시오.
- e. `ampre71upgrade` 스크립트를 실행합니다.

```
./ampre71upgrade
```

스크립트는 Access Manager 구성 파일을 백업하고 릴리스 4 기본 패키지를 제거합니다. 현지화된 패키지는 아래 [단계 3](#)에 따라 수동으로 제거해야 합니다.

- 3. 릴리스 5 업그레이드를 현지화해야 하는 경우 릴리스 4 현지화 패키지를 제거합니다.

[단계 2](#)에서 실행되는 `ampre71upgrade` 스크립트는 현지화 패키지를 제거하지 않으므로 다음과 같이 수동으로 제거해야 합니다.

Solaris의 경우

- a. 현지화 패키지를 확인합니다.

```
pkginfo | grep SUNWaml
pkginfo | grep SUNWamclnt
pkginfo | grep SUNWamdistauth
```

- b. 단계 a에서 찾은 현지화 패키지를 제거합니다.

```
pkgrm SUNWamlLocale
pkgrm SUNWamclntLocale
pkgrm SUNWamdistauthLocale
```

Linux의 경우

- a. 현지화 RPM을 확인합니다.

```
rpm -qa | grep sun-identity-sdk-*
rpm -qa | grep sun-identity-clientsdk-*
rpm -qa | grep sun-identity-distauth-*
```

- b. 단계 a에서 찾은 현지화 RPM을 제거합니다.

```
rpm -e sun-identity-sdk-Locale-*
rpm -e sun-identity-clientsdk-Locale-*
rpm -e sun-identity-distauth-Locale-*
```

4. Access Manager의 Java ES 릴리스 5 버전을 설치합니다.

다음 단계를 수행합니다.

- a. 릴리스 4 Access Manager를 호스팅하는 컴퓨터에서 Java ES 설치 프로그램을 실행합니다.

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

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다.(명령 줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.)

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다.(설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.)

- b. 구성 요소 선택 페이지에서 Access Manager를 선택합니다.
- c. 릴리스 4가 설치된 디렉토리 and 동일한 설치 디렉토리를 지정합니다.
- d. "나중에 구성" 옵션을 선택합니다.
- e. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.
- f. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.

5. Access Manager용 JSP를 다시 사용자 정의합니다.

266페이지의 "웹 컨테이너 사용자 정의 파일 백업"에 저장한 Access Manager 콘솔 및 인증 사용자 인터페이스(UI)를 위한 JSP에 릴리스 4 사용자 정의를 다시 적용합니다.

그 다음 사용자 정의 JSP 파일을 올바른 디렉토리에 복사합니다.

- 영역/레거시 Access Manager 콘솔
AccessManager-base/web-src/services/console
- 레거시 전용 Access Manager 콘솔
AccessManager-base/web-src/applications/console
- 인증 UI:
AccessManager-base/web-src/services/config/auth/default 또는
AccessManager-base/web-src/services/config/auth/default_Locale
(여기서 *Locale*은 ja와 같은 로캘 표시기입니다.)

자세한 내용은 Sun Java System Access Manager 7.1 Developer's Guide (<http://docs.sun.com/doc/819-4675>)를 참조하십시오.

6. Access Manager를 배포 해제하고 웹 컨테이너로 재구성 및 재배포합니다.

amconfig 스크립트를 실행하여 특정 웹 컨테이너를 위한 Access Manager를 구성합니다. amconfig 스크립트(및 연관된 amsamplesilent 템플릿 입력 파일)는 다음 디렉토리에 상주합니다.

AccessManager-base/bin

amconfig 스크립트 및 amsamplesilent 템플릿 파일에 대한 자세한 내용은 Sun Java System Access Manager 7.1 관리 설명서 (<http://docs.sun.com/doc/820-0841>)를 참조하십시오.

Access Manager를 웹 컨테이너에 재구성하고 재배포하려면 다음 단계를 수행하십시오.

- a. 웹 컨테이너 소프트웨어를 업그레이드하기로 선택하면 264페이지의 "Access Manager 종속성 업그레이드"에 설명한 대로 업그레이드를 완료해야 합니다.
- b. 웹 컨테이너의 관리 인스턴스가 실행되고 있는지, 아래 표에 나와 있는 대로 amconfig 스크립트에서 지원하는 모드인지 확인합니다.

표 14-9 amconfig에서 지원하는 관리 서버 모드

웹 컨테이너	지원되는 모드	기본 포트 번호
Application Server (8.x): Java ES 릴리스 3, 4 및 5	SSL(보안) 비SSL	4849
Web Server (7.0): Java ES 릴리스 5	SSL(보안)	8989
Web Server (6.x): Java ES 릴리스 2, 3 및 4	비SSL	8888

- c. 웹 컨테이너가 SSL 모드로 실행 중인 경우 컨테이너의 SSL 인증서가 만료되지 않았고 아직 유효한지 확인합니다.
- d. Access Manager를 릴리스 5 Web Server에 배포하는 경우 Access Manager와 같은 인스턴스에서 실행되는 Access Manager에 종속된 모든 Java ES 구성 요소를 비활성화합니다.

이러한 구성 요소에는 Portal Server와 같은 구성 요소 또는 Communications Express, Instant Messaging, Delegated Administrator와 같은 Sun Java Communications Suite 구성 요소가 있습니다.

다음 절차를 따르십시오.

- I. <https://host:8989>에 admin으로 로그인합니다.
- II. "가상 서버 편집"으로 이동합니다.
- III. "웹 응용 프로그램" 탭을 선택합니다.
- IV. 모든 Access Manager 종속 응용 프로그램을 선택합니다.
- V. "비활성화"를 누릅니다.
- VI. "저장"을 누릅니다.
- VII. "배포 보류 | 배포 구성"을 누릅니다.

구성 변경 사항이 Web Server 인스턴스에 적용됩니다.

- e. Directory Server와 해당 웹 컨테이너가 실행되고 있는지 확인합니다.

- f. `amsamplesilent` 템플릿 입력 파일을 기반으로 `amconfig` 입력 파일을 생성합니다.

```
cp amsamplesilent config-file
```

후속 단계에서 `config-file`은 `amsamplesilent`와 같은 디렉토리에 있다고 가정합니다.

- g. `config-file`에 구성 매개 변수를 설정합니다.

모든 매개 변수는 올바르게 설정되어야 합니다. 일부 값은 `AMConfig.properties` 파일에서 마이그레이션할 수 있으며 일부 값은 다음 표에 표시된 것처럼 업그레이드 절차에 따라 달라집니다.

표 14-10 Access Manager 구성 매개 변수: `amconfig`

매개 변수	값
업그레이드 매개 변수	
<code>DEPLOY_LEVEL</code>	배포 해제: 경우 26으로 설정합니다. 재구성 및 배포: 경우 1로 설정합니다.
<code>DIRECTORY_MODE</code>	5로 설정합니다.
<code>AM_REALM</code> ¹	레거시 모드를 사용하는 경우 <code>disabled</code> 로 설정합니다. 영역 모드를 사용하는 경우 <code>enabled</code> 로 설정합니다. 기본값: <code>enabled</code>
<code>JAVA_HOME</code>	JDK 릴리스 5 디렉토리로 설정: <code>/usr/java/jdk1.5.0_04/</code>
<code>WEB_CONTAINER</code>	Web Server 7.x의 경우 <code>WS</code> 로 설정합니다. Web Server 6.x의 경우 <code>WS6</code> 으로 설정합니다. Application Server 8.x의 경우 <code>AS8</code> 로 설정합니다. IBM WebSphere 5.x의 경우 <code>WAS5</code> 로 설정합니다. BEA WebLogic 8.x의 경우 <code>WL8</code> 로 설정합니다. 그리고 <code>config-file</code> 의 해당 섹션에만 값을 입력합니다.
<code>WS_INSTANCE</code> (Web Server 7.x를 웹 컨테이너로 사용하는 경우)	대/소문자를 구분하는 인스턴스 구성 디렉토리 이름으로 설정: <code>https-configName/</code> 디렉토리 경로: <code>WebServer7Config-base/https-configName/</code>
<code>WS61_INSTANCE</code> (Web Server 6.x를 웹 컨테이너로 사용하는 경우)	대소문자를 구분하는 인스턴스 구성 디렉토리 이름으로 설정: <code>https-instanceName</code> 디렉토리 경로: <code>WebServer6-base/https-instanceName/</code>
<code>AS81_INSTANCE</code> (Application Server 8.x를 웹 컨테이너로 사용하는 경우)	Application Server 8.x <code>instanceName</code> 으로 설정합니다. 기본값: <code>server</code>

표 14-10 Access Manager 구성 매개 변수: amconfig (계속)

매개 변수	값
AS81_INSTANCE_DIR (Application Server 8.x를 웹 컨테이너로 사용하는 경우)	인스턴스의 Application Server 8.x 도메인 디렉토리로 설정합니다. 기본값은 다음과 같습니다. <i>AppServer8Config-base/domains/domain1</i>
AS81_DOCS_DIR (Application Server 8.x를 웹 컨테이너로 사용하는 경우)	인스턴스의 Application Server 8.x docroot 디렉토리로 설정합니다. 기본값은 다음과 같습니다. <i>AppServer8Config-base/domains/domain1/docroot</i>

AMConfig.properties에서 마이그레이션

SERVER_PROTOCOL	com.ipplanet.am.server.protocol
SERVER_PORT	com.ipplanet.am.server.port
SERVER_HOST	com.ipplanet.am.server.host
DS_HOST	com.ipplanet.am.directory.host
DS_PORT	com.ipplanet.am.directory.port
ROOT_SUFFIX ²	com.ipplanet.am.defaultOrg
CONSOLE_DEPLOY_URI	com.ipplanet.am.console.deploymentDescriptor
SERVER_DEPLOY_URI	com.ipplanet.am.services.deploymentDescriptor
PASSWORD_DEPLOY_URI	com.sun.identity.password.deploymentDescriptor
AM_ENC_PWD ²	am.encryption.pwd ³

1. 영역 및 레거시 모드에 대한 자세한 내용은 [259페이지의 "호환성 문제"](#)를 참조하십시오.
2. 이 매개 변수의 값은 이전 버전의 Access Manager와 같아야 합니다.
3. Access Manager 및 Access Manager SDK를 모두 배포한 경우 이 등록 정보 값은 Access Manager 인스턴스 및 해당 Access Manager SDK 인스턴스에서 같아야 합니다.

다른 매개 변수는 웹 컨테이너나 비밀번호를 변경한 경우가 아니면 릴리스 4 구성에 사용된 것과 동일한 값을 제공하십시오. 예를 들어 Web Server를 릴리스 5로 업그레이드한 경우 다음 값을 제공합니다.

표 14-11 amconfig 매개 변수: 릴리스 5 Web Server

매개 변수	값
WS_CONFIG	Web Server 구성의 이름: <i>configName</i>
WS_INSTANCE	<i>https-configName</i>
WS_HOME	<i>WebServer7Config-base</i>
WS_PROTOCOL	http 또는 https
WS_HOST	Web Server 인스턴스가 연결을 수신하는 정규화된 호스트 이름
WS_PORT	Web Server 인스턴스가 연결을 수신하는 포트
WS_ADMINPORT	Web Server 관리 인스턴스가 연결을 수신하는 포트
WS_ADMIN	Web Server 관리자 아이디
WS_ADMINPASSWD	Web Server 관리자 비밀번호

- h. amconfig를 실행하여 Access Manager를 배포 해제합니다.

*config-file*에 있는 DEPLOY_LEVEL의 값을 26로 설정합니다.

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

- i. 공용 에이전트 컨테이너가 실행되고 있는지 확인합니다.

```
netstat -an | grep 11163
```

실행되고 있지 않으면 시작합니다.

```
/usr/sbin/cacaoadm start
```

- j. amconfig를 실행하여 Access Manager를 재구성하고 웹 컨테이너로 배포합니다.

*config-file*에 있는 DEPLOY_LEVEL의 값을 1로 설정합니다.

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

7. 디렉토리 구조와 스키마를 업데이트합니다.

릴리스 5 Access Manager는 릴리스 4 디렉토리 구조와 함께 사용할 수 있지만 구조를 릴리스 5 기능을 지원하도록 수정해야 합니다. 다음 디렉토리에 설치되어 있는 amupgrade 스크립트를 실행하여 Access Manager 디렉토리 구조와 스키마를 릴리스 5로 업데이트합니다.

- o *Solaris의 경우*
AccessManager-base/upgrade/scripts
- o *Linux의 경우*
AccessManager_base/identity/upgrade/scripts
- a. amupgrade 스크립트에 필요한 다음 매개 변수의 값을 구합니다.

표 14-12 Access Manager 구성 매개 변수: amupgrade

매개 변수	값
Directory Server 정규화된 도메인 이름	정규화된 이름 설정: <i>hostname.domain</i>
Directory Server 포트	SSL이 아닌 포트 번호 ¹ 를 지정합니다. 기본값: 389
디렉토리 관리자 DN	기본값: cn=Directory Manager
디렉토리 관리자 비밀번호	
최상위 관리자 DN	기본값: uid=amadmin,ou=People,default_org_DN
최상위 관리자 비밀번호	
영역 모드 사용 (릴리스 4 영역 모드에서 업그레이드하는 경우 이 매개 변수 값은 필요하지 않습니다.)	y/n: 예는 영역 모드를 사용할 수 있고 서비스 데이터가 새 영역 트리로 마이그레이션되었다는 의미입니다. ² 아니요(기본값)는 서비스 데이터가 레거시 모드로 남아있다는 의미입니다.

1. 기본 SSL 값인 636과 다른 Directory Server SSL 포트를 지정해야 합니다.

2. 278페이지의 "영역 모드로 마이그레이션"을 참조하십시오.

- b. amupgrade 스크립트를 실행합니다.

```
cd AccessManager-base/upgrade/scripts
./amupgrade
```

성공적으로 업그레이드되면 스크립트는 "Upgrade completed"라고 표시합니다.

- c. 디렉토리 스키마 확장에 대한 자세한 내용은 다음 업그레이드 로그 파일을 확인하십시오.

Solaris의 경우

`/var/sadm/install/logs/`

`Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm`

Linux의 경우

`/var/log/Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm`

8. [272페이지의 단계 d](#)에서 비활성화했던 모든 구성 요소를 활성화합니다.

9. Access Manager가 배포되어 있는 웹 컨테이너를 다시 시작합니다.

10. Access Manager를 시작합니다.

Access Manager가 배포되어 있는 웹 컨테이너를 다시 시작합니다.

Access Manager 업그레이드 확인

업그레이드 절차를 마친 뒤 성공적으로 업그레이드되었는지 다음과 같이 확인합니다.

1. 다음 명령을 사용하여 Access Manager 패키지의 업그레이드를 확인합니다.

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

출력 값에 대해서는 [264페이지 표 14-5](#)를 참조하십시오.

2. `/var/sadm/install/logs` 디렉토리에서 다음 설치 프로그램 로그 파일을 확인하여 업그레이드 상태를 검토합니다.

- o `Java_Shared_Component_Install.timestamp`
- o `Java_Enterprise_System_install.Atimestamp`
- o `Java_Enterprise_System_install.Btimestamp`
- o `Java_Enterprise_System_Summary_Report_install.timestamp`

3. amupgrade 스크립트를 실행하는 동안 터미널 창에서 오류가 발생했는지 확인하여 Access Manager 마이그레이션 상태를 검토합니다.

또한 `/var/sadm/install/logs` 디렉토리에서 다음 로그 파일을 확인합니다.

`Sun_Java_System_Access_Manager_upgrade_dit_log.timestamp`

4. 오류에 대해 Access Manager 문제 해결 파일을 검토합니다.

이 파일은 AMConfig.properties 파일의 com.ipplanet.services.debug.directory 등록 정보에 지정된 위치에 있으며 기본값은 다음과 같습니다.

Solaris의 경우

/var/opt/SUNWam/debug

Linux의 경우

/var/opt/sun/identity/debug

사후 업그레이드 작업

다음 상황을 해결하려면 사후 업그레이드 절차가 필요합니다.

- [영역 모드로 마이그레이션](#)
- [SAML\(Security Assertion Markup Language\)](#)

영역 모드로 마이그레이션

Access Manager를 릴리스 5로 업그레이드할 때 영역 모드로 마이그레이션하는 경우 (즉 amupgrade를 사용하여 디렉토리 구조 및 스키마를 업그레이드할 때 영역 모드를 사용하도록 예를 선택한 경우) 다음 단계를 수행합니다.

1. *AccessManagerConfig-base/config/AMConfig.properties* 파일을 엽니다.
2. 다음 등록 정보의 값을 확인합니다.
com.sun.identity.sm.ldap.enableProxy
3. 등록 정보가 false로 설정되어 있지 않으면 수동으로 false로 설정합니다.

SAML(Security Assertion Markup Language)

Security Assertion Markup Language(SAML) 서비스를 사용하는 경우 Access Manager 콘솔을 통해 SAM 인증 모듈을 추가하여 사용할 수 있도록 해야 합니다. SAML 인증 모듈 인스턴스 생성에 대한 자세한 내용은 Sun Java System Access Manager 7.1 관리 설명서(<http://docs.sun.com/doc/820-0841>)를 참조하십시오.

업그레이드 롤백

Access Manager를 업그레이드 전 상태로 롤백할 수 있는 스크립트는 없습니다. 이 프로세스는 사전 업그레이드 작업의 일부분으로 백업해 두었던 Access Manager 데이터를 사용하여 직접 수행해야 합니다([266페이지의 "릴리스 4 Access Manager 로그 및 디버그 파일 백업"](#) 참조). 업그레이드를 되돌리는 것은 까다롭기 때문에 그다지 실용적이지 않습니다.

롤백하는 한 가지 방법은 릴리스 4를 다시 설치하고 백업한 구성 파일을 적합한 위치로 마이그레이션하는 것입니다. 또 다른 방법은 업그레이드 전에 백업한 구성 파일을 사용하여 병렬 시스템을 만들고 업그레이드를 시도하기 전에 이 병렬 시스템을 테스트하는 것입니다.

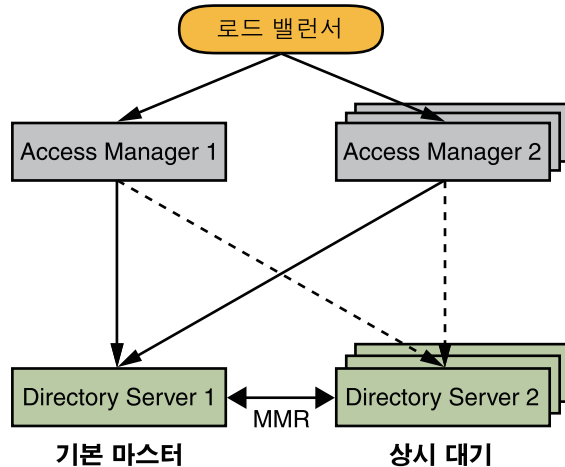
여러 인스턴스 업그레이드

일부 배포 구조에서 Access Manager는 여러 컴퓨터 시스템에 배포되어고가용성과 확장성을 제공합니다.

일반적으로 서비스를 중단하지 않고 Access Manager 인스턴스를 업그레이드하는 것이 좋습니다. 이 섹션에서는 릴리스 4 Access Manager에서 릴리스 5로 이러한 롤링 업그레이드를 수행하는 절차에 대해 설명합니다.

주	동일한 호스트 시스템에 설치된 Access Manager의 여러 인스턴스 업그레이드는 현재 릴리스에서는 지원되지 않습니다. 동일한 호스트에 여러 인스턴스가 있는 경우 주 인스턴스를 업그레이드한 후 추가 인스턴스를 다시 생성해야 합니다.

다음 그림의 배포 아키텍처는 롤링 업그레이드 절차에 대해 설명하는 데 사용됩니다.

그림 14-1 여러 Access Manager 인스턴스를 위한 배포 구조 예

이 구조에서는 로드 밸런서를 통해 여러 Access Manager 인스턴스에 액세스하며 이러한 인스턴스는 다시 다중 마스터 복제 (MMR)를 위해 설정된 디렉토리에 액세스합니다. 다른 Directory Server 복제 방식도 가능하지만 MMR은 가용성이 뛰어나고 확장성이 우수한 대표적인 디렉토리 서비스입니다. [그림 14-1](#)에서는 업그레이드 절차에 대한 설명을 용이하게 진행할 수 있도록 여러 Access Manager 및 Directory Server 인스턴스를 그룹화하였습니다. 예를 들어 Access Manager 2는 n번째 Access Manager 인스턴스에서 두 번째를 나타냅니다.

릴리스 4 Access Manager에서 릴리스 5로 롤링 업그레이드를 수행하는 절차는 디렉토리 스키마가 릴리스 5로 업데이트되지 않은 경우 릴리스 5 Access Manager 및 릴리스 4 Access Manager 인스턴스가 동일한 디렉토리에 공존하고 동시에 실행될 수 있는 상호 운용성에 바탕을 두고 있습니다.

따라서 단일 Directory Server 인스턴스를 가리키는 Access Manager 인스턴스의 경우 모든 Access Manager 인스턴스가 업그레이드될 때까지 디렉토리 스키마 업데이트를 지연시켜 롤링 업그레이드를 수행할 수 있습니다.

다음 절차를 사용하여 릴리스 4 Access Manager에서 릴리스 5로 롤링 업그레이드를 수행할 수 있습니다.

1. 모든 Access Manager 인스턴스의 릴리스 4 구성 정보를 백업합니다.

[258페이지 표 14-3](#)을 참조하십시오.

2. Access Manager 1을 업그레이드합니다.
 - a. 로드 밸런서에서 Access Manager 1을 비활성화합니다.
요청이 더 이상 Access Manager 1로 라우팅되지 않습니다.
 - b. Access Manager 1을 부분적으로 업그레이드합니다.
디렉토리 구조 및 스키마 업데이트를 제외하고 [267페이지의 "릴리스 4 업그레이드 Access Manager"](#)에 있는 [276페이지의 단계 7](#)의 설명에 따라 Access Manager를 업그레이드합니다.
 - c. 로드 밸런서에서 Access Manager 1을 활성화합니다.
3. Access Manager 2부터 Access Manager n까지 업그레이드합니다.
설명을 간략하게 하기 위해 이후 단계에서 "Access Manager 2"는 Access Manager 2부터 Access Manager n까지를 의미합니다.
 - a. 로드 밸런서에서 Access Manager 2를 비활성화합니다.
요청이 더 이상 Access Manager 2로 라우팅되지 않습니다.
 - b. Access Manager 2를 부분적으로 업그레이드합니다.
디렉토리 구조 및 스키마 업데이트를 제외하고 [267페이지의 "릴리스 4 업그레이드 Access Manager"](#)에 있는 [276페이지의 단계 7](#)의 설명에 따라 각 Access Manager 인스턴스를 업그레이드합니다.
 - c. 로드 밸런서에서 Access Manager 2를 활성화합니다.
요청이 다시 Access Manager 2로 라우팅됩니다.
4. Directory Server 1의 디렉토리 구조와 스키마를 업데이트합니다.
[276페이지의 단계 7](#)에 설명된 대로 amupgrade 스크립트를 사용합니다. Access Manager 1부터 n은 Directory Server 1의 스키마가 업데이트되면 계속 작동합니다.

릴리스 4 Access Manager SDK 전용 업그레이드

일부 배포 구조의 경우 Access Manager SDK 구성 요소는 하나 이상의 컴퓨터 시스템에 다른 Access Manager 구성 요소를 설치하지 않은 상태로 설치됩니다. Access Manager SDK는 Access Manager에 대한 원격 인터페이스로 사용되며 Access Manager와 동일한 작업 모드(레거시 또는 영역)를 위해 재구성되어야 합니다.

Access Manager SDK 및 원격 인터페이스의 역할을 할 전체 버전의 Access Manager가 릴리스 5로 업그레이드되어야 합니다. 그러나 릴리스 5 Access Manager는 릴리스 4 Access Manager SDK의 이전 버전과 호환되지 않으므로 다른 컴퓨터에서 Access Manager SDK를 업그레이드하기 전에 Access Manager를 먼저 업그레이드하는 것이 좋습니다.

Access Manager에 대한 원격 인터페이스로서, SDK는 Directory Server에 액세스하도록 구성될 필요가 없습니다. Access Manager SDK를 사용하여 Portal Server와 같은 웹 컨테이너 서비스에 종속되는 웹 구성 요소를 지원하는 경우 Access Manager SDK는 해당 웹 컨테이너에 대해 구성되어야 합니다. 그러나 Access Manager SDK는 웹 구성 요소가 아닌 구성 요소도 지원할 수 있으며 웹 컨테이너는 필요하지 않습니다.

Access Manager SDK를 업그레이드하는 절차는 위의 특징을 기반으로 하는 전체 Access Manager 업그레이드 절차의 일부입니다.

이 절에서는 Access Manager SDK만 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다.

- [사전 업그레이드 작업](#)
- [릴리스 4 Access Manager SDK 업그레이드](#)
- [Access Manager SDK 업그레이드 확인](#)
- [업그레이드 롤백](#)

사전 업그레이드 작업

Access Manager SDK의 사전 업그레이드 작업은 전체 Access Manager 업그레이드 (263페이지의 "[사전 업그레이드 작업](#)" 참조) 작업과 동일하지만 Directory Server 및 Access Manager 관리 도구 JSP 사용자 정의와 관련된 작업은 포함되지 않습니다. Access Manager SDK에 필요한 사전 업그레이드 작업은 다음과 같습니다.

- [264페이지의 "Access Manager 종속성 업그레이드"](#)

그러나 Access Manager SDK의 경우 Directory Server에 대해 종속성이 없으며 Access Manager SDK가 웹 컨테이너에서 실행되는 경우에만 웹 컨테이너 소프트웨어에 대해 종속성이 있습니다.

- [265페이지의 "릴리스 4 Access Manager 구성 정보 백업"](#)

그러나 Access Manager SDK의 경우 Access Manager SDK가 웹 컨테이너에서 실행되는 경우에만 웹 컨테이너 구성 파일을 백업해야만 합니다.

- [266페이지의 "릴리스 4 Access Manager 로그 및 디버그 파일 백업"](#)

또한 이러한 파일에 액세스하기 위해 관리자 이름과 비밀번호를 얻어야 합니다.

릴리스 4 Access Manager SDK 업그레이드

Access Manager SDK의 업그레이드 절차는 전체 Access Manager 업그레이드 절차와 동일하지만 현지화, Access Manager 관리 도구 JSP 사용자 정의 및 디렉토리 스키마 마이그레이션과 관련된 절차는 포함되지 않습니다.

1. Access Manager SDK의 Java ES 릴리스 4 버전을 제거합니다.

Access Manager SDK 제거 작업만 제외하고 [268페이지의 "Access Manager의 Java ES 릴리스 4 버전을 제거합니다."](#)에 있는 지침을 따릅니다.

2. Access Manager SDK의 Java ES 릴리스 5 버전을 설치합니다.

Access Manager SDK 설치 작업만 제외하고 [270페이지의 "Access Manager의 Java ES 릴리스 5 버전을 설치합니다."](#)에 있는 지침을 따릅니다.

3. Access Manager SDK 재구성

DEPLOY_LEVEL 매개 변수 설정을 제외하고 [271페이지의 "Access Manager를 배포 해제하고 웹 컨테이너로 재구성 및 재배포합니다."](#)에 있는 지침을 다음과 같이 따릅니다.

- Access Manager SDK가 웹 컨테이너에 대해 구성된 경우:
DEPLOY_LEVEL=4 (SDK 업그레이드 및 웹 컨테이너 구성)
- Access Manager SDK가 웹 컨테이너에 대해 구성되지 않은 경우:
DEPLOY_LEVEL=3 (SDK만 업그레이드)

Access Manager SDK 업그레이드 확인

세 가지 방법을 사용하여 성공적인 Access Manager SDK 업그레이드를 확인할 수 있습니다.

- Portal Server 또는 Access Manager SDK를 사용하여 Access Manager와 상호 작용하는 다른 구성 요소를 실행하고 인증이 작동하는지 확인합니다.
- 다음 위치에서 제공되는 Access Manager SDK 예를 실행합니다.

AccessManager-base/samples/sdk

- AMConfig.properties 파일에 있는 com.ipplanet.am.version 등록 정보의 값을 확인합니다.

AccessManagerConfig-base/config/AMConfig.properties

업그레이드 롤백

Access Manager를 업그레이드 전 상태로 롤백할 수 있는 스크립트는 없습니다. 이 프로세스는 사전 업그레이드 작업의 일부분으로 백업해 두었던 Access Manager 데이터베이스를 사용하여 직접 수행해야 합니다([266 페이지의 "릴리스 4 Access Manager 로그 및 디버그 파일 백업"](#) 참조). 업그레이드를 되돌리는 것은 까다롭기 때문에 그다지 실용적이지 않습니다.

롤백하는 한 가지 방법은 R4를 다시 설치하고 백업한 구성 파일을 적합한 위치로 마이그레이션하는 것입니다. 또 다른 방법은 업그레이드 전에 백업한 구성 파일을 사용하여 병렬 시스템을 만들고 업그레이드를 시도하기 전에 이 병렬 시스템을 테스트하는 것입니다.

Java ES 릴리스 3에서 Access Manager 업그레이드

Java ES 2003Q1(릴리스 3) Access Manager 또는 Access Manager SDK를 릴리스 5로 업그레이드하는 절차는 여러 인스턴스 업그레이드를 수행하는 방법을 제외하고는 릴리스 4 Access Manager 또는 Access Manager SDK를 릴리스 5로 업그레이드하는 절차와 비슷합니다.

- [릴리스 3 Access Manager 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

주 릴리스 5 Access Manager는 일부 릴리스 3 Sun Java Communications Suite 구성 요소와 호환되지 않습니다. 또한 Access Manager를 릴리스 5로 업그레이드한 경우 Messaging Server 및 Calendar Server 사용자를 지정할 수 있도록 릴리스 3 이전의 Delegated Administrator를 릴리스 5로 업그레이드해야 합니다. 그러나 Messaging Server 및 Calendar Server 자체를 릴리스 5로 업그레이드할 필요는 없습니다.

릴리스 3 Access Manager 업그레이드

릴리스 3 Access Manager 또는 Access Manager SDK를 릴리스 5로 업그레이드하려면 [262페이지의 "Java ES 릴리스 4에서 Access Manager 업그레이드"](#)의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

여러 인스턴스 업그레이드

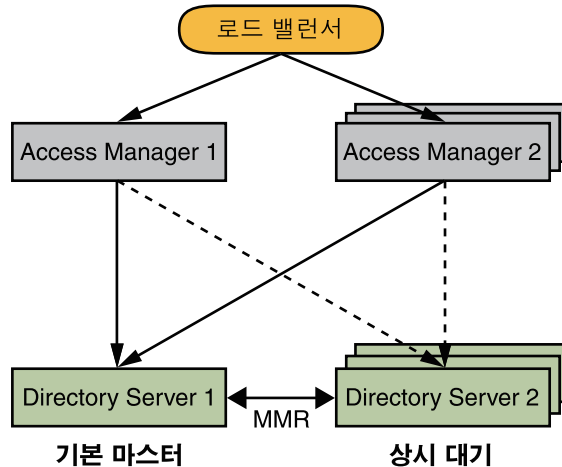
일부 배포 구조에서 Access Manager는 여러 컴퓨터 시스템에 배포되어고가용성과 확장성을 제공합니다.

일반적으로 서비스를 중단하지 않고 여러 Access Manager 인스턴스를 업그레이드하는 것이 좋습니다. 이 섹션에서는 릴리스 3 Access Manager에서 릴리스 5로 이러한 롤링 업그레이드를 수행하는 절차에 대해 설명합니다.

주 동일한 호스트 시스템에 설치된 Access Manager의 여러 인스턴스 업그레이드는 현재 릴리스에서는 지원되지 않습니다. 동일한 호스트에 여러 인스턴스가 있는 경우 주 인스턴스를 업그레이드한 후 추가 인스턴스를 다시 생성해야 합니다.

다음 그림의 배포 아키텍처는 롤링 업그레이드 절차에 대해 설명하는 데 사용됩니다.

그림 14-2 여러 Access Manager 인스턴스를 위한 배포 구조 예



이 구조에서는 로드 밸런서를 통해 여러 Access Manager 인스턴스에 액세스하며 이러한 인스턴스는 다시 다중 마스터 복제 (MMR)를 위해 설정된 디렉토리에 액세스합니다. 다른 Directory Server 복제 방식도 가능하지만 MMR은 가용성이 뛰어나고 확장성이 우수한 대표적인 디렉토리 서비스입니다. [그림 14-2](#)에서는 업그레이드 절차에 대한 설명을 용이하게 진행할 수 있도록 여러 Access Manager 및 Directory Server 인스턴스를 그룹화하였습니다. 예를 들어 Access Manager 2는 n번째 Access Manager 인스턴스에서 두 번째를 나타냅니다.

릴리스 3 Access Manager를 릴리스 5로 롤링 업그레이드하는 절차는 릴리스 5 Access Manager가 릴리스 3 디렉토리 구조와 *공존할 수 없다*는 제한에 바탕을 두고 있습니다. 그러나 [그림 14-2](#)와 같이 Directory Server 인스턴스가 복제된 경우에는 다음 절차를 사용하여 롤링 업그레이드를 수행할 수 있습니다.

