



Service Registry 3 2005Q4 管理指南

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

文件号码 819-3573
2005 年 10 月

版权所有 2005 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 保留所有权利。

本产品或文档受版权保护，其使用、复制、发行和反编译均受许可证限制。未经 Sun 及其许可方（如果有）的事先书面许可，不得以任何形式、任何手段复制本产品或文档的任何部分。第三方软件，包括字体技术，均已从 Sun 供应商处获得版权和使用许可。

本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地区独家许可的注册商标。

Sun、Sun Microsystems、Sun 徽标、docs.sun.com、AnswerBook、AnswerBook2、Java、NetBeans 和 Solaris 是 Sun Microsystems, Inc. 在美国和其他国家/地区的商标或注册商标。所有的 SPARC 商标的使用均已获得许可，它们是 SPARC International, Inc. 在美国和其他国家/地区的商标或注册商标。标有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。

OPEN LOOK 和 Sun™ 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。Sun 感谢 Xerox 在研究和开发可视或图形用户界面的概念方面为计算机行业所做的开拓性贡献。Sun 已从 Xerox 获得了对 Xerox 图形用户界面的非独占性许可证，该许可证还适用于实现 OPEN LOOK GUI 和在其他方面遵守 Sun 书面许可协议的 Sun 许可证持有者。

美国政府权利 — 商业软件。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（Federal Acquisition Regulations，即“联邦政府采购法规”）的适用条款及其补充条款。

本文档按“原样”提供，对于所有明示或默示的条件、陈述和担保，包括对适销性、适用性或非侵权性的默示保证，均不承担任何责任，除非此免责声明的适用范围在法律上无效。



051214@13215



目录

前言 7

1 配置和设置 Service Registry 15

配置 Service Registry 15

▼ 在稍后配置安装后使用自定义属性配置 Service Registry 16

▼ 使用管理工具 17

管理用于 Service Registry 的 Application Server 域 18

▼ 使用 Application Server 管理控制台 19

▼ 停止并重新启动用于 Registry 的 Application Server 域 19

▼ 将根证书添加到 Registry 域的信任证书中 20

为非默认的 Application Server 安装配置 Service Registry 21

▼ 编辑 install.properties 文件 21

为非默认的 Service Registry 安装配置 Service Registry 21

▼ 编辑 install.properties 文件 21

允许访问外部 Web 站点 22

▼ 指定 Web 代理服务器 22

创建管理员 22

▼ 创建管理员 23

配置 Web 控制台 24

添加预定义查询 24

▼ 添加预定义查询 24

更改默认查询 26

▼ 更改默认查询 26

隐藏分类方案 26

▼ 隐藏分类方案 27

配置搜索结果显示 27

▼ 配置搜索结果显示中的行数	27
▼ 配置搜索结果区域中的列	28
重新安装 Service Registry	29
▼ 停止并删除 Registry 的 Application Server 域	29
▼ 重新安装 Service Registry 数据库	30
备份和恢复数据库	30
▼ 查找 Apache Derby 文档	30
2 使用管理工具	31
关于管理工具	31
启动管理工具	31
批处理模式	32
交互式模式	32
管理工具命令行选项	32
管理工具功能	34
权限	34
显示异常	34
识别 Registry 对象	35
语言环境对指定名称的影响	35
区分大小写	35
使用管理工具命令	36
add association	36
add user	37
cd	43
chown	44
cp	44
echo	46
help	46
lcd	47
ls	48
pwd	49
quit	49
rm	50
select	51
set	52
show	53
users	54

前言

《Service Registry 3 2005Q4 管理指南》介绍了如何在安装后配置 Service Registry ("Registry") 以及如何使用 Registry 附带的管理工具。本书还介绍了诸如备份和恢复 Registry 数据库之类的其他管理任务。

目标读者

管理指南适用于需要安装、卸载和管理 Registry 以及要为 Registry 创建大量内容而非使用 Web 控制台创建内容的用户。

读者应熟悉操作系统 (Solaris™ 或 Linux 操作系统) 中的 UNIX® shell 命令环境的基本知识。

阅读本书之前

在阅读本书之前，您必须按照《Sun Java Enterprise System 2005Q4 Installation Guide for UNIX》中所述安装 Registry。

Service Registry 可以作为 Java Web Services Developer Pack (<http://java.sun.com/webservices/jwsdp/>) 的一部分或 Sun Java Enterprise System 的一个组件获得，它是支持分布至整个网络或 Internet 环境的企业应用程序的软件基础结构。如果将 Service Registry 作为 Java Enterprise System 的组件购买，则必须熟悉 <http://docs.sun.com/coll/1286.1> 和 <http://docs.sun.com/coll/1382.1> 中的系统文档。

某些管理任务需要您熟悉以下规范的基本概念：

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

您可以通过转至 OASIS Web 站点 (<http://www.oasis-open.org/>)和按照指向 ebXML RIM V3.0 和 ebXML RS V3.0 的链接查找这些规范的最新发布版本。

本书的结构

本书包括以下内容：

第 1 章介绍了如何在安装后配置 Service Registry 以及如何执行其他管理任务。

第 2 章介绍了管理工具的使用。

Service Registry 文档集

可在 <http://docs.sun.com/app/docs/coll/1314.1> 和 <http://docs.sun.com/app/docs/coll/1383.1> 中获得 Service Registry 文档集。要了解 Service Registry，请参阅下表中列出的书。

表 P-1 Service Registry 文档

文档标题	内容
《Service Registry 3 2005Q4 发行说明》	该文档包含有关 Service Registry 的最新信息（包括已知问题）。
《Service Registry 3 2005Q4 管理指南》	该文档介绍如何在安装后配置 Service Registry 以及如何使用 Registry 附带的管理工具。还介绍了如何执行其他管理任务。
《Service Registry 3 2005Q4 用户指南》	该文档介绍如何使用 Service Registry Web 控制台搜索 Service Registry 和向其发布数据。
《Service Registry 3 2005Q4 Developer's Guide》	该文档介绍如何使用 Java API for XML Registries (JAXR) 搜索 Service Registry 和向其发布数据。

相关书籍

安装 Service Registry 时，它将被部署到 Sun Java System Application Server。有关管理 Application Server 的信息，请参阅《Sun Java System Application Server Enterprise Edition 8.1 2005Q2 Administration Guide》。

Java ES 文档集介绍了如何进行部署规划和系统安装。系统文档的 URL 为 <http://docs.sun.com/coll/1286.1> 和 <http://docs.sun.com/coll/1382.1>。有关 Java ES 的说明，请按照下表中列出的顺序参阅各本书。

表 P-2 Java Enterprise System 文档

文档标题	内容
《Sun Java Enterprise System 2005Q4 Release Notes》	该文档包含有关 Java ES 的最新信息（包括已知问题）。此外，各组件还有各自的发行说明。
《Sun Java Enterprise System 2005Q4 Documentation Roadmap》	提供与 Java ES 相关（作为系统或单独的组件）的所有文档的说明。
《Sun Java Enterprise System 2005Q4 Technical Overview》	介绍了 Java ES 的技术和概念基础。介绍组件、体系结构、过程和功能。
《Sun Java Enterprise System 2005Q4 Deployment Planning Guide》	介绍如何规划和设计基于 Java ES 的企业部署解决方案。介绍部署规划和设计的基本概念和原理，讨论解决方案的生命周期，并提供规划基于 Java ES 的解决方案时可以使用的高级示例和策略。
《Sun Java Enterprise System 2005Q4 Installation Planning Guide》	帮助您开发 Java ES 部署中硬件、操作系统和网络方面的执行规范。介绍诸如在安装和配置规划中组件与地址之间的依赖性问题。
《Sun Java Enterprise System 2005Q4 Installation Guide for UNIX》	指导您完成在 Solaris 操作系统或 Linux 操作系统上安装 Java ES 的过程。还显示了安装后如何配置组件以及验证它们是否运行正常。
《Sun Java Enterprise System 2005Q4 Installation Reference》	给予了有关配置参数的详细信息，提供了配置规划中使用的工作单并列出了相关材料（例如默认路径和文件号码）。
《Sun Java Enterprise System 2005Q1 Deployment Example Series: Evaluation Scenario》	介绍如何在一个系统上安装 Java ES，如何建立一组核心的共享网络服务，以及如何设置可以访问已建立的服务的用户帐户。
《Sun Java Enterprise System 2005Q4 升级指南》	提供了在 Solaris 操作系统或 Linux 操作环境中升级 Java ES 的说明。

表 P-2 Java Enterprise System 文档 (续)

文档标题	内容
《Sun Java Enterprise System Glossary》	定义 Java ES 文档中使用的术语。

有关 Java ES 及其组件的所有文档的 URL 为
<http://docs.sun.com/db/prod/entsys.05q4> 和
<http://docs.sun.com/db/prod/entsys.05q4?l=zh>。

