



Sun Java System Application Server 9.1 빠른 시작 설명서



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

부품 번호: 820-3029
2007년 9월

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

이 제품 또는 문서는 저작권에 의해 보호되며 사용, 복사, 배포 및 역변환을 제한하는 라이선스로 배포됩니다. 본 제품 또는 설명서의 어떠한 부분도 Sun 및 Sun 소속 라이선스 부여자(있는 경우)의 사전 서면 승인 없이 어떠한 형태나 수단으로도 재생산할 수 없습니다. 글꼴 기술을 포함한 타사 소프트웨어에 대한 저작권 및 사용권은 Sun 공급업체에 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, docs.sun.com, AnswerBook, AnswerBook2, 및 Solaris는 미국 및 다른 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다. 모든 상표는 사용허가를 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 Sun™ 그래픽 사용자 인터페이스(GUI)는 Sun Microsystems, Inc.가 자사의 사용자 및 정식 사용자로 개발했습니다. Sun은 컴퓨터 업계에 대한 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.

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

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.

목차

머리말	5
1 기본 기능의 빠른 시작	11
Application Server 관리 도구	11
서버 시작	13
▼ Solaris 및 Linux에서 도메인을 시작하는 방법	13
▼ Windows에서 도메인을 시작하는 방법	14
▼ 관리 콘솔에 로그인하는 방법	15
▼ Domain Administration Server 로그 파일을 보는 방법	16
▼ 노드 에이전트를 시작하는 방법	17
▼ 도메인에 클러스터 지원을 추가하는 방법	19
▼ 번들로 제공된 Java DB 서버를 시작하는 방법	19
다음 작업	20
2 클러스터 설정	21
클러스터 만들기	21
▼ 관리 콘솔을 사용하여 클러스터를 만드는 방법	21
▼ CLI 명령을 사용하여 클러스터를 만드는 방법	22
클러스터링된 서버 인스턴스의 포트 보기	23
▼ 클러스터링된 서버 인스턴스의 포트를 확인하는 방법	23
3 응용 프로그램 배포	25
샘플 웹 응용 프로그램 배포	25
▼ 자동 배포를 이용하여 Hello 응용 프로그램을 배포하는 방법	25
▼ 관리 콘솔을 이용하여 Hello 응용 프로그램을 배포하는 방법	26
클러스터에 샘플 엔터프라이즈 응용 프로그램 배포	28
▼ 관리 콘솔을 이용하여 샘플 엔터프라이즈 응용 프로그램을 배포하는 방법	28

▼ 클러스터에서 인스턴스를 시작하는 방법	29
▼ 응용 프로그램 배포를 확인하는 방법	29
4 로드 밸런서 작업	31
로드 균형 조정 설정	32
▼ 로드 균형 조정을 설정하는 방법	32
로드 균형 조정 시작	33
로드 균형 조정 확인	33
▼ 로드 균형 조정을 확인하는 방법	33
메모리 내 복제 기능을 이용한 고가용성 및 페일오버	34
5 고가용성 페일오버 설정	37
고가용성 클러스터와 HADB	38
HADB 사전 설치 절차	38
▼ HADB를 위한 시스템을 구성하는 방법	39
HADB 설치	40
▼ HADB를 설치하는 방법	40
HADB 시작	41
▼ Solaris 또는 Linux의 Java Enterprise System 설치에서 HADB를 시작하는 방법	41
▼ Windows의 Java Enterprise System 설치에서 HADB를 시작하는 방법	41
▼ Solaris 또는 Linux의 독립 실행형 설치에서 HADB를 시작하는 방법	42
▼ Windows의 독립 실행형 설치에서 HADB를 시작하는 방법	42
고가용성을 위한 클러스터 및 응용 프로그램 구성	42
클러스터 다시 시작	42
▼ 클러스터를 다시 시작하는 방법	42
HTTP 세션 페일오버 확인	43
▼ HTTP 세션 페일오버를 확인하는 방법	43

머리말

이 빠른 시작 설명서에서는 서버 시작, 응용 프로그램 배포 및 로드 균형 조정/페일오버 설정을 위한 기본 절차를 제공합니다.

이 장에서는 전체 Sun Java™ System Application Server 설명서 세트에 대한 정보와 규칙이 제공됩니다.

Application Server 설명서 세트

Application Server 설명서 세트에서는 배포 계획 및 시스템 설치에 대해 설명합니다. Application Server 설명서에 대한 URL(Uniform Resource Locator)은 <http://docs.sun.com/coll/1343.4> 및 <http://docs.sun.com/coll/1776.1>입니다. Application Server에 대한 소개는 다음 표의 설명서를 나열된 순서대로 참조하십시오.

표 P-1 Application Server 설명서 세트에 포함된 설명서

설명서 제목	설명
Documentation Center(설명서 센터)	작업과 주제로 구성된 Application Server 설명서 항목입니다.
릴리스 노트	소프트웨어 및 설명서 관련 최신 정보로 지원되는 하드웨어, 운영 체제, JDK™(Java Development Kit) 및 데이터베이스 드라이버를 표를 기반으로 종합적으로 요약하였습니다.
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.
Application Deployment Guide	응용 프로그램과 응용 프로그램 구성 요소를 Application Server에 배포하는 방법에 대해 설명합니다. 배포 설명자에 대한 정보를 제공합니다.

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

설명서 제목	설명
Developer's Guide	Java EE 구성 요소 및 API용 개방형 Java 표준 모델을 따르는 Application Server에서 실행할 Java Platform, Enterprise Edition(Java EE 플랫폼) 응용 프로그램을 만들고 구현하는 방법에 대해 설명합니다. 개발자 도구, 보안, 디버깅 및 라이프사이클 모듈 생성에 대한 정보를 제공합니다.
Java EE 5 Tutorial	Java EE 5 플랫폼 기술과 API를 사용하여 Java EE 응용 프로그램을 개발하는 방법에 대해 설명합니다.
Java WSIT Tutorial	WSIT(Web Service Interoperability Technologies)를 사용하여 웹 응용 프로그램을 개발하는 방법에 대해 설명합니다. WSIT 기술과 각 기술이 지원하는 기능 및 옵션을 사용하는 방법, 시기 및 이유에 대해 설명합니다.
관리 설명서	구성, 모니터링, 보안, 자원 관리 및 웹 서비스 관리를 비롯한 Application Server 시스템 관리에 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	Application Server의 이전 버전에서 업그레이드하거나 경쟁 응용 프로그램 서버에서 Java EE 응용 프로그램을 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 Sun Java Enterprise System(Java ES)의 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.

Sun Java System의 다른 독립 실행형 서버 제품에 대한 설명서를 보려면 다음을 참조하십시오.

- Message Queue 설명서(<http://docs.sun.com/coll/1343.4> 및 <http://docs.sun.com/coll/1776.1>)

- Directory Server 설명서(<http://docs.sun.com/coll/1224.1> 및 <http://docs.sun.com/coll/1586.1>)
- Web Server 설명서(<http://docs.sun.com/coll/1308.3> 및 <http://docs.sun.com/coll/1410.2>)

Application Server와 함께 제공되는 패키지에 대한 Javadoc™ 도구 참조는 <http://glassfish.dev.java.net/nonav/javaee5/api/index.html>에 있습니다. 또한 다음과 같은 자원을 사용할 수 있습니다.

- Java EE 5 사양 (<http://java.sun.com/javaee/5/javatech.html>)
- Java EE Blueprints (<http://java.sun.com/reference/blueprints/index.html>)

NetBeans™ IDE(Integrated Development Environment)에서 엔터프라이즈 응용 프로그램을 만드는 방법에 대한 자세한 내용은 <http://www.netbeans.org/kb/55/index.html>을 참조하십시오.

Application Server에 포함된 Java DB 데이터베이스에 대한 자세한 내용은 <http://developers.sun.com/javadb/>를 참조하십시오.

GlassFish Samples 프로젝트는 광범위한 Java EE 기술을 보여주는 샘플 응용 프로그램 모음입니다. GlassFish Samples는 Java EE SDK(Software Development Kit)와 함께 번들로 제공되며 GlassFish Samples 프로젝트 페이지(<https://glassfish-samples.dev.java.net/>)에서도 사용 가능합니다.

기본 경로 및 파일 이름

다음 표에서는 본 설명서에서 사용한 기본 경로와 파일 이름에 대해 설명합니다.

표 P-2 기본 경로 및 파일 이름

자리 표시자	설명	기본값
<i>install-dir</i>	Application Server의 기본 설치 디렉토리를 나타냅니다.	Solaris™에서 Java ES와 함께 설치된 Application Server, 루트 사용자의 경우: /opt/SUNWappserver/appserver Linux에서 Java ES와 함께 설치된 Application Server, 루트 사용자의 경우: /opt/sun/appserver/ 기타 Solaris 및 Linux 설치, 루트가 아닌 사용자의 경우: <i>user's-home-directory</i> /SUNWappserver 기타 Solaris 및 Linux 설치, 루트 사용자의 경우: /opt/SUNWappserver Windows, 모든 설치: <i>SystemDrive</i> : \Sun\AppServer
<i>domain-root-dir</i>	모든 도메인을 포함한 디렉토리를 나타냅니다.	Java ES Solaris 설치, 루트 사용자의 경우: /var/opt/SUNWappserver/domains/ Java ES Linux 설치, 루트 사용자의 경우: /var/opt/sun/appserver/domains/ 모든 기타 설치: <i>install-dir</i> /domains/
<i>domain-dir</i>	도메인용 디렉토리를 나타냅니다. 구성 파일에서 <i>domain-dir</i> 이 다음과 같이 표시되어 있을 수 있습니다. \${com.sun.aas.instanceRoot}	<i>domain-root-dir</i> / <i>domain-dir</i>
<i>instance-dir</i>	서버 인스턴스용 디렉토리를 나타냅니다.	<i>domain-dir</i> / <i>instance-dir</i>

활자체 규약

다음 표에서는 본 설명서에 사용된 활자체 규약에 대해 설명합니다.

