



Sun Java System Application Server Enterprise Edition 8.2 릴리스 노트



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

부품 번호: 820-0501
2008년 5월

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 모든 권리는 저작권자의 소유입니다.

Sun Microsystems, Inc.는 이 문서에 설명된 제품의 기술 관련 지적 재산을 소유합니다. 특히 이 지적 재산권에는 하나 이상의 미국 특허권 또는 미국 및 다른 국가에서 특허 출원 중인 응용 프로그램이 포함될 수 있습니다.

미국 정부의 권리 - 상용 소프트웨어. 정부 사용자는 Sun Microsystems, Inc. 표준 사용권 계약과 해당 FAR 규정 및 보충 규정을 준수해야 합니다.

이 배포에는 타사에서 개발한 자료가 포함되어 있을 수 있습니다.

제품 중에는 캘리포니아 대학에서 허가한 Berkeley BSD 시스템에서 파생된 부분이 포함되어 있을 수 있습니다. UNIX는 미국 및 다른 국가에서 X/Open Company, Ltd.를 통해 독점적으로 사용권이 부여되는 등록 상표입니다.

Sun, Sun Microsystems, Sun 로고, Solaris 로고, Java Coffee Cup 로고, docs.sun.com, Java 및 Solaris는 미국 및 다른 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다. 모든 상표는 사용 허가를 받았으며 미국 및 다른 국가에서의 상표 또는 등록 상표입니다. SPARC 상표를 사용하는 제품은 Sun Microsystems, Inc.가 개발한 구조를 기반으로 하고 있습니다.

OPEN LOOK 및 SunTM 그래픽 사용자 인터페이스(GUI)는 Sun Microsystems, Inc.가 자사의 사용자 및 정식 사용자로 개발했습니다. Sun은 컴퓨터 업계를 위한 시각적 또는 GUI의 개념을 연구 개발한 Xerox사의 선구적인 노력을 높이 평가하고 있습니다. Sun은 Xerox와 Xerox 그래픽 사용자 인터페이스(GUI)에 대한 비독점적 사용권을 보유하고 있습니다. 이 사용권은 OPEN LOOK GUI를 구현하는 Sun의 정식 사용자에게도 적용되며 그렇지 않은 경우에는 Sun의 서면 사용권 계약을 준수해야 합니다.

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

설명서는 "있는 그대로" 제공되며, 법률을 위반하지 않는 범위 내에서 상품성, 특정 목적에 대한 적합성 또는 비침해에 대한 묵시적인 보증을 포함하여 모든 명시적 또는 묵시적 조건, 표현 및 보증을 배제합니다.

개요

Sun Java™ System Application Server Enterprise Edition 8.2 제품은 J2EE 응용 프로그램 및 웹 서비스를 만들고 관리하는 작업을 훨씬 간소화합니다. 이 제품은 소프트웨어와 하드웨어 고장에도 불구하고 계속 작동하는 확장 가능한 서비스를 위한 뛰어난 성능, 클러스터링 및고가용성 기능을 제공합니다.

- 3 페이지 “릴리스 노트 정보”
- 4 페이지 “릴리스 노트 개정 내역”
- 4 페이지 “Application Server 설명서 세트”
- 5 페이지 “관련 설명서”
- 6 페이지 “내게 필요한 옵션 기능”
- 6 페이지 “설명서, 지원 및 교육”
- 6 페이지 “문제점 보고 및 사용자 의견 제공 방법”
- 7 페이지 “사용자 의견 환영”

릴리스 노트 정보

이 릴리스 노트는 Sun Java System Application Server 8.2 릴리스 당시에 사용 가능한 중요한 정보가 포함되어 있습니다. 이 문서에서는 향상된 기능, 알려진 문제 및 기타 최신 정보를 제공합니다. Application Server Enterprise Edition 8.2를 사용하기 전에 이 문서를 반드시 읽어보시기 바랍니다.

이 릴리스 노트의 최신 버전은 Sun Java System [설명서 웹 사이트](http://docs.sun.com/app/docs/coll/1310.3) (<http://docs.sun.com/app/docs/coll/1310.3>)에서 보실 수 있습니다. 소프트웨어를 설치 및 설정하기 전과 후에도 웹 사이트를 검토하여 최신 릴리스 노트와 제품 설명서를 정기적으로 확인하시기 바랍니다.

이 설명서에서는 추가 관련 정보를 제공하기 위해 타사 URL을 참조하기도 합니다.

주 - Sun은 이 설명서에 언급된 타사 웹 사이트의 가용성에 대해 책임지지 않습니다. Sun은 이러한 사이트나 자원을 통해 사용할 수 있는 내용, 광고, 제품 또는 기타 자료에 대해서는 보증하지 않으며 책임지지 않습니다. Sun은 해당 사이트 또는 자원을 통해 사용 가능한 내용, 제품 또는 서비스의 사용과 관련해 발생하거나 발생했다고 간주되는 손해나 손실에 대해 책임이나 의무를 지지 않습니다.

릴리스 노트 개정 내역

이 절에서는 Sun Java System Enterprise Edition Application Server 8.2 제품이 처음 릴리스된 이후의 릴리스 노트 변경 사항에 대해서 설명합니다.

표 1-1 릴리스 노트 개정 내역

개정 날짜	설명
2006년 10월	Sun Java System Application Server Enterprise Edition 8.2 제품의 베타 릴리스용 첫 번째 릴리스 노트
2007년 2월	Sun Java System Application Server Enterprise Edition 8.2 제품의 FCS 릴리스용 릴리스 노트
2007년 7월	설치에 대한 알려진 문제점에 결함 6396045 추가
2007년 8월	Java ES 5에 대해 지원되는 Solaris 및 Linux 플랫폼에 맞게 Web Server의 플랫폼 요구 사항 변경
2008년 5월	지원되는 Web Server를 6.0, 6.1, 7.0으로 변경 "시스템 가상화 지원" 절 추가

Application Server 설명서 세트

Application Server 설명서 세트에서는 배포 계획 및 시스템 설치에 대해 설명합니다. 독립 실행형 Application Server 설명서에 대한 URL(Uniform Resource Locator)은 <http://docs.sun.com/app/docs/coll/1310.4>입니다. Java ES(Sun Java Enterprise System) Application Server 설명서에 대한 URL은 <http://docs.sun.com/app/docs/coll/1310.3>입니다. Application Server에 대한 소개 내용을 보려면 다음 표에 나열된 설명서를 순서대로 참조하십시오.

표 1-2 Application Server 설명서 세트에 포함된 설명서

설명서 제목	설명
릴리스 노트	소프트웨어 및 설명서 관련 최신 정보로 지원되는 하드웨어, 운영 체제, JDK™(Java Development Kit) 및 데이터베이스 드라이버를 표를 기반으로 종합적으로 요약하였습니다.

표 1-2 Application Server 설명서 세트에 포함된 설명서 (계속)

설명서 제목	설명
빠른 시작 설명서	Application Server 제품을 시작하는 방법에 대해 설명합니다.
설치 설명서	소프트웨어와 해당 구성 요소 설치에 대해 설명합니다.
배포 계획 설명서	사용자 시스템 요구 사항과 기업 평가를 통해 Application Server를 사용자 사이트에 가장 적합한 방식으로 배포하는 방법에 대해 설명합니다. 서버 배포 시 알아야 할 일반적인 문제와 관심을 기울여야 할 사항에 대해서도 설명합니다.
Developer's Guide	J2EE 구성 요소 및 API용 개방형 Java 표준 모델을 따르는 Application Server에서 실행할 Java 2 Platform, Enterprise Edition(J2EE™ 플랫폼) 응용 프로그램을 만들고 구현하는 방법에 대해 설명합니다. 개발자 도구, 보안, 디버깅, 배포 및 라이프사이클 모듈 생성에 대한 정보를 제공합니다.
J2EE 1.4 Tutorial	J2EE 1.4 플랫폼 기술과 API를 사용하여 J2EE 응용 프로그램을 개발하는 방법에 대해 설명합니다.
관리 설명서	관리 콘솔에서 Application Server 하위 시스템 및 구성 요소 구성, 관리 및 배포에 대해 설명합니다.
고가용성 관리 설명서	고가용성 데이터베이스를 위한 설치 후 구성 및 관리 방법에 대해 설명합니다.
Administration Reference	Application Server 구성 파일인 domain.xml을 편집하는 방법에 대해 설명합니다.
Upgrade and Migration Guide	응용 프로그램을 특히 Application Server 6.x 및 7에서 새로운 Application Server 프로그래밍 모델로 마이그레이션하는 방법에 대해 설명합니다. 제품 사양과 호환되지 않는 결과를 가져올 수 있는 제품 릴리스 및 구성 옵션의 차이점에 대한 설명도 포함되어 있습니다.
Performance Tuning Guide	Application Server를 조정하여 성능을 향상시키는 방법에 대해 설명합니다.
Troubleshooting Guide	Application Server 문제를 해결하는 방법에 대해 설명합니다.
Error Message Reference	Application Server 오류 메시지를 해결하는 방법에 대해 설명합니다.
Reference Manual	설명서 페이지 스타일로 작성되었으며, Application Server와 같이 사용할 수 있는 유틸리티 명령에 대해 설명합니다. asadmin 명령줄 인터페이스를 포함합니다.

관련 설명서

Application Server는 단독으로 구입하거나 네트워크 또는 인터넷 환경에서 배포된 엔터프라이즈 응용 프로그램을 지원하는 소프트웨어 인프라인 Java ES의 구성 요소로 구입할 수 있습니다. Application Server를 Java ES의 구성 요소로 구입한 경우에는 <http://docs.sun.com/coll/1286.2> 및 <http://docs.sun.com/coll/1397.2>의 시스템 설명서를 잘 이해해야 합니다. Java ES 및 해당 구성 요소에 대한 모든 설명서의 URL은 <http://docs.sun.com/prod/entsys.5>입니다.

기타 Sun Java System 서버 설명서를 보려면 다음을 참조하십시오.

- Message Queue 설명서
- Directory Server 설명서

- Web Server 설명서

또한 다음과 같은 자원도 유용할 수 있습니다.

- J2EE 1.4 사양 (<http://java.sun.com/j2ee/1.4/docs/index.html>)
- J2EE 1.4 Tutorial (<http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html>)
- J2EE Blueprints (<http://java.sun.com/reference/blueprints/index.html>)

내게 필요한 옵션 기능

이 매체를 발행한 이후 릴리스된 내게 필요한 옵션 기능을 사용하려면 Sun에 요청하여 구할 수 있는 섹션 508 제품 평가를 참조하여 관련 솔루션을 배포하는 데 가장 적합한 버전을 확인하십시오. 응용 프로그램의 업데이트된 버전은

<http://sun.com/software/javaenterprisesystem/get.html>에서 찾을 수 있습니다.

내게 필요한 옵션 기능 구현을 위한 Sun의 방침에 대해 자세히 알아보려면

<http://sun.com/access>를 방문하십시오.

설명서, 지원 및 교육

Sun 웹 사이트에서는 다음 추가 자원에 대한 정보가 제공됩니다.

- 설명서(<http://www.sun.com/documentation/>)
- 지원(<http://www.sun.com/support/>)
- 교육(<http://www.sun.com/training/>)

문제점 보고 및 사용자 의견 제공 방법

Sun Java System Application Server에 문제가 있으면 다음 방식 중 하나를 사용하여 Sun 고객 지원부에 문의하십시오.

- 사용자 의견 제출 양식
(<http://java.sun.com/docs/forms/J2EE14SubmittalForm.html>) — Application Server 제품에 관한 의견을 제출할 때 쓰이는 양식
- J2EE-INTEREST 목록
(<http://archives.java.sun.com/archives/j2ee-interest.html>) — J2EE 관련 의문 사항에 대한 메일링 목록
- Java Developer Connection의 버그 데이터베이스 (<http://developer.java.sun.com/Servlet/SessionServlet?url=/developer/bugParade/index.jshtml>) — 버그를 보거나 제출하려면 Java Developer Connection Bug Parade를 사용하십시오.
- Java 기술 포럼 (<http://forum.java.sun.com/>) — Java 기술 및 프로그래밍 기술에 대한 지식과 질문을 나누는 대화식 메시지 보드입니다. Sun Java System Application Server Enterprise Edition 8.2에 관련된 토론은 J2EE SDK 포럼을 이용하십시오.
- Sun 소프트웨어 지원 서비스 (<http://www.sun.com/service/sunone/software>) — 유지 보수 프로그램과 지원 연락처뿐 아니라 기술 자료, 온라인 지원 센터 및 제품 검색에 대한 링크가 있습니다.

- 유지 보수 계약과 관련된 직통 전화번호
당사의 지원팀이 문제를 해결하는 데 도움이 될 수 있도록 지원 담당자에게 문의하기 전에 다음 정보를 준비해 두십시오.
- 문제가 발생한 상황과 해당 문제가 작업에 미친 영향 등을 포함한 문제에 대한 자세한 설명
- 컴퓨터 종류, 운영 체제 버전, 제품 버전 및 문제에 영향을 미친 패치와 기타 소프트웨어
- 문제를 재현하기 위해 사용한 방법에 대한 자세한 단계
- 오류 로그 또는 코어 덤프

사용자 의견 환영

Sun은 설명서의 내용을 지속적으로 개선하고자 하며 사용자 여러분의 의견과 제안을 환영합니다. 사용자 의견을 보내시려면 <http://docs.sun.com>을 방문하여 의견 보내기를 누릅니다. 온라인 양식에서 전체 문서 제목과 부품 번호를 기입해 주십시오. 부품 번호는 해당 설명서의 제목 페이지나 문서의 URL에 있는 7자리 또는 9자리 숫자입니다. 예를 들어, 이 설명서의 부품 번호는 820-0501입니다.

Application Server Enterprise Edition 8.2 정보

Sun Java System Application Server Enterprise Edition 8.2는 대규모 생산 환경에서의 J2EE 응용 프로그램 및 Java 기술 기반 웹 서비스 개발 및 배포를 위한 J2EE 1.4 플랫폼 호환 서버입니다.

이 장은 다음 내용으로 구성되어 있습니다.

- 9 페이지 “8.2 릴리스의 새로운 기능”
- 11 페이지 “하드웨어 및 소프트웨어 요구 사항”
- 20 페이지 “Enterprise Edition 8.2 릴리스에서 해결된 버그”
- 23 페이지 “추가 HADB 정보”
- 32 페이지 “호환성 문제”
- 34 페이지 “J2EE 지원”
- 35 페이지 “지원되는 다른 Java 버전으로 전환”
- 35 페이지 “고성능”
- 35 페이지 “확장성”
- 36 페이지 “JavaServer Faces 1.1 지원”

8.2 릴리스의 새로운 기능

Application Server Enterprise Edition 8.2의 향상된 기능은 다음과 같습니다.

- **향상된 관리** — Application Server는 브라우저 기반 콘솔 또는 스크립트 작성 가능 명령줄 인터페이스를 사용하여 복잡한 다중 시스템 개발의 원격 보안 관리를 지원합니다. 또한 관리 및 모니터링 기능에 대한 원격 보안 프로그램 액세스가 가능하도록 풍부한 JMX 기반 API를 제공합니다.
- **메시지 브로커** — Application Server는 고가용성, 신뢰성, 고성능 및 확장 가능 메시징을 제공하는 통합 엔터프라이즈급 메시지 브로커와 함께 제공됩니다.
- **Message Queue 3.7** — Application Server는 이제 MQ 3.7을 구현합니다.
- **확장 플랫폼 지원** — 추가 운영 체제, 데이터베이스, 로컬 및 하드웨어가 지원됩니다.
- **Sun Java Enterprise System** — Sun Java Enterprise System의 주요 구성 요소로서 Application Server는 포털 및 네트워크 아이디 서비스와 완벽하게 통합되어 있습니다.

- **마이그레이션 및 업그레이드 도구** — 이러한 도구는 J2EE 응용 프로그램의 표준 준수 여부와 이식성을 검증하고 다른 J2EE Application Server(JBoss, WebLogic, WebSphere)에서의 마이그레이션 및 Sun ONE Application Server/ iPlanet Application Server 이전 버전에서의 업그레이드를 지원하는 데 사용됩니다.
- **Java 2 Standard Edition 5.0 지원** — Application Server는 향상된 관리 및 모니터링 기능과 개선된 성능 및 확장성 기능을 포함하는 Java 2 Standard Edition 5.0을 지원합니다.
- **Java Web Services Developer Pack 1.6 (JWSDP) 플러그인 지원** — 모든 JWSDP 플러그인이 지원됩니다. JWSDP 1.6은 <http://java.sun.com/webservices/downloads/1.6/index.html>에서 무료로 다운로드할 수 있습니다.
- **Java DB 데이터베이스 지원** — Application Server에는 [Apache Derby](http://db.apache.org/derby/) (<http://db.apache.org/derby/>)에 기반한 Java DB 데이터베이스가 포함되어 있습니다. Pointbase 데이터베이스와의 역방향 호환성이 유지되지만 서버에 새로 생성되는 데이터베이스는 기본적으로 Java DB를 사용합니다. Application Server 8.x에서 업그레이드한 후에 기존 도메인은 계속해서 PointBase를 사용하지만 업그레이드 후 새로 생성되는 도메인은 Java DB를 사용합니다.
- **JDBC 드라이버** — Application Server는 Sun JDBC 드라이버와 함께 제공됩니다.
- **웹 서비스 보안** — 이 컨테이너 메시지 보안 기법은 OASIS WS-Security 표준의 X509 및 아이디/비밀번호 프로필을 사용하여 SOAP 웹 서비스 호출에 대한 메시지 수준 인증(예: XML 디지털 서명 및 암호화)을 구현합니다.
- **WS-I Basic Profile 1.1** — J2EE 1.4 사양에 명시된 대로 이 릴리스는 Web Services Interoperability(WS-I) Basic Profile 1.1을 구현하여 웹 서비스 응용 프로그램을 위한 상호 운용성을 사용 가능하게 합니다.
- **iWay 어댑터와의 백엔드 연결성** — Sun Microsystems는 현재 Application Server 환경 내에서 기존 IT 응용 프로그램을 활용하는 데 도움이 되도록 주요 백엔드 시스템(SAP, Siebel, Oracle, CICS 및 IBM MQ 시리즈)에 대한 22개의 iWay 어댑터를 전매 및 지원합니다. 이러한 어댑터는 J2EE Connector Architecture 1.5 사양과 웹 서비스(SOAP) 표준을 지원하며 백엔드 응용 프로그램에 연결하는 시간을 단축하는 개발자 도구를 포함합니다.
- **최신 HADB 관리 시스템** — UNIX™ 플랫폼에는 데이터베이스 서버, ODBC 2.5 드라이버, JDBC 3.0 Type 4 드라이버, clusql(SQL 문을 입력하고 실행하는 대화식 프로그램) 및 관리 시스템으로 구성된 새로운 고가용성 데이터베이스(HADB) 관리 시스템(HADB 버전 4.4.3)이 포함되어 있습니다. 이로써 SSH/RSH에 대한 종속성은 제거되지만 UDP 멀티캐스트를 위해 네트워크를 구성해야 합니다. HADB 요구 사항 및 제한 사항에 대한 자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.2 고가용성 관리 설명서**를 참조하십시오.
- **Solaris 10 영역 지원** — Application Server는 Solaris 10 시스템의 전역 또는 그 외 영역에 설치할 수 있습니다. Solaris 영역에 대한 자세한 내용은 **Solaris 영역** (<http://www.sun.com/bigadmin/content/zones/>) 페이지를 참조하십시오.

- 동적 콘텐츠 기술 지원 종료 — CGI-bin 및 SHTML과 같은 동적 콘텐츠 기술이 더 이상 지원되지 않습니다.

하드웨어 및 소프트웨어 요구 사항

이 절에서는 Sun Java System Application Server Enterprise Edition 8.2 제품을 설치하기 전에 반드시 갖추어야 하는 요구 사항에 대해 설명합니다.

- 11 페이지 “플랫폼 요구 사항”
- 12 페이지 “시스템 가상화 지원”
- 12 페이지 “중요 패치 정보”
- 12 페이지 “JDBC 드라이버 및 데이터베이스”
- 13 페이지 “번들로 제공된 Java DB 데이터베이스 사용”
- 17 페이지 “지원되는 웹 서버”
- 17 페이지 “브라우저”
- 18 페이지 “HADB 요구 사항 및 지원되는 플랫폼”
- 19 페이지 “Sun Java System Application Server 업그레이드”
- 19 페이지 “기타 요구 사항”

플랫폼 요구 사항

다음 표는 Sun Java System Application Server Enterprise Edition 8.2 제품에서 지원하는 운영 체제 목록과 Application Server를 설치하고 실행하는 데 필요한 최소한의 권장 메모리 요구 사항을 나타냅니다.

표 2-1 Sun Java System Application Server 8.2 플랫폼 요구 사항

운영 체제	최소 메모리	권장 메모리	최소 디스크 공간	필요 하드 디스크 공간	JVM ¹
Sun Solaris 9, 10(SPARC) Solaris 9, 10(x86)	512MB	1GB	250MB의 여유 공간	500MB의 여유 공간	J2SE_5_08
Sun Java Desktop System	512MB	1GB	250MB의 여유 공간	500MB의 여유 공간	J2SE_5_08
Redhat Enterprise Linux 3.0 U1, 4.0	512MB	1GB	250MB의 여유 공간	500MB의 여유 공간	J2SE_5_08
Windows Server 2000 SP4+ Windows 2000 Advanced Server SP4+ Windows Server 2003 Windows XP Pro SP1+	1GB	2GB	500MB의 여유 공간	1GB의 여유 공간	J2SE_5_08

¹ 32비트(64비트 아님) JVM만 지원됩니다.

주 - 위 목록의 Application Server에 대한 시스템 요구 사항과 18 페이지 “HADB 요구 사항 및 지원되는 플랫폼”에 나열된 HADB에 대한 시스템 요구 사항은 꼭 동일하지 않습니다. 이는 설명서의 내용상 오류가 아니며, Application Server와 HADB 서버는 일반적으로 서로 다른 시스템에서 실행됩니다.

UNIX에서는 `uname` 명령을 사용하여 운영 체제 버전을 확인할 수 있습니다. 디스크 공간은 `df` 명령을 사용하여 확인할 수 있습니다.