1. 모든 Access Manager 인스턴스의 릴리스 3 구성 정보를 백업합니다.

[258페이지 표 14-3](#)을 참조하십시오.

2. Access Manager 1의 구성을 수정합니다.

- a. Access Manager 1이 Directory Server 1 대신 Directory Server 2를 가리키도록 구성합니다.
- b. Access Manager 1을 다시 시작합니다.

Access Manager Access Manager 2부터 Access Manager n까지 업그레이드하는 동안 1은 요청을 계속 처리합니다.

3. Access Manager 2부터 Access Manager n까지 업그레이드합니다.

설명을 간략하게 하기 위해 이후 단계에서 "Access Manager 2"는 Access Manager 2부터 Access Manager n까지를 의미합니다.

- a. 로드 밸런서에서 Access Manager 2를 비활성화합니다.

요청이 더 이상 Access Manager 2로 라우팅되지 않습니다.

- b. Access Manager 2를 부분적으로 업그레이드합니다.

디렉토리 구조 및 스키마 업데이트를 제외하고 [267페이지의 "릴리스 4 업그레이드 Access Manager"](#)에 있는 [276페이지의 단계 7](#)의 설명에 따라 각 Access Manager 인스턴스를 업그레이드합니다.

- c. Directory Server MMR을 비활성화합니다.

- d. Directory Server 1의 디렉토리 구조와 스키마를 업데이트합니다.

[276페이지의 단계 7](#)에 설명된 대로 amupgrade 스크립트를 사용합니다.

Access Manager Directory Server 2의 스키마는 아직 업데이트되지 않았으므로 1이 계속 작동합니다.

- e. Access Manager 2를 다시 시작합니다.

- f. 로드 밸런서에서 Access Manager 2를 활성화합니다.

요청이 다시 Access Manager 2로 라우팅됩니다.

4. Access Manager 1을 업그레이드합니다.

- a.** 로드 밸런서에서 Access Manager 1을 비활성화합니다.

요청이 더 이상 Access Manager 1로 라우팅되지 않습니다.

- b.** Access Manager 1을 부분적으로 업그레이드합니다.

디렉토리 구조 및 스키마 업데이트를 제외하고 [267페이지의 "릴리스 4 업그레이드 Access Manager"](#)에 있는 [276페이지의 단계 7](#)의 설명에 따라 Access Manager를 업그레이드합니다.

- c.** Directory Server MMR을 활성화합니다.

이제 Directory Server 2의 스키마 및 데이터가 업데이트됩니다.

- d.** Directory Server 1을 가리키도록 Access Manager 2의 구성을 복원합니다.

- e.** Access Manager 1을 다시 시작합니다.

- f.** 로드 밸런서에서 Access Manager 1을 활성화합니다.

요청이 다시 Access Manager 1은 물론 다른 모든 업그레이드된 Access Manager 인스턴스로 라우팅됩니다.

Java ES 릴리스 2에서 Access Manager 업그레이드

Java ES 2004Q2(릴리스 2) Access Manager를 릴리스 5로 업그레이드하는 절차는 몇 가지 차이점을 제외하고 다음 절에 나와 있는 대로 릴리스 4 Access Manager를 릴리스 5로 업그레이드하는 절차와 비슷합니다.

- [사전 업그레이드 작업](#)
- [릴리스 2 Access Manager 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

또한 Java ES 2004Q2(릴리스 2) Access Manager SDK를 릴리스 5로 업그레이드하는 절차는 몇 가지 예외가 있지만 릴리스 4 Access Manager SDK를 릴리스 5로 업그레이드하는 절차와 비슷합니다([281페이지의 "릴리스 4 Access Manager SDK 전용 업그레이드"](#) 참조). Access Manager SDK 전용 업그레이드에는 현지화, Access Manager 관리 도구 JSP 사용자 정의 및 디렉토리 스키마 마이그레이션과 관련된 절차가 제외됩니다.

릴리스 2 Access Manager SDK 및 원격 인터페이스의 역할을 할 전체 릴리스 2 Access Manager는 릴리스 5로 업그레이드해야 합니다. 릴리스 2와 릴리스 5 구성 요소의 조합은 지원되지 않습니다. 따라서 모든 컴퓨터에서 릴리스 2 Access Manager의 모든 인스턴스 및 릴리스 2 Access Manager SDK를 릴리스 5로 업그레이드해야 합니다.

주

Linux 플랫폼에서 릴리스 2 Access Manager를 업그레이드하는 경우 Access Manager *외* 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Access Manager가 지원되지 않음). 자세한 내용은 [261페이지의 "이중 업그레이드"](#)를 참조하십시오.

사전 업그레이드 작업

Access Manager를 업그레이드하기 전에 다음 예외 및 추가 사항을 적용하고 [263페이지의 "사전 업그레이드 작업"](#)에 설명된 절차를 수행하십시오.

- [Access Manager 종속성 업그레이드](#)
- [디렉토리 스키마 업그레이드](#)
- [디렉토리 색인 다시 작성](#)

Access Manager 종속성 업그레이드

릴리스 4에서의 업그레이드와 비교하여, 릴리스 2에서 릴리스 5로의 사전 업그레이드 작업에는 모든 공유 구성 요소(47페이지 표 1-9 참조)와 Access Manager가 종속되는 모든 로컬 상주 제품 구성 요소를 릴리스 5로 업그레이드 하는 작업이 포함됩니다.

Access Manager 종속성을 업그레이드하는 경우 Access Manager를 업그레이드하기 전에 다음 순서대로 종속성을 업그레이드해야 합니다. 이미 업그레이드된 항목은 건너뛸 수 있습니다.

1. **공유 구성 요소.** 63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다. 그러나 릴리스 5 Access Manager를 새로 설치하면 설치 프로그램이 Java ES 공유 구성 요소를 자동으로 업그레이드합니다.
2. **Directory Server.** Directory Server가 Access Manager와 동일한 컴퓨터에 상주하는 경우는 드물지만 114페이지의 "Java ES 릴리스 2에서 Directory Server 업그레이드"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **웹 컨테이너 소프트웨어.** Web Server 또는 Application Server의 업그레이드 지침은 151페이지의 "Java ES 릴리스 3에서 Web Server 업그레이드" 및 214페이지의 "Java ES 릴리스 2에서 Application Server 업그레이드"에서 각각 제공됩니다.

디렉토리 스키마 업그레이드

Messaging Server 및 Calendar Server와 같은 Communication Suite 구성 요소를 지원하기 위해 Sun Java Communications Suite의 Directory 준비 도구(comm_dssetup.pl)를 사용하여 Directory Server를 구성한 경우 Access Manager를 업그레이드하기 전에 Directory Preparation Tool 6.4를 사용하여 디렉토리 스키마를 먼저 업그레이드해야 합니다(<http://docs.sun.com/doc/819-7561>의 Sun Java Communications Suite 5 Upgrade Guide 참조). Access Manager 종속성을 업그레이드한 후 이 사전 업그레이드 작업을 수행합니다.

디렉토리 색인 다시 작성

Access Manager 업그레이드를 수행할 때 복잡한 문제를 피하려면 디렉토리 스키마를 업그레이드(위의 "디렉토리 스키마 업그레이드" 참조)한 후 아래와 같이 Access Manager 디렉토리 루트 접미어에 대해 수동으로 색인을 다시 작성해야 합니다.

릴리스 2-릴리스 4 Directory Server:

1. `cd serverRoot/slapd-`hostname``
2. `./db2index.pl -D "cn=directory manager" -w passwordFile -n databaseName`

여기서 *databaseName*의 기본값은 *userRoot*입니다.

릴리스 5 Directory Server:

1. `cd DirServer-base/ds6/bin`
2. `./dsconf reindex -D "cn=Directory Manager" -e -w passwordFile suffix`

각 인수에 대한 설명은 다음과 같습니다.

-e는 비보안 연결을 지정합니다.

-D는 디렉토리 관리자입니다.

-w는 비밀번호만 포함된 비밀번호 파일입니다.

*suffix*는 Access Manager 디렉토리 루트 접미어입니다.

디렉토리의 항목 수에 따라 색인을 다시 작성하는 데 긴 시간이 소요될 수 있습니다.

릴리스 2 Access Manager 업그레이드

Access Manager를 릴리스 2에서 릴리스 5로 업그레이드하는 절차는 Access Manager 소프트웨어가 배포되는 웹 컨테이너에 따라 다릅니다.

릴리스 2 Access Manager: Web Server 웹 컨테이너 업그레이드

Web Server 웹 컨테이너로 배포하는 경우 릴리스 2 Access Manager를 릴리스 5로 업그레이드하려면 [267페이지의 "릴리스 4 업그레이드 Access Manager"](#)에 있는 지침을 따릅니다. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바꾸어 생각하십시오.

릴리스 2 Access Manager: Application Server 웹 컨테이너 업그레이드

Application Server 웹 컨테이너로 배포하는 경우 릴리스 2 Access Manager를 릴리스 5로 업그레이드하려면 다음 두 가지 경우가 있습니다.

- **릴리스 5 Application Server가 새로 설치된 경우.** 이 경우 릴리스 2 Access Manager를 릴리스 5로 업그레이드하려면 [267페이지의 "릴리스 4 업그레이드 Access Manager"](#)에 있는 지침을 따릅니다. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바꾸어 생각하십시오.
- **릴리스 2 Application Server가 릴리스 5로 업그레이드된 경우.** 이 경우 Access Manager가 원래 배포된 릴리스 2 Application Server 인스턴스(instanceName)는 릴리스 5로 업그레이드할 때 업그레이드 프로세스로 인해 생성된 노드 에이전트 아래로 마이그레이션됩니다. 이러한 업그레이드된 Application Server 인스턴스 내의 Access Manager를 업그레이드하려면 다음 절의 단계를 수행해야 합니다.

업그레이드 요약

Access Manager 업그레이드 절차는 다음 단계로 구성됩니다.

1. [Access Manager Mobile Access 소프트웨어를 업그레이드합니다.](#)
2. [Access Manager의 Java ES 릴리스 2 버전을 제거합니다.](#) ampre71upgrade 스크립트를 사용합니다.
3. [릴리스 5 업그레이드를 현지화해야 하는 경우 릴리스 2 현지화 패키지를 제거합니다.](#) 이 단계는 수동으로 수행해야 합니다.
4. [Access Manager의 Java ES 릴리스 5 버전을 설치합니다.](#) Java ES 설치 프로그램에서 나중에 구성 옵션을 사용합니다.
5. [Access Manager용 JSP를 다시 사용자 정의합니다.](#)
6. [Directory Server가 실행되고 있는지 확인합니다.](#)
7. [다음 Application Server 인스턴스를 시작합니다.](#) Access Manager가 배포된 DAS(Domain Administration Server), 노드 에이전트 및 서버 인스턴스.
8. [Access Manager를 배포 해제하고 Application Server 인스턴스로 재구성 및 재배포합니다.](#) amconfig 스크립트를 사용합니다.
9. [Access Manager classpath-suffix 및 server-classpath 정보가 릴리스 5 Application Server domain.xml 파일로 마이그레이션되었는지 확인합니다.](#)
10. [DAS\(Domain Administration Server\), 노드 에이전트 및 서버 인스턴스를 중지합니다.](#)
11. [DAS\(Domain Administration Server\), 노드 에이전트 및 서버 인스턴스를 다시 시작합니다.](#)
12. [디렉토리 구조와 스키마를 업데이트합니다.](#) amupgrade 스크립트를 사용합니다.

이러한 단계는 다음 절차에 나와 있습니다.

업그레이드 절차

1. Access Manager Mobile Access 소프트웨어를 업그레이드합니다.

릴리스 2 버전에 패치를 사용하여 Access Manager Mobile Access 소프트웨어를 업그레이드해야 합니다. 다음 표에 필요한 패치가 나와 있습니다.

표 14-13 Access Manager Mobile Access 소프트웨어를 업그레이드하기 위한 패치¹

설명	패치 아이디: Solaris 9 및 10	패치 아이디: Linux
Mobile Access 소프트웨어	119530-05(SPARC) 119531-05(x86)	119532-05 - sun-identity-mobileaccess-6.2-25.3.i386.rpm - sun-identity-mobileaccess-config-6.2-25.3.i386.rpm

1. 패치 버전 번호는 Java ES 릴리스 5로 업그레이드하는 데 필요한 최소 버전입니다. 보다 최신 버전이 있으면 표에 표시된 버전 대신 최신 버전을 사용하십시오.

a. 표 14-6의 패치 번호를 사용하여 필요한 패치를 구합니다.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

b. 패치의 README 파일에 나와 있는 사전 패치 절차를 수행합니다.

c. 패치에서 필요한 다음 매개 변수의 값을 구합니다.

표 14-14 Mobile Access 패치 매개 변수

매개 변수	값
디렉토리 관리자 DN	기본값: cn=Directory Manager
디렉토리 관리자 비밀번호	

- d. 표 14-6에 있는 패치를 적용합니다.

Solaris의 경우
`patchadd patch_ID`

Linux의 경우
`./update`

패치의 README 파일에 나와 있는 사후 패치 절차를 수행합니다.

2. Access Manager의 Java ES 릴리스 2 버전을 제거합니다.
 - a. 릴리스 4 Access Manager를 호스팅하는 컴퓨터에 루트로 로그인하거나 수퍼유저가 됩니다.
`su -`
 - b. 디렉토리를 Java ES 릴리스 5 배포의 `os_arch/Product/identity_svr/Tools`로 변경합니다. 여기서 `os_arch`는 `Solaris_sparc`과 같은 사용자의 플랫폼에 해당합니다.
 - c. `ampre71upgrade` 스크립트에 필요한 다음 매개 변수의 값을 구합니다.

표 14-15 Access Manager 구성 매개 변수: `ampre71upgrade`

매개 변수	값
Directory Server 호스트	정규화된 이름 설정: <code>hostname.domain</code>
Directory Server 포트	SSL이 아닌 포트 번호 ¹ 를 지정합니다. 기본값: 389
최상위 관리자 DN	기본값: <code>uid=amadmin,ou=People,default_org_DN</code>
최상위 관리자 비밀번호	
백업 파일을 저장할 디렉토리	기본값: <code>AccessManager-base</code>

1. 기본 SSL 값인 636과 다른 Directory Server SSL 포트를 지정해야 합니다.

- d. Directory Server가 실행되고 있는지 확인하십시오. 그렇지 않은 경우 Directory Server를 시작하십시오.
- e. `ampre71upgrade` 스크립트를 실행합니다.

`./ampre71upgrade`

스크립트는 Access Manager 구성 파일을 백업하고 릴리스 4 기본 패키지를 제거합니다. 현지화된 패키지는 아래 단계 3에 따라 수동으로 제거해야 합니다.

3. 릴리스 5 업그레이드를 현지화해야 하는 경우 릴리스 2 현지화 패키지를 제거합니다.

단계 2에서 실행되는 `ampre71upgrade` 스크립트는 현지화 패키지를 제거하지 않으므로 다음과 같이 수동으로 제거해야 합니다.

Solaris의 경우

- a. 현지화 패키지를 확인합니다.

```
pkginfo | grep SUNWaml
pkginfo | grep SUNWamclnt
pkginfo | grep SUNWamdistauth
```

- b. 단계 a에서 찾은 현지화 패키지를 제거합니다.

```
pkgrm SUNWamlLocale
pkgrm SUNWamclntLocale
pkgrm SUNWamdistauthLocale
```

Linux의 경우

- a. 현지화 RPM을 확인합니다.

```
rpm -qa | grep sun-identity-sdk-*
rpm -qa | grep sun-identity-clientsdk-*
rpm -qa | grep sun-identity-distauth-*
```

- b. 단계 a에서 찾은 현지화 RPM을 제거합니다.

```
rpm -e sun-identity-sdk-Locale-*
rpm -e sun-identity-clientsdk-Locale-*
rpm -e sun-identity-distauth-Locale-*
```

4. Access Manager의 Java ES 릴리스 5 버전을 설치합니다.

다음 단계를 수행합니다.

- a. 릴리스 2 Access Manager를 호스팅하는 컴퓨터에서 Java ES 설치 프로그램을 실행합니다.

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

여기서 `os_arch`는 `Solaris_sparc`과 같은 사용자의 플랫폼에 해당합니다. 명령 줄 인터페이스를 사용하려면 `installer -nodisplay` 옵션을 사용하십시오.

시작 및 사용권 계약 페이지가 표시된 다음 구성 요소 선택 페이지가 표시됩니다. 설치된 구성 요소를 Java ES 설치 프로그램에서 직접 업그레이드할 수 있는 경우에는 상태가 "업그레이드 가능"으로 표시됩니다.

- b. 구성 요소 선택 페이지에서 Access Manager를 선택합니다.
 - c. 릴리스 2가 설치된 디렉토리와 동일한 설치 디렉토리를 지정합니다.
 - d. "나중에 구성" 옵션을 선택합니다.
 - e. 필요한 경우 현지화된 패키지를 설치하는 옵션을 선택합니다.
 - f. 설치가 완료되면 Java ES 설치 프로그램을 종료합니다.
5. Access Manager용 JSP를 다시 사용자 정의합니다.

266페이지의 "웹 컨테이너 사용자 정의 파일 백업"에 저장한 Access Manager 콘솔 및 인증 사용자 인터페이스(UI)를 위한 JSP에 릴리스 2 사용자 정의를 다시 적용합니다.

그 다음 사용자 정의 JSP 파일을 올바른 디렉토리에 복사합니다.

- 레거시 전용 Access Manager 콘솔
AccessManager-base/web-src/applications/console
- 인증 UI:
AccessManager-base/web-src/services/config/auth/default 또는
AccessManager-base/web-src/services/config/auth/default_Locale
(여기서 Locale은 ja와 같은 로캘 표시기입니다.)

자세한 내용은 Sun Java System Access Manager 7.1 Developer's Guide (<http://docs.sun.com/doc/819-4675>)를 참조하십시오.

- 6. Directory Server가 실행되고 있는지 확인합니다.
- 7. 다음 Application Server 인스턴스를 시작합니다.

다음 명령과 후속 단계에서 다음 규칙이 사용됩니다.

- 여기서 *nodeagentName*은 *hostName_domainName* 형식이지만 기본값은 간단히 *hostName*입니다.
- 기본 *domainName*은 domain1입니다.
- 기본 *instanceName*은 server1입니다.

주 서버 인스턴스를 시작하기 전에 아래와 같이 `startinstances=false` 옵션을 사용하여 노드 에이전트를 별도로 시작하십시오.

- a. DAS(Domain Administration Server)를 시작합니다.

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
domainName
```

- b. 업그레이드된 Application Server 인스턴스가 마이그레이션된 노드 에이전트를 시작합니다.

```
AppServer8-base/bin/asadmin start-node-agent
--startinstances=false --user admin_ID nodeagentName
```

- c. 해당 서버 인스턴스가 실행되고 있지 않은 경우 Access Manager가 배포된 서버 인스턴스(instanceName)를 시작합니다.

```
AppServer8-base/bin/asadmin start-instance --user admin_ID
instanceName
```

8. Access Manager를 배포 해제하고 Application Server 인스턴스로 재구성 및 재배포합니다.

- a. 웹 컨테이너가 SSL 모드로 실행 중인 경우 컨테이너의 SSL 인증서가 만료되지 않았고 아직 유효한지 확인합니다.

- b. amsamplesilent 템플릿 입력 파일을 기반으로 amconfig 입력 파일을 생성합니다.

```
cp amsamplesilent config-file
```

(후속 단계에서 config-file은 amsamplesilent와 같은 디렉토리에 있다고 가정합니다.)

- c. config-file에 구성 매개 변수를 설정합니다.

모든 매개 변수는 올바르게 설정되어야 합니다. 일부 값은 AMConfig.properties 파일에서 마이그레이션할 수 있으며 일부 값은 다음 표에 표시된 것처럼 업그레이드 절차에 따라 달라집니다.

표 14-16 Access Manager 구성 매개 변수: amconfig

매개 변수	값
업그레이드 매개 변수	
DEPLOY_LEVEL	배포 해제의 경우 26으로 설정합니다. 재구성 및 배포의 경우 1로 설정합니다.
DIRECTORY_MODE	5로 설정합니다.
AM_REALM¹	레거시 모드를 사용하는 경우 disabled로 설정합니다. 영역 모드를 사용하는 경우 enabled로 설정합니다. 기본값: enabled

표 14-16 Access Manager 구성 매개 변수: amconfig (계속)

매개 변수	값
JAVA_HOME	JDK 릴리스 5 디렉토리로 설정: /usr/java/jdk1.5.0_04/
WEB_CONTAINER	Application Server 8.x의 경우 AS8로 설정합니다. 그리고 <i>config-file</i> 의 해당 섹션에만 값을 입력합니다.
AS81_INSTANCE (Application Server 7.x에서 업 그레이드한 Application Server 8.x를 웹 컨테이너로 사용하는 경우)	Application Server 7.x <i>instanceName</i> 으로 설정 합니다. 기본값은 server1입니다.
AS81_INSTANCE_DIR (Application Server 8.x를 웹 컨 테이너로 사용하는 경우)	인스턴스의 Application Server 8.x 도메인 디렉 토리로 설정합니다. 기본값은 다음과 같습니다. <i>AppServer8Config-base/domains/domain1</i>
AS81_DOCS_DIR (Application Server 8.x를 웹 컨 테이너로 사용하는 경우)	인스턴스의 Application Server 8.x docroot 디렉 토리로 설정합니다. 기본값은 다음과 같습니다. <i>AppServer8Config-base/domains/domain1/docroot</i>

AMConfig.properties에서 마이그레이션

SERVER_PROTOCOL	com.ipplanet.am.server.protocol
SERVER_PORT	com.ipplanet.am.server.port
SERVER_HOST	com.ipplanet.am.server.host
DS_HOST	com.ipplanet.am.directory.host
DS_PORT	com.ipplanet.am.directory.port
ROOT_SUFFIX ²	com.ipplanet.am.defaultOrg
CONSOLE_DEPLOY_URI	com.ipplanet.am.console.deploymentDescriptor
SERVER_DEPLOY_URI	com.ipplanet.am.services.deploymentDescriptor
PASSWORD_DEPLOY_URI	com.sun.identity.password.deploymentDescriptor
AM_ENC_PWD ²	am.encryption.pwd ³

1. 영역 및 레거시 모드에 대한 자세한 내용은 [259페이지의 "호환성 문제"](#)를 참조하십시오.
2. 이 매개 변수의 값은 이전 버전의 Access Manager와 같아야 합니다.
3. Access Manager 및 Access Manager SDK를 모두 배포한 경우 이 등록 정보 값은 Access Manager 인스턴스 및 해당 Access Manager SDK 인스턴스에서 같아야 합니다.

다른 매개 변수는 웹 컨테이너나 비밀번호를 변경한 경우가 아니면 릴리스 2 구성에 사용된 것과 동일한 값을 제공하십시오.

- d. amconfig를 실행하여 Access Manager를 배포 해제합니다.

*config-file*에 있는 DEPLOY_LEVEL의 값을 26로 설정합니다.

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

- e. 공용 에이전트 컨테이너가 실행되고 있는지 확인합니다.

```
netstat -an | grep 11163
```

실행되고 있지 않으면 시작합니다.

```
/usr/sbin/cacoadm start
```

- f. amconfig를 실행하여 Access Manager를 재구성하고 웹 컨테이너로 배포합니다.

*config-file*에 있는 DEPLOY_LEVEL의 값을 1로 설정합니다.

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

9. Access Manager classpath-suffix 및 server-classpath 정보가 릴리스 5 Application Server domain.xml 파일로 마이그레이션되었는지 확인합니다.

- a. Access Manager가 원래 배포된 릴리스 2 Application Server 인스턴스의 server.xml 파일에 있는 Access Manager classpath-suffix 및 server-classpath 정보를 확인합니다.

```
AppServer7Config-base/domains/domainName/instanceName/config/server.xml
```

- b. Access Manager가 배포된 업그레이드된 Application Server 인스턴스의 domain.xml 파일에 추가된 classpath-suffix 및 server-classpath 항목을 확인합니다.

```
AppServer8Config-base/nodeagents/nodeagentName/instanceName/
config/domain.xml
```

클래스 경로 정보를 릴리스 5 Application Server domain.xml 파일의 *instanceName*-config 블록에 추가해야 합니다. 이 블록은 다음 행으로 시작됩니다.

```
<config dynamic-reconfiguration-enabled="true"
name="instanceName-config">
```

10. 디렉토리 구조와 스키마를 업데이트합니다.

릴리스 5 Access Manager는 릴리스 4 디렉토리 구조와 함께 사용할 수 있지만 구조를 릴리스 5 기능을 지원하도록 수정해야 합니다. 다음 디렉토리에 설치되어 있는 amupgrade 스크립트를 실행하여 Access Manager 디렉토리 구조와 스키마를 릴리스 5로 업데이트합니다.

- *Solaris의 경우*
AccessManager-base/upgrade/scripts
- *Linux의 경우*
AccessManager_base/identity/upgrade/scripts
- a. amupgrade 스크립트에 필요한 다음 매개 변수의 값을 구합니다.

표 14-17 Access Manager 구성 매개 변수: amupgrade

매개 변수	값
Directory Server 호스트	정규화된 이름 설정: <i>hostname.domain</i>
Directory Server 포트	SSL이 아닌 포트 번호 ¹ 를 지정합니다. 기본값: 389
디렉토리 관리자 DN	기본값: cn=Directory Manager
디렉토리 관리자 비밀번호	
최상위 관리자 DN	기본값: uid=amadmin,ou=People,default_org_DN
최상위 관리자 비밀번호	
영역 모드 사용 (릴리스 4 영역 모드에서 업그레이드하는 경우 이 매개 변수 값은 필요하지 않습니다.)	y/n: 예는 영역 모드를 사용할 수 있고 서비스 데이터가 새 영역 트리로 마이그레이션되었다는 의미입니다. 아니요(기본값)는 서비스 데이터가 레거시 모드로 남아있다는 의미입니다.

1. 636과 같은 기본 SSL 값을 갖는 Directory Server SSL 포트를 지정하면 업그레이드 프로세스가 성공적으로 완료되지 않습니다.

- b. amupgrade 스크립트를 실행합니다.

```
cd AccessManager-base/upgrade/scripts
./amupgrade
```

성공적으로 업그레이드되면 스크립트는 "Upgrade completed"라고 표시합니다.

- c. 디렉토리 스키마 확장에 대한 자세한 내용은 다음 업그레이드 로그 파일을 확인하십시오.

Solaris의 경우

`/var/sadm/install/logs/`

`Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm`

Linux의 경우

`/var/log/Sun_Java_System_Access_Manager_upgrade_dit_log.mmddhhmm`

11. DAS(Domain Administration Server), 노드 에이전트 및 서버 인스턴스를 중지합니다.

이러한 인스턴스는 [296페이지의 단계 7](#)에서 시작한 인스턴스입니다.

```
AppServer8-base/bin/asadmin stop-domain --user admin_ID
domainName
```

```
AppServer8-base/bin/asadmin stop-node-agent --user admin_ID
nodeagentName
```

12. DAS(Domain Administration Server), 노드 에이전트 및 서버 인스턴스를 다시 시작합니다.

주 서버 인스턴스를 시작하기 전에 아래와 같이 `startinstances=false` 옵션을 사용하여 노드 에이전트를 별도로 시작하십시오.

```
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`의 기본값은 4848입니다.

Access Manager 업그레이드 확인

업그레이드 절차를 마친 뒤 [277페이지의 "Access Manager 업그레이드 확인"](#)에서 설명한 대로 성공적으로 업그레이드되었는지 확인합니다.

사후 업그레이드 작업

Security Assertion Markup Language(SAML) 서비스를 사용하는 경우 Access Manager 콘솔을 통해 SAM 인증 모듈을 추가하여 사용할 수 있도록 해야 합니다. SAML 인증 모듈 인스턴스 생성에 대한 자세한 내용은 Sun Java System Access Manager 7.1 관리 설명서(<http://docs.sun.com/doc/820-0841>)를 참조하십시오.

업그레이드 롤백

Access Manager를 업그레이드 전 상태로 롤백할 수 있는 스크립트는 없습니다. 이 프로세스는 사전 업그레이드 작업의 일부분으로 백업해 두었던 Access Manager 데이터를 사용하여 직접 수행해야 합니다([266페이지의 "릴리스 4 Access Manager 로그 및 디버그 파일 백업"](#) 참조). 롤백은 너무 까다로워서 실용적이지 않습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서 Access Manager는 여러 컴퓨터 시스템에 배포되어고가용성과 확장성을 제공합니다.

일반적으로 서비스를 중단하지 않고 Access Manager 인스턴스를 업그레이드하는 것이 좋습니다. 릴리스 2 Access Manager를 릴리스 5로 롤링 업그레이드하는 절차는 릴리스 5 Access Manager가 릴리스 2 디렉토리 구조와 공존할 수 없다는 제한에 바탕을 두고 있습니다. 그러나 [그림 14-2](#)와 같이 Directory Server 인스턴스가 복제된 경우에는 [285페이지의 "여러 인스턴스 업그레이드"](#)에 설명된 대로 롤링 업그레이드를 수행할 수 있습니다.

Portal Server

이 장에서는 Portal Server를 Java ES 5(릴리스 5): Sun Java System Portal Server 7.1로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- [305페이지의 "Portal Server 업그레이드 개요"](#)
- [311페이지의 "Java ES 릴리스 4에서 Portal Server 업그레이드"](#)
- [340페이지의 "Java ES 릴리스 3에서 Portal Server 업그레이드"](#)
- [343페이지의 "Java ES 릴리스 2에서 Portal Server 업그레이드"](#)
- [353페이지의 "임시 기능 릴리스 7.0에서 Portal Server 업그레이드"](#)

주 이 장에 설명된 파일 위치는 *PortalServer6-base*와 *PortalServer6Config-base*(Portal Server 6.x) 및 *PortalServer7-base*와 *PortalServer7Config-base*(Portal Server 7.x)라는 디렉토리 경로에 대해 상대적으로 지정되어 있습니다. Portal Server를 처음 설치할 때 최소한 이 두 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 15-1 Portal Server 디렉토리 경로

경로 이름 변수	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 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Portal Server의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Portal Server 정보](#)
- [Portal Server 업그레이드 로드맵](#)
- [Portal Server 데이터](#)
- [Portal Server 업그레이드 전략](#)

Java ES 릴리스 5 Portal Server 정보

Java ES 릴리스 5 Portal Server는 릴리스 4와 비교할 때 여러 가지 새 기능과 향상된 기능이 포함된 주요 릴리스입니다. 이러한 변경 사항 중 상당수는 릴리스 4에 이어 릴리스된 임시 기능 릴리스(IFR)에서 변경된 것이므로 IFR과 비교했을 때 릴리스 5는 사소한 기능 변경만 포함되어 있습니다. IFR의 향상된 기능과 새 기능에 대한 자세한 내용은 Sun Java System Portal Server 7.1 릴리스 노트

(<http://docs.sun.com/app/docs/doc/820-0899/6ncjv9dm5?a=view>)를 참조하십시오. 특히 릴리스 4 명령줄 관리 인터페이스는 psadmin 명령으로 대체되었습니다.

Portal Server 업그레이드 로드맵

[표 15-2](#)는 Portal Server를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 15-2 Java ES 5(릴리스 5) 업그레이드 경로: Portal Server 7.1

Java ES 릴리스	Portal Server 버전	일반적인 방법	재구성 필요
임시 기능 릴리스 (IFR)	Sun Java System Portal Server IFR 7.0 2005Q4	직접 업그레이드: 패치를 적용하고 업그레이드 스크립트를 사용하여 수행합니다.	사용자 정의는 수동으로 다시 적용해야 합니다.
릴리스 4	Sun Java System Portal Server 6.3.1 2005Q4	직접 업그레이드: 업그레이드 스크립트를 사용하여 수행합니다.	사용자 정의는 수동으로 다시 적용해야 합니다.
릴리스 3	Sun Java System Portal Server 6.3.1 2005Q1	직접 업그레이드: 업그레이드 스크립트를 사용하여 수행합니다.	사용자 정의는 수동으로 다시 적용해야 합니다.

표 15-2 Java ES 5(릴리스 5) 업그레이드 경로: Portal Server 7.1 (계속)

Java ES 릴리스	Portal Server 버전	일반적인 방법	재구성 필요
릴리스 2	Sun Java System Portal Server 6.3 2004Q2	직접 업그레이드: 업그레이드 스크립트를 사용하여 수행합니다.	사용자 정의는 수동으로 다시 적용해야 합니다.
릴리스 1	Sun ONE Portal Server 6.2(2003Q4)	직접 업그레이드 불가능: 그러나 먼저 릴리스 3으로 업그레이드한 다음 릴리스 3에서 릴리스 5로 업그레이드할 수 있습니다.	구성 데이터
이전 Java ES 릴리스		직접 업그레이드할 수 없습니다.	