默认的路径和文件名

下表介绍了本书中使用的默认路径和文件名。

表 P-3 默认的路径和文件名

占位符	说明	默认值
<i>ServiceRegistry-base</i>	表示 Service Registry 的基本安装目录。	Solaris 系统 : /opt/SUNWsoar Linux 系统 : /opt/sun/SUNWsoar
<i>RegistryDomain-base</i>	表示 Service Registry 的 Application Server 域和 Service Registry 数据库所在的目录。	Solaris 系统 : /var/opt/SUNWsoar Linux 系统 : /var/opt/sun/SUNWsoar

印刷约定

下表描述了本书中使用的印刷约定。

表 P-4 印刷约定

字体*	含义	示例
AaBbCc123	命令、文件和目录的名称；计算机屏幕输出	编辑 .login 文件。 使用 ls -a 列出所有文件。 machine_name% you have mail.
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	machine_name% su Password:
<i>AaBbCc123</i>	保留未译的新词或术语以及要强调的词。要使用实名或值替换的变量。 (注：在联机状态下，有些需要强调的词以黑体显示。)	要删除文件，请键入 rm <i>filename</i> 。
新术语强调	新词或术语以及要强调的词。	高速缓存 是存储在本地的副本。 请勿保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

* 浏览器的设置可能会与这些设置有所不同。

命令中的 shell 提示符示例

下表列出了默认系统提示符和超级用户提示符。

表 P-5 Shell 提示符

Shell	提示符
UNIX 和 Linux 系统上的 C shell	machine_name%
UNIX 和 Linux 系统上的 C shell 超级用户	machine_name#
UNIX 和 Linux 系统上的 Bourne shell 和 Korn shell	\$
UNIX 和 Linux 系统上的 Bourne shell 和 Korn shell 超级用户	#
Microsoft Windows 命令行	C:\

符号约定

下表介绍了本书中可能使用的符号。

表 P-6 符号约定

符号	说明	示例	含义
[]	包含可选参数和命令选项。	ls [-l]	-l 选项不是必需的。
{ }	包含必需命令选项的一组选项。	-d {y n}	-d 选项需要使用参数 y 或参数 n。
\${ }	表示变量引用。	\${com.sun.javaRoot}	引用 com.sun.javaRoot 变量的值。
-	结合同时发生的多个击键。	Control-A	按 A 键的同时按 Ctrl 键。
+	结合相继发生的多个击键。	Ctrl+A+N	按下 Ctrl 键后放开，然后再按后几个键。
→	表示图形用户界面中的菜单项选择。	“文件”→“新建”→“模板”	从“文件”菜单中选择“新建”。 从“新建”子菜单中选择“模板”。

联机访问 Sun 资源

通过 docs.sun.comSM Web 站点，您可以联机访问 Sun 技术文档。您可以浏览 docs.sun.com 中的归档文件或搜索特定的书名或主题。这些书是以联机文件的形式提供的，有 PDF 和 HTML 两种格式。残障人士用户可以通过辅助技术读取这两种格式。

要访问以下 Sun 资源，请转至 <http://www.sun.com>：

- 下载 Sun 产品
- 服务和解决方案
- 支持（包括修补程序和更新）
- 培训
- 研究
- 团体（例如，Sun 开发者网络）

第三方 Web 站点引用

本文档引用了第三方 URL 以提供其他相关信息。

注 – Sun 对本文档中提到的第三方 Web 站点的可用性不承担任何责任。对于此类站点或资源中的（或通过它们获得的）任何内容、广告、产品或其他资料，Sun 并不表示认可，也不承担任何责任。对于因使用或依靠此类站点或资源中的（或通过它们获得的）任何内容、产品或服务而造成的、名义上造成的或连带产生的实际或名义损坏或损失，Sun 概不负责，也不承担任何责任。

Sun 欢迎您提出意见

Sun 致力于提高其文档的质量，并十分乐意收到您的意见和建议。为了共享您的意见，请访问 <http://docs.sun.com>，并单击 "Send Comments"（发送意见）。在联机表单中，请提供完整的文档标题和文件号码。文件号码是一个七位或九位的数字，可以在书的标题页或文档的 URL 中找到。例如，本书的文件号码为 819-3573。在您提出意见时，可能需要在表单中输入英文版书名和文件号码，本书的英文版文件号码和书名为：819-2684，《Service Registry 3 2005Q4 Administration Guide》。

第 1 章

配置和设置 Service Registry

本章介绍了如何在安装后配置 Service Registry 以及如何执行其他管理任务。

本章包括以下内容：

- 第 15 页中的 “配置 Service Registry”
- 第 22 页中的 “允许访问外部 Web 站点”
- 第 22 页中的 “创建管理员”
- 第 24 页中的 “配置 Web 控制台”
- 第 29 页中的 “重新安装 Service Registry”
- 第 30 页中的 “备份和恢复数据库”

配置 Service Registry

《Sun Java Enterprise System 2005Q4 Installation Guide for UNIX》介绍了如何使用 Registry 的默认属性设置来执行 Service Registry 的安装后配置。要使用自定义属性设置，请在执行配置之前编辑文件

`ServiceRegistry-base/install/install.properties`。

在 Solaris 操作环境中，`ServiceRegistry-base` 位于 `/opt/SUNWsoar`，在 Linux 系统中，`ServiceRegistry-base` 位于 `/opt/sun/SUNWsoar`。

注 – 在配置 Service Registry 之前，您必须先安装和配置 Sun Java System Application Server (“Application Server”)。Service Registry 的配置过程将把 Registry 安装到 Application Server 域中。

建议您将 Application Server 安装到默认位置。如果将 Application Server 安装到了非默认位置，请在配置 Service Registry 之前按照第 21 页中的 “为非默认的 Application Server 安装配置 Service Registry” 中的说明进行操作。

install.properties 文件包含了一组可修改的属性设置。表 1-1 中列出的属性用于安装进程。每个属性名都有 registry.install. 前缀（以句点结尾）。这些属性中的大多数都针对为 Registry 创建的 Application Server 域设置了非默认端口。

表 1-1 Service Registry 安装属性

属性名	说明	默认属性值
DomainName	Application Server 域名	registry
ServerInstancePort	Application Server HTTP 端口	6060
ServerInstanceSecurePort	Application Server HTTPS 端口	6443
ServerJMSPort	Application Server Message Queue 端口	6484
ServerIIOPPort	Application Server IIOP 端口	6485
ServerIIOPSecurePort	Application Server IIOP 安全端口	6486
ServerIIOPMutualAuthPort	Application Server IIOP 相互验证端口	6487
AdministrationJMXPort	Application Server JMX 端口	6488
AdministrationPort	Application Server 管理服务端口	6489
AdministratorUserID	用于访问 Application Server 管理服务器的用户名	admin
AdministratorPassword	用于访问 Application Server 管理服务器的密码	12345678
ApplicationServerKeystorePassword	用于访问 Application Server 密钥库的密码	12345678
RegistryServerKeystorePassword	用于访问 Service Registry 密钥库的密码	12345678

▼ 在稍后配置安装后使用自定义属性配置 Service Registry

开始之前 要配置 Registry，您必须以 root 身份登录或成为超级用户。

步骤 1. 更改到 `ServiceRegistry-base/install` 目录。

2. 编辑文件 `install.properties` 中的可修改属性。

为了安全起见，建议不要编辑此文件以更改密码值。而在命令行中指定这些值。

3. 编辑此文件后，运行以下命令（全都在一行）：

对于 Solaris: `/usr/sfw/bin/ant -f build-install.xml install`

对于 Linux: `/opt/sun/bin/ant --noconfig -f build-install.xml install`

ant 命令要求设置 `JAVA_HOME` 环境变量。通常可将此环境变量设置为以下值：

`/usr/jdk/entsys-j2se`

要在命令行上指定已更改的密码，请将以下选项指定到命令中（全都在一行）：

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

Registry 配置进程将在

`RegistryDomain-base/domains/${registry.install.DomainName}` 创建一个 Application Server 域。默认域名为 `registry`。然后，配置进程将启动该域，部署 Registry 并使该域一直运行。

Registry 配置进程将在 `RegistryDomain-base/3.0` 目录中安装 Registry 数据库和服务密钥库。卸载 Registry 时不会删除此目录，这样可以保留数据库以供将来的版本使用。管理员具有何时及是否删除此目录的控制权。

在 Solaris 操作环境中，`RegistryDomain-base` 位于 `/var/opt/SUNWsoar`，在 Linux 系统中，`RegistryDomain-base` 位于 `/var/opt/sun/SUNWsoar`。

4. 查看 `ant install` 命令的输出以查看有无任何错误。

如果没有错误，您现在就可以开始使用 Web 控制台或管理工具。

▼ 使用管理工具

要执行限于管理员用户才能执行的管理工具任务，需要将一个 JAR 文件添加到管理工具清单类路径中以解决一个错误。

步骤 1. 确保仍在 `ServiceRegistry-base/install` 目录中。

2. 将文件 `soapprocessor.jar` 从已部署的 Registry 复制到 Registry 的 `lib` 目录中。执行以下命令（全都在一行）：