주 - Application Server를 Microsoft Windows 플랫폼에서 실행할 경우 FAT 또는 FAT32 대신 NTFS 파일 시스템을 사용해야 합니다.

시스템 가상화 지원

시스템 가상화는 여러 운영 체제(OS) 인스턴스를 공유 하드웨어에서 독립적으로 실행할 수 있도록 해주는 기술입니다. 기능적으로, 가상화된 환경에서 호스트된 OS에 배포된 소프트웨어는 일반적으로 기반 플랫폼이 가상화되었는지를 인식하지 못합니다. Sun에서는 선별된 시스템 가상화와 OS 조합에서 Sun Java System 제품을 테스트하여 Sun Java System 제품이 적절한 규모로 적절하게 구성된 가상화된 환경에서 가상화되지 않은 시스템에서처럼 계속해서 작동하는지를 검증합니다. 가상화된 환경에서의 Sun Java System 제품 지원에 대한 내용은 System Virtualization Support in Sun Java System Products를 참조하십시오.

중요 패치 정보

Solaris 패치 요구 사항

Solaris 9, 10 (x86, SPARC) 사용자는 “Sun recommended patch cluster”를 설치할 것을 권장합니다. 이 패치 클러스터는 SunSolve의 [Recommended and Security Patches](http://sunsolve.sun.com/) (<http://sunsolve.sun.com/>)에서 다운로드할 수 있습니다.

RedHat Enterprise Linux 3.0 추가 패키지 요구 사항

설치 프로그램을 비롯한 이 제품의 기본 구성 요소를 실행하려면 표준 RedHat Enterprise Linux 3.0 배포의 일부가 아닌 다음 패키지를 설치해야 합니다.

`compat-libstdc++-7.3-2.96.118.i386.rpm`

이 패키지는 <http://rpm.pbone.net/>

[index.php3/stat/4/idpl/843376/com/compat-libstdc++-7.3-2.96.118.i386.rpm.html](http://rpm.pbone.net/index.php3/stat/4/idpl/843376/com/compat-libstdc++-7.3-2.96.118.i386.rpm.html)에서 다운로드할 수 있습니다.

JDBC 드라이버 및 데이터베이스

Sun Java System Application Server는 DBMS와 해당 JDBC 드라이버의 연결을 지원하도록 설계되어 있습니다. Sun에서 테스트를 거쳐 J2EE 호환 데이터베이스 구성을 위해 사용할 수 있는 것으로 확인된 구성 요소 목록은 다음 표를 참조하십시오.

표 2-2 J2EE 호환 JDBC 드라이버

JDBC 공급업체	JDBC 드라이버 유형	지원되는 데이터베이스 서버
i-net Software	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3+, 10.1.x, 10.2.x Sybase ASE 12.5 Microsoft SQL Server 2000 4.0 Service Pack 1
IBM	Type 2	IBM DB2 8.1 Service Pack 3+
Java DB	Type 4	Apache Derby 10.1.3
PointBase	Type 4	PointBase Network Server 5.2
DataDirect	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3+, 10.1.x, 10.2.x Sybase ASE 12.5.2 Microsoft SQL Server IBM DB2 8.1 Service Pack 3+
MySQL	Type 4	5.x
Sun Java System JDBC Driver for Oracle	Type 4	Oracle (R) 9.2.0.3, 10G
Sun Java System JDBC Driver for DB2	Type 4	IBM DB2 8.1 Service Pack 3+
Sun Java System JDBC Driver for Sybase	Type 4	Sybase ASE 12.5.2
Sun Java System JDBC Driver for Microsoft SQL Server	Type 4	Microsoft SQL Server 2000 4.0 Service Pack 1
Oracle	Type 4, Type 2	Oracle (R) 9.2.0.3, 10G

번들로 제공된 Java DB 데이터베이스 사용

이 절에서는 Application Server 8.2에 번들로 제공된 Java DB 데이터베이스 구현을 사용하기 위한 지침을 제공합니다.

- 13 페이지 “Java DB 데이터베이스 시작 및 중지”
- 14 페이지 “Java DB 유틸리티 스크립트”
- 15 페이지 “Pointbase에서 Java DB로 테이블 내보내기”

Java DB 데이터베이스 시작 및 중지

Sun Java System Application Server 8.2에 Java DB Network Server 시작 및 중지를 위한 두 `asadmin` 명령이 새로 추가되었습니다.

- `start-database` 명령을 사용하여 Java DB Network Server의 인스턴스를 시작할 수 있습니다.

```
start-database [--dbhost 0.0.0.0] [--dbport 1527] [--dbhome path/derby]
```

호스트의 기본값은 0.0.0.0이며, 이 경우 Java DB는 IP/호스트 이름 인터페이스와 localhost를 모두 수신할 수 있습니다. dbhome 등록 정보의 값은 Java DB 데이터베이스의 위치입니다. 기본 path는 <appserver_install_dir>/derby입니다.

- asadmin stop-database 명령을 사용하여 실행 중인 Java DB Network Server의 인스턴스를 종료할 수 있습니다.

```
stop-database [--dbhost 0.0.0.0] [--dbport 1527]
```

Java DB 유틸리티 스크립트

또한 Application Server 8.2과 함께 제공되는 Java DB 구성에는 Java DB를 사용하는 데 도움이 되는 여러 가지 유용한 스크립트가 포함되어 있습니다. 다음 스크립트는 <appserver_install_dir>/derby/frameworks/NetworkServer/bin 디렉토리에서 사용할 수 있습니다.

- startNetworkServer.ksh/bat — 네트워크 서버를 시작하는 스크립트
- stopNetworkServer.ksh/bat — 네트워크 서버를 중지하는 스크립트
- ij.ksh/bat — 대화형 JDBC 스크립트 도구
- dblook.ksh/bat — 데이터베이스의 모든 또는 일부 DDL을 보기 위한 스크립트
- sysinfo.ksh/bat — Java DB 환경에 관한 버전 정보를 표시하는 스크립트
- NetworkServerControl.ksh/bat — NetworkServerControl API에서의 명령 실행 방법을 제공하는 스크립트

▼ Java DB 유틸리티 스크립트 실행을 위한 환경을 구성하는 방법

- 1 DERBY_INSTALL 환경 변수가 <appserver_install_dir>/derby 디렉토리를 가리키도록 설정합니다.
- 2 CLASSPATH 환경 변수 설정을 해제합니다.
- 3 선택적으로 다음 등록 정보를 설정할 수도 있습니다.
 - a. DERBY_SERVER_HOST 등록 정보를 네트워크 서버가 수신하는 호스트로 설정합니다. 모든 수신기를 활성화하기 위해 0.0.0.0으로 설정할 수도 있습니다.
 - b. DERBY_SERVER_PORT 등록 정보를 네트워크 서버가 수신하는 포트 번호로 설정합니다.

참조 이러한 유틸리티에 대한 자세한 내용은 Derby [도구](http://db.apache.org/derby/docs/10.1/tools/) (<http://db.apache.org/derby/docs/10.1/tools/>) 및 [관리](http://db.apache.org/derby/docs/10.1/adminguide/) (<http://db.apache.org/derby/docs/10.1/adminguide/>) 설명서를 참조하십시오.

Pointbase에서 Java DB로 테이블 내보내기

이 예에서는 Pointbase 테이블의 DDL을 캡처하고 Netbeans 5.0을 사용하여 Java DB에서 동일한 테이블을 만드는 방법을 보여줍니다. commander 도구와 unload database 명령을 사용하여 이 작업을 수행할 수도 있습니다.

```
./startcommander.sh
Do you wish to create a new Database. (Yes (Y) or No (N))? [default: N]:
Enter product to connect with: (Embedded (E) or Server (S))? [default: E]: e
Enter driver to use? [default: [com.pointbase.jdbc.jdbcUniversalDriver]:
Enter database URL? [default: [jdbc:pointbase:embedded:sample]:
Enter Username? [default: PBPUBLIC]:
Enter Password? [default: PBPUBLIC]:
```

PointBase Commander 5.2 ECF build 294 size restricted version EMBEDDED

Interactive SQL command language. SunOS/5.9

(C) Copyright 2004 DataMirror Mobile Solutions, Inc. All rights reserved.

Licensed to: Sun_customer_demo_use
For commercial version contact PointBase at:
pointbase.com
PHONE: 1-877-238-8798 (US & CANADA)
1-408-961-1100 (International)
WEBSITE: www.pointbase.com

```
SQL> unload database sampled.sql;
SQL> unload database sampled.sql;
SQL> 13 Row(s) Unloaded. (PBPUBLIC.CUSTOMER_TBL)
SQL> 4 Row(s) Unloaded. (PBPUBLIC.DISCOUNT_CODE_TBL)
SQL> 30 Row(s) Unloaded. (PBPUBLIC.MANUFACTURE_TBL)
SQL> 11 Row(s) Unloaded. (PBPUBLIC.MICRO_MARKETS_TBL)
SQL> 9 Row(s) Unloaded. (PBPUBLIC.OFFICE_TBL)
SQL> 4 Row(s) Unloaded. (PBPUBLIC.OFFICE_TYPE_CODE_TBL)
SQL> 15 Row(s) Unloaded. (PBPUBLIC.ORDER_TBL)
SQL> 6 Row(s) Unloaded. (PBPUBLIC.PRODUCT_CODE_TBL)
SQL> 30 Row(s) Unloaded. (PBPUBLIC.PRODUCT_TBL)
SQL> 10 Row(s) Unloaded. (PBPUBLIC.SALES_REP_DATA_TBL)
SQL> 10 Row(s) Unloaded. (PBPUBLIC.SALES_REP_TBL)
SQL> 52 Row(s) Unloaded. (PBPUBLIC.SALES_TAX_CODE_TBL)
SQL> 12 Table(s) Unloaded.
SQL> quit;
```

위의 예에서 unload database 명령 실행 결과가 sampled.sql 파일에 기록됩니다. sampled.sql 파일에는 필수 테이블과 색인을 만드는 데 필요한 모든 DDL이 포함되어 있습니다. 또한, 데이터베이스에 데이터를 다시 삽입하기 위한 DML도 포함되어

있습니다. `commander` 명령 `RUN`은 생성된 스크립트를 사용하여 데이터를 다른 Pointbase 데이터베이스로 가져오는 데 사용됩니다. 다음은 생성된 파일에서의 `INSERT` 문 및 관련 데이터의 모양을 보여주는 예입니다.

```
INSERT INTO "ADVENTURE"."CATEGORY" (
"CATID", "LOCALE", "NAME", "DESCRIPTION", "IMAGEURI" )
VALUES( ?, ?, ?, ?, ? );
{
'ISLAND          ','en_US','Island Adventures','Experience an island /
paradise in a way fit for your needs.','Island_Adventures.gif'
'JUNGLE          ','en_US','Jungle Adventures','Experience a jungle /
paradise in a way fit for your needs.','Jungle_Adventures.gif'
'MOUNTAIN        ','en_US','Mountain Adventures','Experience an /
elevated paradise with a view.','Mountain_Adventures.gif'
'ORBITAL         ','en_US','Orbital Adventures','Experience a vacuum /
paradise with a beautiful view and where no one can hear you scream.',' /
'Space_Adventures.gif'
'WESTERN         ','en_US','Western Adventures','Enjoy the Wild West. /
','Western_Adventures.gif'
'SOUTH_POLE      ','en_US','South Pole Adventures','Experience a /
frozen paradise in a way fit for your needs.','SouthPole_Adventures.gif'
};
```

`commander unload database` 명령에서 생성된 파일이 DDL로만 구성되도록 쉽게 편집할 수 있습니다. 예를 들어, `insert` 문을 처리하는 프로그램을 쉽게 작성할 수 있습니다. 간단한 테스트를 위해 Pointbase의 `sample` 데이터베이스에 `unload database` 명령을 사용한 다음 생성된 스크립트를 편집하여 다음과 같이 변경했습니다.

- 모든 `CREATE Table` 문의 끝에서 `Organization Heap` 구문을 제거했습니다.
- `COMMIT` 명령을 제거했습니다.
- 부울 `datatype`을 `smallint`로 변경했습니다.
- 모든 `INSERT` 문과 관련 데이터를 제거했습니다.

다음으로 `sql` 대상을 사용하여 DDL을 실행하는 데 간단한 `Ant` 스크립트를 사용합니다. 마지막으로 `sun-appserv-samples` 데이터베이스에 대해 동일한 시험을 반복합니다. 여기서 생성된 `SQL` 파일을 다음과 같이 추가로 변경해야 합니다.

- 샘플 데이터베이스에 대해 위에서 설명한 모든 변경을 수행합니다.
- `create user` 명령을 제거합니다.
- `SET PATH` 명령을 제거합니다.
- `Decimal` 정밀도를 38에서 `max`인 31로 변경합니다.
- `float` 정밀도를 64에서 `max`인 52로 변경합니다.
- `CREATE PROCEDURE`에 대한 `SPECIFIC` 키워드는 현재 지원되지 않습니다.
- `GRANT` 명령을 제거합니다.

Java DB에서 작동하는 Pointbase Java 프로시저를 변환하려면 Java 코드와 `CREATE PROCEDURE` 문을 일부 변경해야 합니다. Java DB Java 프로시저 작성에 대한 정보는 [Derby](#)

Reference manual (<http://db.apache.org/derby/docs/10.1/ref/>)을 참조하십시오.
Boolean 데이터 유형은 Java DB의 다음 릴리스에서 지원됩니다.

지원되는 웹 서버

이 절에서는 Sun Java System Application Server Enterprise Edition 8.2에서 지원되는 Web Server 목록을 제공합니다.

표 2-3 지원되는 웹 서버

Web Server	버전	운영 체제
Sun Java System Web Server	6,0, 6.1, 7.0	Solaris SPARC 9, 10 Solaris x86 9, 10 Red Hat Enterprise Linux 3 및 4
Apache Web Server	1.3+, 1.4, 2.0	Solaris SPARC 9, 10 Solaris x86 10 Red Hat Enterprise Linux 3 및 4 Windows Server 2003 Windows 2000 Advanced Server SP4+ Windows Server 2000 SP4+ Windows XP Pro SP1+
Microsoft IIS™	5.0+	Windows Server 2003 Windows 2000 Advanced Server SP4+ Windows Server 2000 SP4+ Windows XP Pro SP1+

브라우저

이 절에서는 Sun Java System Application Server Enterprise Edition 8.2와 함께 지원되는 브라우저 목록을 제공합니다.

표 2-4 지원되는 웹 브라우저

브라우저	버전
Mozilla	1.4, 1.5, 1.6, 1.7.x
Netscape Navigator	4.79, 6.2, 7.0, 8.x
Internet Explorer	5.5 Service Pack 2, 6.0

표 2-4 지원되는 웹 브라우저 (계속)

브라우저	버전
Firefox	1.4, 1.5

HADB 요구 사항 및 지원되는 플랫폼

11 페이지 “하드웨어 및 소프트웨어 요구 사항”의 요구 사항뿐만 아니라 HADB를 실행하기 위한 아래의 요구 사항을 충족하는지 확인합니다.

주 - 11 페이지 “플랫폼 요구 사항”에 나열된 Application Server에 대한 시스템 요구 사항과 여기에 나열된 HADB에 대한 시스템 요구 사항은 꼭 동일하지 않습니다. 이는 설명서의 내용상 오류가 아니며, Application Server와 HADB 서버는 일반적으로 서로 다른 시스템에서 실행됩니다.

- 18 페이지 “지원되는 플랫폼”
- 18 페이지 “HADB 서버 호스트 요구 사항”
- 19 페이지 “HADB 관리 호스트 요구 사항”
- 19 페이지 “HADB 클라이언트 호스트 요구 사항”

주 - 시스템의 Java 구성 요소는 JDK 1.4.2_02로 구축되었고 JDK 1.5_09에서 테스트되었습니다.

지원되는 플랫폼

- **Solaris(SPARC)** – Solaris 8 MU7, Solaris 9 MU7, Solaris 10 RR
- **Solaris(x86)** – Solaris 9 MU7, Solaris 10 RR
- **RedHat Enterprise Linux** - 2.1 U5(ext3이 아닌 ext2 파일 시스템만 지원됨) 및 3.0 U4(ext2와 ext3 모두 지원됨. U4 이전의 업데이트는 과도한 스왑으로 인해 사용하지 않는 것이 좋음). HADB는 이러한 버전의 32비트 모드 운영 체제에서만 테스트 완료되었습니다. 또한 운영 체제의 버그 때문에 HADB에서는 64비트 모드로 실행되는 RedHat Enterprise Linux 3.0을 지원하지 않습니다. 이 문제가 HADB에 미치는 영향에 대한 자세한 내용은 49 페이지 “고가용성” 절에 있는 알려진 버그 6249685를 참조하십시오.
- **Microsoft Windows** – Microsoft Windows 2000 Advanced Server Service Pack 4 및 Microsoft Windows 2003 Enterprise Edition. HADB는 이후 버전의 모든 64비트 모드 Microsoft Windows 운영 체제를 지원하지 않습니다.

HADB 서버 호스트 요구 사항

- **최소 메모리** - 노드당 512MB
- **최소 여유 디스크 공간** - 호스트당 HADB 바이너리용 70MB. 데이터 장치를 위한 디스크 공간도 필요하며 노드당 테스트 설치를 위해 512MB가 필요합니다.
- **권장 메모리** - 노드당 1GB

- **권장 여유 디스크 공간** - 호스트당 HADB 바이너리용 70MB. 데이터 장치를 위한 디스크 공간도 필요하며 노드당 테스트 설치를 위해 1200MB가 필요합니다.

주 - HADB 데이터와 로그 파일을 저장하는 장치에는 쓰기 캐싱을 할 수 없습니다. 쓰기 캐싱은 기본적으로 Solaris x86 같은 일부 Solaris 플랫폼에서 사용할 수 있습니다.

HADB 관리 호스트 요구 사항

- **최소 메모리** - 128MB
- **최소 여유 디스크 공간** - 노드당 HADB 바이너리에 대해 70MB

HADB 클라이언트 호스트 요구 사항

- **최소 메모리** - 120MB
- **최소 여유 디스크 공간** - 20MB

Sun Java System Application Server 업그레이드

이전 Application Server 릴리스로부터의 인 플레이스 업그레이드는 지원되지 않습니다. Application Server의 이전 버전에서 현재 버전으로 업그레이드하는 자세한 방법은 **Application Server Enterprise Edition Upgrade and Migration Guide**를 참조하십시오.

기타 요구 사항

Sun Java System Application Server 소프트웨어를 설치하기 전에 충족되어야 하는 추가 요구 사항은 다음과 같습니다.

- **사용 가능 공간** — 임시 디렉토리에 Sun Java System Application Server 설치를 위한 35MB의 사용 가능 공간과 SDK 설치를 위한 250MB의 사용 가능 공간이 있어야 합니다.
- **설치 제거 프로그램 사용** — Application Server를 시스템에서 제거하려면 소프트웨어에 포함된 설치 제거 프로그램을 사용해야 합니다. 다른 방법을 사용하려고 시도한다면 동일한 버전을 다시 설치할 때 또는 새 버전을 설치할 때 문제가 발생합니다.
- **사용 가능 포트** — 사용하지 않은 포트가 7개 있어야 합니다.
 - 설치 프로그램에서 사용 중인 포트를 자동으로 감지하여 기본 설정을 위해 사용되고 있지 않은 포트를 제시합니다. 기본적으로 초기 기본 포트는 8080(HTTP용), 8181(HTTPS용) 및 4849(Administration Server용)입니다.
 - 설치 프로그램에서 사용된 포트를 감지하고 다음과 같이 두 개의 다른 포트를 지정합니다. - Sun Java System Message Queue(기본값은 7676) 및 IIOP(기본값은 3700(IIOP), 1060 및 1061(IIOP/SSL)). 이러한 기본 포트 번호가 사용 중인 경우에는 동적 포트 범위에서 임의의 포트 번호가 지정됩니다. 지정되는 포트 번호는 사용 가능한 다음 포트 번호가 아닐 수도 있습니다.

이미 설치한 서버 시작(UNIX) — 설치한 서버를 교체하는 경우가 아니라면, Sun Java System Application Server 8.2 설치 프로세스를 시작하기 전에 서버를 시작해야 합니다. 이렇게 해야 설치 프로그램에서 사용 중인 포트를 감지하여 다른 용도를 위해 할당하는 것을 피할 수 있습니다.

- **이미 설치한 서버 교체(UNIX)** — 이전 Sun Java System Application Server를 현재 Application Server로 교체하려면 새 서버를 설치하기 전에 이전 서버를 중지해야 합니다. 서버를 업그레이드하려면 설치 프로그램 업그레이드 마법사를 사용합니다.
- **방화벽 사용 안 함(Microsoft Windows)** — Sun Java System Application Server 소프트웨어를 설치하기 전에 방화벽 소프트웨어를 중지해야 합니다. 일부 방화벽 소프트웨어는 기본적으로 모든 포트를 사용 불가능하게 만들기 때문입니다. 설치 프로그램에서 어떤 포트를 사용할 수 있는지 정확히 판단할 수 있어야 합니다.

호환성에 대한 자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide**를 참조하십시오.

Enterprise Edition 8.2 릴리스에서 해결된 버그

이 절에서는 Sun Java System Application Server Enterprise Edition 8.2 제품에 대해 고객이 제기한 문제점 중 해결된 사항을 설명합니다.