Portal Server 데이터

다음 표는 Portal Server 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 15-3 Portal Server 데이터 사용

데이터 유형	위치	사용
구성 데이터	<i>PortalServer6Config-base/</i>	Portal Server 구성
웹 컨테이너 액세스 제어 및 구성 파일	Web Server 7.0(Java ES 릴리스 5) server.policy 및 server.xml 파일의 경우 <i>WebServer7Config-base/https-configName/config</i> Web Server 6.x(Java ES 릴리스 2, 3 및 4) server.policy 및 server.xml 파일의 경우 <i>WebServer6-base/https-hostname/config</i> Application Server 8.x(Java ES 릴리스 3, 4 및 5) server.policy 및 domain.xml 파일의 경우 <i>AppServer8Config-base/domains/domainName/config</i> Application Server 7.x(Java ES 릴리스 2): server.policy 및 server.xml 파일의 경우 <i>AppServer7Config-base/domains/domainName/config</i>	Portal Server 웹 컨테이너 인스턴스의 구성
사용자 정의 데이터	<i>PortalServer6Config-base/desktop</i>	사용자 정의 모듈용 JAR 파일 사용자 정의 샘플 Portal Server 데스크탑

표 15-3 Portal Server 데이터 사용 (계속)

데이터 유형	위치	사용
디렉토리 스키마 서비스 구성 사용자 데이터	Directory Server	Portal Server는 포털 데스크탑 및 디렉토리에 저장된 사용자 프로필 데이터와 같은 서비스 구성에 따라 달라집니다.
동적 응용 프로그램 데이터	없음	Portal Server는 세션 상태와 같은 응용 프로그램 데이터를 지속적으로 저장하지 않습니다.

Portal Server 업그레이드 전략

Portal Server 업그레이드를 위한 전략은 일반적으로 1장, "업그레이드 계획"에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Portal Server 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Portal Server에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Portal Server는 Portal Server 및 Portal Server Secure Remote Access 구성 요소를 관리하는 데 사용되는 psadmin 명령에서 공용 인터페이스가 변경되었습니다.

Sun Java System Portal Server 7.1 Command-Line Reference

(<http://docs.sun.com/doc/819-5030>)를 참조하십시오.

따라서 릴리스 5 Portal Server는 여러 인스턴스 배포를 롤링 업그레이드하는 전환 기간을 제외하면 이전 버전 또는 Portal Server Secure Remote Access 구성 요소의 이전 버전(SRA Gateway, Rewriter 프록시 및 Netlet 프록시 포함)과 호환되지 않습니다. 모든 Portal Server 인스턴스는 Portal Server Secure Remote Access 구성 요소 인스턴스와 함께 Java ES 릴리스 5로 동기화되어야 합니다.

또한 Mobile Access 구성 요소가 포함된 개별 Portal Server 구성 요소는 이전 버전과 호환되지 않으며 모두 Java ES 릴리스 5로 동기화되어야 합니다.

또한 릴리스 5 Portal Server 및 이전 Portal Server 버전에서 사용된 Directory Server 데이터 구조 간에는 호환되지 않습니다. 이러한 비호환성은 동일한 Directory Server 데이터를 사용하는 여러 Portal Server 인스턴스의 롤링 업그레이드에 영향을 줍니다.

Portal Server 종속성

다른 Java ES 구성 요소에 대한 Portal Server 종속성은 Portal Server 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어, Portal Server 인터페이스 또는 기능의 변경 사항을 적용하려면 Portal Server가 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Portal Server는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Portal Server는 특정한 Java ES 공유 구성 요소(47페이지 표 1-9 참조)에 종속됩니다.
- **웹 컨테이너.** Portal Server는 웹 컨테이너 서비스에 대해 필수 종속성이 있으며 Java ES Web Server, Java ES Application Server 또는 Weblogic 및 WebSphere와 같은 타사 웹 컨테이너에서 이 서비스를 제공할 수 있습니다.

주 타사 웹 컨테이너에 배포하는 경우에는 Portal Server를 릴리스 5로 업그레이드할 수 없습니다. Weblogic 및 WebSphere의 웹 컨테이너에 배포하는 경우 릴리스 5 Portal Server를 새로 설치해야 합니다.

- **Access Manager(또는 Access Manager SDK).** Portal Server는 단일 사인온을 포함하여 최종 사용자를 위한 인증 및 권한 부여 서비스를 제공하는 Access Manager에 대해 필수 종속성이 있습니다. Access Manager가 원격 컴퓨터에서 실행되는 경우 Access Manager SDK는 로컬에서 사용할 수 있어야 합니다.
- **Directory Server.** Portal Server는 Access Manager를 통해 액세스되는 사용자 데이터를 저장하는 Directory Server에 대해 필수 종속성이 있습니다. 따라서, Portal Server를 업그레이드하려면 디렉토리 스키마를 확장해야 합니다.
- **Portal Server Secure Remote Access.** Portal Server는 Gateway, Rewriter 프록시 및 Netlet 프록시 구성 요소를 통해 보안 원격 액세스를 제공하는 Portal Server Secure Remote Access에 대해 선택적 종속성이 있습니다.
- **Java DB.** Portal Server는 여러 포틀릿 응용 프로그램에 대한 지원을 제공하는 Java DB에 대해 선택적 종속성이 있습니다.
- **Service Registry.** Portal Server는 컴파일에 필요한 라이브러리를 제공하는 Service Registry에 대해 필수 종속성이 있습니다.
- **Communications Express.** Portal Server는 최종 사용자에게 메시징 및 달력 채널을 제공하는 데 사용되는 Sun Java Communications Suite 구성 요소인 Communications Express에 대해 선택적 종속성이 있습니다. Communications Express는 더 이상 Java ES 제품 구성 요소가 아닙니다.

선택적 업그레이드 문제

일반적으로 Java ES 릴리스 5는 컴퓨터에서 모든 구성 요소에 대한 선택적 업그레이드를 지원하지만 Portal Server는 많은 다른 Java ES 구성 요소에 대한 종속성이 있기 때문에 다양한 Java ES 릴리스 버전에서 임의의 구성 요소 조합을 선택하기가 매우 어렵습니다.

이런 이유로 Portal Server는 Access Manager 및 웹 컨테이너와 관련하여 제한된 업그레이드 시나리오를 지원합니다.

- **Java ES 릴리스 4에서 Portal Server를 업그레이드하는 경우.** Directory Server, Access Manager 및 웹 컨테이너(Web Server 또는 Application Server)를 릴리스 5로 업그레이드한 다음 Portal Server를 업그레이드하거나 Portal Server ^만 릴리스 5(다른 구성 요소는 릴리스 4 수준으로 유지)로 업그레이드할 수 있지만 일부 종속성을 릴리스 4로 유지하고 다른 구성 요소를 릴리스 5로 업그레이드할 수는 없습니다.
- **Java ES 릴리스 3에서 Portal Server를 업그레이드하는 경우.** Directory Server, Access Manager 및 웹 컨테이너(Web Server 또는 Application Server)를 릴리스 4 또는 릴리스 5로 업그레이드한 다음 Portal Server를 업그레이드해야 합니다. 그러나 종속성을 릴리스 3으로 유지하거나 일부 종속성을 릴리스 4로 업그레이드하고 다른 종속성을 릴리스 5로 업그레이드할 수는 없습니다.
- **Java ES 릴리스 2에서 Portal Server를 업그레이드하는 경우.** Portal Server를 업그레이드하기 전에 Directory Server, Access Manager 및 웹 컨테이너(Web Server 또는 Application Server)를 릴리스 4 또는 릴리스 5로 업그레이드해야 합니다. 릴리스 2의 종속성을 남겨 두거나 일부 종속성을 릴리스 4로 업그레이드하고 나머지는 릴리스 5로 업그레이드할 수 없습니다.

웹 컨테이너 업그레이드 시나리오

Portal Server는 Web Server 또는 Application Server에서 제공하는 웹 컨테이너에 배포할 수 있습니다. 따라서 배포되는 웹 컨테이너를 릴리스 5로 업그레이드해야 할 수 있기 때문에 릴리스 5로 Portal Server를 업그레이드하는 작업은 복잡해질 수 있습니다. 이와 관련하여 가능한 여러 가지 웹 컨테이너 업그레이드 시나리오가 다음 표에 정리되어 있습니다.

표 15-4 Portal Server 업그레이드를 위한 웹 컨테이너 업그레이드 시나리오

시나리오	원래 Portal Server가 배포된 웹 컨테이너	업그레이드 후 Portal Server를 배포할 웹 컨테이너	적용 가능 Portal Server업그레이드 경로: 업그레이드 원본
시나리오 1	Web Server 6.x	Web Server 6.x	릴리스 2 릴리스 3 릴리스 4 IFR 7.0
시나리오 2	Web Server 6.x	Web Server 7.0	릴리스 2 릴리스 3 릴리스 4
시나리오 3	Application Server 8.1	Application Server 8.1	릴리스 3 릴리스 4 IFR 7.0
시나리오 4	Application Server 8.1	Application Server 8.2	릴리스 3 릴리스 4 IFR 7.0
시나리오 5	Application Server 7x	Application Server 8.2	릴리스 2

Portal Server를 업그레이드할 때(예를 들어 psupgrade 스크립트를 사용하는 경우) 특히, 웹 컨테이너에 대한 주요 버전 업그레이드를 수행하는 경우 해당하는 표 15-4의 업그레이드 시나리오에 대해 적절한 값을 제공해야 합니다.

이중 업그레이드

Portal Server와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 43페이지의 "[이중 업그레이드: Java ES 및 운영 체제 소프트웨어](#)"에서 설명한 대로 기존 운영 체제 업그레이드 방식을 사용하여 수행할 수 있습니다.

1. 기존 Portal Server 데이터를 백업합니다.

필수 데이터의 위치는 306페이지의 "[Portal Server 데이터](#)"를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Portal Server로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 해당 절을 참조하십시오.

Java ES 릴리스 4에서 Portal Server 업그레이드

이 절에는 Portal Server를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다.

주 이 절은 Portal Server가 Application Server 웹 컨테이너에 배포되고 릴리스 5로 업그레이드하기 전에 릴리스 2에서 릴리스 3 또는 릴리스 4로 업그레이드한 특수한 경우에는 적용되지 않습니다. 이러한 업그레이드 경로는 현재 지원되지 않습니다.

이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Portal Server 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Portal Server를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 업그레이드 스크립트 `psupgrade`를 사용하여 업그레이드를 수행합니다. 이 스크립트는 새 패키지를 설치하고 필요한 경우 구성 데이터를 마이그레이션하며 현지화 파일을 업데이트하고 Portal Server 웹 응용 프로그램을 웹 컨테이너로 다시 배포합니다.
- **업그레이드 종속성.** Portal Server에는 여러 Java ES 공유 구성 요소에 대한 종속성이 있습니다([47페이지 표 1-9](#) 참조). 릴리스 5 Portal Server는 릴리스 4 버전의 이러한 공유 구성 요소와 호환되지만 Portal Server를 업그레이드하는 데 사용되는 `psupgrade` 스크립트가 릴리스 5 버전의 ANT 공유 구성 요소를 사용하므로 공유 구성 요소를 필수적으로 업그레이드해야 합니다.

또한 릴리스 5 Portal Server는 [308페이지의 "Portal Server 종속성"](#)에 설명한 대로 웹 컨테이너, Access Manager 및 Directory Server에 대한 종속성이 있습니다. 이러한 종속성을 업그레이드하는 두 가지 방법이 지원됩니다([309페이지의 "선택적 업그레이드 문제"](#) 참조).

- 릴리스 4 구성 요소로 모든 종속성 충족(*전혀* 릴리스 5로 업그레이드되지 않음)
- 릴리스 5 구성 요소로 모든 종속성이 충족(*모두* 릴리스 5로 업그레이드됨)
- **역방향 호환성.** 릴리스 5 Portal Server는 릴리스 4 버전과 호환되지 않습니다.
- **업그레이드 롤백.** Portal Server의 릴리스 5 업그레이드를 릴리스 4로 롤백하는 작업은 릴리스 4 패키지 복원, 릴리스 4 디렉토리 데이터 복원 및 웹 컨테이너로 Portal Server 웹 응용 프로그램 재배포로 구성됩니다.
- **플랫폼 문제.** Portal Server를 업그레이드하는 일반적인 방법은 Solaris와 Linux 운영 체제에서 동일하지만, Solaris OS의 경우 릴리스 5 Portal Server는 새 경로에 설치되고 Linux OS의 경우에는 릴리스 4와 같은 경로에 설치됩니다.

릴리스 4 Portal Server 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Portal Server를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [릴리스 4 사전 업그레이드 작업](#)
- [릴리스 4 Portal Server 업그레이드\(Solaris\)](#)
- [릴리스 4 Portal Server 업그레이드\(Linux\)](#)
- [업그레이드 확인](#)
- [릴리스 4 사후 업그레이드 작업](#)
- [업그레이드 롤백\(Solaris\)](#)
- [업그레이드 롤백\(Linux\)](#)

릴리스 4 사전 업그레이드 작업

Portal Server를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- 현재 버전 정보 확인
- Portal Server 종속성 업그레이드
- 필수 구성 정보 및 비밀번호 얻기
- 릴리스 4 Portal Server 구성 정보 백업
- JVM(Java Virtual Machine) 설정 기록
- 로드 밸런서의 구성 제거
- Directory Proxy Server 구성 제거

현재 버전 정보 확인

다음 명령을 사용하여 Portal Server의 현재 버전을 확인할 수 있습니다.

```
PortalServer6-base/bin/version
```

표 15-5 Portal Server 버전 확인 출력

Java ES 릴리스	Portal Server 버전 번호
릴리스 2	6.3
릴리스 3	6.3.1
릴리스 4	6.3.1 ¹
IFR 릴리스	7.0
릴리스 5	7.1

1. 릴리스 3과 릴리스 4의 유일한 차이점은 패치입니다. 릴리스 4 패치를 확인하려면 Solaris에서는 `showrev -p | grep patch_ID` 명령을 사용하고 Linux에서는 `rpm -qa sun-portal-core` 명령을 사용한 다음 Java ES 릴리스 4 업그레이드 설명서에 나와 있는 버전과 비교하면 됩니다.

Portal Server 종속성 업그레이드

일반적으로 컴퓨터 시스템(및 컴퓨팅 환경 내)의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다.

릴리스 5 Portal Server는 릴리스 4 버전의 Java ES 공유 구성 요소와 호환되지만 Portal Server를 업그레이드하는 데 사용되는 `psupgrade` 스크립트는 릴리스 5 버전의 ANT 공유 구성 요소를 사용하므로 공유 구성 요소를 필수적으로 업그레이드해야 합니다.

Portal Server 제품 구성 요소 종속성 중 하나라도 릴리스 5로 업그레이드하기로 한 경우에는 모두 업그레이드해야 합니다(309페이지의 "선택적 업그레이드 문제" 참조). 종속성은 Portal Server를 업그레이드하기 전에 다음 순서대로 업그레이드해야 하며 이미 업그레이드한 항목에 대한 작업은 생략할 수 있습니다.

1. **공유 구성 요소.** 63페이지의 "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다.
2. **Directory Server.** 99페이지의 5장, "Directory Server"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **웹 컨테이너 소프트웨어.** Web Server 또는 Application Server의 업그레이드 지침은 129페이지의 7장, "Web Server" 및 199페이지의 11장, "Application Server"에서 각각 제공됩니다.

주 WebLogic 및 WebSphere의 웹 컨테이너와 같은 타사 웹 컨테이너를 업그레이드하는 경우 Portal Server를 지원하는 이러한 컨테이너에 대해 사용자 정의한 내용을 덮어 쓰게 되므로 Portal Server에 문제가 발생할 수 있습니다.

이와 같은 경우 업그레이드된 웹 컨테이너 환경에 맞도록 Portal Server를 다시 설치하고 재구성해야 합니다.

4. **Access Manager(Access Manager SDK).** 255페이지의 14장, "Access Manager"에서 Access Manager를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
5. **Portal Server Secure Remote Access.** 373페이지의 16장, "Portal Server Secure Remote Access"에서 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
6. **Java DB.** 153페이지의 8장, "Java DB"에서 Java DB를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
7. **Service Registry.** 227페이지의 12장, "Service Registry"에서 Service Registry를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
8. **Communications Express.** Sun Java Communications Suite Upgrade Guide (<http://docs.sun.com/doc/819-7561>)에서 Communications Express를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

필수 구성 정보 및 비밀번호 얻기

웹 컨테이너 업그레이드 시나리오(310페이지 표 15-4 참조)에 따라 psupgrade 스크립트에서 비밀번호 및 기타 웹 컨테이너 구성 데이터에 대한 정보를 입력해야 할 수도 있습니다. 여러 웹 컨테이너 업그레이드 시나리오에 필요한 정보는 표 15-6에 나와 있습니다. Portal Server 업그레이드를 시작하기 전에 필요한 관련 정보를 수집하십시오.

표 15-6 웹 컨테이너 업그레이드 시나리오마다 psupgrade 스크립트에 필요한 정보

정보	업그레이드 시나리오 ³	Web Server 7.x 값의 예: 시나리오 2	Application Server 8.x 값의 예: 시나리오 5 ²
Web Server 7.0에서 Portal Server 업그레이드(예/아니오)	2	예	N/A
웹 컨테이너 설치 디렉토리	2 및 5	WebServer7-base	AppServer8Install-base
웹 컨테이너 가상 서버 인스턴스 이름	2	https-configName ⁴	N/A
웹 컨테이너 인스턴스 이름	5	N/A	server1
웹 컨테이너 인스턴스 디렉토리	2	WebServer7Config-base/ https-configName ⁴ /	N/A
포털 인스턴스 배포 디렉토리	5	N/A	AppServer8Config-base/ domains/domainName
웹 컨테이너 인스턴스 포트	2 및 5	80	80
웹 컨테이너 인스턴스 프로토콜	2 및 5	http	http
웹 컨테이너 구성 이름	2	configName ⁴	N/A
웹 컨테이너 도메인 이름	5	N/A	domain1
웹 컨테이너 문서 루트 디렉토리	2 및 5	WebServer7Config-base/ https-configName ⁴ /docs/	AppServer8Config-base/ domains/domainName/ docroot
웹 컨테이너 관리 호스트 이름	2 및 5	localhost	localhost
웹 컨테이너 관리 포트	2 및 5	8989	4848
웹 컨테이너 관리 프로토콜	2 및 5	https	https
웹 컨테이너 관리자 아이디	2 및 5	admin	admin
웹 컨테이너 관리자 비밀번호	2부터 5까지		
웹 컨테이너 마스터 비밀번호	3부터 5까지	N/A	
디렉토리 관리자(cn=Directory manager) 비밀번호	1부터 5까지		
SRA 로그 사용자 비밀번호 ¹	1부터 5까지		

표 15-6 웹 컨테이너 업그레이드 시나리오마다 psupgrade 스크립트에 필요한 정보 (계속)

정보	업그레이드 시나리오 ³	Web Server 7.x 값의 예: 시나리오 2	Application Server 8.x 값 의 예: 시나리오 5 ²
Access Manager 관리자 비밀번호	1부터 5까지		
Directory Server ldapuser 비밀번호	1부터 5까지		
포털 인스턴스 아이디 ²	1부터 5까지		

1. 이 정보는 Portal Server와 함께 설치하는 경우 Portal Server Secure Remote Access 구성 요소를 구성하는 데 필요합니다.
2. 이 매개 변수에는 null이 아닌 고유 값을 제공해야 합니다. 값은 영숫자여야 하며 하이픈(-)을 포함할 수 있습니다.
3. 웹 컨테이너 업그레이드 시나리오 #5는 릴리스 2에서 Portal Server를 업그레이드하는 경우 적용됩니다.
4. configName의 기본값은 hostName.domainName입니다.

릴리스 4 Portal Server 구성 정보 백업

Portal Server를 릴리스 5로 업그레이드하기 위해 Portal Server 소프트웨어를 재구성해야 하는 것은 아닙니다. 그러나 안전을 위한 조치로 psupgrade 스크립트에서 구성 정보가 저장되는 다음 디렉토리를 백업합니다.

PortalServer6Config-base/

주	Portal Server가 사용자 프로필 및 기타 데이터를 저장하는 Directory Server 인스턴스를 백업하는 것이 좋습니다. 이러한 데이터가 없으면 릴리스 5 Portal Server 업그레이드를 롤백할 수 없습니다.

JVM(Java Virtual Machine) 설정 기록

기본값과 다른 경우에는 Portal Server를 업그레이드하기 전에 다음 웹 컨테이너 JVM 설정을 기록해 두십시오.

```
<jvm-options>-XX:MaxPermSize=256m</jvm-options>
<jvm-options>-XX:+CMSPermGenSweepingEnabled</jvm-options>
<jvm-options>-XX:+CMSClassUnloadingEnabled</jvm-options>
```

JVM 설정의 위치는 다음 표에 나와 있는 것처럼 웹 컨테이너에 따라 다릅니다.

표 15-7 JVM 설정의 위치

웹 컨테이너	구성 파일
릴리스 2, 3, 4 Web Server (6.x)	<i>WebServer6-base/https-instanceName/config/server.xml</i>
릴리스 5 Web Server (7.0)	<i>WebServer7Config-base/https-configName/config/server.xml</i>
릴리스 3, 4, 5 Application Server (8.x)	<i>AppServer8Config-base/domains/domainName/config/domain.xml</i>
Application Server 인스턴스 노드 에이전트에서 관리	<i>AppServer8Config-base/nodeagents/nodeagentName/config/domain.xml</i>

Portal Server 업그레이드 절차를 수행한 결과 이러한 JVM 설정이 변경되지 않았는지 나중에 확인해야 합니다.

로드 밸런서의 구성 제거

로드 밸런서를 통해 Portal Server 인스턴스에 액세스하는 경우 `LOAD_BALANCER_URL` 등록 정보 값으로 인해 이러한 액세스 작업이 Portal Server 업그레이드를 방해할 수 있습니다. 따라서 이 설정은 업그레이드를 수행하기 전에 변경해야 합니다.

`LOAD_BALANCER_URL` 등록 정보 설정을 수정하려면 다음을 수행합니다.

1. 다음 구성 파일 중 로컬에 있는 파일을 확인합니다(로컬로 설치된 Portal Server Secure Remote Access 구성 요소를 지원하는 파일 중 일부).

PortalServer6Config-base/PSConfig.properties

PortalServer6Config-base/GWConfig.properties (Gateway가 로컬인 경우)

PortalServer6Config-base/RWPCConfig.properties (Rewriter 프록시가 로컬인 경우)

PortalServer6Config-base/NLPConfig.properties (Netlet 프록시가 로컬인 경우)

2. 이러한 구성 요소 파일에 `LOAD_BALANCER_URL` 등록 정보의 현재 값을 기록합니다.
3. `LOAD_BALANCER_URL` 등록 정보 값이 해당 Portal Server 인스턴스를 가리키도록 수정합니다.

`LOAD_BALANCER_URL=portalHostName:port/portal`

Directory Proxy Server 구성 제거

Portal Server 인스턴스가 Directory Proxy Server 인스턴스를 통해 Directory Server에 액세스하는 경우 업그레이드를 수행하기 전에 Directory Proxy Server 호스트 및 포트 번호 설정을 수정하고 업그레이드가 완료되면 원래 값으로 복원해야 합니다.

적절한 설정을 수정하려면 다음을 수행합니다.

1. 다음 Access Manager 구성 파일에 DS_HOST 및 DS_PORT 등록 정보의 현재 값을 기록합니다.

AccessManagerConfig-base/config/AMConfig.properties

2. DS_HOST 및 DS_PORT 등록 정보의 값을 해당 Directory Server 인스턴스를 직접 가리키도록 수정합니다.

릴리스 4 Portal Server 업그레이드(Solaris)

이 절에서는 Portal Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항(Solaris)

Portal Server 소프트웨어를 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- 동일한 설치 Portal Server 이미지에 해당하는 모든 Portal Server 인스턴스가 동시에 업그레이드됩니다.
- Portal Server 소프트웨어는 여러 다른 역할을 수행하는 하위 구성 요소로 구성되지만 모두 함께 업그레이드됩니다.
 - **Portal-base.** 관리 Mbeans 및 해당 관리 소프트웨어, 로깅 프레임워크 및 모니터링 관련 소프트웨어가 포함되며 이들은 모두 함께 패키지에 포함됩니다.
 - **Portal Server 웹 응용 프로그램.** 웹 컨테이너에 배포되는 여러 웹 응용 프로그램으로 구성됩니다. 이러한 웹 응용 프로그램 중 최소한 몇 개는 Access Manager에서 지원을 필요로 하며 결과적으로 Directory Server에서도 지원이 필요합니다.
 - **Secure Remote Access Core.** Portal Server Secure Remote Access를 지원하는 소프트웨어: jar 파일에 포함된 일부 서블릿 및 애플릿, 그리고 웹 컨테이너에 배포할 수 없는 일부 지원 파일.

- psupgrade 스크립트는 호스트 컴퓨터에 설치되어 있는 Portal Server 하위 구성 요소와 웹 컨테이너 종속성을 자동으로 검색합니다. 예를 들어, 스크립트는 Portal Server 웹 응용 프로그램을 배포하려는 Application Server 또는 Web Server의 버전을 검색하기 위해 시스템을 쿼리하고 검색되는 정보에 따라 요청하는 정보를 조정합니다.

업그레이드 절차(Solaris)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server에 적용됩니다.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

자세한 내용은 [63페이지의 2장](#), "Java ES 공유 구성 요소 업그레이드"를 참조하십시오.

이 단계는 [320페이지의 단계 8](#)에서 psupgrade 스크립트를 실행하기 위한 필수 사전 요구 사항입니다.

3. 로컬에서 실행 중인 Portal Server Secure Remote Access Gateway, Rewriter 프록시 또는 Netlet 프록시의 인스턴스를 중지합니다.

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxycd stop
```

프로세스가 중지되었는지 확인합니다.

```
게이트웨이: netstat -an | grep 443
Rewriter 프록시: netstat -an | grep 10443
Netlet 프록시: netstat -an | grep 10555
```

4. Portal Server가 배포된 웹 컨테이너와 다른 웹 컨테이너에 Access Manager가 배포된 경우 실행되고 있는지 확인합니다.
5. 아직 실행되고 있지 않은 경우 Portal Server가 배포된 웹 컨테이너를 시작하여 Portal Server를 시작합니다.

```
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. psupgrade 스크립트에 필요한 두 환경 변수 (ANT_HOME 및 JAVA_HOME)를 설정합니다. 예를 들면 다음과 같습니다.

```
export ANT_HOME=/usr/sfw
```

```
export JAVA_HOME=/usr/jdk/entsys-j2se
```

7. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

8. Java ES 릴리스 5 배포판에서 psupgrade 스크립트를 실행합니다.

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
```

```
./psupgrade
```

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다.

주 실수로 잘못된 *os_arch* 디렉토리에서 psupgrade를 실행한 경우 다음과 같이 절차를 되돌려야 합니다.

- a. 올바른 *os_arch* 디렉토리로 변경합니다.

- b. Portal Server 데이터에 대한 변경을 되돌립니다.

```
./psupgrade rollback
```

요청된 매개 변수 및 비밀번호를 제공합니다.

- c. psupgrade를 다시 실행합니다.

psupgrade 스크립트는 설치된 Portal Server 구성 요소 및 현지화 패키지를 검색하고 Java ES 설치 프로그램을 호출하여 새 패키지를 설치한 다음 Portal Server 웹 응용 프로그램을 배포하는 웹 컨테이너의 위치, 포트 번호 및 기타 정보를 검색하기 위해 시스템을 쿼리합니다. 웹 컨테이너 업그레이드 시나리오(310페이지 표 15-4 참조)에 따라 스크립트에서 Portal Server를 적절한 웹 컨테이너에 배포하는 데 필요한 추가 정보를 입력해야 할 수도 있습니다.

315페이지 표 15-6에는 표 15-4의 여러 웹 컨테이너 업그레이드 시나리오에 필요한 정보가 나와 있습니다.

주	다시 돌아가서 변경할 수 없으므로 psupgrade 매개 변수에 정확한 값을 입력해야 하는데, psupgrade 스크립트에서 수정한 변경 사항을 롤백하기는 매우 어렵습니다. Portal Server 데이터에 대한 변경 사항을 되돌리려면 다음을 수행해야 합니다. <pre>./psupgrade rollback</pre> 이 작업은 psupgrade를 다시 실행하기 전에 수행해야 합니다.

9. 필요한 경우 웹 컨테이너 JVM 설정을 복원합니다.

JVM 설정이 릴리스 5 Portal Server를 지원하는지 확인하려면 다음 단계를 수행합니다.

- a. 업그레이드 전에 기록한 Portal Server의 웹 컨테이너 JVM 설정이 업그레이드 절차에서 변경되지 않았는지 확인합니다.

316페이지의 "JVM(Java Virtual Machine) 설정 기록"을 참조하십시오.

- b. 설정이 변경된 경우 업그레이드하기 전에 기록했던 값으로 복원합니다.

이전에 설정하지 않았어도 다음 JVM 설정은 포함되어야 합니다.

```
<jvm-options>-XX:MaxPermSize=256m</jvm-options>
<jvm-options>-XX:+CMSPermGenSweepingEnabled</jvm-options>
<jvm-options>-XX:+CMSClassUnloadingEnabled</jvm-options>
```

10. 웹 컨테이너를 중지하고 다시 시작합니다.

모든 상황에 반드시 적용되는 것은 아니지만 웹 컨테이너를 다시 시작하면 Portal Server가 원래 상태로 다시 시작됩니다.

- a. 다음과 같이 웹 컨테이너를 중지합니다.

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. [319페이지의 단계 5](#)에 있는 명령을 사용하여 웹 컨테이너를 다시 시작합니다.

릴리스 4 Portal Server 업그레이드(Linux)

이 절에서는 Portal Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항(Linux)

Linux 플랫폼에서 Portal Server 소프트웨어를 릴리스 5로 업그레이드하는 경우 Solaris 플랫폼에서 업그레이드할 때의 고려 사항을 참조하십시오([318페이지의 "업그레이드 고려 사항\(Solaris\)"](#) 참조). Linux OS의 릴리스 4와 동일한 경로에 릴리스 5 Portal Server가 설치된 경우는 제외됩니다. 결과적으로 psupgrade 스크립트는 릴리스 5 RPM을 설치할 때 이전 RPM을 제거합니다.

업그레이드 절차(Linux)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server에 적용됩니다.

주의	Linux에서 Java ES 릴리스 4를 릴리스 5로 업그레이드하는 작업은 롤백할 수 없습니다. 다음 절차를 수행하기 전에 시스템을 백업하십시오.

- 루트로 로그인하거나 슈퍼유저가 됩니다.

`su -`

2. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

자세한 내용은 [63페이지의 2장](#), "Java ES 공유 구성 요소 업그레이드"를 참조하십시오.

이 단계는 [324페이지의 단계 8](#)에서 psupgrade 스크립트를 실행하기 위한 필수 사전 요구 사항입니다.

3. 로컬에서 실행 중인 Portal Server Secure Remote Access Gateway, Rewriter 프록시 또는 Netlet 프록시의 인스턴스를 중지합니다.

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxysd stop
```

프로세스가 중지되었는지 확인합니다.

```
게이트웨이: netstat -an | grep 443
Rewriter 프록시: netstat -an | grep 10443
Netlet 프록시: netstat -an | grep 10555
```

4. Portal Server가 배포된 웹 컨테이너와 다른 웹 컨테이너에 Access Manager가 배포된 경우 실행되고 있는지 확인합니다.
5. 아직 실행되고 있지 않은 경우 Portal Server가 배포된 웹 컨테이너를 시작하여 Portal Server를 시작합니다.

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. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다. 예를 들면 다음과 같습니다.

```
export ANT_HOME=/opt/sun
export JAVA_HOME=/usr/jdk/entsys-j2se
```

7. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

8. Java ES 릴리스 5 배포판에서 psupgrade 스크립트를 실행합니다.

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

여기서 *os_arch*는 Linux_x86과 같은 사용자의 플랫폼에 해당합니다.

psupgrade 스크립트는 설치된 Portal Server 구성 요소 및 현지화 패키지를 검색하고 Java ES 설치 프로그램을 호출하여 새 패키지를 설치한 다음 Portal Server 웹 응용 프로그램을 배포하는 웹 컨테이너의 위치, 포트 번호 및 기타 정보를 검색하기 위해 시스템을 쿼리합니다. 웹 컨테이너 업그레이드 시나리오([310페이지 표 15-4](#) 참조)에 따라 스크립트에서 Portal Server를 적절한 웹 컨테이너에 배포하는 데 필요한 추가 정보를 입력해야 할 수도 있습니다.

[315페이지 표 15-6](#)에는 [표 15-4](#)의 여러 웹 컨테이너 업그레이드 시나리오에 필요한 정보가 나와 있습니다.

주 다시 돌아가서 변경할 수 없으므로 psupgrade 매개 변수에 정확한 값을 입력해야 하는데, psupgrade 스크립트에서 수정한 변경 사항을 롤백하기는 매우 어렵습니다. 주의: psupgrade 스크립트를 실행하기 전에 시스템을 백업하십시오.

9. *PortalServer7Config-base/platform.conf.default* 구성 파일을 수정합니다.

psupgrade에서 백업한 다음 파일에서 gateway.logging.password가 있는 줄을 복사합니다.

```
PortalServer6Config-base.bak/platform.conf.default
```

그리고 해당 줄을 *PortalServer7Config-base/platform.conf.default*에 삽입합니다.

