



Sun Java™ System

Content Delivery Server 5.1

安装指南

Sun Microsystems, Inc.
www.sun.com

文件号码: 820-5378-10
2008 年 6 月

请将有关本文档的意见和建议提交至: <http://www.sun.com/sunsurveys/dsc/dsc-feedback.jsp>

版权所有 ©2008 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 保留所有权利。

对于本档中介绍的产品，Sun Microsystems, Inc. 对其所涉及的技术拥有相关的知识产权。需特别指出的是（但不限于此），这些知识产权可能包含在 <http://www.sun.com/patents> 中列出的一项或多项美国专利，以及在美国和其他国家/地区申请的一项或多项其他专利或待批专利。

美国政府权利 - 商业用途。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。

本发行版可能包含由第三方开发的内容。

Sun、Sun Microsystems、Sun 徽标、Java、JDK、JDBC、Java Naming and Directory Interface、JavaServer Pages、JSP 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。

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。

Adobe 徽标是 Adobe Systems, Incorporated 的注册商标。

本服务手册所介绍的产品以及所包含的信息受美国出口控制法制约，并应遵守其他国家/地区的进出口法律。严禁将本产品直接或间接地用于核设施、导弹、生化武器或海上核设施，也不能直接或间接地出口给核设施、导弹、生化武器或海上核设施的最终用户。严禁出口或转口到美国禁运的国家/地区以及美国禁止出口清单中所包含的实体，包括但不限于被禁止的个人以及特别指定的国家/地区的公民。

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。



请回收



目录

前言 xiii

1. 安装概述 1-1

1.1 规划注意事项 1-1

1.1.1 服务器布局 1-1

1.1.2 可缩放性 1-2

1.1.3 容量 1-2

1.2 部署方案 1-2

1.2.1 基本配置 1-3

1.2.2 Catalog Manager 配置 1-3

1.2.3 Vending Manager 配置 1-4

1.2.4 高级配置 1-5

2. 准备环境 2-1

2.1 验证先决条件 2-1

2.1.1 硬件要求 2-2

2.1.2 软件要求 2-2

2.1.3 可选组件 2-3

2.1.4 数据库要求 2-4

2.2 解压缩 Content Delivery Server 文件 2-5

2.3 设置环境变量 2-5

3. 部署 Content Delivery Server 3-1

3.1 基本部署 3-1

3.1.1 编辑 `init_env` Shell 脚本 3-2

3.1.2 设置数据库 3-2

3.1.3 设置部署配置文件 3-7

3.1.4 部署 3-8

3.1.5 设置应用服务器 3-8

3.1.5.1 设置 Sun Java System Application Server 3-9

3.1.5.2 设置 WebLogic Server 3-11

3.1.6 准备 Content Delivery Server 的使用 3-13

3.1.6.1 配置 Content Delivery Server 3-13

3.1.6.2 准备 Catalog Manager 3-14

3.1.6.3 准备 Vending Manager 3-15

3.2 Catalog Manager 部署 3-16

3.2.1 编辑 `init_env` Shell 脚本 3-16

3.2.2 设置数据库 3-16

3.2.3 设置部署配置文件 3-17

3.2.4 部署 3-18

3.2.5 设置应用服务器 3-18

3.2.6 准备 Catalog Manager 以供使用 3-18

3.3 Vending Manager 部署 3-18

3.3.1 编辑 `init_env` Shell 脚本 3-19

3.3.2 设置数据库 3-19

3.3.3 设置部署配置文件 3-20

3.3.4 部署 3-21

3.3.5 设置应用服务器 3-21

3.3.6 准备 Vending Manager 以供使用 3-21

- 3.4 高级部署 3-22
 - 3.4.1 编辑 `init_env` Shell 脚本 3-22
 - 3.4.2 设置数据库 3-22
 - 3.4.3 设置部署配置文件 3-23
 - 3.4.4 部署 3-24
 - 3.4.5 设置应用服务器 3-24
 - 3.4.6 准备高级部署以供使用 3-24

- 4. 配置访问 4-1
 - 4.1 针对防火墙配置 Content Delivery Server 4-1
 - 4.2 控制外部系统的访问 4-2
 - 4.2.1 访问控制文件 4-2
 - 4.2.2 访问控制进程 4-3
 - 4.3 配置别名转发 4-4
 - 4.4 控制对 Sun Java System Application Server 的访问 4-6

- 5. 操作 Content Delivery Server 5-1
 - 5.1 启动 Content Delivery Server 5-1
 - 5.2 停止 Content Delivery Server 5-2

- 6. 卸载 Content Delivery Server 6-1
 - 6.1 卸载部署 6-1
 - 6.1.1 卸载 Sun Java System Application Server 上的部署 6-1
 - 6.1.2 卸载 WebLogic Server 上的部署 6-2
 - 6.2 卸载整个产品 6-3