버그 번호	설명
6368745	AS: AS7(Java ES 2)에서 AS8.2(Java ES 5)로 업그레이드할 수 없음
6432308	JES2에서 JES5로의 asupgrade 실패
6378409	AS 8.2: 8.2에 포함된 JSF 라이브러리로 인해 역방향 호환성이 중단됨
6371534	AS82EE: 설치 경로에 공백이 있을 경우 Windows에서 configure-ha-cluster가 중단됨
6242761	설명서에 따라 init를 사용하여 노드 에이전트를 시작하면 오류가 발생함
6267772	Borland Optimizelt에 대한 구성 지침이 올바르지 않음
6273226	-Xrs jvm 옵션을 추가하는 텍스트 설명을 추가하여 Windows 서비스로 실행되는 서버/NA 실행
6361145	8.1EE에서 8.2EE로 업그레이드하는 중에 LB 플러그인을 업그레이드할 수 없음
6362881	8.1ur2에서 8.2ee로 업그레이드할 때 설치 프로그램이 업그레이드 옵션을 표시하지 않음
6325988	FVD/codeBase의 첫 번째 들어오는 RMI-IIOP 요청 시 상호 운용성 문제 발생
6363689	JES5 ASEE8.2 build03 - 인스턴스를 중지할 수 없음

버그 번호	설명
6364900	1 웹 응용 프로그램에 두 번째 웹 응용 프로그램이 포함되어 있는 경우 패일오버 시 세션 값이 손실됨
6370993	클러스터에서 응용 프로그램 컨텍스트 루트가 "/"로 수정되면 세션 패일오버가 중지됨
6373729	ORB가 충돌하여 Appserver 8.1 코드가 WebLogic 9.0과 통신할 수 없음
6377594	Weblogic initialcontext 팩토리에 조희 문제 발생
6381538	독립 실행형 클라이언트가 NPE와 함께 실패
6406055	경고: "IOP00110205: (BAD_PARAM) 외래 ORB에서 객체 참조를 가져왔습니다." org.omg.CORBA.BAD_PARAM
6388329	Access Manager를 업그레이드한 후 Application Server에서 JSP 컴파일 오류가 발생함
6419659	transport-guarantee가 CONFIDENTIAL일 때 요청이 LB 플러그인에 의해 올바르게 리디렉션되지 않음
6390584	OutOfMemoryError: PermGen space
6401424	바이트로 요청될 때 libns-httpd40.so에 있는 service_plain_range의 SEGV가 PDF 파일로 사용됨
6401704	AppServer 8.#에 대한 WebDAV 지원 필요
6416478	jsp testsuite 오류: javax.servlet.jsp.el.ELEXception
6438908	relativeRedirectAllowed=true인 경우 헤더 위치 손상
6456553	응답에 쿠키를 추가할 때 java.lang.IllegalArgumentException 발생
6295010	고정 풀에서 이루어지는 연결의 경우는 방화벽과 충돌할 가능성이 있으므로 유효 시간 초과를 확인하지 않음
6350435	두 데이터베이스에 대한 XA 작업 중에 Application Server가 데이터베이스의 오류를 처리하지 못함
6377830	다음 사용자가 동일한 연결을 사용할 경우 setAutoCommit=false가 전파됨
6399830	IT 319: domain.xml에서 비밀번호 별칭 기능이 작동하지 않음
6360040	SJAS 8.x: AppServer LDAP 영역 바인드 사용자가 모든 그룹과 구성원에 액세스하는 경향이 있음
6370095	acceptor-thread를 10보다 크게 설정할 수 없음
6399365	InvokerServlet이 Enterprise Edition에서만 작동하지 않음
6303835	과도한 로깅: 서버 로그에 잘못된 보안 메시지 기록
6349541	8.1 EE UR2 - SSL 수신기를 특정 IP 주소에 바인드하도록 설정할 수 없음

버그 번호	설명
6380040	로그 파일의 자동 정리 필요
6387278	클라이언트 인증이 중단되었거나 스레드에 안전하지 않음(ProgrammaticLogin)
6407896	getUserPrincipal()을 대체하는 HttpServletRequestWrapper에서 ClassCastException 발생
6321194	라운드 로빈 정책이 작동하지 않음
6362269	설치 경로에 공백이 있을 경우 Windows에서 검증자가 올바르게 작동하지 않음
6365888	기본 커넥터 연결 풀의 연결이 트랜잭션에 나열되지 않음
6369554	연결 풀에서 응용 프로그램에 연결하기 전에 연결을 검증해야 함
6370574	나중에 구성(Configre Later)을 사용하여 AS를 업그레이드한 후 /var/opt/SUNWappserver 디렉토리가 없음
6371723	lbplugin이 모든 웹 서버 버전에 대해 메모리 누출(Apache mod_loadbalancer의 경우 더 많이 메모리 누출)
6395390	패일오버되는 http 요청 시 라운드 로빈이 발생하지 않음
6402713	로드 밸런서가 HTTPS 요청에 연결하지 못함
6409992	인증서를 사용하여 8.1pe에서 8.2EE로 업그레이드되지 않음
6413224	업그레이드 도구가 인증서 업그레이드 옵션을 건너뛸
6422893	HTTPS 라우팅이 작동하지 않음
6424051	8.xPE에서 9.1 EE로 업그레이드할 때 기존의 관리자 자격 증명과 MP를 사용해야 함
6424053	8.XEE에서 9.1EE로 업그레이드할 때 start-domain 예외가 발생하며 실패
6430394	네트워크 중단 발생 시 메시지가 손실됨
6444052	JMS 버전 1.5의 일반 RA를 AS 8.2 EE로 통합
6444308	AS 8.1 UR2 EE-> 8.2 EE SS: 8.2의 domain1을 시작할 수 없음. 8.1UR2의 도메인 시작이 잘못되었음
6444368	win2003 병렬 GUI에서 8.0PE UR1에서 9.1 ee로의 업그레이드가 중단됨
6446558	connector-connection-pool 자원에 대해 수동 트랜잭션 복구가 작동하지 않음
6447895	포함된 RA를 사용하여 자원에 대한 트랜잭션 복구가 작동하지 않음
6454007	업그레이드 도구에 필요한 입력 변경

버그 번호	설명
6455396	8.1EE에서 9.1EE SBS로의 업그레이드 후에 노드 에이전트와 인스턴스가 시작하지 않음
6374533	성능 및 안정성의 이유로 Application Server가 XWSS 1.0 대신 XWSS 1.1을 번들로 제공해야 함
6358422	Appserver 7.1/8.1 EE: 웹 서버 LB 프록시 플러그인이 연결 유지 연결을 올바르게 지원해야 함
6382063	com.sun.enterprise.iioop.IORToSocketInfoImpl에서 메모리 누출

추가 HADB 정보

이 절에서는 Application Server 8.2에 포함된 HADB 구현에 관한 중요한 추가 정보를 설명합니다.

- 23 페이지 “HADB의 향상된 기능”
- 24 페이지 “HADB 파일 시스템 지원”
- 25 페이지 “고가용성 데이터베이스 업그레이드”
- 31 페이지 “알려진 SQL 제한 사항”
- 32 페이지 “고가용성 로드 균형 조정”

HADB의 향상된 기능

- 새 관리 명령 `hadbm setadminpassword`가 데이터베이스 관리에 사용되는 암호를 변경할 수 있도록 구현되었습니다. 이 명령에는 사용할 관리 에이전트와 이전 암호 및 새 암호를 나타내는 옵션이 있습니다. 자세한 내용은 `hadbm setadminpassword` 설명서 페이지를 참조하십시오.
- 기존 관리 명령 `hadbm listpackages`가 수정되었습니다. 이전의 명령에서는 피연산자를 가져가지 않았으며 해당 관리 도메인에 모든 패키지를 나열했습니다. 수정한 결과 패키지 이름 피연산자를 옵션으로 제공하고 해당 이름의 패키지만 나열됩니다. 피연산자를 제공하지 않으면 모든 패키지가 나열됩니다. 자세한 내용은 `hadbm listpackages` 설명서 페이지를 참조하십시오.
- 기존 관리 명령 `hadbm createdomain`이 수정되었습니다. `hostlist` 피연산자가 확장되어 관리 에이전트의 포트 번호도 지정합니다. 이와 같이 도메인은 `hostlist` 피연산자만 사용하여 완전히 지정됩니다. 이전 버전과의 호환성을 위해 이전 동작은 여전히 지원됩니다. 자세한 내용은 `hadbm createdomain` 설명서 페이지를 참조하십시오.
- 관리 시스템의 일부 오류 메시지가 수정되었습니다. 오류 메시지의 이해도, 일관성 및 정확도를 개선하기 위해 수정이 이뤄졌습니다. 실제 수정한 내용은 이 릴리스 노트에 없습니다.
- 설치와 제거 동작이 약간 변경되었습니다. HADB를 설치하거나 제거할 때 항상 소프트웨어 링크 `/opt/SUNWhadb/4`를 유지해야 하지만 그렇지 않은 경우가 있습니다.

- 명령줄에 명령 옵션으로 암호를 입력할 가능성이 없어졌습니다. 이는 암호를 명령줄 옵션으로 가져가는 모든 `hadbm` 명령의 경우와 관련이 있습니다. `hadbm` 명령의 경우 이전까지 암호는 다음과 같은 형식으로서 입력할 수 있었습니다.

1. 암호 파일
2. 명령줄 옵션
3. 대화식 입력

두 번째 방법인 명령줄 옵션은 안전하지 않은 것으로 간주되어 더 이상 지원되지 않습니다. 이런 식으로 암호를 입력하면 경고 메시지가 표시됩니다. 대신 첫 번째 방법인 암호 파일 또는 세 번째 방법인 대화식 입력을 사용합니다. 명령줄에 암호를 사용하는 방법은 다음 릴리스에서는 더 이상 사용하지 않습니다. 이는 명령줄 암호 옵션을 사용하는 모든 `hadbm` 명령에 적용됩니다.

- HADB는 JGroups 2.2를 사용하도록 업그레이드되었고 해당 소스 코드는 HADB와 함께 배포되었습니다. 이전 HADB 버전으로부터 온라인 업그레이드를 지원하기 위해 JGroups 2.1과 2.2가 HADB와 함께 제공됩니다. JGroups 2.1의 경우 바이트 코드만 제공됩니다.

HADB 파일 시스템 지원

다음 파일 시스템 중 하나를 사용하기 위해 HADB를 구성하려면 몇 가지 중요한 사항을 고려해야 합니다.

- **ext2 및 ext3**– HADB는 Red Hat Application Server 3.0을 위한 ext2와 ext3 파일 시스템을 지원합니다. Red Hat Application Server 2.1의 경우에는 ext2 파일 시스템만 지원합니다.
- **Veritas**– Veritas 파일 시스템을 Solaris 플랫폼에서 사용하면 “WRN:Direct disk I/O mapping failed”라는 메시지가 내역 파일에 기록됩니다. 이 메시지는 HADB에서 데이터 및 로그 장치에 대해 직접 I/O를 사용할 수 없다는 것을 의미합니다. 직접 I/O는 디스크 페이지를 기록하는 CPU 비용을 절감하는 성능 향상 기능입니다. 이 기능을 사용하면 운영 체제에서 사용한 데이터 페이지를 관리하는 오버헤드를 줄일 수 있습니다.

직접 I/O를 Veritas 파일 시스템과 함께 사용하려면 다음 중 한 가지 작업을 수행합니다.

- `mincache=direct` 옵션을 사용하여 마운트한 파일 시스템에 데이터 및 로그 장치를 생성합니다. 이 옵션은 파일 시스템에 생성된 모든 파일에 적용됩니다. 자세한 내용은 `mount_vxfs(1M)` 명령을 참조하십시오.
- Veritas Quick I/O 기능을 사용하여 파일 시스템에 대한 원시 I/O를 수행합니다. 자세한 내용은 **VERITAS File System 4.0 Administrator's Guide for Solaris**를 참조하십시오.

이러한 구성은 Application Server 8.2에서 테스트되지 않았습니다.

Application Server 소프트웨어와 함께 HADB를 설치하고 구성하는 데 대한 자세한 내용은 **Application Server Enterprise Edition 고가용성 관리 설명서**를 참조하십시오.

고가용성 데이터베이스 업그레이드

- 25 페이지 “사전 업그레이드 작업/데이터 마이그레이션”
- 25 페이지 “업그레이드 절차”
- 26 페이지 “업그레이드 테스트”
- 27 페이지 “특수 배포 및 업그레이드 정보”

▼ 사전 업그레이드 작업/데이터 마이그레이션

시작하기 전에 사용자는 HADB 내역 파일, 관리 에이전트 구성 파일, 로그 파일 및 리포지토리, 그리고 모든 데이터 장치를 설치 경로 밖에 두어야 합니다. 그렇지 않은 경우 업그레이드 전에 이를 실시해야 합니다. 관리 리포지토리 및 구성 파일을 이동하는 방법은 다음과 같습니다.

- 1 이전 관리 에이전트를 모두 중단하고 HADB 모드를 실행 상태로 유지합니다.
- 2 각 호스트에서 리포지토리 디렉토리를 새 위치로 이동합니다.
- 3 각 호스트에서 dbconfig 디렉토리를 새 위치로 복사합니다.
- 4 각 호스트에서 mgt.cfg 파일을 업데이트하고, dbconfig 및 리포지토리 디렉토리의 정확한 경로를 설정합니다.
- 5 업데이트된 mgt.cfg 파일을 사용하여 관리 에이전트를 시작합니다.

▼ 업그레이드 절차

HADB 4.4.x에서 4.4.3으로 업그레이드하는 방법은 다음과 같습니다.

- 1 위에서 설명한 사전 업그레이드 작업을 필요한 단계 만큼 수행합니다.
- 2 HADB 4.4.3 버전을 모든 HADB 호스트의 4.4.x 버전 경로가 아닌 /opt/SUNWhadb/4.4.3 등의 다른 경로에 설치합니다.
- 3 HADB 호스트와 다른 경우 HADB 4.4.3 버전을 hadbm 클라이언트 호스트에 설치합니다.
- 4 모든 HADB 호스트에서 실행 중인 관리 에이전트를 모두 중단합니다.
- 5 새 버전의 소프트웨어를 사용하여 이전 구성 파일로 관리 에이전트 프로세스를 시작합니다. 나머지 단계에서는 새 버전의 bin 디렉토리에 있는 hadbm 명령을 사용합니다.
- 6 관리 도메인에 패키지를 등록합니다. (기본 패키지 이름은 V4.4가 되므로 동일한 이름의 기존 패키지와 충돌을 방지하기 위해 다른 패키지 이름이 필요할 수 있습니다.)

```
hadbm registerpackage --packagepath=/opt/SUNWhadb/4.4.3 V4.4.3
```

- 7 `hadbm listpackages` 명령을 실행하고 새 패키지가 도메인에 등록되어 있는지 확인합니다.
- 8 새 `hadbm 4.4.3` 버전으로 데이터베이스를 다시 시작합니다. 장치 및 내역 파일을 이동해야 하는 경우 다음과 같이 장치와 내역 파일의 새 경로 설정과 결합된 온라인 업그레이드를 한 번에 실행합니다.

```
hadbm set packagename=V4.4.3,devicepath=new_devpath,  
historypath=new_histpath
```

그러나 장치 및 내역 파일이 이미 설치 디렉토리의 외부에 있다면 노드에 대해 롤링 재시작만 수행하는 다음 명령을 실행합니다.

```
hadbm set packagename=V4.4.3 database name
```

- 9 `hadbm status` 명령을 사용하여 데이터베이스 상태가 "실행 중"인지 확인하고, 정상적으로 작동하여 클라이언트 트랜잭션을 제공하는지 확인합니다.
 - 10 모두 제대로 작동한다면 나중에 이전 설치를 제거할 수 있습니다. 이전 패키지를 등록 취소하기 전에 `ma` 리포지토리에서 이전 패키지에 대한 모든 참조를 제거합니다. 그렇지 않으면 `hadbm unregisterpackage`가 "사용 중인 패키지"로 인해 실패합니다. 예를 들어, 가상 재구성 작업인 `hadbm set connectiontrace=same as previous value`는 이전 패키지에 대한 모든 참조를 제거합니다. 이제 다음 명령으로 이전 패키지의 등록을 취소합니다.
- ```
hadbm unregisterpackage [--hosts=host-list] old package name
```
- 11 파일 시스템에서 이전 설치를 제거합니다.

## ▼ 업그레이드 테스트

업그레이드가 성공적이었는지 테스트하기 위해 업그레이드가 제대로 수행되었는지 확인합니다(Solaris만 해당).

- 1 실행 중인 프로세스는 새 바이너리를 사용해야 합니다. 모든 HADB 노드에서 다음을 확인합니다.

```
new path/bin/ma -v
new path/bin/hadbm -v
```

- 2 데이터베이스가 실행 중인지 확인합니다. 다음 명령은 모든 HADB 노드가 "실행 중"인 상태인지를 보여주어야 합니다.

```
new path/bin/hadbm status -n
```

- 3 HADB를 사용하는 제품의 포인터가 새 HADB 경로를 가리키도록 변경되었는지 확인합니다.

#### 4 HADB를 사용하는 제품은 업그레이드 테스트를 실행하여 HADB 업그레이드가 작동하는지도 확인할 수 있습니다.

온라인 업그레이드 후에 새 버전이 제대로 작동하지 않을 경우 다시 이전 HADB 버전을 사용합니다. 그러나, 관리 에이전트 리포지토리가 변경되었다면 HADB 자체는 다운그레이드될 수 있지만 새 관리 에이전트가 실행 중이어야 합니다.

#### 특수 배포 및 업그레이드 정보

이 절에서는 HADB 배포 및 업그레이드에 관한 추가 정보를 제공합니다.

- 27 페이지 “배포”
- 30 페이지 “4.4.1에서 4.4.2로 온라인 업그레이드”

#### 배포

- 로컬 디스크의 저장 장치, 로그 및 내역 파일만 원격으로 마운트된 파일 시스템을 사용하지 않습니다.
- 호스트에 두 개 이상의 노드가 있을 경우 각 노드에 속한 장치를 각각 다른 디스크에 두십시오. 그렇지 않으면 디스크 충돌로 성능이 저하됩니다. 이 문제의 증상은 내역 파일의 BEWARE - last flush/fputs took too long과 같은 메시지로 확인할 수 있습니다. 단일 노드에 두 개 이상의 데이터 장치 파일이 있다면 이 장치 파일에 각각 다른 디스크를 사용하는 것이 좋습니다.
- 로컬 디스크(데이터 장치에 사용되는 디스크 외의 개별 디스크)를 사용하여 HADB 호스트에 HADB 바이너리를 설치합니다. NFS 지연 또는 디스크 충돌로 인해 내역 파일에서 “Process blocked for nnn, max block time is nnn”이라는 경고와 함께 노드가 다시 시작할 수 있습니다.
- HADB 장치, 내역 파일, 관리 에이전트 디렉토리 및 에이전트 구성 파일은 HADB 패키지 경로에 두지 마십시오. 그렇지 않으면 새 버전으로 업그레이드할 때와 이전 패키지 경로를 삭제할 때 문제가 생길 수 있습니다.
- HADB의 이 릴리스는 최대 28개 노드에 대해 지원됩니다. 이 중 24개는 활성 데이터 노드이고 4개는 예비 노드입니다.
- JDBC 드라이버와 HADB 서버에 동일 버전을 사용하는 것이 좋습니다.
- IPv6이 아닌 IPv4만 지원합니다.
- Windows에서 명령줄 길이는 2048바이트 수로 제한됩니다.
- 네트워크는 UDP 멀티캐스트를 위해 구성되어야 합니다.
- RedHat Enterprise Linux 3.0 업데이트 1에서 3까지에서 과도한 스와핑이 발견되었으므로 이것을 배포 플랫폼으로 권장하지 않습니다. 이 문제는 RedHat Enterprise Linux 3.0 업데이트 4에서 해결되었습니다.
- 실시간 우선 순위로 NSUP를 실행할 수 있습니다.

노드 수퍼바이저(NSUP) 프로세스(c lu\_nsup\_srv)는 "하트 비트" 메시지를 적절한 시기에 교환할 수 있도록 하여 HADB의고가용성을 보장합니다. 타이밍은 NSUP가 자원 고갈을 일으키는 다른 프로세스와 같은 곳에 놓여질 때 영향을 받습니다. 그

결과 잘못된 네트워크 분할이 이루어지고 노드가 재시작되어(내역 파일에 “Process blocked for n seconds” 경고가 표시된 다음) 트랜잭션과 기타 예외가 중단됩니다.

이 문제를 해결하려면 `clu_nsup_srv` (`installpath/lib/server`에 있음)에 `suid` 비트 세트가 있어야 하며 파일은 루트로 소유해야 합니다. 이는 다음 명령을 사용하여 직접 수행할 수 있습니다.

```
chown root clu_nsup_srv
chmod u+s clu_nsup_srv
```

이로 인해 시작 시 `clu_nsup_srv` 프로세스가 `root` 사용자로 실행되고 시작된 이후에는 프로세스에서 자동으로 실시간 우선 순위가 자체 제공됩니다. `setuid` 사용으로 인해 보안에 영향을 미치지 않도록 처음에 실시간 우선 순위가 설정되고, 우선 순위가 변경되면 프로세스가 유효한 `uid`로 돌아갑니다. 다른 HADB 프로세스는 시간 공유 우선 순위에 따라 우선 순위를 낮춥니다.

NSUP가 실시간 우선 순위를 설정할 수 없는 경우 "Could not set realtime priority"(unix: errno will be set to EPERM)라는 경고 메시지가 나타나고 실시간 우선 순위 없이 계속됩니다. 해당 경고는 `ma.log` 파일에 기록됩니다.

실시간 우선 순위를 설정할 수 없는 경우는 예를 들면 다음과 같습니다.

- Solaris 10 비전역 영역에 설치된 경우
- `PRIV_PROC_LOCK_MEMORY`(프로세스가 물리적 메모리에서 페이지를 잠그도록 허용) 및/또는 `PRIV_PROC_PRIOCNTRL` 권한이 Solaris 10에서 호출된 경우
- 사용자가 `setuid` 권한을 끈 경우
- 사용자가 소프트웨어를 `tar` 파일(App.server에 대한 `nonroot` 설치 옵션)로 설치한 경우

`clu_nsup_srv` 프로세스는 CPU를 사용하지 않고 용량이 작으며 실시간 우선 순위로 실행해도 성능에는 영향을 주지 않습니다.

- Solaris에서 HADB에 대한 IP 네트워크 다중 경로를 구성합니다(Solaris 9에서만 테스트됨).

HADB를 실행 중인 Solaris 호스트는 가능한 최대 네트워크 가용성을 보장하기 위해 네트워크 다중 경로로 설정하는 것이 좋습니다. 네트워크 다중 경로 설정은 **IP Network Multipathing Administration Guide**에 자세히 설명되어 있습니다. HADB와 함께 다중 경로를 사용할 경우, 아래에 설명된 대로 HADB의 다중 경로 설정을 채택하기 전에 다중 경로를 설정하기 위해 **IP Network Multipathing Administration Guide**의 Administering Network Multipathing 절을 참조하십시오. **IP Network Multipathing Administration Guide**는 Solaris 9 System Administrator Collection의 일부로, <http://docs.sun.com>에서 다운로드할 수 있습니다.