표 P-3 활자체 규약

서체	의미	예
AaBbCc123	명령, 파일 및 디렉토리의 이름과 컴퓨터 화면에 출력되는 내용	.login 파일을 편집합니다. 모든 파일을 나열하려면 <code>ls -a</code> 를 사용합니다. machine_name% you have mail.
AaBbCc123	화면 상의 컴퓨터 출력과는 반대로 사용자가 직접 입력하는 사항입니다.	machine_name% su Password:
AaBbCc123	명령줄 자리 표시자: 실제 이름이나 값으로 대체됩니다.	파일을 삭제하려면 <code>rm filename</code> 을 입력하십시오.
AaBbCc123	책 제목, 새로 나오는 용어, 강조 표시할 단어입니다. (강조 표시된 일부 항목은 온라인에서 볼드체로 표시됩니다.)	사용자 설명서 의 6장을 참조하십시오. <code>cache</code> 는 로컬로 저장된 복사본입니다. 파일을 저장하지 마십시오 .

기호 규칙

다음 표에서는 본 설명서에 사용된 기호에 대해 설명합니다.

표 P-4 기호 규칙

기호	설명	예	의미
[]	선택 인수 및 명령 옵션을 포함합니다.	<code>ls [-l]</code>	-l 옵션은 사용하지 않아도 됩니다.
{ }	필수 명령 옵션에 대한 일련의 선택 항목을 포함합니다.	<code>-d {y n}</code>	-d 옵션에서는 y 인수나 n 인수를 사용해야 합니다.
\${ }	변수 참조를 나타냅니다.	<code>\${com.sun.javaRoot}</code>	com.sun.javaRoot 변수 값을 참조합니다.
-	동시에 입력하는 여러 키를 결합합니다.	Control-A	Ctrl 키를 누른 채로 A 키를 누릅니다.
+	연속해서 입력하는 여러 키를 결합합니다.	Ctrl+A+N	Ctrl 키를 눌렀다가 놓은 다음 후속 키를 누릅니다.
→	그래픽 사용자 인터페이스의 메뉴 항목 선택을 나타냅니다.	파일 → 새로 만들기 → 템플릿	파일 메뉴에서 새로 만들기를 선택합니다. 새로 만들기 하위 메뉴에서 템플릿을 선택합니다.

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/>)

Sun 제품 설명서 검색

docs.sun.comSM 웹 사이트에서 Sun 제품 설명서를 검색하는 방법 이외에 검색 필드에 다음 구문을 입력하여 검색 엔진을 이용하는 방법도 있습니다.

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

예를 들어 "broker"를 검색하려면 다음을 입력합니다.

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

검색에 다른 Sun 웹 사이트를 포함하려면(예: java.sun.com, www.sun.com 및 developers.sun.com) 검색 필드에 docs.sun.com 대신 sun.com을 입력합니다.

타사 웹 사이트 참조 사항

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.

사용자 의견 환영

Sun은 설명서의 내용을 지속적으로 개선하고자 하며 사용자 여러분의 의견과 제안을 환영합니다. 사용자 의견을 보내시려면 <http://docs.sun.com>에서 의견 보내기를 누르십시오. 온라인 양식에서 전체 문서 제목과 부품 번호를 기입해 주십시오. 부품 번호는 해당 설명서의 제목 페이지나 문서의 URL에 있으며 7자리 또는 9자리 숫자로 되어 있습니다. 예를 들어, 이 설명서의 부품 번호는 820-3029입니다.

기본 기능의 빠른 시작

빠른 시작 설명서를 시작합니다. 이 설명서에는 Application Server를 시작하기 위해 사용할 수 있는 샘플 절차 세트가 제공됩니다.

이 문서에 나와 있는 절차를 사용하기 전에 Application Server 소프트웨어를 설치해야 합니다.

이 절은 다음 내용으로 구성되어 있습니다.

- 11 페이지 “Application Server 관리 도구”
- 13 페이지 “서버 시작”
- 20 페이지 “다음 작업”

Application Server 관리 도구

관리자가 여러 호스트에서 실행되고 있는 서버 인스턴스와 클러스터를 관리할 수 있도록 Application Server에서는 다음 도구를 제공합니다.

- 브라우저 기반 그래픽 사용자 인터페이스(GUI)인 관리 콘솔
- `asadmin` 유틸리티와 같은 명령줄 도구. Application Server에서 사용 가능한 명령줄 도구의 전체 목록을 보려면 표 1-1을 참조하십시오.
- 프로그래밍 방식의 Java Management Extensions(JMX™) API

이 도구는 모든 관리 작업을 중개하도록 특별히 지정된 Application Server 인스턴스인 *Domain Administration Server*라는 서버에 연결됩니다. *Domain Administration Server*(DAS)는 사용되는 인터페이스에 상관없이 관리 명령을 검증 및 실행하는 단일의 보안 인터페이스를 제공합니다.

도메인이란 구성 데이터, 배포된 응용 프로그램 및 지정된 관리자가 있는 시스템의 모음입니다. 도메인 정의에서는 여러 응용 프로그램, 독립 실행형 응용 프로그램 서버 인스턴스 및 클러스터의 작업을 설명하고 제어할 수 있으며 잠재적으로 여러 시스템에 분산되어 있을 수 있습니다. DAS를 설치할 때 항상 *domain1*이라는 기본 도메인이 설치됩니다. 본 설명서에서는 기본 도메인을 사용합니다.

Application Server 소프트웨어에서는 관리 콘솔 이외에 관리 기능을 수행하기 위한 다양한 명령줄 도구를 제공합니다. 도구를 실행하려면 명령 창에 도구 이름을 입력합니다. 표 1-1의 첫 번째 열에는 도구 이름이 나열되어 있고 두 번째 열은 해당 도구에 대한 설명입니다.

표 1-1 명령줄 도구

도구 이름	설명
appclient	Application Client Container를 시작하고 응용 프로그램 JAR(Java ARchive) 파일에 패키징된 클라이언트 응용 프로그램을 호출합니다.
asadmin	Application Server 소프트웨어를 구성하기 위한 하위 명령 세트를 제공하는 Application Server 관리 도구를 실행합니다.
asant	반복적인 개발 및 배포 작업을 자동화할 수 있도록 Jakarta Ant 도구를 실행합니다.
asapt	Java EE 주석이 있는 Java 소스를 컴파일합니다. 도구에서 자동으로 wsimport 명령을 호출합니다.
asupgrade	Application Server 소프트웨어를 업그레이드하기 위한 Application Server 관리 도구입니다.
capture-schema	데이터베이스에서 스키마 정보를 추출하여 서버에서 CMP(Container Managed Persistence)에 사용할 수 있는 스키마 파일을 생성합니다.
jspc	JSP 페이지를 컴파일합니다.
package-appclient	응용 프로그램 클라이언트 컨테이너 라이브러리와 JAR 파일을 패키징합니다.
schemagen	Java 클래스에서 참조되는 각 이름 공간에 대한 스키마 파일을 만듭니다.
verifier	DTD를 사용하여 Java EE 배포 설명자를 검증합니다. 이 도구에는 그래픽 사용자 인터페이스도 제공됩니다. GUI를 보려면 -u 옵션을 지정합니다. 일부 Windows 시스템에서는 이름이 같은 드라이버 검증자 유틸리티가 실행됩니다. Application Server 검증자를 실행하려면 <i>install-dir/bin</i> 디렉토리에 있어야 합니다.
wscompile	서비스 정의 인터페이스를 사용하여 JAX-RPC의 클라이언트 스텝 또는 서버측 스텝을 생성하거나 제공된 인터페이스에 대한 WSDL(Web Services Description Language) 설명을 생성합니다.
wsdeploy	JAX-RPC를 사용하는 웹 서비스 응용 프로그램에 대한 구현별 배포 대기 WAR 파일을 생성합니다.

표 1-1 명령줄 도구 (계속)

도구 이름	설명
wsgen	웹 서비스 종점 클래스를 읽고 웹 서비스 배포 및 호출에 필요한 모든 아티팩트를 생성합니다.
wsimport	SEI(Service Endpoint Interface), 서비스, <code>wSDL:fault</code> 및 <code>soap:headerfault</code> 태그에서 매핑된 예외 클래스, <code>wSDL:message</code> 태그에서 파생된 비동기 응답 Bean, JAXB 생성 값 유형과 같은 이식 가능한 JAX-WS 아티팩트를 생성합니다.
xjc	Java 프로그래밍 언어로 소스 XML 스키마를 JAXB 내용 클래스 집합으로 변환하거나 바인드합니다.

주 -

- 이러한 명령줄 도구를 Windows에서 실행하려면 `install-dir/bin/` 디렉토리를 가리키는 PATH라는 환경 변수가 있는지 확인합니다.
- 모든 하위 명령 앞에 `asadmin` 접두어를 사용하거나 `install-dir/bin/` 디렉토리에 `asadmin` 프롬프트를 입력(`asadmin` 입력 후 Return 키 누름)하여 `asadmin` 하위 명령을 실행할 수 있습니다.

서버 시작

이 항목에서는 Domain Administration Server(DAS) 시작,관리 콘솔 로그인, 로그 파일 검사 및 노드 에이전트 시작과 같은 기본 작업에 대한 지침을 설명합니다. 각 단계는 마쳐야 하는 순서 순으로 표시되어 있습니다.

- 13 페이지 “Solaris 및 Linux에서 도메인을 시작하는 방법”
- 14 페이지 “Windows에서 도메인을 시작하는 방법”
- 15 페이지 “관리 콘솔에 로그인하는 방법”
- 16 페이지 “Domain Administration Server 로그 파일을 보는 방법”
- 17 페이지 “노드 에이전트를 시작하는 방법”
- 19 페이지 “도메인에 클러스터 지원을 추가하는 방법”
- 19 페이지 “번들로 제공된 Java DB 서버를 시작하는 방법”

▼ Solaris 및 Linux에서 도메인을 시작하는 방법

Domain Administration Server를 시작하려면 기본 도메인 `domain1`을 시작합니다.