- A. 部署配置文件 A-1
 - A.1 配置属性 A-1
 - A.2 与 Sun Java System Application Server 一起使用的样例文件 A-5
 - A.3 与 WebLogic Server 一起使用的样例文件 A-7

B. 访问控制定义 B-1

C. X11 库 C-1

索引 索引 -1

图

- 图 1-1 基本配置 1-3
- 图 1-2 Catalog Manager 配置 1-4
- 图 1-3 Vending Manager 配置 1-5

表

表 2-1	Content Delivery Server 组件	2-2
表 2-2	软件要求	2-2
表 2-3	可选组件	2-3
表 3-1	在 <code>init_env</code> 中设置的环境变量	3-2
表 3-2	<code>deploy</code> 命令的可选参数	3-8
表 3-3	<code>deploymq.sh</code> 参数	3-9
表 3-4	<code>deploy.sh</code> 参数	3-11
表 3-5	样例语言环境	3-14
表 4-1	别名转发文件元素	4-4
表 A-1	常规属性	A-1
表 A-2	与 WebLogic Server 一起使用的常规属性	A-4
表 A-3	组件前缀	A-4
表 C-1	所需的 X11 库	C-1

代码样例

代码示例 3-1	样例数据库配置文件	3-3
代码示例 3-2	样例数据库用户文件	3-5
代码示例 4-1	样例 aliasForwarding.xml 文件	4-5
代码示例 A-1	与 Sun Java System Application Server 一起使用的样例部署配置文件	A-5
代码示例 A-2	与 WebLogic Server 一起使用的样例部署配置文件	A-8
代码示例 B-1	访问控制语句的 BNF 规则	B-2

前言

《Sun Java™ System Content Delivery Server 5.1 安装指南》介绍了如何安装和运行 Content Delivery Server。它指明了所需的硬件和软件，并介绍了 Content Delivery Server 组件的部署。

阅读本文档之前

本指南适用于负责部署和维护 Content Delivery Server 的系统管理员。它假定管理员具有联网、数据库和 Web 技术方面的知识。

本文档的组织结构

- **第 1 章**介绍了规划注意事项和部署方案。
- **第 2 章**介绍了部署 Content Delivery Server 的硬件和软件要求。此外还介绍了如何设置 Content Delivery Server 的环境。
- **第 3 章**提供了有关安装 Content Delivery Server 和部署组件的说明。
- **第 4 章**提供了有关配置对 Content Delivery Server 的访问权限的说明，其中包括运行防火墙和使用别名。
- **第 5 章**介绍了如何启动和停止 Content Delivery Server。
- **第 6 章**提供了有关卸载单个 Content Delivery Server 部署以及卸载全部 Content Delivery Server 部署的说明。
- **附录 A**介绍了部署和配置工具所使用的配置文件的格式。此外还提供了样例配置文件。
- **附录 B**介绍了用于创建访问控制文件的语句。Content Delivery Server 使用访问控制文件来确定允许访问其组件的系统。

- 附录 C 指明了水印预览所需的 X11 库。

Shell 提示符

Shell	提示符
C shell	<i>machine-name%</i>
C shell 超级用户	<i>machine-name#</i>
Bourne shell 和 Korn shell	\$
Bourne shell 和 Korn shell 超级用户	#

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 <code>ls -a</code> 列出所有文件。 % You have mail.
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	% 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命令行变量。	这些称为 <i>class</i> 选项。 要删除文件，请键入 rm filename 。
新词术语强调	新词或术语以及要强调的词。	您 必须 成为超级用户才能执行此操作。
《书名》	书名	阅读《用户指南》的第 6 章。

注 – 字符的显示方式随浏览器设置的不同而有所不同。如果未正确显示字符，请在浏览器中将字符编码更改为 Unicode UTF-8。

相关文档

下表列出了本产品的文档。可以从以下位置获取联机文档：

<http://docs.sun.com/app/docs/prod/cds>

应用	书名	文件号码	格式	位置
门户署名和本地化	《Sun Java™ System Content Delivery Server 5.1 署名和本地化指南》	820-5374-10	PDF HTML	\$(CDS_HOME)/Documentation/branding 和联机文档。
规划系统	《Sun Java™ System Content Delivery Server 5.1 容量规划指南》	820-1939-10	PDF HTML	\$(CDS_HOME)/Documentation/capacity 和联机文档。
使用 Developer Portal	《Sun Java™ System Content Delivery Server 5.1 内容开发者指南》	820-5376-10	PDF HTML	\$(CDS_HOME)/Documentation/devguide 和联机文档。
使用提供的 API	《Sun Java™ System Content Delivery Server 5.1 定制指南》	820-5375-10	PDF HTML	\$(CDS_HOME)/Documentation/customization 和联机文档。
疑难解答	《Sun Java™ System Content Delivery Server 5.1 错误消息》	820-5377-10	PDF HTML	\$(CDS_HOME)/Documentation/errmsgs 和联机文档。
安装系统	《Sun Java™ System Content Delivery Server 5.1 安装指南》	820-5378-10	PDF HTML	\$(CDS_HOME)/Documentation/install 和联机文档。
设置和集成现有的基础架构	《Sun Java™ System Content Delivery Server 5.1 集成和配置指南》	820-5379-10	PDF HTML	\$(CDS_HOME)/Documentation/integration 和联机文档。
迁移到最新发行版本	《Sun Java™ System Content Delivery Server 5.1 迁移指南》	820-1945-10	PDF HTML	\$(CDS_HOME)/Documentation/migration 和联机文档。
产品参考信息	《Sun Java™ System Content Delivery Server 5.1 参考手册》	820-5380-10	PDF HTML	\$(CDS_HOME)/Documentation/refman 和联机文档。
监视和管理系统	《Sun Java™ System Content Delivery Server 5.1 系统管理指南》	820-5381-10	PDF HTML	\$(CDS_HOME)/Documentation/system-mgmt 和联机文档。