- 네트워크 인터페이스 실패 감지 시간 설정

HADB가 다중 경로 페일오버를 올바르게 지원하려면 네트워크 인터페이스 실패 감지 시간이 /etc/default/mpathd의 FAILURE\_DETECTION\_TIME 매개 변수에서 지정된 1000밀리초를 초과해서는 안 됩니다. 원래 값이 큰 경우 파일을 편집하여 이 매개 변수의 값을 1000으로 변경합니다.

```
FAILURE_DETECTION_TIME=1000
```

변경된 내용을 적용하려면 다음 명령을 호출합니다.

```
pkill -HUP in.mpathd
```

#### ■ HADB에서 사용할 IP 주소

**Solaris IP Network Multipathing Administration Guide**에 설명된 대로 다중 경로는 물리적 네트워크 인터페이스를 다중 경로 인터페이스 그룹으로 그룹화하는 것입니다. 해당 그룹 내의 물리적 인터페이스는 물리적 인터페이스 주소와 테스트 주소 등 연관된 IP 주소가 두 개씩 있습니다. 물리적 인터페이스 주소는 데이터를 전송하는 데 사용되고 테스트 주소는 Solaris 내부용으로만 사용됩니다. `hadbm create --hosts`가 실행되면 각 호스트는 다중 경로 그룹에서 물리적 인터페이스 주소 하나만으로 지정되어야 합니다.

#### ■ 예제

호스트 1과 호스트 2에 물리적 네트워크 인터페이스가 두 개씩 있다고 가정합니다. 각 호스트에서 이러한 두 인터페이스는 다중 경로 그룹으로 설정되어 있으며 `ifconfig -a`를 실행하면 다음 결과가 생성됩니다.

##### 호스트 1

```
bge0: flags=1000843<mtu 1500 index 5 inet 129.159.115.10 netmask fffffff0
broadcast 129.159.115.255 groupname mp0
bge0:1: flags=9040843<mtu 1500 index 5 inet 129.159.115.11 netmask fffffff0
broadcast 129.159.115.255
bge1: flags=1000843<mtu 1500 index 6 inet 129.159.115.12 netmask fffffff0
broadcast 129.159.115.255 groupname mp0
bge1:1: flags=9040843<mtu 1500 index 6 inet 129.159.115.13 netmask ff000000
broadcast 129.159.115.255
```

##### 호스트 2

```
bge0: flags=1000843<mtu 1500 index 3 inet 129.159.115.20 netmask fffffff0
broadcast 129.159.115.255 groupname mp0
bge0:1: flags=9040843<mtu 1500 index 3 inet 129.159.115.21 netmask ff000000
broadcast 129.159.115.255
bge1: flags=1000843<mtu 1500 index 4 inet 129.159.115.22 netmask fffffff0
broadcast 129.159.115.255 groupname mp0
bge1:1: flags=9040843<mtu 1500 index 4 inet 129.159.115.23 netmask ff000000
broadcast 129.159.115.255
```

여기서, 두 호스트의 물리적 네트워크 인터페이스는 bge0과 bge1로 나열되어 있습니다. bge0:1과 bge1:1로 나열된 인터페이스는 **IP Network Multipathing Administration Guide**에 설명된 대로 다중 경로 테스트 인터페이스(이에 따라 ifconfig 출력에 DEPRECATED로 표시됨)입니다.

이 환경에서 HADB를 설정하려면 각 호스트에서 물리적 인터페이스 주소 하나를 선택합니다. 이 예제의 호스트 1에서 129.159.115.10을 선택하고 호스트 2에서 129.159.115.20을 선택합니다. 호스트당 데이터베이스 노드가 한 개인 데이터베이스를 만들려면 hadbm create에 다음 인수를 사용합니다.

```
--host 129.159.115.10,129.159.115.20
```

각 호스트에 데이터베이스 노드가 두 개인 데이터베이스를 만들려면 다음 인수를 사용합니다.

```
--host 129.159.115.10,129.159.115.20,129.159.115.10,129.159.115.20
```

두 경우 모두 두 호스트의 ma.server.mainternal.interfaces 변수를 129.159.115.0/24로 설정해야 합니다.

#### 4.4.1에서 4.4.2로 온라인 업그레이드

4.2 또는 4.3에서 4.4로 온라인 업그레이드할 수 없습니다. 그러나, 4.4에서는 향후 버전에 대해 온라인 업그레이드를 지원합니다. 4.4.1에서 4.4.2로 업그레이드하려면 다음 단계를 수행합니다.

1. 모든 HADB 호스트(/opt/SUNWhadb/4.4.2-6와 같이 4.4.1과 다른 경로)에 4.4.2를 설치합니다.
2. hadbm client 호스트에 새 버전을 설치합니다.
3. HADB 호스트에서 실행 중인 모든 관리 에이전트를 중단합니다.
4. 새 버전의 소프트웨어를 사용하여 이전 구성 파일로 관리 에이전트 프로세스를 시작합니다. 나머지 단계에서는 새 버전의 bin 디렉토리에 있는 hadbm 명령을 사용합니다.
5. 관리 도메인에 패키지를 등록합니다. (여기서 기본 패키지 이름은 V4.4가 되므로 동일한 이름의 기존 패키지와 충돌을 방지하기 위해 다른 패키지 이름이 필요할 수 있습니다.)

```
hadbm registerpackage --packagepath=/opt/SUNWhadb/4.4.2-6 V4.4.2
```

6. 새 버전으로 데이터베이스를 다시 시작합니다. (다음 명령은 노드에 대해 롤링 재시작을 수행합니다.)

```
hadbm set packagename=V4.4.2 database_name
```

7. (hadbm status 명령을 사용하여) 데이터베이스 상태가 “실행 중”인지 그리고 정상적으로 기능하여 클라이언트 트랜잭션을 제공하는지 확인합니다.
8. 모두 제대로 작동한다면 나중에 이전 설치를 제거할 수 있습니다.

이전 패키지의 등록을 취소하기 전에 `ma` 리포지토리에서 이전 패키지에 대한 모든 참조를 제거합니다. 그렇지 않으면 `hadbm unregisterpackage`는 "사용 중인 패키지"로 인해 실패합니다. 예를 들어 가상 재구성 작업인 `hadbm set connectiontrace=<same_as_previous_value>`는 이전 패키지에 대한 모든 참조를 제거합니다. 이제 다음 명령으로 이전 패키지의 등록을 취소합니다.

```
hadbm unregisterpackage [--hosts=<host_list>] <old_package_name>
```

HADB 설치 지침 (<http://clustra.norway.sun.com/intraweb/download/products/hadb/packages/pdf/4.4.2-6.pdf>)에 설명된 대로 파일 시스템에서 이전 설치를 제거합니다.

### 알려진 SQL 제한 사항

- 테이블에 UNIQUE 보조 색인을 만들 수 없습니다.
- 표현식 (DISTINCT column)은 유일하게 선택된 표현식인 경우에만 종합 표현식에 허용됩니다.
- 모든 테이블은 기본 키 사양을 사용해서 만들어야 합니다. (즉, 기본 키가 없는 테이블은 지원되지 않습니다.)
- FULL OUTER JOIN은 지원되지 않습니다.
- 다음과 같이 테이블 하위 쿼리인 IN 하위 테이블은 지원되지 않습니다.

```
SELECT SNAME FROM S WHERE (S1#,S2#) IN (SELECT S1#,S2# FROM SP
WHERE P#='P2')
```

- NOT NULL 및 PRIMARY KEY 외의 제약 조건은 지원되지 않습니다.
- 새 소유자를 자원에 할당할 수 있지만 이때 현재 소유자에게 부여된 권한이 새 소유자에게는 부여되지 않습니다.
- 각 하위 쿼리가 쿼리의 외부 수준과 (직접적인) 상관 관계가 없는 두 개 이상의 중첩된 NOT EXISTS 하위 쿼리는 지원되지 않습니다.
- 열 권한은 지원되지 않습니다.
- 행 값 구성자는 VALUES 절에만 허용됩니다.
- 하위 쿼리는 행 값 구성자에서 값 표현식으로 허용되지 않습니다.
- 다음 데이터 유형은 기본 키를 만들 때 사용할 수 없습니다.
  - REAL
  - FLOAT
  - DOUBLE PRECISION
  - DECIMAL
  - NUMERIC



## 고가용성 로드 균형 조정

Application Server에는 HTTP, IIOP 및 JMS 클라이언트에 대한 로드 균형 조정, HTTP 세션 페일오버 지원, EJB 클러스터링 및 페일오버 지원, 고가용성 EJB 타이머, 분산 트랜잭션 복구, 롤링 응용 프로그램 업그레이드, J2EE 응용 프로그램의 일시적 상태 저장용 고가용성 데이터베이스가 포함되어 있습니다.

가용성은 클러스터에서 Application Server 인스턴스의 페일오버 보호를 가능하게 합니다. 하나의 Application Server 인스턴스가 중지될 경우 사용할 수 없게 된 서버에 할당되었던 세션을 다른 Application Server 인스턴스에서 인수합니다. 세션 정보는 HADB에 저장됩니다. HADB는 HTTP 세션, Stateful Session Bean 및 단일 사인 온 자격 증명의 지속성을 지원합니다.

## 호환성 문제

Sun Java System Application Server Enterprise Edition의 다음 주요 릴리스에서 다음 비호환성 문제가 발생합니다.

- HTTP 서비스에서는 성능을 향상하기 위해 계속해서 DNS 캐시를 사용하지만 DNS 캐시에 대한 모니터링은 지원되지 않습니다.
- HTTP 파일 캐싱에 대한 지원이 개정되어 구성과 모니터링이 변경됩니다.
- 액세스 로그 회전 접미어의 형식은 <http://java.sun.com/j2se/1.5.0/docs/api/java/text/SimpleDateFormat.html>에 지정된 대로 날짜 및 시간 객체에서 지원하는 형식으로 변경됩니다. 이 릴리스에서 기본값 “%YYYY;%MM;%DD;-%hh;h:mm;m:ss;s”는 계속 지원되지만 다른 변형은 지원되지 않습니다.
- 더 이상 지원되지 않는 domain.xml 요소, 속성 및 특성은 서버 로그에 경고로 표시되고 업그레이드 로그 파일에는 생략됩니다.
- server.http-service.dns 노드는 더 이상 모니터링 뷰에서 사용할 수 없습니다.
- server.http-service.file-cache 노드에서 일부 속성이 제거될 수 있습니다. 결과적으로 이런 노드에서 제거된 속성을 액세스하려고 하는 asadmin 모니터링 명령은 실패합니다.

## Deploytool

Deploytool은 더 이상 사용할 수 없습니다. 동일한 기능이 NetBeans IDE에 있습니다. 자세한 내용은 <http://www.netbeans.org/kb/>에서 NetBeans 4.1용 J2EE 1.4 자습서를 참조하십시오.

## 검증자

- (verifier -u로 호출된) 검증자 GUI 모드는 더 이상 사용할 수 없습니다. 동일한 기능이 NetBeans IDE에 있습니다.



- 검증자 도구를 사용할 때 응용 프로그램 검증의 기본 모드는 "J2EE 규칙 검증"에서 "J2EE 규칙 및 Sun Application Server 구성 규칙 검증"으로 변경됩니다. 즉, 기본적으로 검증자는 응용 프로그램이 J2EE 규칙을 충족하고 Sun Application Server에서 실행되도록 구성되었는지 여부를 테스트합니다. 검증자 명령은 J2EE 규칙에 대해서만 응용 프로그램을 테스트하는 명령줄 스위치를 가지고 있습니다.

### 클래스로더 변경 사항

현재 릴리스에서 `domain.xml`(응용 프로그램 서버 구성 파일)의 `classpath-prefix`, `server-classpath`, `classpath-suffix` 등의 속성에 추가된 JAR 및 디렉토리 항목은 JVM 시스템 클래스 경로에서 사용할 수 있습니다. 이 동작에 따라 응용 프로그램은 `java.lang.ClassLoader` 클래스에서 다음 방법을 사용하여 JVM 시스템 클래스 경로에서 클래스나 다른 자원에 액세스할 수 있습니다.

- `getSystemClassLoader()`
- `getSystemResource()`
- `getSystemResourceAsStream()`
- `getSystemResources`

다음 주요 릴리스에서 `classpath-prefix`, `server-classpath`, `classpath-suffix` 등에 추가된 JAR 및 디렉토리 항목은 더 이상 JVM 시스템 클래스 경로에서 사용할 수 없습니다. 응용 프로그램에서 위의 방법 중 하나를 사용할 경우에는 해당 자원이 시스템 클래스 경로에서 사용 가능하다는 조건이 필요 없는 해당 방법을 사용하는 것이 좋습니다. JVM 시스템 클래스 경로에 영향을 받지 않는 해당 방법은 `java.lang.ClassLoader`에서 사용할 수 있으며 가능한 경우 다음과 같이 사용해야 합니다.

#### 예 2-1 이전 코드

```
java.net.URL url = ClassLoader.getSystemResource
("com/acme/tools/tools.properties");
```

#### 예 2-2 권장하는 변경된 내용

```
java.net.URL url = this.getClass().getClassLoader().getResource
("com/acme/tools/tools.properties");
```

코드를 변경할 수 없다면 다음 릴리스에 추가될 새 구성 옵션을 선택적으로 사용하여 JVM 시스템 클래스 경로를 설정할 수 있습니다.

### 웹 서비스 보안 구성

웹 서비스 보안은 `wss-client-config.xml`과 `wss-server-config.xml` 파일을 사용하여 구성할 수 있습니다. 이러한 구성 파일의 내용과 이름은 불안정하여 변경되기 쉽습니다. 동일한 기능을 계속 사용할 수 있습니다.

## J2EE 지원

Sun Java System Application Server Enterprise Edition 8.2는 J2EE 1.4 플랫폼을 지원합니다. 다음 표에서는 J2EE 1.4 플랫폼에서 사용할 수 있는 향상된 API에 대해 설명합니다.

표 2-5 J2EE 1.4 플랫폼에서 사용할 수 있는 API

| API                                                        | 설명                                                                                               |
|------------------------------------------------------------|--------------------------------------------------------------------------------------------------|
| 구성 요소                                                      |                                                                                                  |
| 응용 프로그램 및 응용 프로그램 클라이언트                                    | XML 스키마를 사용한 표준 배포 설명자 구현                                                                        |
| Enterprise JavaBeans(EJB) 2.1                              | 타이머 서비스 및 EJB 웹 서비스 종점                                                                           |
| Java 서블릿 2.4                                               | 웹 서비스 종점 필터                                                                                      |
| JavaServer Pages(JSP) 2.0 구조                               | 표현식 언어 및 태그 라이브러리                                                                                |
| J2EE Connector Architecture 1.5                            | 인바운드 자원 어댑터 및 Java Message Service(JMS) 플러그 가능성                                                  |
| 웹 서비스                                                      |                                                                                                  |
| Java Web Services Developer Pack 1.5                       | XML 응용 프로그램, 웹 서비스 및 웹 응용 프로그램의 구축, 테스트 및 배포를 위한 통합 툴킷                                           |
| Java API for XML-based Remote Procedure Calls(JAX-RPC) 1.1 | WSDL과 Java 기술에 대한 매핑과 웹 서비스 클라이언트 및 종점의 배포 지원                                                    |
| WS-I Basic Profile 1.0                                     | WSDL 및 SOAP를 사용한 상호 운용성 활성화 요소                                                                   |
| SOAP with attachment API for Java(SAAJ) 1.2                | SOAP 기반 메시징을 위한 API. 첨부 파일이 있는 SOAP 메시지 작성 조장                                                    |
| Java APIs for XML Registries (JAXR) 1.0                    | Universal Description Discovery and Integration(UDDI 및 ebXML)을 위한 것과 같은 XML 레지스트리 액세스를 위한 표준 API |
| 기타                                                         |                                                                                                  |
| J2EE Deployment 1.1                                        | J2EE 구성 요소 및 응용 프로그램의 배포를 가능하게 하는 표준 API                                                         |
| J2EE Management 1.0                                        | J2EE 플랫폼 관리를 위한 정보 모델의 정의                                                                        |
| Java Management Extensions(JMX) 1.2                        | 표준 관리 API                                                                                        |
| Java Authorization Contract for Containers(JACC) 1.0       | J2EE Application Server와 인증 정책 공급자 간의 보안 계약 정의                                                   |
| Java API for XML Processing(JAXP) 1.2                      | 응용 프로그램에서 XML 문서를 구문 분석하고 변환할 수 있으며 또한 XML 스키마 처리에 대한 지원을 추가하는 API                               |
| JMS 1.1                                                    | J2EE 응용 프로그램 구성 요소에서 메시지를 작성하여 보내고 받으며 읽을 수 있도록 하며 대기열 및 항목을 위한 일관된 API 지원을 추가하는 메시지 표준          |

표 2-5 J2EE 1.4 플랫폼에서 사용할 수 있는 API (계속)

JavaMail 1.3

메일 시스템을 모델화하는 일련의 추상 클래스, API에 대한 사소한 업데이트 포함

## 지원되는 다른 Java 버전으로 전환

Sun Java System Application Server 8.2를 사용하려면 J2SE 5.0 이상을 기본 JVM으로 실행해야 합니다. Java를 다른 버전으로 전환하려면 다음의 일반적인 단계를 수행합니다. (Windows 및 Unix)

### ▼ 지원되는 다른 Java 버전으로 전환하는 방법

- 1 **Java SDK(JRE가 아님)를 시스템에 설치하지 않았다면 다운로드하여 설치합니다.**  
Java SDK는 <http://java.sun.com/j2se>에서 다운로드할 수 있습니다.
- 2 **Application Server를 완전히 중지합니다.**  
다음 명령줄을 사용할 수 있습니다.  
`as-install/bin/asadmin stop-domain`  
관리 콘솔 GUI를 사용할 수도 있습니다.
  - a. **Application Server 노드를 누릅니다.**
  - b. **인스턴스 중지를 누릅니다.**
- 3 **install\_dir/config/asenv.conf 파일(Windows에서는 asenv.bat)을 편집하여 AS\_JAVA의 값을 새 J2SE 홈 디렉토리로 변경합니다.**
- 4 **as-install/samples/common.properties 파일을 편집하여 com.sun.aas.javaRoot...로 시작하는 줄을 새 J2SE 홈 디렉토리에 대한 참조로 변경합니다.**
- 5 **서버를 다시 시작합니다.**  
`as-install/bin/asadmin start-domain`

## 고성능

Application Server에는 고성능 EJB 컨테이너, 웹 컨테이너 및 서비스가 포함되어 있으며 Sun Java System Message Queue 소프트웨어와의 동시 메시지 전달을 지원합니다.

## 확장성

Application Server는 서버 인스턴스 클러스터링과 요청 로드 균형 조정을 통한 수평 확장성을 지원합니다. 또한 대형 다중 프로세서 시스템을 지원하는 수직적 확장성도 제공합니다. 메시지 브로커를 클러스터링하면 보다 나은 확장성과 가용성을 얻을 수

있습니다. HTTP 클라이언트, RMI/IIOP 기반 Rich Client Application, 웹 서비스 클라이언트 및 JRM 클라이언트로부터의 클라이언트 액세스 로드를 Application Server 클러스터에 대해 균형있게 조정할 수 있습니다.

## JavaServer Faces 1.1 지원

Sun Java System Application Server Enterprise Edition 8.2은 JavaServer Faces 1.1 기술을 지원합니다. JavaServer Faces 기술은 상태, 이벤트, 처리 및 입력 검증을 관리하는 사용자 인터페이스 구성 요소를 나타내는 일련의 서버측 API로 구성됩니다. API는 또한 페이지 탐색을 정의하고 국제화와 내게 필요한 옵션 기능을 지원합니다. 사용자 정의 UI 구성 요소를 JSP 사용자 정의 태그 라이브러리와 함께 추가할 수 있습니다.

JavaServer Faces 기술을 개발하는 동안 개발 팀의 각 구성원은 프로세스의 한 부분에 집중할 수 있습니다. 그 후 간단한 프로그래밍 모델로 각 부분들을 연결하면 개발 주기가 훨씬 효율적이고 간단해집니다.

## 알려진 문제점 및 제한 사항

---

이 장에서는 Sun Java System Application Server Enterprise Edition 8.2 소프트웨어의 알려진 문제점과 이를 해결하는 방법에 대해 설명합니다. 문제를 설명하는 부분에서 특정 플랫폼을 언급하지 않는 경우에는 해당 문제가 모든 플랫폼에 적용됩니다. 이 정보는 다음 내용으로 구성되어 있습니다.

- 37 페이지 “관리”
- 43 페이지 “Apache 및 로드 밸런서 플러그인”
- 44 페이지 “응용 프로그램 클라이언트”
- 44 페이지 “번들로 제공되는 Sun JDBC 드라이버”
- 47 페이지 “커넥터”
- 47 페이지 “설명서”
- 49 페이지 “고가용성”
- 56 페이지 “설치”
- 59 페이지 “J2EE Tutorial”
- 59 페이지 “라이프사이클 관리”
- 60 페이지 “로깅”
- 61 페이지 “Message Queue”
- 61 페이지 “모니터링”
- 64 페이지 “지속성”
- 64 페이지 “PointBase”
- 65 페이지 “샘플”
- 68 페이지 “보안”
- 68 페이지 “업그레이드 유틸리티”
- 72 페이지 “웹 컨테이너”

### 관리

이 절에서는 관리와 관련된 알려진 문제점과 해결 방법을 설명합니다.

**domain1이 없으면 package-appclient 스크립트가 작동하지 않음(아이디 6171458)**

### 설명

기본적으로 \$INSTALL/lib/package-appclient.xml에는 asenv.conf 파일이 가리키는 domain1의 AS\_ACC\_CONFIG 변수를 위한 하드 코드된 값이 있습니다. domain1이 삭제되고 새 도메인이 만들어지는 경우 AS\_ACC\_CONFIG 변수가 새 도메인 이름으로 업데이트되지 않아 package-appclient 스크립트의 실패 요인이 됩니다.

### 해결 방법

다음 중 한 가지를 수행합니다.

- domain1을 그대로 두고 그 주위에 다른 도메인을 만듭니다.
- domain1을 제거하고 \$INSTALL/lib/package-appclient.xml에서 domain1의 하드 코드된 값을 새 도메인 이름으로 변경합니다. domain1이 없는 경우에는 새 도메인이 생성될 때마다 이 작업을 수행해야 합니다.