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

3. 运行 `admin.jar.manifest.fix` 安装目标，如下所示：

```
对于 Solaris: /usr/sfw/bin/ant -f build-install.xml  
admin.jar.manifest.fix
```

```
对于 Linux: /opt/sun/bin/ant --noconfig -f build-install.xml  
admin.jar.manifest.fix
```

管理用于 Service Registry 的 Application Server 域

默认情况下，Service Registry 的配置进程将创建一个名为 `registry` 的 Application Server 域。Service Registry Web 应用程序将部署到此域中。此域位于 `RegistryDomain-base/domains/registry` 目录中。

此位置不同于 Application Server 域的默认位置，
`/var/opt/SUNWappserver/domains` (Solaris) 或
`/var/opt/sun/appserver/domains` (Linux)。

要管理 `registry` 域，您可以使用 Application Server 管理控制台（“管理控制台”）。您可以使用管理控制台来启动和停止域、查看服务器日志和执行其他管理任务。有关详细信息，请参见第 19 页中的“使用 Application Server 管理控制台”。

您还可以直接查看服务器日志。该日志位于文件
`RegistryDomain-base/domains/registry/logs/server.log` 中。

您还可以使用 `asadmin` 命令来管理 `registry` 域。由于域不在默认位置，您在使用提供 `--domaindir` 选项的 `asadmin` 命令时必须指定该选项。

`registry` 域的密码文件为 `ServiceRegistry-base/pw.txt`。当您管理 `registry` 域时，将此密码文件作为参数指定给 `asadmin` 命令的 `--passwordfile` 选项。

`registry` 域使用一组非默认端口，以免与默认的 Application Server 域 `domain1` 发生冲突。表 1-2 列出了这些端口并对其进行了说明。有关更多信息，请参见《Sun Java System Application Server Enterprise Edition 8.1 2005Q2 Administration Guide》中的“Ports in the Application Server”。

表 1-2 Service Registry 域的默认端口

端口值	说明
6060	HTTP 端口
6443	基于 SSL 的 HTTPS
6484	Message Queue 端口
6485	IOP 端口
6486	IOP SSL 端口

表 1-2 Service Registry 域的默认端口 (续)

端口值	说明
6487	IIOP 相互验证端口
6488	JMX 端口
6489	Application Server 域管理端口

▼ 使用 Application Server 管理控制台

- 步骤
1. 在 Web 浏览器中，转至 URL `https://hostname:6489/`。
hostname 是运行 Application Server 和 Service Registry 的系统。
 2. 接受提供的证书。
系统将显示登录页面
 3. 在登录页面的“用户名”字段中键入 `admin`。
 4. 在“密码”字段中键入 Application Server 管理员密码。使用在配置 Registry 时为 `AdministratorPassword` 属性指定的值。默认值为 `12345678`。
 5. 单击“登录”。

另请参见 有关使用管理控制台的详细信息，请参阅管理控制台的联机帮助或《Sun Java System Application Server Enterprise Edition 8.1 2005Q2 Administration Guide》。

要更改 Service Registry 的日志记录级别，请按照《Sun Java System Application Server Enterprise Edition 8.1 2005Q2 Administration Guide》中的“To configure log levels”的说明进行操作。要在“附加属性”区域中指定的属性为 `org.apache.commons.logging.simplelog.log.org.freebxml.omar`。

▼ 停止并重新启动用于 Registry 的 Application Server 域

Registry 的配置进程将启动 Registry 部署在其中的 Application Server 域。当您执行了某些管理任务后，必须停止域，然后再重新启动。第 22 页中的“允许访问外部 Web 站点”和第 22 页中的“创建管理员”举例说明了这些任务。

管理控制台将通知您是否需要重新启动域。您可以使用管理控制台来执行此任务。如果使用的是 `asadmin` 命令，您可以使用 Ant 任务停止和启动域。

- 步骤
1. 更改到 Service Registry 安装目录。
`cd ServiceRegistry-base/install`
 2. 运行以下命令（全都在一行）：
`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
```

此目标将停止域，再将其重新启动。

build-install.xml 文件还包含用于停止和启动 Registry 域的单独的 Ant 目标。要停止域，请使用 Ant 目标 appserver.domain.stop。要启动域，请使用 Ant 目标 appserver.domain.start。

▼ 将根证书添加到 Registry 域的信任证书中

此任务扩展了 Application Server registry 域中的信任证书列表。

仅当您使用第三方证书并且针对第三方的根证书授权机构 (Certificate Authority, CA) 证书尚未存在于 Application Server 信任存储中时，才执行此任务。如果您仅使用 registry 发布的证书，请勿执行此任务。

步骤 1. 下载您希望支持的所有根证书。以下站点提供根证书：

- <http://www.entrust.net/developer/>
- http://www.geotrust.com/resources/root_certificates/
- <http://www.thawte.com/roots/>
- <http://www.verisign.com/support/roots.html>

2. 如果有必要，请使用 `unzip` 命令从下载的归档文件中解压缩 `.cer` 文件。

注 - 一些文件有 `.der` 后缀。

3. 将 `.cer` 文件复制到目录 `ServiceRegistry-base/install/cacerts` 中。

4. 更改到目录 `ServiceRegistry-base/install`。

5. 运行以下命令（全都在一行）：

```
Solaris: /usr/sfw/bin/ant -f build-install.xml install.cacerts
```

```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml
install.cacerts
```

此命令将把在目录 `ServiceRegistry-base/install/cacerts` 中找到的所有证书都安装到 Application Server 域信任存储中。

您可以使用 `list.cacerts` 目标以确保证书已被正确安装。

6. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。

为非默认的 Application Server 安装配置 Service Registry

在 Solaris 操作环境中，Application Server 的默认安装位置为 `/opt/SUNWappserver/appserver`，在 Linux 系统中，Application Server 的默认安装位置为 `/opt/sun/appserver`。如果您在其他位置安装了 Application Server，则必须在配置 Service Registry 之前编辑文件 `install.properties`。

▼ 编辑 `install.properties` 文件

- 步骤
1. 在 `ServiceRegistry-base/install` 目录中，用文本编辑器打开文件 `install.properties`。
 2. 查找被注释掉的 `appserver.root.dir` 属性的定义。
 3. 删除注释字符 (#) 并用 Application Server 的实际位置替换属性定义。
 4. 保存并关闭 `install.properties` 文件。

接下来的操作 按照第 15 页中的“配置 Service Registry”中的说明继续进行操作。

为非默认的 Service Registry 安装配置 Service Registry

在 Solaris 操作环境中，Service Registry 的默认安装位置为 `/opt/SUNWsoar`，在 Linux 系统中，Service Registry 的默认安装位置为 `/opt/sun/SUNWsoar`。如果您在其他位置安装了 Service Registry，则必须在配置 Service Registry 之前编辑文件 `install.properties`。

▼ 编辑 `install.properties` 文件

- 步骤
1. 在 `ServiceRegistry-base/install` 目录中，用文本编辑器打开文件 `install.properties`。
 2. 查找被注释掉的 `soar.sdk.home` 和 `soar.server.home` 属性的定义。
 3. 对于每个属性，删除注释字符 (#) 并用 Service Registry 的实际位置替换属性定义。
 4. 保存并关闭 `install.properties` 文件。

接下来的操作 按照第 15 页中的“配置 Service Registry”中的说明继续进行操作。

允许访问外部 Web 站点

任何 Registry 对象都可以有一个 ExternalLink 对象，该对象用于指定与该 Registry 对象相关的 URL。为使用户能够创建 ExternalLink 对象，Service Registry 必须能够验证 URL，而执行这项任务需要访问外部 Web 站点。如果 Registry 被部署到防火墙之后，则需要设置允许此类访问的代理配置。

代理配置要求指定 Web 代理主机和端口作为部署了 Service Registry 的 Application Server 实例的 Java 虚拟机 (Java Virtual Machine, JVM) 选项。

▼ 指定 Web 代理服务器

- 步骤
1. 按照第 19 页中的“使用 Application Server 管理控制台”中所述登录到 Application Server 管理控制台。
 2. 展开“配置”节点。
 3. 展开服务器节点 `server-config (Admin Config)`。
 4. 单击“JVM 设置”。
 5. 单击“JVM 选项”选项卡。
 6. 单击“添加选项”。
 7. 在文本字段中键入以下内容（全都在一行）：