10. 필요한 경우 웹 컨테이너 JVM 설정을 복원합니다.

JVM 설정이 릴리스 5 Portal Server를 지원하는지 확인하려면 다음 단계를 수행합니다.

- a. 업그레이드 전에 기록한 Portal Server의 웹 컨테이너 JVM 설정이 업그레이드 절차에서 변경되지 않았는지 확인합니다.

[316페이지의 "JVM\(Java Virtual Machine\) 설정 기록"](#)을 참조하십시오.

- b. 설정이 변경된 경우 업그레이드하기 전에 기록했던 값으로 복원합니다.

이전에 설정하지 않았어도 다음 JVM 설정은 포함되어야 합니다.

```
<jvm-options>-XX:MaxPermSize=256m</jvm-options>
<jvm-options>-XX:+CMSPermGenSweepingEnabled</jvm-options>
<jvm-options>-XX:+CMSClassUnloadingEnabled</jvm-options>
```

11. 웹 컨테이너를 중지하고 다시 시작합니다.

모든 상황에 반드시 적용되는 것은 아니지만 웹 컨테이너를 다시 시작하면 Portal Server가 원래 상태로 다시 시작됩니다.

- a. 다음과 같이 웹 컨테이너를 중지합니다.

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. 323페이지의 단계 5에 있는 명령을 사용하여 웹 컨테이너를 다시 시작합니다.

업그레이드 확인

다음 명령을 사용하여 릴리스 5 패키지가 설치되었는지 확인할 수 있습니다.

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID
-f adminpasswordfile.
```

출력 값에 대해서는 313페이지 표 15-5를 참조하십시오.

전체 업그레이드를 확인하려면 포털 데스크탑이 실행되고 psadmin 관리 유틸리티가 설명된 대로 작동하는지 확인합니다.

다음 업그레이드 로그 파일을 확인할 수도 있습니다.

```
/var/sadm/install/logs/Sun_Java_System_Portal_Server_upgrade.log*
PortalServer7Data-base/logs/admin/
PortalServer7Data-base/logs/config/
```

릴리스 4 사후 업그레이드 작업

다음 상황을 해결하려면 사후 업그레이드 절차가 필요합니다.

- 사용자 정의 **web-src** 데이터 마이그레이션
- 사용자 정의 포틀릿 응용 프로그램 재배포
- 사용자 정의 포틀릿 응용 프로그램 마이그레이션
- 책갈피 및 응용 프로그램 채널의 정확한 링크
- Search Server에 대한 올바른 액세스
- Directory Proxy Server의 구성 복원
- 수동으로 Portal Server Secure Remote Access 구성 요소 등록
- URLScrapper 채널 활성화
- 로그아웃 페이지의 변경 사항

사용자 정의 web-src 데이터 마이그레이션

portal.war을 구성하기 위해 이미지, javascript 파일 또는 다른 파일 등의 사용자 정의 데이터를 다음 디렉토리에 추가한 경우

PortalServer6-base/web-src

이러한 추가 파일을 릴리스 5 Portal Server의 해당하는 디렉토리로 복사해야 합니다.

PortalServer7-base/web-src

사용자 정의 포틀릿 응용 프로그램 재배포

사용자 정의 포틀릿 응용 프로그램을 만들고 배포한 경우에는 릴리스 5 Portal Server로 업그레이드한 다음 이러한 포틀릿을 수동으로 재배포해야 합니다. 디스플레이 프로파일 항목이 있고 채널 이름이 표시되더라도 사용자 정의 포틀릿을 재배포하기 전에는 내용이 보이지 않습니다.

다음 명령을 사용하여 포틀릿을 재배포합니다.

PortalServer7-base/bin/psadmin deploy-portlet

다음 위치에서 해당 .war 및 XML 파일을 찾아보면 재배포를 확인할 수 있습니다.

PortalServer7Data-base/portals/Upgraded/war

사용자 정의 포틀릿 응용 프로그램 마이그레이션

SJWC(Sun Java Web Console)에서 제공하는 사용자 인터페이스 프레임워크를 바탕으로 한 포틀릿 응용 프로그램은 수동으로 릴리스 5로 마이그레이션 및 재배포해야 합니다.

이러한 요구 사항은 특히 Portal Server와 함께 샘플 포틀릿 응용 프로그램으로 배포되어 포털로 사용자 정의 및 설치되는 4개의 웹 응용 프로그램인 filesharing, surveys, wiki, rssportlet에 해당됩니다. 업그레이드 중에 이러한 샘플 포틀릿 응용 프로그램의 버그 수정 버전이 포틀릿 응용 프로그램 영역에 배치됩니다. 용도에 따라 이러한 포틀릿 응용 프로그램을 사용자 정의한 경우 수동으로 릴리스 5로 마이그레이션한 다음 재배포해야 하며, Portal Server 업그레이드 프로세스에서는 처리되지 않습니다.

기본적으로 이러한 포틀릿 응용 프로그램의 일부(filesharing 및 surveys)는 커뮤니티를 만들 때 배포되어 사용자에게 제공됩니다.

다음 절차를 사용하여 SJWC 기반 포틀릿 응용 프로그램을 업그레이드, 사용자 정의 및 재배포할 수 있습니다. filesharing 포틀릿 응용 프로그램을 예로 들 수 있습니다.

1. 릴리스 5 SJWC jar 파일을 추출합니다.

- 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. 파일 중 하나의 이름을 변경합니다.
`mv WEB-INF/lib/commons-collections-3.1.jar
WEB-INF/lib/commons-collections.jar`

2. filesharing 포틀릿 응용 프로그램을 찾습니다.

```
cd PortalServer7Config-base/portals/portall/portletapps/filesharing
```

3. 업데이트된 SJWC 라이브러리를 추가합니다.

```
jar uvf src/filesharing.war.tokenized -C /tmp/lh WEB-INF
```

4. filesharing 포틀릿 응용 프로그램을 사용자 정의합니다.

`ant customize`

5. filesharing 포틀릿 응용 프로그램을 재배포합니다.

- a. `PortalServer7-base/bin/psadmin undeploy-portlet -u amadmin -f passwordfile -p portal_id -i instance_id -g filesharing`

- b. `ant deploy`

- c. 다음 디렉토리(웹 컨테이너에 따라 다름)로 이동합니다.

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. `sun-web.xml` 파일을 열고 다음 줄을 마지막 줄 바로 앞(즉, `sun-web-app` 종료 태그 앞)에 추가합니다.

`<class-loader delegate="false"/>`

- e. `filesharing/WEB-INF`의 `sun-web.xml` 파일에 대해 단계 c 및 단계 d를 반복합니다.

6. `surveys` 및 `SJWC` 프레임워크를 바탕으로 한 다른 사용자 정의 포틀릿 응용 프로그램에 대해 단계 2에서 단계 5까지 반복합니다.

7. 웹 컨테이너를 다시 시작합니다.

책갈피 및 응용 프로그램 채널의 정확한 링크

릴리스 4 Portal Server를 릴리스 5로 업그레이드하면 책갈피와 응용 프로그램 채널에 중복된 링크 및 잘못된 링크가 생성됩니다. 이러한 링크 문제를 해결하려면 다음 절차를 수행하십시오.

1. `PSConsole`에 로그인합니다.
2. "일반적인 작업" 탭에서 "채널 및 컨테이너 관리"를 누릅니다.
3. `DeveloperSample [Org]`를 DN으로 선택하고 "확인"을 누릅니다.

4. "보기 유형"으로 JSPTabContainer [기본값]을 선택합니다.
5. MyFrontPageTabPanelContainer에서 "App Channel"을 누릅니다.
 맨 오른쪽 프레임에 응용 프로그램 채널 등록 정보가 표시됩니다.
 특정 로캘의 등록 정보를 보려면 "표 기본 설정"을 누르고 "로캘" 값(de, fr, es, ja, ko, zh, zh_CN, zh_TW)을 지정합니다.
6. userApps 등록 정보를 편집합니다.
 - a. userApps 등록 정보에서 "[값 편집...]" 링크를 누릅니다.
 기존 응용 프로그램이 표시된 팝업 창이 나타납니다.
 - b. 목록에서 다음 응용 프로그램을 제거합니다.
 NetMail Lite
 NetMail
 - c. 목록에 다음 응용 프로그램을 추가합니다.
 NetFile
 - d. "저장"을 누른 다음 "닫기"를 누릅니다.
7. 대상 등록 정보를 편집합니다.
 - a. 대상 등록 정보에서 "[값 편집...]" 링크를 누릅니다.
 기존 대상이 표시된 팝업 창이 나타납니다.
 - b. 목록에서 다음 대상을 제거합니다.
 NetMailLite|
 NetMailServlet?nsid=newHTMLSessionNetMailLite|
 NetMailServlet?nsid=newHTMLSession

 NetMail|NetMailServlet?nsid=newAppletSession
 - c. 다음 대상이 중복 표시되는 경우 중복되는 항목을 제거합니다.
 Instant Messenger (Java WebStart) |
 IMLaunch?provider=IMChannel&launch=jnlp&last=false

 Instant Messenger (Browser) |
 IMLaunch?provider=IMChannel&launch=plugin&last=false
 - d. 목록에 다음 대상을 추가합니다.
 NetFile|/portal/NetFileApplet?Refer=java2
 - e. "저장"을 누른 다음 "닫기"를 누릅니다.

Search Server에 대한 올바른 액세스

릴리스 4 Portal Server를 릴리스 5로 업그레이드할 때 검색 서버가 Portal Server에서 분리되므로 검색 서버의 URL 액세스 또한 변경됩니다.

- 릴리스 4의 경우 URL은 `http://hostName:port/portal/search`입니다.
- 릴리스 5의 경우 URL은 `http://hostName:port/UpgradedSearch/search`입니다.

결과적으로 Search, DiscussionLite, Discussions 및 Instant Messaging 채널 등의 SearchProvider 또는 DiscussionProvider 인터페이스를 구현하는 모든 포털 채널의 디스플레이 프로필을 수동으로 수정해야 합니다. 특히 디스플레이 프로필 내에서 어떤 조직 또는 역할 수준에서든지 이러한 채널의 searchServer 등록 정보가 올바른 검색 서버를 가리키도록 수정해야 합니다. searchServer 값을 다음과 같이 수정합니다.

```
value="http://hostName:port/UpgradedSearch/search"
```

또한 Instant Messaging 채널의 경우 Instant Messaging Server 구성 등록 정보 `iim_arch.portal.search`를 새로운 검색 서버 URL로 업데이트해야 합니다.

Directory Proxy Server의 구성 복원

Portal Server 인스턴스가 Directory Proxy Server 인스턴스를 통해 Directory Server에 액세스한 경우 업그레이드하기 전의 원래 값으로 Directory Proxy Server 호스트 및 포트 번호 설정을 복원해야 합니다. 업그레이드 준비 시 이러한 등록 정보 값을 수정한 경우 [318페이지의 "Directory Proxy Server 구성 제거"](#)를 참조하십시오.

수동으로 Portal Server Secure Remote Access 구성 요소 등록

업그레이드 절차 마지막에 Null 포인터 예외가 표준 출력으로 보고되면 Portal Server Secure Remote Access 구성 요소 등록이 실패했음을 의미합니다.

이 경우 다음 명령을 실행하여 수동으로 Portal Server Secure Remote Access 구성 요소를 등록(활성화)할 수 있습니다.

```
PortalServer7-base/bin/psadmin provision-sra -u amadminUser -f passwordFile
-p Portal_ID --gateway-profile profileName --enable
```

URLScrapper 채널 활성화

릴리스 4에서 릴리스 5로 업그레이드하는 경우 URLScrapper 채널을 활성화해야 합니다. 다음 절차를 수행하십시오.

1. Portal Server 콘솔에 로그인합니다.

"포털" 탭을 누른 다음 업그레이드한 포털을 누릅니다.

2. "DN 선택" 드롭다운 메뉴에서 "TopLevel (전역)"을 선택하고 "디스플레이 프로필 다운로드" 링크를 누릅니다.

다운로드한 파일을 임시 위치에 저장합니다.

3. `com.sun.portal.providers.urlscraper.URLScraperProvider`를 찾습니다.
4. 다음으로 시작하는 XML 부분을 찾습니다.

```
<Provider advanced="false"
class="com.sun.portal.providers.urlscraper.URLScraperProvider"
```

다음으로 끝나는 XML 부분을 찾습니다.

```
</Provider>
```

5. 단계 4의 XML 부분을 다음으로 대체합니다.

```
<Provider advanced="false"
class="com.sun.portal.providers.urlscraper.URLScraperProvider"
container="false" lock="false" merge="fuse"
name="URLScraperProvider" 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="UrlScraper Channel"/>

<String advanced="false" lock="false" merge="replace"
name="description" propagate="true" value="This is a test for
urlscraper"/>

<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="urlScraperRulesetID" 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="UrlScraper Channel"/>

<String advanced="false" lock="false" merge="replace"
name="description" propagate="true" value="This is a test for
urlscraper"/>

</ConditionalProperties>

</Properties>

</Provider>

```

6. 수정한 파일을 저장하고 업로드합니다.

로그아웃 페이지의 변경 사항

릴리스 5 Portal Server 로그아웃 페이지가 이전 Access Manager 로그아웃 페이지와 달라졌습니다. 이러한 변경 사항이 소프트웨어의 결함을 나타내는 것은 아닙니다.

업그레이드 롤백(Solaris)

이 절에서는 Portal Server의 업그레이드 롤백 절차에 영향을 미치는 고려 사항에 대해 설명하고 뒤이어 절차 자체에 대해 설명합니다.

롤백 고려 사항(Solaris)

릴리스 5 업그레이드를 롤백하는 절차는 *PortalServer6-base*에서 릴리스 4 설치로 되돌리고 릴리스 4 웹 응용 프로그램을 재배포하는 것으로 구성됩니다.

롤백 절차(Solaris)

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. Directory Server를 업그레이드하기 전 상태로 복원합니다.

Directory Server backup/restore 명령줄 및 GUI 유틸리티를 사용합니다. Sun Java System Directory Server Enterprise Edition 6.0 관리 설명서 (<http://docs.sun.com/doc/820-0292>)에서 Directory Server 백업 및 복원 장을 참조하십시오.

3. 해당 웹 컨테이너를 중지하여 Portal Server를 중지하십시오.

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. 릴리스 5 Portal Server 패키지를 제거합니다.

- a. Java ES 제거 프로그램을 시작합니다.

```
/var/sadm/prod/SUNWentsys5/uninstall
```

- b. 설치한 모든 Portal Server 구성 요소를 선택합니다.

- c. 제거 선택을 확인합니다.

- d. Java ES 제거 프로그램을 종료합니다.

5. 해당 웹 컨테이너를 다시 시작하여 Portal Server를 다시 시작하십시오.

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. Java ES 릴리스 5 배포에서 다음 명령을 사용하여 릴리스 4 Portal Server 웹 응용 프로그램을 재배포합니다.

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade rollback
```

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다.

psupgrade rollback 명령은 릴리스 5 Portal Server 웹 응용 프로그램을 배포 해제하고 릴리스 4 Portal Server 웹 응용 프로그램을 재배포합니다.

이 명령은 *PortalServer6-base/web-src*에서

*/var/PortalServe6-base/https-hostName/deploy-dir/web-apps*로 콘텐츠를 재배포합니다. 따라서 Portal Server 웹 응용 프로그램에 대한 모든 사용자 정의를 먼저 */web-src*로 만든 다음 */web-apps*에 배포해야 합니다. *psupgrade rollback* 명령을 실행하기 전에 */web-apps*에서 변경한 모든 사항을 */web-src*에 복제해야 합니다. 그렇지 않으면 변경 사항을 덮어 쓰게 됩니다.

7. 웹 컨테이너를 중지하고 다시 시작합니다.

모든 상황에 반드시 적용되는 것은 아니지만 웹 컨테이너를 다시 시작하면 Portal Server가 원래 상태로 다시 시작됩니다.

업그레이드 롤백(Linux)

릴리스 5로 업그레이드하려면 릴리스 4 바이너리를 제거해야 하므로 Linux에서는 업그레이드를 롤백하기가 매우 어렵습니다.

롤백하는 한 가지 방법은 업그레이드를 수행하기 전에 병렬 시스템을 만들고 업그레이드를 수행하기 전에 해당 시스템을 테스트하는 것입니다. 업그레이드를 롤백해야 하는 경우 해당 병렬 시스템으로 되돌릴 수 있습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Portal Server를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Portal Server 인스턴스를 실행하여 로드를 분산시킬 수 있습니다.

Portal Server의 인스턴스가 로드 균형 조정된 경우 아래 설명과 같이 서비스를 중단하지 않고도 Portal Server 인스턴스를 순서에 따라 업그레이드하는 롤링 업그레이드를 수행할 수 있습니다. 이 절차에서는 릴리스 4 Portal Server가 릴리스 5 Portal Server 디렉토리 데이터를 사용하여 작동하지 않는 제한 사항에 주의해야 합니다.

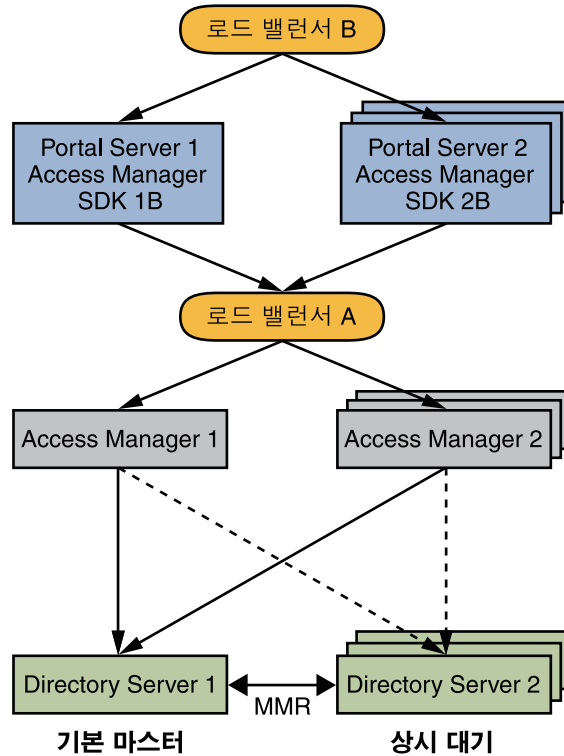
릴리스 4 Portal Server 인스턴스를 릴리스 5로 롤링 업그레이드하는 절차를 설명하는 데 [337페이지 그림 15-1](#)의 배포 구조가 사용됩니다.

주	Portal Server Secure Remote Access 구성 요소가 포함된 아키텍처의 경우에는 393페이지의 "여러 인스턴스 업그레이드" 를 참조하십시오.

[그림 15-1](#)의 구조에서는 가용성 및 확장성을 제공하기 위해 로드 밸런서에서 여러 Portal Server 인스턴스가 액세스됩니다. 결과적으로 Portal Server 인스턴스는 로드 밸런서를 통해 Access Manager에 액세스합니다. Access Manager 및 Access Manager SDK 인스턴스는 다중 마스터 복제(MMR)를 위해 설정된 디렉토리에 액세스합니다. 다른 Directory Server 복제 방식도 가능하지만 MMR은 가용성이 뛰어나고 확장성이 우수한 대표적인 디렉토리 서비스입니다.

[그림 15-1](#)에서는 업그레이드 절차를 쉽게 설명할 수 있도록 여러 Portal Server, Access Manager 및 Directory Server 인스턴스를 그룹화했습니다. Portal Server 예를 들어 2는 두 번째부터 n번째 Portal Server 인스턴스를 나타냅니다.

그림 15-1 여러 Portal Server 인스턴스를 위한 배포 구조 예



릴리스 4 Portal Server에서 릴리스 5로 롤링 업그레이드는 다음과 같이 수행됩니다.

1. 릴리스 4 Access Manager를 릴리스 5로 업그레이드하는 경우 [279페이지의 "여러 인스턴스 업그레이드"](#)에서 설명한 대로 롤링 업그레이드를 수행하십시오. 릴리스 4 Portal Server를 릴리스 5로 업그레이드할 때 릴리스 4 Access Manager를 릴리스 5로 업그레이드할 필요는 없습니다.
2. Portal Server 2가 Directory Server 1 대신 Directory Server 2를 가리키도록 구성합니다.

설명을 간략하게 하기 위해 이번 단계와 이후 단계에서 "Portal Server 2"는 Portal Server 2부터 Portal Server n까지를 의미합니다.

3. Portal Server 1을 업그레이드합니다.
 - a. 로드 밸런서 B에서 Portal Server 1을 비활성화합니다.
요청이 더 이상 Portal Server 1로 라우팅되지 않습니다.

- b. Directory Server MMR을 비활성화합니다.

Directory Server 2는 더 이상 Directory Server 1과 동기화되지 않습니다.

- c. Access Manager SDK 1B를 릴리스 5로 업그레이드합니다.

[281페이지의 "릴리스 4 Access Manager SDK 전용 업그레이드"](#)의 절차를 사용하십시오.

- d. Portal Server 1을 릴리스 5로 업그레이드합니다.

다음 내용에 주의하면서 [312페이지의 "릴리스 4 Portal Server 업그레이드"](#)에 설명된 대로 Portal Server 인스턴스를 업그레이드합니다.

- [317페이지의 "로드 밸런서의 구성 제거"](#)에 있는 사전 업그레이드 절차에 특히 주의하십시오.
- 업그레이드를 수행하기 전에
AccessManagerConfig-base/config/AMConfig.properties 파일의
`am.encrypted.pwd` 값이 연결된 원격 Access Manager 인스턴스의 로컬
Access Manager SDK와 동일한지 확인합니다.
- 업그레이드하는 각 Portal Server 인스턴스에 대해 `psupgrade`에서 요청하는 Portal Instance ID 매개 변수에 `null`이 아닌 고유한 값을 제공합니다.

Directory Server 1용 Portal Server 데이터가 릴리스 5로 업그레이드됩니다.

- e. 로드 밸런서 B에서 Portal Server 1을 활성화합니다.

요청이 다시 Portal Server 1로 라우팅됩니다.

- 4. Portal Server 2를 업그레이드합니다.

- a. 로드 밸런서 B에서 Portal Server 2를 비활성화합니다.

요청이 더 이상 Portal Server 2로 라우팅되지 않습니다.

- b. Directory Server 1을 가리키도록 Portal Server 2의 구성을 복원합니다.

- c. Access Manager SDK 2B를 릴리스 5로 업그레이드합니다.

[338페이지의 단계 c](#)에 설명된 절차를 따르십시오.

- d. Portal Server 2를 릴리스 5로 업그레이드합니다.

338페이지의 단계 d에 설명된 절차를 따르십시오.

- e. 로드 밸런서 B에서 Portal Server 2를 활성화합니다.

요청이 다시 Portal Server 2로 라우팅됩니다.

5. Directory Server MMR을 활성화합니다.

Directory Server 2용 Portal Server 데이터가 Directory Server 1과 동기화됩니다.

Java ES 릴리스 3에서 Portal Server 업그레이드

Java ES 2005Q1(릴리스 3) Portal Server를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Portal Server를 릴리스 5로 업그레이드하는 절차와 다음 예외를 제외하고 동일합니다.

- 릴리스 3 사전 업그레이드 작업: Portal Server 종속성 업그레이드
- 릴리스 3 업그레이드 Portal Server
- 여러 인스턴스 업그레이드

릴리스 3 사전 업그레이드 작업: Portal Server 종속성 업그레이드

Portal Server를 릴리스 3에서 업그레이드할 때 Access Manager 및 웹 컨테이너(Web Server 또는 Application Server)를 모두 릴리스 4 또는 릴리스 5로 업그레이드해야 Portal Server를 업그레이드할 수 있으며, 일부 종속성을 릴리스 3으로 유지하거나 일부 종속성을 릴리스 4로 업그레이드하고 다른 종속성을 릴리스 5로 업그레이드할 수는 없습니다([309페이지의 "선택적 업그레이드 문제"](#) 참조).

다음 종속성은 아래 순서에 따라 업그레이드해야 합니다.

1. **공유 구성 요소.** [63페이지의 2장](#), "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
2. **Directory Server.** [99페이지의 5장](#), "[Directory Server](#)"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **웹 컨테이너 소프트웨어.** Web Server 또는 Application Server의 업그레이드 지침은 [129페이지의 7장](#), "[Web Server](#)" 및 [199페이지의 11장](#), "[Application Server](#)"에서 각각 제공됩니다.

주 WebLogic 및 WebSphere의 웹 컨테이너와 같은 타사 웹 컨테이너를 업그레이드하는 경우 Portal Server를 지원하는 이러한 컨테이너에 대해 사용자 정의한 내용을 덮어 쓰게 되므로 Portal Server에 문제가 발생할 수 있습니다.

이와 같은 경우 업그레이드된 웹 컨테이너 환경에 맞도록 Portal Server를 다시 설치하고 재구성해야 합니다.

4. **Access Manager(Access Manager SDK).** 255페이지의 14장, "Access Manager"에서 Access Manager를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

릴리스 3 업그레이드 Portal Server

릴리스 3 Portal Server를 릴리스 5로 업그레이드하려면 311페이지의 "Java ES 릴리스 4에서 Portal Server 업그레이드"의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

릴리스 3 사후 업그레이드 작업

Portal Server를 릴리스 3에서 릴리스 5로 업그레이드할 때 326페이지의 "릴리스 4 사후 업그레이드 작업"에 설명된 사후 업그레이드 절차와 더불어 다음과 같은 문제를 해결하기 위한 사후 업그레이드 절차를 수행해야 합니다.

- 토론 가입

토론 가입

전역 디스플레이 프로필의 최상위 등록 정보를 편집하여 다음 문자열 등록 정보를 추가하지 않으면 커뮤니티에서 토론 가입에 실패합니다.

helpURL=en/desktop/usedesk.htm

다음 절차를 수행하십시오.

1. 디스플레이 프로필 XML 조각 파일 helpUrl.xml을 만듭니다.

```
<?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. 다음 명령을 사용하여 전역 디스플레이 프로필 등록 정보를 실행합니다.

```
./psadmin modify-dp -u amadminUser -f /tmp/passwordFile -p portal_ID
-m -g helpUrl.xml
```

여기서 -m 옵션은 전체 전역 디스플레이 프로필을 덮어 쓰지 않도록 하는 데 필요합니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Portal Server를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Portal Server 인스턴스를 실행하여 로드를 분산시킬 수 있습니다.

Portal Server의 인스턴스가 로드 균형 조정된 경우 아래 설명과 같이 서비스를 중단하지 않고도 Portal Server 인스턴스를 순서에 따라 업그레이드하는 롤링 업그레이드를 수행할 수 있습니다. 이 절차에서는 릴리스 3 Portal Server가 릴리스 5 Portal Server 디렉토리 데이터를 사용하여 작동하지 않는 제한 사항에 주의해야 합니다.

릴리스 3 Portal Server를 릴리스 5로 롤링 업그레이드하려면 [336페이지의 "여러 인스턴스 업그레이드"](#)에 설명된 것과 동일한 절차를 사용하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오. 또한 [337페이지의 단계 1](#)에서 설명한 대로 Access Manager도 업그레이드해야 합니다.

Java ES 릴리스 2에서 Portal Server 업그레이드

이 절에는 Java ES 2004Q2(릴리스 2) Portal Server를 릴리스 5로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 업그레이드 절차는 다음 절에서 설명하는 대로 몇 가지 변경 사항을 제외하고는 릴리스 4 Portal Server를 릴리스 5로 업그레이드하는 절차와 비슷합니다.

- [릴리스 2 사전 업그레이드 작업](#)
- [릴리스 2 업그레이드 Portal Server](#)
- [릴리스 2 사후 업그레이드 작업](#)
- [여러 인스턴스 업그레이드](#)

주 Linux 플랫폼에서 릴리스 2 Portal Server를 업그레이드하는 경우 Portal Server *와* 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Portal Server가 지원되지 않음). 자세한 내용은 [310페이지의 "이중 업그레이드"](#)를 참조하십시오.

릴리스 2 사전 업그레이드 작업

릴리스 2에서 Portal Server를 업그레이드하기 위한 사전 업그레이드 작업은 Portal Server 종속성 업그레이드를 제외하고 [313페이지의 "릴리스 4 사전 업그레이드 작업"](#)에 설명된 내용과 동일합니다.

Portal Server를 릴리스 2에서 업그레이드할 때 Access Manager 및 웹 컨테이너(Web Server 또는 Application Server)를 *모두* 릴리스 4 또는 릴리스 5로 업그레이드해야 Portal Server를 업그레이드할 수 있습니다. 하지만 종속성을 릴리스 2로 유지하거나 일부 종속성을 릴리스 4로 업그레이드하고 다른 종속성을 릴리스 5로 업그레이드할 수는 없습니다. 자세한 내용은 [309페이지의 "선택적 업그레이드 문제"](#)를 참조하십시오.

특히 웹 컨테이너 소프트웨어는 릴리스 2에서 업그레이드해야 하며 이는 psupgrade 스크립트를 실행하는 경우 [310페이지 표 15-4](#)의 시나리오 2와 시나리오 5만 지원된다는 것을 의미합니다.

다음 종속성은 아래 순서에 따라 업그레이드해야 합니다.

1. **공유 구성 요소.** [63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"](#)에서 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
2. **Directory Server.** [114페이지의 "Java ES 릴리스 2에서 Directory Server 업그레이드"](#)에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

3. **웹 컨테이너 소프트웨어.** Web Server 또는 Application Server의 업그레이드 지침은 [129페이지의 7장, "Web Server"](#) 및 [199페이지의 11장, "Application Server"](#)에서 각각 제공됩니다.

주	WebLogic 및 WebSphere의 웹 컨테이너와 같은 타사 웹 컨테이너를 업그레이드하는 경우 Portal Server를 지원하는 이러한 컨테이너에 대해 사용자 정의한 내용을 덮어 쓰게 되므로 Portal Server에 문제가 발생할 수 있습니다. 이와 같은 경우 업그레이드된 웹 컨테이너 환경에 맞도록 Portal Server를 다시 설치하고 재구성해야 합니다.

4. **Access Manager(Access Manager SDK).** [255페이지의 14장, "Access Manager"](#)에서 Access Manager를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.

릴리스 2 업그레이드 Portal Server

Portal Server를 릴리스 2에서 릴리스 5로 업그레이드하는 절차는 다음 절에서 설명하는 대로 Portal Server 소프트웨어가 배포되는 웹 컨테이너에 따라 다릅니다.

릴리스 2 Portal Server: Web Server 웹 컨테이너 업그레이드

Web Server 웹 컨테이너로 배포하는 경우 릴리스 2 Portal Server를 릴리스 5로 업그레이드하려면 릴리스 4을 참조하는 릴리스 2 대체를 제외하고 [311페이지의 "Java ES 릴리스 4에서 Portal Server 업그레이드"](#)에 있는 지침을 따릅니다.

그러나 Portal Server가 릴리스 5 Web Server(Web Server 7.0)에 배포된 경우 다음 단계를 수행한 다음 릴리스 2 Web Server를 업그레이드해야 합니다.

1. Web Server 관리 콘솔에 로그인합니다.
2. "가상 서버 편집" > "웹 응용 프로그램"을 누릅니다.
3. URI에 /portal 또는 /portalsamples가 포함된 배포된 웹 응용 프로그램을 모두 제거합니다.
4. "저장"을 누릅니다.
5. "배포 대기"를 누릅니다.

릴리스 2 Portal Server: Application Server 웹 컨테이너 업그레이드

릴리스 2 Portal Server를 릴리스 5로 업그레이드하는 경우 Application Server 웹 컨테이너로 배포하면 Application Server가 릴리스 2에서 릴리스 5로 업그레이드되었습니다.

Portal Server가 원래 배포된 릴리스 2 Application Server 인스턴스(*instanceName*)는 릴리스 5로 업그레이드할 때 Application Server 업그레이드 과정에서 만든 노드 에이전트로 마이그레이션됩니다. 이러한 업그레이드된 Application Server 인스턴스 내의 Portal Server를 업그레이드하려면 다음과 같은 단계가 필요합니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

자세한 내용은 [63페이지의 2장](#), "Java ES 공유 구성 요소 업그레이드"를 참조하십시오.

이 단계는 [346페이지의 단계 9](#)에서 psupgrade 스크립트를 실행하기 위한 필수 사전 요구 사항입니다.

3. 로컬에서 실행 중인 Portal Server Secure Remote Access Gateway, Rewriter 프록시 또는 Netlet 프록시의 인스턴스를 중지합니다.

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxyd stop
```

프로세스가 중지되었는지 확인합니다.

```
게이트웨이: netstat -an | grep 443
Rewriter 프록시: netstat -an | grep 10443
Netlet 프록시: netstat -an | grep 10555
```

4. Portal Server가 배포된 웹 컨테이너와 다른 웹 컨테이너에 Access Manager가 배포된 경우 실행되고 있는지 확인합니다.
5. 아직 실행되고 있지 않은 경우 Portal Server가 배포된 웹 컨테이너를 시작하여 Portal Server를 시작합니다.
 - a. 아직 시작하지 않은 경우 DAS(Domain Administration Server)를 시작합니다.