**로드 균형 조정 플러그인을 설치하면 기존 플러그인을 덮어씀(아이디 6172977)**

### 설명

7.1EE처럼 로드 밸런서 플러그인이 이미 설치되어 있는 Application Server 설치에 대해 로드 밸런서 플러그인을 설치하면, 플러그인을 실행할 새 서버 인스턴스를 만들었더라도 8.2EE 플러그인이 기존 로드 밸런서를 자동으로 대체합니다.

플러그인 파일은 기본적으로 install\_dir/plugins/lbplugin 디렉토리에 설치됩니다. 즉, Application Server 설치당 한 버전의 플러그인만 사용할 수 있습니다. 콘솔 설치 프로그램에 설치 제거 중이라는 메시지가 표시되며, 가끔씩 이 메시지가 표시되지 않는 경우도 있습니다.

### 해결 방법

누구에게나 이 문제가 발생하지는 않습니다. 이 문제가 발생하면 업그레이드 설치를 수행하는 대신 이전 Application Server 설치를 제거한 후 새로 설치하십시오.

**JES2 AS7과 비교 시, JES3 Application Server 8.2의 asadmin 스크립트에서의 몇 가지 변경 사항(아이디 6189433, 6189436)**

Application Server 7.x와 비교하여 Application Server 8.2의 asadmin 명령에 몇 가지 변경된 사항이 있습니다. 예를 들어, 7.x에서 서버 인스턴스를 시작하는 명령은 다음과 같습니다.

```
asadmin start-instance
```

8.2에서 이에 해당하는 명령은 다음과 같습니다.

```
asadmin start-domain --user admin domain1
```

최신 asadmin 명령 구문에 대한 자세한 내용은 다음 문서를 참조하십시오.

- Sun Java System Application Server Enterprise Edition 8.2 관리 설명서
- Sun Java System Application Server Enterprise Edition 8.2 Reference Manual
- Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide

### Application Server에서 기본 포트가 변경됨(아이디 6198555)

#### 설명

JES2/Application Server 7.x에서 JES5/Application Server 8.2로 업그레이드하면 기본 포트가 변경되어 비호환성 또는 오류가 발생할 수 있습니다.

#### 해결 방법

Application Server 8.2에서 사용되는 기본 포트 목록은 이 릴리스 노트의 [19 페이지 “기타 요구 사항”](#)을 참조하십시오.

### 백업 도메인을 새 이름을 사용하여 복원할 수 없음(아이디 6196993)

#### 설명

`asadmin restore-domain` 명령은 도메인 이름 변경 옵션을 제공하지만 원래 이름과 다른 이름을 사용하여 도메인을 복원할 수 없기 때문에 `backup-domain` 명령과 `restore-domain` 명령을 사용하여 동일한 Application Server 설치에서 도메인 미러링을 수행할 수 없습니다. 백업한 도메인의 이름을 변경하는데 성공한 것처럼 보이지만 이름을 변경한 도메인을 시작하려고 시도하면 도메인 구성의 항목이 변경되지 않고 `startserv` 및 `stopserv`가 원래 도메인 이름을 사용하여 경로를 설정하기 때문에 실패하게 됩니다.

#### 해결 방법

`restore-domain` 명령에 사용된 도메인 이름이 원래 `backup-domain` 명령에 사용된 이름과 같아야 합니다. Application Server 8.2의 `backup-domain` 및 `restore-domain` 명령은 같은 시스템의 동일한 도메인을 백업하고 복구하는 경우에만 사용할 수 있습니다.

### Application Server를 추가 JMX 에이전트와 함께 시작하는 것은 지원되지 않음(아이디 6200011)

#### 설명

J2SE 1.4.x, 5.0 이상을 Application Server에서 구성할 수 있습니다. J2SE 5.0 플랫폼의 중요한 기능은 JMX 에이전트를 시작하는 능력입니다. 이 기능은 서버를 시작할 때 시스템 등록 정보를 명시적으로 설정하면 활성화됩니다.

예를 들면 다음과 같습니다.

```
name="com.sun.management.jmxremote" value="true"
name="com.sun.management.jmxremote.port" value="9999"
name="com.sun.management.jmxremote.authenticate" value="false"
name="com.sun.management.jmxremote.ssl" value="false"
```

JMX 등록 정보를 구성하고 서버를 시작한 후 새 jmx-connector 서버가 Application Server VM 내에서 시작됩니다. 이로 인한 원하지 않는 부작용으로 관리 기능이 역으로 영향을 받고 관리 Application Server GUI와 CLI에서 예기치 못한 결과가 발생할 수 있습니다. 문제는 내장된 jmx-connector 서버와 새 jmx-connector 서버 간에 충돌이 발생하는 것입니다.

### 해결 방법

jconsole(또는 다른 JMX 준수 클라이언트)을 사용할 경우 Application Server 시작과 함께 시작되는 표준 JMX Connector Server를 다시 사용할 것을 고려합니다.

서버를 시작하면 아래에서 설명한 것과 유사한 행이 server.log에 나타납니다. 거기에 지정된 JMXServiceURL에 연결하여 자격 증명을 성공적으로 제공한 후 동일한 관리/구성 작업을 수행할 수 있습니다. 예를 들면 다음과 같습니다.

```
[#|2004-11-24T17:49:08.203-0800|INFO|sun-appserver-ee8.1|javax.enterprise.
system.tools.admin|_ThreadID=10;|ADM1501: Here is the JMXServiceURL for the
JMXConnectorServer: [service:jmx:rmi:///jndi/rmi://hostname:8686/management/
rmi-jmx-connector]. This is where the remote administrative clients should
connect using the JSR 160 JMX Connectors.|#]
```

자세한 내용은 **Sun Java System Application Server 8.2 Administration Guide**를 참조하십시오.

### UNIX 상에서 Application Server start 및 stop 스크립트에 대한 실행 권한이 지나치게 제한됨(아이디 6206176)

#### 설명

사용자 "A"로 로그인하여 asadmin restore-domain 명령을 실행하면 스크립트의 사용자 권한이 744(rwxr--r--)로 종료됩니다. 이후에 사용자 "B"로 도메인을 시작하거나 중지하려고 시도하면 "B"가 루트더라도 "A"에 대해서만 스크립트를 실행할 수 있기 때문에 실패하게 됩니다.

### 해결 방법

스크립트의 사용자 권한을 다음과 같이 변경합니다.

```
chmod 755 appserv/domains/domain-name/bin/*
```

### 웹 서비스의 종점 URL을 사용하여 로드 밸런서 구성 파일을 만들 수 없음(아이디 6236544, 6275436)

#### 설명

웹 서비스 URL을 내보내는 EJB 모듈이 있는 응용 프로그램을 사용하여 로드 밸런서 구성을 설정하면 loadbalancer.xml 결과 파일에 웹 서비스의 컨텍스트 루트가 없게 됩니다.



## 해결 방법

1. loadbalancer.xml 파일을 편집하여 누락된 웹 모듈을 다음과 같이 추가합니다.

```
<web-module context-root="context-root-name"
 disable-timeout-in-minutes="30" enabled="true"/>
```

2. *context-root-name* 값을 EJB로 노출된 웹 서비스의 컨텍스트 루트 이름과 바꿉니다.

## 구성에 있는 Java 홈 설정이 적용되지 않음(아이디 6240672)

### 설명

Application Server 도메인/서버가 연결된 구성의 java-config 요소에 대한 java-home 속성에 지정된 JDK를 사용하지 않습니다.

### 해결 방법

지정된 서버 설치에서 모든 도메인의 Application Server 프로세스에 사용되는 JDK는 appserver-installation-dir /config/asenv.conf 파일에서 결정됩니다. 이 파일의 AS\_JAVA 등록 정보는 사용되는 JDK를 결정하며 설치 시에 설정됩니다. 설치가 완료된 후에 Application Server 프로세스에서 다른 JDK를 사용하려면 이 값을 다른 JDK로 수정할 수 있습니다. 이 설치에 있는 모든 도메인은 이 변경의 영향을 받습니다.

---

주 - asenv.conf 파일을 수동으로 변경할 경우 유효성이 검사되지 않으므로 주의해야 합니다. AS\_JAVA 값을 수정할 때의 최소 JDK 버전 요구 사항은 제품 설명서를 참조하십시오.

---

sun-appserv-admin을 사용하여 Application Server를 다시 시작하면 LoginException 오류가 발생함(아이디 6288893)

### 설명

이 문제는 %CONFIG\_HOME%의 잘못된 값으로 인해 발생합니다.

### 해결 방법

1. 기존 이름을 asant.bak으로 바꿉니다.
2. <as\_install> /lib/install/templates/ee(SE/EE 버전용)에서 asant.template 파일을 <as\_install>/bin/ 디렉토리에 복사하고 asant 파일 이름을 변경합니다.
3. 새로 복사된 <as\_install> /bin/asant 스크립트를 편집하여 %CONFIG\_HOME% 토큰을 <as\_install>/config로 바꿉니다.
4. 원본 asant.bak 파일을 수동으로 변경했다면 이러한 변경 사항을 모두 새 asant 스크립트로 병합합니다.

**Application Server 설명서에 .asadmintruststore 파일에 대한 설명 없음(아이디 6315957)****설명**

이 파일이 서버 관리자의 홈 디렉토리에 없으면 서버에 호스트된 일정 응용 프로그램을 업그레이드할 때 심각한 버그가 발생할 수 있습니다.

**해결 방법**

- 가능하다면 서버를 설치한 사용자가 `asadmin start-domain domain1` 명령을 실행해야 합니다.
- 해당 사용자가 실행하지 않는 경우 `.asadmintruststore`를 설치 사용자의 홈 디렉토리에서 실행 중인 사용자의 홈 디렉토리로 이동 또는 복사해야 합니다.
- 파일이 설치 사용자의 홈 디렉토리에서 실행 중인 사용자의 홈 디렉토리로 이동되면 6309079, 6310428, 6312869 버그에서 설명한 대로 응용 프로그램의 업그레이드에 문제가 생길 수 있습니다. 이는 업그레이드/설치 사용자(일반적으로 Java ES의 루트)의 홈 디렉토리에 더 이상 `.asadmintruststore` 파일이 없기 때문입니다.

**create-domain 마스터 비밀번호에 특수 문자가 있는 경우 도메인이 시작되지 않음(아이디 6345947)****설명**

도메인의 마스터 비밀번호에 백분율(%) 문자가 있는 경우 도메인이 시작되지 않습니다.

**해결 방법**

도메인의 마스터 비밀번호에는 백분율 문자(%)를 사용할 수 없습니다. 이 사항은 새 도메인을 만들거나 기존 도메인의 마스터 비밀번호를 변경할 때 적용됩니다.

**magnus.conf 및 obj.conf의 로드 밸런서 구성 변경 사항을 덮어씀(아이디 6394181)****설명**

보안 `http-listener`를 만들고 `lbplugin`을 설치한 후 `webserver_instance_dir/config`의 `magnus.conf` 및 `obj.conf` 파일이 수정되고 `lbplugin` 내용이 제거됩니다.

설치 프로그램이 로드 밸런서 플러그인 설치 과정에서 Application Server의 `magnus.conf` 및 `obj.conf` 구성 파일을 수정합니다. Application Server 관리 콘솔에 로그인하여 로드 밸런서가 설치된 인스턴스에 대한 인스턴스 구성을 관리하려고 하면 Application Server에서 구성이 수동으로 편집되었다는 경고 메시지가 표시됩니다. 이 경고는 실제로 설치 프로그램에서 수행한 변경을 나타냅니다.

**해결 방법**

설치 프로그램에서 수행된 변경 내용을 덮어쓰지 않았는지 확인합니다.

## Apache 및 로드 밸런서 플러그인

이 절에서는 Apache Web Server 및 로드 밸런서 플러그인과 관련된 알려진 문제점과 해결 방법을 설명합니다.

### High-Availability Administration Guide에 Apache에서의 openssl 사용에 대한 지침이 잘못되어 있음(아이디 6306784)

openssl을 컴파일하고 빌드할 때 다음 명령을 실행합니다.

```
cd openssl-0.9.7e
config
make
```

또한 Apache 1.3의 경우 mod\_ssl 소스의 디렉토리 이름은 사용되는 Apache 릴리스에 따라 달라집니다. 예를 들어 Apache 1.3.33에서 이름은 mod\_ssl-2.8.22-1.3.33입니다.

### High-Availability Administration Guide에 Apache 2.0 인증서 사용에 대한 지침 없음(아이디 6307976)

Apache 보안을 실행하기 위해서는 인증서를 사용해야 합니다. 인증 기관에서 인증서를 얻는 방법은 [modssl FAQ \(http://www.modssl.org/docs/2.8/ssl\\_faq.html#ToC24\)](http://www.modssl.org/docs/2.8/ssl_faq.html#ToC24)에서 인증서에 대한 정보를 참조하십시오.

### Apache Web Server를 루트로 시작해야 함(아이디 6308021)

Application Server가 루트에서 설치되었을 경우 Apache Web Server를 루트로 시작해야 합니다(Solaris만 해당). Java Enterprise System 설치하는 루트로 설치됩니다. Apache 2.0의 경우 Apache는 루트로 시작한 후 사용자가 지정하는 다른 사용자로 전환되어 실행됩니다. /conf/httpd.conf 파일에서 해당 사용자를 지정합니다. 루트로 시작하려면 대부분의 시스템에서 httpd.conf 파일을 편집하여 정확한 그룹을 지정해야 합니다. 다음 명령줄을

```
Group #-1
```

아래와 같이 바꿉니다.

```
Group nobody
```

사용자/그룹 사용에 대한 자세한 내용은 httpd.conf 파일에 포함되어 있습니다.

### Solaris에서 Apache Web Server 2.0의 openssl 사용에 대한 지침에 추가 사항이 있음(아이디 6308043)

Apache 2.0과 로드 균형 조정기 플러그인을 설치한 후에 ssl.conf와 ssl-std.conf를 다음과 같이 편집합니다.

다음 명령줄을

```
<VirtualHost _default_:9191>
```

아래와 같이 바꿉니다.

```
<VirtualHost machine_name:9191>
```

여기서, *machine\_name*은 사용하는 시스템의 이름이고 9191은 보안 포트 번호입니다.

## 응용 프로그램 클라이언트

이 절에서는 응용 프로그램 클라이언트와 관련된 알려진 문제점과 해결 방법을 설명합니다.

**Application Client Archive에 패키지로 포함된 라이브러리 JAR이 MANIFEST 파일을 덮어씀(아이디 6193556)**

### 설명

클라이언트 JAR 내에 최상위 JAR 파일이 있는 경우(이 경우에는 *reporter.jar*) 클라이언트 JAR을 배포할 때 해당 JAR의 MANIFEST 파일이 클라이언트 JAR의 MANIFEST 파일을 덮어씁니다.

### 해결 방법

현재는 해결 방법이 없습니다.

**CGI-bin 및 SHTML 기능과 같은 동적 콘텐츠 기술이 지원되지 않음(아이디 6373043)**

### 설명

CGI-bin 및 SHTML과 같은 동적 콘텐츠 기술은 더 이상 지원되지 않습니다.

### 해결 방법

JSP 및 웹 서비스 기술을 대신 사용하십시오.

## 번들로 제공되는 Sun JDBC 드라이버

이 절에서는 번들로 제공되는 Sun JDBC 드라이버와 관련된 알려진 문제점과 해결 방법을 설명합니다.

**번들로 제공되는 Microsoft SQL Server용 Sun 드라이버와 함께 TRANSACTION\_SERIALIZABLE 격리 수준을 사용하는 응용 프로그램은 두 개의 트랜잭션이 병행하여 실행 중이고 그 중 하나가 롤백하면 준비된 명령문을 사용하여 업데이트할 경우 중단됨(아이디 6165970)**

연결을 위해 원하는 격리 수준을 설정하려면 같은 격리 수준에 상응하는 연결 풀을 만들어야 합니다. 연결 풀 구성에 대한 자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.2 관리 설명서**를 참조하십시오.

## PreparedStatement 오류(아이디 6170432)

### 설명 1

응용 프로그램이 하나의 트랜잭션에서 3000개가 넘는 PreparedStatement 객체를 생성하면 DB2에 다음 오류가 발생할 수 있습니다.

[sunm][DB2 JDBC Driver] No more available statements. Please recreate your package with a larger dynamicSections value.

### 해결 방법 1

연결 풀 정의에 다음 등록 정보를 추가하여 드라이버에서 더 큰 동적 섹션 값으로 DB2 패키지를 다시 바인드하도록 합니다.

```
createDefaultPackage=true replacePackage=true dynamicSections=1000
```

연결 풀 구성에 대한 자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.2 관리 설명서**를 참조하십시오.

### 설명 2

위의 PreparedStatement 오류와 관련하여 발생할 수 있는 다른 오류 메시지는 다음과 같습니다.

[sunm][DB2 JDBC Driver][DB2]Virtual storage or database resource is not available.

### 해결 방법 2

DB2 서버 구성 매개 변수 APPLHEAPSZ를 증가시킵니다. 권장 값은 4096입니다.

### 설명 3

TRANSACTION\_SERIALIZABLE 격리 수준 응용 프로그램에서 TRANSACTION\_SERIALIZABLE 격리 수준을 사용하고 위에 제시한 매개 변수 중 하나를 사용하면 연결하는 동안 응용 프로그램이 중단될 수 있습니다.

### 해결 방법 3

연결을 위해 바람직한 격리 수준을 설정하려면 상응하는 연결 풀을 같은 격리 수준에 만들어야 합니다. 지침 사항은 **Sun Java System Application Server Enterprise Edition 8.2 관리 설명서**를 참조하십시오.

## 번들로 제공된 Sybase Adaptive Server용 Sun 드라이버의 격리 수준을 설정할 때 문제가 발생함(아이디 6189199)

### 설명

- 번들로 제공되는 Sybase Adaptive Server용 Sun 드라이버와 함께 TRANSACTION\_SERIALIZABLE 격리 수준을 사용하는 응용 프로그램은 두 개의 트랜잭션이 병행하여 실행 중이고 그 중 하나가 롤백하면 준비된 명령문을 사용하여 업데이트할 경우 중단됩니다. 연결 롤백은 다음 메시지와 함께 실패하고 롤백한 연결은 더 이상 사용할 수 없습니다.

```
java.sql.SQLException:[sunm][Sybase JDBC Driver]Request cannot be submitted
due to wire contention
```

- Sybase Adaptive Server는 TRANSACTION\_REPEATABLE\_READ 격리 수준을 지원하지 않습니다. 그러나, DatabaseMetaData에 대해 쿼리를 실행하면 번들로 제공된 Sun 드라이버는 데이터베이스에서 이 격리 수준이 지원된다는 결과를 반환합니다. 이 격리 수준을 사용하는 응용 프로그램은 실패하게 됩니다.
- 번들로 제공되는 Sun 드라이버를 사용하는 응용 프로그램은 TRANSACTION\_READ\_UNCOMMITTED 격리 수준을 설정할 수 없습니다. 응용 프로그램은 첫 번째 DataBaseMetaData 액세스에서 다음과 같은 예외를 발생시킵니다.

```
java.sql.SQLException:[sunm][Sybase JDBC Driver][Sybase]The optimizer could
not find a unique index which it could use to perform an isolation level 0
scan on table 'sysystemprocs.dbo.spt_server_info'.
```

### 해결 방법

현재는 해결 방법이 없습니다.

## Solaris 10과 Enterprise Linux 3.0에서는 Sun에서 번들로 제공하는 Oracle JDBC 드라이버로 연결을 만들 수 없음(아이디 6247468)

SUN JDBC Oracle 데이터 소스(com.sun)를 사용할 때 JDBC 연결 풀에서 다음 등록 정보를 설정합니다. sql.jdbcx.oracle.OracleDataSource):

```
<property name="serverType" value="dedicated"/>
```

등록 정보 값은 Oracle 서버의 수신기가 구성된 방식에 따라 달라집니다. "공유" 모드로 구성되어 있으면 위 값을 "dedicated"로 변경해야 합니다.

```
java.lang.SecurityException: Sealing violation exception(아이디 6554602)
```

### 설명

CLASSPATH에 두 개 이상의 JDBC jar 파일이 있는 JDBC 10.2 드라이버에서 시작하는 경우 java.lang.SecurityException: Sealing violation exception이 발생할 수 있습니다.

Oracle의 자세한 설명은 다음 Oracle 문서 ID에 기록되어 있습니다.

주: 405446.1 주제: JDBC 드라이버 10.2는 Sealed JAR 파일을 사용하며 May Cause SecurityException Sealing Violatio

### 해결 방법

(Oracle 제안) CLASSPATH 는 하나의 JDBC 드라이버 JAR 파일만 포함하는지 확인하십시오.

## 커넥터

이 절에서는 J2EE 커넥터 구조와 관련된 알려진 문제점과 해결 방법을 설명합니다.

### DAS 인스턴스를 다시 시작한 후 cascade를 false로 설정하면 커넥터 모듈 배포 해제가 실패함(아이디 6188343)

#### 설명

이 시나리오에서 독립형 또는 내장형 커넥터 모듈은 DAS와 커넥터 연결 풀에 배포되며 배포된 모듈을 위한 자원이 만들어집니다. DAS 인스턴스를 다시 시작한 후 cascade를 false로 설정하면 다음 예외가 발생하고 커넥터 모듈 배포 해제가 실패하게 됩니다:

```
[#|2004-10-31T19:52:23.049-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system
.core|_ThreadID=14;|CORE5023:Error while unloading application [foo]|#]
```

### 해결 방법

DAS 인스턴스를 다시 시작한 후 독립형 및 내장형 커넥터의 배포를 해제하려면 종속 배포 해제(cascade 옵션을 true로 설정)를 사용합니다.

### JMS create-jms-resource;CLI가 기본값을 올바르게 설정하지 않음(아이디 6294018)

#### 설명

명령줄에서 asadmin create-jms-resource 명령을 사용하여 새 JMS 자원을 만들 때 최소 풀 크기와 최대 풀 크기를 지정할 수 없기 때문에 asadmin 명령은 기본 풀 크기 값(최소값 8, 최대값 32)을 사용하여 자원을 만들어야 하지만 여기서는 해당되지 않습니다. 대신 명령줄에서 자원을 만들 경우 기본 최소 풀 크기와 최대 풀 크기(1 및 250)가 각각 적용됩니다.