```
-Dhttp.proxyHost=hostname.domainname -Dhttp.proxyPort=8080
```

通常，端口值为 8080。如果使用不同的端口，请指定正确的值。
 8. 单击“保存”。
 9. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。

创建管理员

Service Registry 管理工具的某些任务只有注册为管理员的用户才可以执行。此外，可能需要管理员对其他用户提交的对象执行生命周期更改（例如，批准）。

管理员还可以更改默认的访问控制策略 (Access Control Policy, ACP)。不过, 撰写 ACP 目前还是一个手动过程, 需要具备 OASIS 可扩展访问控制标记语言 (eXtensible Access Control Markup Language, XACML) 的知识。有关详细信息, 请参阅 ebXML RIM 3.0 的第 9 章“访问控制信息模型”, 特别是第 9.7.6 节到第 9.7.8 节中的示例。有关如何查找 ebXML RIM 3.0 规范的信息, 请参见第 7 页中的“阅读本书之前”。

▼ 创建管理员

要注册为管理员, 请执行以下步骤。

步骤 1. 按照《Service Registry 3 2005Q4 用户指南》中的“创建用户帐户”中所述执行用户注册操作。

记住所下载的证书的路径名。证书文件的默认名为 `generated-key.p12`。

2. 获取 User 对象的唯一标识符, 如下所示:

a. 使用 Web 控制台执行基本查询, 并将“对象类型”设置为“用户”。

b. 单击“详细信息”链接以查看 Registry 为您创建的 User 对象。

c. 请将“唯一标识符”字段值记录下来。

3. 将证书复制到主目录中的以下位置 (根据所需创建目录):

```
$HOME/soar/3.0/jaxr-ebxml/security
```

4. 更改到目录

```
RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/cla
```

5. 在文本编辑器中打开文件 `omar.properties`。

6. 查找 `omar.security.authorization.registryAdministrators` 属性的定义。

7. 通过在逻辑标识符字符串 (在步骤 2 中做了记录) 前面添加垂直条 (|) 编辑属性定义。

属性定义必须全都在一行并且不能包含空格。完成后, 将类似于以下所示 (全都在一行):

```
omar.security.authorization.registryAdministrators=  
urn:freebxml:registry:predefinedusers:registryoperator|  
urn:uuid:77f5c196-79de-4286-8483-8d80def3583b
```

8. 保存并关闭 `omar.properties` 文件。

9. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。

接下来的操作 要创建其他管理员, 不必编辑 `omar.properties` 文件。如果您是管理员, 将可以使用管理工具或 Web 控制台来添加用户, 并可以使用 Web 控制台将用户分类为管理员。

配置 Web 控制台

作为管理员，您可以通过编辑配置文件对 Web 控制台显示的某些方面进行自定义。本节介绍了以下任务：

- 第 24 页中的“添加预定义查询”
- 第 26 页中的“更改默认查询”
- 第 26 页中的“隐藏分类方案”
- 第 27 页中的“配置搜索结果显示”

有关使用 Web 控制台的信息，请参见《Service Registry 3 2005Q4 用户指南》。

添加预定义查询

Service Registry 包括几个显示在“选择预定义查询”下拉式列表的“Web 控制台搜索”表单中的预定义查询。作为管理员，您可以根据具体的 Registry 安装向下拉式列表中添加新的查询。

▼ 添加预定义查询

- 步骤 1. 使用 Web 控制台将 AdhocQuery 对象发布到 Registry 中。**
- 为查询指定的名称和说明将显示在预定义查询的下拉式列表中。在查询的 SQL 语句中，通过将用户提供的数据用双重单引号引起来指定此类数据的占位符，如下所示：

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

2. 将 AdhocQuery 对象的唯一标识符和 SQL 语句中所有占位符记录下来。
3. 更改到目录 *RegistryDomain-base/3.0/jaxr-ebxml*。
4. 用文本编辑器打开文件 **registry-browser-config.xml**。
5. 使用以下格式将条目添加至 **registry-browser-config.xml** 文件。为 SQL 语句中的每个占位符指定 **Parameter** 元素。

```
<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 是 *AdhocQuery* 对象的唯一标识符。

每个参数的 *parameterName* 属性值必须来自查询的 SQL 语句的占位符。

datatype 属性可以拥有以下任一值：

- *string*：该参数显示为“搜索”表单中的文本字段。
- *taxonomyElement*：该参数显示为“搜索”表单中的下拉式列表。如果指定 *taxonomyElement* 数据类型，*Name* 和 *Description* 元素必须后跟 *SlotList* 元素，如下所示：

```

<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 是分类方案或概念的唯一标识符，它们的概念（或子概念）必须显示在下拉式列表中。如果分类方案尚未存在于 *Registry* 中，您必须将其发布。

插槽名称必须为 "domain"。

- *boolean*：该参数显示为“搜索”表单中的复选框。

如果 *datatype* 为 *string* 或 *boolean*，您还可以将 *defaultValue* 属性添加到 *Parameter* 元素以指定要在“搜索”表单中显示的默认值。

为您支持的所有语言环境指定每个参数名称和说明的本地化字符串值。当前语言环境中的 *parameter_name* 在“搜索”表单中显示为参数的标签。

使用 *registry-browser-config.xml* 文件中的现有条目作为参考。

6. 保存并关闭 *registry-browser-config.xml* 文件。
7. 按照第 19 页中的“停止并重新启动用于 *Registry* 的 *Application Server* 域”中的说明进行操作。

更改默认查询

在“选择预定义查询”下拉式列表中显示为默认值的查询为基本查询，该查询允许用户按名称、说明和分类搜索 Registry 对象。

作为管理员，您可以将此默认值更改为适用于您的安装的查询。例如，您可能要将默认查询设为添加到 Registry 中的新预定义查询（如第 24 页中的“添加预定义查询”中所述）。要进行此更改，请在配置文件中编辑属性。

▼ 更改默认查询

步骤 1. 更改到目录

```
RegistryDomain-base/domains/registry/applications/j2ee-modules  
/soar/WEB-INF/classes。
```

2. 用文本编辑器打开文件 `jaxr-ebxml.properties`。

3. 查找属性 `jaxr-ebxml.thin.defaultQueryPanel` 的定义。默认情况下，此属性被注释掉：

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

4. 删除注释字符 (#)。

5. 通过指定将为默认值的查询的逻辑标识符设置属性值，如以下示例所示：

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

6. 保存并关闭 `jaxr-ebxml.properties` 文件。

7. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。

隐藏分类方案

分类方案的树结构在 Web 控制台的以下区域显示：

- 选定“基本查询”后，显示在“搜索”表单区域
- “资源管理器”菜单区域
- 需要为某些类型的 Registry 对象选择概念时显示的“ClassificationScheme/概念选择器”窗口

作为管理员，如果您不想将分类方案提供给 Service Registry 用户，则可以在视图中隐藏分类方案。要隐藏分类方案，请在配置文件中定义一个属性。

▼ 隐藏分类方案

- 步骤 1. 更改到目录
`RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes。`
2. 用文本编辑器打开文件 `jaxr-ebxml.properties。`
3. 使用以下语法设置属性
`jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList。`
所有属性定义必须都在一行上并且不能包含空格。
- ```
jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList=
class_scheme_id1|class_scheme_id2|...
```
- 指定每个要隐藏的分类方案的逻辑标识符。如果指定多个标识符，请使用垂直条 (|) 分隔这些标识符，如以下示例所示：
- ```
jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList=  
urn:oasis:names:tc:ebxml-regrep:classificationScheme:StatusType|  
urn:oasis:names:tc:ebxml-regrep:profile:ws:classificationScheme:BindingType
```
4. 保存并关闭 `jaxr-ebxml.properties` 文件。
5. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。

配置搜索结果显示

默认情况下，Web 控制台一次显示每个查询的 10 个搜索结果。如果返回的搜索结果超过 10 个，用户可以显示结果的附加页。作为管理员，您可以修改每个页面上显示的搜索结果数。

默认情况下，Web 控制台显示搜索结果区域中的特定列。将显示每个对象的对象类型、名称、说明、版本和版本注释。对于某些对象类型将配置非默认显示。例如，对于 `ServiceBinding` 对象，其显示包括端点而不是版本信息。作为管理员，您可以添加配置信息以显示您选择的对象类型的非默认数据。

要执行这些任务，请编辑配置文件。

▼ 配置搜索结果显示中的行数

- 步骤 1. 更改到目录
`RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes。`
2. 用文本编辑器打开文件 `jaxr-ebxml.properties。`

3. 查找属性 `omar.client.thinbrowser.numSearchResults` 的定义：