文档、支持和培训

Sun 服务	URL
文档	http://www.sun.com/documentation/
支持	http://www.sun.com/support/
培训	http://www.sun.com/training/

第三方 Web 站点

Sun 对本文中提到的第三方 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或连带产生的实际或名义损害或损失，Sun 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。可通过以下方式发送您的意见：访问 <http://docs.sun.com>，然后单击“发送意见” (Send Comments)。

请在反馈中包括文档的标题和文件号码：

《Sun Java™ System Content Delivery Server 5.1 安装指南》，文件号码 820-5378-10。

第 1 章

安装概述

Sun Java™ System Content Delivery Server 安装包括：确保您具有必备的硬件和软件、部署组件以及设置系统。

本节包括以下主题：

- [安装和部署](#)
- [部署方案](#)

1.1 规划注意事项

规划对于确保成功安装 Content Delivery Server 至关重要。本节介绍规划部署时应注意的一些事项以及关于容量规划的一些提示。

1.1.1 服务器布局

可以在一个或多个应用服务器上部署所有产品组件，也可以在同一主机或多个主机上进行部署。在多个主机上安装 Content Delivery Server 组件、Web 服务器、应用服务器和数据库可以提高容错能力并改善系统性能。

开始规划的一种方法是确定要部署的 Vending Manager 的数量。可以通过部署单个 Vending Manager 向所有订户提供由 Catalog Manager 管理的所有内容，或者通过部署多个定制的 Vending Manager 来提供面向不同读者的内容。

已将 Content Delivery Server 优化成可与多 CPU 计算机（例如 UNIX® 系统四路服务器）一起使用。可以使用多托管功能在一个主机上安装多个 Vending Manager。有关设置多托管功能的信息，请参见所使用的应用服务器的文档。

1.1.2 可缩放性

Content Delivery Server 是具有完全缩放功能的系统。可以很容易地添加组件的其他实例，以便提高系统容量。Content Delivery Server 体系结构还支持群集。群集是一种将多个服务器作为一个服务器进行设置的技术。在群集中，每个服务器都与群集中的其他服务器等价。

1.1.3 容量

要开始 Content Delivery Server 的容量规划，请先从所需的 Catalog Manager 和 Vending Manager 数量开始。在容量规划中，必须考虑日后的增长。例如，订户群可能会呈指数增长。此外，一个非常流行的应用程序可能表明了会加重系统负担的使用峰值。

请从以下系统特征考虑容量估计值：

- 现有或潜在的订户群
- 平均内容下载大小
- 高峰下载条件下订户允许的延迟时间
- 订户的使用模式
- 平均内容类型

例如，Content Delivery Server 在通常情况下的响应时间都小于一秒钟，但随着并行客户机数量的增长，响应性能会线性降低。

每秒钟可以下载的高峰量取决于多种因素，包括 Web 服务器、应用服务器和 Oracle 数据库的性能。Content Delivery Server 维护数据库中的订户状态，因此，不同于其他系统，它几乎没有事务时间消息传送开销。

有关容量规划的其他信息，请参见《Sun Java™ System Content Delivery Server 5.1 容量规划指南》。

1.2 部署方案

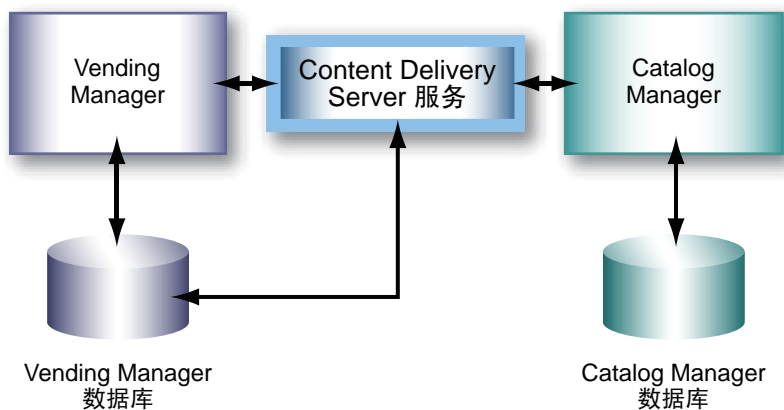
Content Delivery Server 体系结构支持各种部署方案：从在同一台服务器上部署所有组件到将各个组件分别部署在其专用的服务器上。为每个部署创建的部署配置文件用于确定如何部署组件。