### 해결 방법

명령줄에서 JMS 자원을 만든 후 관리 콘솔을 사용하여 최소 및 최대 풀 크기 값을 수정합니다.

## 설명서

이 절에서는 설명서와 관련된 알려진 문제점과 해결 방법을 설명합니다.

### Javadoc 비일관성 문제가 발생함(여러 아이디)

몇 가지 AMX 인터페이스와 메소드를 위한 Javadoc가 누락되었거나 잘못되어 있습니다.

- NumConnAcquired 및 NumConnReleased 통계를 위한 getter 메소드가 ConnectorConnectionPoolStats 및 AltJDBCConnectionPoolStats에 없습니다. 이러한 getter 메소드는 이후 릴리스에서 getNumConnAcquired() 및 getNumConnReleased()로 추가됩니다.
- EJBCacheStats에서 다음 메소드를 호출하면 예외가 발생합니다. 메소드를 호출하면 예외가 발생합니다. 이 문제는 이후 릴리스에서 수정될 예정입니다.
- 서버를 시작한 후 몇 초가 지나야 AMX Mbeans을 등록하고 사용할 수 있습니다. 이후 릴리스에서는 AMX MBeans이 언제 가득 차는지 확인할 수 있습니다.
- XTypes.CONNNECTOR\_CONNECTION\_POOL\_MONITOR 상수가 잘못 표기되어 있습니다("NNN"). 이 문제는 이후 릴리스에서 수정될 예정입니다.

**번들로 제공된 ANT에서 java.lang.NoClassDefFoundError가 발생함(아이디 6265624)**

## 설명

스레드에서 main: java.lang.NoClassDefFoundError:  
org/apache/tools/ant/launch/Launcher 예외가 발생합니다.

## 해결 방법

Application Server 외부의 항목에 대해 번들로 제공되는 ANT를 사용하는 것은 좋지 않습니다.

**로깅 옵션 설명이 잘못되었음(아이디 6463965)**

**Sun Java System Application Server Enterprise Edition 8.2 Performance Tuning Guide**에 로그 옵션에 대해 다음과 같이 잘못 설명되어 있습니다.

관리 GUI는 다음과 같은 두 가지 로깅 옵션을 제공합니다.

- 옵션 1 - 이벤트 로그에 stdout(System.out.print) 내용 기록
- 옵션 2 - 이벤트 로그에 stderr(System.err.print) 내용 기록

Application Server Enterprise Edition 8.2에는 이러한 로그 옵션이 더 이상 존재하지 않습니다.

**Application Server 8.2의 HTTP 파일 캐시 기능과 관련된 정보가 충돌함(아이디 6474799)**

Application Server Enterprise Edition 8.2 설명서에는 **Sun Java System Application Server Enterprise Edition 8.2 Performance Tuning Guide**의 “HTTP File Cache”에 HTTP 파일 캐시 기능에 대한 설명이 있습니다. 그러나 이 기능은 Application Server Enterprise Edition 8.2에 포함되어 있지 않습니다. 이 기능은 Application Server 9.0에서 다시 도입되었습니다.



래핑된 연결에서 물리적 연결을 가져오는 방법에 대한 설명서가 더 이상 올바르지 않음(아이디 6486123)

다른 결함(6295215일 수 있음)으로 인해 **Sun Java System Application Server Enterprise Edition 8.2 Developer's Guide**의 11 장, “Using the JDBC API for Database Access”의 11 장, “Using the JDBC API for Database Access”에서 **Sun Java System Application Server Enterprise Edition 8.2 Developer's Guide**의 “Obtaining a Physical Connection from a Wrapped Connection” 절에 제공된 코드가 올바르지 않습니다. 특히,

```
Connection drivercon = ds.getConnection(con);
```

줄은 다음과 같이 변경되었습니다.

```
Connection drivercon = ((com.sun.gjc.spi.DataSource)ds).getConnection(con);
```

## 고가용성

이 절에서는 고가용성 데이터베이스(HADB)와 관련된 알려진 문제점과 해결 방법을 설명합니다.

### 이중 네트워크를 사용한 HADB 구성 문제가 발생함(아이디 없음)

Solaris SPARC에서는 두 개의 서브넷에 이중 네트워크와 함께 구성된 HADB가 제대로 작동합니다. 그러나, 운영 체제 또는 일부 하드웨어 플랫폼의 네트워크 드라이버 문제 때문에 Solaris x86과 Linux 플랫폼에서는 이중 네트워크를 제대로 처리할 수 없는 경우가 있는 것으로 확인되었습니다. 이로 인해 HADB에 발생하는 문제는 다음과 같습니다.

- Linux에서는 메시지를 보낼 때 일부 HADB 프로세스가 차단됩니다. 이 때문에 HADB 노드가 다시 시작되고 네트워크가 분할됩니다.
- Solaris x86에서는 네트워크 실패 후 발생할 수 있는 몇 가지 문제로 인해 다른 네트워크 인터페이스로 전환할 수 없습니다. 이 문제는 항상 발생하는 것은 아니므로 네트워크를 하나만 사용하는 것보다는 두 개를 사용하는 것이 더 낫습니다. Solaris 10에서는 이러한 문제가 부분적으로 해결되었습니다.
- Trunking은 지원되지 않습니다.
- HADB는 Windows 2003에서 이중 네트워크를 지원하지 않습니다(ID 5103186).

### HADB 데이터베이스 생성이 실패함(아이디 없음)

사용할 수 있는 공유 메모리 세그먼트가 너무 적다는 뜻의 다음 오류로 새 데이터베이스 생성에 실패할 수 있습니다.

#### 설명

```
HADB-E-21054: System resource is unavailable: HADB-S-05512: Attaching shared memory segment with key "xxxxx" failed, OS status=24 OS error message: Too many open files.
```

## 해결 방법

공유 메모리가 구성되었는지와 구성이 작동하는지를 확인합니다. 특히, Solaris 8에서는 파일 `/etc/system`을 검사하고 변수 `shmsys:`의 값이 다음과 같은지 확인합니다. `shminfo_shmseg`는 적어도 호스트당 노드 수의 6배입니다.

**공유 메모리 세그먼트가 잠겨 있어서 페이지 아웃될 수 없음(아이디 5052548)**

## 설명

HADB 4.3-0.16 이상은 해당 공유 메모리 세그먼트(`SHM_SHARE_MMU` 플래그 사용)를 만들어 연결할 때 Intimate Shared Memory를 사용하도록 구성되었습니다. 이 플래그를 사용하면 공유 메모리 세그먼트가 물리적 메모리로 잠기기 때문에 페이지 아웃되지 않습니다. 이 경우 저사양의 시스템에서 설치할 때 문제가 발생하기 쉽습니다.

따라서 개발자가 메모리 용량이 512MB이고 Application Server 7.0 EE 사용 시 사용 가능한 스왑 공간이 충분한 시스템에 7.1 EE 이상을 설치한 경우 `clsetup` 클러스터를 구성하는데 문제가 발생합니다. 이 클러스트는 HADB 노드 2개를 만들며 각각의 `devicesize`가 512이므로 2개의 노드에서 필요한 공유 메모리를 지원하기에 물리적 RAM이 충분하지 않습니다.

## 해결 방법

Application Server와 HADB를 함께 배치할 경우 권장량의 메모리를 설치해야 합니다. 자세한 내용은 18 페이지 “HADB 요구 사항 및 지원되는 플랫폼”을 참조하십시오.

**hadbm set가 자원 가용성(디스크 및 메모리 공간)을 확인하지 않음(아이디 5091280)**

## 설명

`hadbm set`를 사용하여 장치 또는 버퍼 크기를 증가시키면 관리 시스템에서 데이터베이스를 만들거나 노드를 추가할 때 자원 가용성을 확인하지만 장치나 주 메모리 버퍼 크기가 변경될 때는 충분한 자원이 있는지 확인하지 않습니다.

## 해결 방법

`devicesize` 또는 `buffersize` 구성 속성을 증가시키기 전에 모든 호스트에서 사용 가능 디스크와 메모리 공간이 충분한지 확인합니다.

**packagepath에 대한 이기종 경로가 지원되지 않음(아이디 5091349)**

## 설명

같은 이름의 동일한 소프트웨어 패키지를 서로 다른 호스트의 여러 위치에 등록할 수 없습니다. 예를 들면 다음과 같습니다.

```
hadbm registerpackage test --packagepath=/var/install1 --hosts europa11
Package successfully registered.
```

```
hadbm registerpackage test --packagepath=/var/install2 --hosts europa12
hadbm:Error 22171: A software package has already been registered with
the package name test.
```

### 해결 방법

HADB는 데이터베이스 클러스터의 여러 노드에 걸친 이기종 경로를 지원하지 않습니다. HADB 서버 설치 디렉토리(--packagepath)가 모든 참여 호스트에서 동일한지 확인합니다.

**createdomain이 실패할 수 있음(아이디 6173886, 6253132)**

### 설명

네트워크 인터페이스가 여러 개인 호스트에 관리 에이전트를 실행하면 create domain 명령은 모든 네트워크 인터페이스가 동일한 서브넷에 있지 않을 경우 실패하게 됩니다.

```
hadbm:Error 22020: The management agents could not establish a
domain, please check that the hosts can communicate with UDP multicast.
```

관리 에이전트(다르게 구성되지 않은 경우)는 UDP 멀티캐스트에 대해 "첫 번째" 인터페이스를 사용합니다("첫 번째"는 java.net.NetworkInterface.getNetworkInterfaces()의 결과로 정의됨).

### 해결 방법

최고의 해결 방법은 관리 에이전트에 사용할 서브넷을 요청하는 것입니다(구성 파일에서 ma.server.mainternal.interfaces 설정. 예: ma.server.mainternal.interfaces=10.11.100.0). 서브넷 사이의 라우터를 구성하여 멀티캐스트 패킷을 라우팅할 수도 있습니다(관리 에이전트는 멀티캐스트 주소 228.8.8.8을 사용).

관리 에이전트의 새 구성으로 재시도하기 전에 관리 에이전트 리포지토리를 정리해야 합니다. 도메인의 모든 에이전트를 중단하고 리포지토리 디렉토리에서 모든 파일과 디렉토리를 삭제합니다(관리 에이전트 구성 파일에 repository.dr.path로 표시). 새 구성 파일을 가진 에이전트를 다시 시작하기 전에 모든 호스트에서 이를 수행해야 합니다.

**HADB 인스턴스를 삭제한 후 디렉토리를 정리해야 함(아이디 6190878)**

### 설명

HADB 인스턴스를 삭제한 후 configure-ha-cluster 명령으로 새 인스턴스를 만들려고 하면 실패합니다. 문제는 이전 디렉토리가 ha\_install\_dir/rep/\* 및 ha\_install\_dir/config/hadb/instance\_name에 있는 원래 HADB 인스턴스에 남겨진다는 것입니다.

### 해결 방법

HADB 인스턴스를 삭제한 후 이러한 디렉토리를 수동으로 삭제하십시오.

## HADB 시작, 중지 및 재구성이 실패하거나 중지됨(아이디 6230792,6230415)

### 설명

Solaris 10 Operating System에서 `hadbm` 명령을 사용하여 HADB를 시작하거나 중단 또는 재구성하는 것이 다음 오류 중 하나로 실패하거나 중단될 수 있습니다.

```
hadbm:Error 22009: The command issued had no progress in the last
300 seconds.
```

```
HADB-E-21070: The operation did not complete within the time limit,
but has not been cancelled and may complete at a later time.
```

`clu_noman_srv` 프로세스가 사용하는 파일(`nomandev`)에 일관성 없는 읽기/쓰기가 있을 경우 이렇게 됩니다. HADB 내역 파일에서 다음 메시지를 찾으시면 이 문제가 있다는 뜻입니다.

```
n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Child process noman3 733
does not respond.
n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Have not heard from it in
104.537454 sec.
n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Child process noman3 733
did not start.
```

### 해결 방법

문제가 재현되지는 않았으므로 다음 해결 방법은 검증된 내용이 아닙니다. 그러나 영향을 받은 노드에 대해 이 명령을 실행하면 문제가 해결됩니다.

```
hadbm restartnode --level=clear nodeno dbname
```

노드에 대한 모든 장치가 다시 초기화됩니다. 다시 초기화하기 전에 노드를 중단시켜야 할 수도 있습니다.

## 관리 에이전트가 "IPv6\_MULTICAST\_IF failed" 예외로 종료됨(아이디 6232140)

### 설명

NIC 카드가 여러 개 설치된 Solaris 8을 실행 중인 호스트에서 시작할 경우 IPv6과 IPv4를 사용할 수 있는 상태의 카드들이 혼합되어 있다면 관리 에이전트는 "IPv6\_MULTICAST\_IF failed" 예외로 종료될 수 있습니다.

### 해결 방법

JAVA\_OPTIONS 환경 변수를 `-Djava.net.preferIPv4Stack=true`로 설정합니다. 예를 들면 다음과 같습니다.

```
export JAVA_OPTIONS="-Djava.net.preferIPv4Stack=true"
```

또는 이러한 문제가 없는 Solaris 9 이상을 사용하십시오.

**clu\_trans\_srv를 중단할 수 없음(아이디 6249685)****설명**

비동기 I/O를 수행할 때 Red Hat Enterprise Linux 3.0의 64비트 버전에서 clu\_trans\_srv 프로세스를 무중단 모드로 만드는 버그가 있습니다. 즉, kill -9가 동작하지 않아 운영 체제를 재부팅해야 합니다.

**해결 방법**

Red Hat Enterprise Linux 3.0의 32비트 버전을 사용합니다.

**hadbm은 대문자로 된 암호를 지원하지 않음(아이디 6262824)****설명**

암호가 hadb에 저장될 때 대문자로 된 암호는 소문자로 변환됩니다.

**해결 방법**

암호에 대문자를 사용하지 마십시오.

**HADB 버전 4.4.2.5에서 HADB 버전 4.4.1.7로 다운그레이드하면 다른 오류 코드가 표시되면서 ma는 실패함(아이디 6265419)****설명**

이전 HADB 버전으로 다운그레이드하면 다른 오류 코드가 표시되면서 관리 에이전트는 실패합니다.

**해결 방법**

HADB 데이터베이스를 다운그레이드할 수 있지만, 리포지토리 객체가 변경되었을 경우 관리 에이전트는 다운그레이드할 수 없습니다. 다운그레이드 후에는 최신 HADB 버전에서 관리 에이전트를 사용해야 합니다.

**설치/제거 및 symlink 유지(아이디 6271063)****설명**

HADB c 패키지(Solaris: SUNWhadb, Linux: sun-hadb-c) 버전 <m.n.u-p>의 설치/제거와 관련하여 symlink /opt/SUNWhadb/<m>는 일단 존재하는 경우 수정되지 않습니다. 따라서 연결이 끊어진 symlink가 있을 수 있습니다.

**해결 방법**

설치 전이나 제거 후에 사용 중이지 않은 경우 symlink를 삭제합니다.

**전역 및 로컬 영역에서 여러 관리 에이전트가 충돌함(아이디 6273681)****설명**

Solaris 10에서는, 전역 영역에서 `ma-initd` 스크립트를 사용하여 관리 에이전트를 중단하면 로컬 영역에서도 관리 에이전트가 중단됩니다.

**해결 방법**

전역 및 로컬 영역 모두에 관리 에이전트를 설치하지 마십시오.

`hadbm/ma`는 세션 객체가 시간 초과되어 MA에서 삭제될 때 더 나은 오류 메시지를 표시해야 함(아이디 6275103)

**설명**

가끔 서버의 자원 충돌 문제로 인해 관리 클라이언트의 연결이 끊어질 수 있으며, 다시 연결하면 다음 오류 메시지가 반환될 수 있습니다. "hadbm:Error 22184:A password is required to connect to the management agent" 오류 메시지가 반환될 수 있습니다.

**해결 방법**

서버에 자원 문제가 있다면 자원을 추가하는 등 적절한 조치를 취한 다음 다시 시도해보십시오.

**루트가 아닌 사용자는 HADB를 관리할 수 없음(아이디 6275319)****설명**

Java Enterprise System에서 (루트로) 설치하면 루트가 아닌 사용자는 HADB를 관리할 수 없습니다.

**해결 방법**

항상 루트로 로그인하여 HADB를 관리합니다.

**관리 에이전트는 특수 용도의 인터페이스를 사용해서는 안 됨(아이디 6293912)****설명**

0.0.0.0과 같은 IP 주소를 가진 특수 용도의 인터페이스는 관리 에이전트에서 HADB 노드에 사용할 유효한 인터페이스로 등록되지 않아야 합니다. 이러한 인터페이스를 등록하면 IP 주소 대신 호스트 이름을 사용하여 `hadbm create` 명령을 호출하는 사용자에게 HADB 노드가 이런 인터페이스에 설정되었을 경우 문제가 생길 수 있습니다. 그러면 노드가 통신할 수 없게 되어 `create` 명령이 중단될 수 있습니다.

**해결 방법**

인터페이스가 여러 개 있는 호스트에서 `hadbm create`를 사용할 때 항상 DDN 표기 형식을 사용하여 IP 주소를 명확하게 지정해야 합니다.

## Windows에서 리어셈블리 오류(아이디 6291562)

### 설명

특정 구성과 로드 của Windows 플랫폼에서 운영 체제에 많은 리어셈블리 오류가 있을 수 있습니다. 여러 테이블에 대한 스캔을 동시에 실행할 때(select \*) 20개가 넘는 노드로 된 구성에 문제가 있었습니다. 트랜잭션이 자주 중단되거나 복구를 완료하는 데 시간이 오래 걸리는 등의 징후가 있거나 시스템의 여러 부분에서 자주 시간 초과가 일어날 수 있습니다.

### 해결 방법

이 문제를 해결하기 위해 Windows 레지스트리 변수

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters 를 기본값인 100보다 높게 설정할 수 있습니다. 이 값을 0x1000 (4096)까지 증가시키는 게 좋습니다. 자세한 내용은 Microsoft 지원 페이지에서 811003

(<http://support.microsoft.com/default.aspx?scid=kb;en-us;811003>) 자료를 참조하십시오.

hadbm start <db\_name>을 실행하면 입력한 비밀번호의 일부가 마스크 처리되지 않은 상태로 표시됨(아이디 6303581, 6346059, 6307497)

### 설명

시스템이 로드 중인 경우 시스템 마스크 처리가 실패할 수 있으며 입력되는 비밀번호의 일부분자가 노출될 수 있습니다. 이는 사소한 보안 위험에 해당되며 비밀번호는 항상 마스크 처리되어야 합니다.

### 해결 방법

비밀번호를 고유 비밀번호 파일에 입력(Application Server 8.1 이후 일반적으로 권장되는 방법)하고 --adminpassword 또는 --dbpasswordfile 옵션과 함께 참조하십시오.

전역 영역에 설치된 JES5 HADB를 스파스 로컬 영역에서 액세스할 수 없음(아이디 6460979)

### 설명

Application Server를 Solaris 전역 영역의 /usr/SUNWappserver에 설치하면 Application Server 인스턴스와 함께 설치된 HADB 구성 요소를 스파스 로컬 영역에서 사용할 수 없습니다.

문제는 HADB를 전역 영역의 /opt/SUNWhadb에 설치했지만 이 디렉토리를 스파스 로컬 영역에서 읽을 수 없다는 것입니다. JES5의 HADB 번들은 재배포할 수 없습니다.

### 해결 방법

Application Server HADB 구성 요소를 재배포할 수 없기 때문에 HADB에 액세스하려는 각 스파스 로컬 영역에서 HADB 구성 요소를 개별적으로 설치해야 합니다.

## 설치

이 절에서는 설치와 관련된 알려진 문제점과 해결 방법을 설명합니다.

### 일부 Linux 시스템에서 "마침" 버튼을 누른 후 설치가 중단된 채 종료됨(아이디 5009728)

#### 설명

이 문제는 몇몇 Linux 시스템에서 발견되었습니다. Java Desktop System 2에서는 가장 일반적으로 나타나는 문제이지만 Linux RedHat 배포에서도 발견되었습니다.

설치 프로그램의 마지막 화면에서 **마침** 버튼을 누른 후 설치 프로그램에서 제품 정보 페이지나 제품 등록 페이지가 있는 브라우저 창을 시작하는 데 실패하고 명령 프롬프트를 반환하지 않은 채 무기한 중단됩니다.

#### 해결 방법

설치 프로그램을 시작했던 단말기 창에서 Ctrl+C를 눌러 설치 프로그램을 종료합니다. 이렇게 하면 제품 정보 페이지나 등록 페이지가 있는 브라우저 창이 시작됩니다. 그러나 브라우저 창이 나타나지 않는 경우에는 브라우저를 시작하고 다음 URL을 입력하면 정보 페이지를 볼 수 있습니다.

```
file://install_dir/docs-ee/about.html
```

제품을 등록하는 설치 옵션을 선택한 경우에는 제품 정보 페이지에서 등록 페이지로 연결되는 링크를 따라갑니다.

### 설치 중 imq 디렉토리를 만들어야 합니다(Windows만 해당)(아이디 6199697)

#### 설명

Windows에서 Application Server Enterprise Edition을 설치한 직후 Message Queue 브로커가 시작되지 않고 drive:\as\domains\domain1\imq 디렉토리가 없다는 메시지가 표시됩니다.

domain1을 시작한 후 브로커가 시작된 경우에는 디렉토리가 Application Server에 의해 만들어지기 때문에 문제가 발생하지 않습니다.

#### 해결 방법

1. 브로커를 만들기 전에 var\_home\_dir\_location을 만듭니다.

```
$imqbrokerd -varhome var_home_dir_location
```

예를 들면 다음과 같습니다.

```
$imqbrokerd -varhome D:\as\domains\domain1\imq
```



## Application Server를 RHLAS 3.0 및 RHLAS 4.0에서 compat-libstdc++ 없이 설정할 수 없음(아이디 6396102)

### 설명

compat-libstdc++ 라이브러리가 이미 시스템에 설치되지 않은 경우 Application Server Enterprise Edition 8.2를 RHLAS(Red Hat Linux Advanced Server) 3.0 또는 4.0 시스템에 설치할 수 없습니다. Application Server는 RHLAS 시스템에서 compat-libstdc++ 라이브러리가 필요하지만 이 라이브러리는 기본적으로 설치되지 않습니다. 이 문제는 RHLAS 시스템에서만 발생합니다.