```
omar.client.thinbrowser.numSearchResults=10
```
4. 将值 10 更改为您所希望的值。
5. 保存并关闭 `jaxr-ebxml.properties` 文件。
6. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。

▼ 配置搜索结果区域中的列

- 步骤
1. 更改到目录 `RegistryDomain-base/3.0/jaxr-ebxml`。
 2. 用文本编辑器打开文件 `registry-browser-config.xml`。
 3. 使用以下格式将条目添加至 `registry-browser-config.xml` 文件。
此示例配置 Service 对象的非默认显示。

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

`registry-browser-config.xml` 提供 `ObjectTypeConfig` 元素的语法。将文件中已有的元素用作示例。这些元素为 `Registry` 对象配置默认显示，并为 `ExternalLink`、`ExtrinsicObject` 和 `ServiceBinding` 对象配置非默认显示。

大多数情况下，您可以从 ebXML Registry 信息模型版本 3.0 规范的类属性中推出方法名称（有关详细信息，请参见第 7 页中的“阅读本书之前”）。可以在 `RegistryObjectImpl` 实现类中找到 `getStatusAsString` 方法。（但是，此发行版的 Service Registry 不包括 API 文档。）

4. 保存并关闭 `registry-browser-config.xml` 文件。

5. 按照第 19 页中的“停止并重新启动用于 Registry 的 Application Server 域”中的说明进行操作。
6. 要验证重新配置，请使用 Web 控制台的“搜索”或“资源管理器”菜单来显示已更改其列的对象。

重新安装 Service Registry

如果需要卸载和重新安装 Service Registry，请在重新安装之前执行以下任务：

- 如果 Registry 数据库包含要保留的数据，请按照第 30 页中的“备份和恢复数据库”中所述备份该数据库。
- 停止 Registry 的 Application Server 域，然后删除该域。如果不删除该域，则对重新安装的 Registry 执行安装后配置时将失败。

如果需要重新安装 Service Registry 数据库（例如，如果数据库损坏），请按照第 30 页中的“重新安装 Service Registry 数据库”中的说明进行操作。重新安装数据库之前，您不需要将其卸载。

▼ 停止并删除 Registry 的 Application Server 域

步骤 1. 更改到 `ServiceRegistry-base/install` 目录。

2. 运行以下命令（全都在一行）：

```
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. 运行以下命令（全都在一行）：

```
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
```

▼ 重新安装 Service Registry 数据库

步骤 1. 更改到 *ServiceRegistry-base/install* 目录。

2. 运行以下命令（全都在一行）：

```
Solaris: /usr/sfw/bin/ant -f build-install.xml install.db
```

```
Linux: /opt/sun/bin/ant --noconfig -f build-install.xml  
install.db
```

备份和恢复数据库

Registry 使用 Apache Derby 数据库。默认情况下，该数据库位于目录 *RegistryDomain-base/3.0/data/registry/soar/* 中。

要了解如何备份和恢复数据库，请参阅 Apache Derby 文档。

▼ 查找 Apache Derby 文档

步骤 1. 在 Web 浏览器中，转至 Apache Derby Web 站点 (<http://db.apache.org/derby/>)。

2. 单击 "Manuals" 选项卡。

3. 单击 "10.0 Manuals"。

4. 找到 "Server & Admin Guide"。

5. 找到关于备份和恢复数据库的章节。

第 2 章

使用管理工具

本章介绍了如何使用 Service Registry 的管理工具（“管理工具”）。

本章包括以下内容：

- 第 31 页中的 “关于管理工具”
- 第 31 页中的 “启动管理工具”
- 第 34 页中的 “管理工具功能”
- 第 36 页中的 “使用管理工具命令”

关于管理工具

Service Registry 管理工具提供了一个简单的命令行界面，可用于执行公用管理任务（例如，向 Registry 中添加关联和从 Registry 中删除对象）。

该工具能够以两种模式运行：

- 在批处理模式中，可以在工具的命令行中指定一个或多个命令。
- 在交互式模式中，可以在工具的交互式 Shell 中输入命令。

某些命令（例如 `ls` 和 `rm`）模仿了众所周知的用于操作文件和文件夹的 UNIX® 命令的名称和行为。其他命令没有相应的等效 UNIX 命令。

启动管理工具

要启动管理工具，请执行 `admin-tool.jar` 文件：

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

在 Solaris 操作环境中, *ServiceRegistry-base* 位于 `/opt/SUNWsoar`, 在 Linux 系统中, *ServiceRegistry-base* 位于 `/opt/sun/SUNWsoar`。

忽略您启动工具时显示的警告; 这些警告并不重要。

要退出管理工具, 请使用 `quit` 命令。

要执行管理员才能执行的命令, 您需要在命令行中指定证书文件 (即密钥库) 的类型、位置和密码 (全都在一行), 如下所示:

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

其中 *filename* 为证书文件的根名称 (默认情况下为 `generated-key`), *password* 为您在创建证书时指定用于保护此文件的密码。`security/filename.p12` 是相对于 `$HOME/soar/3.0/jaxr-ebxml` 目录的位置。

要节省输入, 请创建一个脚本来执行这个命令。

批处理模式

要在批处理模式中运行管理工具, 请在启动管理工具时在命令行中指定 `-command` 选项。

例如, 以下命令可执行 `ls` 命令:

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

管理工具将把您的命令和工具的响应回显到屏幕上, 然后在执行完命令后退出。

确保针对您的 Shell 使用了正确的转义符。

交互式模式

要在交互式模式中运行管理工具, 请通过在命令行中指定除了 `-command` 选项以外的任意选项 (或不指定选项) 来启动管理工具 Shell:

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

管理工具将显示以下提示符并等待您输入命令:

```
admin>
```

管理工具命令行选项

管理工具可以识别第 33 页中的“一般形式”中列出且在第 33 页中的“选项”中说明的命令行选项。

一般形式

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

选项

-alias	访问密钥库中的用户证书时使用的别名。请指定您在注册为用户时使用的别名。
-command	在不从交互式 Shell 中获取命令的情况下运行的管理工具命令序列。使用分号 (;) 可以分隔多个命令。不必在 <i>commands</i> 中包括 quit 命令。如果需要不将分号用作命令分隔符，请在分号前加上一个反斜杠： \; 运行管理工具的 Shell 可能会需要使用另一个反斜杠来转义该反斜杠： \\; 如果命令中包含空格，请把整个命令序列用单引号或双引号引起来，以使管理工具将该序列作为一个命令行参数而不是多个命令行参数进行处理。如果您的 Shell 还将分号解释为 Shell 命令的分隔符，则必须始终把多个管理工具命令序列用引号引起来。
-create	如果有必要，则创建由 -root 选项指定的 RegistryPackage，并根据需要创建任意个 RegistryPackage 父对象。仅当运行管理工具的用户被授权创建对象时，此选项才有效。
-debug	输出在调试时会有用的附加信息。
-help	提供这些选项的列表。
-keypass	访问密钥库中的用户证书时使用的密码。请指定您在注册为用户时使用的密码。
-localdir	与本地文件系统中的文件相关的命令所使用的本地文件系统中的基本目录。
-locale	选择资源包时使用的语言环境（例如 en 或 fr），将使用该语言环境显示错误消息和状态消息。默认值由 Java 虚拟机 (Java Virtual Machine, JVM) 确定。
-registry	连接指向的 ebXML Registry 的 URL。默认 URL 为 http://localhost:6060/soar/registry/soap。
-root	RegistryPackage 的定位器（例如，/registry/userData），用作把系统信息库作为 RegistryPackage 对象的树（每个对象都包含其他 RegistryObject 和 RegistryPackage 对象）处理的命令的基

基础。默认值是为所有用户的数据定义的
RegistryPackage: /registry/userData。

-sqlselect 执行 *SQL_statement* 来选取 Registry 对象。该语句应该是以 *select* 为开头的完整的 SQL 语句。SQL 语句必须用引号引起来，但无需以分号结尾。如果指定此选项，然后使用不带参数的 *select* 命令，则命令将执行 *SQL_statement*，直到您使用带参数（而不是 *SQL_statement*）的 *select* 命令为止。

-v | -verbose 指定状态消息的详细输出。

注 --help 选项的输出将列出不受此版本支持的两个选项：-class 和 -property。

管理工具功能

此部分介绍了管理工具的以下功能：

- 第 34 页中的“权限”
- 第 34 页中的“显示异常”
- 第 35 页中的“识别 Registry 对象”
- 第 35 页中的“语言环境对指定名称的影响”
- 第 35 页中的“区分大小写”

权限

使用管理工具时，您只能执行在启动工具时指定其主别名和密码的用户可以执行的操作。只有具有系统管理员角色的用户才能执行特定的命令（例如 *chown*）。有关详细信息，请参见第 22 页中的“创建管理员”。

显示异常

使用管理工具可以使您避免在命令失败时查看较长的堆栈追踪。

当命令失败时，管理工具将打印堆栈追踪的第一行及以下消息：

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

如果需要更多信息，则接着执行 *show exception* 命令以查看完整的堆栈追踪。

`show exception` 命令用于始终显示紧接在前面的命令的堆栈追踪。

识别 Registry 对象

识别 Registry 对象的主要方法是通过名称来识别。但是，识别 RegistryPackage 对象通常是通过从 `registry` 根到 RegistryPackage 的路径来识别。例如，`/registry/userData` 是指向 `userData RegistryPackage` 的路径。

匹配某些名称时可以使用通配符。使用问号 (?) 来匹配单个字符。使用星号 (*) 来匹配零个或多个字符。

一些命令 (例如 `cd` 和 `chown`) 支持通过对象的统一资源名称 (Uniform Resource Name, URN) (必须以 `urn:` 为开头) 来对其进行识别。例如，`urn:uuid:2702f889-3ced-4d49-82d1-e4cd846cb9e4` 就是一个有效的 URN。