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password password domainName
```

- b. 해당 서버 인스턴스가 실행되고 있지 않은 경우 Portal Server가 배포된 업그레이드된 Application Server 인스턴스(*instanceName*)를 시작합니다.

다음과 같이 업그레이드된 Application Server 인스턴스가 마이그레이션된 노드 에이전트를 시작하여 이 작업을 수행합니다.

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

위 명령과 이어지는 단계에서 다음 규칙이 사용됩니다.

- 여기서 *nodeagentName*은 *hostName_domainName* 형식이지만 기본값은 간단히 *hostName*입니다.
- 기본 *domainName*은 domain1입니다.
- 기본 *instanceName*은 server1입니다.

6. Portal Server 구성 요소를 배포 해제합니다.

```
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. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다. 예를 들면 다음과 같습니다.

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. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

9. Java ES 릴리스 5 배포판에서 psupgrade 스크립트를 실행합니다.

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다.

주 실수로 잘못된 *os_arch* 디렉토리에서 *psupgrade*를 실행한 경우 다음과 같이 절차를 되돌려야 합니다.

- a. 올바른 *os_arch* 디렉토리로 변경합니다.
- b. Portal Server 데이터에 대한 변경을 되돌립니다.

```
./psupgrade rollback
```

요청된 매개 변수 및 비밀번호를 제공합니다.

- c. *psupgrade*를 다시 실행합니다.
-

psupgrade 스크립트는 Java ES 설치 프로그램을 호출하여 새 패키지를 설치한 다음 Portal Server 웹 응용 프로그램을 배포하는 웹 컨테이너의 위치, 포트 번호 및 기타 정보를 검색하기 위해 시스템을 쿼리합니다. 이 경우 시나리오 5인 웹 컨테이너 업그레이드 시나리오(310페이지 표 15-4 참조)에 따라 스크립트에서 Portal Server를 적절한 웹 컨테이너에 배포하는 데 필요한 정보를 입력해야 할 수도 있습니다.

315페이지 표 15-6에서는 릴리스 2 Application Server가 릴리스 5(시나리오 5)로 업그레이드될 때 필요한 정보를 보여 줍니다.

주 다시 돌아가서 변경할 수 없으므로 *psupgrade* 매개 변수에 정확한 값을 입력해야 하는데, *psupgrade* 스크립트에서 수정한 변경 사항을 롤백하기는 매우 어렵습니다. Portal Server 데이터에 대한 변경 사항을 되돌리려면 다음을 수행해야 합니다.

```
./psupgrade rollback
```

이 작업은 *psupgrade*를 다시 실행하기 전에 수행해야 합니다.

10. 345페이지의 단계 5에서 시작한 DAS(Domain Administration Server) 및 노드 에이전트를 중지합니다.

```
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. 단계 10에서 중지한 DAS(Domain Administration Server), 노드 에이전트 및 서버 인스턴스를 다시 시작합니다.

주 서버 인스턴스를 시작하기 전에 아래와 같이 `startinstances=false` 옵션을 사용하여 노드 에이전트를 별도로 시작하십시오.

```
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*의 기본값은 4848입니다.

릴리스 2 사후 업그레이드 작업

Portal Server를 릴리스 2에서 릴리스 5로 업그레이드할 때 [326페이지의 "릴리스 4 사후 업그레이드 작업"](#)에 설명된 사후 업그레이드 절차와 더불어 다음과 같은 문제를 해결하기 위한 사후 업그레이드 절차를 수행해야 합니다.

- 단일 사인 온 구성
- URLScrapper 채널 활성화
- 게이트웨이 서비스 항목 삭제

단일 사인 온 구성

릴리스 2에서 Portal Server를 업그레이드하면 메일, 달력 및 주소록과 같이 ssoadapter 메타 템플리트를 사용하는 포털 데스크탑 통신 채널에서 백엔드 서버에 액세스하지 못하게 됩니다.

예를 들어 릴리스 2 메일 ssoadapter 메타 템플리트인 SUN-UWC-MAIL을 Messaging Server에 해당하는 설정으로 수정한 경우 릴리스 5로 업그레이드하면 SUN-UWC-MAIL ssoadapter 메타 템플리트가 두 개 존재하게 됩니다. 이 경우 하나는 아직 변경되지 않은 릴리스 2 버전이고 다른 하나는 새로운 릴리스 5 버전입니다. Portal Server 콘솔 및 psadmin 명령줄 인터페이스에서 같은 이름을 사용하는 중복된 ssoadapter 메타 템플리트를 볼 수 있습니다.

ssoadapter 메타 템플리트를 사용하는 채널은 백엔드 서버와 연결하여 데이터를 검색할 수 없습니다.

이 문제를 해결하려면 ssoadapter 메타 템플릿 데이터를 검색하고 중복된 항목의 이름을 변경한 다음 수정된 데이터로 대체해야 합니다. 다음 절차를 수행하십시오.

1. ssoadapter 메타 템플릿 데이터를 내보냅니다.

amadmin 유틸리티를 사용하여 다음과 같이 Access Manager 서비스 데이터를 내보냅니다.

a. amadmin 요청 파일 /tmp/ssoadapter-template-gets.xml을 만듭니다.

이 파일은 ssoadapter 메타 템플릿 데이터를 검색하는 유틸리티에서 사용됩니다.

```
<?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. 다음 asadmin 명령을 실행합니다.

```
AccessManager-base/bin/amadmin -u amadminUser -w password
-t ssoadapter-templates-get.xml > /tmp/ssoadapter-templates.xml
```

명령의 출력은 /tmp/ssoadapter-templates.xml에 저장됩니다.

/tmp/ssoadapter-templates.xml 파일의 형식은 다음과 같습니다.

```
sunConfigurationTemplates=
[<ssoadapter meta-template1>, <ssoadapter meta-template2>, ...]
```

또한 각 <ssoadapter meta-template>의 구문은 다음과 같습니다.

```
default|imap:/?configName=SUN-UWC-MAIL
&proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D&subType=sun-one
&enableProxyAuth=false ...
```

2. /tmp/ssoadapter-templates.xml 파일을 수정하여 중복된 ssoadapter 메타 템플릿의 이름을 변경합니다.

- a. /tmp/ssoadapter-templates.xml 파일에서 각 템플릿을 찾습니다.

default|imap:/?configName= 문자열을 찾습니다.

- b. 중복된 ssoadapter 메타 템플릿 이름을 고유한 값으로 대체합니다.

예를 들어 SUN-UWC-MAIL ssoadapter 메타 템플릿이 두 개 있는 경우 이 중 하나에 대해 configName 값을 SUN-UWC-MAIL2로 바꾸면 다음과 같이 두 템플릿 모두 고유한 이름을 갖게 됩니다.

```
default|imap:/?configName=SUN-UWC-MAIL ...
default|imap:/?configName=SUN-UWC-MAIL2 ...
```

3. 수정된 ssoadapter 메타 템플릿을 가져오고 원래 데이터를 덮어 쓸 amadmin 요청 파일을 만듭니다.

- a. /tmp/ssoadapter-templates.xml를 /tmp/ssoadapter-new-templates.xml로 복사합니다.

- b. /tmp/ssoadapter-new-templates.xml에서 다음 문자열을 찾습니다.

```
sunConfigurationTemplates=[
```

위의 문자열을 다음으로 바꿉니다.

```
<?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. 모든 앰퍼샌드("&")를 "&"로 바꿉니다.

예를 들어 다음 줄의 경우

```
default|imap:/?configName=SUN-UWC-MAIL
&proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&subType=sun-one&enableProxyAuth=false...
```

다음과 같이 바꿉니다.

```
default|imap:/?configName=SUN-UWC-MAIL
&amp;proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&amp;subType=sun-one&amp;enableProxyAuth=false ...
```

- d. 각 ssoadapter 메타 템플릿 끝에 있는 쉼표(",")를 제거합니다.
- e. 각 ssoadapter 메타 템플릿을 시작 <Value> 태그와 종료 </Value> 태그 안에 넣습니다.

예를 들면 다음과 같습니다.

```
<Value>default|imap:/?configName=SUN-UWC-MAIL
&amp;proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&amp;subType=sun-one&amp;enableProxyAuth=false ...</Value>
```

- f. 마지막 ssoadapter 메타 템플릿에서 닫는 대괄호("]")를 제거합니다.
- g. 각 파일 끝에 다음 줄을 추가합니다.

```
</AttributeValuePair>
</ModifyDefaultValues>
</SchemaRequests>
</Requests>
```

위의 단계에서 사용한 단일 템플릿의 결과 ssoadapter-new-templates.xml 파일은 다음과 같습니다.

```
<?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
```

```
&proxyAdminPassword=%5BPROXY-ADMIN_PASSWORD%5D
&subType=sun-one&enableProxyAuth=false ...</Value>
</AttributeValuePair>
</ModifyDefaultValues>
</SchemaRequests>
</Requests>
```

4. 새 ssoadapter-new-templates.xml 파일을 가져옵니다.

```
AccessManager-base/bin/amadmin -u amadminUser -w password -v
-t ssoadapter-new-templates.xml
```

이제 Portal Server 콘솔에서 ssoadapter 탭에 액세스하여 업데이트된 ssoadapter를 볼 수 있습니다.

URLScrapper 채널 활성화

릴리스 3에서 릴리스 5로 업그레이드하는 경우 URLScrapper 채널을 활성화해야 합니다. [330페이지의 "URLScrapper 채널 활성화"](#)를 참조하십시오.

게이트웨이 서비스 항목 삭제

Portal Server를 릴리스 2에서 업그레이드하는 경우 amService-srapGateway 사용자 항목을 수동으로 삭제해야 합니다. 그렇지 않으면 Portal Server Secure Remote Access Gateway 구성 요소를 사용하는 경우 업그레이드한 다음 시작하지 못하게 됩니다. 다음 단계를 수행합니다.

1. Access Manager 콘솔에 로그인합니다.
2. 조직 DN에 있는 모든 사용자를 나열합니다.
3. amService-srapGateway 사용자를 삭제합니다.

여러 인스턴스 업그레이드

릴리스 2 Portal Server에서 릴리스 5로 업그레이드하면 여러 인스턴스의 롤링 업그레이드([336페이지의 "여러 인스턴스 업그레이드"](#) 참조)가 지원되지 않습니다.

임시 기능 릴리스 7.0에서 Portal Server 업그레이드

이 절에는 Portal Server를 임시 기능 릴리스(IFR) 7.0 2005Q4에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다.

이 절은 다음 내용으로 구성되어 있습니다.

- [Portal Server IFR 업그레이드 소개](#)
- [Portal Server IFR 7.0 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

Portal Server IFR 업그레이드 소개

Portal Server IFR 7.0을 릴리스 5 Portal Server로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- Portal Server IFR은 Application Server 7.x에서 지원되지 않습니다. 따라서 [310페이지 표 15-4](#)의 업그레이드 시나리오 5는 적용되지 않습니다.
- Portal Server IFR을 릴리스 5로 업그레이드하는 데 사용되는 psupgrade 스크립트는 릴리스 4에서 업그레이드하는 경우와 마찬가지로 새 패키지를 설치하지 않습니다. 그 대신 업그레이드 절차에서 다음 패치를 적용해야 합니다.

표 15-8 Portal Server IFR을 릴리스 5로 업그레이드하는 패치¹

설명	패치 아이디: Solaris 9 및 10	패치 아이디: Linux
Portal Server 7.1	121465-28(SPARC) 121466-28(x86)	121467-28
Portal Server 7.1 현지화	123254-02(SPARC) 124590-02(x86)	123255-02

1. 패치 개정판 번호는 Java ES 릴리스 5로 업그레이드하는 데 필요한 최소 버전입니다. 보다 최신 개정판이 있으면 표에 표시된 개정판 대신 최신 개정판을 사용하십시오.

- psupgrade 스크립트는 릴리스 5 Web Server에 대해 Portal Server IFR의 릴리스 5 업그레이드를 지원하지 않습니다. 이러한 제한에 두 가지 의미가 있습니다.
 - Web Server가 이미 릴리스 5로 업그레이드된 경우 이전 Web Server 컨테이너에 배포된 Portal Server IFR 소프트웨어를 업그레이드할 수 없습니다. 따라서 [310페이지 표 15-4](#)의 웹 컨테이너 업그레이드 시나리오 2는 지원되지 않습니다. [57페이지](#)의 "특별 사례"의 경고에 주의하십시오.
 - Web Server가 릴리스 5로 업그레이드되지 않은 경우, 아래의 "[Portal Server IFR 7.0 업그레이드](#)"에서 설명한 대로 먼저 Portal Server IFR 소프트웨어를 이전 Web Server (6.x) 컨테이너에서 릴리스 5로 업그레이드한 다음 Web Server(및 필요한 경우 Access Manager)를 릴리스 5로 업그레이드할 수 있습니다. 이러한 업그레이드 시나리오에 해당하는 경우 [364페이지](#)의 "[릴리스 5 Web Server \(7.0\) 웹 컨테이너에서 업그레이드](#)"에서 추가 사후 업그레이드 지침을 확인하십시오.

Portal Server IFR 7.0 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Portal Server를 IFR에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [IFR 7 사전 업그레이드 작업](#)
- [Portal Server IFR 7.0 업그레이드\(Solaris\)](#)
- [IFR 7 Portal Server 업그레이드\(Linux\)](#)
- [업그레이드 확인](#)
- [IFR 7 사후 업그레이드 작업](#)
- [업그레이드 롤백\(Solaris\)](#)
- [업그레이드 롤백\(Linux\)](#)

IFR 7 사전 업그레이드 작업

IFR 업그레이드의 사전 업그레이드 작업은 다음 예외를 제외하고 릴리스 4 업그레이드와 동일합니다([313페이지](#)의 "[릴리스 4 사전 업그레이드 작업](#)" 참조).

- [필요한 구성 정보 얻기](#)
- [공동 에이전트 컨테이너 구성](#)

필요한 구성 정보 얻기

315페이지의 "필수 구성 정보 및 비밀번호 얻기"에서 자세히 설명한 대로 psupgrade 스크립트에 필요한 일부 정보는 Portal Server IFR에서 업그레이드하는 경우 적용되지 않습니다. Portal Server IFR이 Application Server 7.x에서 지원되지 않으므로 315페이지 표 15-6의 웹 컨테이너 업그레이드 시나리오 5는 적용되지 않습니다.

공통 에이전트 컨테이너 구성

공통 에이전트 컨테이너는 Java ES 모니터링 및 관리 에이전트에 컨테이너 서비스를 제공하는 공유 구성 요소입니다. Portal Server 콘솔 및 psadmin 명령줄 인터페이스와 같은 Portal Server 관리 도구는 공통 에이전트 컨테이너에 배포된 통칭, 포털 관리 서버라고 하는 모니터링 및 관리 에이전트 집합을 사용합니다.

Portal Server IFR 7.0을 업그레이드하기 전에 Java ES 공유 구성 요소를 릴리스 5로 업그레이드한 경우 릴리스 5 Portal Server 콘솔에 로그인하고 psadmin 명령줄 인터페이스를 사용하려면 다음과 같은 추가 단계가 필요합니다. Portal Server IFR 7.0을 업그레이드하기 전에 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하지 않은 경우에는 다음 추가 단계를 무시하십시오.

1. 공통 에이전트 컨테이너를 재구성합니다.

```
PortalServer7-base/bin/psconfig --config
PortalServer7-base/samples/example2.xml
```

example2.xml 파일에 재구성 정보가 포함되어 있습니다. psconfig 명령을 실행하기 전에 필요한 비밀번호를 제공하도록 먼저 example2.xml 파일을 편집해야 합니다. 기본이 아닌 Portal Server 위치를 사용하는 경우 정확한 디렉토리도 제공해야 합니다.

2. 공통 에이전트 컨테이너를 참조하도록 웹 컨테이너의 클래스 경로를 편집합니다.

웹 컨테이너의 클래스 경로에는 이전 공통 에이전트 컨테이너 릴리스의 위치 참조(*rel4CAC-base-dir/lib/cacao_cacao.jar*)가 포함됩니다.

이 참조를 릴리스 5 위치(*rel5CAC-admin-dir/lib/cacao_cacao.jar*)로 바꿉니다.

3. 다음 등록 정보를 사용하여 *PortalServer7Config-base* 디렉토리에 등록 정보 파일 *pasconnect.properties*를 만들었는지 확인합니다.

```
pas.host=
```

등록 정보 값은 null, localhost 또는 실제 Portal Server 호스트 이름일 수 있습니다.

4. 공통 에이전트 컨테이너를 다시 시작합니다.

```
rel5CAC-admin-dir/bin/cacoadm start
```

Portal Server IFR 7.0 업그레이드(Solaris)

이 절에서는 Portal Server IFR의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 그 절차에 대해 설명합니다.

IFR 7 업그레이드 고려 사항(Solaris)

Portal Server IFR을 Java ES 릴리스 5로 업그레이드하는 경우 릴리스 4 업그레이드와 동일한 고려 사항이 적용됩니다([318페이지의 "업그레이드 고려 사항\(Solaris\)"](#) 참조).

또한 [353페이지의 "Portal Server IFR 업그레이드 소개"](#)에 나와 있는 문제점도 참조하십시오.

IFR 7 업그레이드 절차(Solaris)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server에 적용됩니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 로컬에서 실행 중인 Portal Server Secure Remote Access Gateway, Rewriter 프록시 또는 Netlet 프록시의 인스턴스를 중지합니다.

```
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
```

프로세스가 중지되었는지 확인합니다.

게이트웨이: `netstat -an | grep 443`

Rewriter 프록시: `netstat -an | grep 10443`

Netlet 프록시: `netstat -an | grep 10555`

3. Portal Server가 배포된 웹 컨테이너와 다른 웹 컨테이너에 Access Manager가 배포된 경우 실행되고 있는지 확인합니다.

4. 아직 실행되고 있지 않은 경우 Portal Server가 배포된 웹 컨테이너를 시작하여 Portal Server를 시작합니다.

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. 353페이지 표 15-8에 따라 필수 패치를 구합니다.

특정 개정판을 사용하도록 명시한 경우를 제외하고는 항상 최신 개정판을 사용하십시오.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

6. 적합한 Portal Server 패치를 적용하고 필요한 경우 표 15-8의 현지화 패치를 적용합니다.

```
patchadd /tmp/patch_ID
```

7. 패치가 성공적으로 업그레이드되었는지 확인합니다.

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

단계 6에서 적용된 패치 아이디의 버전이 출력으로 반환되어야 합니다.

8. 단계 6에서 현지화 패키지를 업그레이드한 경우 Portal Server 콘솔 JVM의 로캘을 UTF-8로 설정합니다.

```
export LC_ALL=ja_JP.UTF-8
export LANG=ja_JP.UTF-8
```

9. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다.

```
export ANT_HOME=/usr/sfw
export JAVA_HOME=/usr/jdk/entsys-j2se
```

10. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

11. psupgrade 스크립트를 실행합니다.

```
cd PortalServer7-base/bin
./psupgrade
```

psupgrade 스크립트는 Java ES 릴리스 5 배포에서 실행되지 않으며 Java ES 설치 프로그램을 호출하지도 않습니다(패키지는 이미 패치됨).

스크립트는 Portal Server 웹 응용 프로그램을 배포하는 웹 컨테이너의 위치, 포트 번호 및 기타 정보를 검색하기 위해 시스템을 쿼리합니다. 웹 컨테이너 업그레이드 시나리오([310페이지 표 15-4](#) 참조)에 따라 스크립트에서 Portal Server를 적절한 웹 컨테이너에 배포하는 데 필요한 추가 정보를 입력해야 할 수도 있습니다.

[315페이지 표 15-6](#)에는 [310페이지 표 15-4](#)의 여러 웹 컨테이너 업그레이드 시나리오에 필요한 정보가 나와 있습니다.

주	다시 돌아가서 변경할 수 없으므로 psupgrade 매개 변수에 정확한 값을 입력해야 하는데, psupgrade 스크립트에서 수정한 변경 사항을 롤백하기는 매우 어렵습니다.

12. 웹 컨테이너를 중지하고 다시 시작합니다.

모든 상황에 반드시 적용되는 것은 아니지만 웹 컨테이너를 다시 시작하면 Portal Server가 원래 상태로 다시 시작됩니다.

IFR 7 Portal Server 업그레이드(Linux)

이 절에서는 Portal Server의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

IFR 7 업그레이드 고려 사항(Linux)

Linux 플랫폼에서 Portal Server IFR 소프트웨어를 릴리스 5로 업그레이드하는 경우 Solaris 플랫폼에서 업그레이드할 때의 고려 사항을 참조하십시오([356페이지의 "IFR 7 업그레이드 고려 사항\(Solaris\)"](#) 참조). 단, 릴리스 5 패치를 Linux OS에 설치하면 이전 RPM이 제거됩니다.

IFR 7 업그레이드 절차(Linux)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server에 적용됩니다.

주의 Linux 에서 Portal Server IFR 을 릴리스 5 로 업그레이드하는 작업은 롤백할 수 없습니다. 다음 절차를 수행하기 전에 시스템을 백업하십시오.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 로컬에서 실행 중인 Portal Server Secure Remote Access Gateway, Rewriter 프록시 또는 Netlet 프록시의 인스턴스를 중지합니다.

```
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
```

프로세스가 중지되었는지 확인합니다.

게이트웨이: `netstat -an | grep 443`

Rewriter 프록시: `netstat -an | grep 10443`

Netlet 프록시: `netstat -an | grep 10555`

3. Portal Server가 배포된 웹 컨테이너와 다른 웹 컨테이너에 Access Manager가 배포된 경우 실행되고 있는지 확인합니다.
4. 아직 실행되고 있지 않은 경우 Portal Server가 배포된 웹 컨테이너를 시작하여 Portal Server를 시작합니다.

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. [353페이지 표 15-8](#)에서 패치 번호와 RPM 이름을 사용하여 필요한 패치를 가져옵니다.

특정 개정판을 사용하도록 명시한 경우를 제외하고는 항상 최신 개정판을 사용하십시오.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

6. 353페이지 표 15-8의 순서대로 Portal Server 패치를 적용하고 필요한 경우 Portal Server용 현지화 RPM을 적용합니다.

스크립트를 사용하여 패치의 RPM을 적용하는 방법을 설명하는 Portal Server 패치용 Readme 파일을 참조합니다.

```
cd /tmp
```

여기서 /tmp는 패치가 다운로드되는 디렉토리입니다.

```
./upgradeportalrpm
```

업데이트 스크립트가 RPM을 설치합니다.

현지화 패치의 경우 다음 명령을 사용하여 각 RPM을 설치합니다.

```
rpm -Fvh patchName-version.rpm
```

7. 패치가 성공적으로 업그레이드되었는지 확인합니다.

```
rpm -qa | grep sun-portal
```

RPM의 업그레이드 개정판 번호가 반환되어야 합니다.

8. 단계 6에서 현지화 패키지를 업그레이드한 경우 Portal Server 콘솔 JVM의 로캘을 UTF-8로 설정합니다.

```
export LC_ALL=ja_JP.UTF-8
```

```
export LANG=ja_JP.UTF-8
```

9. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다.

```
export ANT_HOME=/opt/sun
```

```
export JAVA_HOME=/usr/jdk/entsys-j2se
```

10. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

11. psupgrade 스크립트를 실행합니다.

```
cd PortalServer7-base/bin
```

```
./psupgrade
```

psupgrade 스크립트는 Java ES 릴리스 5 배포에서 실행되지 않으며 Java ES 설치 프로그램을 호출하지도 않습니다(패키지는 이미 패치됨).

스크립트는 Portal Server 웹 응용 프로그램을 배포하는 웹 컨테이너의 위치, 포트 번호 및 기타 정보를 검색하기 위해 시스템을 쿼리합니다. 웹 컨테이너 업그레이드 시나리오(310페이지 표 15-4 참조)에 따라 스크립트에서 Portal Server를 적절한 웹 컨테이너에 배포하는 데 필요한 추가 정보를 입력해야 할 수도 있습니다.

315페이지 표 15-6에는 310페이지 표 15-4의 여러 웹 컨테이너 업그레이드 시나리오에 필요한 정보가 나와 있습니다.

주	다시 돌아가서 변경할 수 없으므로 psupgrade 매개 변수에 정확한 값을 입력해야 하는데, psupgrade 스크립트에서 수정한 변경 사항을 롤백하기는 매우 어렵습니다.

12. 웹 컨테이너를 중지하고 다시 시작합니다.

모든 상황에 반드시 적용되는 것은 아니지만 웹 컨테이너를 다시 시작하면 Portal Server가 원래 상태로 다시 시작됩니다.

업그레이드 확인

다음 명령을 사용하여 Portal Server 패키지가 릴리스 5로 패치되었는지 확인할 수 있습니다.

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID
-f adminpasswordfile.
```

출력 값에 대해서는 313페이지 표 15-5를 참조하십시오.

전체 업그레이드를 확인하려면 포털 데스크탑이 실행되고 psadmin 관리 유틸리티가 설명된 대로 작동하는지 확인합니다.

또한 /var/sadm/install/logs에서 다음 업그레이드 로그 파일을 확인할 수 있습니다.

- Sun_Java_System_Portal_Server_upgrade.log
- Sun_Java_System_Portal_Server_upgrade.log_ant_xxx.log

여기서 xxx는 preupgrade, upgrade 또는 postupgrade일 수 있습니다.

IFR 7 사후 업그레이드 작업

Portal Server를 IFR 7에서 릴리스 5로 업그레이드하는 경우 다음과 같은 문제를 해결하는 데 필요한 사후 업그레이드 절차를 수행해야 합니다.

- [Java ES Monitoring Framework 활성화](#)
- [Application Server 웹 컨테이너에서 업그레이드](#)
- [릴리스 5 Web Server \(7.0\) 웹 컨테이너에서 업그레이드](#)
- [사용자 정의 포틀릿 응용 프로그램 재배포](#)
- [사용자 정의 포틀릿 응용 프로그램 마이그레이션](#)
- [책갈피 및 응용 프로그램 채널의 정확한 링크](#)
- [Struts 기반 포틀릿의 수동 마이그레이션](#)

Java ES Monitoring Framework 활성화

Java ES Monitoring Framework(MFWK 공유 구성 요소)를 사용하면 Portal Server 콘솔 및 psadmin 명령줄 유틸리티와 같은 Portal Server 관리 도구에서 방문자 수 또는 자주 사용하는 포털 등의 정보를 보고할 수 있습니다. MFWK를 사용하려면 다음을 수행하십시오.

1. 다음 두 파일을 찾습니다.

```
MFWK-base/template/jesmf/desktopmfwk.properties
MFWK-base/template/jesmf/com.sun.cmm.ps.xml
```

여기서 *MFWK-base*는 다음 경로입니다.

```
/opt/SUNWmfwk (Solaris)
```

```
/opt/sun/mfwk (Linux)
```

2. 두 파일을 다음 디렉토리로 복사합니다.

```
PortalServer7Data-base/portals/portal_ID/config/Portal_Instance/
```

3. desktopmfwk.properties 파일에서 다음을 찾습니다.

```
com.sun.portal.ProductCollectionId=%PS_DIR%
```

그리고 다음으로 바꿉니다.

```
com.sun.portal.ProductCollectionId=Portal_Installed_Location
```

4. 다음 두 jar 파일을 찾습니다.

```
MFWK-base/lib/mfwk_instrum_tk.jar
MFWK-base/lib/mfwk_agent.jar
```

그리고 적절한 웹 컨테이너 클래스 경로(Web Server의 경우 server.xml 파일 및 Application Server의 경우 domain.xml 파일)에 추가합니다.

5. 해당 웹 컨테이너를 다시 시작합니다.

Application Server 웹 컨테이너에서 업그레이드

Portal Server가 Application Server 웹 컨테이너에 배포된 경우 Portal Server를 성공적으로 재배포하려면 다음과 같은 추가 절차를 수행해야 합니다.

1. *AppServer8Config-base/domains/domainName/config/server.policy* 파일에서 psconsole 설정을 정의하는 부분을 찾습니다.
2. 찾은 섹션 끝에 다음 줄을 추가합니다.

```
permission java.lang.RuntimePermission "getProtectionDomain"
```

3. Application Server 인스턴스를 다시 시작합니다.

```
AppServer8-base/bin/asadmin start-domain --user admin_ID
--password passworddomainName
```

```
AppServer8-base/bin/asadmin start-node-agent --user admin_ID
--password password nodeagentName
```

여기서 *nodeagentName*은 *hostName_domainName* 형식이지만 기본값은 간단히 *hostName*입니다.

Application Server 웹 컨테이너에서 연결 중단

업그레이드된 Portal Server가 Application Server 웹 컨테이너에 배포되면 포털 응용 프로그램이 Java DB에 연결되기를 기다리며 중단될 수 있습니다. 이 문제를 해결하려면 다음 단계를 수행합니다.

1. *PortalServer7Data-base/derby/derby.properties*에서 다음 두 매개 변수 설정을 제거합니다.

```
derby.drda.maxThreads
derby.drda.timeslice
```

2. Java DB를 다시 시작합니다.

```
ANT_HOME/bin/ant
-DPS_CONFIG=PortalServer7Config-base/PSConfig.properties
-buildfile PortalServer7-base/lib/derby.xml
[stop-instance|start-instance]
```

여기서 *ANT_HOME*은 */usr/sfw*(Solaris) 및 */opt/sun*(Linux)입니다.

3. Application Server의 Java DB 구성 설정을 변경합니다.

Application Server 콘솔을 사용하여 communitymcPool, FileSharingDBPool, PointBasePool, SurveyDBPool 연결 풀 자원의 속성 값을 변경합니다.

속성 값을 다음과 같이 변경합니다.

Idle Timeout을 300 이상으로 설정

Resource Type을 javax.sql.ConnectionPoolDataSource으로 설정

Datasource classname을

org.apache.derby.jdbc.ClientConnectionPoolDataSource로 설정

4. Portal Server가 배포된 Application Server 인스턴스를 다시 시작합니다.

릴리스 5 Web Server (7.0) 웹 컨테이너에서 업그레이드

psupgrade 스크립트가 릴리스 5 Web Server에서 Portal Server IFR 업그레이드를 지원하지 않기 때문에(353페이지의 "Portal Server IFR 업그레이드 소개" 참조) 아직 릴리스 5로 업그레이드되지 않은 Web Server에서 Portal Server IFR을 업그레이드한 경우에는 다음과 같이 사후 업그레이드 단계를 추가로 수행해야 합니다.

1. Web Server 및 필요한 경우 Access Manager를 릴리스 5로 업그레이드합니다.
2. Portal Server 콘솔 및 psadmin 명령줄 유틸리티에 필요한 Web Server 컨테이너 값을 재구성합니다.

a. LDAP 브라우저를 엽니다.

구성 값은 Directory Server에 저장되어 있습니다.

b. DN 아래에서 다음 항목을 찾습니다.

sunPortalAdminPortalDomainID=defaultDomain

->sunPortalAdminPortalDomainPortalID=portal1

->sunPortalAdminPortalDomainPortalServerInstanceIn=host-port

c. 다음과 같이 편집합니다.

모든 항목 앞에는 sunPortalAdminPortalDomainPortalServerInstance 문
자열이 삽입되어 있습니다.

- WebContainerInstanceDir 항목을 삭제합니다.
- WebContainerDomainName 항목을 추가하고 315페이지 표 15-6에 있는 Web Container Config Name의 값을 할당합니다.
- InstallDir, WebContainerType, DocRoot 및 315페이지 표 15-6에 나와 있는 다른 매개 변수 등의 항목을 릴리스 5 Web Server (7.0) 값에 해당하도록 편집합니다.

3. 릴리스 5 Portal Server 인스턴스를 만듭니다.

```
PortalServer7-base/bin/psadmin create-instance newInstance_ID
```

`newInstance_ID` 값이 이미 있는 경우 오류가 발생하므로 아래의 **단계 4**를 수행하기 전에 이 단계를 수행하는 것이 좋습니다.

4. Portal Server IFR 인스턴스를 삭제합니다.

```
PortalServer7-base/bin/psadmin delete-instance oldInstance_ID
```

사용자 정의 포틀릿 응용 프로그램 재배포

사용자 정의 포틀릿 응용 프로그램을 만들고 배포한 경우에는 릴리스 5 Portal Server로 업그레이드한 다음 이러한 포틀릿을 수동으로 재배포해야 합니다. 디스플레이 프로파일 항목이 있고 채널 이름이 표시되더라도 사용자 정의 포틀릿을 재배포하기 전에는 내용이 보이지 않습니다.

다음 명령을 사용하여 포틀릿을 재배포합니다.

```
PortalServer7-base/bin/psadmin deploy-portlet
```

다음 위치에서 해당 .war 및 XML 파일을 찾아보면 재배포를 확인할 수 있습니다.