- 1 PATH 환경 변수에 `install-dir/bin/` 디렉토리를 추가합니다.

2 다음 명령을 입력하여 서버를 시작합니다.

```
asadmin start-domain domain1
```

사용자, 관리자 비밀번호 및 마스터 비밀번호를 묻는 메시지가 표시되면 설치 시 입력했던 사용자 이름과 비밀번호를 입력합니다.

시작 프로세스가 완료되면 다음 메시지가 표시됩니다.

도메인 domain1이(가) 시작되었습니다.

도메인마다 엔터프라이즈, 클러스터 또는 개발자와 같은 해당 프로필이 있습니다. domain1은 asadminenv.conf 파일에서 AS_ADMIN_PROFILE 변수로 정의된 기본 프로필을 사용하여 만듭니다. 세 가지 프로필에 대한 자세한 내용은 **Sun Java System Application Server 9.1 관리 설명서**의 “사용 프로필”을 참조하십시오. domain1이 개발자 프로필을 사용하여 만들어진 경우 관리 콘솔에 로그인하면 클러스터링 및 노드 에이전트와 같은 기능을 사용할 수 없습니다. 클러스터 및 서버 인스턴스를 사용하려면 create-domain 명령을 사용하여 클러스터 프로필을 사용하여 도메인을 만듭니다. 도메인을 만드는 방법에 대한 자세한 내용을 보려면 asadmin create-domain --help를 입력하거나 **Sun Java System Application Server 9.1 Reference Manual**의 create-domain(1)을 참조하십시오.

▼ Windows에서 도메인을 시작하는 방법

Domain Administration Server를 시작하려면 기본 도메인 domain1을 시작합니다.

- 1 탐색기 창 또는 바탕 화면에서 마우스 오른쪽 버튼으로 내 컴퓨터를 누릅니다.
- 2 등록 정보를 선택하여 시스템 등록 정보 대화 상자를 표시합니다.
- 3 고급 탭을 누릅니다.
- 4 환경 변수를 누릅니다.
- 5 사용자 변수 섹션에서 다음을 수행합니다.
 - PATH 변수가 있는 경우 다음과 같이 *install-dir\bin*이 경로에 있는지 확인합니다:
install-dir\bin; other_entries. PATH 변수에 *install-dir\bin*이 없는 경우에는 추가합니다.
 - PATH 변수가 없는 경우에는 새로 만들기를 누릅니다. 변수 이름에 PATH를 입력합니다. 변수 값에 서버의 bin 디렉토리 경로(*install-dir\bin*)를 입력합니다. 확인을 눌러 변경 사항을 완결합니다.
- 6 새 환경 변수 AS_ADMIN_USER를 추가하고 이를 설치 시 할당할 관리자 이름으로 설정합니다.

- 7 OK를 눌러 변경을 완결하고 열려 있는 나머지 창을 닫습니다.
- 8 프로그램 메뉴에서 관리 서버 시작 옵션을 눌러 Application Server를 시작합니다.
- 9 관리자 비밀번호 및 마스터 비밀번호를 묻는 메시지가 표시되면 설치 시 입력했던 비밀번호를 입력합니다.

서버를 시작한다는 메시지가 창에 표시됩니다.

```
Starting Domain domain1, please wait.
Log redirected to domain_dir\domain1\logs\server.log...
```

시작 프로세스가 완료되면 다른 메시지가 표시됩니다.

```
Domain domain1 started.
Press any key to continue ...
```

도메인마다 엔터프라이즈, 클러스터 또는 개발자와 같은 해당 프로필이 있습니다. domain1은 asadminenv.conf 파일에서 AS_ADMIN_PROFILE 변수로 정의된 기본 프로필을 사용하여 만듭니다. 프로필에 대한 자세한 내용은 **Sun Java System Application Server 9.1 관리 설명서**의 “사용 프로필”을 참조하십시오. domain1이 개발자 프로필을 사용하여 만들어진 경우 관리 콘솔에 로그인하면 클러스터링 및 노드 에이전트와 같은 기능을 사용할 수 없습니다. 클러스터 및 서버 인스턴스를 사용하려면 create-domain 명령을 사용하여 클러스터 프로필을 사용하여 도메인을 만듭니다. 도메인을 만드는 방법에 대한 자세한 내용을 보려면 asadmin create-domain --help를 입력하거나 **Sun Java System Application Server 9.1 Reference Manual**의 create-domain(1)을 참조하십시오.

- 10 아무 키나 눌러 메시지 창을 닫습니다.

▼ 관리 콘솔에 로그온하는 방법

관리 콘솔은 여러 관리 및 구성 작업을 간편하게 수행할 수 있는 브라우저 인터페이스입니다. 일반적으로 다음 작업을 수행하는 데 사용됩니다.

- 응용 프로그램 배포 및 배포 해제
- 응용 프로그램 활성화, 비활성화 및 관리
- 자원 및 기타 서버 설정 구성
- 로그 파일 선택 및 보기

관리 콘솔을 사용에 대한 자세한 내용은 온라인 도움말이나 **Sun Java System Application Server 9.1 관리 설명서**를 참조하십시오.

- 1 브라우저에 다음 URL을 입력합니다.

```
http://localhost:4848
```

브라우저가 서버 이외의 다른 시스템에 있는 경우에는 localhost 변수를 Domain Administration Server가 실행 중인 시스템의 이름으로 바꿉니다.

4848은 관리 콘솔의 기본 포트 번호입니다. 설치 시 포트 번호를 변경하거나 다른 포트 번호를 사용하여 도메인을 만든 경우에는 해당 번호를 사용합니다.

주 - Website Certified by an Unknown Authority와 같은 메시지가 팝업 창에 표시되면 확인을 누릅니다.

이 메시지는 보안 전송 프로토콜을 통해 관리 콘솔을 실행하는 Domain Administration Server에서 사용하는 자체 서명된 인증서를 브라우저에서 인식하지 못하기 때문에 표시되는 것입니다.

- 2 로그인 창이 표시되면 관리자 이름과 비밀번호를 입력합니다.
- 3 로그인을 누릅니다.
기본 사용자 이름은 admin이고 비밀번호는 adminadmin입니다.



왼쪽 창에 제공된 트리에서 관리할 작업을 선택합니다. 오른쪽 창의 일반 작업이라는 머리글 아래에 여러 관리 작업이 나열됩니다.

▼ Domain Administration Server 로그 파일을 보는 방법

Application Server 인스턴스와 Domain Administration Server가 파일 시스템에 주석이 달린 로그를 만듭니다. 기본적으로 모든 오류, 경고 또는 유용한 정보 메시지가 기록됩니다.

- 1 오른쪽 창의 일반 작업 목록에서 로그 파일 검색을 눌러 새 브라우저 창에 로그 뷰어를 시작합니다.



- 2 로그 뷰어 창의 인스턴스 이름 드롭다운 목록에서 "서버"를 선택하고 검색을 누릅니다. Domain Administration Server의 최신 로그 파일 항목이 표시됩니다.
- 3 메시지를 검색하여 서버 시작 시 발생한 문제를 나타내는 WARNING 또는 SEVERE 메시지를 찾습니다.
로그 뷰어를 언제든지 닫을 수 있습니다. 클러스터를 만들고 응용 프로그램을 배포한 후에는 로그 파일을 검사하여 실패한 작업이 있는지 확인합니다. 로그 뷰어를 사용하여 도메인에서 실행 중인 모든 Application Server 인스턴스의 로그 파일을 봅니다.

▼ 노드 에이전트를 시작하는 방법

노드 에이전트는 Application Server 관리 도메인에 참여하는 각 시스템에서 실행되는 경량 프로세스입니다. 노드 인스턴스는 호스트에서 서버 인스턴스를 시작하고 중지하는 일을 담당합니다. 또한 Domain Administration Server와 통신하여 새 Application Server 인스턴스를 만듭니다.

주 - 클러스터 프로필 또는 엔터프라이즈 프로필을 갖는 도메인에서만 노드 에이전트를 사용할 수 있습니다. 개발자 도메인만 실행 중인 경우 노드 에이전트를 만들고 사용하려면 클러스터 프로필을 사용하여 도메인을 만들어야 합니다. 도메인을 만드는 방법에 대한 자세한 내용은 `asadmin create-domain --help`를 입력하거나 **Sun Java System Application Server 9.1 Reference Manual**의 `create-domain(1)`을 참조하십시오.

시스템이 속한 각 Application Server 관리 도메인당 하나의 노드 에이전트가 시스템에 있어야 합니다. Sun Java System Application Server 9.1(Enterprise 번들)을 사용 중인 경우 설치 시 노드 에이전트 구성 요소를 선택하면 `hostname`이라는 기본 노드 에이전트가 생성됩니다. Glassfish를 설치한 경우에는 노드 에이전트가 자동으로 생성되지 않습니다. 노드 에이전트를 만들려면 **Sun Java System Application Server 9.1 고가용성 관리 설명서**의 “노드 에이전트 만들기”에 설명된 지침을 수행합니다.

1 단말기 창에서 다음 명령을 입력합니다.

```
asadmin start-node-agent nodeagentname
```

nodeagentname 변수를 앞서 만든 노드 에이전트의 이름으로 바꿉니다. Sun Java System Application Server 9.1에서 생성한 기본 노드 에이전트를 사용하려면 *nodeagentname*을 Application Server가 실행되고 있는 호스트 이름으로 바꿉니다. 포트 4848의 기본 도메인(`domain1`)이 클러스터 프로필 또는 엔터프라이즈 프로필에서 실행되고 있지 않는 경우에는 `--port` 옵션을 사용하여 클러스터 또는 엔터프라이즈 도메인의 포트를 지정합니다.

2 메시지가 표시되면 마스터 비밀번호를 입력합니다.

노드 에이전트가 시작하고 Domain Administration Server에 연결합니다.

