



# Service Registry 3 2005Q4 관리 설명서

---

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

부품 번호: 819-3575  
2005년 10월

Copyright 2005 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 없이는 어떠한 형태나 수단으로도 재생산할 수 없습니다. 끝말 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 Java, NetBeans, 및 Solaris는 미국 및 다른 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다. 모든 SPARC 상표는 사용 허가를 받았으며 미국 및 다른 국가에서 SPARC International, Inc.의 상표 또는 등록 상표입니다. SPARC 상표를 사용하는 제품은 Sun Microsystems, Inc.가 개발한 구조를 기반으로 하고 있습니다.

Sun Microsystems, Inc.는 사용자 및 사용 허가자를 위해 OPEN LOOK 및 Sun™ GUI(그래픽 사용자 인터페이스)를 개발했습니다. Sun은 컴퓨터 업계를 위한 시각적 그래픽 사용자 인터페이스의 개념을 연구 개발한 Xerox사의 선구적인 노력을 높이 평가하고 있습니다. Sun은 Xerox 그래픽 사용자 인터페이스에 대해 비 독점적 사용권을 부여 받았으며, 이 사용권은 Sun으로부터 사용권을 부여 받아 OPEN LOOK GUI를 구현하는 이들과 SUN의 서면 동의로 사용권을 부여 받은 이들에게도 적용됩니다.

미국 정부 권한 - 상용 소프트웨어. 정부 기관 사용자는 Sun Microsystems, Inc.의 표준 사용권 조항과 FAR 및 그 부록의 해당 규정을 준수해야 합니다.

이 문서에서는 본문의 내용을 “있는 그대로” 제공하며, 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.



051214@13215



# 목차

---

## 머리말 7

- 1 Service Registry 구성 및 설정 15
  - Service Registry 구성 15
    - ▼ 사용자 정의 등록 정보를 사용하여 나중에 구성 설치 후에 Service Registry를 구성하는 방법 17
    - ▼ 관리 도구 사용 활성화 방법 18
  - Service Registry에 대한 Application Server 도메인 관리 18
    - ▼ Application Server 관리 콘솔 사용 방법 19
    - ▼ 레지스트리에 대해 Application Server 도메인 중단 및 다시 시작 20
  - ▼ 레지스트리 도메인의 신뢰할 수 있는 인증서에 루트 인증서 추가 20
  - 기본값이 아닌 Application Server 설치에 대한 Service Registry 구성 21
    - ▼ install.properties 파일 편집 방법 21
  - 기본값이 아닌 Service Registry 설치에 대한 Service Registry 구성 22
    - ▼ install.properties 파일 편집 방법 22
  - 외부 웹 사이트에 대한 액세스 허용 22
    - ▼ 웹 프록시 지정 방법 23
  - 관리자 만들기 23
    - ▼ 관리자를 만드는 방법 23
  - 웹 콘솔 구성 25
    - 미리 정의된 쿼리 추가 25
      - ▼ 미리 정의된 쿼리를 추가하는 방법 25
    - 기본 쿼리 변경 27
      - ▼ 기본 쿼리를 변경하는 방법 27
    - 분류 체계 숨기기 27
      - ▼ 분류 체계 숨기기 방법 28

검색 결과 표시 구성	28
▼ 검색 결과 표시에 행 수 구성	28
▼ 검색 결과 영역의 열 구성	29
Service Registry 다시 설치	30
▼ 레지스트리에 대한 Application Server 도메인 중지 및 삭제	30
▼ Service Registry 데이터베이스 다시 설치	31
데이터베이스 백업 및 복원	31
▼ Apache Derby 설명서를 찾는 방법	31
<b>2 관리 도구 사용</b>	<b>33</b>
관리 도구 정보	33
관리 도구 시작	34
일괄 처리 모드	34
대화형 모드	34
관리 도구 명령줄 옵션	35
관리 도구 기능	36
권한	37
예외 표시	37
레지스트리 객체 식별	37
이름 지정에 대한 로깅 효과	38
대소문자 구분	38
관리 도구 명령 사용	38
add association	39
add user	40
cd	46
chown	47
cp	47
echo	49
help	49
lcd	50
ls	51
pwd	52
quit	52
rm	53
select	54
set	55
show	56
users	57





# 머리말

---

**Service Registry 3 2005Q4 관리 설명서**는 설치 후 Service Registry (“레지스트리”)를 구성하는 방법 및 레지스트리와 함께 제공되는 관리 도구를 사용하는 방법에 대해 설명합니다. 또한 레지스트리 데이터베이스 백업 및 복원과 같은 기타 관리 작업도 설명합니다.

---

## 이 설명서의 대상

**관리 설명서**는 레지스트리를 설치, 설치 제거 및 관리해야 하는 작업자와 웹 콘솔을 사용하지 않고 레지스트리에 대한 내용을 대량으로 만들기를 원하는 작업자를 대상으로 합니다.

작업자는 사용 중인 운영 체제(Solaris™ 운영 체제 또는 Linux)에 대한 UNIX® 명령 쉘 환경의 기본 개념을 잘 알고 있어야 합니다.

---

## 이 설명서를 읽기 전에

이 설명서를 읽기 전에 **UNIX용 Sun Java Enterprise System 2005Q4 설치 설명서**에 설명된 대로 레지스트리를 설치해야 합니다.

Service Registry는 네트워크나 인터넷 환경에서 배포된 엔터프라이즈 응용 프로그램을 지원하는 소프트웨어 인프라이며, Java Web Services Developer Pack(<http://java.sun.com/webservices/jwsdp/>)의 일부로서 또는 Sun Java Enterprise System의 구성 요소로서 사용할 수 있습니다. Java Enterprise System의 구성 요소로 Service Registry를 구입한 경우 <http://docs.sun.com/co11/1286.1> 및 <http://docs.sun.com/co11/1397.1>에서 시스템 설명서를 숙지해야 합니다.

일부 관리 작업에는 다음 사양의 기본 개념에 대한 숙지가 필요합니다.

- **ebXML Registry Information Model Version 3.0**
- **ebXML Registry Services and Protocols Version 3.0**

OASIS 웹 사이트 (<http://www.oasis-open.org/>)를 방문하여 ebXML RIM V3.0 및 ebXML RS V3.0에 대한 링크로 이동하면 이 사양에 대한 최신 공개 버전을 찾을 수 있습니다.

---

## 이 설명서의 구성

이 설명서의 내용은 다음과 같습니다.

1 장에서는 설치 후 Service Registry를 구성하는 방법 및 기타 관리 작업을 수행하는 방법에 대해 설명합니다.

2 장에서는 관리 도구 사용에 대해 설명합니다.

---

## Service Registry 설명서 세트

Service Registry 설명서 세트는 <http://docs.sun.com/app/docs/coll/1314.1> 및 <http://docs.sun.com/app/docs/coll/1398.1>에 있습니다. Service Registry에 대한 자세한 내용은 다음 표에 나열된 설명서를 참조하십시오.

표 P-1 Service Registry 설명서

문서 제목	내용
Service Registry 3 2005Q4 릴리스 노트	알려진 문제를 포함하여 Service Registry에 대한 최신 정보를 설명합니다.
Service Registry 3 2005Q4 관리 설명서	설치 후 Service Registry 구성 방법 및 레지스트리와 함께 제공되는 관리 도구의 사용 방법에 대해 설명합니다. 또한 기타 관리 작업을 수행하는 방법에 대해서도 설명합니다.
Service Registry 3 2005Q4 사용자 설명서	Service Registry를 검색하고 이 Service Registry에 데이터를 게시하기 위해 Service Registry 웹 콘솔을 사용하는 방법에 대해 설명합니다.

표 P-1 Service Registry 설명서 (계속)

문서 제목	내용
Service Registry 3 2005Q4 Developer's Guide	Service Registry를 검색하고 이 Service Registry에 데이터를 게시하기 위해 JAXR(Java API for XML Registries)을 사용하는 방법에 대해 설명합니다.

## 관련 설명서

Service Registry를 설치하면 Sun Java System Application Server에 배포됩니다. Application Server 관리에 대한 자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.1 2005Q2 관리 설명서**를 참조하십시오.

Java ES 설명서 세트에서는 배포 계획 및 시스템 설치를 설명합니다. 시스템 설명서의 URL은 <http://docs.sun.com/coll/1286.1> 및 <http://docs.sun.com/coll/1397.1>입니다. Java ES에 대한 소개는 다음 표에 나열된 설명서를 순서대로 참조하십시오.

표 P-2 Java Enterprise System 설명서

문서 제목	내용
Sun Java Enterprise System 2005Q4 릴리스 노트	알려진 문제를 포함하여 Java ES에 대한 최신 정보를 설명합니다. 또한, 구성 요소마다 릴리스 노트가 있습니다.
Sun Java Enterprise System 2005Q4 설명서 로드맵	시스템과 개별 구성 요소 모두에 대하여 Java ES와 연관된 모든 설명서의 정보를 제공합니다.
Sun Java Enterprise System 2005Q4 기술 개요	Java ES의 기술 및 개념적 기초를 소개합니다. 구성 요소, 구조, 프로세스 및 기능을 설명합니다.
Sun Java Enterprise System 2005Q4 배포 계획 설명서	Java ES 기반 엔터프라이즈 배포 솔루션의 계획 및 설계를 소개합니다. 배포 계획 및 설계의 기본 개념 및 원칙을 제시하고, 솔루션 라이프 사이클을 설명하며, Java ES 기반 솔루션 설계 시에 사용하는 수준 높은 전략 및 예제를 제공합니다.
Sun Java Enterprise System 2005Q4 설치 계획 설명서	Java ES 배포의 하드웨어, 운영 체제 및 네트워크 측면에 대한 구현 사양을 개발할 수 있도록 도와줍니다. 설치 및 구성 계획에서 제시되는 구성 요소 종속성과 같은 문제에 대해 설명합니다.

표 P-2 Java Enterprise System 설명서 (계속)

문서 제목	내용
<b>Sun Java Enterprise System 2005Q4 Installation Guide for UNIX</b>	Solaris 운영 체제 또는 Linux 운영 체제에 Java ES를 설치하는 과정에 대해 설명합니다. 또한 설치 후 구성 요소를 구성하는 방법과 구성 요소가 제대로 작동하는지를 확인하는 방법에 대해 설명합니다.
<b>Sun Java Enterprise System 2005Q4 설치 참조 설명서</b>	구성 매개 변수에 대한 추가 정보 및 구성 계획에 사용하는 워크시트를 제공하며, 기본 디렉토리 및 포트 번호와 같은 참조 자료를 나열합니다.
<b>Sun Java Enterprise System 2005Q1 배포 예 시리즈: 평가 시나리오</b>	한 시스템에 Java ES를 설치하고 핵심 공유 네트워크 서비스를 구축하며 구축한 서비스에 액세스할 수 있도록 사용자 계정을 설정하는 방법을 설명합니다.
<b>Sun Java Enterprise System 2005Q4 업그레이드 설명서</b>	Solaris 운영 체제 또는 Linux 운영 환경의 Java ES 업그레이드 지침을 제공합니다.
<b>Sun Java Enterprise System 용어집</b>	Java ES 설명서에서 사용되는 용어를 정의합니다.

Java ES 및 해당 구성 요소의 모든 설명서에 대한 URL은 <http://docs.sun.com/app/docs/coll/1314.1> 및 <http://docs.sun.com/app/docs/coll/1398.1>입니다.

## 기본 경로 및 파일 이름

다음 표에서는 이 설명서에서 사용되는 기본 경로 및 파일 이름을 설명합니다.

표 P-3 기본 경로 및 파일 이름

자리 표시자	설명	기본값
<i>ServiceRegistry-base</i>	Service Registry의 기본 설치 디렉토리를 나타냅니다.	Solaris 시스템: /opt/SUNWsoar  Linux 시스템: /opt/sun/SUNWsoar
<i>RegistryDomain-base</i>	Service Registry에 대한 Application Server 도메인 및 Service Registry 데이터베이스가 있는 디렉토리를 나타냅니다.	Solaris 시스템: /var/opt/SUNWsoar  Linux 시스템: /var/opt/sun/SUNWsoar

---

## 활자체 규약

다음 표는 이 책에서 사용된 활자체 변경 사항에 대하여 설명합니다.