注 – 讨论 Content Delivery Server 部署时，“服务器”是指应用服务器实例。“主机”是指实际的计算机。Content Delivery Server 在服务器上进行部署。如果需要，一台主机可以支持多个服务器实例。但是，建议在同一台计算机上的部署不要超过四个。

1.2.1 基本配置

基本配置包括在同一个服务器上部署所有组件。这代表运行 Content Delivery Server 所需的最低设置，并且是一般用于最初部署的配置。下图说明了基本配置。

图 1-1 基本配置

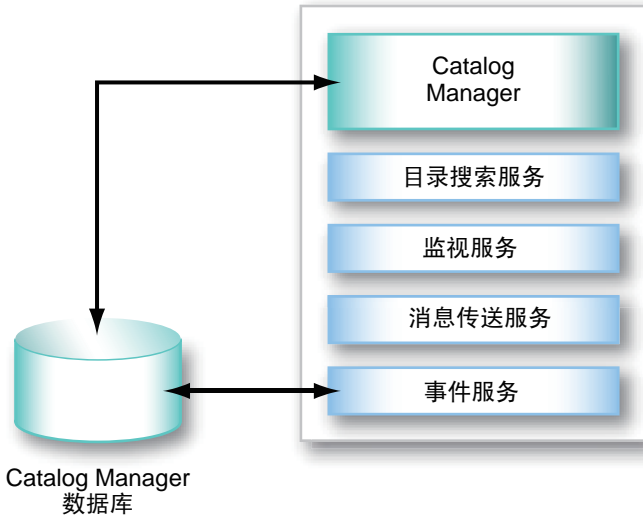


1.2.2 Catalog Manager 配置

如果业务模型包括提供远程 Vending Manager 且集中管理的 Catalog Manager，则可能不需要本地 Vending Manager。在 Catalog Manager 配置中，仅部署了 Catalog Manager、Developer Portal、事件服务、消息传送服务、监视服务以及目录搜索服务。在 Vending Manager 配置中，单独部署所有 Vending Manager。

下图说明了 Catalog Manager 配置。

图 1-2 Catalog Manager 配置



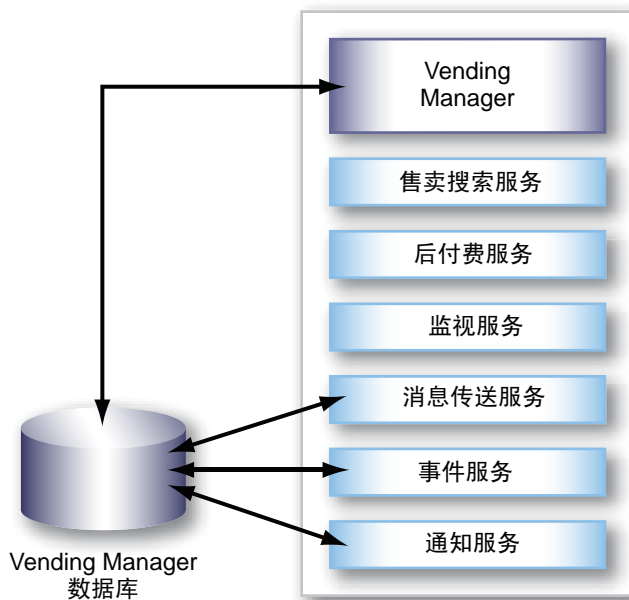
1.2.3 Vending Manager 配置

Content Delivery Server 的一个优点是能够使用定制的 Vending Manager 为各种订户群提供服务。在该配置中，部署和配置了 Vending Manager，以便从现有的 Catalog Manager 中接收内容。可以将 Vending Manager 与 Catalog Manager 部署在同一位置，也可以将其部署到远程位置。

每个 Vending Manager 都具有其自己的记帐管理器、后付费处理器和订户群。Subscriber Portal 和 Fulfillment Manager 以及每个 Content Delivery Server 服务（目录搜索服务除外）的实例都是使用 Vending Manager 部署的。

下图说明了 Vending Manager 配置。

图 1-3 Vending Manager 配置



1.2.4 高级配置

部署配置文件支持将任何组件组合部署到服务器。例如，可以在第一台服务器上部署 Catalog Manager，在第二台服务器上部署 Vending Manager，然后在第三台服务器上部署服务组件。以下情况可能需要定制部署：

- 服务器上的资源有限。
- 需要在专用服务器上隔离资源密集组件。
- 需要管理器组件的各分散实例共享一个服务组件。

第 2 章

准备环境

在部署 Sun Java System Content Delivery Server 之前，必须先设置环境。

本节包括以下主题：

- [验证先决条件](#)
- [解压缩 Content Delivery Server 文件](#)
- [设置环境变量](#)