주 - 노드 에이전트를 시작하려면 Domain Administration Server가 실행 중이어야 합니다. 클러스터 인스턴스의 `server.log` 파일을 검사하여 노드 에이전트 또는 클러스터 인스턴스 시작 시 발생한 문제를 확인합니다.

Windows에서 노드 에이전트나 클러스터를 시작하는 동안 문제가 발생한 경우 가능한 문제 해결 방법을 보려면 **Sun Java System Application Server 9.1 릴리스 노트**를 참조하십시오.

노드 에이전트가 시작되면 해당 노드 에이전트를 관리 콘솔에서 볼 수 있습니다.

다음 순서 이 절에서는 Domain Administration Server를 시작하고 실행 중인지 확인했습니다. 또한 관리 콘솔에 로그인하고 로그 뷰어를 사용했으며 노드 에이전트를 시작했습니다. 작업을 계속하지 않으려면 여기서 빠른 시작 과정을 중지할 수 있으며, 다음 과정으로 이동할 수도 있습니다.

▼ 도메인에 클러스터 지원을 추가하는 방법

관리 콘솔을 사용하여 개발자 도메인에 클러스터 지원을 추가할 수 있습니다.

- 1 관리 콘솔에 로그인합니다. 자세한 내용은 [15 페이지 "관리 콘솔에 로그인하는 방법"](#)을 참조하십시오.
- 2 왼쪽 영역에서 **Application Server**를 누릅니다.
- 3 일반 탭에서 클러스터 지원 추가 버튼을 누릅니다.
- 4 클러스터 지원 추가 페이지가 나타납니다. 이 페이지의 정보를 검토하고 확인을 누릅니다.
- 5 다시 시작해야 함 페이지에서 인스턴스 중지 버튼을 눌러 도메인을 중지합니다.
- 6 도메인이 설치된 시스템에 로그인하여 다음 명령을 실행합니다. `install-dir/bin/asadmin list-domains`. 도메인이 중지되었는지 확인합니다.
- 7 다음 명령을 실행합니다. `install-dir/bin/asadmin start-domain <domain-name>`
다시 시작된 도메인은 클러스터링을 지원합니다.

자세한 정보 문제 해결

다시 시작된 도메인에 클러스터링 기능이 표시되지 않으면 도메인을 중지하고 도메인이 설치된 시스템을 재부트한 후 단계 7에 표시된 대로 도메인을 다시 시작합니다.

▼ 번들로 제공된 Java DB 서버를 시작하는 방법

Application Server에는 Java DB 서버가 번들로 포함되어 있습니다. 서버를 시작하려면 다음 절차를 사용합니다.

- 1 데이터베이스를 설치한 사용자로 로그인했는지 확인합니다.
- 2 `asadmin start-database`를 실행합니다.
`start-database` 명령에 대한 자세한 내용은 명령의 도움말 또는 **Sun Java System Application Server 9.1 Reference Manual**에서 `start-database(1)`의 내용을 참조하십시오.

다음 작업

빠른 시작 설명서의 다음 단계는 다음 장을 참조하십시오.

- 2 장

샘플 클러스터를 만드는 절차입니다. 개발자 프로필을 사용하여 도메인을 실행 중인 경우 다음 장으로 건너뛵니다.

- 3 장

샘플 웹 응용 프로그램을 배포하는 절차입니다. 클러스터 기능이 있는 도메인을 사용하는 경우에는 클러스터에 엔터프라이즈 응용 프로그램을 배포하는 절차를 수행할 수 있습니다.

- 4 장

로드 밸런서를 만들고 사용하는 절차입니다. 로드 밸런서 플러그인과 Web Server를 설치해야 합니다.

그 외 Application Server를 이해하고 사용하는 데 도움이 되는 다른 자원도 있습니다. 다음과 같습니다.

- 제품 세부 정보는 *install-dir/docs-ee/about.html*에 있습니다.
- 샘플 응용 프로그램은 *install-dir/samples*에 있습니다.
- **Sun Java System Application Server 9.1 릴리스 노트**
- **Sun Java System Application Server 9.1 관리 설명서**
- **Sun Java System Application Server 9.1 Reference Manual**(설명서 페이지)
- **Sun Java System Application Server 9.1 Reference Manual**(설명서 페이지)
- **Sun Java System Application Server 9.1 Application Deployment Guide**
- **Sun Java System Application Server 9.1 고가용성 관리 설명서**
- **The Java EE 5 Tutorial**
- 엔터프라이즈용 **Java BluePrints** (<http://java.sun.com/blueprints>) 지침

클러스터 설정

이 장에서는 클러스터를 만들고 설정하는 방법에 대해 설명합니다. 클러스터는 같은 구성, 자원 및 응용 프로그램을 공유하는, 일반적으로 여러 호스트에 존재하는 서버 인스턴스 그룹입니다. 클러스터를 사용하면 서버 인스턴스 전체의 로드 균형 조정과 페일오버를 통한고가용성 구현이 용이합니다. 여러 시스템을 연결하는 클러스터를 만들고 각 시스템에서 노드 에이전트 프로세스를 사용하여 관리할 수 있습니다.

이 장에서는 클러스터 프로필 또는 엔터프라이즈 프로필을 사용하여 도메인을 실행 중이라고 가정합니다. 이 장은 다음 내용으로 구성됩니다.

- 21 페이지 “클러스터 만들기”
- 23 페이지 “클러스터링된 서버 인스턴스의 포트 보기”

클러스터 만들기

이 절에서는 두 개의 Application Server 인스턴스가 포함된 클러스터를 만드는 방법에 대해 설명합니다.

▼ 관리 콘솔을 사용하여 클러스터를 만드는 방법

작업상 편의를 위해 클러스터는 완전히 한 시스템 내에서 실행됩니다. 이 절차에서는 DAS가 실행되고 있는 동일한 호스트에서 샘플 클러스터를 만듭니다.

시작하기 전에 17 페이지 “노드 에이전트를 시작하는 방법”의 설명대로 이미 노드 에이전트를 시작한 상태여야 합니다. 클러스터를 만드는 과정에서 인스턴스를 지정할 때, 인스턴스를 실행할 시스템에 대해 실행 중인 노드 에이전트와 인스턴스를 연결합니다. 노드 에이전트가 실행되고 있지 않으면 인스턴스가 시작되지 않습니다. 노드 에이전트 및 인스턴스 이름은 한 도메인에서 생성된 클러스터 전체에서 고유해야 합니다.

- 1 아직 로그인하지 않은 경우 관리 콘솔(<http://localhost:portnumber>)에 로그인합니다.
localhost 변수를 Domain Administration Server가 실행되고 있는 시스템 이름으로 바꿉니다. *portnumber*를 클러스터 또는 엔터프라이즈 도메인의 포트 번호로 바꿉니다.
- 2 오른쪽 영역의 일반 작업에서 새 클러스터 만들기를 눌러 새 클러스터 페이지를 표시합니다.
- 3 새 클러스터 이름으로 `cluster1`을 입력합니다.
- 4 사용 가능한 구성 템플릿의 드롭다운 목록에서 `default-config` 구성을 선택하고 선택한 구성의 복사본을 만듭니다. 를 선택합니다.
- 5 클러스터에 두 개의 인스턴스를 지정하기 위해 새로 만들기 버튼을 두 번 눌러 항목을 두 개 만듭니다.
- 6 인스턴스 이름으로 `instance1` 및 `instance2`를 입력합니다.
또는, 만들어서 시작한 노드 에이전트의 이름을 지정할 수도 있습니다. 노드 에이전트를 만들고 시작하려면 `asadmin create-node-agent(1)` 명령을 사용합니다.
- 7 확인을 누릅니다.
클러스터를 만들었습니다. 페이지가 나타나고 `cluster1`이 왼쪽 영역의 트리에 나타납니다. 이 클러스터에 대해 구성 템플릿 `default-config`의 복사본이 생성되어 `cluster1-config`이라는 이름이 할당되었습니다.
- 8 `cluster1` 클러스터를 시작합니다.

▼ CLI 명령을 사용하여 클러스터를 만드는 방법

- 1 다음 `asadmin` 명령을 실행하여 `cluster1` 클러스터를 만듭니다. `asadmin create-cluster --user admin --passwordfile adminpassword.txt --host localhost -port 4848 cluster1`
- 2 다음 `asadmin` 명령을 실행하여 `cluster1` 클러스터를 만듭니다. `asadmin create-cluster --user admin --passwordfile adminpassword.txt --host localhost -port 4848 cluster1`
- 3 다음 명령을 실행하여 `cluster1-nodeagent`라는 노드 에이전트를 만듭니다. `asadmin create-node-agent --user admin --passwordfile adminpassword.txt --host localhost --port 4848 cluster1-nodeagent`

- 4 다음 명령을 실행하여 이 노드 에이전트를 시작합니다.


```
asadmin start-node-agent
--user admin --passwordfile adminpassword.txt --host localhost --port 4848
cluster1-nodeagent
```
- 5 다음 명령을 실행하여 클러스터에 두 개의 인스턴스를 만듭니다.
 - ```
asadmin create-instance --user admin --passwordfile adminpassword.txt --host
localhost -port 4848 --cluster cluster1 --nodeagent cluster1-nodeagent
--systemproperties
"JMX_SYSTEM_CONNECTOR_PORT=8687:IIOP_LISTENER_PORT=3330:IIOP_SSL_LISTENER
_PORT=4440:IIOP_SSL_
MUTUALAUTH_PORT=5550:HTTP_LISTENER_PORT=1110:HTTP_SSL_LISTENER_PORT=2220"
instance1
```
  - ```
asadmin create-instance --user admin --passwordfile adminpassword.txt --host
localhost -port 4848 --cluster cluster1 --nodeagent cluster1-nodeagent
--systemproperties
"JMX_SYSTEM_CONNECTOR_PORT=8688:IIOP_LISTENER_PORT=3331:IIOP_SSL_LISTENER
_PORT=4441:IIOP_SSL_
MUTUALAUTH_PORT=5551:HTTP_LISTENER_PORT=1111:HTTP_SSL_LISTENER_PORT=2221"
instance2
```
- 6 다음 명령을 실행하여 클러스터를 시작합니다.


```
asadmin start-cluster --user admin
--passwordfile adminpassword.txt --host localhost --port 4848 cluster1
```

 클러스터를 만들고 구성하는 방법에 대한 자세한 내용은 **Sun Java System Application Server 9.1 고가용성 관리 설명서**의 6 장, “Application Server 클러스터 사용”을 참조하십시오.
 노드 에이전트를 만들고 구성하는 방법에 대한 자세한 내용은 **Sun Java System Application Server 9.1 고가용성 관리 설명서**의 8 장, “노드 에이전트 구성”을 참조하십시오.