표 P-4 활자체 규약

서체	의미	예
AaBbCc123	명령, 파일 및 디렉토리의 이름, 그리고 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오. ls -a 명령을 사용하여 모든 파일을 나열하십시오. machine_name% you have mail.
<b>AaBbCc123</b>	컴퓨터 화면 상의 출력과는 달리 사용자가 직접 입력하는 사항입니다.	machine_name% <b>su</b> Password:
AaBbCc123	명령줄 자리 표시자: 실제 이름이나 값으로 대체됩니다.	파일을 제거하려면 rm filename을 입력하십시오.
AaBbCc123	책 제목, 새로 나오는 용어, 강조 표시할 단어입니다. (일부 강조된 항목은 온라인상에서 볼드로 표시됩니다.)	<b>사용자 설명서</b> 의 6장을 읽으십시오. <b>캐시</b> 는 로컬로 저장된 복사본입니다. 파일을 저장하지 <b>마십시오</b> .

---

## 명령 예의 셸 프롬프트

기본 시스템 프롬프트 및 슈퍼유저 프롬프트는 다음 표와 같습니다.

표 P-5 셸 프롬프트

셸	프롬프트
UNIX 및 Linux 시스템의 C 셸	machine_name%
UNIX 및 Linux 시스템의 C 셸 슈퍼유저	machine_name#
UNIX 및 Linux 시스템의 Bourne 셸 및 Korn 셸	\$
UNIX 및 Linux 시스템의 Bourne 셸 및 Korn 셸 슈퍼유저	#
Microsoft Windows 명령줄	C:\

---

## 기호 규약

다음 표는 이 설명서에서 사용되는 기호를 설명합니다.

표 P-6 기호 규약

기호	설명	예	의미
[ ]	선택 인수 및 명령 옵션을 포함합니다.	ls [-l]	-l 옵션은 필수가 아닙니다.
{   }	필수 명령 옵션에 대한 일련의 선택 항목을 포함합니다.	-d {y n}	-d 옵션에서는 y 인수 또는 n 인수를 사용해야 합니다.
\${ }	변수 참조를 나타냅니다.	\${com.sun.javaRoot}	com.sun.javaRoot 변수 값을 참조합니다.
-	동시에 입력하는 여러 키를 결합합니다.	Ctrl-A	A 키를 누른 채로 Ctrl 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.

---

## Sun 자원 온라인 액세스

docs.sun.com<sup>SM</sup> 웹 사이트에서 Sun 기술 관련 설명서를 온라인으로 이용할 수 있습니다. docs.sun.com 아카이브를 탐색하거나 특정 책 제목 또는 주제를 검색할 수 있습니다. 설명서는 PDF 및 HTML 형식의 온라인 파일로 제공됩니다. 장애가 있는 사용자도 보조 기술을 이용하여 두 형식을 모두 읽을 수 있습니다.

다음의 Sun 자원에 액세스하려면 <http://www.sun.com>을 방문하십시오.

- Sun 제품 다운로드
- 서비스 및 솔루션
- 지원(패치 및 업데이트 포함)
- 교육
- 리서치
- 커뮤니티(예: Sun 개발자 네트워크)

---

## 타사 웹 사이트 참조

이 문서에서는 추가적인 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 발생한 사실이 있거나 발생했다고 주장이 제기되는 손해나 손실에 대해 책임이나 의무를 지지 않습니다.

---

---

## 사용자 의견 환영

Sun은 설명서의 내용 개선에 노력을 기울이고 있으며, 여러분의 의견과 제안을 환영합니다. 사용자 의견을 보내시려면 <http://docs.sun.com>에서 Send Comments(의견 보내기)를 누릅니다. 온라인 양식에 전체 설명서 제목과 부품 번호를 기입해 주시기 바랍니다. 부품 번호는 해당 설명서의 제목 페이지나 문서의 URL에 있으며 7자리 또는 9자리 숫자로 되어 있습니다. 예를 들어, 이 설명서의 부품 번호는 819-3575입니다.



# 1장

---

## Service Registry 구성 및 설정

---

이 장에서는 Service Registry를 설치한 후 구성하는 방법과 다른 관리 작업을 수행하는 방법에 대해 설명합니다.

이 장은 다음 절로 구성되어 있습니다.

- 15 페이지 “Service Registry 구성”
- 22 페이지 “외부 웹 사이트에 대한 액세스 허용”
- 23 페이지 “관리자 만들기”
- 25 페이지 “웹 콘솔 구성”
- 30 페이지 “Service Registry 다시 설치”
- 31 페이지 “데이터베이스 백업 및 복원”

---

## Service Registry 구성

**Sun Java Enterprise System 2005Q4 Installation Guide for UNIX**에서는 레지스트리에 대한 기본 등록 정보 설정을 사용하여 Service Registry를 설치한 다음 구성하는 방법에 대해 설명합니다. 사용자 정의 등록 정보 설정을 사용하려면 구성을 수행하기 전에 *ServiceRegistry-base* /install/install.properties를 편집합니다.

*ServiceRegistry-base* 위치는 Solaris 운영 환경에서는 /opt/SUNWsoar이고, Linux 시스템에서는 /opt/sun/SUNWsoar입니다.

주 - Service Registry를 구성하기 전에 Sun Java System Application Server("Application Server")를 설치하여 구성해야 합니다. Service Registry 구성 프로세스에서는 레지스트리를 Application Server 도메인에 설치합니다.

Application Server를 해당 기본 위치에 설치하는 것이 좋습니다. 기본 위치가 아닌 위치에 Application Server를 설치한 경우에는 21 페이지 "기본값이 아닌 Application Server 설치에 대한 Service Registry 구성"의 지침을 수행한 다음 Service Registry를 구성합니다.

install.properties 파일에는 일련의 수정 가능한 등록 정보 설정이 포함되어 있습니다. 표 1-1에 나열된 등록 정보를 설치 프로세스에서 사용합니다. 각 등록 정보 이름에는 registry.install. 접두어(마침표로 종료)가 붙어 있습니다. 대부분의 이러한 등록 정보에서는 레지스트리에 대해 생성된 Application Server 도메인에 대한 기본 포트가 아닌 포트를 설정합니다.

표 1-1 Service Registry 설치 등록 정보

등록 정보 이름	설명	기본 등록 정보 값
DomainName	Application Server 도메인 이름	registry
ServerInstancePort	Application Server HTTP 포트	6060
ServerInstanceSecurePort	Application Server HTTPS 포트	6443
ServerJMSPort	Application Server Message Queue 포트	6484
ServerIIOPPort	Application Server IIOP 포트	6485
ServerIIOPSecurePort	Application Server IIOP 보안 포트	6486
ServerIIOPMutualAuthPort	Application Server IIOP 상호 인증 포트	6487
AdministrationJMXPort	Application Server JMX 포트	6488
AdministrationPort	Application Server Administrative Server 포트	6489
AdministratorUserID	Application Server Administrative Server에 액세스하는 데 사용하는 사용자 이름	admin

표 1-1 Service Registry 설치 등록 정보 (계속)

등록 정보 이름	설명	기본 등록 정보 값
AdministratorPassword	Application Server Administrative Server에 액세스하는 데 사용하는 비밀번호	12345678
ApplicationServerKeystorePassword	Application Server 키 저장소에 액세스하는 데 사용하는 비밀번호	12345678
RegistryServerKeystorePassword	Service Registry 키 저장소에 액세스하는 데 사용하는 비밀번호	12345678

## ▼ 사용자 정의 등록 정보를 사용하여 나중에 구성 설치 후에 Service Registry를 구성하는 방법

시작하기 전에 레지스트리를 구성하려면 root로 로그인하거나 슈퍼유저가 되어야 합니다.

단계 1. *ServiceRegistry-base/install* 디렉토리로 변경합니다.

2. *install.properties* 파일에서 수정 가능한 등록 정보를 편집합니다.  
보안상의 이유로 비밀번호 값을 변경할 때는 이 파일을 편집하지 않는 것이 좋습니다. 대신 명령줄에서 이러한 값을 지정합니다.

3. 파일을 편집한 후 다음 명령을 실행합니다(모두 한 줄로 표시).

Solaris의 경우: `/usr/sfw/bin/ant -f build-install.xml install`

Linux의 경우: `/opt/sun/bin/ant --noconfig -f build-install.xml install`

ant 명령을 사용하려면 JAVA\_HOME 환경 변수가 설정되어 있어야 합니다. 일반적으로 이 환경 변수를 다음 값으로 설정합니다.

```
/usr/jdk/entsys-j2se
```

명령줄에서 변경된 비밀번호를 지정하려면 명령에 다음 옵션을 지정합니다(모두 한 줄로 표시).

```
/usr/sfw/bin/ant -f build-install.xml
-Dregistry.install.RegistryServerKeystorePassword=passwd1
-Dregistry.install.AdministratorPassword=passwd2
-Dregistry.install.ApplicationServerKeystorePassword=passwd3 install
```

레지스트리 구성 프로세스에서는

*RegistryDomain-base/domains/\${registry.install.DomainName}*에 Application Server 도메인을 만듭니다. 기본 도메인 이름은 *registry*입니다. 그런 다음 도메인을 시작하고 레지스트리를 배포하여 도메인을 실행 중인 상태로 둡니다.

레지스트리 구성 프로세스에서는 *RegistryDomain-base/3.0* 디렉토리에 레지스트리 데이터베이스 및 서버 키 저장소를 설치합니다. 이 디렉토리는 레지스트리를 제거해도 제거되지 않으므로 이후 릴리스에서 사용하도록 데이터베이스를 유지할 수 있습니다. 관리자는 이 디렉토리의 제거 시기와 제거 여부를 관리할 수 있습니다.

*RegistryDomain-base* 위치는 Solaris 운영 환경에서는 `/var/opt/SUNWsoar`이고 Linux 환경에서는 `/var/opt/sun/SUNWsoar`입니다.

4. **ant install** 명령의 출력에서 오류를 검토합니다.  
오류가 없는 경우 웹 콘솔 또는 관리 도구를 사용할 수 있습니다.

## ▼ 관리 도구 사용 활성화 방법

관리자 역할을 가진 사용자로 제한된 관리 도구 작업을 수행하려면 관리 도구 매니페스트 클래스 경로에 JAR 파일을 추가하여 버그를 해결해야 합니다.

- 단계
1. *ServiceRegistry-base/install* 디렉토리에 있어야 합니다.
  2. **soapprocessor.jar** 파일을 배포된 레지스트리에서 레지스트리 **lib** 디렉토리로 복사합니다. 다음 명령을 실행합니다(모두 한 줄로 표시).

```
cp
RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/lib/soapprocessor.jar
../lib
```

3. **admin.jar.manifest.fix** 설치 대상을 다음과 같이 실행합니다.  
Solaris의 경우: `/usr/sfw/bin/ant -f build-install.xml`  
**admin.jar.manifest.fix**

Linux의 경우: `/opt/sun/bin/ant --noconfig -f build-install.xml`  
**admin.jar.manifest.fix**

## Service Registry에 대한 Application Server 도메인 관리

Service Registry 구성 프로세스에서는 기본적으로 *registry*라는 Application Server 도메인을 만들며, 이 도메인에 Service Registry 웹 응용 프로그램이 배포됩니다. 이 도메인은 *RegistryDomain-base/domains/registry* 디렉토리에 있습니다.

이 위치는 Application Server 도메인의 기본 위치인 `/var/opt/SUNWappserver/domains`(Solaris) 또는 `/var/opt/sun/appserver/domains`(Linux)와 다릅니다.

*registry* 도메인을 관리하는 데 Application Server 관리 콘솔("관리 콘솔")을 사용할 수 있습니다. 관리 콘솔을 사용하여 도메인 시작 및 중지, 서버 로그 보기 및 기타 관리 작업을 수행할 수 있습니다. 자세한 내용은 19 페이지 "Application Server 관리 콘솔 사용 방법"을 참조하십시오.

서버 로그를 직접 검사할 수도 있습니다. 로그는 `RegistryDomain-base/domains/registry/logs/server.log` 파일에 있습니다.

`asadmin` 명령을 사용하여 `registry` 도메인을 관리할 수도 있습니다. 도메인이 기본 위치에 없으므로 `asadmin` 명령을 사용할 때 `--domaindir` 옵션을 지정해야 합니다.

`registry` 도메인에 대한 비밀번호 파일은 `ServiceRegistry-base/pw.txt`입니다. `registry` 도메인을 관리할 때 이 비밀번호 파일을 `asadmin` 명령의 `--passwordfile` 옵션에 대한 인수로 지정합니다.