2.1 验证先决条件

开始安装 Content Delivery Server 之前，请确保具有所需的硬件，并且每个要部署 Content Delivery Server 组件的服务器上都安装了所需的软件。还必须安装要使用的可选功能所需的软件。

本节介绍以下要求：

- [硬件要求](#)
- [软件要求](#)
- [可选组件](#)
- [数据库要求](#)

2.1.1 硬件要求

下表显示了计划部署 Content Delivery Server 组件的所有主机的最低硬件要求。

表 2-1 Content Delivery Server 组件

组件	要求
平台	下列平台之一： <ul style="list-style-type: none">• SPARC™ 处理器上的 Solaris 9® 操作系统• SPARC 处理器上的 Solaris 10 操作系统• 用于 x86 平台的 Solaris 10 操作系统*
内存	最低为 2 GB RAM。
交换空间	大小为内存的两倍；例如，如果 RAM 为 2 GB，则建议的交换空间大小为 4 GB。有关设置交换空间大小的信息，请参见操作系统文档。
硬盘驱动器	最低为 5 GB 磁盘空间，用于系统安装、数据库和日志文件。所需的额外空间取决于所管理的内容的数量和大小。

* 如果使用的是 WebLogic Server，则必须使用 32 位 x86 平台。

2.1.2 软件要求

下表显示了计划部署 Content Delivery Server 组件的主机所需的软件。

表 2-2 软件要求

组件	要求
数据库	Oracle Database 10g 版本 1 (10.1.0.2) 或更高版本， Standard Edition 或 Enterprise Edition 注 – 字符集必须为 AL32UTF8。
应用服务器	使用下列应用服务器之一： <ul style="list-style-type: none">• Sun Java System Application Server 9.0 Update 1 Platform Edition• BEA WebLogic Server 9.2 MP2
Bash Shell	Bash Shell 版本 2.0.5 或更高版本（可以从以下位置获取）： http://www.gnu.org/software/bash/bash.html 注 – Bash Shell 必须安装在 /bin 目录中。

表 2-2 软件要求（续）

组件	要求
Java 软件开发工具包 (Java Software Development Kit, JDK™)	Java Platform Standard Edition Development Kit 5.0 Update 11 (1.5.0_11)。
浏览器	支持以下浏览器： <ul style="list-style-type: none"> • Firefox 2.0 或更高版本 • Internet Explorer 6.0 或更高版本 对于基于设备的 Subscriber Portal，还支持任何支持无线标记语言 (Wireless Markup Language, WML) 1.0、简化 HTML (compact HTML, cHTML) 或可扩展 HTML (extensible HTML, XHTML) 的设备浏览器。
解压缩	解压缩 Zip 文件的实用程序。

2.1.3 可选组件

Content Delivery Server 的某些功能具有一些其他要求，仅当要使用该功能时才需要满足这些要求。下表介绍了这些要求。

表 2-3 可选组件

组件	要求
设备库	设备库包含应用程序使用的 API。Content Delivery Server 中包括多个标准库。要支持使用其他 API 的应用程序，必须先将库提交到 Content Delivery Server。例如，要支持 iAppli 应用程序，则需要使用 DoJa 库。
轻量目录访问协议 (Lightweight Directory Access Protocol, LDAP) 服务器	要将 Content Delivery Server 与存储在 LDAP 目录中的现有订户数据库集成在一起，您需要一个 LDAP 服务器，例如 Sun Java System Directory Server 5.2。
WAP 网关	处理无线数据传输的服务器。要使用无线 (over-the-air, OTA) 技术传送内容，您需要一个 WAP 网关。
WAP 推送代理网关 (Push Proxy Gateway, PPG)	处理 WAP 推送消息的网关。要使用 WAP 推送来传送消息或数字权限对象，您需要一个 WAP PPG。
对等短消息 (Short Message Peer to Peer, SMPP) API 库	要使用 SMPP 发送或接收消息，您需要从 SourceForge.net 中下载 Java SMPP API 库 0.3.7 版。
短消息服务中心 (Short Messaging Service Center, SMSC)	处理短消息服务 (Short Messaging Service, SMS) 消息的服务器。要提供 SMS 服务或以 SMS 消息的形式传送内容，您需要一个 SMSC。要使用 SMS 推送来传送数字权限对象，您需要一个支持此内容类型的 SMSC。