### 해결 방법

Application Server 소프트웨어를 설치하기 전에 <http://rpm.pbone.net/index.php3/stat/4/idpl/843376/com/compat-libstdc++-7.3-2.96.118.i386.rpm.html>에서 compat-libstdc++ RPM을 다운로드하여 설치합니다.

## 서버가 64비트 모드로 실행 중인 경우 lbplugin(libpassthrough.so)을 사용할 수 없음(아이디 6480952)

### 설명

64비트 모드로 Web Server 7.0과 Application Server Enterprise Edition 8.2를 실행할 경우 64비트 버전의 로드 밸런서 플러그인이 실행되지 않고 다음과 같은 오류가 발생합니다.

```
failure: CORE2253: Error running Init function load-modules: dlopen
of /export/home/mareks/opt/webserver7/plugins/lbplugin/bin/libpassthrough.so
failed (ld.so.1: websrvd: fatal: /export/home/mareks/opt/webserver7/plugins/
lbplugin/bin/libpassthrough.so: wrong ELF class: ELFCLASS32)
failure: server initialization failed
```

문제는 Application Server Enterprise Edition 8.2에 64비트 로드 밸런서 플러그인이 없고, 64비트 Web Server에서는 64비트 플러그인이 필요하다는 것입니다.

다음 명령을 사용하여 Web Server를 64비트 모드로 실행할지 32비트 모드로 실행할지를 결정할 수 있습니다.

```
wadm get-config-prop --user=admin --config=xxx --password-file=xxx platform
```

### 해결 방법

Application Server Enterprise Edition 8.2용 64비트 로드 밸런서는 계획되어 있지 않습니다. 이 문제를 해결하려면 Web Server 7.0 역 프록시 기능을 사용하거나 Web Server 7.0을 32비트 모드로 실행하도록 구성합니다. 자세한 내용은 Web Server 설명서를 참조하십시오.

## asant deploy를 실행할 수 없음: "The input line is too long" 오류 발생(Windows 2000)(아이디 6485174)

### 설명

Windows 2000의 기본 위치에 Application Server 8.2를 설치할 경우 asant deploy를 실행할 때 다음과 같은 오류가 발생할 수 있습니다.

```
$ C:/Sun/JavaES5/appserver/bin/asant deploy
The input line is too long.
The syntax of the command is incorrect.
```

문제는 Windows 2000의 명령줄에는 1000자를 초과하여 입력할 수 없지만 시스템 구성에 따라 기본 ANT\_OPTS 환경에서 asant deploy 명령줄이 길게 확장될 수 있다는 것입니다. 이 문제는 Windows 2000에서만 발생하는 문제입니다.

### 해결 방법

Windows 2000에서는 Application Server를 매우 짧은 디렉토리 경로(예: C:\JES5\_AS)에 설치합니다.

## 잘못된 common.properties 서버 인스턴스 AppServer1로 JES5 b12, AS 설치(아이디 6485254)

### 설명

Windows에서 Application Server가 선택 구성 요소 설치 패널의 최상위 수준으로 선택된 경우 JES 5 b12를 사용하면 기본적으로 노드 에이전트 하위 구성 요소가 선택됩니다. 설치 과정에서 노드 에이전트 및 이 노드 에이전트에 속하는 AppServer1이라는 서버 인스턴스를 차례로 생성합니다. 이것은 올바른 동작입니다.

그러나 노드 에이전트 하위 구성 요소가 선택 해제된 경우라도 설치 과정에서 도메인에 대한 common.properties 파일에 AppServer1 인스턴스를 생성합니다. 예를 들면 다음과 같습니다.

```
domain.name=domain1
appserver.instance=AppServer1
```

이후에 asant를 사용하여 응용 프로그램을 배포할 수 없습니다.

### 해결 방법

appserver.instance=AppServer1을 appserver.instance=server로 교체하여 common.properties 파일을 편집합니다.

래핑된 연결에서 물리적 연결을 가져오는 방법에 대한 설명서가 더 이상 올바르지 않음(아이디 6486123)

다른 결함(6295215일 수 있음)으로 인해 **Sun Java System Application Server Enterprise Edition 8.2 Developer's Guide**의 11 장, “Using the JDBC API for Database Access”의 11 장, “Using the JDBC API for Database Access”에서 **Sun Java System Application Server Enterprise Edition 8.2 Developer's Guide**의 “Obtaining a Physical Connection from a Wrapped Connection” 절에 제공된 코드가 올바르지 않습니다. 특히,

```
Connection drivercon = ds.getConnection(con);
```

줄은 다음과 같이 변경되었습니다.

```
Connection drivercon = ((com.sun.gjc.spi.DataSource)ds).getConnection(con);
```

**Application Server는 NFS를 지원하지 않습니다(6396045).**

이 소프트웨어 버전에서 Application Server는 네트워크 파일 시스템(NFS)을 지원하지 않습니다.

#### 해결 방법

없음

## J2EE Tutorial

Sun Java System Application Server Enterprise Edition 8.2에서 J2EE 1.4 Tutorial을 실행하려면 다음 작업을 수행합니다.

- 파일 예/common/build.properties를 “About this Tutorial” 장의 “About the Examples” 절에 설명된 대로 편집할 때 포트를 4848에서 4849로 변경합니다.
- Deploytool 사용 시 예를 배포하기 전에 server localhost:4849를 추가합니다.
- 관리 콘솔을 사용하여 자원을 만들 때 대상 탭을 사용하여 서버를 대상으로 지정합니다. 명령줄 또는 asant 대상을 사용하는 경우에는 서버가 기본 대상이며 더 이상의 조치가 필요하지 않습니다.

## 라이프사이클 관리

이 절에서는 라이프사이클 관리와 관련된 알려진 문제점과 해결 방법을 설명합니다.

ejb-timer-service 등록 정보 minimum-delivery-interval을 9000으로 설정한 후  
ejb-timer-service 등록 정보 redelivery-interval-in-millis를 7000으로 설정하면 set  
명령이 다음 오류와 함께 실패함(아이디 6193449)

### 설명

```
[echo] Doing admin task set
[exec] [Attribute(id=redelivery-interval-internal-in-millis) : Redelivery-
Interval (7,000) should be greater than or equal to Minimum-delivery-
interval-in-millis (9,000)]
[exec] CLI137 Command set failed.
```

- minimum-delivery-interval은 같은 기간 타이머 전달 간의 최소 간격 기간입니다.
- redelivery-interval-in-millis는 타이머 서비스가 실패한 ejbTimeout 후 재전달을 시도하기 전에 기다리는 시간입니다.

문제는 재전달 간격 등록 정보를 최소 전달 등록 정보와 관련시키는 로직이 잘못되어 최소 전달 간격이 재전달 간격보다 큰 곳에서 GUI 또는 CLI를 사용하여 값을 설정하지 못한다는 것입니다.

minimum-delivery-interval-in-millis는 항상 ejb-timer-service 등록 정보 redelivery-interval-in-millis와 같거나 그보다 크게 설정해야 합니다. 문제는 redelivery-interval-in-millis 값이 minimum-delivery-interval-in-millis 값보다 크다고 확인하는 잘못된 검증 확인이 Application Server에 있는 것입니다.

### 해결 방법

다음과 같이 등록 정보의 기본값을 사용합니다.

```
minimum-delivery-interval(default)=7000
redelivery-interval-in-millis(default)=5000
```

기본값 외의 값을 사용하면 오류가 발생합니다.

## 로깅

이 절에서는 로깅과 관련된 알려진 문제점과 해결 방법을 설명합니다.

access.failure에 대한 디버그 문을 설정하면 Application Server 시작이 중단됨(아이디 6180095)

### 설명

JVM에 대해 java.security.debug 옵션을 설정하면 서버 인스턴스 시작이 교착 상태로 중단됩니다. 예를 들어 domain.xml에 다음과 같이 설정하면 이러한 문제가 발생합니다.

```
<jvm-options>-Djava.security.debug=access,failure</jvm-options>
```

### 해결 방법

현재는 해결 방법이 없습니다. 이 플래그를 설정하는 것을 피하십시오.

### JES3 Application Server의 로깅 위치/인스턴스 위치가 변경되었음(아이디 6189409)

Sun Java System 8.2에서는 7.x과(와) 비교하여 기본 로깅 및 서버 인스턴스 위치가 변경되었습니다.

자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.2 관리 설명서** 또는 **Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide**를 참조하십시오.

## Message Queue

이 절에서는 Java Message Queue와 관련된 알려진 문제점과 해결 방법을 설명합니다.

### 타이밍 종속인 특정한 경우 JMS 재연결이 성공적으로 완료되지 않음(아이디 6173308, 6189645, 6198481, 6199510, 6208728)

#### 설명

몇 가지 문제로 인해 타이밍 종속 시나리오에서 재연결에 실패할 수 있습니다.

#### 해결 방법

다음과 같은 방법으로 이 문제를 해결할 수 있습니다.

- 관련된 브로커를 다시 시작
- 관련된 Application Server 인스턴스를 다시 시작

### 비동기 message listener 동작이 appclient 8.0에서 8.1 Update 2로 변경됨(아이디 6198465)

#### 설명

최근 변경 작업으로 인해 비동기 message listener가 app-client 컨테이너의 유일한 활성 스레드일 때 남은 appclient VM은 데몬으로 존재합니다. 이러한 동작은 ACC에서 비동기 수신을 수행하는 과거 응용 프로그램에 대한 회귀입니다. 이 문제는 JMS message listener를 설정하고 주 스레드를 종료하는 응용 프로그램 클라이언트에 영향을 미칩니다.

#### 해결 방법

주 스레드를 종료하지 마십시오. 주 스레드를 종료하기 전에 message listener에서 주 스레드에 알릴 때까지 기다립니다.

## 모니터링

이 절에서는 모니터링과 관련된 알려진 문제점과 해결 방법을 설명합니다.

## 커넥터 서비스 및 커넥터 연결 풀 모니터링 수준을 변경할 수 없음(아이디 6089026)

### 설명

모니터링 수준 설정 페이지를 사용하여 커넥터 서비스 또는 커넥터 연결 풀 모니터링 수준을 **낮음** 또는 **높음**으로 변경한 다음 저장하면 도메인의 `domain.xml` 파일에서 둘 다 변경되지 않습니다. JMS 서비스 모니터링 수준을 **낮음** 또는 **높음**으로 변경한 다음 저장하면 커넥터 서비스 값과 커넥터 연결 풀 값이 동시에 변경됩니다. 해당 명령을 명령줄에서 실행하면 이 문제가 발생하지 않습니다.

### 해결 방법

모니터링 수준 페이지에서 모니터링 수준을 변경하려면 JMS 서비스 구성 요소만 사용하십시오.

## 일부 HTTP 서비스 모니터링 통계는 유용한 정보를 제공하지 않으며 무시해야 함(아이디 6174518)

HTTP 서비스의 일부 요소에 대한 모니터링 통계를 보면 제시된 값이 현재 값과 일치하지 않거나 항상 0인 경우가 있습니다. 특히, 다음 HTTP 서비스 통계는 Application Server에 적용할 수 있는 정보를 제공하지 않으며 무시해야 합니다.

### ■

```
http-service
load1MinuteAverage
load5MinuteAverage
load15MinuteAverage
rateBytesTransmitted
rateBytesReceived
```

### ■ pwc-thread-pool(요소)

배포되지 않은 EJB 모듈의 `mbean`에 대한 모니터링이 해당 모니터링 이름 하의 모든 통계가 이동되더라도 제거되지 않음(아이디 6191092)

### 설명

예를 들면 다음과 같습니다.

```
EJBModuleMonitorMap().size() = 1 eventhough ejb module is
undeployed EJBModuleMonitor().getName() = sqe_ejb_s1_01
```

이러한 현상은 EJB 모듈과 응용 프로그램에 공통적으로 나타납니다. 프로그램(MBean API) 및 `asadmin list/get`을 통해 비어 있는 모니터링인 MBean이 여전히 존재합니다.

### 진단

```
asadmin list -m "server.applications" shows the following output:
server.applications.MEjbApp
server.applications.__ejb_container_timer_app
```

```
server.applications.adminapp
server.applications.admingui
server.applications.com_sun_web_ui
server.applications._export_install_nov-11_domains_domain1_applications
_j2ee-modules_sqe_ejb_s1_01
```

다음과 같은 통계를 볼 수 있습니다.

```
bin/asadmin list -m "server.applications._export_install_nov-11_domains_
_domain1_applications_j2ee-modules_sqe_ejb_s1_01"
server.applications._export_install_nov-11_domains_domain1_applications_
j2ee-modules_sqe_ejb_s1_01.SQEMessage
server.applications._export_install_nov-11_domains_domain1_applications_
j2ee-modules_sqe_ejb_s1_01.TheGreeter
```

일단 배포를 해제합니다.

```
_export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_
ejb_s1_01
```

list 명령을 수행해도 여전히 응용 프로그램을 보게 됩니다.

```
asadmin list -m "server.applications"
server.applications.MEjbApp
server.applications._ejb_container_timer_app
server.applications._export_install_nov-11_domains_domain1_applications_
j2ee-modules_sqe_ejb_s1_01
server.applications.adminapp
server.applications.admingui
server.applications.com_sun_web_ui
```

그러나 모니터링 통계는 포함되어 있지 않습니다.

```
asadmin list -m "server.applications._export_install_nov-11_domains_
domain1_applications_j2ee-modules_sqe_ejb_s1_01"
Nothing to list at server.applications.-export-install-nov-11-domains-
domain1-applications-j2ee-modules-sqe-ejb-s1-01.
```

문자열로 시작하는 유효한 이름을 얻으려면 와일드카드('\*') 문자를 사용합니다. 예를 들어 server로 시작하는 모든 모니터링 가능 항목의 이름 목록을 나열하려면 list "server.\*"를 사용합니다.

## 해결 방법

이 문제는 해가 되지 않습니다. 모듈을 아무 문제 없이 안전하게 재배포할 수 있습니다. 루트 모니터링인 Mbean은 제거되지는 않지만 비어 있습니다.

## 지속성

이 절에서는 Java 데이터 객체 및 컨테이너 관리 지속성과 관련하여 알려진 해결 방법에 대해 설명합니다.

### **JD076018: 순환 종속성 때문에 지속성 인스턴스를 플러시할 수 없음(아이디 6500961)**

#### **설명**

트랜잭션에서 수정되거나 생성된 인스턴스 간의 외래 키 체인 종속성으로 인해 데이터베이스에 순환 종속성이 발생할 경우 이 예외가 반환됩니다.

#### **해결 방법**

원래의 작업들을 여러 트랜잭션으로 분리합니다.

## PointBase

이 절에서는 PointBase와 관련된 알려진 문제점과 해결 방법을 설명합니다.

### **응용 프로그램의 연결 풀에 격리 수준을 설정하면 PointBase에서 예외가 발생함(아이디 6184797)**

#### **설명**

PointBase 데이터베이스 설치를 가리키는 JDBC 연결 풀의 transaction-isolation-level 풀 속성을 기본값(Connection.TRANSACTION\_READ\_COMMITTED) 이외의 값으로 설정하면 예외가 발생합니다. 그러나, 동일한 매개 변수를 다른 데이터베이스를 가리키는 풀의 기본값이 아닌 값으로 설정하면 예외가 발생하지 않습니다.

#### **해결 방법**

PointBase 데이터베이스 설치를 가리키는 JDBC 연결 풀에 대해 transaction-isolation-level을 설정하려고 시도하지 마십시오.

### **네트워크 서버와 내장 드라이버가 함께 사용되는 경우 PointBase에서 예외가 발생함(아이디 6204925)**

#### **설명**

네트워크 서버 드라이버와 내장 드라이버를 동시에 사용하면 번들로 제공된 PointBase에서 예외가 발생하는 경우가 있습니다.

#### **해결 방법**

내장 드라이버나 네트워크 서버 드라이버 중 하나를 사용하고 함께는 사용하지 않습니다.



## 기본 PointBase 데이터베이스를 덮어쓰는 업그레이드 문제(아이디 6264969, 6275448)

### 설명

Application Server Enterprise Edition 8.2로 업그레이드하면 업데이트 릴리스 패치는 Pointbase 기본 데이터베이스를 덮어씁니다.

### 해결 방법

업그레이드 전에 있던 스키마 또는 데이터를 다시 만들거나 다시 입력합니다. 테이블 생성 옵션으로 CMP bean과 함께 응용 프로그램을 배포했으면 해당 응용 프로그램의 배포를 취소하거나 다시 배포하여 테이블을 다시 생성해야 합니다.

## 샘플

이 절에서는 Application Server 8.2 제품에 포함된 샘플 코드와 관련된 알려진 문제점과 해결 방법을 설명합니다.

**setup-one-machine-cluster**는 Windows에서는 중단되지만 Solaris에서는 작동함.  
mqfailover는 Ctrl+C를 눌러 취소한 다음 다시 실행해야 함(아이디 6195092)

### 설명

`install_dir\samples\ee-samples\failover\apps\mqfailover\docs\index.html`에서 다음 명령을 실행합니다.

#### ■ 콘솔 1

```
cd install_dir\samples\ee-samples asant start-mq-master-broker1
```

#### ■ 콘솔 2

```
cd install_dir\samples\ee-samples asant start-mq-cluster-broker1
```

#### ■ 콘솔 3

```
cd install_dir\samples\ee-samples asant start-mq-cluster-broker2
```

#### ■ 콘솔 4

```
cd install_dir\samples\ee-samples asadmin start-domain domain1
```

다른 Enterprise Edition 샘플에 대해 `asant setup-one-machine-cluster-without-ha` 또는 `asant setup-one-machine-cluster-with-ha`를 이미 실행한 경우에는 `asant configure-mq`를 실행하거나 `asant setup-one-machine-cluster-and-configure-mq`를 실행합니다. 이 경우 명령은 성공적으로 실행된 것처럼 보입니다.

```
start_nodeagent: [echo] Start the node agent cluster1-nodeagent
[exec] Command start-node-agent executed successfully.
```

그러나 그런 다음에 시스템이 중단됩니다.

### 해결 방법

현재는 해결 방법이 없습니다. 이 문제는 Windows에서 이 ant 대상을 사용하는 모든 Enterprise Edition 샘플에 비슷한 영향을 미칩니다. 해결 방법은 중단된 프로세스에서 벗어나 Ctrl+C를 눌러 중단된 프로세스를 끝낸 다음 다시 실행하는 것입니다.

asadmin 배포 지침에 따라 MQ 페일오버 샘플 응용 프로그램을 실행하기 전에 JMS 자원을 만들어야 한다는 점이 명시적으로 설명되어 있지 않음(아이디 6198003)

### 설명

다음과 같은 오류가 발생합니다.

```
/opt/SUNWappserver/domains/domain1/config/sun-acc.xml -name
MQFailoverTestClient -textauth -user j2ee -password j2ee
ov 18, 2004 10:50:17 PM com.sun.enterprise.naming.NamingManagerImpl
bindObjects
SEVERE: NAM0006: JMS Destination object not found: jms/durable/TopicA
Nov 18, 2004 10:50:18 PM com.sun.enterprise.naming.NamingManagerImpl
bindObjects
SEVERE: javax.naming.NameNotFoundException
javax.naming.NameNotFoundException
```

asadmin deploy 명령을 사용하여 수동 배포를 수행할 경우 JMS 자원을 직접 만들어야 한다는 점과 샘플 응용 프로그램 배포를 위해 제공된 ant 대상을 사용해야 한다는 점이 명시적으로 설명되어 있지 않습니다.

### 해결 방법

응용 프로그램을 실행하는 데 필요한 JMS 자원을 만드는 build.xml 스크립트에 asant 배포 대상을 사용합니다.

Linux에서 웹 서비스/보안 샘플에서 인증서를 만드는 중 런타임 오류가 표시됨(아이디 6198239)

### 설명

install\_dir/samples/webservices/security 샘플(basicSSL)을 Linux에 배포할 때 인증서는 만들어지지 않고 다음과 유사한 오류가 발생합니다.

```
generate_certs: [echo] ***Exporting certificate from NSS database
[exec] Result: 1 [echo] ***Generating Java Keystore from generated
certificate [exec] keytool error: java.lang.Exception: Input not an
X.509 certificate [exec] Result: 1 [echo] ***Generating Java trust
store from generated certificate [exec] keytool error: java.lang.
Exception: Input not an X.509 certificate [exec] Result: 1
```

```

.
.
.
generate_certs: [echo] ***Exporting server certificate from NSS database to
a PKCS12 certificate file [exec] /opt/sun/appserver/lib/pk12util: /usr/lib/
libnss3.so: version 'NSS_3.9' not found (required by /opt/sun/appserver/lib/
pk12util) [exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so:
version 'NSS_3.6' not found (required by /opt/sun/appserver/lib/pk12util)
[exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version
'NSS_3.7' not found (required by /opt/sun/appserver/lib/pk12util) [exec]
Result: 1

```

문제는 Linux 설치에서 NSS 라이브러리가 Solaris 설치와 다른 위치에 있다는 점입니다. Linux에서 배포할 때 LD\_LIBRARY\_PATH가 올바른 NSS 라이브러리를 가리키는 지 확인해야 합니다. LD\_LIBRARY\_PATH를 사용자 환경 또는 *install\_dir/bin/asant* 쉘 래퍼 스크립트에서 설정합니다.

### 해결 방법

다음 중 한 가지를 수행합니다.

- LD\_LIBRARY\_PATH=/opt/sun/private/lib를 설정합니다.
- *install\_dir/bin/asant* 스크립트에 다음 줄을 추가합니다.

```
LD_LIBRARY_PATH=$AS_NSS:$LD_LIBRARY_PATH;export LD_LIBRARY_PATH
```

## 8.0 Platform Edition에서 8.2 Enterprise Edition으로 업그레이드한 후 샘플 문서가 없음

### 설명

Application Server Platform Edition 8.0에서 Application Server Enterprise Edition 8.2로 업그레이드한 후 샘플 페이지에 액세스하면 HTTP 404 "파일을 찾을 수 없음" 오류가 발생할 수 있습니다.

### 해결 방법

샘플 문서를 8.0 도메인에서 8.2 도메인으로 복사합니다.