`chown` 命令还支持使用 `%number` 来引用上一个 `users` 命令列出的 User。

对于某些命令，您可以输入包含空格的名称，方法为把整个名称用双引号引起来，或在名称中的每个空格前加一个反斜杠。

`select` 命令支持使用 SQL 通配符：百分号 (%) 用于匹配多个字符，下划线 (_) 用于匹配单个字符。

语言环境对指定名称的影响

一个 RegistryObject (或 RegistryPackage) 可以有多个名称，每个名称又可与不同的语言环境相关联。

您指定的路径和对象名称仅是针对当前语言环境求得的结果。当您尝试按名称选择具有多个名称的 Registry 对象时，Registry 仅尝试根据 Registry 对象的一个名称 (其语言环境与当前语言环境的匹配程度最接近的选项) 来对您提供的名称进行匹配，而不根据 Registry 对象的所有多个名称进行匹配。

例如，假定当前 RegistryPackage 有一个成员对象具有两个名称，每个名称与不同的语言环境相关联：`en` (英语) 语言环境中的 `red` 和 `fr` (法语) 语言环境中的 `rouge`。在当前语言环境为 `en` 时，命令 `ls rouge` 不显示该成员对象，但在语言环境为 `fr` (或其变形之一) 时，则显示该成员对象。

区分大小写

管理工具识别的命令名称和字符参数不区分大小写。例如，`ls`、`Ls` 和 `LS` 是等效的。

赋值的选项将被逐字地传递给使用该选项的代码。

使用管理工具命令

下面几个部分将介绍可用的命令。对于每个命令，选项和操作数的一般形式和说明遵守以下印刷约定：

- *Italics* 表示一个选项参数或操作数，它应由运行命令时的实际值替代。
- 大括号 ({ }) 用于限定一组选项或操作数，您必须在命令中包括其中的一项。用垂直条 (|) 分隔选项或操作数。
- 方括号 ([]) 用于限定一个或一组选项或操作数，您可以省略其中的项目。
- 选项或操作数之后的省略号 (...) 表示可以重复使用该参数或操作数。

其他内容是运行命令时必须包括的字符文本。

add association

向 Registry 中添加 Association 对象。

一般形式

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

说明

add association 命令用于向 Registry 中添加指定类型的 Association 对象。您可以使用以下类型：

- AccessControlPolicyFor
- AffiliatedWith (拥有 EmployeeOf 和 MemberOf 子概念)
- Contains
- ContentManagementServiceFor
- EquivalentTo
- Extends
- ExternallyLinks
- HasFederationMember
- HasMember
- Implements

- InstanceOf
- InvocationControlFileFor (拥有 CatalogingControlFileFor 和 ValidationControlFileFor 子概念)
- OffersService
- OwnerOf
- RelatedTo
- Replaces
- ResponsibleFor
- SubmitterOf
- Supersedes
- Uses

选项

`-type` Association 对象的类型。

操作数

`sourceURN` 源对象的 URN。

`targetURN` 目标对象的 URN。

示例

以下命令（全都在一行）将在具有两个指定 URN 的对象之间创建 RelatedTo 关系。

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

add user

向 Registry 中添加用户。

一般形式

```
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]
```

说明

`add user` 命令用于添加 User 对象。通常，User 对象至少包含一个 PostalAddress、TelephoneNumber 和 EmailAddress 对象。

在命令行中或通过使用 `-load` 选项指定含有信息的 Java 属性文件来指定有关用户的信息。将按照信息选项和 `-load` 选项在命令行中显示的顺序求值。例如，您可以在命令行中指定一些属性，从某个属性文件中装入其他属性，然后用后面的命令行选项覆盖该属性文件中的信息。

您最多可以为一个新用户指定三组地址、电话号码和电子邮件地址。如果需要指定更多信息，您可以在以后使用 Web 控制台或 JAXR 来添加。

当您指定地址、电话号码或电子邮件地址时，必须提供针对其类型的值：例如，`-emailType OfficeEmail`。

您可以在命令行中使用简写选项（例如 `-fn`）以提供每个用户需要的一些通用信息。但是，当您在属性文件中提供信息时，必须使用选项的完整形式。例如，您可以在命令行中使用 `-email1.address`、`-emailAddress` 或 `-email` 指定用户的第一个电子邮件地址。但是，当您在属性文件中指定第一个电子邮件地址时，必须使用 `email1.address=`。由于用户的第二个电子邮件地址只有一个选项，因此在命令行中必须使用 `-email2.address`，在属性文件中必须使用 `email2.address=`。

如果指定了 `-edit` 选项，管理工具将启动编辑器，以使您可以编辑新用户的消息。有关详细信息，请参见选项说明。

注 – 使用 `-load` 装入或使用 `-edit` 编辑的属性文件使用 ISO-8859-1 字符集，就象所有 Java 属性文件一样。有关如何在属性文件中表示不属于 ISO-8859-1 的字符的详细信息，请参见 `java.util.Properties.load(InputStream)` 文档。

选项

`-edit`

使管理工具启动编辑器，以使您可以编辑新用户的消息。计算其他命令行参数之后，该工具将启动编辑器。因此，将启动编辑程序以便求得在命令行或属性文件中指定的所有信息的值。编辑程序必须在不出错的情况下终止，然后命令才能继续执行。管理工具将启动由 `set editor` 命令指定的编辑器（请参见第 52 页中的“`set`”）；默认情况下，此编辑器为 `vi` 编辑器。

注 – 在此版本中，`-edit` 可与 `emacsclient` 和 `NetBeans™` 命令 `bin/runide.sh --open` 结合使用（但效果不是很好），但还不能与 `vi` 结合使用。

`-load`

指定用于指定用户属性的 Java 属性文件。属性名称与 `add user` 命令选项的长型名称（例如 `lastName` 和 `post1.type`）相同。

`-fn` | `-firstName`

指定用户的名字。

`-ln` | `-lastName`

指定用户的姓氏。姓氏是必需的信息，必须在命令行或属性文件中指定姓氏。

`-mn` | `-middleName`

指定用户的中间名。

`-alias`

访问密钥库中的用户证书时使用的别名。此选项是必需的。别名的长度不能少于三个字符。

- keypass
访问密钥库中的用户证书时使用的密码。此选项是必需的。密码的长度不能少于六个字符。
- postalType | -post1.type
第一个 PostalAddress 的类型。类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串（例如 Office 或 Home）。
- city | -post1.city
第一个 PostalAddress 的城市。
- country | -post1.country
第一个 PostalAddress 的国家/地区。
- postalCode | -postcode | - zip | -post1.postalcode
第一个 PostalAddress 的邮政编码。
- stateOrProvince | -state | - province | -post1.stateOrProvince
第一个 PostalAddress 的省/自治区/直辖市。
- street | -post1.street
第一个 PostalAddress 的街道名称。
- streetNumber | -number | - -post1.streetNumber
第一个 PostalAddress 的门牌号码。
- post2.type
第二个 PostalAddress 的类型。如果指定了第二个 PostalAddress，则类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串（例如 Office 或 Home）。
- post2.city
第二个 PostalAddress 的城市。
- post2.country
第二个 PostalAddress 的国家/地区。
- post2.postalcode
第二个 PostalAddress 的邮政编码。
- post2.stateOrProvince
第二个 PostalAddress 的省/自治区/直辖市。
- post2.street
第二个 PostalAddress 的街道名称。
- post2.streetNumber
第二个 PostalAddress 的门牌号码。
- post3.type
第三个 PostalAddress 的类型。如果指定了第三个 PostalAddress，则类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串（例如 Office 或 Home）。
- post3.city
第三个 PostalAddress 的城市。

-post3.country
第三个 PostalAddress 的国家/地区。

-post3.postalcode
第三个 PostalAddress 的邮政编码。

-post3.stateOrProvince
第三个 PostalAddress 的省/自治区/直辖市。

-post3.street
第三个 PostalAddress 的街道名称。

-post3.streetNumber
第三个 PostalAddress 的门牌号码。

-phoneType | -telephone1.type
第一个 TelephoneNumber 的类型。类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串，但您可以指定下面的一种已知类型：
: Beeper、FAX、HomePhone、MobilePhone 或 OfficePhone。

-areaCode | -telephone1.areaCode
第一个 TelephoneNumber 的区号。

-countryCode | -telephone1.countryCode
第一个 TelephoneNumber 的国家/地区代码。

-extension | -telephone1.extension
第一个 TelephoneNumber 的分机。

-number | -telephone1.number
第一个 TelephoneNumber 的电话号码后缀（不包括国家/地区代码或区号）。该号码是必需的信息，必须在命令行或属性文件中指定该号码。

-URL | -telephone1.URL
第一个 TelephoneNumber 的 URL（可以在网上拨打此号码的 URL）。

-telephone2.type
第二个 TelephoneNumber 的类型。如果指定了第二个 TelephoneNumber，则类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串，但您可以指定下面的一种已知类型：
: Beeper、FAX、HomePhone、MobilePhone 或 OfficePhone。

-telephone2.areaCode
第二个 TelephoneNumber 的区号。

-telephone2.countryCode
第二个 TelephoneNumber 的国家/地区代码。

-telephone2.extension
第二个 TelephoneNumber 的分机。

-telephone2.number
第二个 TelephoneNumber 的电话号码后缀（不包括国家/地区代码或区号）。如果指定了第二个 TelephoneNumber，则该号码是必需的信息，必须在命令行或属性文件中指定该号码。