`registry` 도메인은 기본 Application Server 도메인 `domain1`과 충돌이 발생하지 않도록 기본 포트가 아닌 포트를 사용합니다. 표 1-2에서는 이러한 포트를 나열하고 설명합니다. 자세한 내용은 **Sun Java System Application Server Enterprise Edition 8.1 2005Q2 관리 설명서**의 “Application Server의 포트”를 참조하십시오.

표 1-2 Service Registry 도메인 기본 포트

포트 값	설명
6060	HTTP 포트
6443	SSL에서의 HTTPS
6484	Message Queue 포트
6485	IIOF 포트
6486	IIOF SSL 포트
6487	IIOF 상호 인증 포트
6488	JMX 포트
6489	Application Server 도메인 관리 포트

## ▼ Application Server 관리 콘솔 사용 방법

- 단계
1. 웹 브라우저에서 URL `https://hostname:6489/`로 이동합니다.  
*hostname*은 Application Server 및 Service Registry가 실행 중인 시스템입니다.
  2. 제공되는 인증서에 동의합니다.  
로그인 페이지가 나타납니다.
  3. 로그인 페이지에서 사용자 이름 필드에 `admin`을 입력합니다.
  4. 비밀번호 필드에 Application Server 관리자 비밀번호를 입력합니다. 레지스트리를 구성할 때 `AdministratorPassword` 등록 정보에 대해 지정한 값을 사용합니다.  
기본값은 `12345678`입니다.
  5. 로그인을 누릅니다.

**참조** 관리 콘솔에 대한 자세한 내용은 관리 콘솔에 대한 온라인 도움말 또는 **Sun Java System Application Server Enterprise Edition 8.1 2005Q2 관리 설명서**를 참조하십시오.

Service Registry 로깅 수준을 변경하려면 **Sun Java System Application Server Enterprise Edition 8.1 2005Q2 관리 설명서**의 “로그 수준 구성” 지침을 따릅니다. 추가 등록 정보 영역에서 지정할 등록 정보는 `org.apache.commons.logging.simplelog.log.org.freebxml.omar`입니다.

## ▼ 레지스트리에 대해 Application Server 도메인 중단 및 다시 시작

레지스트리 구성 프로세스에서 레지스트리가 배포되는 Application Server 도메인을 시작합니다. 특정 관리 작업을 수행한 후에는 도메인을 중지했다가 다시 시작해야 합니다. 이러한 작업의 예는 22 페이지 “외부 웹 사이트에 대한 액세스 허용” 및 23 페이지 “관리자 만들기”입니다.

도메인을 다시 시작해야 하는지 여부를 관리 콘솔이 알려줍니다. 관리 콘솔을 사용하여 이 작업을 수행할 수 있습니다. `asadmin` 명령을 사용하는 경우 Ant 작업을 사용하여 도메인을 중지 및 시작할 수 있습니다.

**단계 1. Service Registry 설치 디렉토리로 변경합니다.**

```
cd ServiceRegistry-base/install
```

**2. 다음 명령을 실행합니다(모두 한 줄로 표시).**

```
Solaris: /usr/sfw/bin/ant -f build-install.xml  
appserver.domain.bounce
```

```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml  
appserver.domain.bounce
```

이 대상은 도메인을 중지하였다가 다시 시작시킵니다.

또한 `build-install.xml` 파일은 레지스트리 도메인 중지 및 시작을 위한 개별 Ant 대상을 포함합니다. 도메인을 중지하려면 Ant 대상 `appserver.domain.stop`을 사용합니다. 도메인을 시작하려면 Ant 대상 `appserver.domain.start`를 사용합니다.

## ▼ 레지스트리 도메인의 신뢰할 수 있는 인증서에 루트 인증서 추가

이 작업에서는 Application Server registry 도메인에서 신뢰할 수 있는 인증서 목록을 확장합니다.

아직 Application Server truststore에 없는 타사 인증서 및 해당 루트 인증 기관(CA) 인증서를 사용하는 경우에만 이 작업을 수행합니다. 레지스트리 발급 인증서만을 사용하는 경우 이 작업을 수행하지 않습니다.

- 단계 1. 지원할 루트 인증서를 다운로드합니다. 루트 인증서를 제공하는 사이트는 다음과 같습니다.
- <http://www.entrust.net/developer/>
  - [http://www.geotrust.com/resources/root\\_certificates/](http://www.geotrust.com/resources/root_certificates/)
  - <http://www.thawte.com/roots/>
  - <http://www.verisign.com/support/roots.html>
2. 필요한 경우 `unzip` 명령을 사용하여 다운로드한 아카이브에서 `.cer` 파일을 추출합니다.

---

주 - 일부 파일은 `.der` 접미어를 가집니다.

---

3. `.cer` 파일을 `ServiceRegistry-base/install/cacerts` 디렉토리에 복사합니다.
4. `ServiceRegistry-base/install` 디렉토리로 변경합니다.
5. 다음 명령을 실행합니다(모두 한 줄로 표시).
- ```
Solaris: /usr/sfw/bin/ant -f build-install.xml install.cacerts
```
- ```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml
install.cacerts
```
- 이 명령은 `ServiceRegistry-base/install/cacerts` 디렉토리에서 찾은 인증서를 Application Server 도메인 truststore에 설치합니다.
- `list.cacerts` 대상을 사용하여 인증서가 제대로 설치되었는지 확인할 수 있습니다.
6. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.

## 기본값이 아닌 Application Server 설치에 대한 Service Registry 구성

Application Server의 기본 설치 위치는 Solaris 운영 환경에서는 `/opt/SUNWappserver/appserver`이고, Linux 시스템에서는 `/opt/sun/appserver`입니다. Application Server를 다른 위치에 설치한 경우에는 Service Registry를 구성하기 전에 `install.properties` 파일을 편집해야 합니다.

### ▼ `install.properties` 파일 편집 방법

- 단계 1. `ServiceRegistry-base/install` 디렉토리의 `install.properties` 파일을 텍스트 편집기에서 엽니다.

2. `appserver.root.dir` 등록 정보의 주석 처리된 정의를 찾습니다.
3. 주석 문자(#)를 제거하고 등록 정보 정의를 `Application Server`의 실제 위치로 바꿉니다.
4. `install.properties` 파일을 저장하고 닫습니다.

다음 순서 15 페이지 “Service Registry 구성”의 지침에 따라 작업을 계속 수행합니다.

## 기본값이 아닌 Service Registry 설치에 대한 Service Registry 구성

Service Registry의 기본 설치 위치는 Solaris 운영 환경에서는 `/opt/SUNWsoar`이고, Linux 시스템에서는 `/opt/sun/SUNWsoar`입니다. Service Registry를 다른 위치에 설치한 경우에는 Service Registry를 구성하기 전에 `install.properties` 파일을 편집해야 합니다.

### ▼ `install.properties` 파일 편집 방법

- 단계
1. `ServiceRegistry-base/install` 디렉토리의 `install.properties` 파일을 텍스트 편집기에서 엽니다.
  2. `soar.sdk.home` 및 `soar.server.home` 등록 정보의 주석 처리된 정의를 찾습니다.
  3. 각 등록 정보에 대해 주석 문자(#)를 제거하고 등록 정보 정의를 Service Registry의 실제 위치로 바꿉니다.
  4. `install.properties` 파일을 저장하고 닫습니다.

다음 순서 15 페이지 “Service Registry 구성”의 지침에 따라 작업을 계속 수행합니다.

---

## 외부 웹 사이트에 대한 액세스 허용

레지스트리 객체는 해당 레지스트리 객체에 연결된 URL을 지정하는 `ExternalLink` 객체를 가질 수 있습니다. 사용자가 `ExternalLink` 객체를 만들려면 Service Registry에서 URL을 검증할 수 있어야 하며, 이 작업을 수행하려면 외부 웹 사이트에 액세스해야 합니다. 레지스트리가 방화벽 뒤에 배포되어 있는 경우 이 액세스를 허용하도록 프록시 구성을 설정해야 합니다.

프록시 구성에서는 웹 프록시 호스트 및 포트를 Service Registry가 배포되는 Application Server 인스턴스의 JVM(Java Virtual Machine) 옵션으로 지정해야 합니다.

## ▼ 웹 프록시 지정 방법

- 단계
1. 19 페이지 “Application Server 관리 콘솔 사용 방법”에 설명한 대로 Application Server 관리 콘솔에 로그인합니다.
  2. 구성 노드를 확장합니다.
  3. 서버 노드 `server-config (Admin Config)` 를 확장합니다.
  4. JVM 설정을 누릅니다.
  5. JVM 옵션 탭을 누릅니다.
  6. JVM 옵션 추가를 누릅니다.
  7. 텍스트 필드에 다음을 입력합니다(모두 한 줄로 표시).  
`-Dhttp.proxyHost=hostname.domainname -Dhttp.proxyPort=8080`  
포트 값은 일반적으로 8080입니다. 포트가 사용자 위치와 다른 경우 올바른 값을 지정합니다.
  8. 저장을 누릅니다.
  9. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.

---

## 관리자 만들기

Service Registry 관리 도구에는 관리자로 등록된 사용자만 수행할 수 있는 몇 가지 작업이 있습니다. 또한 관리자를 호출하여 다른 사용자가 제출하는 객체에 대한 라이프 사이클 변경(예: 승인)을 구현할 수 있습니다.

또한 관리자는 기본 액세스 제어 정책(ACP)을 변경해야 할 수도 있습니다. 그러나, 현재는 수동 프로세스를 통해 ACP를 작성하므로 OASIS XACML(eXtensible Access Control Markup Language)을 잘 알고 있어야 합니다. 자세한 내용은 ebXML RIM 3.0의 9장, “액세스 제어 정보 모델”에서 특히, 9.7.6 ~ 9.7.8 절의 예를 참조하십시오. ebXML RIM 3.0 사양을 찾는 방법에 대한 자세한 내용은 7 페이지 “이 설명서를 읽기 전에”를 참조하십시오.

## ▼ 관리자를 만드는 방법

자신을 관리자로 등록하려면 다음 단계를 수행합니다.

- 단계
1. **Service Registry 3 2005Q4 사용자 설명서의 “사용자 계정 만들기”에 설명된 대로 사용자 등록을 수행합니다.**  
 다운로드한 인증서의 경로 이름을 기억해 두십시오. 인증서 파일의 기본 이름은 `generated-key.p12`입니다.
  2. 다음과 같이 사용자 `User` 객체의 고유 식별자를 가져옵니다.
    - a. 사용자로 설정된 객체 유형으로 웹 콘솔을 사용하여 기본 쿼리를 수행합니다.
    - b. 사용자에 대해 설정된 레지스트리의 `User` 객체를 보려면 세부 정보 링크를 클릭합니다.
    - c. 고유 식별자 필드 값을 적어둡니다.
  3. 인증서를 홈 디렉토리의 다음 위치에 복사합니다. 필요한 경우 디렉토리를 만듭니다.  
`$HOME/soar/3.0/jaxr-ebxml/security`
  4. `RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes` 디렉토리로 변경합니다.
  5. `omar.properties` 파일을 텍스트 편집기에서 엽니다.
  6. `omar.security.authorization.registryAdministrators` 등록 정보에 대한 정의를 찾습니다.
  7. 단계 2에 대해 적어 둔 논리 식별자 문자열 앞에 세로 막대(|)를 추가하여 등록 정보 정의를 편집합니다.  
 등록 정보 정의는 한 줄로 입력해야 하며 공백을 포함할 수 없습니다. 완료하면 다음과 같이 나타납니다(모두 한 줄로 표시).  

```
omar.security.authorization.registryAdministrators=
urn:freebxml:registry:predefinedusers:registryoperator|
urn:uuid:77f5c196-79de-4286-8483-8d80def3583b
```
  8. `omar.properties` 파일을 저장한 다음 닫습니다.
  9. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.

다음 순서 추가 관리자를 만들기 위해 `omar.properties` 파일을 편집할 필요가 없습니다. 관리자는 관리 도구 또는 웹 콘솔을 사용하여 사용자를 추가하고 웹 콘솔을 사용하여 사용자를 관리자로 분류할 수 있습니다.

---

## 웹 콘솔 구성

관리자는 구성 파일을 편집하여 웹 콘솔 표시의 일부 측면을 사용자 정의할 수 있습니다. 이 절에서는 다음 작업에 대해 설명합니다.

- 25 페이지 “미리 정의된 쿼리 추가”
- 27 페이지 “기본 쿼리 변경”
- 27 페이지 “분류 체계 숨기기”
- 28 페이지 “검색 결과 표시 구성”

웹 콘솔 사용에 대한 자세한 내용은 **Service Registry 3 2005Q4 사용자 설명서**를 참조하십시오.