```
PortalServer7Data-base/portals/Upgraded/war
```

사용자 정의 포틀릿 응용 프로그램 마이그레이션

SJWC(Sun Java Web Console)에서 제공하는 사용자 인터페이스 프레임워크를 바탕으로 한 포틀릿 응용 프로그램은 수동으로 릴리스 5로 마이그레이션 및 재배포해야 합니다.

이러한 요구 사항은 특히 Portal Server와 함께 샘플 포틀릿 응용 프로그램으로 배포되어 포털로 사용자 정의 및 설치되는 4개의 웹 응용 프로그램인 filesharing, surveys, wiki, rssportlet에 해당됩니다. 업그레이드 중에 이러한 샘플 포틀릿 응용 프로그램의 버그 수정 버전이 포틀릿 응용 프로그램 영역에 배치됩니다. 용도에 따라 이러한 포틀릿 응용 프로그램을 사용자 정의한 경우 수동으로 릴리스 5로 마이그레이션한 다음 재배포해야 하며, Portal Server 업그레이드 프로세스에서는 처리되지 않습니다.

기본적으로 이러한 포틀릿 응용 프로그램의 일부(filesharing 및 surveys)는 커뮤니티를 만들 때 배포되어 사용자에게 제공됩니다.

다음 절차를 사용하여 SJWC 기반 포틀릿 응용 프로그램을 업그레이드, 사용자 정의 및 재배포할 수 있습니다. filesharing 포틀릿 응용 프로그램을 예로 들 수 있습니다.

1. 릴리스 5 SJWC jar 파일을 추출합니다.

- 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`

`PortalServer7-base/portlet/communityportlets.war`이 없는 경우
`PortalServer7-base/portlet/core/communityportlets.war`을 사용합니다.

d. 파일 중 하나의 이름을 변경합니다.

```
mv WEB-INF/lib/commons-collections-3.1.jar
WEB-INF/lib/commons-collections.jar
```

2. filesharing 포틀릿 응용 프로그램을 찾습니다.

```
cd PortalServer7Config-base/portals/portall/portletapps/filesharing
```

3. 업데이트된 SJWC 라이브러리를 추가합니다.

```
jar uvf src/filesharing.war.tokenized -C /tmp/lh WEB-INF
```

4. filesharing 포틀릿 응용 프로그램을 사용자 정의합니다.

```
ant customize
```

5. filesharing 포틀릿 응용 프로그램을 재배포합니다.

- a. `PortalServer7-base/bin/psadmin undeploy-portlet -u amadmin`
`-f passwordfile -p portal_id -i instance_id -g filesharing`
- b. `ant deploy`

- c. 다음 디렉토리(웹 컨테이너에 따라 다름)로 이동합니다.

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. sun-web.xml 파일을 열고 다음 줄을 마지막 줄 바로 앞(즉, sun-web-app 종료 태그 앞)에 추가합니다.

```
<class-loader delegate="false"/>
```

- e. filesharing/WEB-INF의 sun-web.xml 파일에 대해 단계 c 및 단계 d를 반복합니다.

6. surveys 및 SJWC 프레임워크를 바탕으로 한 다른 사용자 정의 포틀릿 응용 프로그램에 대해 단계 2에서 단계 5까지 반복합니다.

7. 웹 컨테이너를 다시 시작합니다.

책갈피 및 응용 프로그램 채널의 정확한 링크

릴리스 4 Portal Server를 릴리스 5로 업그레이드하면 책갈피와 응용 프로그램 채널에 중복된 링크 및 잘못된 링크가 생성됩니다. 이러한 링크 문제를 해결하려면 다음 절차를 수행하십시오.

1. PSConsole에 로그인합니다.
2. "일반적인 작업" 탭에서 "채널 및 컨테이너 관리"를 누릅니다.
3. DeveloperSample [Org]를 DN으로 선택하고 "확인"을 누릅니다.
4. "보기 유형"으로 JSPTabContainer [기본값]을 선택합니다.
5. MyFrontPageTabPanelContainer에서 "App Channel"을 누릅니다.

맨 오른쪽 프레임에 응용 프로그램 채널 등록 정보가 표시됩니다.

특정 로캘의 등록 정보를 보려면 "표 기본 설정"을 누르고 "로캘" 값(de, fr, es, ja, ko, zh, zh_CN, zh_TW)을 지정합니다.

6. userApps 등록 정보를 편집합니다.
 - a. userApps 등록 정보에서 "[값 편집...]" 링크를 누릅니다.
기존 응용 프로그램이 표시된 팝업 창이 나타납니다.
 - b. 목록에 다음 응용 프로그램을 추가합니다.
NetFile
 - c. "저장"을 누른 다음 "닫기"를 누릅니다.
7. 대상 등록 정보를 편집합니다.
 - a. 대상 등록 정보에서 "[값 편집...]" 링크를 누릅니다.
기존 대상이 표시된 팝업 창이 나타납니다.
 - b. 목록에 다음 대상을 추가합니다.
NetFile|/portal/NetFileApplet?Refer=java2
 - c. "저장"을 누른 다음 "닫기"를 누릅니다.

Struts 기반 포틀릿의 수동 마이그레이션

Struts 프레임워크를 바탕으로 한 코드를 사용하는 사용자 정의 포틀릿이 있는 경우 릴리스 5 Portal Server에 포함된 struts.jar 파일을 사용하려면 이러한 포틀릿을 업데이트해야 합니다. 다음 절차를 수행하십시오.

1. Struts 기반 포틀릿 응용 프로그램을 배포 해제합니다.
`PortalServer7-base/bin psadmin undeploy-portlet`
2. 정확한 버전의 struts.jar 파일로 .war 파일을 업데이트합니다.
`PortalServer7-base/lib/struts.jar` 을 `strutsbasedPortlet/WEB-INF/lib/struts.jar`로 복사합니다.

여기서 `strutsbasedPortlet`은 Struts 기반 포틀릿 파일이 있는 디렉토리입니다.
3. `strutsbasedPortlet` 디렉토리의 .war 아카이브를 만듭니다.
4. 포틀릿 응용 프로그램을 재배포합니다.
`PortalServer7-base/bin psadmin deploy-portlet`

업그레이드 롤백(Solaris)

이 절에서는 Portal Server의 업그레이드 롤백 절차에 영향을 미치는 고려 사항에 대해 설명하고 뒤이어 절차 자체에 대해 설명합니다.

롤백 고려 사항(Solaris)

릴리스 5 업그레이드를 롤백하는 절차는 *PortalServer7-base*에서 IFR 설치로 되돌리고 IFR 웹 응용 프로그램을 재배포하는 작업으로 구성됩니다.

롤백 절차(Solaris)

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. Directory Server를 업그레이드하기 전 상태로 복원합니다.

Directory Server backup/restore 명령줄 및 GUI 유틸리티를 사용합니다.

Sun Java System Directory Server Enterprise Edition 6.0 관리 설명서

(<http://docs.sun.com/doc/820-0292>)에서 Directory Server 백업 및 복원 장을 참조하십시오.

3. 릴리스 5로 업그레이드하는 동안 재배포된 릴리스 5 Portal Server 웹 응용 프로그램을 배포 해제합니다.

웹 컨테이너의 관리 유틸리티(명령줄 또는 콘솔)를 사용하여 다음 패키지를 배포 해제합니다.

```
portal
psconsole
search1
wsssoportlet
guessnumber
portletsamples
```

4. 해당 웹 컨테이너를 중지하여 Portal Server를 중지하십시오.

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. 표 15-8의 Portal Server 7.1 패치를 원래대로 되돌립니다.

```
patchrm patch_ID
```

6. 해당 웹 컨테이너를 다시 시작하여 Portal Server를 다시 시작하십시오.

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. 369페이지의 단계 3에서 배포 해제된 릴리스 5 Portal Server 웹 응용 프로그램을 배포합니다.

웹 컨테이너의 관리 유틸리티(명령줄 또는 콘솔)를 사용하여 다음 패키지를 배포합니다.

8. 웹 컨테이너를 중지하고 다시 시작합니다.

모든 상황에 반드시 적용되는 것은 아니지만 웹 컨테이너를 다시 시작하면 Portal Server가 원래 상태로 다시 시작됩니다.

업그레이드 롤백(Linux)

Linux에서는 업그레이드를 롤백할 수 없습니다.

하지만 업그레이드를 수행하기 전에 병렬 시스템을 만든 다음 해당 시스템을 업그레이드하여 테스트할 수 있습니다. 업그레이드를 롤백해야 하는 경우 해당 병렬 시스템으로 되돌릴 수 있습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Portal Server를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Portal Server 구성 요소를 실행하여 로드를 분산시킬 수 있습니다.

Portal Server의 인스턴스가 로드 균형 조정된 경우 서비스를 중단하지 않고도 Portal Server 인스턴스를 순서에 따라 업그레이드하는 롤링 업그레이드를 수행할 수 있습니다. 다른 인스턴스가 실행되는 동안 Portal Server의 각 인스턴스를 업그레이드할 수 있습니다. IFR에서 릴리스 5로 롤링 업그레이드하려면 [336페이지의 "여러 인스턴스 업그레이드"](#)에 설명된 것과 동일한 절차를 사용하십시오. 단, 릴리스 4로 참조된 부분은 IFR로 바꾸어 생각하십시오.

Portal Server Secure Remote Access

이 장에서는 Portal Server Secure Remote Access를 Java ES 5(릴리스 5): Sun Java System Portal Server Secure Remote Access 7.1로 업그레이드하는 방법을 설명합니다.

또한 릴리스 5에서 지원하는 다른 업그레이드 경로에 대한 업그레이드 고려 사항의 개요를 제공하며, Solaris와 Linux 운영 체제에서의 업그레이드에 대해서도 설명합니다.

- 375페이지의 "Portal Server Secure Remote Access 업그레이드 개요"
- 380페이지의 "Java ES 릴리스 4로부터 Portal Server Secure Remote Access 업그레이드"
- 399페이지의 "Java ES 릴리스 3에서 Portal Server Secure Remote Access 업그레이드"
- 401 페이지의 "Java ES 릴리스 2에서 Portal Server Secure Remote Access 업그레이드"
- 404페이지의 "임시 기능 릴리스 7.0에서 Portal Server Secure Remote Access 업그레이드"

주 이 장에 설명된 파일 위치는 *PortalServer6-base*와 *PortalServer6Config-base*(Portal Server 6.x) 및 *PortalServer7-base*와 *PortalServer7Config-base*(Portal Server 7.x)라는 디렉토리 경로에 대해 상대적으로 지정되어 있습니다. Portal Server를 처음 설치할 때 최소한 이 두 경로의 일부를 설치 디렉토리로 지정해야 합니다. 그렇지 않은 경우 Java ES 설치 프로그램에서 기본값을 할당합니다.

다음 표에서는 이러한 디렉토리 경로의 기본값을 보여 줍니다.

표 16-1 Portal Server Secure Remote Access 디렉토리 경로

경로 이름 변수	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 업그레이드 개요

이 절에서는 Java ES 5(릴리스 5)로 업그레이드하는 데 영향을 주는 다음과 같은 Portal Server Secure Remote Access의 일반 요소를 설명합니다.

- [Java ES 릴리스 5 Portal Server Secure Remote Access 정보](#)
- [Portal Server Secure Remote Access 업그레이드 로드맵](#)
- [Portal Server Secure Remote Access 데이터](#)
- [Portal Server Secure Remote Access 업그레이드 전략](#)

Java ES 릴리스 5 Portal Server Secure Remote Access 정보

Portal Server Secure Remote Access는 Gateway, Rewriter 프록시, Netlet 프록시 구성 요소로 이루어져 있으며 일반적으로 Portal Server를 호스팅하는 컴퓨터와는 다른 컴퓨터에 배포되지만 Portal Server와 밀접하게 연결되어 있습니다. Portal Server Secure Remote Access 구성 요소는 Portal Server와 동일한 관리 인프라를 사용하며 Portal Server를 호스팅하는 컴퓨터에 있는 서블릿 및 애플릿과 상호 작용합니다.

Java ES 릴리스 5 Portal Server Secure Remote Access는 릴리스 4와 비교할 때 여러 가지 새 기능과 향상된 기능이 포함된 주요 릴리스입니다. 이러한 변경 사항 중 상당수는 릴리스 4에 이어 릴리스된 임시 기능 릴리스(IFR)에서 변경된 것이므로 IFR과 비교했을 때 릴리스 5는 사소한 기능 변경만 포함되어 있습니다. IFR의 향상된 기능과 새 기능에 대한 자세한 내용은 Sun Java System Portal Server 7.1 릴리스 노트 (<http://docs.sun.com/app/docs/doc/820-0899/6ncjv9dm5?a=view>)를 참조하십시오. 특히 릴리스 4 명령줄 관리 인터페이스는 psadmin 명령으로 대체되었습니다.

Portal Server Secure Remote Access

업그레이드 로드맵

표 16-2는 Portal Server Secure Remote Access를 Java ES 릴리스 5로 업그레이드하는 방법을 보여 줍니다. 이 표는 Solaris 및 Linux 운영 체제 모두에 적용됩니다.

표 16-2 Java ES 5(릴리스 5) 업그레이드 경로: Portal Server Secure Remote Access 7.1

Java ES 릴리스	Portal Server Secure Remote Access 버전	일반적인 방법	재구성 필요
임시 기능 릴리스 (IFR)	Sun Java System Portal Server Secure Remote Access IFR 7.0 2005Q4	직접 업그레이드: 패치를 적용하고 업그레이드 스크립트를 사용하여 수행합니다.	없음
릴리스 4	Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q4	직접 업그레이드: 업그레이드 스크립트를 사용하여 수행합니다.	없음
릴리스 3	Sun Java System Portal Server Secure Remote Access 6.3.1 2005Q1	직접 업그레이드: 업그레이드 스크립트를 사용하여 수행합니다.	없음
릴리스 2	Sun Java System Portal Server Secure Remote Access 6.3 2004Q2	직접 업그레이드: 업그레이드 스크립트를 사용하여 수행합니다.	없음
릴리스 1	Sun ONE Portal Server Secure Remote Access 6.2(2003Q4)	직접 업그레이드 불가능: 그러나 먼저 릴리스 3으로 업그레이드한 다음 릴리스 3에서 릴리스 5로 업그레이드할 수 있습니다.	구성 데이터
이전 Java ES 릴리스		직접 업그레이드할 수 없습니다.	

Portal Server Secure Remote Access 데이터

다음 표는 Portal Server Secure Remote Access 소프트웨어의 업그레이드에 의해 영향을 받을 수 있는 데이터의 유형을 보여 줍니다.

표 16-3 Portal Server Secure Remote Access 데이터 사용

데이터 유형	위치	사용
구성 데이터	<i>PortalServer6Config-base/</i>	Portal Server Secure Remote Access 구성
디렉토리 스키마 서비스 구성 사용자 데이터	Directory Server	Portal Server Secure Remote Access는 포털 데스크탑 및 디렉토리에 저장된 사용자 프로필 데이터와 같은 서비스 구성에 따라 달라집니다.
동적 응용 프로그램 데이터	없음	Portal Server Secure Remote Access는 세션 상태와 같은 응용 프로그램 데이터를 지속적으로 저장하지 않습니다.

Portal Server Secure Remote Access 업그레이드 전략

Portal Server Secure Remote Access 업그레이드를 위한 전략은 일반적으로 [1장, "업그레이드 계획"](#)에서 설명한 업그레이드 경로, Java ES 구성 요소 간의 종속성, 선택적 업그레이드와 전체 업그레이드, 여러 인스턴스 배포 등과 같은 여러 고려 사항에 따라 달라집니다.

이 절에서는 Portal Server Secure Remote Access 업그레이드 계획에 영향을 줄 수 있는 문제를 통해 Portal Server Secure Remote Access에 대한 일반적인 내용을 설명합니다.

호환성 문제

릴리스 5 Portal Server Secure Remote Access는 Gateway, Rewriter 프록시 및 Netlet 프록시 구성 요소를 시작 및 중지하는 데 사용되는 psadmin 명령에서 공용 인터페이스가 변경되었습니다. Sun Java System Portal Server 7.1 Command-Line Reference (<http://docs.sun.com/doc/819-5030>)를 참조하십시오.

개별 Portal Server Secure Remote Access 구성 요소(Gateway, Rewriter 프록시 및 Netlet 프록시 포함)는 이전 버전과 호환되지 않으며 Java ES 릴리스 5에서 Portal Server와 함께 모두 동기화해야 합니다. 이러한 요구 사항은 로컬은 물론 분산된 Portal Server Secure Remote Access 구성 요소에도 모두 적용됩니다.

또한 릴리스 5 Portal Server 및 이전 Portal Server 버전에서 사용된 Directory Server 데이터 구조 간에는 호환되지 않습니다. 이러한 비호환성은 동일한 Directory Server 데이터를 사용하는 여러 Portal Server 인스턴스의 롤링 업그레이드에 영향을 줍니다.

Portal Server Secure Remote Access 종속성

Portal Server Secure Remote Access는 Portal Server와 밀접하게 연결되어 있으며 Portal Server가 포함된 소프트웨어 패키지에 종속되고 Portal Server와 같은 컴퓨터에서 실행됩니다.

하지만 Portal Server Secure Remote Access는 다른 Java ES 구성 요소에도 종속됩니다. 이러한 종속성은 Portal Server Secure Remote Access 소프트웨어를 업그레이드하고 재구성하는 절차에 영향을 줄 수 있습니다. 예를 들어, Portal Server Secure Remote Access 인터페이스 또는 기능의 변경 사항을 적용하려면 Portal Server Secure Remote Access가 종속되어 있는 구성 요소의 업그레이드된 버전이 필요할 수 있습니다. 그러한 구성 요소의 업그레이드 필요 여부는 특정 업그레이드 경로에 의해 결정됩니다.

Portal Server Secure Remote Access 구성 요소는 다음 Java ES 구성 요소에 종속됩니다.

- **공유 구성 요소.** Portal Server Secure Remote Access 구성 요소는 특정한 Java ES 공유 구성 요소(47페이지 표 1-9 참조)에 종속됩니다.
- **Portal Server.** Portal Server Secure Remote Access 구성 요소는 Portal Server Secure Remote Access 기능을 지원하는 데 필요한 로컬 구성 요소가 포함된 Portal Server에 대해 필수 종속성이 있습니다.
- **Access Manager(또는 Access Manager SDK).** Portal Server Secure Remote Access 구성 요소에는 단일 사인온 등 최종 사용자를 위한 인증 및 권한 부여 서비스를 제공하는 Access Manager가 반드시 필요합니다. Access Manager가 원격 컴퓨터에서 실행되는 경우 Access Manager SDK는 로컬에서 사용할 수 있어야 합니다.
- **Directory Server.** Portal Server Secure Remote Access는 사용자 데이터를 저장하는 데 사용하는 Directory Server에 대해 필수 종속성이 있습니다. 따라서, Portal Server Secure Remote Access를 업그레이드하려면 디렉토리 스키마를 확장해야 합니다.

선택적 업그레이드 문제

일반적으로 Java ES 릴리스 5는 컴퓨터에서 모든 구성 요소의 선택적 업그레이드를 지원하지만 Portal Server Secure Remote Access는 Portal Server와 밀접하게 연결되므로 Portal Server가 업그레이드되는 경우 Portal Server Secure Remote Access도 업그레이드해야 합니다. 비슷하게 Portal Server Secure Remote Access를 업그레이드하는 경우 Portal Server도 업그레이드해야 합니다.

결과적으로 Portal Server Secure Remote Access 업그레이드는 Portal Server의 제한과 동일한 제한에 묶이게 됩니다(Portal Server 309페이지의 "[선택적 업그레이드 문제](#)" 참조). 따라서 Portal Server Secure Remote Access 및 해당하는 모든 제품 구성 요소 종속성을 릴리스 5로 업그레이드하거나 Portal Server Secure Remote Access 및 Portal Server만 릴리스 5로 업그레이드하고 기타 제품 구성 요소 종속성을 4로 유지해야 합니다.

이중 업그레이드

Portal Server Secure Remote Access와 운영 체제를 모두 업그레이드하는 이중 업그레이드는 43페이지의 "[이중 업그레이드: Java ES 및 운영 체제 소프트웨어](#)"에 설명한 대로 기존 운영 체제 업그레이드 방식을 사용하여 수행할 수 있습니다.

1. 기존 Portal Server Secure Remote Access 데이터를 백업합니다.

필수 데이터의 위치는 377페이지의 "[Portal Server Secure Remote Access 데이터](#)"를 참조하십시오.

2. 운영 체제를 업그레이드합니다.

업그레이드해도 기존 파일 시스템은 그대로 남아 있습니다.

3. 릴리스 5 Portal Server Secure Remote Access로 업그레이드합니다.

업그레이드 방법에 따라 이 장에서 해당 절을 참조하십시오.

Java ES 릴리스 4로부터 Portal Server Secure Remote Access 업그레이드

이 절에는 Portal Server Secure Remote Access를 Java ES 2005Q4(릴리스 4)에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다.

이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [릴리스 4 Portal Server Secure Remote Access 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Java ES 릴리스 4 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

- **일반적인 업그레이드 방법.** 업그레이드 스크립트 `psupgrade`를 사용하여 업그레이드를 수행합니다. 이 스크립트는 이전 패키지를 제거하고 새 패키지를 설치하며 필요한 경우 구성 데이터를 마이그레이션합니다.
- **업그레이드 종속성.** Portal Server Secure Remote Access에는 여러 Java ES 공유 구성 요소에 대한 종속성이 있습니다([47페이지 표 1-9](#) 참조). 릴리스 5 Portal Server Secure Remote Access는 릴리스 4 버전의 이러한 공유 구성 요소와 호환되지만 Portal Server Secure Remote Access를 업그레이드하는 데 사용되는 `psupgrade` 스크립트가 릴리스 5 버전의 ANT 공유 구성 요소를 사용하므로 공유 구성 요소를 필수적으로 업그레이드해야 합니다.

또한 릴리스 5 Portal Server Secure Remote Access는 [378페이지의 "Portal Server Secure Remote Access 종속성"](#)에 설명한 대로 Portal Server, Access Manager 및 Directory Server에 대한 종속성이 있습니다. 이러한 종속성을 업그레이드하는 두 가지 방법이 지원됩니다([379페이지의 "선택적 업그레이드 문제"](#) 참조).

- **업그레이드 종속성.** Portal Server Secure Remote Access는 여러 Java ES 공유 구성 요소에 대해 종속성이 있으며([47페이지 표 1-9](#) 참조), 릴리스 5 Portal Server Secure Remote Access는 이러한 구성 요소의 릴리스 4 버전과 호환됩니다. 따라서 이러한 공유 구성 요소의 업그레이드는 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 경우에 선택 사항입니다.

하지만 릴리스 5 Portal Server Secure Remote Access 는 Portal Server에 대해서만 엄격한 업그레이드 종속성이 있습니다. 또한 릴리스 5 Portal Server Secure Remote Access는 [378페이지의 "Portal Server Secure Remote Access 종속성"](#)에 설명된 대로 Access Manager 및 Directory Server에 대해 엄격하지 않은 업그레이드 종속성이 있습니다.

이러한 제품 구성 요소 종속성을 업그레이드하는 두 가지 방법이 지원됩니다 ([379페이지의 "선택적 업그레이드 문제"](#) 참조).

- 릴리스 4 구성 요소로 모든 종속성 충족(*전혀* 릴리스 5로 업그레이드되지 않음, Portal Server 제외)
- 릴리스 5 구성 요소로 모든 종속성 충족(*모두* 릴리스 5로 업그레이드됨)

Portal Server Secure Remote Access에서 사용한 방법은 Portal Server에서 사용한 방법과 동일해야 합니다.

- **역방향 호환성.** 릴리스 5 Portal Server Secure Remote Access는 릴리스 4 버전과 호환됩니다.
- **업그레이드 롤백.** Portal Server Secure Remote Access의 릴리스 5 업그레이드를 릴리스 4로 롤백하려면 릴리스 4 패키지를 복원한 다음 릴리스 4 디렉토리 데이터를 복원해야 합니다.
- **플랫폼 문제.** Portal Server Secure Remote Access를 업그레이드하는 일반적인 방법은 Solaris와 Linux 운영 체제에서 동일하지만, Solaris OS의 경우 릴리스 5 Portal Server Secure Remote Access는 새 경로에 설치되고 Linux OS의 경우에는 릴리스 4와 같은 경로에 설치됩니다.

릴리스 4 Portal Server Secure Remote Access 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Portal Server Secure Remote Access를 Java ES 릴리스 4에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [릴리스 4 사전 업그레이드 작업](#)
- [릴리스 4 Portal Server Secure Remote Access 업그레이드\(Solaris\)](#)
- [릴리스 4 Portal Server Secure Remote Access 업그레이드\(Linux\)](#)
- [업그레이드 확인](#)

- 릴리스 4 사후 업그레이드 작업
- 업그레이드 롤백(Solaris)
- 업그레이드 롤백(Linux)

릴리스 4 사전 업그레이드 작업

Portal Server Secure Remote Access를 업그레이드하기 전에 다음 작업을 수행해야 합니다.

- 현재 버전 정보 확인
- Portal Server Secure Remote Access 종속성 업그레이드
- 릴리스 4 Portal Server Secure Remote Access 구성 정보 백업
- 로드 밸런서의 구성 제거
- Directory Proxy Server 구성 제거
- 필수 구성 정보 및 비밀번호 얻기

현재 버전 정보 확인

다음 명령을 사용하여 Portal Server Secure Remote Access의 현재 버전을 확인할 수 있습니다.

```
PortalServer6-base/bin/version
```

표 16-4 Portal Server Secure Remote Access 버전 확인 출력

Java ES 릴리스	Portal Server Secure Remote Access 버전 번호
릴리스 2	6.3
릴리스 3	6.3.1
릴리스 4	6.3.1 ¹
IFR 릴리스	7.0
릴리스 5	7.1

1. 릴리스 3과 릴리스 4의 유일한 차이점은 패치입니다. 릴리스 4 패치를 확인하려면 Solaris에서는 `showrev -p | grep patch_ID` 명령을 사용하고 Linux에서는 `rpm -qa sun-portal-core` 명령을 사용한 다음 Java ES 릴리스 4 업그레이드 설명서에 나와 있는 버전과 비교하면 됩니다.

Portal Server Secure Remote Access 종속성 업그레이드

일반적으로 컴퓨터 시스템 (및 컴퓨팅 환경 내)의 모든 Java ES 구성 요소를 Java ES 릴리스 5로 업그레이드하는 것이 좋습니다.

릴리스 5 Portal Server Secure Remote Access는 릴리스 4 버전의 Java ES 공유 구성 요소와 호환되지만 Portal Server Secure Remote Access를 업그레이드하는 데 사용되는 psupgrade 스크립트는 릴리스 5 버전의 ANT 공유 구성 요소를 사용하므로 공유 구성 요소를 필수적으로 업그레이드해야 합니다.

또한 Portal Server Secure Remote Access가 제대로 작동하려면 Portal Server를 업그레이드해야 합니다. 그러나 종속된 다른 Java ES 릴리스 4 제품 구성 요소를 업그레이드할 필요는 없습니다.

사실 종속성 업그레이드 방식은 Portal Server에 적용되는 방식과 동일합니다. 따라서 종속된 구성 요소 중 하나라도 릴리스 5로 업그레이드하면 모두 업그레이드해야 합니다(309페이지의 "선택적 업그레이드 문제" 참조). 그러나 Portal Server에 대한 Portal Server Secure Remote Access 종속성 때문에 Portal Server를 업그레이드하는 경우 공유 구성 요소를 제외한 Portal Server Secure Remote Access 종속성에 주의해야 합니다.

Portal Server Secure Remote Access 종속성을 릴리스 5로 업그레이드하는 경우 Portal Server Secure Remote Access를 업그레이드하기 전에 다음 순서대로 종속성을 업그레이드해야 하며, 이미 업그레이드된 항목은 건너뛸 수 있습니다.

1. **공유 구성 요소.** 63페이지의 "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5와 동기화하는 방법에 대한 지침을 제공합니다.
2. **Portal Server.** 303페이지의 15장, "Portal Server"에서 Portal Server를 업그레이드하는 방법에 대한 지침을 제공합니다.

릴리스 4 Portal Server Secure Remote Access 구성 정보 백업

Portal Server Secure Remote Access를 릴리스 5로 업그레이드하기 위해 Portal Server Secure Remote Access 소프트웨어를 재구성해야 하는 것은 아닙니다. 그러나 안전을 위한 조치로 psupgrade 스크립트에서 구성 정보가 저장되는 다음 디렉토리를 백업합니다.

PortalServer6Config-base/

로드 밸런서의 구성 제거

로드 밸런서를 통해 Portal Server Secure Remote Access 인스턴스에 액세스하는 경우 LOAD_BALANCER_URL 등록 정보 값으로 인해 이러한 액세스 작업이 Portal Server Secure Remote Access 업그레이드를 방해할 수 있습니다. 따라서 이 설정은 Portal Server Secure Remote Access 구성 요소를 업그레이드하기 전에 변경해야 합니다. LOAD_BALANCER_URL 등록 정보 설정을 수정하려면 다음을 수행합니다.

1. 다음 구성 파일 중 로컬에 있는 파일을 확인합니다(로컬로 설치된 Portal Server 구성 요소를 지원하는 파일 중 일부).

PortalServer6Config-base/PSConfig.properties (Portal Server가 로컬인 경우)
PortalServer6Config-base/GWConfig.properties (Gateway가 로컬인 경우)
PortalServer6Config-base/RWPConfig.properties (Rewriter 프록시가 로컬인 경우)
PortalServer6Config-base/NLPConfig.properties (Netlet 프록시가 로컬인 경우)

2. 이러한 구성 요소 파일에 LOAD_BALANCER_URL 등록 정보의 현재 값을 기록합니다.
3. LOAD_BALANCER_URL 등록 정보 값이 업그레이드되는 Portal Server Secure Remote Access 인스턴스를 가리키도록 수정합니다.

LOAD_BALANCER_URL=*hostName:port*/portal

4. 다음 구성 등록 정보가 있는 경우 로드 밸런서가 *아닌* 해당하는 Portal Server Secure Remote Access 구성 요소를 참조하도록 하십시오.

PortalServer6Config-base/platform.conf.default 파일의 경우

gateway.host=*Gateway_hostName*

PortalServer6Config-base/GWConfig.properties 및

PortalServer6Config-base/GWConfig-default.properties 파일의 경우

GW_HOST=*Gateway_hostName*

GW_IP=*Gateway_hostIP*

PortalServer6Config-base/RWPConfig.properties 및

PortalServer6Config-base/RWPConfig-default.properties 파일의 경우

RWP_HOST=*RewriterProxy_hostName*

RWP_IP=*RewriterProxy_hostIP*

PortalServer6Config-base/NLPConfig.properties 및

PortalServer6Config-base/NLPConfig-default.properties 파일의 경우

NLP_HOST=*NetLetProxy_hostName*

NLP_IP=*NetLetProxy_hostIP*

Directory Proxy Server 구성 제거

Portal Server Secure Remote Access 인스턴스가 Directory Proxy Server 인스턴스를 통해 Directory Server에 액세스하는 경우 업그레이드를 수행하기 전에 Directory Proxy Server 호스트 및 포트 번호 설정을 수정하고 업그레이드가 완료되면 원래 값으로 복원해야 합니다.

적절한 설정을 수정하려면 다음을 수행합니다.

1. 다음 Access Manager 구성 파일에 DS_HOST 및 DS_PORT 등록 정보의 현재 값을 기록합니다.

AccessManagerConfig-base/config/AMConfig.properties

2. DS_HOST 및 DS_PORT 등록 정보의 값을 해당 Directory Server 인스턴스를 직접 가리키도록 수정합니다.

필수 구성 정보 및 비밀번호 얻기

업그레이드 시나리오에 따라 psupgrade 스크립트에서 관리 계정에 대해 다음과 같은 정보를 입력해야 할 수도 있습니다.

- Directory Server 관리자 아이디 및 비밀번호
- Access Manager 관리자 아이디 및 비밀번호
- Directory Server amldapuser 아이디 및 비밀번호

릴리스 4 Portal Server Secure Remote Access 업그레이드(Solaris)

이 절에서는 Portal Server Secure Remote Access의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항(Solaris)

Portal Server Secure Remote Access 소프트웨어를 릴리스 5로 업그레이드하는 경우 다음 사항을 고려하십시오.

- Portal Server Secure Remote Access 소프트웨어는 여러 다른 역할을 수행하는 하위 구성 요소로 구성되지만 모두 함께 릴리스 5로 업그레이드해야 합니다.
 - **Portal-base.** 관리 Mbeans 및 해당 관리 소프트웨어, 로깅 프레임워크 및 모니터링 관련 소프트웨어가 포함되며 이들은 모두 함께 SUNWportal-base 패키지에 포함됩니다.
 - **Secure Remote Access 응용 프로그램.** Gateway, Rewriter 프록시 및 Netlet가 포함됩니다. 이러한 응용 프로그램은 일반적으로 Portal Server를 호스팅하는 컴퓨터가 아닌 한 대 이상의 다른 컴퓨터에 배포됩니다. Secure Remote Access 응용 프로그램에는 웹 컨테이너가 필요하지 않습니다.

- Gateway, Rewriter 프록시 및 Netlet 프록시를 같은 컴퓨터에 배포하지 않는 경우 Gateway를 업그레이드하기 전에 Rewriter 프록시 및 Netlet 프록시를 먼저 업그레이드해야 합니다.
- 모든 Portal Server Secure Remote Access 하위 구성 요소는 설치된 동일한 Portal Server Secure Remote Access 이미지에 해당하며 업그레이드되는 컴퓨터에 있는 경우 동시에 업그레이드됩니다.
- psupgrade 스크립트는 호스트 컴퓨터에 설치되어 있는 Portal Server Secure Remote Access 하위 구성 요소를 자동으로 검색하여 해당 구성 요소를 업그레이드합니다.

업그레이드 절차(Solaris)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server Secure Remote Access 구성 요소에 적용됩니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

자세한 내용은 [63페이지의 2장](#), "Java ES 공유 구성 요소 업그레이드"를 참조하십시오.

이 단계는 [387페이지의 단계 8](#)에서 psupgrade 스크립트를 실행하기 위한 필수 사전 요구 사항입니다.

3. 로컬에서 실행 중인 Gateway, Rewriter 프록시 또는 Netlet 프록시의 모든 인스턴스를 중지합니다.

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxyd stop
```

프로세스가 중지되었는지 확인합니다.

```
게이트웨이: netstat -an | grep 443
Rewriter 프록시: netstat -an | grep 10443
Netlet 프록시: netstat -an | grep 10555
```

4. Access Manager가 실행되고 있는지 확인합니다.
5. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다. 예를 들면 다음과 같습니다.

```
export ANT_HOME=/usr/sfw
export JAVA_HOME=/usr/jdk/entsys-j2se
```

6. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

7. 업그레이드하는 Portal Server Secure Remote Access 구성 요소가 Portal Server에서 원격인 경우 dpadmin 실행 파일을 Portal Server를 호스팅하는 컴퓨터에서 Portal Server Secure Remote Access 구성 요소를 호스팅하는 컴퓨터로 복사합니다.

dpadmin 실행 파일은 다음 위치에 있습니다.