- telephone2.URL
第二个 TelephoneNumber 的 URL（可以在网上拨打此号码的 URL）。
- telephone3.type
第三个 TelephoneNumber 的类型。如果指定了第三个 TelephoneNumber，则类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串，但您可以指定下面的一种已知类型：Beeper、FAX、HomePhone、MobilePhone 或 OfficePhone。
- telephone3.areaCode
第三个 TelephoneNumber 的区号。
- telephone3.countryCode
第三个 TelephoneNumber 的国家/地区代码。
- telephone3.extension
第三个 TelephoneNumber 的分机。
- telephone3.number
第三个 TelephoneNumber 的电话号码后缀（不包括国家/地区代码或区号）。如果指定了第三个 TelephoneNumber，则该号码是必需的信息，必须在命令行或属性文件中指定该号码。
- telephone3.URL
第三个 TelephoneNumber 的 URL（可以在网上拨打此号码的 URL）。
- emailType | -email1.type
第一个 EmailAddress 的类型。类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串，但您可以指定下面的一种已知类型：HomeEmail 或 OfficeEmail。
- emailAddress | -email | -email1.address
第一个电子邮件地址。第一个电子邮件地址是必需的信息。
- email2.type
第二个 EmailAddress 的类型。如果指定了第二个 EmailAddress，则类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串，但您可以指定下面的一种已知类型：HomeEmail 或 OfficeEmail。
- email2.address
第二个电子邮件地址。
- email3.type
第三个 EmailAddress 的类型。如果指定了第三个 EmailAddress，则类型是必需的信息，必须在命令行或属性文件中指定类型。该值是一个任意字符串，但您可以指定下面的一种已知类型：HomeEmail 或 OfficeEmail。
- email3.address
第三个电子邮件地址。

示例

以下命令将从用户主目录的 JaneSmith.properties 文件中装入 User 属性。

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

以下命令（全都在一行）将指定创建 User 对象所必需的最少属性。

```
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

更改 RegistryPackage 的位置。

一般形式

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

说明

cd 命令将目录（广义的）更改到指定路径处或具有指定 URN 的 RegistryPackage。

当多个 RegistryPackage 对象以相同路径存在（对于当前语言环境）时，该命令将更改到指定的 URN。

操作数

locator 从系统信息库的根目录到系统信息库中某个对象的 Registry 对象名称的路径，每个名称前加上一个正斜杠 (/)。

例如，属于 registry RegistryPackage（其本身不是任何 RegistryPackage 的成员）成员的 userData RegistryPackage 的定位器为 /registry/userData。属于 userData RegistryPackage 成员的 folder1 RegistryPackage 的定位器为 /registry/userData/folder1。

URN RegistryPackage 的 URN，必须是以 urn: 为开头的 URN。

示例

以下命令将目录更改到 URN 为

urn:uuid:92d3fd01-a929-4eba-a5b4-a3f036733017 的 RegistryPackage。

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

以下命令将把目录更改到位置 /registry/userData/myData。

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

chown

更改 RegistryObject 的所有者。

一般形式

```
chown {URN | %index}
```

说明

chown 命令用于把前面的 select 命令所选取的对象的拥有权更改到前面的 users 命令列出时通过 URN 或引用用户的 URN 指定的用户。

只有具有管理员角色的用户才能成功地执行此命令。

操作数

URN URN 指定的 User 对象。

%index 在前面的 users 命令中列出的 User 对象的 URN 的数字引用。

示例

以下命令将把选定对象的拥有权更改到 URN

urn:uuid:26aa17e6-d669-4775-bfe8-a3a484d3e079 所指定的用户。

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

以下命令将把选定对象的拥有权更改到前面的 users 命令中的编号为 2 的用户。

```
admin> chown %2
```

cp

将文件和文件夹复制到 Registry 中。

一般形式

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

说明

`cp` 命令将文件夹和文件分别作为 `RegistryPackage` 和 `ExtrinsicObject` 对象复制到 `Registry` 中。

从其中复制文件和文件夹的本地文件系统中的本地目录默认为启动管理工具的当前目录。启动管理工具时，您可以使用 `-localdir` 选项更改本地目录。管理工具启动后，您可以使用 `lcd` 命令更改本地目录。您可以使用 `show localdir` 命令获取当前本地目录的绝对路径。

该命令是递归命令。即，如果指定一个目录，则该命令将复制该目录下的所有文件和文件夹。

选项

- `-owner` 将复制的 `Registry` 对象的拥有者设置为通过 `URN` 或 `%index` 参数指定的用户。有关这些参数的说明，请参见 `chown` 命令的说明。您必须具有管理员角色才能指定您之外的拥有者。
- `-exclude` 复制其文件名不包含指定模式的所有文件，其中 `pattern` 是由文字字符和特殊字符星号 (*)（表示零个或多个字符）和问号 (?)（表示一个字符，且仅表示一个字符）组成的模式。

可以多次指定此选项。
- `-include` 复制其文件名包含指定模式的所有文件，其中 `pattern` 是由文字字符和特殊字符星号 (*)（表示零个或多个字符）和问号 (?)（表示一个字符，且仅表示一个字符）组成的模式。

可以多次指定此选项。

操作数

- `pattern` 要复制的文件或文件夹，通过由文字字符和特殊字符星号 (*)（表示零个或多个字符）和问号 (?)（表示一个字符，且仅表示一个字符）组成的模式指定。可以指定多个 `pattern`。

示例

以下命令将把目录 `mydir` 复制到 `Registry`，以将其归前面的 `users` 命令中编号为 4 的用户拥有。

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

以下命令将把目录 `mydir` 复制到 `Registry` 中（以字符串 `.z` 或 `.c` 结尾的文件和目录除外）。

```
admin> cp mydir -exclude /\.z -exclude /\.c
```

echo

回显字符串。

一般形式

echo *string*

说明

`echo` 命令用于将指定的 *string* 回显到输出中。在以批处理模式运行管理工具时，如果您在 `-command` 选项中指定此命令会十分有用。

操作数

string 一个字符序列。

示例

以下命令将把日期和 `ls` 命令的结果打印到日志文件中。

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

help

显示关于命令的信息。

一般形式

help [*command_name*]

说明

`help` 命令用于显示关于可用命令或指定命令的信息。

对于具有子命令（例如 `add` 和 `show`）的命令，`help` 命令将显示关于子命令的信息。

如果不指定参数，`help` 命令将显示所有命令的用法信息。

操作数

`command_name` 管理工具命令的名称。

示例

以下命令将显示所有命令的用法信息。

```
admin> help
```

以下命令将显示 `lcd` 命令的用法信息。

```
admin> help lcd
```

以下命令将显示 `add` 子命令的用法信息。

```
admin> help add
```

lcd

更改本地文件系统中的当前目录。

一般形式

```
lcd [path_name]
```

说明

`lcd` 命令用于更改本地文件系统中的当前本地目录。

如果不指定参数，`lcd` 命令将把当前目录更改到您的默认主目录。

操作数

path_name 目录名称，可以是绝对目录，也可以是相对目录。

示例

以下命令将把当前本地目录更改到 `/usr/share` 目录。

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

以下命令将把当前本地目录更改到本地文件系统中的默认主目录。

```
admin> lcd
```

ls

列出当前 RegistryPackage 中的对象。

一般形式

```
ls {pattern | URN}
```

说明

不带有参数的情况下，`ls` 命令将列出当前 RegistryPackage 中的对象。如果提供了 *pattern* 或 *URN*，该命令将列出当前 RegistryPackage 中其名称（在当前语言环境中）或唯一标识符与 *pattern* 或 *URN* 匹配的对象。

操作数

pattern 由文字字符和特殊字符星号 (*)（表示零个或多个字符）和问号 (?)（表示一个字符，且仅表示一个字符）组成的模式。可以指定多个 *pattern*。

URN 以 `urn:` 为开头的 URN，例如 `urn:uuid:4a6741e7-4be1-4cfb-960a-e5520356c4fd`。可以指定多个 *URN*。URN 必须是对象的唯一标识符（不能是逻辑标识符）。

示例

以下命令将列出当前 RegistryPackage 中的所有对象。

```
admin> ls
```

以下命令将列出名称与模式 urn:bird:poultry:chicken 相匹配或 ID 为 urn:bird:poultry:chicken 的所有对象。

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

以下命令将列出名称与模式 *bird* 相匹配的所有对象。（如果 *bird* 是有效 ID，该命令还将列出 ID 为 *bird* 的对象。）

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

以下命令将列出名称与模式 *bird* 或 urn:bird:poultry:chicken 相匹配或 ID 为 urn:bird:poultry:chicken 的所有对象。

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

pwd

显示指向当前 RegistryPackage 的路径。

一般形式