### 미리 정의된 쿼리 추가

Service Registry에는 미리 정의된 쿼리 선택 드롭다운 목록의 웹 콘솔 검색 양식에 표시되는 미리 정의된 여러 쿼리가 포함되어 있습니다. 관리자는 해당 레지스트리 설치에 특정한 새 쿼리를 드롭다운 목록에 추가할 수 있습니다.

#### ▼ 미리 정의된 쿼리를 추가하는 방법

- 단계 1. 웹 콘솔을 사용하여 레지스트리에 **AdhocQuery** 객체를 게시합니다.

쿼리에 대해 지정하는 이름 및 설명이 미리 정의된 쿼리의 드롭다운 목록에 나타납니다. 쿼리에 대한 SQL 문에서 다음과 같이 사용자가 제공한 데이터를 작은따옴표 쌍으로 묶어 자리 표시자를 지정합니다.

```
select * from registryobject where id = '$lid'
```

2. SQL 문의 자리 표시자 및 **AdhocQuery** 객체에 대한 고유 식별자를 적어둡니다.
3. *RegistryDomain-base/3.0/jaxr-ebxml* 디렉토리로 변경합니다.
4. **registry-browser-config.xml** 파일을 텍스트 편집기에서 엽니다.
5. 다음 형식을 사용하여 **registry-browser-config.xml** 파일에 항목을 추가합니다. SQL 문의 각 자리 표시자에 대한 **Parameter** 요소를 지정합니다.

```
<Query>
  <AdhocQueryRef id="unique_identifier"/>
  <Parameter parameterName="$placeholder_name" datatype="string">
    <rim:Name>
      <rim:LocalizedString xml:lang="en" charset="UTF-8"
        value="parameter_name_in_en_locale"/>
      <rim:LocalizedString xml:lang="fr" charset="UTF-8"
```

```

        value="parameter_name_in_fr_locale" />
    </rim:Name>
    <rim:Description>
        <rim:LocalizedString xml:lang="en" charset="UTF-8"
            value="parameter_description_in_en_locale" />
        <rim:LocalizedString xml:lang="fr" charset="UTF-8"
            value="parameter_description_in_fr_locale" />
    </rim:Description>
</Parameter>
...
</Query>

```

*unique\_identifier*는 AdhocQuery 객체의 고유 식별자입니다.

각 매개 변수의 *parameterName* 속성 값은 쿼리에 대한 SQL 문의 자리 표시자에서 와야 합니다.

*datatype* 속성은 다음 값 중 하나를 가질 수 있습니다.

- **string**: 매개 변수가 검색 양식의 텍스트 필드로 나타납니다.
- **taxonomyElement**: 매개 변수가 검색 양식의 드롭다운 목록으로 나타납니다. *taxonomyElement* 데이터 유형을 지정하는 경우 이름 및 설명 요소 뒤에 다음과 같은 *SlotList* 요소가 있어야 합니다.

```

<rim:SlotList>
  <rim:Slot name="domain">
    <rim:ValueList>
      <rim:Value>
        classification_scheme_or_concept_id
      </rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:SlotList>

```

*classification\_scheme\_or\_concept\_id*는 해당 개념(또는 하위 개념)이 드롭다운 목록에 나타나야 하는 분류 체계 또는 개념의 고유 식별자입니다. 레지스트리에 해당 체계가 아직 없는 경우 분류 체계를 게시해야 합니다.

슬롯 이름은 "domain"이어야 합니다.

- **boolean**: 매개 변수가 검색 양식에서 확인란으로 표시됩니다.

*datatype*이 **string** 또는 **boolean**이면 *Parameter* 요소에 *defaultValue* 속성을 추가하여 검색 양식에 표시할 기본값을 지정할 수도 있습니다.

지원하는 로케일의 각 매개 변수 이름 및 설명에 대해 현지화된 문자열 값을 지정합니다. 현재 로케일의 *parameter\_name*은 검색 양식에 매개 변수 레이블로 표시됩니다.

*registry-browser-config.xml* 파일의 기존 항목을 참조용으로 사용합니다.

## 6. **registry-browser-config.xml** 파일을 저장한 다음 닫습니다.

7. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.

## 기본 쿼리 변경

미리 정의된 쿼리 선택 드롭다운 목록에 기본값으로 표시되는 쿼리는 사용자가 이름, 설명 및 분류 기준으로 레지스트리 객체를 검색할 수 있는 기본 쿼리입니다.

관리자는 이 기본값을 설치에 적절한 쿼리로 변경할 수 있습니다. 예를 들어, [25 페이지 "미리 정의된 쿼리 추가"](#)에 설명된 대로 기본 쿼리를 레지스트리에 추가한 미리 정의된 새 쿼리로 지정할 수 있습니다. 이를 변경하려면 구성 파일의 등록 정보를 편집합니다.

### ▼ 기본 쿼리를 변경하는 방법

단계 1. `RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes` 디렉토리로 변경합니다.

2. 텍스트 편집기에서 `jaxr-ebxml.properties` 파일을 엽니다.

3. `jaxr-ebxml.thin.defaultQueryPanel`의 등록 정보 정의를 찾습니다. 기본적으로 이 등록 정보는 다음과 같이 주석 처리됩니다.

```
#jaxr-ebxml.thin.defaultQueryPanel=
```

4. 주석 문자(#)를 제거합니다.

5. 다음 예와 같이 기본값이 될 쿼리의 논리 식별자를 지정하여 등록 정보 값을 설정합니다.

```
jaxr-ebxml.thin.defaultQueryPanel=urn:oasis:names:tc:ebxml-regrep:query:MyQuery
```

6. `jaxr-ebxml.properties` 파일을 저장한 다음 닫습니다.

7. [20 페이지 "레지스트리에 대해 Application Server 도메인 중단 및 다시 시작"](#)의 지침을 따릅니다.

## 분류 체계 숨기기

분류 체계의 트리 구조는 다음 웹 콘솔 영역에 표시됩니다.

- 기본 쿼리 선택 시의 검색 양식 영역
- 탐색 영역 메뉴
- 일부 유형의 레지스트리 객체에 대해 개념을 선택해야 할 때 표시되는 ClassificationScheme/개념 선택기 창

Service Registry 사용자가 분류 체계를 사용하지 못하게 하려면 관리자는 보기에서 해당 분류 체계를 숨길 수 있습니다. 분류 체계를 숨기려면 구성 파일에서 등록 정보를 정의합니다.

## ▼ 분류 체계 숨기기 방법

단계 1. *RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes* 디렉토리로 변경합니다.

2. 텍스트 편집기에서 `jaxr-ebxml.properties` 파일을 엽니다.

3. 다음 구문을 사용하여

`jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList` 등록 정보를 설정합니다. 모든 등록 정보 정의는 한 줄로 입력해야 하며 공백을 포함할 수 없습니다.

```
jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList=  
class_scheme_id1|class_scheme_id2|...
```

숨길 각 분류 체계의 논리 식별자를 지정합니다. 2개 이상의 식별자를 지정하려면 다음 예와 같이 세로 막대(|)를 사용하여 식별자를 구분합니다.

```
jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList=  
urn:oasis:names:tc:ebxml-regrep:classificationScheme:StatusType|  
urn:oasis:names:tc:ebxml-regrep:profile:ws:classificationScheme:BindingType
```

4. `jaxr-ebxml.properties` 파일을 저장한 다음 닫습니다.

5. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.

## 검색 결과 표시 구성

웹 콘솔은 기본적으로 각 쿼리에 대해 한 번에 10개의 검색 결과를 표시합니다. 10개가 넘는 검색 결과가 반환되면 사용자는 추가 결과 페이지를 표시할 수 있습니다. 관리자는 각 페이지에 표시되는 검색 결과 수를 수정할 수 있습니다.

웹 콘솔은 기본적으로 검색 결과 영역에 특정 열을 표시합니다. 각 객체에 대해 객체 유형, 이름, 설명, 버전 및 버전 주석을 표시합니다. 일부 객체 유형의 경우에는 기본 표시와는 다른 표시로 구성됩니다. 예를 들어, `ServiceBinding` 객체의 경우 버전 정보 대신에 종점이 포함되어 표시됩니다. 관리자는 구성 정보를 추가하여 선택 객체 유형에 대한 기본값이 아닌 데이터를 표시할 수 있습니다.

이러한 작업을 각각 수행하려면 구성 파일을 편집합니다.

## ▼ 검색 결과 표시에 행 수 구성

단계 1. *RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes* 디렉토리로 변경합니다.

2. 편집기에서 `jaxr-ebxml.properties` 파일을 엽니다.

3. `omar.client.thinbrowser.numSearchResults` 등록 정보의 정의를 찾습니다.  

```
omar.client.thinbrowser.numSearchResults=10
```
4. 10을 원하는 값으로 변경합니다.
5. `jaxr-ebxml.properties` 파일을 저장한 다음 닫습니다.
6. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.

## ▼ 검색 결과 영역의 열 구성

- 단계
1. `RegistryDomain-base/3.0/jaxr-ebxml` 디렉토리로 변경합니다.
  2. 텍스트 편집기에서 `registry-browser-config.xml` 파일을 엽니다.
  3. 다음 형식을 사용하여 `registry-browser-config.xml` 파일에 항목을 추가합니다.

다음 예제는 Service 객체에 대해 기본 표시와 다른 표시를 구성합니다.

```
<ObjectTypeConfig
  className="org.freebxml.omar.client.xml.registry.infomodel.ServiceImpl"
  id="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Service">
  <SearchResultsConfig>
    <SearchResultsColumn columnName="Object Type" columnWidth="25" editable="false"
      method="getObjectType"/>
    <SearchResultsColumn columnName="Name" columnWidth="25" editable="true" method="getName"/>
    <SearchResultsColumn columnName="Description" columnWidth="30" editable="true"
      method="getDescription"/>
    <SearchResultsColumn columnName="Status" columnWidth="15" method="getStatusAsString"/>
    <SearchResultsColumn columnName="Version" columnWidth="5" method="getVersionName"/>
  </SearchResultsConfig>
</ObjectTypeConfig>
```

`registry-browser-config.xml`은 `ObjectTypeConfig` 요소에 대한 구문을 제공합니다. 파일에 있는 기존 요소를 예로 사용합니다. 이러한 요소는 `ExternalLink`, `ExtrinsicObject` 및 `ServiceBinding` 객체 및 레지스트리 객체에 대해 기본 표시와는 다른 표시를 구성합니다.

대부분의 경우 **ebXML Registry Information Model Version 3.0** 사양의 클래스 속성에서 메소드 이름을 추정할 수 있습니다(자세한 내용은 7 페이지 “이 설명서를 읽기 전에” 참조). `getStatusAsString` 메소드는 `RegistryObjectImpl` 구현 클래스에서 찾을 수 있습니다. (그러나 이번 Service Registry 릴리스에는 API 문서가 포함되지 않습니다.)

4. registry-browser-config.xml 파일을 저장한 다음 닫습니다.
5. 20 페이지 “레지스트리에 대해 Application Server 도메인 중단 및 다시 시작”의 지침을 따릅니다.
6. 재구성을 확인하려면 웹 콘솔의 검색 또는 탐색 메뉴를 사용하여 해당 열이 변경된 객체를 표시합니다.

---

## Service Registry 다시 설치

Service Registry를 제거하고 다시 설치해야 하는 경우에는 다시 설치하기 전에 다음 작업을 수행합니다.

- 레지스트리 데이터베이스에 보존할 데이터가 포함되어 있는 경우 31 페이지 “데이터베이스 백업 및 복원”에 설명된 대로 데이터베이스를 백업합니다.
- 레지스트리에 대한 Application Server 도메인을 중지한 다음 삭제합니다. 도메인을 삭제하지 않은 경우에는 다시 설치된 레지스트리의 사후 설치 구성이 실패합니다.

Service Registry 데이터베이스를 다시 설치해야 하는 경우(예: 데이터베이스가 손상된 경우) 31 페이지 “Service Registry 데이터베이스 다시 설치”의 지침을 따릅니다. 이 데이터베이스는 다시 설치하기 전에 제거할 필요가 없습니다.

### ▼ 레지스트리에 대한 Application Server 도메인 중지 및 삭제

단계 1. ServiceRegistry-base/install 디렉토리로 변경합니다.