表 2-3 可选组件（续）

组件	要求
多媒体消息服务中心 (Multimedia Message Service Center, MMSC)	处理多媒体消息服务 (Multimedia Message Service, MMS) 消息的服务器。要提供 MMS 服务，您需要一个 MMSC。
Open Mobile Alliance Digital Rights Management (OMA DRM) 1.0 指导实现	要按照 OMA DRM 1.0 指导来保护内容，您需要一个实现该指导的应用程序。Content Delivery Server 附带提供了 SafeNet DRM Fusion Toolkit 2.1.1 支持。有关 SafeNet DRM Fusion Toolkit 的信息，请参见 http://www.safenet-inc.com/digital_rights_management/DRM_Fusion_Toolkit.asp 。
流式传输服务器	存储和传送按需和实时以流的方式传输的内容的服务器。要在 Content Delivery Server 中支持流式传输的内容，您需要使用流式传输服务器。
Java 高级图像处理 (Java Advanced Imaging, JAI) 图像 I/O 工具	Java 图像 I/O 框架的插件，用于读取、写入和流式传输图像格式。要使用 Content Delivery Server 附带的水印实用程序，您需要 1.1 版。
X11 库	X 服务器的一种实现，它为其他系统提供显示资源。要使用 Content Delivery Server 附带的水印实用程序，请参见附录 C 中的所需软件包列表。

2.1.4 数据库要求

需要使用 Oracle Database 10g 版本 1 (10.1.0.2) 或更高版本 Standard Edition 或 Enterprise Edition。要设置 Oracle 数据库，请执行以下操作：

1. 验证 Oracle 数据库的安装是否有效。

可以从 Oracle 的 TechNet Web 站点 <http://technet.oracle.com/> 获得相关文档（包括安装和升级详细信息）。将数据库的字符集设置为 AL32UTF8。

2. 按照 Oracle 文档中所述的预安装步骤操作。

3. 确保获得了以下 Oracle 数据库安装信息：

- 主机名或 IP 地址
- 端口号
- 系统用户 ID
- 系统用户密码
- 数据库 SID

以后配置数据库时需要此信息。

2.2 解压缩 Content Delivery Server 文件

有关接收到的文件的信息，请参见《Sun Java™ System Content Delivery Server 5.1 发行说明》。请按照发行说明所述，创建一个主目录并从 Zip 文件中提取产品文件。将在主目录下面创建以下目录：

```
/$CDS_HOME
  /bin
  /cfg
  /deployment
  /dist
  /Documentation
  /external
  /javadoc
  /LicenseReadme
  /ReleaseNotes
  /tools
```

2.3 设置环境变量

Content Delivery Server 要求每台托管 Content Delivery Server 组件的计算机上正确设置环境变量。您必须创建一个名为 CDS_HOME 的变量，该变量指向包含 Content Delivery Server 产品文件的目录。另外，PATH 变量还必须包括包含可执行文件的目录。

例如，如果将产品文件解压缩到名为 /cds 的目录中，则下列命令可正确设置环境变量：

```
CDS_HOME=/cds
export CDS_HOME

PATH=$PATH:$CDS_HOME/bin
export PATH
```


第 3 章

部署 Content Delivery Server

本章提供了有关部署 Content Delivery Server 组件的说明。继续操作之前，请确保按照第 2 章的说明正确设置您所在的环境。

本节介绍了以下部署类型：

- **基本部署。** 将所有组件部署到同一主机上。这是可实现运行 Content Delivery Server 的最简单配置。
- **Catalog Manager 部署。** 将 Catalog Manager 和支持组件部署到主机上。
- **Vending Manager 部署。** 将 Vending Manager 和支持组件部署到主机上。要设置多售卖配置，请执行一个基本部署或 Catalog Manager 部署以及一个或多个 Vending Manager 部署。
- **高级部署。** 您可以创建完全定制配置，并有选择地将组件部署到不同主机上。只有在基本配置或多售卖配置无法满足需要时，才需要执行高级部署。

本节包括以下主题：

- [基本部署](#)
- [Catalog Manager 部署](#)
- [Vending Manager 部署](#)
- [高级部署](#)

3.1 基本部署

本节介绍了部署基本配置的步骤，其中所有 Content Delivery Server 组件都部署到同一服务器。基本部署通常是开始使用 Content Delivery Server 的第一个部署。

如果要运行具有单个 Vending Manager 的简单配置，则只需要执行基本部署。如果要运行具有多个 Vending Manager 的比较复杂的配置，请先执行基本部署，然后再执行一个或多个 Vending Manager 部署。如果不想同时部署 Catalog Manager 和 Vending Manager，请先执行 Catalog Manager 部署，然后再执行一个或多个 Vending Manager 部署。

3.1.1 编辑 init_env Shell 脚本

Content Delivery Server 包含一个名为 `init_env.sh` 的 shell 脚本（位于 `$CDS_HOME/bin` 目录）。此脚本将根据您提供的值设置下表中所述的环境变量。