클러스터링된 서버 인스턴스의 포트 보기

▼ 클러스터링된 서버 인스턴스의 포트를 확인하는 방법

포트 번호를 확인한 후 이전 21 페이지 “클러스터 만들기” 절에서 만든 클러스터링된 서버 인스턴스의 기본값을 대체할 수 있습니다. 기본 HTTP 포트는 instance1의 경우 38081이고 instance2의 경우 38080입니다. 클러스터를 만들 때 시스템에서 이 포트가 사용 중이거나 이미 다른 인스턴스와 클러스터에 이 포트가 할당된 경우에는 다른 포트 번호가 할당됩니다.

포트 번호를 확인한 후 선택적으로 기본값을 대체하려면 다음 단계를 수행합니다.

- 1 왼쪽 영역에서 클러스터를 확장하고 cluster1을 눌러 클러스터에 대한 일반 정보 페이지를 표시합니다.
- 2 인스턴스 탭을 눌러 앞에서 만든 instance1 및 instance2 인스턴스를 표시합니다.
 - a. instance1을 눌러 이 인스턴스를 검토합니다.

인스턴스에 할당된 HTTP 및 HTTPS 포트가 HTTP 포트 필드에 표시됩니다.
 - b. instance2에 대해 이 절차를 반복합니다.

다음순서 이 절에서는 단일 시스템에서 단순 클러스터를 만들었습니다. 소프트웨어가 설치되어 있고 각 시스템에서 노드 에이전트가 실행 중일 경우 이 기본 절차를 사용하여 여러 시스템이 연결된 클러스터를 만들 수도 있습니다.

응용 프로그램 배포

이 장에서는 응용 프로그램 배포 방법에 대해 설명합니다. 개발자 프로필을 사용하여 도메인을 실행 중인 경우에는 25 페이지 “샘플 웹 응용 프로그램 배포”를 참조하십시오. 클러스터 프로필을 사용하여 도메인을 실행 중인 경우에는 28 페이지 “클러스터에 샘플 엔터프라이즈 응용 프로그램 배포”를 참조하십시오.

이 장은 다음 내용으로 구성되어 있습니다.

- 25 페이지 “샘플 웹 응용 프로그램 배포”
- 28 페이지 “클러스터에 샘플 엔터프라이즈 응용 프로그램 배포”
- 29 페이지 “클러스터에서 인스턴스를 시작하는 방법”
- 29 페이지 “응용 프로그램 배포를 확인하는 방법”

샘플 웹 응용 프로그램 배포

다음 절차에서는 개발자 프로필을 사용하여 도메인을 실행 중이라고 가정합니다.

▼ 자동 배포를 이용하여 Hello 응용 프로그램을 배포하는 방법

서버 설치에는 **The Java EE 5 Tutorial**에서 가져온 "Hello"라고 말하는 미리 패키지화된 응용 프로그램이 포함되어 있습니다. Hello 응용 프로그램은 `samples/quickstart` 디렉토리의 WAR(Web ARchive) 파일에 포함되어 있습니다. 패키지화된 응용 프로그램은 JAR(Java ARchive) 또는 EAR(Enterprise ARchive) 파일 형식일 수도 있습니다.

<https://glasfish.dev.java.net/downloads/quickstart/hello.war>에서도 이 응용 프로그램을 다운로드할 수 있습니다.

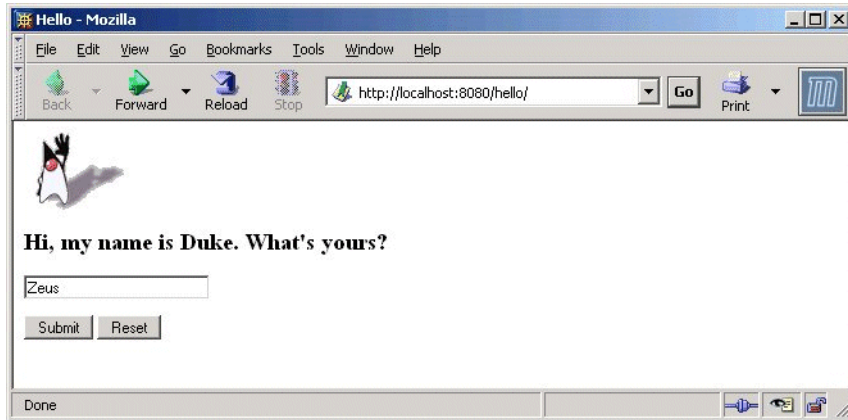
도메인의 `autodeploy` 디렉토리로 복사하면 최소한의 작업으로 미리 패키지화된 응용 프로그램을 즉시 사용할 수 있습니다.

- 1 `install-dir/samples/quickstart/hello.war`를 찾거나 <https://glasfish.dev.java.net/downloads/quickstart/hello.war>에서 다운로드합니다.
- 2 `hello.war`를 `install-dir/domains/domain1/autodeploy/` 디렉토리로 복사합니다.

정보 - Windows의 경우 바탕 화면에 `autodeploy` 디렉토리의 바로 가기를 만든 다음 `hello.war` 파일을 이 바로 가기에 끌어서 놓을 수 있습니다.

서버에서 응용 프로그램 배포가 완료되면 `autodeploy` 디렉토리에 `hello.war_deployed` 파일이 생성됩니다. 시스템 속도에 따라 이 프로세스는 몇 초 정도 소요될 수 있습니다. 이 파일이 나타나기 전에 응용 프로그램 페이지로 이동하려고 하면 404-File Not Found 오류가 발생합니다.

- 3 응용 프로그램 페이지로 이동하려면 다음 URL에 액세스합니다.
`http://localhost:8080/hello`
이름 입력을 요청하는 응용 프로그램의 첫 번째 페이지가 표시됩니다.



- 4 이름을 입력하고 제출을 누릅니다.
사용자에게 맞추어 Hello라고 말하는 사용자 정의 응답이 표시됩니다.

▼ 관리 콘솔을 이용하여 Hello 응용 프로그램을 배포하는 방법

- 1 웹 브라우저에서 `http://localhost:4848/asadmin`을 엽니다.

- 브라우저와 Application Server가 모두 같은 시스템에서 실행 중이면 *localhost*를 사용하고 Application Server가 다른 시스템에서 실행 중이면 URL에서 해당 시스템 이름으로 대체합니다.
 - 4848은 관리 콘솔의 기본 포트 번호입니다. 설치 중에 포트 번호를 변경한 경우 해당 번호를 사용합니다.
- 2 관리자 이름과 비밀번호를 입력합니다.
 - 3 왼쪽 영역에서 응용 프로그램 노드를 눌러 확장합니다.
 - 4 웹 응용 프로그램을 누릅니다.
배포된 웹 응용 프로그램이 있는 경우 목록을 볼 수 있습니다.
 - 5 Hello 응용 프로그램이 이미 배포된 경우에는 옆에 있는 확인란을 선택하고 배포 해제를 눌러 배포를 해제합니다.
 - 6 배포를 누릅니다.
 - 7 Application Server와 함께 번들로 제공되는 Hello 응용 프로그램을 사용하거나 <https://glasfish.dev.java.net/downloads/quickstart/hello.war>에서 다운로드한 샘플을 업로드할 수 있습니다. 다음 중 한 가지를 수행합니다.
 - Application Server에서 액세스할 수 있는 패키지화된 로컬 파일 또는 디렉토리 옵션을 선택하고 파일 찾아보기를 누릅니다. *install-dir/samples/quickstart* 디렉토리로 이동하여 *hello.war*를 선택합니다.
 - 8 Hello 응용 프로그램을 다운로드한 경우 Application Server에 업로드할 패키지 파일 옵션을 선택하고 찾아보기를 누릅니다. 응용 프로그램을 다운로드한 디렉토리로 이동합니다.
 - 9 확인을 누릅니다.
웹 응용 프로그램 목록에 응용 프로그램이 나타납니다.
 - 10 제대로 배포되었는지 확인하려면 웹 응용 프로그램 페이지에서 Hello 응용 프로그램을 선택하고 시작을 누릅니다.

자세한 정보 CLI 명령을 사용하여 응용 프로그램 배포

`asadmin deploy` 명령을 사용하여 응용 프로그램을 배포할 수 있습니다. **Sun Java System Application Server 9.1 Reference Manual**의 `deploy(1)`를 참조하십시오. `asant` 명령을 사용하여 배포할 수도 있습니다. 응용 프로그램 배포 방법에 대한 자세한 내용은 **Sun Java System Application Server 9.1 Application Deployment Guide**의 “Deploying Modules and Applications”를 참조하십시오.

클러스터에 샘플 엔터프라이즈 응용 프로그램 배포

다음 절차에서는 클러스터 프로필을 사용하여 도메인을 실행 중이라고 가정합니다. 이 장에 소개된 절차는 2장의 설명대로 cluster1이라는 클러스터를 이미 만든 것으로 가정합니다. 이 절에서는 clusterjsp 샘플 응용 프로그램을 사용하여 HTTP 로드 균형 조정 기능을 보여줍니다.