2. 다음 명령을 실행합니다(모두 한 줄로 표시).

```
Solaris: /usr/sfw/bin/ant -f build-install.xml
appserver.domain.stop
```

```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml
appserver.domain.stop
```

3. 다음 명령을 실행합니다(모두 한 줄로 표시).

```
Solaris: /usr/sfw/bin/ant -f build-install.xml
appserver.domain.delete
```

```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml
appserver.domain.delete
```

## ▼ Service Registry 데이터베이스 다시 설치

- 단계 1. *ServiceRegistry-base/install* 디렉토리로 변경합니다.
2. 다음 명령을 실행합니다(모두 한 줄로 표시).
- ```
Solaris: /usr/sfw/bin/ant -f build-install.xml install.db
```
- ```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml  
install.db
```

---

## 데이터베이스 백업 및 복원

레지스트리에서는 Apache Derby 데이터베이스를 사용합니다. 기본적으로 데이터베이스는 *RegistryDomain-base/3.0/data/registry/soar/* 디렉토리에 있습니다.

데이터베이스를 백업 및 복원하는 방법은 Apache Derby 설명서를 참조하십시오.

## ▼ Apache Derby 설명서를 찾는 방법

- 단계 1. 웹 브라우저에서 Apache Derby 웹 사이트 (<http://db.apache.org/derby/>)로 이동합니다.
2. 설명서 탭을 누릅니다.
3. 10.0 설명서를 누릅니다.
4. 서버 및 관리 설명서를 찾습니다.
5. 데이터베이스 백업 및 복원을 설명하는 절을 찾습니다.



# 관리 도구 사용

---

이 장에서는 Service Registry에 대한 관리 도구("관리 도구")를 사용하는 방법에 대해 설명합니다.

이 장은 다음 절로 구성되어 있습니다.

- 33 페이지 "관리 도구 정보"
- 34 페이지 "관리 도구 시작"
- 36 페이지 "관리 도구 기능"
- 38 페이지 "관리 도구 명령 사용"

---

## 관리 도구 정보

Service Registry 관리 도구는 레지스트리에 연관 추가, 레지스트리에서 객체 제거 등과 같은 일반적인 관리 작업을 위한 간단한 명령줄 인터페이스를 제공합니다.

관리 도구는 다음 두 모드 중 하나에서 작동할 수 있습니다.

- 일괄 처리 모드 - 도구의 명령줄에서 하나 이상의 명령을 지정합니다.
- 대화형 모드 - 도구의 대화식 셸에 명령을 입력합니다.

ls 및 rm과 같은 여러 명령은 파일 및 폴더에 실행되는 잘 알려진 UNIX® 명령의 이름과 동작을 따릅니다. 다른 명령에는 해당하는 UNIX 명령이 없습니다.

---

## 관리 도구 시작

관리 도구를 시작하려면 `admin-tool.jar` 파일을 다음과 같이 실행합니다.

```
java -jar ServiceRegistry-base/lib/admin-tool.jar [options]...
```

`ServiceRegistry-base` 위치는 Solaris 운영 환경에서는 `/opt/SUNWsoar`이고, Linux 시스템에서는 `/opt/sun/SUNWsoar`입니다.

도구를 시작할 때 표시되는 경고는 중요하지 않으므로 무시합니다.

관리 도구를 종료하려면 `quit` 명령을 사용합니다.

관리자에게 제한된 명령을 수행하려면 명령줄에 다음과 같이 인증서 파일(키 저장소)의 유형, 위치 및 비밀번호를 명령줄에 지정해야 합니다(모두 한 줄로 표시).

```
java -Djaxr-ebxml.security.storetype=PKCS12 \  
-Djaxr-ebxml.security.keystore=security/filename.p12 \  
-Djaxr-ebxml.security.storepass=password -jar admin-tool.jar
```

여기서 `filename`은 인증서 파일의 루트 이름(기본값: `generated-key`)이고 `password`는 인증서를 만들 때 이 파일을 보호하기 위해 지정한 비밀번호입니다.

`security/filename.p12` 위치는 `$HOME/soar/3.0/jaxr-ebxml` 디렉토리에 상대적입니다.

입력 내용을 저장하려면 이 명령을 실행하는 스크립트를 만듭니다.

## 일괄 처리 모드

관리 도구를 일괄 처리 모드로 실행하려면 관리 도구를 시작할 때 명령줄에 `-command` 옵션을 지정합니다.

예를 들어, 다음 명령은 `ls` 명령을 실행합니다.

```
java -jar ServiceRegistry-base/lib/admin-tool.jar -command "ls *.html"
```

관리 도구는 명령 및 도구 응답을 화면에 표시하며 명령이 실행된 후에는 종료됩니다.

해당 쉘에 중요한 문자를 적절한 이스케이프 문자로 처리해야 합니다.

## 대화형 모드

관리 도구를 대화형 모드로 실행하려면 명령줄에서 `-command` 이외의 옵션을 지정하여(또는 옵션 없음) 관리 도구 쉘을 시작합니다.

```
java -jar ServiceRegistry-base/lib/admin-tool.jar
```

관리 도구는 다음 프롬프트를 표시하고 사용자의 입력을 기다립니다.

```
admin>
```

## 관리 도구 명령줄 옵션

관리 도구는 다음의 35 페이지 “개요”에 나열되고 35 페이지 “옵션”에 설명된 명령줄 옵션을 인식합니다.

### 개요

```
[-alias alias] [- command commands] [-debug] [-help]  
[-keypass keypass] [-localdir localdir] [-locale locale] [- registry url]  
[-root locator [-create]] [-sqlselect SQL_statement] [-verbose]
```

### 옵션

- alias** 키 저장소에서 사용자 인증서에 액세스할 때 사용하는 별칭입니다. 사용자로 등록할 때 사용한 별칭을 지정합니다.
- command** 대화형 셸에서 명령을 가져오는 대신 실행할 관리 도구 명령 시퀀스입니다. 세미콜론(;)을 사용하여 여러 명령을 구분합니다. *commands*에 quit 명령을 포함할 필요는 없습니다. 명령 구분자가 아닌 세미콜론을 사용해야 하는 경우 백슬래시 앞에 세미콜론을 입력합니다.  

```
\;
```

관리 도구를 실행하는 셸에서는 이 백슬래시에 백슬래시를 하나 더 사용하여 이스케이프 문자로 처리해야 할 수도 있습니다.  

```
\\;
```

명령에 공백이 포함되어 있는 경우에는 도구에서 해당 시퀀스를 여러 명령줄 매개 변수가 아닌 단일 명령줄 매개 변수로 취급하도록 전체 명령 시퀀스를 작은따옴표 또는 큰따옴표로 묶습니다. 또한 셸에서 세미콜론을 셸 구분 명령으로 해석하는 경우에는 여러 관리 도구 명령의 시퀀스를 항상 따옴표로 묶어야 합니다.
- create** 필요한 경우 -root 옵션에 지정된 RegistryPackage 및 상위 RegistryPackage 객체를 만듭니다. 이 옵션은 관리 도구를 실행하는 사용자가 객체 만들기 권한이 있는 경우에만 사용할 수 있습니다.
- debug** 디버깅할 때 유용한 추가 정보를 출력합니다.
- help** 이러한 옵션 목록을 제공합니다.

-keypass	키 저장소에서 사용자 인증서에 액세스할 때 사용하는 비밀번호입니다. 사용자로 등록할 때 사용한 비밀번호를 지정합니다.
-localdir	로컬 파일 시스템에 있는 파일과 관련된 명령에 대한 로컬 파일 시스템의 기본 디렉토리입니다.
-locale	오류 및 상태 메시지에 대해 사용할 자원 번들을 선택하는 데 사용되는 로케일(예: en 또는 fr)입니다. 기본값은 JVM(Java Virtual Machine)에 의해 결정됩니다.
-registry	연결할 ebXML 레지스트리의 URL입니다. 기본값은 http://localhost:6060/soar/registry/soap입니다.
-root	저장소를 서로 다른 RegistryObject 및 RegistryPackage 객체를 포함하는 RegistryPackage 객체의 트리로 취급하는 이러한 명령의 기초로 사용하는 RegistryPackage의 로케이터(예: /registry/userData)입니다. 기본값은 모든 사용자 데이터 /registry/userData에 대해 정의된 RegistryPackage입니다.
-sqlselect	SQL_statement를 실행하여 레지스트리 객체를 선택합니다. 이 문은 select로 시작하는 전체 SQL 문이어야 합니다. SQL 문은 따옴표로 묶어야 하지만 세미콜론으로 종료할 필요는 없습니다. 이 옵션을 지정한 다음 select 명령을 인수 없이 사용하면 SQL_statement 이외의 인수와 함께 select 명령을 사용할 때까지는 SQL_statement가 실행됩니다.
-v   -verbose	상태 메시지의 세부 정보 표시 출력을 지정합니다.

---

주 - -help 옵션의 출력에는 이 릴리스에서 지원되지 않는 -class 및 -property가 나열됩니다.

---

## 관리 도구 기능

이 절에서는 다음과 같은 관리 도구 기능에 대해 설명합니다.

- 37 페이지 “권한”
- 37 페이지 “예외 표시”
- 37 페이지 “레지스트리 객체 식별”
- 38 페이지 “이름 지정에 대한 로케일 효과”
- 38 페이지 “대소문자 구분”

## 권한

관리 도구를 사용하는 경우에는 도구를 시작할 때 키 별칭 및 비밀번호를 지정한 사용자에게 허용되는 작업만 수행할 수 있습니다. 특정 명령(예: `chown`)은 관리자 역할을 가진 사용자만 수행할 수 있습니다. 자세한 내용은 23 페이지 “관리자 만들기”를 참조하십시오.

## 예외 표시

관리 도구를 사용하면 명령이 실패할 때 자세한 스택 추적 정보가 표시되지 않습니다.

명령이 실패할 경우 관리 도구는 스택 추적의 첫 번째 행을 인쇄하고 다음과 같은 메시지를 표시합니다.

```
An error occurred when executing the function. Use the show exception command to view messages.
```

자세한 내용을 보려면 `show exception` 명령을 실행하여 전체 스택 추적을 참조하십시오.

`show exception` 명령은 항상 바로 앞에 실행된 명령의 스택 추적을 표시합니다.

## 레지스트리 객체 식별

레지스트리 객체를 식별할 때는 기본적으로 이름을 사용합니다. 하지만 일반적으로 `registry` 루트에서 `RegistryPackage`까지의 경로를 통해 `RegistryPackage` 객체를 식별합니다. 예를 들어, `/registry/userData`는 `userData` `RegistryPackage`의 경로입니다.

일부 이름 일치에서는 와일드카드를 지원합니다. 단일 문자와 일치시키려면 물음표(?)를 사용하고 0개 이상의 문자를 일치시키려면 별표(\*)를 사용합니다.

일부 명령(예: `cd` 및 `chown`)은 URN(Uniform Resource Name)을 통한 객체 식별을 지원하므로 앞에 `urn:`을 붙여야 합니다. 예를 들어, `urn:uuid:2702f889-3ced-4d49-82d1-e4cd846cb9e4`는 유효한 URN입니다.

또한 `chown` 명령에서는 `%number`를 사용하여 이전 `users` 명령에 나열된 User를 참조할 수 있습니다.

일부 명령에서는 전체 이름을 큰따옴표로 묶거나 이름의 각 공백 앞에 백슬래시를 입력하여 공백을 포함하는 이름을 입력할 수 있습니다.

`select` 명령은 다음과 같은 SQL 와일드카드 사용을 지원합니다. 백분율 기호(%)는 여러 문자와 일치하고 밑줄(\_)은 단일 문자와 일치합니다.

## 이름 지정에 대한 로케일 효과

RegistryObject(또는 RegistryPackage)는 각각 서로 다른 로케일에 연결되는 여러 이름을 가질 수 있습니다.

지정하는 경로 및 객체 이름은 현재 로케일에 대해서만 평가됩니다. 여러 이름을 가진 레지스트리 객체를 이름별로 선택할 경우 레지스트리는 제공된 이름을 해당 레지스트리 객체의 모든 이름이 아니라 레지스트리 객체 이름에 대한 단일 선택 항목(로케일이 현재 로케일과 가장 일치하는 선택 항목)에 대해서만 일치시킵니다.

예를 들어, 현재 RegistryPackage에 en(영어) 로케일의 red 및 fr(프랑스어) 로케일의 rouge와 같이 서로 다른 로케일에 연관된 두 개의 이름을 가진 구성원 객체가 있다고 가정합니다. 현재 로케일이 en인 경우 ls rouge 명령은 해당 구성원 객체를 표시하지 않지만 로케일이 fr(또는 해당 변형 중 하나)인 경우에는 해당 구성원 객체를 표시합니다.