## 스파스로컬 영역에서 실행할 때 샘플이 런타임에 실패함(아이디 6460970)

### 설명

Application Server Enterprise Edition 8.2를 Solaris 전역 영역에 설치하고 Application Server 도메인을 스파스로컬 영역에 설치한 경우 배포 중에 스파스 영역의 도메인에서 파일 열기 권한이 없으면 샘플 응용 프로그램을 실행할 수 없습니다.

## 해결 방법

배포 중에 Application Server에서 클라이언트 JAR 파일 `xmsClient.jar`를 검색한 후 샘플 위치(`/usr/SUNWappserver/appserver/samples/webservices/security/ejb/apps/xms/xmsClient.jar`)에 복사할 수 있는지 확인합니다. 이 과정은 일반적으로 샘플 하니스에서 자동으로 실행되지만, `xmsClient.jar` 파일에 대한 권한이 없는 경우 실패합니다.

## 보안

이 절에서는 Application Server 및 웹 응용 프로그램 보안과 인증에 관련된 알려진 문제점과 해결 방법을 설명합니다.

**WebServiceSecurity 응용 프로그램을 Enterprise Edition에서 J2SE 5.0과 함께 실행할 수 없음(아이디 6183318)**

### 설명

WebServiceSecurity 응용 프로그램은 다음과 같은 이유 때문에 J2SE 5.0과 함께 실행할 수 없습니다.

- J2SE 5.0 PKCS11은 UNWRAP 모드를 지원하지 않습니다.
- J2SE 5.0 PKCS11D은 RSA/ECB/OAEPWithSHA1AndMGF1Padding을 PKCS1과 함께 지원하지 않습니다.

J2SE 팀에서는 이 버그를 "CR 6190389: Add support for the RSA-PKCS1 and RSA-OAEP wrap/unwrap mechanisms"로 보고했습니다.

### 해결 방법

J2SE 1.4.2를 기본적으로 포함된 공급자가 아닌 다른 JCE 공급자와 함께 사용합니다. 하드웨어 가속기는 이 구성에 포함되지 않습니다.

**SSL 종료가 작동하지 않음(아이디 6269102)**

### 설명

로드 밸런서(하드웨어)가 SSL 종료에 맞게 구성되었다면 Application Server는 리디렉션 동안 프로토콜을 `https` 에서 `http`로 변경합니다.

### 해결 방법

하드웨어 로드 밸런서와 Application Server 사이에 소프트웨어 로드 밸런서를 추가합니다.

## 업그레이드 유틸리티

이 절에서는 업그레이드 유틸리티와 관련된 알려진 문제점과 해결 방법을 설명합니다.

*install\_dir/domains* 디렉토리가 아닌 사용자 정의 경로에서 생성된 도메인은 **Application Server Enterprise Edition 8**에서 **Application Server Enterprise Edition 8.2**로 업그레이드되는 동안 직접 업그레이드되지 않음(아이디 6165528)

#### 설명

업그레이드 유틸리티를 실행하고 *install\_dir*을 소스 설치 디렉토리로 식별하면 *install\_dir/domains* 디렉토리에 생성되는 도메인만 업그레이드됩니다. 다른 위치에 생성된 도메인은 업그레이드되지 않습니다.

#### 해결 방법

업그레이드 프로세스를 시작하기 전에 모든 도메인 디렉토리를 다른 위치에서 *install\_dir/domains* 디렉토리로 복사합니다.

일부 Linux 시스템에서 "현재 위치에서 업그레이드"를 실행하고 있는 설치 프로그램의 "업그레이드 마법사 시작" 버튼을 누른 후 업그레이드 도구가 시작되지 않음(6207337)

#### 설명

이 문제는 몇몇 Linux 시스템에서 발견되었으며 Java Desktop System 2에서 가장 일반적으로 나타나지만 RedHat 배포에서도 볼 수 있습니다.

설치 프로그램 화면의 업그레이드 도구 시작 버튼을 누른 후 설치 프로그램에서 업그레이드 도구를 시작하여 업그레이드 프로세스를 완료하는 데 실패하고 명령 프롬프트가 반환되지 않은 채 중단됩니다.

#### 해결 방법

이 문제는 명령줄 설치 모드를 사용하여 현재 위치에서 업그레이드를 실행하는 경우에는 발생하지 않습니다.

### ▼ 명령줄 설치 모드를 사용하는 방법

- 1 현재 위치에서 업그레이드를 GUI 모드에서 실행하고 이 문제가 발생하면 설치 프로그램이 시작된 단말기 창에서 **Ctrl+C**를 눌러 설치 프로그램을 종료합니다.

- 2 단말기 창에서 다음 명령을 사용하여 업그레이드 도구를 시작합니다.

```
install_dir/bin/asupgrade --source install_dir/domains --target install_dir
--adminuser adminuser--adminpassword adminpassword --masterpassword changeit
```

*adminuser* 및 *adminpassword*는 업그레이드 중인 설치를 위해 사용되는 값이 일치해야 합니다.

- 3 업그레이드 도구에서 업그레이드 프로세스를 완료하면 브라우저를 시작하고 다음 URL을 입력하여 정보 페이지를 검토할 수 있습니다.

```
file://install_dir/docs/about.html
```

제품을 등록하는 설치 옵션을 선택한 경우에는 제품 정보 페이지에서 등록 페이지로 연결되는 링크를 따라갑니다.

**자체 서명된 인증서는 8.0 Platform Edition(PE)에서 8.1 Enterprise Edition(EE) UR2로 업그레이드 하는 도중 및 후에는 신뢰할 수 없음(아이디 6296105)**

(업그레이드 후) 대상 domain.xml에서 다음 항목을 제거하고 서버를 다시 시작합니다.

```
<jvm-options>-Djavax.net.ssl.keyStore=${com.sun.aas.instanceRoot}
/config/keystore.jks</jvm-options>-
<jvm-options>Djavax.net.ssl.trustStore=${com.sun.aas.instanceRoot}
/config/cacerts.jks</jvm-options>
```

## Application Server를 JES2에서 JES5로 업그레이드한 후 포트 충돌

### 설명

Application Server 7.x에서 8.2로 업그레이드하면 이전 설치와 새 설치 간에 포트 충돌이 발생할 수 있습니다. 주로 기본 포트 8080과 8181이 충돌합니다.

### 해결 방법

포트 충돌을 해결하려면 Application Server 8.2에서 사용되는 포트를 변경합니다.

**샘플 스크립트에 사용되는 Derby 데이터베이스가 잘못된 위치에 만들어짐(아이디 6377804)**

### 설명

이 버그는 다음과 같은 두 가지 특징이 있습니다.

1. Derby 데이터베이스를 사용하는 샘플 응용 프로그램 설정 스크립트를 실행하면 Derby 데이터베이스가 현재 디렉토리 또는 <install\_root>/bin에 만들어집니다.
2. 샘플 build Ant 스크립트는 현재 디렉토리에서 관리 비밀번호 파일을 저장하는 password.txt 파일을 만듭니다. 루트가 아닌 스파스 영역 시나리오에서는 이 파일에 쓸 수 없습니다.

### 해결 방법

1. **Derby 데이터베이스 위치** - --dbhome 옵션을 start-database 명령과 함께 사용하여 --dbhome에 대해 지정된 값에 데이터베이스를 만듭니다. 예를 들어, start-database에 대한 asadmin 명령 구문은 다음과 같습니다.

```
start-database [--dbhost 0.0.0.0] [--dbport 1527] [--dbhome db_directory] [--echo=false]
[--verbose=false]
```

2. **password.txt 파일의 위치** - 모든 빌드 명령은 password.txt 파일을 해당 디렉토리에 만들도록 설계되었기 때문에 샘플 디렉토리를 쓸 수 있습니다. 쓸 수 있는 위치에 샘플의 작업 복사본을 설치합니다.

## 8.0UR1PE에서 8.2EE로 업그레이드하는 중에 LoginException이 발생함. 업그레이드 프로세스가 중지됨(아이디 6445419)

### 설명

이 문제는 기본값 이외의 관리자 자격 증명을 사용하여 업그레이드 설치를 실행할 때 발생합니다.

### 해결 방법

8.xPE의 파일 기반 설치 프로그램을 사용하여 8.2EE로 병렬 업그레이드할 경우 새 Application Server에 대해 다음과 같은 관리자 자격 증명을 사용합니다.

- 관리자: admin
- 관리자 비밀번호: adminadmin
- 마스터 비밀번호: changeit

업그레이드를 수행한 후 필요에 따라 비밀번호를 변경할 수 있습니다.

## 업그레이드 도구가 기존 디렉토리를 감지하지 못하고 소스 디렉토리 필드에 잘못된 디렉토리를 입력함(아이디 6460122)

### 설명

업그레이드 도구가 기존 디렉토리를 감지하지 못하고 소스 디렉토리 필드에 잘못된 디렉토리를 입력하므로 디렉토리 구성이 맞는 듯한 느낌을 줍니다.

소스 디렉토리에 대한 잘못된 경로가 입력될 경우 "잘못된 디렉토리" 메시지가 표시될 것으로 기대합니다. /opt/SUNWappserverEE81UR2/가 소스 디렉토리에 입력되는 경우에는 잘못된 디렉토리라는 메시지가 제대로 표시됩니다. /opt/SUNWappserverEE81UR2/domains가 입력되는 경우에는 경로가 잘못되었더라도 경고 메시지를 표시하지 않고 업그레이드를 계속 수행합니다. 이 문제는 입력 값에 따라 동작이 달라진다는 점만 제외하고 아이디 6440710과 비슷합니다.

### 해결 방법

Application Server 7 또는 8.x에서 Application Server 8.2로 업그레이드할 때 설명서에 권장하는 값을 소스 디렉토리에 입력해야 합니다. 즉, 제자리에서 업그레이드할 경우 도메인 루트를 입력하고, 병렬 업그레이드를 수행할 경우 도메인 디렉토리를 입력해야 합니다.

## 관리자/비밀번호 이름에 세미콜론(;) 문자를 사용해서는 안 됨(아이디 6473341)

### 설명

Application Server Enterprise Edition 8.2 설치에서는 관리자 이름에 특수 문자를 사용할 수 없습니다. 특수 문자를 사용할 경우 도메인이 만들어지지 않습니다. 관리 비밀번호에는 특수 문자를 사용할 수 있습니다.

## 해결 방법

Application Server 7에서 Application Server 8.2로 업그레이드하는 경우 관리자 이름에 특수 문자가 포함되어 있는지 확인합니다.

## 웹 컨테이너

이 절에서는 웹 컨테이너와 관련된 알려진 문제점과 해결 방법을 설명합니다.

**Windows에서 --precompilejsp=true를 사용하여 응용 프로그램을 배포하면 응용 프로그램에서 JAR 파일이 잠길 수 있으며 이렇게 되면 나중에 배포를 해제하거나 재배포할 때 실패하는 원인이 됨(아이디 5004315)**

### 설명

Windows에서 응용 프로그램을 배포할 때 JSP의 사전 컴파일을 요청하고 나중에 해당 응용 프로그램의 배포를 해제하거나 해당 응용 프로그램(또는 동일한 모듈 아이디를 가진 응용 프로그램)을 재배포하려고 시도하면 예상한 것처럼 작동하지 않습니다. 문제는 JSP 사전 컴파일을 수행하면 응용 프로그램의 JAR 파일을 열지만 닫지 않고 Windows에서는 배포 해제 시 그러한 파일을 삭제하지 못하거나 재배포 시 덮어쓰지 못합니다.

배포 해제는 응용 프로그램이 Application Server에서 논리적으로 제거된다는 점에서 어느 정도는 성공한 것으로 볼 수 있습니다. 또한 **asadmin** 유틸리티는 오류 메시지를 반환하지 않지만 응용 프로그램의 디렉토리 및 잠긴 **jar** 파일은 서버에 남아 있습니다. 서버의 로그 파일에는 파일 및 응용 프로그램 디렉토리를 삭제하는 데 실패한 것을 설명하는 메시지가 포함됩니다.

배포 해제에 실패한 후 응용 프로그램을 재배포하려는 시도는 서버에서 기존 파일과 디렉토리를 제거하려고 하기 때문에 역시 실패하게 됩니다. 이러한 문제는 원래 배포한 응용 프로그램과 동일한 모듈 아이디를 사용하는 응용 프로그램을 배포하려고 시도하면 서버가 응용 프로그램 파일을 저장할 디렉토리 이름을 선택할 때 모듈 아이디를 사용하기 때문에 발생할 수 있습니다.

먼저 응용 프로그램의 배포를 해제하지 않고 재배포하려고 시도하는 경우도 같은 이유 때문에 실패합니다.

### 진단

응용 프로그램의 배포를 해제한 후 재배포하려고 시도하면 **asadmin** 유틸리티는 아래와 유사한 오류를 반환합니다.

```
An exception occurred while running the command. The exception
message is: CLI171 Command deploy failed : Deploying application in
domain failed; Cannot deploy. Module directory is locked and can't
be deleted.
```



## 해결 방법

응용 프로그램을 배포할 때 `--precompilejsp=false`(기본 설정)를 지정한 경우에는 이 문제가 발생하지 않습니다. 응용 프로그램을 처음 사용하면 JSP 컴파일이 트리거되어 첫 번째 요청에 대한 응답 시간은 이후의 요청에 대한 응답 시간보다 더 길습니다.

사전 컴파일을 수행하면 응용 프로그램을 배포 해제 또는 재배포하기 전에 서버를 중단하고 다시 시작해야 합니다. 서버를 종료하면 잠긴 JAR 파일의 잠금이 해제되어 재시작한 후 배포 해제 또는 재배포를 성공적으로 수행할 수 있습니다.

## WAR을 빈 <load-on-startup> 요소가 있는 Servlet 2.4 기반 web.xml과 함께 배포할 수 없음(아이디 6172006)

### 설명

web.xml 파일에서 선택 요소인 `load-on-startup` 서블릿 요소는 연관된 서블릿이 선언된 웹 응용 프로그램 시작의 일부로 로드되고 초기화된다는 것을 나타냅니다.

이 요소의 옵션 부분은 서블릿이 웹 응용 프로그램의 다른 서블릿과 관련하여 로드되고 초기화되는 순서를 나타내는 정수입니다. `<load-on-startup>` 요소가 비어 있으면 서블릿이 웹 응용 프로그램을 포함하는 시작 중에 로드되고 초기화되는 한 순서는 상관없이 없음을 의미합니다.

web.xml을 위한 Servlet 2.4 스키마는 더 이상 빈 `<load-on-startup>`을 지원하지 않습니다. 즉, Servlet 2.4 기반 web.xml을 사용할 때 정수를 지정해야 합니다. `<load-on-startup>`과 같이 빈 `<load-on-startup/>`을 지정하면 web.xml은 web.xml을 위한 Servlet 2.4 스키마에 대한 검증에 실패하며 이로 인해 웹 응용 프로그램 배포에 실패하게 됩니다.

역호환성 문제로 빈 `<load-on-startup>`을 지정해도 Servlet 2.3 기반의 web.xml과는 아무 문제 없이 작동합니다.

## 해결 방법

Servlet 2.4 기반 web.xml을 사용하여 서블릿 로드 순서가 중요하지 않다는 것을 나타낼 때 `<load-on-startup>0</load-on-startup>`을 지정합니다.

## 자원이 제한된 서버에서 JSP 페이지를 컴파일할 수 없음(아이디 6184122)

### 설명

JSP 페이지에 액세스하지만 컴파일에 실패하고 서버 로그에는 다음과 같은 스택 추적과 함께 "Unable to execute command"라는 오류 메시지가 포함됩니다.

```
at org.apache.tools.ant.taskdefs.Execute$Java13CommandLauncher.
exec(Execute.java:655) at org.apache.tools.ant.taskdefs.Execute.
launch(Execute.java:416)
at org.apache.tools.ant.taskdefs.Execute.execute(Execute.java:427)
at org.apache.tools.ant.taskdefs.compilers.DefaultCompilerAdapter.
```

```
executeExternalCompile(DefaultCompilerAdapter.java:448)
at org.apache.tools.ant.taskdefs.compilers.JavacExternal.execute
(JavacExternal.java:81)
at org.apache.tools.ant.taskdefs.Javac.compile(Javac.java:842)
at org.apache.tools.ant.taskdefs.Javac.execute(Javac.java:682)
at org.apache.jasper.compiler.Compiler.generateClass(Compiler.java:396)
```

## 해결 방법

JSP 컴파일 스위치인 "fork"를 "false"로 설정합니다.

이 작업은 다음 중 한 가지 방법으로 수행할 수 있습니다.

- 전역적으로 \${S1AS\_HOME}/domains/domain1/config/default-web.xml에서 JspServlet의 fork init 매개 변수를 false로 설정합니다.

```
<servlet> <servlet-name>jsp</servlet-name>
<servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class>
.... <init-param>
<param-name>fork</param-name> <param-value>>false</param-value>
</init-param> </servlet>
```

- 웹 응용 프로그램별로 sun-web.xml의 fork JSP 구성 등록 정보를 false로 설정합니다.

```
<sun-web-app> <jsp-config> <property name="fork" value="false" />
</jsp-config> </sun-web-app>
```

어떤 방법으로 설정하든 ant에서 javac 컴파일을 위한 새로운 프로세스를 생성하지 못하도록 합니다.

**Application Server는 auth-passthrough Web Server 6.1 Add-On을 지원하지 않음(아이디 6188932)**

## 설명

Sun Java System Application Server Enterprise Edition 8.2는 Sun Java System Application Server Enterprise Edition 7.1과 함께 사용할 수 있는 auth-passthrough 플러그인 함수에서 제공하는 기능에 대한 지원을 추가합니다. 그러나 Application Server Enterprise Edition 8.2에서 auth-passthrough 플러그인 기능은 다르게 구성됩니다.

Application Server Enterprise Edition 7.1에서 auth-passthrough 플러그인 함수는 2계층 배포 시나리오에서 사용할 수 있었습니다. 여기서,

- Application Server 인스턴스가 회사 방화벽 뒤의 두 번째 방화벽에 의해 보호됩니다.
- Application Server 인스턴스에 직접 허용된 클라이언트 연결이 없습니다.

이러한 네트워크 아키텍처에서 클라이언트는 프론트엔드 웹 서버에 연결됩니다. 이 웹 서버는 service-passthrough 플러그인 함수로 구성되어 있으며 프록시를 거친 Application Server 인스턴스에 처리하도록 HTTP 요청을 전달합니다. Application Server

인스턴스는 웹 서버 프록시의 요청만을 받을 수 있는데 클라이언트 호스트로부터는 직접 받지 못합니다. 결과적으로, 클라이언트의 IP 주소 같은 클라이언트 정보를 쿼리하는 프록시를 거친 Application Server 인스턴스에 배포된 응용 프로그램은 프록시 호스트 IP를 받습니다. 이것이 전달된 요청의 실질적인 보낸 호스트이기 때문입니다.

## 해결 방법

Application Server Enterprise Edition 7.1에서는, 프록시를 거친 Application Server 인스턴스가 service-passthrough 플러그인이 실행되고 있는 중간 웹 서버를 통하지 않고 직접 요청을 받은 것처럼 프록시를 거친 Application Server 인스턴스에 배포된 모든 응용 프로그램에서 원격 클라이언트 정보를 직접 사용할 수 있도록 해당 인스턴스에서 auth-passthrough 플러그인 함수를 구성할 수 있습니다.

Application Server Enterprise Edition 8.2에서 auth-passthrough 기능은 다음과 같이 domain.xml의 <http-service> 요소의 authPassthroughEnabled 등록 정보를 TRUE로 설정하여 사용할 수 있습니다.

```
<property name="authPassthroughEnabled" value="true"/>
```

Application Server Enterprise Edition 7.1에서 auth-passthrough 플러그인 함수의 동일한 보안 고려 사항은 Application Server Enterprise Edition 8.2의 authPassthroughEnabled 등록 정보에도 적용됩니다. authPassthroughEnabled를 사용하면 인증 목적으로 사용될 수 있는 정보(예를 들어 요청을 보낸 측의 IP 주소 또는 SSL 클라이언트 인증서)를 무시하는 것이 가능하기 때문에 authPassthroughEnabled를 TRUE로 설정하여 Application Server Enterprise Edition 8.2 인스턴스에 신뢰할 수 있는 클라이언트나 서버만 연결할 수 있도록 하는 것이 중요합니다. 더욱 주의하는 의미에서 회사 방화벽 뒤의 서버에서만 authPassthroughEnabled를 TRUE로 설정하여 구성하는 것이 좋습니다. 인터넷을 통해 액세스할 수 있는 서버는 authPassthroughEnabled를 TRUE로 설정하여 구성해서는 안 됩니다.

프록시 웹 서버가 service-passthrough 플러그인으로 구성되어 있고 authPassthroughEnabled가 TRUE로 설정된 Application Server 8.1 Update 2 인스턴스에 요청을 전달하는 시나리오에서 SSL 클라이언트 인증은 웹 서버 프록시에서 사용할 수 있으며 프록시를 거친 Application Server 8.1 Update 2 인스턴스에서는 사용할 수 없습니다. 이 경우 프록시를 거친 Application Server 8.1 Update 2 인스턴스는 마치 SSL을 통해 인증된 것처럼 요청을 취급하고 클라이언트의 SSL 인증서를 요청하는 배포된 응용 프로그램에 해당 인증서를 제공합니다.

--enabled=false를 사용하여 만든 HTTP Listener에서 Listener를 비활성화할 수 없음(아이디 6190900)

## 설명

--enabled=false 플래그를 사용하여 httplistener를 만들 경우 해당 수신기가 비활성화되지 않습니다. --enabled 플래그는 수신기를 만드는 동시에 사용되는 경우에는 적용되지 않습니다.

## 해결 방법

수신기를 활성 상태에서 만든 다음 나중에 수동으로 비활성화합니다.

**Windows에서는 verify\_file\_user\_exists\_common이 실행되지 않기 때문에 재배포가 실패함(아이디 6490227)**

## 설명

Windows에서 배포하기 전에 사용자를 만드는 응용 프로그램을 재배포하면 verify\_file\_user\_exists\_common을 호출하더라도 실행되지 않기 때문에 create-file-user 명령이 실패하여 사용자가 이미 있는지 식별하지 못합니다. 이 지점에서 deploy 대상의 실행이 중지되고 배포 및 배포 해제가 실패합니다.

## 해결 방법

먼저 keydel 대상을 사용하여 파일 사용자를 삭제한 다음 deploy 대상을 다시 실행합니다.

```
asant keydel
asant deploy
```