PortalServer7-base/SUNWps.bak/bin/dpadmin(Portal Server를 업그레이드한 경우)

PortalServer6-base/bin/dpadmin(Portal Server가 업그레이드되지 않은 경우)

8. Java ES 릴리스 5 배포판에서 psupgrade 스크립트를 실행합니다.

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

여기서 *os_arch*는 Solaris_sparc과 같은 사용자의 플랫폼에 해당합니다.

psupgrade 스크립트는 Java ES 설치 프로그램을 실행하여 새 패키지를 설치하고 다음 정보를 요청합니다.

- Directory Server 관리자 아이디 및 비밀번호
- Access Manager 관리자 아이디 및 비밀번호
- Directory Server amldapuser 아이디 및 비밀번호

9. [386페이지의 단계 3](#)에서 중지시킨 Gateway, Rewriter 프록시 또는 Netlet 프록시의 인스턴스를 시작합니다.

```
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
```

위 명령이 실패하면 먼저 Portal Server Secure Remote Access 구성 요소를 등록(활성화)해야 합니다.

```
PortalServer7-base/bin/psadmin provision-sra -u amadminUser
-f passwordFile -p Portal_ID --gateway-profile profileName --enable
```

릴리스 4 Portal Server Secure Remote Access 업그레이드(Linux)

이 절에서는 Portal Server Secure Remote Access의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

업그레이드 고려 사항(Linux)

Linux 플랫폼에서 Portal Server Secure Remote Access 소프트웨어를 릴리스 5로 업그레이드하는 경우 Solaris 플랫폼에서 업그레이드할 때의 고려 사항을 참조하십시오([385페이지의 "업그레이드 고려 사항\(Solaris\)"](#) 참조). Linux OS의 릴리스 4와 동일한 경로에 릴리스 5 Portal Server Secure Remote Access가 설치된 경우는 제외됩니다. 결과적으로 psupgrade 스크립트는 릴리스 5 RPM을 설치할 때 이전 RPM을 제거합니다.

업그레이드 절차(Linux)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server Secure Remote Access에 적용됩니다.

주의 Linux에서 Java ES 릴리스 4를 릴리스 5로 업그레이드하는 작업은 롤백할 수 없습니다. 다음 절차를 수행하기 *전에* 시스템을 백업하십시오.

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. 아직 동기화하지 않은 경우 모든 공유 구성 요소를 릴리스 5로 동기화합니다.

자세한 내용은 [63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"](#)를 참조하십시오.

이 단계는 [389페이지의 단계 8](#)에서 psupgrade 스크립트를 실행하기 위한 필수 사전 요구 사항입니다.

3. 로컬에서 실행 중인 Gateway, Rewriter 프록시 또는 Netlet 프록시의 모든 인스턴스를 중지합니다.

```
PortalServer6-base/bin gateway stop
PortalServer6-base/bin netletd stop
PortalServer6-base/bin rwproxycd stop
```

프로세스가 중지되었는지 확인합니다.

게이트웨이: `netstat -an | grep 443`
 Rewriter 프록시: `netstat -an | grep 10443`
 Netlet 프록시: `netstat -an | grep 10555`

4. Access Manager가 실행되고 있는지 확인합니다.
5. `psupgrade` 스크립트에 필요한 두 환경 변수(`ANT_HOME` 및 `JAVA_HOME`)를 설정합니다. 예를 들면 다음과 같습니다.

```
export ANT_HOME=/opt/sun
export JAVA_HOME=/usr/jdk/entsys-j2se
```

6. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.
 지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.
7. 업그레이드하는 Portal Server Secure Remote Access 구성 요소가 Portal Server에서 원격인 경우 `dpadmin` 실행 파일을 Portal Server를 호스팅하는 컴퓨터에서 Portal Server Secure Remote Access 구성 요소를 호스팅하는 컴퓨터로 복사합니다.

`dpadmin` 실행 파일은 다음 위치에 있습니다.

PortalServer7-base/SUNWps.bak/bin/dpadmin(Portal Server를 업그레이드한 경우)

PortalServer6-base/bin/dpadmin(Portal Server가 업그레이드되지 않은 경우)

8. Java ES 릴리스 5 배포판에서 `psupgrade` 스크립트를 실행합니다.

```
cd os_arch/Products/portal_svr/Tools/upgrade/bin
./psupgrade
```

여기서 `os_arch`는 `Solaris_sparc`과 같은 사용자의 플랫폼에 해당합니다.

`psupgrade` 스크립트는 Java ES 설치 프로그램을 실행하여 새 패키지를 설치하고 다음 정보를 요청합니다.

- Directory Server 관리자 아이디 및 비밀번호
- Access Manager 관리자 아이디 및 비밀번호
- Directory Server `amldapuser` 아이디 및 비밀번호

업그레이드 확인

업그레이드하는 Portal Server Secure Remote Access 구성 요소가 Portal Server에서 원격인 경우 다음 파일에서 버전 정보를 확인하여 릴리스 5 패키지 설치를 확인할 수 있습니다.

```
PortalServer7-base/lib/PSversion.properties
```

업그레이드하는 Portal Server Secure Remote Access 구성 요소가 Portal Server와 같은 컴퓨터에 있는 경우에는 다음 명령을 사용하여 업그레이드를 확인할 수 있습니다.

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID  
-f adminpasswordfile.
```

출력 값에 대해서는 [382페이지 표 16-4](#)를 참조하십시오.

다음 위치에서 업그레이드 로그 파일을 확인할 수도 있습니다.

```
/var/sadm/install/logs/Sun_Java_System_Portal_Server_upagrede.log
```

릴리스 4 사후 업그레이드 작업

Portal Server Secure Remote Access를 릴리스 5로 업그레이드할 때는 사후 업그레이드 작업이 필요하지 않습니다. 단, 다음 상황은 제외됩니다.

- [로드 밸런서의 구성 복원](#)
- [Directory Proxy Server의 구성 복원](#)
- [릴리스 4 현지화된 공급자 삭제](#)

로드 밸런서의 구성 복원

Portal Server Secure Remote Access 인스턴스가 로드 밸런서를 통해 액세스된 경우 업그레이드한 다음 로드 밸런서 구성을 복원하려면 다음 단계를 수행해야 합니다.

1. *PortalServer7Config-base/platform.conf.default* 파일에서 다음 매개 변수를 설정합니다.

```
gateway.virtualhost=loadBalancer_hostName loadBalancer_hostIP  
gateway.external.ip=loadBalancer_hostIP  
gateway.dsame.agent=http\://loadBalancer_hostName\  
80/portal/RemoteConfigServlet
```

2. *PortalServer7Config-base/GWConfig-default.properties* 파일에서 다음 매개 변수를 설정합니다.

```
gateway.ipaddress=Gateway_hostIP
```

3. Rewriter 프록시 및 Netlet 프록시의 경우 [단계 1](#) 및 [단계 2](#)에 해당하는 매개 변수를 이러한 인스턴스가 Portal Server 호스트에서 원격 컴퓨터로 배포될 때 설정합니다.
4. Portal Server 및 로드 균형 조정된 Portal Server Secure Remote Access 인스턴스를 다시 시작합니다.

Directory Proxy Server의 구성 복원

Portal Server Secure Remote Access 인스턴스가 Directory Proxy Server 인스턴스를 통해 Directory Server에 액세스한 경우 업그레이드하기 전의 원래 값으로 Directory Proxy Server 호스트 및 포트 번호 설정을 복원해야 합니다. 업그레이드 준비 시 이러한 등록 정보 값을 수정한 경우 [385페이지의 "Directory Proxy Server 구성 제거"](#)를 참조하십시오.

릴리스 4 현지화된 공급자 삭제

다음과 같이 릴리스 4 현지화된 공급자를 삭제하기 전까지는 현지화된 Proxylet 서비스가 로드되지 않습니다.

1. `PortalServer7Data-base/portals/Upgraded/desktop` 디렉토리로 이동합니다.
2. `default_Locale`에서 다음을 제외한 모든 디렉토리 및 파일을 삭제합니다.
 - 사용자가 만든 파일 및 디렉토리 (Portal Server Secure Remote Access에서 제공하지 않음)

- `message.properties` 파일
- 다음 디렉토리

```
AddressBookProvider
BookmarkProvider
CalendarProvider
LoginProvider
LotusNotesAddressBookProvider
LotusNotesCalendarProvider
LotusNotesMailProvider
MSExchangeAddressBookProvider
MSExchangeCalendarProvider
MSExchangeMailProvidervMailProvider
NotesProvider
PersonalNoteProvider
Register
SampleRSS
SampleURLScrapper
SampleXML
```

```
TemplateEditContainerProvider
TemplateTabContainerProvider
URLScraperProvider
UWCAddressBookProvider
UserInfo
UserInfoProvider
XMLProvider
error
```

3. Portal Server가 배포된 웹 컨테이너를 다시 시작합니다.

업그레이드 롤백(Solaris)

이 절에서는 Portal Server Secure Remote Access의 업그레이드 롤백 절차에 영향을 미치는 고려 사항에 대해 설명하고 뒤이어 절차 자체에 대해 설명합니다.

롤백 고려 사항(Solaris)

릴리스 5 업그레이드를 롤백하는 절차는 *PortalServer6-base*에서 릴리스 4 설치로 되돌리는 작업으로 구성됩니다.

롤백 절차(Solaris)

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. Directory Server를 업그레이드하기 전 상태로 복원합니다.

Directory Server backup/restore 명령줄 및 GUI 유틸리티를 사용합니다.
Sun Java System Directory Server Enterprise Edition 6.0 관리 설명서
(<http://docs.sun.com/doc/820-0292>)에서 Directory Server 백업 및 복원 장을 참조하십시오.

3. 릴리스 5 Portal Server Secure Remote Access 패키지를 제거합니다.

- a. Java ES 제거 프로그램을 시작합니다.

```
/var/sadm/prod/SUNWentsys5/uninstall
```

- b. 설치한 모든 Portal Server Secure Remote Access 구성 요소를 선택합니다.
- c. 제거 선택을 확인합니다.
- d. Java ES 제거 프로그램을 종료합니다.

4. *PortalServer6-base* 및 *PortalServer6Config-base* 디렉토리를 원래 위치로 복원합니다.

이러한 디렉토리는 업그레이드 중에 .bak 확장자가 포함된 디렉토리로 이동됩니다.

업그레이드 롤백(Linux)

릴리스 5로 업그레이드하려면 릴리스 4 바이너리를 제거해야 하므로 Linux에서는 업그레이드를 롤백하기가 매우 어렵습니다.

롤백하는 한 가지 방법은 업그레이드를 수행하기 전에 병렬 시스템을 만들고 업그레이드를 수행하기 전에 해당 시스템을 테스트하는 것입니다. 업그레이드를 롤백해야 하는 경우 해당 병렬 시스템으로 되돌릴 수 있습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Gateway와 같은 Portal Server Secure Remote Access 구성 요소를 여러 컴퓨터 시스템에 배포하여 보안과 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Gateway 구성 요소를 실행하여 로드를 분산시킬 수 있습니다.

Gateway의 인스턴스가 로드 균형 조정된 경우 아래 설명과 같이 서비스를 중단하지 않고도 Gateway 인스턴스를 순서에 따라 업그레이드하는 롤링 업그레이드를 수행할 수 있습니다. 절차를 수행할 때 각각의 Portal Server Secure Remote Access 구성 요소가 이전 버전과 호환되지 않으므로 모두 Portal Server와 함께 Java ES 릴리스 5로 동기화되어야 한다는 제약 사항에 주의해야 합니다. 하지만 릴리스 4 Portal Server Secure Remote Access 인스턴스를 롤링 업그레이드하는 동안 Portal Server 인스턴스를 업그레이드할 수 있습니다.

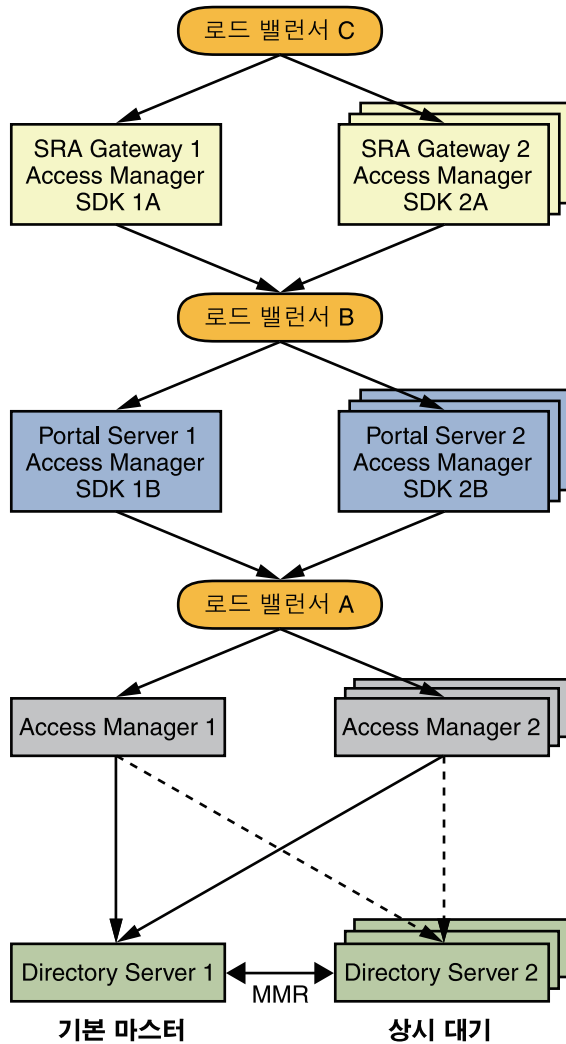
394페이지 그림 16-1의 배포 구조는 롤링 업그레이드 절차에 대해 설명하는 데 사용됩니다.

이 구조에서는 Portal Server Secure Remote Access Gateway 인스턴스를 통해 여러 Portal Server 인스턴스가 액세스됩니다. Portal Server 인스턴스 및 Gateway 인스턴스는 가용성과 확장성을 제공하기 위해 로드 균형 조정됩니다.

결과적으로 Portal Server 인스턴스는 로드 밸런서를 통해 Access Manager에 액세스합니다. Access Manager 및 Access Manager SDK 인스턴스는 다중 마스터 복제(MMR)를 위해 설정된 디렉토리에 액세스합니다. 다른 Directory Server 복제 방식도 가능하지만 MMR은 가용성이 뛰어나고 확장성이 우수한 대표적인 디렉토리 서비스입니다.

그림 16-1에서는 업그레이드 절차를 쉽게 설명할 수 있도록 여러 Gateway, Portal Server, Access Manager 및 Directory Server 인스턴스를 그룹화했습니다. Portal Server 예를 들어 2는 두 번째부터 n번째 Portal Server 인스턴스를 나타냅니다.

그림 16-1 여러 Portal Server 인스턴스를 위한 배포 구조 예



릴리스 4 Gateway(및 Portal Server)에서 릴리스 5로 롤링 업그레이드는 다음과 같이 수행됩니다.

1. 릴리스 4 Access Manager를 릴리스 5로 업그레이드하는 경우 [279페이지의 "여러 인스턴스 업그레이드"](#)에서 설명한 대로 롤링 업그레이드를 수행하십시오. 릴리스 4 Gateway 또는 릴리스 4 Portal Server를 릴리스 5로 업그레이드할 때 릴리스 4 Access Manager를 릴리스 5로 업그레이드할 필요는 없습니다.

2. 다음과 같이 Portal Server 및 Gateway 인스턴스의 구성을 수정합니다.
 - a. Portal Server 2가 Directory Server 1 대신 Directory Server 2를 가리키도록 구성합니다.
 설명을 간략하게 하기 위해 이번 단계와 이후 단계에서 "Portal Server 2"는 Portal Server 2부터 Portal Server n까지를 의미합니다.
 - b. Gateway 2가 Directory Server 1 대신 Directory Server 2를 가리키도록 구성합니다.
 설명을 간략하게 하기 위해 이번 단계와 이후 단계에서 "게이트웨이 2"는 게이트웨이 2부터 게이트웨이 n까지를 의미합니다.
 3. Portal Server 1을 업그레이드합니다.
 - a. 로드 밸런서 B에서 Portal Server 1을 비활성화합니다.
 요청이 더 이상 Portal Server 1로 라우팅되지 않습니다.
 - b. Directory Server MMR을 비활성화합니다.
 Directory Server 2는 더 이상 Directory Server 1과 동기화되지 않습니다.
 - c. Access Manager SDK 1B를 릴리스 5로 업그레이드합니다.
[281페이지의 "릴리스 4 Access Manager SDK 전용 업그레이드"](#)의 절차를 사용하십시오.
 - d. Portal Server 1을 릴리스 5로 업그레이드합니다.
 다음 내용에 주의하면서 [381페이지의 "릴리스 4 Portal Server Secure Remote Access 업그레이드"](#)에 설명된 대로 Portal Server 인스턴스를 업그레이드합니다.
 - [384페이지의 "로드 밸런서의 구성 제거"](#)에 있는 사전 업그레이드 절차에 특히 주의하십시오.
 - 업그레이드를 수행하기 전에 `AccessManagerConfig-base/config/AMConfig.properties` 파일의 `am.encrypted.pwd` 값이 연결된 원격 Access Manager 인스턴스의 로컬 Access Manager SDK와 동일한지 확인합니다.
 - 업그레이드하는 각 Portal Server 인스턴스에 대해 `psupgrade`에서 요청하는 Portal Instance ID 매개 변수에 null이 아닌 고유한 값을 제공합니다.
- Directory Server 1용 Portal Server 데이터가 릴리스 5로 업그레이드됩니다.

4. Gateway 1을 업그레이드합니다.
 - a. 로드 밸런서 C에서 Gateway 1을 비활성화합니다.

요청이 더 이상 Gateway 1로 라우팅되지 않습니다.
 - b. Access Manager SDK 1A를 릴리스 5로 업그레이드합니다.

[281페이지의 "릴리스 4 Access Manager SDK 전용 업그레이드"](#)의 절차를 사용하십시오.
 - c. Gateway 1을 릴리스 5로 업그레이드합니다.

다음 내용에 주의하면서 [381페이지의 "릴리스 4 Portal Server Secure Remote Access 업그레이드"](#)에 설명된 대로 Gateway를 업그레이드합니다.

 - [384페이지의 "로드 밸런서의 구성 제거"](#)에 있는 사전 업그레이드 절차에 특히 주의하십시오.
 - 업그레이드를 수행하기 전에 `AccessManagerConfig-base/config/AMConfig.properties` 파일의 `am.encrypted.pwd` 값이 연결된 원격 Access Manager 인스턴스의 로컬 Access Manager SDK와 동일한지 확인합니다.
5. 이전에 비활성화한 Portal Server 1 및 Gateway 1을 다음과 같이 해당 로드 밸런서에서 활성화합니다.
 - a. 로드 밸런서 B에서 Portal Server 1을 활성화합니다.

요청이 다시 Portal Server 1로 라우팅됩니다.
 - b. 로드 밸런서 C에서 Gateway 1을 활성화합니다.

요청이 다시 Gateway 1로 라우팅됩니다.
6. Portal Server 2 및 Gateway 2를 다음과 같이 해당 로드 밸런서에서 비활성화합니다.
 - a. 로드 밸런서 B에서 Portal Server 2를 비활성화합니다.

요청이 더 이상 Portal Server 2로 라우팅되지 않습니다.
 - b. 로드 밸런서 C에서 Gateway 2를 비활성화합니다.

요청이 더 이상 Gateway 2로 라우팅되지 않습니다.

7. Portal Server 2를 업그레이드합니다.
 - a. Directory Server 1을 가리키도록 Portal Server 2의 구성을 복원합니다.
 - b. Access Manager SDK 2B를 릴리스 5로 업그레이드합니다.
 281페이지의 "릴리스 4 Access Manager SDK 전용 업그레이드"의 절차를 사용하십시오.
 - c. Portal Server 2를 릴리스 5로 업그레이드합니다.
 Portal Server 1 업그레이드(395페이지의 단계 d)와 같은 절차를 사용합니다.
 - d. 로드 밸런서 B에서 Portal Server 2를 활성화합니다.
 요청이 다시 Portal Server 2로 라우팅됩니다.
8. Gateway 2를 업그레이드합니다.
 - a. Directory Server 1을 가리키도록 Gateway 2의 구성을 복원합니다.
 - b. Access Manager SDK 2A를 릴리스 5로 업그레이드합니다.
 281페이지의 "릴리스 4 Access Manager SDK 전용 업그레이드"의 절차를 사용하십시오.
 - c. Gateway 2를 릴리스 5로 업그레이드합니다.
 Gateway 1 업그레이드(396페이지의 단계 c)와 같은 절차를 사용합니다.
 - d. 로드 밸런서 C에서 Gateway 2를 활성화합니다.
 요청이 다시 Gateway 2로 라우팅됩니다.
9. Directory Server MMR을 활성화합니다.
 Directory Server 2용 Portal Server 데이터가 Directory Server 1과 동기화됩니다.

주 게이트웨이 구성 요소의 이전 릴리스가 활성 상태(위의 절차에 *해당되지 않는* 경우)로 유지되는 동안 Portal Server 인스턴스를 릴리스 5로 업그레이드하고, 로드 밸런서를 통해 게이트웨이 인스턴스에 액세스하는 롤링 업그레이드 시나리오에서는 *PortalServer6Config-base/GWConfig.properties* 파일과 *GWConfig-default.properties* 파일에 있는 다음 구성 등록 정보가 로드 밸런서가 *아닌* 게이트웨이를 참조하는지 모든 게이트웨이 인스턴스를 확인해야 합니다.

GW_IP=Gateway_hostIP
GW_HOST=Gateway_hostName

이러한 등록 정보가 로드 밸런서를 가리키면 Gateway가 업그레이드된 Portal Server 인스턴스에 더 이상 액세스할 수 없게 됩니다.

Java ES 릴리스 3에서 Portal Server Secure Remote Access 업그레이드

Java ES 2005Q1(릴리스 3) Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 절차와 다음 예외를 제외하고 동일합니다.

- [Portal Server Secure Remote Access 종속성 업그레이드](#)
- [릴리스 3 업그레이드 Portal Server Secure Remote Access](#)
- [여러 인스턴스 업그레이드](#)

Portal Server Secure Remote Access 종속성 업그레이드

Portal Server Secure Remote Access를 릴리스 3에서 업그레이드할 때 Access Manager를 릴리스 4 또는 릴리스 5로 업그레이드해야 Portal Server Secure Remote Access를 업그레이드할 수 있으며 일부 종속성을 릴리스 3으로 유지하거나 일부 종속성을 릴리스 4로 업그레이드하고 다른 종속성을 릴리스 5로 업그레이드할 수는 없습니다([379페이지의 "선택적 업그레이드 문제" 참조](#)).

다음 종속성은 아래 순서에 따라 업그레이드해야 합니다.

1. **공유 구성 요소.** [63페이지의 2장](#), "[Java ES 공유 구성 요소 업그레이드](#)"에서 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
2. **Directory Server.** [114페이지의 "Java ES 릴리스 2에서 Directory Server 업그레이드"](#)에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **Access Manager(Access Manager SDK).** [255페이지의 14장](#), "[Access Manager](#)"에서 Access Manager를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
4. **Portal Server.** [303페이지의 15장](#), "[Portal Server](#)"에서 Portal Server를 업그레이드하는 방법에 대한 지침을 제공합니다.

릴리스 3 업그레이드 Portal Server Secure Remote Access

릴리스 3 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하려면 [380페이지](#)의 "[Java ES 릴리스 4로부터 Portal Server Secure Remote Access 업그레이드](#)"의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 3으로 바꾸어 생각하십시오.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Gateway와 같은 Portal Server Secure Remote Access 구성 요소를 여러 컴퓨터 시스템에 배포하여 보안과 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Gateway 구성 요소를 실행하여 로드를 분산시킬 수 있습니다.

릴리스 3 Portal Server Secure Remote Access에서 여러 인스턴스 업그레이드를 수행할 때는 [393페이지](#)의 "[여러 인스턴스 업그레이드](#)"에 나와 있는 절차를 사용하십시오. 단 릴리스 4로 참조된 부분은 "릴리스 4"를 "릴리스 3"으로 바꾸어 생각하십시오. 또한 [394페이지](#)의 [단계 1](#)에서 설명한 대로 Access Manager도 업그레이드해야 합니다.

Java ES 릴리스 2에서 Portal Server Secure Remote Access 업그레이드

이 절에는 Java ES 2004Q2(릴리스 2) Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다. 업그레이드 절차는 다음 절에서 설명하는 대로 몇 가지 변경 사항을 제외하고는 릴리스 4 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 절차와 비슷합니다.

- [릴리스 2 사전 업그레이드 작업](#)
- [릴리스 2 업그레이드 Portal Server Secure Remote Access](#)
- [릴리스 2 사후 업그레이드 작업](#)
- [여러 인스턴스 업그레이드](#)

주

Linux 플랫폼에서 릴리스 2 Portal Server Secure Remote Access를 업그레이드하는 경우 Portal Server Secure Remote Access ~~와~~ 운영 체제를 모두 업그레이드하는 이중 업그레이드를 수행해야 합니다(RHEL 2.1에서는 릴리스 5 Portal Server Secure Remote Access가 지원되지 않음). 자세한 내용은 [379페이지](#)의 "[이중 업그레이드](#)"를 참조하십시오.

릴리스 2 사전 업그레이드 작업

릴리스 2에서 Portal Server Secure Remote Access를 업그레이드하기 위한 사전 업그레이드 작업은 다음 작업을 제외하고 [382페이지](#)의 "[릴리스 4 사전 업그레이드 작업](#)"에서 설명된 내용과 동일합니다.

- [Portal Server Secure Remote Access 종속성 업그레이드](#)
- [게이트웨이 서비스 항목 삭제](#)

Portal Server Secure Remote Access 종속성 업그레이드

Portal Server Secure Remote Access를 릴리스 2에서 업그레이드할 때 Access Manager를 릴리스 4 또는 릴리스 5로 업그레이드해야 Portal Server Secure Remote Access를 업그레이드할 수 있으며 일부 종속성을 릴리스 2로 유지하거나 일부 종속성을 릴리스 4로 업그레이드하고 다른 종속성을 릴리스 5로 업그레이드할 수는 없습니다([379페이지](#)의 "[선택적 업그레이드 문제](#)" 참조).

다음 종속성은 아래 순서에 따라 업그레이드해야 합니다.

1. **공유 구성 요소.** 63페이지의 2장, "Java ES 공유 구성 요소 업그레이드"에서 Java ES 공유 구성 요소를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다. 그러나 공유 구성 요소를 아직 업그레이드하지 않은 경우에는 psupgrade 스크립트에서 자동으로 업그레이드합니다.
2. **Directory Server.** 114페이지의 "Java ES 릴리스 2에서 Directory Server 업그레이드"에서 Directory Server를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
3. **Access Manager(Access Manager SDK).** 255페이지의 14장, "Access Manager"에서 Access Manager를 릴리스 5로 업그레이드하는 방법에 대한 지침을 제공합니다.
4. **Portal Server.** 303페이지의 15장, "Portal Server"에서 Portal Server를 업그레이드하는 방법에 대한 지침을 제공합니다.

릴리스 2 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하려면 380페이지의 "Java ES 릴리스 4로부터 Portal Server Secure Remote Access 업그레이드"의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바꾸어 생각하십시오.

게이트웨이 서비스 항목 삭제

Portal Server를 릴리스 2에서 업그레이드하는 경우 amService-srapGateway 사용자 항목을 수동으로 삭제해야 합니다. 그렇지 않으면 업그레이드한 후에 게이트웨이를 시작하지 못하게 됩니다. 다음 단계를 수행합니다.

1. Access Manager 콘솔에 로그인합니다.
2. 조직 DN에 있는 모든 사용자를 나열합니다.
3. amService-srapGateway 사용자를 삭제합니다.

릴리스 2 업그레이드 Portal Server Secure Remote Access

Java ES 2004Q2(릴리스 2) Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 절차는 릴리스 4 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하는 절차와 동일합니다.

릴리스 2 Portal Server Secure Remote Access를 릴리스 5로 업그레이드하려면 380페이지의 "Java ES 릴리스 4로부터 Portal Server Secure Remote Access 업그레이드"의 지침을 참조하십시오. 단, 릴리스 4로 참조된 부분은 릴리스 2로 바꾸어 생각하십시오.

릴리스 2 사후 업그레이드 작업

릴리스 2에서 업그레이드를 위한 사전 업그레이드 작업은 다음 작업을 제외하고 [390페이지](#)의 "릴리스 4 사후 업그레이드 작업"에서 설명된 내용과 동일합니다.

- [Proxylet 서비스용 Portal Server 도메인 설정](#)

Proxylet 서비스용 Portal Server 도메인 설정

릴리스 2 Portal Server Secure Remote Access를 릴리스 5로 업그레이드한 다음 올바른 Portal Server 도메인 값을 설정해야 합니다.

1. Portal Server 콘솔에 로그인하고 "Secure Remote Access"에서 "Proxylet" 탭으로 이동합니다.
2. Proxylet 서비스가 있는 조직의 고유 이름(DN)을 선택합니다.
3. Proxylet 규칙의 "도메인" 필드에서 `SERVER_DOMAIN`을 Portal Server가 설치된 도메인 이름으로 대체합니다.
4. 위의 단계를 Proxylet이 있는 모든 조직에 대해 반복합니다.

여러 인스턴스 업그레이드

릴리스 2 Portal Server Secure Remote Access 구성 요소(또는 Portal Server)를 릴리스 5로 업그레이드할 때는 여러 인스턴스 롤링 업그레이드([393페이지](#)의 "여러 인스턴스 업그레이드" 참조)가 지원되지 않습니다.

임시 기능 릴리스 7.0에서 Portal Server Secure Remote Access 업그레이드

이 절에는 Portal Server Secure Remote Access를 임시 기능 릴리스(IFR) 7.0 2005Q4에서 Java ES 5(릴리스 5)로 업그레이드하는 방법에 대한 정보가 포함되어 있습니다.

이 절은 다음 내용으로 구성되어 있습니다.

- [소개](#)
- [Portal Server Secure Remote Access IFR 7.0 업그레이드](#)
- [여러 인스턴스 업그레이드](#)

소개

Portal Server Secure Remote Access IFR 7.0을 릴리스 5로 업그레이드하는 경우 업그레이드 프로세스의 다음 측면을 고려해야 합니다.

Portal Server Secure Remote Access IFR을 릴리스 5로 업그레이드하는 데 사용되는 psupgrade 스크립트는 릴리스 4에서 업그레이드하는 경우와 마찬가지로 새 패키지를 설치하지 않습니다. 그 대신 업그레이드 절차에서 다음 패치를 적용해야 합니다.

표 16-5 Portal Server Secure Remote Access IFR을 릴리스 5로 업그레이드하는 패치¹

설명	패치 아이디: Solaris 9 및 10	패치 아이디: Linux
Portal Server 7.1	121465-10(SPARC) 121466-10(x86)	121467-10
Portal Server 7.1 현지화	123254-02(SPARC) 124590-02(x86)	123255-02

1. 패치 버전 번호는 Java ES 릴리스 5로 업그레이드하는 데 필요한 최소 버전입니다. 보다 최신 버전이 있으면 표에 표시된 버전 대신 최신 버전을 사용하십시오.

Portal Server Secure Remote Access IFR 7.0 업그레이드

이 절에서는 Solaris 및 Linux 플랫폼에서 Portal Server Secure Remote Access를 IFR에서 Java ES 릴리스 5로 업그레이드하는 방법에 대해 설명합니다. 항목에서 특정 플랫폼의 절차를 사용하는 경우 적용되는 운영 체제가 표시됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

- [사전 업그레이드 작업](#)
- [Portal Server Secure Remote Access IFR 7.0 업그레이드\(Solaris\)](#)
- [Portal Server Secure Remote Access IFR 7.0 업그레이드\(Linux\)](#)
- [업그레이드 확인](#)
- [사후 업그레이드 작업](#)
- [업그레이드 롤백\(Solaris\)](#)
- [업그레이드 롤백\(Linux\)](#)

사전 업그레이드 작업

IFR 업그레이드의 사전 업그레이드 작업은 릴리스 4 업그레이드와 동일합니다([382페이지의 "릴리스 4 사전 업그레이드 작업"](#) 참조).

Portal Server Secure Remote Access IFR 7.0 업그레이드(Solaris)

이 절에서는 Portal Server Secure Remote Access의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

IFR 7 업그레이드 고려 사항(Solaris)

Portal Server Secure Remote Access IFR을 릴리스 5로 업그레이드하는 경우 릴리스 4 업그레이드와 동일한 고려 사항이 적용됩니다([385페이지의 "업그레이드 고려 사항\(Solaris\)"](#) 참조).

IFR 7 업그레이드 절차(Solaris)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server Secure Remote Access에 적용됩니다.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 로컬에서 실행 중인 Gateway, Rewriter 프록시 또는 Netlet 프록시의 모든 인스턴스를 중지합니다.