## 대소문자 구분

관리 도구에서 인식되는 명령 이름 및 리터럴 매개 변수는 대소문자 구분이 없습니다. 예를 들어, ls, Ls 및 LS는 모두 같은 값을 갖습니다.

값이 제공된 옵션은 해당 옵션을 사용하는 코드에 문자 그대로 전달됩니다.

---

## 관리 도구 명령 사용

다음 절에서는 사용 가능한 명령에 대해 설명합니다. 각 명령의 옵션 및 피연산자에 대한 개요 및 설명에서는 다음의 활자체 규약을 따릅니다.

- *Italics*는 명령을 실행할 때 실제 값으로 대체되어야 하는 옵션 인수 또는 피연산자를 나타냅니다.
- 중괄호({ })는 옵션 또는 피연산자 선택을 구분하며, 그 안에는 옵션 또는 피연산자 중 하나가 포함되어야 합니다. 옵션 또는 피연산자는 세로 막대(|)로 구분됩니다.
- 대괄호([ ])는 옵션 또는 피연산자나 생략될 수 있는 옵션 또는 피연산자 선택을 구분합니다.
- 옵션 또는 피연산자 뒤의 줄임표(...)는 해당 인수 또는 피연산자를 반복할 수 있음을 나타냅니다.

그 외의 활자체는 명령을 실행할 때 포함시켜야 하는 리터럴 텍스트입니다.

## add association

레지스트리에 Association 객체를 추가합니다.

### 개요

```
add association -type association-type sourceURN targetURN
```

### 설명

add association 명령은 지정된 유형의 Association 객체를 레지스트리에 추가합니다. 다음과 같은 유형을 사용할 수 있습니다.

- AccessControlPolicyFor
- AffiliatedWith(EmployeeOf 및 MemberOf 하위 개념 포함)
- Contains
- ContentManagementServiceFor
- EquivalentTo
- Extends
- ExternallyLinks
- HasFederationMember
- HasMember
- Implements
- InstanceOf
- InvocationControlFileFor(CatalogingControlFileFor 및 ValidationControlFileFor 하위 개념 포함)
- OffersService
- OwnerOf
- RelatedTo
- Replaces
- ResponsibleFor
- SubmitterOf
- Supersedes
- Uses

## 옵션

-type Association 객체 유형입니다.

## 피연산자

*sourceURN* 소스 객체의 URN입니다.

*targetURN* 대상 객체의 URN입니다.

## 예

다음 명령(모두 한 줄로 표시)은 지정된 두 URN이 있는 객체 간의 RelatedTo 관계를 만듭니다.

```
admin> add association -type RelatedTo
urn:uuid:ab80d8f7-3bea-4467-ad26-d04a40045446
urn:uuid:7a54bbca-2131-4a49-8ecc-e7b4ac86c4fd
```

## add user

레지스트리에 사용자를 추가합니다.

## 개요

```
add user [-edit] [- load filename] [-firstname string]
[-lastname string] [-middleName string] - alias string - keypass string
[-post1.type string] [-post1.city string] [-post1.country string]
[- post1.postalcode string] [-post1.stateOrProvince string]
[-post1.street string] [-post2.streetNumber string ]
[-post2.type string] [-post2.city string] [- post2.country string]
[-post2.postalcode string] [-post2.stateOrProvince string]
[-post2.street string] [-post2.streetNumber string ]
[-post3.type string] [-post3.city string] [- post3.country string]
[-post3.postalcode string] [-post3.stateOrProvince string]
[-post3.street string] [-post3.streetNumber string ]
[-telephone1.type string] [-telephone1.areaCode string]
[-telephone1.countryCode string] [-telephone1.extension string]
[-telephone1.number string] [- telephone1.URL string]
[-telephone2.type string] [-telephone2.areaCode string]
[-telephone2.countryCode string] [-telephone2.extension string]
[-telephone2.number string] [-telephone2.URL string ]
[-telephone3.type string] [-telephone3.areaCode string]
```

```
[-telephone3.countryCode string] [-telephone3.extension string]
[-telephone3.number string] [- telephone3.URL string]
[-email1.type string] [-email1.address string] [-email2.type string]
[-email12address string] [- email3.type string]
[-email3.address string]
```

## 설명

add user 명령은 User 객체를 추가합니다. User 객체는 일반적으로 하나 이상의 PostalAddress, TelephoneNumber 및 EmailAddress 객체를 포함합니다.

명령줄을 사용하거나 -load 옵션을 통해 정보가 있는 Java 등록 정보 파일을 지정하여 사용자에게 대한 정보를 지정합니다. 정보 옵션 및 -load 옵션은 명령줄에 표시된 순서대로 평가됩니다. 예를 들어, 명령줄에서 일부 등록 정보를 지정하고 등록 정보 파일에서 다른 등록 정보를 로드한 다음 이후의 명령줄 옵션으로 등록 정보 파일의 정보를 대체할 수 있습니다.

새 사용자에게 대해 최대 세 개의 주소, 전화 번호 및 전자 메일 주소를 지정할 수 있습니다. 더 많은 정보를 지정할 필요가 있는 경우에는 나중에 웹 콘솔 또는 JAXR을 사용하여 추가할 수 있습니다.

주소, 전화 번호 또는 전자 메일 주소를 지정할 때는 -emailType OfficeEmail과 같이 해당 유형에 대한 값을 제공해야 합니다.

모든 사용자에게 필요한 일부 일반 정보의 경우 명령줄에서 약자 옵션(예: -fn)을 사용할 수 있지만 등록 정보에서 정보를 입력할 때는 긴 형식을 사용해야 합니다. 예를 들어, 명령줄에서 -email1.address, -emailAddress 또는 -email을 사용하여 사용자의 첫 번째 전자 메일 주소를 지정할 수 있지만, 등록 정보 파일에서 첫 번째 전자 메일 주소를 지정할 때는 email1.address=를 사용해야 합니다. 사용자의 두 번째 전자 메일 주소 지정 옵션은 하나만 있으므로 명령줄에서는 -email2.address를 사용하고 등록 정보 파일에서는 email2.address=를 사용해야 합니다.

-edit 옵션을 지정하는 경우 관리 도구는 사용자가 새 사용자 정보를 편집할 수 있도록 편집기를 시작합니다. 자세한 내용은 옵션 설명을 참조하십시오.

---

주 - -load를 사용하여 로드하거나 -edit를 사용하여 편집하는 등록 정보 파일은 모든 Java 등록 정보 파일과 마찬가지로 ISO-8859-1 문자 집합을 사용합니다. 등록 정보 파일에서 ISO-8859-1이 아닌 다른 문자를 표시하는 방법에 대한 자세한 내용은 java.util.Properties.load(InputStream)에 대한 설명서를 참조하십시오.

---

## 옵션

### -edit

새 사용자 정보를 편집할 수 있도록 관리 도구에서 편집기를 시작합니다. 관리 도구는 다른 명령줄 매개 변수를 평가한 후에 편집기를 시작하므로 명령줄이나 등록 정보 파일에 지정된 정보의 평가 결과를 사용하여 편집이 시작됩니다. 편집 프로그램이 오류 없이 종료되어야 명령을 계속 실행할 수 있습니다. 관리 도구는 `set editor` 명령(55 페이지 “set” 참조)에서 지정한 편집기(기본적으로 vi 편집기)를 시작합니다.

---

주 - 이 릴리스에서 `-edit`는 `emacsclient` 및 `NetBeans™` 명령 `bin/runide.sh` `--open`(원활하게 작동되지는 않음)과 함께 작동하지만 `vi`에서는 작동하지 않습니다.

---

### -load

해당 콘텐츠에서 사용자에 대한 등록 정보를 지정하는 Java 등록 정보 파일을 지정합니다. 등록 정보 이름은 긴 형식의 `add user` 명령 옵션과 동일합니다(예: `lastName` 및 `post1.type`).

### -fn | -firstName

사용자의 이름을 지정합니다.

### -ln | -lastName

사용자의 성을 지정합니다. 성은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다.

### -mn | -middleName

사용자의 중간 이름을 지정합니다.

### -alias

키 저장소에서 사용자 인증서에 액세스할 때 사용하는 별칭입니다. 이 옵션은 필수 항목입니다. 별칭은 최소 길이가 3자 이상이어야 합니다.

### -keypass

키 저장소에서 사용자 인증서에 액세스할 때 사용하는 비밀번호입니다. 이 옵션은 필수 항목입니다. 비밀번호는 최소 길이가 6자 이상이어야 합니다.

### -postalType | -post1.type

첫 번째 `PostalAddress`의 유형입니다. 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열(예: `Office` 또는 `Home`)입니다.

### -city | -post1.city

첫 번째 `PostalAddress`의 구/군/시입니다.

### -country | -post1.country

첫 번째 `PostalAddress`의 국가입니다.

### -postalCode | -postcode | -zip | -post1.postalcode

첫 번째 `PostalAddress`의 우편 번호입니다.

-stateOrProvince | -state | -province | -post1.stateOrProvince  
 첫 번째 PostalAddress의 시/도입니다.

-street | -post1.street  
 첫 번째 PostalAddress의 주소입니다.

-streetNumber | -number | -post1.streetNumber  
 첫 번째 PostalAddress의 상세 주소입니다.

-post2.type  
 두 번째 PostalAddress의 유형입니다. 두 번째 PostalAddress를 지정하는 경우 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열(예: Office 또는 Home)입니다.

-post2.city  
 두 번째 PostalAddress의 구/군/시입니다.

-post2.country  
 두 번째 PostalAddress의 국가입니다.

-post2.postalcode  
 두 번째 PostalAddress의 우편 번호입니다.

-post2.stateOrProvince  
 두 번째 PostalAddress의 시/도입니다.

-post2.street  
 두 번째 PostalAddress의 주소입니다.

-post2.streetNumber  
 두 번째 PostalAddress의 상세 주소입니다.

-post3.type  
 세 번째 PostalAddress의 유형입니다. 세 번째 PostalAddress를 지정하는 경우 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열(예: Office 또는 Home)입니다.

-post3.city  
 세 번째 PostalAddress의 구/군/시입니다.

-post3.country  
 세 번째 PostalAddress의 국가입니다.

-post3.postalcode  
 세 번째 PostalAddress의 우편 번호입니다.

-post3.stateOrProvince  
 세 번째 PostalAddress의 시/도입니다.

-post3.street  
 세 번째 PostalAddress의 주소입니다.

-post3.streetNumber  
 세 번째 PostalAddress의 상세 주소입니다.

- phoneType | -telephone1.type  
 첫 번째 TelephoneNumber의 유형입니다. 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열이지만 Beeper, FAX, HomePhone, MobilePhone 또는 OfficePhone과 같은 알려진 유형 중에서 하나를 지정할 수 있습니다.
- areaCode | -telephone1.areaCode  
 첫 번째 TelephoneNumber의 지역 번호입니다.
- countryCode | -telephone1.countryCode  
 첫 번째 TelephoneNumber의 국가 코드입니다.
- extension | -telephone1.extension  
 첫 번째 TelephoneNumber의 내선 번호입니다.
- number | -telephone1.number  
 국가 코드 또는 지역 번호를 제외한 첫 번째 TelephoneNumber의 전화 번호 접미어입니다. 번호는 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다.
- URL | -telephone1.URL  
 첫 번째 TelephoneNumber의 URL(이 번호에 온라인으로 전화를 걸 수 있는 URL)입니다.
- telephone2.type  
 두 번째 TelephoneNumber의 유형입니다. 두 번째 TelephoneNumber를 지정하는 경우 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열이지만 Beeper, FAX, HomePhone, MobilePhone 또는 OfficePhone과 같은 알려진 유형 중에서 하나를 지정할 수 있습니다.
- telephone2.areaCode  
 두 번째 TelephoneNumber의 지역 번호입니다.
- telephone2.countryCode  
 두 번째 TelephoneNumber의 국가 코드입니다.
- telephone2.extension  
 두 번째 TelephoneNumber의 내선 번호입니다.
- telephone2.number  
 국가 코드 또는 지역 번호를 제외한 두 번째 TelephoneNumber의 전화 번호 접미어입니다. 두 번째 TelephoneNumber를 지정하는 경우 번호는 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다.
- telephone2.URL  
 두 번째 TelephoneNumber의 URL(이 번호에 온라인으로 전화를 걸 수 있는 URL)입니다.
- telephone3.type  
 세 번째 TelephoneNumber의 유형입니다. 세 번째 TelephoneNumber를 지정하는 경우 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열이지만 Beeper, FAX, HomePhone, MobilePhone 또는 OfficePhone과 같은 알려진 유형 중에서 하나를 지정할 수 있습니다.