```
pwd
```

说明

pwd 命令用于使用与当前语言环境匹配度最好的名称显示指向当前 RegistryPackage 的路径，该命令还用于显示路径的语言环境。

示例

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

quit

退出管理工具。

一般形式

quit

说明

`quit` 命令用于退出管理工具。

示例

```
admin> quit
```

rm

从 `RegistryPackage` 中删除对象。

一般形式

rm [-d] [-r] {*pattern* | *URN*}

说明

`rm` 命令用于删除当前 `RegistryPackage` 中名称（当前语言环境中）与 *pattern* 或 *URN* 指定的模式相匹配的成员对象。

如果匹配的 `RegistryObject` 是多个 `RegistryPackage` 对象的成员，则此命令仅删除当前 `RegistryPackage` 与该对象之间的关联。仅当删除关联后该对象与任何其他 `RegistryObject` 都没有关联时才从 `Registry` 中删除该对象。

如果匹配的成员对象本身是一个包含其他对象的 `RegistryPackage`，则除非指定 `-r` 选项或 `-d` 选项，否则该对象和当前 `RegistryPackage` 与成员 `RegistryPackage` 之间的关联都不会被删除。

同时指定 `-d` 和 `-r` 选项时将递归应用 `-d` 选项，因此无论是否有其他关联，所有由 `-r` 选取的对象（及其关联）都将被删除。

选项

- d 删除当前 RegistryPackage 与指定 RegistryPackage 之间的关联。仅当指定的 RegistryPackage 只剩下与其成员对象之间的关联时才将前者删除。对于立即删除的 RegistryPackage 的成员对象，它们并非其他 HasMember 关联的目标，现在可以作为 Registry 根的成员来访问它们了。
- r 删除指定的 RegistryPackage 对象及其所有的子对象（对象具有其他关联时除外）。

操作数

pattern 由文字字符和特殊字符星号 (*)（表示零个或多个字符）和问号 (?)（表示一个字符，且仅表示一个字符）组成的模式。可以指定多个 *pattern*。

URN 以 urn: 为开头的 URN，例如
urn:uuid:4a6741e7-4be1-4cfb-960a-e5520356c4fd。可以指定多个 *URN*。

示例

以下命令将删除包含字符串 "stat" 的所有 RegistryPackage 对象及其所有子对象。

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

select

执行一个 SQL select 语句。

一般形式

```
select [SQL]
```

说明

select 命令用于选取并列出于通过将整条命令作为一个 SQL 查询执行所指定的对象。如果不指定参数，该命令将列出由前面的 select 命令或 sqlselect 选项选取的所有对象。

操作数

SQL SQL `select` 语句（前面无需加 `select`，因为已经以命令名称的方式提供了该语句）。

示例

以下命令将列出 Registry 中的所有 `ClassificationScheme` 对象：

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

set

设置属性值。

一般形式

```
set property value
```

说明

`set` 命令用于设置管理工具 Shell 的属性的值。

管理工具支持以下属性和值。

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

启用或禁用调试信息的输出。

```
set editor string
```

设置在管理工具启动交互式编辑器时使用的命令。在 UNIX 和 Linux 系统上的默认值为 `/bin/vi`。

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

启用或禁用执行命令时更详细的消息输出。

操作数

property 下列属性之一：`debug`、`editor`、`verbose`。

value 指定的属性所支持的值。有关详细信息，请参见“说明”部分。

示例

以下命令将把编辑器设置为 `/usr/bin/vi`，而不是默认的 `/bin/vi`。

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

以下命令将启用调试。

```
admin> set debug true
```

以下命令将禁用详细输出。

```
admin> set verbose off
```

show

显示属性值。

一般形式

```
show [property]
```

说明

`show` 命令用于显示管理工具 Shell 的属性的值。

如果不指定参数，此命令将显示所有属性的值。

此命令支持以下属性：

<code>debug</code>	是否启用调试输出。
<code>editor</code>	管理工具启动交互式编辑器时使用的编辑器。
<code>exception</code>	从前面刚执行的命令返回的异常堆栈追踪（如果有）。
<code>localdir</code>	本地文件系统中的当前目录。使用 <code>lcd</code> 命令设置此属性。有关详细信息，请参见第 47 页中的“ <code>lcd</code> ”。
<code>locale</code>	当前语言环境。
<code>verbose</code>	是否启用详细输出。

操作数

property 要显示其当前值的属性。可以显示 `exception` 和 `locale` 属性，但不能使用 `set` 命令设置这两个属性。

示例

以下命令将显示上一个命令返回的异常。

```
admin> show exception
```

users

列出当前的 User 对象。

一般形式

```
users
```

说明

users 命令用于列出 Registry 中当前的 User 对象。

输出采用以下格式：

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

在输出中，*index* 是可以使用的数字值，包括百分号 (%)，以在运行 `chown` 或 `cp` 命令时引用用户。*lastname*、*firstname* 和 *middlename* 分别是用户的姓氏、名字和中间名。

示例

以下命令将显示当前用户：

```
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
```

索引

A

add association 命令, 36-37
add user 命令, 37-43
AdhocQuery 对象, 添加到 Web 控制台, 24-29
-alias 命令行选项, 33
Application Server 管理控制台, 19
Application Server 域
 管理, 18-20
 将根证书添加到信任存储中, 20
 停止并重新启动, 19-20

C

cd 命令, 43
chown 命令, 44
command-line 选项, -keypass, 33
-command 命令行选项, 33
cp 命令, 44-46
-create 命令行选项, 33

D

-debug 命令行选项, 33
debug 属性
 设置, 52-53
 显示值, 53-54

E

echo 命令, 46

editor 属性

 设置, 52-53
 显示值, 53-54
exception 属性, 显示值, 53-54
ExternalLink 对象, 允许验证 URL, 22

H

help 命令, 46-47
-help 命令行选项, 33

K

-keypass 命令行选项, 33

L

lcd 命令, 47-48
-localdir 命令行选项, 33
-locale 命令行选项, 33
locale 属性, 显示值, 53-54
ls 命令, 48-49

P

pwd 命令, 49

Q

quit 命令, 49-50

R

Registry 对象

更改所有者, 44

列出, 48-49

识别, 35

-registry 命令行选项, 33

registry 域

管理, 18-20

将根证书添加到信任存储中, 20

停止并重新启动, 19-20

RegistryPackage 对象

创建, 33

列出内容, 48-49

删除成员对象, 50-51

显示路径, 49

RegistryPackage 位置, 更改, 43

rm 命令, 50-51

-root 命令行选项, 33

S

select 命令, 51-52

Service Registry 的数据库, 备份和恢复, 30

ServiceBinding 对象, 允许验证 URL, 22

set 命令, 52-53

show 命令, 53-54

SQL select 语句, 执行, 34

SQL 语句, 执行, 51-52

-sqlselect 命令行选项, 34

U

users, 列出, 54

users 命令, 54

V

-v 命令行选项, 34

-verbose 命令行选项, 34

verbose 属性

设置, 52-53

显示值, 53-54

W

Web 控制台, 添加预定义的查询, 24-29

安

安装属性, 16

创

创建管理员, 22-23

词

词汇表, 链接到, 10

从

从 RegistryPackage 中删除对象, 50-51

代

代理服务器主机和端口, 设置, 22

当

当前目录, 更改, 47-48

端

端口, Service Registry 的默认, 18-19

根

根证书, 添加到 registry 域的信任存储中, 20

关

关联, 添加到 Registry, 36-37

管

管理工具

介绍, 31

命令行选项, 32-34

启动, 31-34

使用, 17-18

停止, 49-50

管理控制台, Application Server, 19

管理员, 创建, 22-23

将

将文件和文件夹复制到 Registry, 44-46

交

交互式模式, 32

命

命令

add association, 36-37

add user, 37-43

cd, 43

chown, 44

cp, 44-46

echo, 46

help, 46-47

lcd, 47-48

ls, 48-49

pwd, 49

quit, 49-50

rm, 50-51

select, 51-52

set, 52-53

show, 53-54

users, 54

命令行选项, 32-34

-alias, 33

-command, 33

命令行选项 (续)

-create, 33

-debug, 33

-help, 33

-localdir, 33

-locale, 33

-registry, 33

-root, 33

-sqlselect, 34

-v, 34

-verbose, 34

目

目录, 更改, 47-48

配

配置 Service Registry, 15-21

批

批处理模式, 32

启

启动管理工具, 31-34

区

区分大小写, 35

设

设置属性值, 52-53

属

属性值

设置, 52-53

显示, 53-54

停

停止管理工具, 49-50

预

预定义的查询, 添加到 Web 控制台, 24-29

通

通配符, 使用, 35

退

退出管理工具, 49-50

外

外部 Web 站点, 允许访问, 22

文

文件和文件夹, 复制到 Registry, 44-46

文件系统, 本地

 更改当前目录, 47-48

 基本目录, 33

显

显示, 属性值, 53-54

异

异常, 显示, 34-35

用

用户, 添加到 Registry, 37-43

用户名, 在命令行中指定, 33

语

语言环境, 对指定名称的影响, 35