表 3-1 在 `init_env` 中设置的环境变量

变量名	描述
BEA_HOME	BEA 主目录。请将其设置为包含 <code>license.bea</code> 文件的目录，例如， <code>/usr/local/BEA</code> 。 注 - 只有在使用 WebLogic Server 时才需要此变量。
WL_HOME	WebLogic Server 安装目录。请将其设置为包含 <code>/server</code> 目录的目录，例如， <code>/usr/local/BEA/weblogic92</code> 。 注 - 只有在使用 WebLogic Server 时才需要此变量。
SUN_AS_HOME	Sun Java System Application Server 的安装目录，例如， <code>/opt/SUNWappserver/appserver</code> 。 注 - 只有在使用 Sun Java System Application Server 时才需要此变量。
JAVA_HOME	Java 编程语言的主目录。必须正确设置此变量才能运行 <code>Content Delivery Server install</code> 命令 (<code>cdsi</code>) 和 <code>Content Delivery Server execute</code> 命令 (<code>cdse</code>)。 注 - Sun Java System Application Server 和 WebLogic Server 均需要此变量。

3.1.2 设置数据库

请提供所使用的 Oracle 数据库实例的以下信息：

- 主机名或 IP 地址
- 端口号
- 系统用户 ID
- 系统用户密码
- 数据库 SID

Content Delivery Server 需要 Catalog Manager 和每个 Vending Manager 的数据库结构。这些结构通过创建时提供的前缀进行标识。这些结构可以位于相同数据库服务器上，也可以位于不同数据库服务器上。

随 Content Delivery Server 提供的 `Content Delivery Server Install` 命令将根据数据库配置文件中提供的信息创建模式。此配置文件用于标识数据库的位置和创建的模式。

执行基本部署时，将为 Catalog Manager 和 Vending Manager 创建模式。要创建基本部署模式，请执行以下操作：

1. 创建数据库配置文件。

请将位于 \$CDS_HOME/cfg 目录中的样例文件 DBConf.xml 用作指南。以下代码示例显示了样例数据库配置文件。

代码示例 3-1 样例数据库配置文件

```
<CDSDatabase>

  <DBServer name="DBServer1">
    <Hostname>dbhost.domain.com</Hostname>
    <Port>1521</Port>
    <SID>dbsid</SID>
    <Tablespace>USERS</Tablespace>
    <Tablespace name="catalog">USERS</Tablespace>
    <Tablespace name="catalog indexes">USERS</Tablespace>
    <Tablespace name="content">USERS</Tablespace>
    <Tablespace name="content indexes">USERS</Tablespace>
    <Tablespace name="event">USERS</Tablespace>
    <Tablespace name="event indexes">USERS</Tablespace>
    <Tablespace name="subscriber">USERS</Tablespace>
    <Tablespace name="subscriber indexes">USERS</Tablespace>
    <Tablespace name="vending">USERS</Tablespace>
    <Tablespace name="vending indexes">USERS</Tablespace>
    <TempTablespace>TEMP</TempTablespace>
  </DBServer>

  <Catalog>
    <Prefix>CDS</Prefix>
    <Password>dbpassword</Password>
    <DBServer>DBServer1</DBServer>
  </Catalog>

  <Vending name="suncds.vending.server.1">
    <Prefix>CDS</Prefix>
    <Password>dbpassword</Password>
    <DBServer>DBServer1</DBServer>
  </Vending>

  <!-- Additional Vending Managers, if any
  <DBServer name="DBServer2">
    <Hostname>dbhost2.domain.com</Hostname>
    <Port>1521</Port>
    <SID>dbsid2</SID>
    <Tablespace>USERS</Tablespace>
    <Tablespace name="catalog">USERS</Tablespace>
    <Tablespace name="catalog indexes">USERS</Tablespace>
    <Tablespace name="content">USERS</Tablespace>
    <Tablespace name="content indexes">USERS</Tablespace>
    <Tablespace name="event">USERS</Tablespace>
```

代码示例 3-1 样例数据库配置文件（续）

```
<Tablespace name="event indexes">USERS</Tablespace>
<Tablespace name="subscriber">USERS</Tablespace>
<Tablespace name="subscriber indexes">USERS</Tablespace>
<Tablespace name="vending">USERS</Tablespace>
<Tablespace name="vending indexes">USERS</Tablespace>
<TempTablespace>TEMP</TempTablespace>
</DBServer>

<Vending name="suncds.vending.server.2">
  <Prefix>CDS2</Prefix>
  <Password>dbpassword2</Password>
  <DBServer>DBServer2</DBServer>
</Vending>
-->

</CDSDatabase>
```