-telephone3.areaCode  
세 번째 TelephoneNumber의 지역 번호입니다.

-telephone3.countryCode  
세 번째 TelephoneNumber의 국가 코드입니다.

-telephone3.extension  
세 번째 TelephoneNumber의 내선 번호입니다.

-telephone3.number  
국가 코드 또는 지역 번호를 제외한 세 번째 TelephoneNumber의 전화 번호 접미어입니다. 세 번째 TelephoneNumber를 지정하는 경우 번호는 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다.

-telephone3.URL  
세 번째 TelephoneNumber의 URL(이 번호에 온라인으로 전화를 걸 수 있는 URL)입니다.

-emailType | -email1.type  
첫 번째 EmailAddress의 유형입니다. 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열이지만 HomeEmail 또는 OfficeEmail과 같은 알려진 유형 중에서 하나를 지정할 수 있습니다.

-emailAddress | -email | -email1.address  
첫 번째 전자 메일 주소입니다. 첫 번째 전자 메일 주소는 필수 항목입니다.

-email2.type  
두 번째 EmailAddress의 유형입니다. 두 번째 EmailAddress를 지정하는 경우 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열이지만 HomeEmail 또는 OfficeEmail과 같은 알려진 유형 중에서 하나를 지정할 수 있습니다.

-email2.address  
두 번째 전자 메일 주소입니다.

-email3.type  
세 번째 EmailAddress의 유형입니다. 세 번째 EmailAddress를 지정하는 경우 유형은 필수 항목이며 명령줄 또는 등록 정보 파일에서 지정해야 합니다. 이 값은 임의의 문자열이지만 HomeEmail 또는 OfficeEmail과 같은 알려진 유형 중에서 하나를 지정할 수 있습니다.

-email3.address  
세 번째 전자 메일 주소입니다.

## 예

다음 명령은 사용자 홈 디렉토리의 JaneSmith.properties 파일에서 User 등록 정보를 로드합니다.

```
admin> add user -load ~/JaneSmith.properties
```

다음 명령(모두 한 줄로 표시)은 User 객체를 만드는 데 필요한 최소 등록 정보를 지정합니다.

```
admin> add user -ln Smith -postaltype Office -country US
-phonetype Office -number 333-3333 -emailtype OfficeEmail
-emailaddress JaneSmith@JaneSmith.com -alias 123 -keypass 123456
```

## cd

RegistryPackage 위치를 변경합니다.

## 개요

```
cd { locator | URN }
```

## 설명

cd 명령은 디렉토리(상징적인 개념)를 지정된 경로에 있거나 지정된 URN을 가진 RegistryPackage로 변경합니다.

이 명령은 현재 로컬에 대해 동일한 경로를 가진 RegistryPackage 객체가 여러 개 있는 경우 지정된 URN으로 변경합니다.

## 피연산자

**locator** 레지스트리 객체 이름에 대해 저장소 루트에서 저장소 객체까지에 해당하는 경로이며 각 이름 앞에는 슬래시(/)가 붙어 있습니다.

예를 들어, registry RegistryPackage의 구성원(로케이터 자체는 RegistryPackage의 구성원이 아님)인 userData RegistryPackage에 대한 로케이터는 /registry/userData이고, userData RegistryPackage의 구성원인 folder1 RegistryPackage에 대한 로케이터는 /registry/userData/folder1입니다.

**URN** RegistryPackage의 URN이며 urn:으로 시작해야 합니다.

## 예

다음 명령은 urn:uuid:92d3fd01-a929-4eba-a5b4-a3f036733017 URN을 사용하여 디렉토리를 RegistryPackage로 변경합니다.

```
admin> cd urn:uuid:92d3fd01-a929-4eba-a5b4-a3f036733017
```

다음 명령은 디렉토리를 /registry/userData/myData 위치로 변경합니다.

```
admin> cd /registry/userData/myData
```

## chown

RegistryObject의 소유자를 변경합니다.

### 개요

**chown** { *URN* | *%index* }

### 설명

chown 명령은 이전 select 명령을 사용하여 선택한 객체의 소유자를 이전 users 명령에 나열된 사용자 URN에 대한 참조 또는 URN에 지정된 사용자로 변경합니다.

관리자 역할을 가진 사용자만 이 명령을 실행할 수 있습니다.

### 피연산자

*URN* URN에 지정된 User 객체입니다.

*%index* 이전 users 명령에 나열된 User 객체의 URN에 대한 숫자 참조입니다.

### 예

다음 명령은 선택된 객체의 소유권을

urn:uuid:26aa17e6-d669-4775-bfe8-a3a484d3e079 URN에 지정된 사용자로 변경합니다.

```
admin> chown urn:uuid:26aa17e6-d669-4775-bfe8-a3a484d3e079
```

다음 명령은 선택된 객체의 소유권을 이전 users 명령에서 숫자 2로 표시된 사용자로 변경합니다.

```
admin> chown %2
```

## cp

파일 및 폴더를 레지스트리에 복사합니다.

## 개요

**cp** [-owner {URN | % index}] [- exclude pattern]... [-include pattern]... pattern...

## 설명

cp 명령은 폴더 및 파일을 레지스트리에 RegistryPackage 객체와 ExtrinsicObject 객체로 각각 복사합니다.

로컬 파일 시스템에서 파일 및 폴더를 복사할 원본 로컬 디렉토리의 기본값은 관리 도구를 시작한 현재 디렉토리입니다. 관리 도구를 시작할 때 -localdir 옵션을 사용하여 로컬 디렉토리를 변경할 수 있으며 관리 도구를 시작한 후에는 lcd 명령을 사용하여 로컬 디렉토리를 변경할 수 있습니다. show localdir 명령을 사용하여 현재 로컬 디렉토리의 절대 경로를 가져올 수 있습니다.

명령은 순환적입니다. 즉, 디렉토리를 지정하면 이 명령이 해당 디렉토리의 모든 파일과 폴더를 복사합니다.

## 옵션

- owner 복사된 레지스트리 객체의 소유자를 URN 또는 %index 인수에 지정된 사용자로 설정합니다. 이러한 인수에 대한 설명은 chown 명령의 설명을 참조하십시오. 자신 이외의 소유자를 지정하려면 관리자 역할을 갖고 있어야 합니다.
- exclude 이름에 지정된 패턴이 있는 파일을 제외한 모든 파일을 복사합니다. 여기서 pattern은 리터럴 문자와 특수 문자인 별표(\*) (0개 이상의 문자를 나타냄) 및 물음표(?) (1과 하나의 문자를 나타냄)로 구성되는 패턴입니다.  
  
이 옵션은 여러 번 지정할 수 있습니다.
- include 이름에 지정된 패턴이 있는 모든 파일을 복사합니다. 여기서 pattern은 리터럴 문자와 특수 문자인 별표(\*) (0개 이상의 문자를 나타냄) 및 물음표(?) (1과 하나의 문자를 나타냄)로 구성되는 패턴입니다.  
  
이 옵션은 여러 번 지정할 수 있습니다.

## 피연산자

pattern 리터럴 문자와 특수 문자인 별표(\*) (0개 이상의 문자를 나타냄) 및 물음표(?) (1과 하나의 문자를 나타냄)로 구성되는 패턴으로 지정된 복사할 파일 또는 폴더입니다. 여러 pattern을 지정할 수 있습니다.

## 예

다음 명령은 이전 `users` 명령에서 숫자 4인 사용자가 소유하도록 `mydir` 디렉토리를 레지스트리에 복사합니다.

```
admin> cp -owner %4 mydir
```

다음 명령은 `.z` 또는 `.c` 문자열로 끝나는 파일과 디렉토리를 제외하고 `mydir` 디렉토리를 레지스트리에 복사합니다.

```
admin> cp mydir -exclude *.z -exclude *.c
```

## echo

문자열을 표시합니다.

## 개요

`echo string`

## 설명

`echo` 명령은 지정된 `string`을 출력으로 표시합니다. 이 명령은 일괄 처리 모드에서 관리 도구를 실행할 때 `-command` 옵션에서 지정하면 가장 유용합니다.

## 피연산자

`string` 문자 시퀀스입니다.

## 예

다음 명령은 `ls` 명령의 결과와 날짜를 로그 파일에 인쇄합니다.

```
java -jar admin-tool.jar -command "echo "date"; ls" > admin.log
```

## help

명령에 대한 정보를 표시합니다.

## 개요

**help** [*command\_name*]

## 설명

help 명령은 사용 가능한 명령 또는 지정된 명령에 대한 정보를 표시합니다.

add 및 show와 같이 명령에 하위 명령이 있는 경우 help 명령은 하위 명령에 대한 정보를 표시합니다.

인수를 지정하지 않은 경우 help 명령은 모든 명령에 대한 사용 정보를 표시합니다.

## 피연산자

*command\_name*    관리 도구 명령의 이름입니다.

## 예

다음 명령은 모든 명령에 대한 사용 정보를 표시합니다.

```
admin> help
```

다음 명령은 lcd 명령에 대한 사용 정보를 표시합니다.

```
admin> help lcd
```

다음 명령은 add 하위 명령에 대한 사용 정보를 표시합니다.

```
admin> help add
```

## lcd

로컬 파일 시스템에서 현재 디렉토리를 변경합니다.

## 개요

**lcd** [*path\_name*]

## 설명

lcd 명령은 로컬 파일 시스템에서 현재 로컬 디렉토리를 변경합니다.

인수를 지정하지 않을 경우 `lcd` 명령은 현재 디렉토리를 기본 홈 디렉토리로 변경합니다.

## 피연산자

*path\_name* 절대적 또는 상대적 디렉토리 이름입니다.

## 예

다음 명령은 현재 로컬 디렉토리를 `/usr/share` 디렉토리로 변경합니다.

```
admin> lcd /usr/share
```

다음 명령은 현재 로컬 디렉토리를 로컬 파일 시스템의 기본 홈 디렉토리로 변경합니다.

```
admin> lcd
```

## ls

현재 `RegistryPackage`에 있는 객체를 나열합니다.

## 개요

```
ls [ { pattern | URN }...]
```

## 설명

인수를 지정하지 않을 경우 `ls` 명령은 현재 `RegistryPackage`에 있는 객체를 나열합니다. *pattern* 또는 *URN*이 제공된 경우 현재 `RegistryPackage`에서 이름(현재 로컬) 또는 고유 식별자가 *pattern* 또는 *URN*과 일치하는 객체를 나열합니다.

## 피연산자

*pattern* 리터럴 문자와 특수 문자인 별표(\*) (0개 이상의 문자를 나타냄) 및 물음표(?) (1과 하나의 문자를 나타냄)로 구성되는 패턴입니다. 여러 *pattern*을 지정할 수 있습니다.

*URN* `urn:`으로 시작하는 URN(예: `urn:uuid:4a6741e7-4be1-4cfb-960a-e5520356c4fd`)입니다. 여러 *URN*을 지정할 수 있습니다. URN은 로컬 식별자가 아니라 객체의 고유 식별자여야 합니다.

## 예

다음 명령은 현재 RegistryPackage에 있는 모든 객체를 나열합니다.

```
admin> ls
```

다음 명령은 이름이 urn:bird:poultry:chicken 패턴과 일치하거나 ID가 urn:bird:poultry:chicken인 모든 객체를 나열합니다.

```
admin> ls urn:bird:poultry:chicken
```

다음 명령은 이름이 \*bird\* 패턴과 일치하는 모든 객체를 나열합니다. 또한 ID가 \*bird\*인 객체를 나열합니다(\*bird\*가 유효한 ID인 경우).

```
admin> ls *bird*
```

다음 명령은 이름이 \*bird\* 또는 urn:bird:poultry:chicken 패턴과 일치하거나 ID가 urn:bird:poultry:chicken인 모든 객체를 나열합니다.

```
admin> ls *bird* urn:bird:poultry:chicken
```

## pwd

현재 RegistryPackage의 경로를 표시합니다.

## 개요

**pwd**

## 설명

pwd 명령은 현재 로컬에 대해 가장 일치하는 이름을 사용하여 현재 RegistryPackage의 경로와 함께 해당 경로에 대한 로컬을 표시합니다.