▼ 관리 콘솔을 이용하여 샘플 엔터프라이즈 응용 프로그램을 배포하는 방법

- 1 일반 작업 페이지에 있지 않은 경우 홈 버튼을 누르거나 왼쪽 영역에서 일반 작업을 눌러 이동합니다.
- 2 오른쪽 영역의 일반 작업에서 엔터프라이즈 응용 프로그램(.ear) 배포를 누릅니다.
- 3 **Application Server**가 로컬 시스템에 설치되어 있는 경우 업로드할 패키지화된 파일 텍스트 상자에서 찾아보기를 누르고
*install-dir/samples/ee-samples/highavailability/apps/clusterjsp/clusterjsp.ear*로 이동합니다. 다른 시스템에 설치된 **Application Server**에서 원격으로 관리 콘솔에 액세스하고 있는 경우에는 **Application Server**에서 액세스할 수 있는 패키지화된 로컬 파일 또는 디렉토리 텍스트 상자에서 찾아보기를 누릅니다.
*install-dir/samples/quickstart/clusterjsp/clusterjsp.ear*로 이동합니다.
- 4 페이지의 대상 섹션에 있는 사용 가능 목록에서 cluster1을 선택하고 추가를 눌러 선택 목록으로 이동합니다.
- 5 확인을 누릅니다.

이제 clusterjsp 응용 프로그램이 cluster1에 배포되었습니다. 이 응용 프로그램을 시작하기 전에 클러스터에서 인스턴스를 시작해야 합니다. 29 페이지 “클러스터에서 인스턴스를 시작하는 방법”을 참조하십시오.

자세한 정보 CLI 명령을 사용하여 응용 프로그램 배포

asadmin deploy 명령을 사용하여 응용 프로그램을 배포할 수 있습니다. **Sun Java System Application Server 9.1 Reference Manual**의 deploy(1)를 참조하십시오. asant 명령을 사용하여 배포할 수도 있습니다. 응용 프로그램 배포 방법에 대한 자세한 내용은 **Sun Java System Application Server 9.1 Application Deployment Guide**의 “Deploying Modules and Applications”를 참조하십시오.

▼ 클러스터에서 인스턴스를 시작하는 방법

배포된 응용 프로그램을 시작하기 전에 클러스터에서 인스턴스를 시작해야 합니다.

- 1 왼쪽 영역의 트리에서 클러스터를 누르고 cluster1 노드를 누릅니다.
- 2 오른쪽 영역에서 일반 탭으로 이동합니다(아직 활성화되지 않은 경우).
- 3 인스턴스가 아직 실행되고 있지 않으면 인스턴스 시작 버튼이 활성화됩니다. 이 버튼을 눌러 클러스터의 인스턴스를 시작합니다.
- 4 실행 중인 인스턴스 수를 나타내는 상태 필드를 확인하여 인스턴스가 시작되었는지 확인합니다.

응용 프로그램을 시작할 준비가 되었습니다. 29 페이지 “클러스터에서 인스턴스를 시작하는 방법”을 참조하십시오.

자세한 정보 CLI 명령을 사용하여 인스턴스 시작

asadmin start-instance 명령을 사용하여 인스턴스를 시작할 수 있습니다. **Sun Java System Application Server 9.1 Reference Manual**을 참조하십시오.

▼ 응용 프로그램 배포를 확인하는 방법

- 1 브라우저에 다음 URL을 입력합니다.
http://localhost:port/clusterjsp
localhost 변수를 Domain Admin Server가 실행되고 있는 시스템 이름으로 바꿉니다.
port 변수를 instance1의 HTTP-LISTENER-PORT 값으로 바꿉니다. 이 예에서는 http://localhost:38081/clusterjsp를 사용합니다.
- 2 세션 속성 데이터를 추가합니다.
- 3 표시된 세션 및 호스트 정보를 검토합니다. 예를 들면 다음과 같습니다.
 - Executed From Server: localhost
 - Server Port Number: 38081
 - Executed Server IP Address: 198.19.255.255
 - Session Created: Tue Aug 23 15:26:07 PDT 2005
- 4 세션 데이터를 추가하고 세션에 추가 버튼을 누릅니다.
- 5 브라우저에 다음 URL을 입력하여 instance2 인스턴스에 대해 이 절차를 반복합니다.
http://localhost:38080/clusterjsp

다음 순서 축하합니다. **빠른 시작 설명서**에서 이 절을 마쳤습니다. 로드 균형 조정에 대한 단계를 진행하지 않으려는 경우 여기서 빠른 시작 과정을 중지할 수 있습니다.

로드 밸런서 작업

이 절에서는 Application Server 클러스터의 로드 밸런서 역할을 하도록 Web Server 소프트웨어를 설정하는 방법에 대한 지침을 제공합니다. 또한 로드 밸런서를 구성하여 Web Server로 내보내는 절차도 제공합니다. 엔터프라이즈 프로파일이나 클러스터 프로 파일을 사용하는 도메인을 실행 중인 경우에만 로드 밸런서 기능을 사용할 수 있습니다.

로드 밸런서는 클러스터에 배포됩니다. 로드 밸런서는 다음과 같은 기능을 제공합니다.

- 여러 실제(또는 논리) 호스트에 응용 프로그램이나 서비스를 수평으로 확장할 수 있게 하면서 사용자에게는 단일 URL을 제공합니다.
- 세션 지속성과 함께 사용할 경우 호스트 오류나 서버 충돌로부터 사용자를 격리합니다.
- 사용자에게 내부 네트워크가 표시되지 않도록 하여 보안을 강화합니다.

Application Server에는 Sun Java™ System Web Server, Apache 및 Microsoft Windows IIS와 같은 널리 사용되는 웹 서버용 로드 균형 조정 플러그인이 포함되어 있습니다.

이 절을 완료하기 위해서는 Domain Administration Server, 그리고 지금까지 이 설명서에서 만든 두 개의 Application Server 인스턴스와 함께 Web Server를 실행할 만큼 충분한 메모리가 시스템에 있어야 합니다. 512 ~ 1024MB 메모리의 시스템이 권장됩니다.

이 항목은 다음 내용으로 구성됩니다.

- 32 페이지 “로드 균형 조정 설정”
- 33 페이지 “로드 균형 조정 시작”
- 33 페이지 “로드 균형 조정 확인”
- 34 페이지 “메모리 내 복제 기능을 이용한 고가용성 및 페일오버”

로드 균형 조정 설정

로드 균형 조정을 설정하기 전에 로드 밸런서 플러그인을 설치해야 합니다. 로드 균형 조정 플러그인을 설치하는 방법에 대한 자세한 내용은 **Sun Java System Application Server 9.1 Installation Guide**의 “To Install the Load Balancing Plug-in (standalone)”을 참조하십시오. 이 절차에서는 클러스터 프로필 또는 엔터프라이즈 프로필을 사용하여 도메인을 실행 중이라고 가정합니다.

▼ 로드 균형 조정을 설정하는 방법

- 1 관리 콘솔을 사용하여 로드 밸런서를 만듭니다. `asadmin create-http-lb(1)` 명령을 사용하여 만들 수도 있습니다.
 - a. 관리 콘솔에서 HTTP 로드 밸런서 노드를 누릅니다.
 - b. 새로 만들기를 누릅니다.
 - c. 로드 밸런서 이름을 `lb1`로 입력하고 Web Server가 설치된 호스트 및 Web Server 인스턴스 포트를 입력합니다. 이 샘플 시나리오에서 Web Server 호스트는 `localhost`이고 포트는 `38000`입니다.
 - d. 변경 사항 자동 적용 확인란을 선택합니다. 이 옵션을 선택하면 로드 밸런서 구성을 내보낼 필요가 없습니다. 로드 밸런서 구성에서 변경한 모든 내용이 자동으로 전파됩니다.
 - e. 대상으로 `cluster1`을 선택합니다.
21 페이지 “클러스터 만들기”에서 샘플 클러스터(`cluster1`)를 만드는 방법에 대해 설명합니다.
 - f. 저장을 누릅니다.
- 2 `cluster1`에서 로드 균형 조정을 사용하도록 설정합니다.
`asadmin enable-http-lb-server cluster1`
- 3 `clusterjsp` 응용 프로그램에서 로드 균형 조정을 사용하도록 설정합니다.
`asadmin enable-http-lb-application clusterjsp`

참조 로드 밸런서 구성 변경 또는 상태 검사기 만들기와 같은 고급 항목에 대한 자세한 내용은 **Sun Java System Application Server 9.1 고가용성 관리 설명서**의 5장, “HTTP 로드 균형 조정 구성”을 참조하십시오.

로드 균형 조정 시작

Web Server를 시작하거나 다시 시작하여 로드 균형 조정을 시작합니다.

- 로드 밸런서로 사용되는 Web Server 인스턴스가 아직 실행되고 있지 않으면 Web Server를 시작합니다.

Web Server 7.0을 사용 중인 경우에는 `wadm start-instance` 명령을 사용합니다.

Web Server 6.1의 경우 `<websvr-instance-dir>` 디렉토리의 `start` 스크립트를 실행합니다.
- 로드 밸런서로 사용되는 Web Server 인스턴스가 이미 실행 중인 경우에는 Web Server를 중지했다가 다시 시작합니다.

Web Server 6.1의 경우 `web_server_install_dir/https-hostname`의 중지 프로그램을 사용한 후 시작 프로그램을 실행하여 서버를 다시 시작합니다.

Web Server 7.0의 경우 `wadm stop-instance` 명령 다음에 `wadm start-instance` 명령을 사용합니다.

로드 균형 조정 확인

응용 프로그램이 배포된 후 로드 밸런서가 실행되면 로드 균형 조정이 작동하는지 확인합니다.

▼ 로드 균형 조정을 확인하는 방법

- 1 clusterjsp 응용 프로그램의 첫 페이지를 표시하기 위해 브라우저에 다음 URL을 입력합니다.

`http://localhost:web_server_port /clusterjsp`

`localhost` 변수를 Web Server가 실행되고 있는 시스템 이름으로 바꿉니다.