```
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
```

프로세스가 중지되었는지 확인합니다.

게이트웨이: `netstat -an | grep 443`

Rewriter 프록시: `netstat -an | grep 10443`

Netlet 프록시: `netstat -an | grep 10555`

3. Access Manager가 실행되고 있는지 확인합니다.

4. [404페이지 표 16-5](#)에 따라 필수 패치를 구합니다.

특정 개정판을 사용하도록 명시한 경우를 제외하고는 항상 최신 개정판을 사용하십시오.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

5. 적합한 Portal Server 패치를 적용하고 필요한 경우 [표 16-5](#)의 현지화 패치를 적용합니다.

```
patchadd patch_ID
```

6. 패치가 성공적으로 업그레이드되었는지 확인합니다.

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

[단계 5](#)에서 적용된 패치 아이디의 버전이 출력으로 반환되어야 합니다.

7. [단계 5](#)에서 현지화 패키지를 업그레이드한 경우 Portal Server 콘솔 JVM의 로컬을 UTF-8로 설정합니다.

```
export LC_ALL=ja_JP.UTF-8
```

```
export LANG=ja_JP.UTF-8
```

8. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다.

```
export ANT_HOME=/usr/sfw
```

```
export JAVA_HOME=/usr/jdk/entsys-j2se
```

9. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

10. psupgrade 스크립트를 실행합니다.

```
cd PortalServer7-base/bin
./psupgrade
```

psupgrade 스크립트는 Java ES 릴리스 5 배포에서 실행되지 않으며 Java ES 설치 프로그램을 호출하지도 않습니다(패키지는 이미 패치됨).

Portal Server Secure Remote Access IFR 7.0 업그레이드(Linux)

이 절에서는 Portal Server Secure Remote Access의 업그레이드 절차에 영향을 미치는 고려 사항을 설명하고 뒤이어 절차 자체에 대해 설명합니다.

IFR 7 업그레이드 고려 사항(Linux)

Linux 플랫폼에서 Portal Server Secure Remote Access 소프트웨어를 릴리스 5로 업그레이드하는 경우 Solaris 플랫폼에서 업그레이드할 때의 고려 사항을 참조하십시오 (385페이지의 "[업그레이드 고려 사항\(Solaris\)](#)" 참조). 단, 릴리스 5 패치를 Linux OS에 설치하면 이전 RPM이 제거됩니다.

IFR 7 업그레이드 절차(Linux)

아래의 문서화된 절차는 업그레이드가 진행 중인 컴퓨터의 Portal Server Secure Remote Access에 적용됩니다.

주의	Linux에서 Portal Server Secure Remote Access IFR을 릴리스 5로 업그레이드하는 작업은 롤백할 수 없습니다. 다음 절차를 수행하기 <i>전에</i> 시스템을 백업하십시오.

1. 루트로 로그인하거나 슈퍼유저가 됩니다.

```
su -
```

2. 로컬에서 실행 중인 Gateway, Rewriter 프록시 또는 Netlet 프록시의 모든 인스턴스를 중지합니다.

```
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
```

프로세스가 중지되었는지 확인합니다.

게이트웨이: `netstat -an | grep 443`

Rewriter 프록시: `netstat -an | grep 10443`

Netlet 프록시: `netstat -an | grep 10555`

3. Access Manager가 실행되고 있는지 확인합니다.
4. [404페이지 표 16-5](#)에서 패치 번호와 RPM 이름을 사용하여 필요한 패치를 가져옵니다.

특정 개정판을 사용하도록 명시한 경우를 제외하고는 항상 최신 개정판을 사용하십시오.

<http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access>에서 /tmp로 패치를 다운로드할 수 있습니다.

5. [표 16-5](#)의 순서대로 Portal Server 패치를 적용하고 필요한 경우 Portal Server용 현지화 RPM을 적용합니다.

스크립트를 사용하여 패치의 RPM을 적용하는 방법을 설명하는 Portal Server 패치용 Readme 파일을 참조합니다.

```
cd /tmp
```

여기서 /tmp는 패치가 다운로드되는 디렉토리입니다.

```
./update
```

업데이트 스크립트가 RPM을 설치합니다.

현지화 패치의 경우 다음 명령을 사용하여 각 RPM을 설치합니다.

```
rpm -Fvh patchName-version.rpm
```

6. 패치가 성공적으로 업그레이드되었는지 확인합니다.

```
rpm -qa | grep sun-portal-core
```

RPM의 업그레이드 개정판 번호가 반환되어야 합니다.

7. **단계 5**에서 현지화 패키지를 업그레이드한 경우 Portal Server 콘솔 JVM의 로캘을 UTF-8로 설정합니다.

```
export LC_ALL=ja_JP.UTF-8
export LANG=ja_JP.UTF-8
```

8. psupgrade 스크립트에 필요한 두 환경 변수(ANT_HOME 및 JAVA_HOME)를 설정합니다.

```
export ANT_HOME=/opt/sun
export JAVA_HOME=/usr/jdk/entsys-j2se
```

9. 컴퓨터에 적합한 크기의 스왑 공간이 있는지 확인합니다.

지침에 따라 스왑 공간은 실제 RAM의 두 배 크기로 설정해야 합니다.

10. psupgrade 스크립트를 실행합니다.

```
cd PortalServer7-base/bin
./psupgrade
```

psupgrade 스크립트는 Java ES 릴리스 5 배포에서 실행되지 않으며 Java ES 설치 프로그램을 호출하지도 않습니다(패키지는 이미 패치됨).

업그레이드 확인

다음 명령을 사용하여 Portal Server Secure Remote Access 패키지가 릴리스 5로 패치되었는지 확인할 수 있습니다.

```
PortalServer7-base/bin/psadmin --version --adminuser admin_ID
-f adminpasswordfile.
```

출력 값에 대해서는 [382페이지 표 16-4](#)를 참조하십시오.

다음 위치에서 업그레이드 로그 파일을 확인할 수도 있습니다.

```
/var/sadm/install/logs/Sun_Java_System_Portal_Server_upagrede.log
```

사후 업그레이드 작업

Portal Server Secure Remote Access를 릴리스 5로 업그레이드할 때는 사후 업그레이드 작업이 필요하지 않습니다.

업그레이드 롤백(Solaris)

이 절에서는 Portal Server Secure Remote Access의 업그레이드 롤백 절차에 영향을 미치는 고려 사항에 대해 설명하고 뒤이어 절차 자체에 대해 설명합니다.

롤백 고려 사항(Solaris)

릴리스 5 업그레이드를 롤백하는 절차는 *PortalServer7-base*에서 IFR 설치로 되돌리는 작업으로 구성됩니다.

롤백 절차(Solaris)

1. 루트로 로그인하거나 수퍼유저가 됩니다.

```
su -
```

2. Directory Server를 업그레이드하기 전 상태로 복원합니다.

Directory Server backup/restore 명령줄 및 GUI 유틸리티를 사용합니다.
Sun Java System Directory Server Enterprise Edition 6.0 관리 설명서
(<http://docs.sun.com/doc/820-0292>)에서 Directory Server 백업 및 복원 장을 참조하십시오.

3. [404페이지 표 16-5](#)의 Portal Server 7.1 패치를 원래대로 되돌립니다.

```
patchrm patch_ID
```

업그레이드 롤백(Linux)

Linux 플랫폼에서 업그레이드를 롤백하는 방법은 없습니다.

여러 인스턴스 업그레이드

일부 배포 구조에서는 Portal Server Secure Remote Access를 여러 컴퓨터 시스템에 배포하여 확장성을 제공하고 가용성을 향상시킵니다. 예를 들어, 로드 밸런서가 있는 여러 컴퓨터에서 Portal Server Secure Remote Access 구성 요소를 실행하여 로드를 분산시킬 수 있습니다.

Portal Server Secure Remote Access의 인스턴스가 로드 균형 조정된 경우 서비스를 중단하지 않고도 Portal Server Secure Remote Access 인스턴스를 순서에 따라 업그레이드하는 롤링 업그레이드를 수행할 수 있습니다. 다른 인스턴스가 실행되는 동안 Portal Server Secure Remote Access의 각 인스턴스를 업그레이드할 수 있습니다. [405페이지의 "Portal Server Secure Remote Access IFR 7.0 업그레이드"](#)의 설명을 참조하여 각 인스턴스의 업그레이드를 수행하십시오.

IFR Portal Server Secure Remote Access에서 여러 인스턴스를 업그레이드할 때는 [393페이지의 "여러 인스턴스 업그레이드"](#)의 절차를 사용하십시오. 단, 릴리스 4로 참조된 부분은 "릴리스 4"를 "IFR"로 바꾸어 생각하십시오. 또한 [394페이지의 단계 1](#)에서 설명한 대로 Access Manager도 업그레이드해야 합니다.

Java Enterprise System 릴리스 내용

이 부록에서는 Java Enterprise System 릴리스에 대한 여러 가지 내용을 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

- [412페이지의 "Java ES 2003Q4\(릴리스 1\)"](#)
- [414페이지의 "Java ES 2004Q2\(릴리스 2\)"](#)
- [417페이지의 "Java ES 2005Q1\(릴리스 3\)"](#)
- [422페이지의 "Java ES 2005Q4\(릴리스 4\)"](#)
- [427페이지의 "Java ES 5\(릴리스 5\)"](#)

Java ES 2003Q4(릴리스 1)

이 절에서는 Java Enterprise System 2003Q4의 내용을 나열합니다.

릴리스 1 설치 프로그램-선택 가능한 구성 요소

Sun ONE(Open Network Environment)과 Sun Cluster 구성 요소 제품은 분산된 엔터프라이즈 응용 프로그램을 지원하는 데 필요한 인프라 서비스를 제공합니다. 각 구성 요소 제품은 다음과 같습니다.

- Sun Cluster 3.1 및 Sun ONE용 Sun Cluster 에이전트
- Sun ONE Administration Server 5.2
- Sun ONE Application Server 7, 업데이트 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 서비스 팩 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

릴리스 1 공유 구성 요소

공유 구성 요소는 구성 요소 제품에 필요한 로컬 서비스 및 기술 지원을 제공합니다. 구성 요소 제품을 설치하면 Java ES 설치 프로그램이 아직 설치되지 않은 필수 공유 구성 요소를 자동으로 설치합니다.

Java Enterprise System에는 다음 공유 구성 요소가 포함되어 있습니다.

- ANT(Jakarta ANT Java/XML 기반 빌드 도구)
- 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)

주 Perl은 Application Server 및 Directory Server에 필요하지만 Java ES 공유 구성 요소로 자동 설치되지 않습니다.

Java ES 2004Q2(릴리스 2)

이 절에서는 Java Enterprise System 2004Q2의 내용을 나열합니다.

릴리스 2 설치 프로그램-선택 가능한 구성 요소

구성 요소 제품은 분산된 엔터프라이즈 응용 프로그램을 지원하기 위해 필요한 인프라 서비스를 제공합니다. Java Enterprise System을 특정 호스트에 설치하는 경우 전체 배포 구조에 따라 호스트에 설치할 구성 요소 제품을 선택하십시오.

Java Enterprise System 2004Q2에는 다음 구성 요소 제품이 포함되어 있습니다.

통신 및 공동 작업 서비스

- 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

웹 및 응용 프로그램 서비스

- Sun Java System Application Server 7.0 업데이트 3(Standard 및 Platform Edition)
- Sun Java System Web Server 6 2004Q2 업데이트 1 서비스 팩 2
- Sun Java System Message Queue 3.5 SP1(Platform 및 Enterprise Editions)

디렉토리 및 Identity 서비스

- Sun Java System Identity Server 6.2 2004Q2
Sun Java System Communications Services 6 2004Q2 사용자 관리 유틸리티 포함
- Sun Java System Directory Server 5 2004Q2
- Sun Java System Directory Proxy Server 5 2004Q2

가용성 서비스

- Sun Cluster 3.1 4/04 및 Sun Java System용 Sun Cluster 에이전트

관리 서비스

- Sun Java System Administration Server 5 2004Q2
- Sun Remote Services Net Connect 3.5

Linux OS에서는 Sun Cluster, Sun Cluster 에이전트 및 Sun Remote Services Net Connect를 사용할 수 없습니다.

릴리스 2 공유 구성 요소

공유 구성 요소는 구성 요소 제품에 필요한 로컬 서비스 및 기술 지원을 제공합니다. 구성 요소 제품을 설치하면 Java ES 설치 프로그램이 아직 설치되지 않은 필수 공유 구성 요소를 자동으로 설치합니다.

Java Enterprise System에는 다음과 같은 공유 구성 요소가 있습니다.

- Ant(Jakarta ANT Java/XML 기반 빌드 도구)
- 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(NSPERL 포함)
- 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(릴리스 3)

이 절에서는 Java Enterprise System 2005Q1의 내용을 나열합니다.

릴리스 3 설치 프로그램에서 선택 가능한 구성 요소

Java ES 설치 프로그램의 구성 요소 선택 페이지에 선택 가능한 구성 요소가 제공되는 서비스별로 그룹화되어 있습니다. 각 구성 요소와 함께 설치되는 하위 구성 요소는 다음과 같습니다.

통신 및 공동 작업 서비스

- 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(서버 및 멀티플렉서 소프트웨어 포함)
 - Instant Messaging 자원
 - Access Manager Instant Messaging 서비스
- Sun Java System Portal Server 6 2005Q1
- Sun Java System Portal Server Secure Remote Access 6 2005Q1
 - Secure Remote Access Core
 - 게이트웨이
 - Netlet 프록시
 - Rewriter 프록시
- Sun Java System Communications Express 2005Q1
- Sun Java System 디렉토리 준비 도구

웹 및 응용 프로그램 서비스

- Sun Java System Application Server Enterprise Edition 8.1 2005Q1

- Domain Administration Server
- Application Server 노드 에이전트
- 명령줄 관리 도구
- 로드 균형 조정 플러그인

Web Server 또는 Apache Web Server와 함께 사용할 수 있으며 구성에서 선택할 수 있습니다. 기본값은 Web Server입니다.

- PointBase
- 샘플 응용 프로그램
- Sun Java System Web Server 6 2005Q1 업데이트 1 서비스 팩 4
- Sun Java System Message Queue 3 2005Q1

디렉토리 및 Identity 서비스

- Sun Java System Access Manager 6.3 2005Q1

Portal Server 및 Messaging Server용 Delegated Administrator 지정 도구는 Access Manager와 함께 자동으로 설치됩니다.

- Identity Management 및 Policy Services Core(Delegated Administrator 유틸리티 포함)
- Access Manager 관리 콘솔
- 연합 관리를 위한 공통 도메인 서비스
- Access Manager SDK
- Sun Java System Directory Server 5 2005Q1
- Sun Java System Directory Proxy Server 5 2005Q1

가용성 서비스

- Sun Cluster 3.1 9/04
 - Sun Cluster 핵심
- Sun Java System용 Sun Cluster 에이전트
 - 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(고가용성 세션 저장소에 사용)

관리 서비스

- Sun Java System Administration Server 5 2005Q1
- SunSM Remote Services Net Connect 3.1.1

주	Sun Cluster, Sun Cluster 에이전트와 Sun Remote Services Net Connect는 Solaris 10 또는 Linux 운영 체제에서 사용할 수 없습니다. Sun Remote Services Net Connect는 Solaris x86 플랫폼에서 사용할 수 없습니다.

릴리스 3 공유 구성 요소

공유 구성 요소는 선택 가능한 구성 요소에 대한 로컬 서비스 및 기술 지원을 제공합니다. Java ES 구성 요소를 설치하면 설치 프로그램이 아직 설치되지 않은 필수 공유 구성 요소를 자동으로 설치합니다.

이 Java ES 릴리스에는 다음 구성 요소가 포함되어 있습니다.

- Ant(Jakarta ANT Java/XML 기반 빌드 도구)
- Apache SOAP(Simple Object Access Protocol) Runtime
- Berkeley 데이터베이스
- 공통 에이전트 컨테이너
- 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(NSPERL 포함)

- SAAJ(SOAP with Attachments API for Java)
- SAML(Security Assertions Markup Language)
- SASL(Simple Authentication and Security Layer)
- SNMP(Simple Network Management Protocol) Peer
- Sun Explorer Data Collector(Solaris 전용)
- Sun Java Monitoring Framework
- Sun Java Web Console
- Tomcat Servlet JSP Container
- XML C Library(libxml)
- WSCL(Web Services Common Library)

Java ES 2005Q4(릴리스 4)

이 절에서는 Java Enterprise System 2005Q4의 내용을 나열합니다.

릴리스 4 설치 프로그램-선택 가능한 구성 요소

Java ES 설치 프로그램의 구성 요소 선택 페이지에 선택 가능한 구성 요소가 제공되는 서비스별로 그룹화되어 있습니다. 각 구성 요소와 함께 설치되는 하위 구성 요소는 다음과 같습니다.

통신 및 공동 작업 서비스

- 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(서버 및 멀티플렉서 소프트웨어 포함)
 - Instant Messaging 자원
 - Access Manager Instant Messaging 서비스
- 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
 - 게이트웨이
 - Netlet 프록시
 - Rewriter 프록시
- 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 콘솔 및 유틸리티
 - Delegated Administrator 서버

웹 및 응용 프로그램 서비스

- Sun Java System Application Server Enterprise Edition 8.1 2005Q4

- Domain Administration Server
- Application Server 노드 에이전트
- 명령줄 관리 도구
- 로드 균형 조정 플러그인

Web Server 또는 Apache Web Server와 함께 사용할 수 있으며 구성에서 선택할 수 있습니다. 기본값은 Web Server입니다.

- PointBase 데이터베이스
- 샘플 응용 프로그램

- Sun Java System Web Server 6.1 서비스 팩 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

디렉토리 및 Identity 서비스

- Sun Java System Access Manager 7.0 2005Q4
 - Identity Management 및 Policy Services Core
 - Access Manager 관리 콘솔
 - 연합 관리를 위한 공통 도메인 서비스
 - Access Manager SDK
- Sun Java System Directory Server 5.2 2005Q4
- Sun Java System Directory Proxy Server 5.2 2005Q4

가용성 서비스

- Sun Cluster 3.1 8/05
 - Sun Cluster 핵심
 - Sun Java System용 Sun Cluster 에이전트
 - 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
- HADB(High Availability Session Store) 4.4.2

관리 서비스

- Sun Java System Administration Server 5 2005.2Q4

주	Sun Cluster 및 Sun Cluster 에이전트는 Solaris OS에서 지원되며 Linux OS에서는 지원되지 않습니다.

릴리스 4 공유 구성 요소

공유 구성 요소는 선택 가능한 구성 요소에 대한 로컬 서비스 및 기술 지원을 제공합니다. Java ES 구성 요소를 설치하면 설치 프로그램이 아직 설치되지 않은 필수 공유 구성 요소를 자동으로 설치합니다.

이 Java ES 릴리스에는 다음 구성 요소가 포함되어 있습니다.

- ANT(Jakarta ANT Java/XML 기반 빌드 도구)
- ACL(Apache Commons Logging)
- BDB(Berkeley 데이터베이스)
- CAC(공용 에이전트 컨테이너)
- Derby 데이터베이스
- 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, Solaris 전용)
- MFWK(Java ES Monitoring Framework)
- SJWC(Sun Java Web Console)
- WSCL(Web services Common Library)

Java ES 5(릴리스 5)

이 절에서는 Java Enterprise System 5의 내용을 나열합니다.

릴리스 5 설치 프로그램-선택 가능한 구성 요소

Java ES 설치 프로그램의 구성 요소 선택 페이지에 선택 가능한 구성 요소가 제공되는 서비스별로 그룹화되어 있습니다. 각 구성 요소와 함께 설치되는 하위 구성 요소는 다음과 같습니다. Java ES 설치 프로그램에서는 더 이상 통신 서비스 구성 요소를 지원하지 않습니다.

공동 작업 서비스

- Sun Java System Portal Server 7.1
- Sun Java System Portal Server Secure Remote Access 7.1
 - 게이트웨이
 - Netlet 프록시
 - Rewriter 프록시

웹 및 응용 프로그램 서비스

- Sun Java System Application Server Enterprise Edition 8.2
 - Domain Administration Server
 - Application Server 노드 에이전트
 - 명령줄 관리 도구
 - 로드 균형 조정 플러그인

Web Server 또는 Apache Web Server와 함께 사용할 수 있으며 구성에서 선택할 수 있습니다. 기본값은 Web Server입니다.

- 샘플 응용 프로그램
- Sun Java System Web Server 7.0
 - Web Server 7.0 CLI
 - Web Server 7.0 Core
 - Web Server 7.0 샘플
- Sun Java Web Proxy Server 4.0.4

- Sun Java System Message Queue 3.7UR1
- Service Registry 3.1
 - Service Registry 클라이언트 지원
 - Service Registry 배포 지원

디렉토리 및 Identity 서비스

- Sun Java System Access Manager 7.1
 - Identity Management 및 Policy Services Core
 - Access Manager 관리 콘솔
 - 연합 관리를 위한 공통 도메인 서비스
 - Access Manager SDK
 - 분산 인증
 - 클라이언트 SDK
 - 세션 페일오버 클라이언트
- 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 명령줄 유틸리티
 - Sun Java System Directory Proxy Server 6 Core Server

가용성 서비스

- Sun Cluster 3.1
 - Sun Cluster 핵심
- 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

공유 서비스

- 모든 공유 구성 요소
430페이지의 "릴리스 5 공유 구성 요소"를 참조하십시오.
- Sun Java System Monitoring Console 1.0
- Java DB
 - Java DB 클라이언트
 - Java DB 서버

주 Sun Cluster 및 Sun Cluster 에이전트는 Solaris OS에서 지원되며 Linux OS에서는 지원되지 않습니다.

릴리스 5 공유 구성 요소

공유 구성 요소는 선택 가능한 구성 요소에 대한 로컬 서비스 및 기술 지원을 제공합니다. Java ES 구성 요소를 설치하면 설치 프로그램이 아직 설치되지 않은 필수 공유 구성 요소를 자동으로 설치합니다.

이 Java ES 릴리스에는 다음 구성 요소가 포함되어 있습니다.

- ANT(Jakarta ANT Java/XML 기반 빌드 도구)
- ACL(Apache Commons Logging)
- BDB(Berkeley 데이터베이스)
- Sun Cluster 전용 CAC(공용 에이전트 컨테이너)
- CAC(공용 에이전트 컨테이너)
- 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, Solaris 전용)
- MFWK(Java ES Monitoring Framework)
- SJWC(Sun Java Web Console)
- WSCL(Web services Common Library)
- XWSS(XML Web Services Security)

A

Access Manager

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 51

하위 구성 요소 423, 428

AccessManager-base 경로 255

AccessManagerConfig-base 경로 255

ACL 공유 구성 요소

전체 이름 29

종속성 표 47

amconfig 스크립트 (Access Manager) 271

ampre71upgrade 스크립트 (Access Manager) 269, 294

amupgrade 스크립트 (Access Manager) 276, 300

ANT 공유 구성 요소

전체 이름 29

종속성 표 47

ANT_HOME 환경 변수 320, 323, 346, 357, 360, 386, 389, 406, 409

Apache Commons Logging, ACL 참조

Application Server

공유 구성 요소 종속성 47

사후 설치 구성 220

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 51

하위 구성 요소 423, 427

AppServer7-base 경로 200

AppServer7Config-base 경로 200

AppServer8-base 경로 200

AppServer8Config-base 경로 200

AppServer8Install-base 경로 200

B

BDB 공유 구성 요소

전체 이름 29

종속성 표 47

Berkeley Database, BDB 참조

C

CAC 공유 구성 요소

전체 이름 29

종속성 표 47

D

db2bak 유틸리티 265

Derby 데이터베이스 공유 구성 요소 425

Directory Proxy Server

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 51

Directory Server

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 52

DirServer-base 경로 (Directory Server 6.0) 99, 115

F

FastInfoSet, FIS 참조

FIS 공유 구성 요소

전체 이름 29

종속성 표 47

H

HADB(High Availability Session Store)

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 52

HADB-base 경로 163

I

ICU 공유 구성 요소

전체 이름 29

종속성 표 47

IM-SDK 공유 구성 요소

전체 이름 29

종속성 표 47

Instant Messenger SDK, IM-SDK 참조

International Components for Unicode, ICU 참조

J

JAF 공유 구성 요소

전체 이름 29

종속성 표 47

Jakarta ANT Java/XML 기반 빌드 도구, ANT 참조

JATO 공유 구성 요소

전체 이름 29

종속성 표 47

Java 2 Platform, Standard Edition, J2SE 참조

Java 2 Platform, Standard Edition, Java SE 참조

Java API for XML Processing, JAXP 참조

Java API for XML Registries, JAXR 참조

Java API for XML-based Remote Procedure Call,
JAX-RPC 참조

Java Architecture for XML Binding, JAXB 참조

Java Calendar API, JCAPI 참조

Java DB

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 52

Java Dynamic Management Kit, JDMK 참조

Java ES 2003Q4(릴리스 1)

공유 구성 요소 413

제품 구성 요소 412

Java ES 2004Q2(릴리스 2)
 공유 구성 요소 [415](#)
 제품 구성 요소 [414](#)
 Java ES 2005Q1(릴리스 3)
 공유 구성 요소 [420](#)
 제품 구성 요소 [417](#)
 Java ES 2005Q4(릴리스 4)
 공유 구성 요소 [425](#)
 제품 구성 요소 [422](#)
 Java ES 5(릴리스 5)
 공유 구성 요소 [430](#)
 제품 구성 요소 [427](#)
 Java ES Monitoring Framework, MFWK [참조](#)
 Java SE 공유 구성 요소
 전체 이름 [29](#)
 종속성 표 [47](#)
 Java Security Services, JSS [참조](#)
 Java Studio Web Application Framework, JATO [참조](#)
 JAVA_HOME [346](#), [357](#), [360](#), [386](#), [389](#), [406](#), [409](#)
 JAVA_HOME 환경 변수 [273](#), [298](#), [320](#), [323](#)
 JavaBeans Activation Framework, JAF [참조](#)
 JavaHelp 공유 구성 요소
 전체 이름 [29](#)
 종속성 표 [47](#)
 JavaMail 공유 구성 요소
 전체 이름 [29](#)
 종속성 표 [47](#)
 JAXB 공유 구성 요소
 전체 이름 [29](#)
 종속성 표 [47](#)
 JAXP 공유 구성 요소
 전체 이름 [29](#)
 종속성 표 [47](#)
 JAXR 공유 구성 요소
 전체 이름 [29](#)
 종속성 표 [47](#)
 JAX-RPC 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)

JAXWS 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)
 JCAPI 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)
 JDMK 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)
 JSP Standard Library Template, JSTL [참조](#)
 JSP 파일, 사용자 정의 [258](#), [266](#), [271](#), [296](#)
 JSS 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)
 JSTL 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)

K

KT Search Engine, KTSE [참조](#)
 KTSE 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)

L

LDAP C Language SDK 공유 구성 요소 [430](#)
 LDAP C SDK 공유 구성 요소
 전체 이름 [30](#)
 종속성 표 [47](#)
 LDAP J SDK 공유 구성 요소 [431](#)
 전체 이름 [30](#)
 종속성 표 [47](#)

M

M

MA Core 공유 구성 요소

전체 이름 30

종속성 표 47

Message Queue

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 52

MessageQueue-base 경로 175

MFWK 공유 구성 요소

전체 이름 30

종속성 표 47

Mobile Access Core, MA Core 참조

Monitoring Console

공유 구성 요소 종속성 47

약어 28

제품 구성 요소 종속성 52

mqupgrade 스크립트 (Message Queue) 194, 195

N

Netscape Portable Runtime, NSPR 참조

Network Security Services, NSS 참조

NSPR 공유 구성 요소

전체 이름 30

종속성 표 47

NSS 공유 구성 요소

전체 이름 30

종속성 표 48

P

Portal Server

공유 구성 요소 종속성 47

약어 28, 44

업그레이드 방법 34

제품 구성 요소 종속성 52

Portal Server Secure Remote Access

공유 구성 요소 종속성 47

약어 29, 44

업그레이드 방법 34

제품 구성 요소 종속성 53

PortalServer6-base 경로 304, 374

PortalServer6Config-base 경로 304, 374

PortalServer7-base 경로 304, 374

PortalServer7Config-base 경로 304, 374

PortalServer7Data-base 경로 304, 374

postInstall 스크립트 (Application Server) 220

R

RegistryDomainR4-base 경로 227

RegistryDomainR5-base 경로 227

rel4CAC-base-dir 경로 71

rel5CAC-admin-dir 경로 71

S

SAAJ 공유 구성 요소

전체 이름 30

종속성 표 48

SAML 278, 302

SASL 공유 구성 요소

전체 이름 30

종속성 표 48

Security Assertion Markup Language, SAML 참조

SEDC 공유 구성 요소

전체 이름 30

종속성 표 48

serverRoot 경로 (Directory Server 5.x) 99, 115

Service Registry

공유 구성 요소 종속성 47

약어 29, 44

업그레이드 방법 34

제품 구성 요소 종속성 53

ServiceRegistryR4-base 경로 153, 227
ServiceRegistryR5-base 경로 227
 Simple Authentication and Security Layer, SASL 참조
 SJWC 공유 구성 요소
 전체 이름 30
 종속성 표 48
 SML Web Services Security, XWSS 참조
 SOAP runtime with Attachments API for Java, SAAJ 참조
 Sun Cluster
 공유 구성 요소 종속성 47
 약어 29, 44
 업그레이드 방법 34
 제품 구성 요소 종속성 53
 Sun Cluster Geographic Edition
 공유 구성 요소 종속성 47
 약어 29, 45
 업그레이드 방법 34
 제품 구성 요소 종속성 53
 Sun Explorer Data Collector, SEDC 참조
 Sun Java Web Console, SJWC 참조
SunClusterGeo-base 경로 87

W

Web Proxy Server
 공유 구성 요소 종속성 47
 약어 29, 45
 업그레이드 방법 34
 제품 구성 요소 종속성 53
 Web Server
 공유 구성 요소 종속성 47
 약어 29, 45
 업그레이드 방법 34
 제품 구성 요소 종속성 54
 Web services Common Library, WSCL 참조
WebProxyServer-base 경로 239
WebServer6-base 경로 129
WebServer7-base 경로 129
WebServer7Config-base 경로 129

WSCL 공유 구성 요소
 전체 이름 30
 종속성 표 48

X

XWSS 공유 구성 요소
 전체 이름 30
 종속성 표 48

ㄱ

공용 에이전트 컨테이너, CAC 참조
 공유 구성 요소
 Java ES 2003Q4(릴리스 1) 413
 Java ES 2004Q2(릴리스 2) 415
 Java ES 2005Q1(릴리스 3) 420
 Java ES 2005Q4(릴리스 4) 425
 Java ES 5(릴리스 5) 430
 상호 종속성 49
 종속되는 제품 구성 요소 46

규칙
 기호 21
 활자체 21
 기호 규칙 21

人

스크립트
 amconfig(Access Manager) 271
 ampre71upgrade(Access Manager) 269, 294
 amupgrade(Access Manager) 276, 300
 mqmigrate(Message Queue) 195
 mqupgrade(Message Queue) 194
 postInstall(Application Server) 220

o

○

운영 체제

- 계획 고려 사항 [42](#)
- 이중 업그레이드 [43](#)
- 지원 매트릭스 [43](#)
- 패치 [42](#)

이중 업그레이드

- 구성 요소 지원 요약 [44](#)
- 접근 방식 [44](#)
- 정의됨 [43](#)

ㅈ

제품 구성 요소

- Java ES 2003Q4(릴리스 1) [412](#)
- Java ES 2004Q2(릴리스 2) [414](#)
- Java ES 2005Q1(릴리스 3) [417](#)
- Java ES 2005Q4(릴리스 4) [422](#)
- Java ES 5(릴리스 5) [427](#)
- 공유 구성 요소에 대한 종속성 [46](#)
- 상호 종속성 [51](#)

종속성

- 공유 구성 요소 간 [49](#)
- 제품 구성 요소 간 [51](#)
- 제품 구성 요소, 공유 구성 요소 [46](#)

ㅎ

활자체 규칙 [21](#)