## 예

```
admin> pwd  
(en_US) /registry/userData
```

## quit

관리 도구를 종료합니다.

## 개요

**quit**

## 설명

quit 명령은 관리 도구를 종료합니다.

## 예

```
admin> quit
```

## rm

RegistryPackage에서 객체를 제거합니다.

## 개요

```
rm [-d] [-r] { pattern | URN}...
```

## 설명

rm 명령은 이름(현재 로컬)이 *pattern* 또는 *URN*에 지정된 패턴과 일치하는 현재 RegistryPackage의 구성원 객체를 제거합니다.

일치하는 RegistryObject가 여러 RegistryPackage 객체의 구성원인 경우 이 명령은 현재 RegistryPackage와 객체 간의 연관만 제거합니다. 연관을 제거할 때 다른 RegistryObject와 연관이 없는 객체가 남아 있는 경우에만 레지스트리에서 해당 객체가 제거됩니다.

일치하는 구성원 객체 자체가 다른 객체를 포함하는 RegistryPackage일 때는 -r 또는 -d 옵션을 지정한 경우에만 현재 RegistryPackage와 구성원 RegistryPackage 간의 연관과 해당 객체가 제거됩니다.

-d 옵션과 -r 옵션을 모두 지정할 경우 -d 옵션이 순환적으로 적용되므로 -r에 의해 선택되는 모든 객체와 해당 연관은 다른 연관이 있는지 여부에 관계 없이 모두 제거됩니다.

## 옵션

- d 현재 RegistryPackage와 지정된 RegistryPackage 사이의 연관을 제거합니다. 구성원 객체에 대한 연관만 남아 있는 경우에만 지정된 RegistryPackage를 제거합니다. 다른 HasMember 연관의 대상으로 지정되어 고정되지 않은 현재 제거된 RegistryPackage 구성원 객체를 레지스트리의 루트 구성원으로 액세스할 수 있습니다.
- r 지정된 RegistryPackage 객체와 모든 해당 하위 객체(객체에 다른 연관이 있는 경우는 제외)를 제거합니다.

## 피연산자

*pattern* 리터럴 문자와 특수 문자인 별표(\*) (0개 이상의 문자를 나타냄) 및 물음표(?) (1과 하나의 문자를 나타냄)로 구성되는 패턴입니다. 여러 *pattern*을 지정할 수 있습니다.

*URN* urn:으로 시작하는 URN(예: urn:uuid:4a6741e7-4be1-4cfb-960a-e5520356c4fd)입니다. 여러 URN을 지정할 수 있습니다.

## 예

다음 명령은 "stat" 문자열을 포함하는 모든 RegistryPackage 객체와 모든 해당 하위 객체를 제거합니다.

```
admin> rm -r *stat*
```

## select

select SQL 문을 실행합니다.

## 개요

```
select [SQL]
```

## 설명

select 명령은 전체 명령을 SQL 쿼리로 평가하여 지정된 객체를 선택하고 나열합니다. 인수를 지정하지 않을 경우 이 명령은 이전 select 명령 또는 -sqlselect 옵션으로 선택한 모든 객체를 나열합니다.

## 피연산자

**SQL** `select` SQL 문입니다. 선행 `select`는 명령의 이름으로 이미 표시되어 있으므로 사용되지 않습니다.

## 예

다음 명령은 레지스트리에 있는 모든 `ClassificationScheme` 객체를 나열합니다.

```
admin> select s.* from ClassificationScheme s
```

## set

등록 정보 값을 설정합니다.

## 개요

```
set property value
```

## 설명

`set` 명령은 관리 도구 쉘의 등록 정보 값을 설정합니다.

관리 도구는 다음과 같은 등록 정보와 값을 지원합니다.

```
set debug {true | on | yes | false | off | no}
```

디버깅 메시지 출력을 활성화 또는 비활성화합니다.

```
set editor string
```

관리 도구에서 대화형 편집기를 시작할 때 사용할 명령을 설정합니다. UNIX 및 Linux 시스템에서 기본값은 `/bin/vi`입니다.

```
set verbose { true | on | yes | false | off | no}
```

명령을 실행할 때 추가적인 세부 정보 표시 메시지의 출력을 활성화 또는 비활성화합니다.

## 피연산자

*property* 다음 등록 정보 중 하나입니다. `debug`, `editor`, `verbose`.

*value* 지정된 등록 정보에 지원되는 값입니다. 자세한 내용은 설명 절을 참조하십시오.

## 예

다음 명령은 편집기를 기본 `/bin/vi` 대신 `/usr/bin/vi`로 설정합니다.

```
admin> set editor /usr/bin/vi
```

다음 명령은 디버깅을 설정합니다.

```
admin> set debug true
```

다음 명령은 세부 정보 표시 출력을 해제합니다.

```
admin> set verbose off
```

## show

등록 정보 값을 표시합니다.

## 개요

**show** [*property*]

## 설명

`show` 명령은 관리 도구 셸의 등록 정보 값을 표시합니다.

인수를 지정하지 않을 경우 이 명령은 모든 등록 정보 값을 표시합니다.

이 명령은 다음 등록 정보를 지원합니다.

<code>debug</code>	디버깅 출력이 활성화되는지 여부를 지정합니다.
<code>editor</code>	관리 도구에서 대화형 편집기를 시작할 때 사용하는 편집기를 지정합니다.
<code>exception</code>	바로 앞에 실행한 명령에서 예외 스택(있는 경우)을 추적합니다.
<code>localdir</code>	로컬 파일 시스템의 현재 디렉토리를 지정합니다. <code>lcd</code> 명령을 사용하여 이 등록 정보를 설정합니다. 자세한 내용은 <a href="#">50 페이지 "lcd"</a> 를 참조하십시오.
<code>locale</code>	현재 로케일을 지정합니다.
<code>verbose</code>	세부 정보 표시 출력을 활성화할지 여부를 지정합니다.

## 피연산자

*property* 현재 값을 표시할 등록 정보를 지정합니다. *exception* 및 *locale* 등록 정보를 표시할 수 있지만 *set* 명령을 사용하여 해당 등록 정보를 설정할 수는 없습니다.

## 예

다음 명령은 이전 명령의 예외를 표시합니다.

```
admin> show exception
```

## users

현재 User 객체를 나열합니다.

## 개요

```
users
```

## 설명

*users* 명령은 현재 레지스트리에 있는 User 객체를 나열합니다.

다음과 같은 형식으로 출력됩니다.

```
%index: URN lastname, firstname middlename
```

출력에서 *index*는 *chown* 또는 *cp* 명령을 실행할 때 사용자를 참조하는 데 사용할 수 있는 백분율 기호(%)를 포함한 숫자 값입니다. *lastname*, *firstname* 및 *middlename*은 사용자의 성, 이름 및 중간 이름입니다.

## 예

다음 명령은 현재 사용자를 표시합니다.

```
admin> users
%0: urn:freebxml:registry:predefinedusers:registryoperator Operator, Registry
%1: urn:freebxml:registry:predefinedusers:registryguest Guest, Registry
%2: urn:freebxml:registry:predefinedusers:farrukh Najmi, Farrukh Salahudin
%3: urn:freebxml:registry:predefinedusers:nikola Stojanovic, Nikola
%4: urn:uuid:799cc524-b7cd-4e51-8b34-d93b79ac52de User, Test
%5: urn:uuid:85428d8e-1bd5-473b-a8c8-b9d595f82728 Parker, Miles
```



# 색인

---

## A

AdhocQuery 객체, 웹 콘솔에 추가, 25-30  
-alias 명령줄 옵션, 35  
Application Server 관리 콘솔, 19-20  
Application Server 도메인  
  truststore에 루트 인증서 추가, 20-21  
  관리, 18-20  
  중지 및 다시 시작, 20

## C

cd 명령, 46  
chown 명령, 47  
-command 명령줄 옵션, 35  
cp 명령, 47-49  
-create 명령줄 옵션, 35

## D

-debug 명령줄 옵션, 35

## E

echo 명령, 49  
ExternalLink 객체, URL 검증 허용, 22-23

## G

Glossary, link to, 10

## H

help 명령, 49-50  
-help 명령줄 옵션, 35

## K

-keypass 명령줄 옵션, 35

## L

lcd 명령, 50-51  
-localdir 명령줄 옵션, 36  
-locale 명령줄 옵션, 36  
ls 명령, 51-52

## P

pwd 명령, 52

## Q

quit 명령, 52-53

## R

-registry 명령줄 옵션, 36  
RegistryPackage 객체  
  경로 표시, 52

RegistryPackage 객체 (계속)  
구성원 객체 제거, 53-54  
내용 나열, 51-52  
만들기, 35  
RegistryPackage 위치, 변경, 46  
RegistryPackage에서 객체 삭제, 53-54  
RegistryPackage에서 객체 제거, 53-54  
rm 명령, 53-54  
-root 명령줄 옵션, 36

## S

select 명령, 54-55  
Service Registry 구성, 15-22  
Service Registry에 대한 데이터베이스, 백업 및 복원, 31  
ServiceBinding 객체, URL 검증 허용, 22-23  
set 명령, 55-56  
show 명령, 56-57  
SQL select 문, 실행, 36  
SQL 문, 실행, 54-55  
-sqlselect 명령줄 옵션, 36

## U

users, 나열, 57  
users 명령, 57

## V

-v 명령줄 옵션, 36  
-verbose 명령줄 옵션, 36

## 관

관리 도구  
명령줄 옵션, 35-36  
사용 활성화, 18  
소개, 33  
시작, 34-36  
중지, 52-53  
관리 도구 시작, 34-36  
관리 도구 종료, 52-53  
관리 도구 중지, 52-53

관리 콘솔, Application Server, 19-20  
관리자, 만들기, 23-24  
관리자 만들기, 23-24

## 대

대소문자 구분, 38  
대화형 모드, 34-35

## 등

등록 정보 값  
설정, 55-56  
표시, 56-57  
등록 정보 값 설정, 55-56

## 디

디렉토리, 변경, 50-51  
디버그 등록 정보  
값 표시, 56-57  
설정, 55-56

## 레

레지스트리 객체  
나열, 51-52  
소유자 변경, 47  
식별, 37  
레지스트리 도메인  
truststore에 루트 인증서 추가, 20-21  
관리, 18-20  
중지 및 다시 시작, 20  
레지스트리에 파일 및 폴더 복사, 47-49

## 로

로컬, 이름 지정 효과, 38  
로컬 등록 정보, 값 표시, 56-57

## 루

루트 인증서, 레지스트리 도메인 truststore에  
추가, 20-21

## 명

### 명령

cd, 46  
chown, 47  
cp, 47-49  
echo, 49  
help, 49-50  
lcd, 50-51  
ls, 51-52  
pwd, 52  
quit, 52-53  
rm, 53-54  
select, 54-55  
set, 55-56  
show, 56-57  
users, 57  
사용자 추가, 40-46  
연관 추가, 39-40  
명령줄 옵션, 35-36  
-alias, 35  
-command, 35  
-create, 35  
-debug, 35  
-help, 35  
-keypass, 35  
-localdir, 36  
-locale, 36  
-registry, 36  
-root, 36  
-sqlselect, 36  
-v, 36  
-verbose, 36

## 미

미리 정의된 쿼리, 웹 콘솔에 추가, 25-30

## 사

사용자, 레지스트리에 추가, 40-46

사용자 이름, 명령줄에서 지정, 35

사용자 추가 명령, 40-46

## 설

설치 등록 정보, 16-17

## 세

세부 정보 표시 등록 정보

값 표시, 56-57

설정, 55-56

## 연

연관, 레지스트리에 추가, 39-40

연관 추가 명령, 39-40

## 예

예외, 표시, 37

예외 등록 정보, 값 표시, 56-57

## 와

와일드카드, 사용, 37

## 외

외부 웹 사이트, 액세스 허용, 22-23

## 웹

웹 콘솔, 미리 정의된 쿼리 추가, 25-30

## 일

일괄 처리 모드, 34

## **파**

파일 및 폴더, 레지스트리에 복사, 47-49

파일 시스템, 로컬

    기본 디렉토리, 36

    현재 디렉토리 변경, 50-51

## **편**

편집기 등록 정보

    값 표시, 56-57

    설정, 55-56

## **포**

포트, Service Registry 기본값, 19

## **표**

표시, 등록 정보 값, 56-57

## **프**

프록시 호스트 및 포트, 설정, 22-23

## **현**

현재 디렉토리, 변경, 50-51