`web_server_port` 변수를 `web_server_install_dir/https-hostname/config/server.xml`에 있는 LS 요소의 포트 속성 값으로 바꿉니다. 이 예에서는 포트 38000이 사용됩니다.

29 페이지 “응용 프로그램 배포를 확인하는 방법”에 표시된 것과 비슷한 페이지가 나타납니다.

- 2 표시된 세션 및 호스트 정보를 검토합니다. 예를 들면 다음과 같습니다.

- Executed From Server: `localhost`
- Server Port Number: `38000`
- Executed Server IP Address: `192.18.145.133`
- Session Created: Day Mon 05 14:55:34 PDT 2005

- 3 서버 포트 번호는 Web Server 포트인 38000입니다. 로드 밸런서가 클러스터의 두 인스턴스에 대한 요청을 전송했습니다.
- 4 다른 브라우저 소프트웨어나 다른 시스템의 브라우저를 사용하여 새 세션을 만듭니다. 동일한 브라우저의 요청은 "고정"되어 동일한 인스턴스로 이동합니다.
이 세션은 클러스터의 두 인스턴스에 배포되어야 합니다. 다음 위치에 있는 서버 액세스 로그 파일을 찾아 이 내용을 확인할 수 있습니다.
 - Solaris Java Enterprise System 설치:

```
/var/opt/SUNWappserver/nodeagents/nodeagent_name /instance1/logs  
/access/server_access_log  
  
/var/opt/SUNWappserver/nodeagents/nodeagent_name /instance2/logs  
/access/server_access_log
```
 - Linux Java Enterprise System 설치:

```
/var/opt/sun/appserver/nodeagents/ nodeagent_name/instance1/logs  
/access/server_access_log  
  
/var/opt/sun/appserver/nodeagents/ nodeagent_name/instance2/logs  
/access/server_access_log
```
 - Windows Java Enterprise System 설치:

```
install-dir \nodeagents\nodeagent_name \instance1\logs\access\server_access_log  
install-dir\nodeagents\nodeagent_name \instance1\logs\access\server_access_log
```
 - 독립 실행형 Application Server 설치:

```
install-dir /nodeagents/nodeagent_name/instance1/logs/access/server_access_log  
install-dir/nodeagents/nodeagent_name /instance2/logs/access/server_access_log
```
- 5 HttpSession에 저장하기 위해 이름 및 값 쌍(Name=Value=Duke)을 추가합니다.
- 6 세션 데이터에 추가 버튼을 누릅니다.
- 7 세션 데이터가 추가되었는지 확인합니다.

메모리 내 복제 기능을 이용한 고가용성 및 페일오버

GlassFish v2는 HADB를 제공하지 않습니다. GlassFish에서는 고가용성 및 페일오버를 위해 메모리 내 복제 기능을 제공합니다. 다음 절차에서는 이 기능에 대해 설명합니다.

1. 응용 프로그램을 배포하기 전에 로드 밸런서 플러그인이 설치된 웹 서버를 다시 시작합니다. 그러면 loadbalancer.xml 파일에 설정된 순서대로 인스턴스가 요청을 처리합니다. 이 장에 제공된 loadbalancer.xml 파일을 사용할 경우 instance1이 첫 번째 요청에 사용됩니다.

2. 세션 데이터를 저장하는 `clusterjsp` 웹 응용 프로그램은 이미 앞에서 배포했습니다. 첫 번째 요청을 처리한 인스턴스가 후속 요청도 처리하고 요청 전체에서 세션 데이터가 유지되는 것을 확인할 수 있어야 합니다.
3. 요청을 몇 개 전송한 다음 해당 요청을 처리하는 인스턴스를 기록한 후 해당 인스턴스를 종료합니다. 다음 명령을 사용하여 인스턴스를 중지합니다. `asadmin stop-instance --user adminuser --password adminpassword instance1`
4. 다음 요청을 전송하여 새 데이터가 저장되고 이전에 추가한 데이터가 계속 세션에 남아있는지 확인합니다. 요청을 처리하는 서버 중 하나가 사용할 수 없는 상태일 경우 동일한 클러스터에 속한 다른 서버에서 모든 이전 세션 데이터와 함께 요청을 인계 받아 요청을 완료합니다.

고가용성 페일오버 설정

이전 장에서 사용된 구성을 사용하면 서버 인스턴스가 다운될 경우 세션 상태가 손실될 수 있습니다. 두 고급 항목 중 두 번째인 이 절에서는 고가용성 데이터베이스(HADB) 설치, 고가용성 클러스터 만들기 및 HTTP 세션 지속성 테스트를 위한 절차를 제공합니다.

GlassFish v2는 HADB를 제공하지 않습니다. GlassFish에서는 고가용성과 페일오버를 위해 메모리 내 복제 기능을 제공합니다.

Application Server에서는 HTTP 세션 지속성과 Stateful Session Bean의 지속성을 모두 지원합니다. 이 장의 절차에서는 메모리 내 복제 또는 HADB를 이용한 고가용성에 대해 설명합니다.

이러한 절차는 이 빠른 시작의 앞부분에 설명된 단계를 이미 수행한 것으로 가정합니다. 각 단계는 마쳐야 하는 순서순으로 표시되어 있습니다. HADB 기능을 사용하려면 엔터프라이즈 프로필을 사용하여 도메인을 실행 중이어야 합니다.

주 - 이 절의 절차를 마치려면 하드웨어 자원이 추가로 필요할 수 있습니다.

이 항목은 다음 내용으로 구성됩니다.

- 38 페이지 “고가용성 클러스터와 HADB”
- 38 페이지 “HADB 사전 설치 절차”
- 40 페이지 “HADB 설치”
- 41 페이지 “HADB 시작”
- 42 페이지 “고가용성을 위한 클러스터 및 응용 프로그램 구성”
- 42 페이지 “클러스터 다시 시작”
- 43 페이지 “HTTP 세션 페일오버 확인”

고가용성 클러스터와 HADB

Sun Java System Application Server의 고가용성 클러스터는 상태 복제 서비스를 앞서 만든 클러스터 및 로드 밸런서와 통합하여 HTTP 세션의 페일오버를 가능하게 합니다.

HttpSession 객체와 Stateful Session Bean 상태는 세션 상태를 저장하기 위한 고가용성 데이터베이스인 HADB에 저장됩니다. 수평 확장이 가능한 이 상태 관리 서비스는 Application Server 계층과 별도로 관리할 수 있으며 로드 균형 조정, 페일오버 및 상태 복구 기능을 통해 최고 99.999%의 서비스 및 데이터 가용성을 지원하도록 설계되었습니다.

Application Server와 별도로 상태 관리를 수행하면 큰 이점을 얻을 수 있습니다. Application Server 인스턴스는 외부 고가용성 상태 서비스에 상태 복제를 위임하는 확장 가능하고 성능이 뛰어난 Java™ Platform, Enterprise Edition 5(Java EE™ 5 플랫폼) 컨테이너의 역할을 수행합니다. 이와 같이 구조가 느슨하게 결합되어 있으므로 Application Server 인스턴스를 클러스터에서 쉽게 추가 또는 삭제할 수 있습니다. HADB 상태 보존형 복제 서비스는 최적의 가용성 및 성능을 위해 독립적으로 확장될 수 있습니다. 또한 Application Server 인스턴스가 복제를 수행하면 J2EE 응용 프로그램의 성능이 저하될 수 있으며 가비지 모음이 더 오랫동안 중단될 수 있습니다.

각 HADB 노드에는 512MB의 메모리가 필요하므로 같은 시스템에서 두 개의 HADB 노드를 실행하려면 1GB의 메모리가 있어야 합니다. 메모리가 부족하면 노드를 각각 다른 시스템에 설치합니다. 내결함성이 없으므로 하나의 호스트에서 2노드 데이터베이스를 실행하여 배포하는 것은 권장하지 않습니다.

HADB 사전 설치 절차

이 절차에서는 가장 일반적인 사전 설치 작업을 설명합니다. HADB 설치, 네트워크 중복 구성 및 파일 시스템 지원을 위한 필수 항목 등 다른 사전 설치 항목에 대한 자세한 내용은 **Sun Java System Application Server 9.1 고가용성 관리 설명서**의 2장, “고가용성 데이터베이스 설치 및 설정”을 참조하십시오.

이 절의 권장 시스템 구성 값은 최대 6개의 HADB 노드를 실행하는 데 충분한 값으로, 공유 메모리를 함께 사용하는 시스템의 다른 응용 프로그램은 고려하지 않았습니다.

▼ HADB를 위한 시스템을 구성하는 방법

1 루트로 액세스합니다.

2 공유 메모리 및 세마포와 관련된 변수를 정의합니다.

■ Solaris의 경우:

a. /etc/system 파일에 다음 줄을 추가합니다. 다음 줄이 파일에 주석으로 포함되어 있는 경우에는 주석 처리를 제거하고 값이 일치하는지 확인합니다.

```
set shmsys:shminfo_shmmax=0x80000000
```

```
set shmsys:shminfo_shmseg=36
```

```
set semsys:seminfo_semmnu=600
```

shminfo_shmmax를 시스템의 전체 메모리로 설정합니다(16진수로 표시된 0x80000000 값은 2GB의 메모리를 나타냄).

이미 seminfo_* 변수가 정의되어 있는 경우에는 표시된 양만큼 충분합니다. seminfo_semmni 및 seminfo_semmns의 기본값은 변경할 필요가 없습니다. shminfo_shmseg 변수는 Solaris 8 이후에 더 이상 사용되지 않습니다.

b. 다음 명령을 사용하여 재부트합니다.

```
sync; sync; reboot
```

■ Linux의 경우:

a. /etc/sysctl.conf 파일에 다음 줄을 추가합니다. 다음 줄이 파일에 주석으로 포함되어 있는 경우에는 주석 처리를 제거합니다. 값을 시스템의 실제 메모리 크기로 설정합니다. 값을 십진 바이트 값으로 지정합니다. 예를 들어 실제 메모리가 2GB인 시스템의 경우에는 다음과 같습니다.

```
echo 2147483648 > /proc/sys/shmmax
```

```
echo 2147483648 > /proc/sys/shmall
```

b. 다음 명령을 사용하여 재부트합니다.

```
sync; sync; reboot
```

■ Windows의 경우: 특별한 시스템 설정이 필요하지 않습니다.

- 3 독립 실행형 **Application Server**를 설치할 때 기존 JDK 소프트웨어를 사용한 경우에는 JDK 버전을 확인합니다.

HADB를 사용하려면 Sun JDK 1.4.1_03 이상이 필요합니다(JDK 버전에 대한 최신 정보는 **Sun Java System Application Server 9.1 릴리스 노트** 참조). 설치된 버전을 확인하고 아직 설정하지 않은 경우 JAVA_HOME 환경 변수를 JDK가 설치된 디렉토리로 설정합니다.

- 4 필요한 경우 재부트한 후 도메인, **Web Server** 및 **노드 에이전트**를 다시 시작합니다.

도메인을 다시 시작하려면 `asadmin start-domain domain1` 명령을 사용합니다.

Web Server를 다시 시작하려면 `web_server_install_dir/https-hostname`에서 시작 프로그램을 실행합니다.

노드 에이전트를 다시 시작하려면 `asadmin start-node-agent hostname` 명령을 사용합니다. `hostname` 변수를 Application Server가 실행 중인 호스트의 이름으로 바꿉니다.

HADB 설치

이 절에서는 고가용성 데이터베이스(HADB)를 설치하는 절차를 제공합니다.

주 - Application Server 시스템에서 고가용성 데이터베이스를 실행하려고 하거나 Application Server를 설치할 때 HADB를 설치한 경우 [41 페이지 “HADB 시작”](#)으로 건너뛴니다.

2GB의 메모리와 1~2개의 CPU가 있으면 Application Server 시스템과 동일한 시스템에 HADB 구성 요소를 설치할 수 있습니다. 그렇지 않은 경우 추가 하드웨어를 사용합니다. 예를 들면 다음과 같습니다.

- 각각 512MB ~ 1GB의 메모리가 있는 두 대의 1CPU 시스템
- 1 ~ 2GB의 메모리가 있는 한 대의 1CPU 또는 2CPU 시스템

▼ HADB를 설치하는 방법

- 1 Application Server 또는 Java Enterprise System 설치 프로그램을 실행합니다.
- 2 HADB를 설치할 옵션을 선택합니다.
- 3 호스트에서 설치를 완료합니다.

HADB 시작

이 절에서는 대부분의 경우에 `ma-initd` 스크립트를 실행하여 HADB 관리 에이전트를 시작하는 방법에 대해 설명합니다. 프로덕션 배포의 경우 가용성 확보를 위해 관리 에이전트를 서비스로 시작합니다. 자세한 내용은 **Sun Java System Application Server 9.1 고가용성 관리 설명서**의 “HADB 관리 에이전트 시작”을 참조하십시오.

여러 호스트에 있는 HADB 노드를 사용하여 데이터베이스를 시작하는 경우에는 각 호스트에서 관리 에이전트를 시작합니다.

▼ Solaris 또는 Linux의 Java Enterprise System 설치에서 HADB를 시작하는 방법

- 1 `/etc/init.d` 디렉토리로 이동합니다.
`cd /etc/init.d`
- 2 다음 명령을 실행하여 에이전트를 시작합니다.
`./ma-initd start`

▼ Windows의 Java Enterprise System 설치에서 HADB를 시작하는 방법

Sun Java System이 구성되어 실행 중이면 HADB가 기본적으로 시작됩니다. 그러나 수동으로 시작해야 할 경우에는 다음 절차를 수행합니다.

- 1 시작 ⇒ 설정 ⇒ 제어판으로 이동하여 관리 도구를 두 번 누릅니다.
- 2 서비스 바로 가기를 두 번 누릅니다.
- 3 서비스 목록에서 `HADBManagementAgent` 서비스를 선택합니다.
- 4 동작 메뉴에서 시작을 선택합니다.

▼ Solaris 또는 Linux의 독립 실행형 설치에서 HADB를 시작하는 방법

- 1 Application Server 설치에서 HADB bin 디렉토리로 이동합니다: `install-dir/hadb/4/bin`
- 2 다음 명령을 실행하여 에이전트를 시작합니다.
`./ma-initd start`

▼ Windows의 독립 실행형 설치에서 HADB를 시작하는 방법

- 1 단말기 창에서 Application Server 설치의 HADB bin 디렉토리로 이동합니다:
`install-dir\hadb\4.x\bin`
`x`는 HADB의 릴리스 번호를 나타냅니다.
- 2 다음 명령을 실행하여 에이전트를 시작합니다.
`ma -i ma.cfg`

고가용성을 위한 클러스터 및 응용 프로그램 구성

HTTP 세션 지속성을 확인하기 위해서는 FirstCluster 클러스터에서 HADB를 사용하도록 구성해야 하고 clusterjsp 응용 프로그램에 대해 고가용성을 활성화해야 합니다. `asadmin configure-ha-cluster` 명령을 사용하여 기존 클러스터에서 고가용성을 사용하도록 구성합니다. 이 명령을 사용하는 방법에 대한 자세한 내용은 `asadmin` 명령 프롬프트에서 `configure-ha-cluster --help`를 입력하거나 `configure-ha-cluster(1)` 설명서 페이지를 참조하십시오.

클러스터 다시 시작

이전 절의 변경 사항을 적용하기 위해 클러스터 인스턴스를 다시 시작해야 합니다.

▼ 클러스터를 다시 시작하는 방법

- 1 관리 콘솔 트리에서 클러스터 노드를 확장합니다.
- 2 FirstCluster를 누릅니다.

- 3 오른쪽 영역에서 인스턴스 중지를 누릅니다.
- 4 인스턴스가 중지되면 인스턴스 시작을 누릅니다.

HTTP 세션 페일오버 확인

세션 데이터 페일오버를 테스트하는 절차는 33 페이지 “로드 균형 조정 확인” 항목에 설명된 로드 균형 조정 테스트 절차와 비슷합니다. 오류가 발생한 후에 해당 시간 세션 데이터가 보존됩니다. 샘플 응용 프로그램은 오류가 발생한 후 자동으로 재시도하도록 구성되어 있으므로 페일오버가 사용자에게 투명합니다.

▼ HTTP 세션 페일오버를 확인하는 방법

- 1 **clusterjsp** 응용 프로그램의 첫 페이지를 표시하기 위해 브라우저에 다음 URL을 입력합니다.

`http://localhost :web_server_port /clusterjsp`

`localhost` 변수를 Web Server가 실행되고 있는 시스템 이름으로 바꿉니다.

`web_server_port` 변수를 `web_server_install_dir/https-hostname/config/server.xml`에 있는 LS 요소의 포트 속성 값으로 바꿉니다. 이 예에서는 포트 38000이 사용됩니다.

29 페이지 “응용 프로그램 배포를 확인하는 방법”에 표시된 것과 비슷한 페이지가 나타납니다.

- 2 표시된 세션 및 호스트 정보를 검토합니다. 예를 들면 다음과 같습니다.
 - Executed From Server: *localhost*
 - Server Port Number: 38000
 - Executed Server IP Address: 192.18.145.133
 - Session ID: 41880f618e4593e14fb5d0ac434b1
 - Session Created: Wed Feb 23 15:23:18 PST 2005
- 3 서버 액세스 로그 파일에서 응용 프로그램을 처리하는 응용 프로그램 서버 인스턴스를 확인합니다. 로그 파일은 다음 위치에 있습니다.
 - Solaris Java Enterprise System 설치:


```
/var/opt/SUNWappserver/nodeagents/nodeagent_name
/i1/logs/access/server_access_log

/var/opt/SUNWappserver/nodeagents/nodeagent_name
/i2/logs/access/server_access_log
```
 - Linux Java Enterprise System 설치:

```
/var/opt/sun/appserver/nodeagents/  
nodeagent_name/i1/logs/access/server_access_log
```

```
/var/opt/sun/appserver/nodeagents/  
nodeagent_name/i2/logs/access/server_access_log
```

- Windows Java Enterprise System 설치:

```
install-dir \nodeagents\nodeagent_name\i1\logs\access\server_access_log
```

```
install-dir\nodeagents\nodeagent_name\i2\logs\access\server_access_log
```

- 독립 실행형 Application Server 설치:

```
install-dir /nodeagents/nodeagent_name/i1/logs/access/server_access_log
```

```
install-dir/nodeagents/nodeagent_name/i2/logs/access/server_access_log
```

4 페이지를 처리하고 있는 Application Server 인스턴스를 중지합니다.

- a. 관리 콘솔의 왼쪽 영역에서 클러스터를 확장합니다.
- b. FirstCluster를 누릅니다.
- c. 오른쪽 영역에서 인스턴스 탭을 누릅니다.
- d. 요청을 처리한 서버 인스턴스 옆에 있는 확인란을 누르고 중지 버튼을 누릅니다.

5 clusterjsp 샘플 응용 프로그램 페이지를 다시 로드합니다.

세션 ID와 세션 속성 데이터가 보존됩니다.

6 다른 Application Server 인스턴스의 액세스 로그를 확인해 보면 해당 인스턴스가 요청을 처리하고 있음을 알 수 있습니다.

HTTP 세션이 HADB에 영구적으로 저장되기 때문에 상태 페일오버 기능이 작동합니다. Application Server에서는 HTTP 세션 상태뿐만 아니라 HADB에 EJB 상태를 저장할 수도 있습니다.

다음 순서 축하합니다. Application Server 빠른 시작을 마쳤습니다.

이 절에서는 HADB를 설치, 구성 및 시작했고 고가용성을 사용하도록 클러스터와 응용 프로그램을 구성했습니다. Application Server에 대한 자세한 내용은 [20 페이지 “다음 작업”](#)을 참조하십시오.