注 – 该文件包含未加密的密码。请设置文件权限以限制对此文件的访问。

按照以下步骤中所述，编辑所创建的数据库配置文件：

a. 按照以下步骤中所述，设置 DBServer 元素：

i. 将名称属性设置为用于标识数据库服务器的字符串。

在 Catalog 和 Vending 元素中使用该字符串来标识在其上创建每个模式的数据库服务器。

ii. 将 Hostname、Port 和 SID 元素设置为所定义的 Oracle 数据库实例的主机名、端口号和数据库 SID。

iii. 将 Tablespace 和 TempTablespace 元素设置为所定义的 Oracle 数据库实例的名称。

要对每一组表使用单独的表空间，请将各组的 Tablespace 元素设置为不同的值。每组中包括的表都基于大小和活动，而且在以下列表中进行了说明：

- default - 包含不会发生更改的小型表。
- catalog - 包含不经常更改的从小型到中型的表。
- content - 包含随内容更改而经常发生更改的小型到大型表。
- event - 包含当事务发生时发生更改的中型表。
- subscriber - 包含随添加订户而发生更改的小型到中型表。
- vending - 包含不经常更改的从小型到大型的表。

b. 按照以下步骤所述，设置 Catalog 元素：

i. 将 Prefix 元素设置为要用于标识和访问此模式的名称，例如， CDS。

前缀是不超过 21 个字符的字母数字字符串。有关其他可能的命名限制，请参见 Oracle 数据库文档。

- ii. 将 Password 元素设置为用于访问此模式的密码。
 - iii. 将 DBServer 元素设置为标识要使用的数据库服务器的字符串。
该字符串必须与用于步骤 a 中设置的 DBServer 元素的名称相匹配。
- c. 按照以下步骤所述，设置 Vending 元素：

- i. 将名称属性设置为用于标识 Vending Manager 的字符串。
为 Vending 元素指定的名称必须用于在 第 3-7 页上的第 3.1.3 节“设置部署配置文件”中设置的部署配置文件中的 vs.account.name 属性。使用 Catalog Manager 管理控制台创建 Vending Manager Server 帐户时，还必须将此名称用作帐户 ID。请参见第 3-14 页上的第 3.1.6.2 节“准备 Catalog Manager”。
- ii. 将 Prefix 元素设置为要用于标识和访问此模式的名称，例如，CDS。
前缀是不超过 21 个字符的字母数字字符串。有关其他可能的命名限制，请参见 Oracle 数据库文档。
- iii. 将 Password 元素设置为用于访问此模式的密码。
- iv. 将 DBServer 元素设置为标识要使用的数据库服务器的字符串。
该字符串必须与用于步骤 a 中设置的 DBServer 元素的名称相匹配。

提示 – 如果具有其他 Vending Manager 的信息，则可以在创建基本部署模式的同时创建所需的所有模式。对于要部署的每个其他 Vending Manager，请另外添加一个 Vending 元素，并为名称属性和 Prefix 元素提供唯一的值。另外添加一个 DBServer 元素，以使该 Vending Manager 使用不同的数据库服务器。

d. 将创建的文件保存在 \$CDS_HOME/cfg 目录中。

2. 为所使用的数据库服务器创建一个用于标识系统用户名和密码的数据库用户文件。
请将位于 \$CDS_HOME/cfg 目录中的样例文件 DBSysUser.xml 用作指南。以下代码示例显示了样例数据库用户文件。

代码示例 3-2 样例数据库用户文件

```
<DBUsers>

  <DBServer name="DBServer1">
    <User>system</User>
    <Password>dbsyspassword</Password>
  </DBServer>

  <!--
  <DBServer name="DBServer2">
```

代码示例 3-2 样例数据库用户文件（续）

```
<User>system2</User>
<Password>dbsyspassword2</Password>
</DBServer>
-->

</DBUsers>
```

注 – 该文件包含未加密的密码。请设置文件权限以限制对此文件的访问。

对于所使用的每个数据库服务器，请按照以下步骤所述创建一个 DBServer 元素：

a. 将名称属性设置为用于标识数据库服务器的字符串。

此名称必须与在 [步骤 1](#) 中创建的数据库配置文件中所使用的名称相匹配。

b. 将 User 元素设置为用于访问数据库服务器的系统用户名。

c. 将 Password 元素设置为与系统用户名相关联的密码。

d. 将创建的文件保存在 \$CDS_HOME/cfg 目录中。

3. 在 `init_env.sh` shell 脚本中设置以下变量：

- `DEFAULT_DB`。在 [步骤 1](#) 中创建的数据库配置文件的名称。没有给命令（该命令接受文件名作为参数）提供文件名时，此变量指定所使用的默认数据库配置文件。
- `DEFAULT_SYS_USER`。在 [步骤 2](#) 中创建的数据库用户文件的名称。没有给命令（该命令接受文件名作为参数）提供文件名时，此变量指定所使用的默认数据库用户文件。

有关此 shell 脚本的详细信息，请参见 [第 3-2 页上的第 3.1.1 节“编辑 `init_env` Shell 脚本”](#)。

4. （可选）要支持 iAppli 应用程序，请跳过此步骤。如果不需要支持 iAppli 应用程序，请从 `$CDS_HOME/dist/cds/database/cs` 目录中删除 `iappli.sql` 文件。

将 `iappli.sql` 文件的副本保存在系统中的其他位置。如果删除了此文件，但以后又决定支持 iAppli 应用程序，请参见《Sun Java™ System Content Delivery Server 5.1 集成和配置指南》以了解相应的说明。

5. 将 `ojdbc14.jar` 文件从数据库服务器复制到 `$CDS_HOME/dist/cds/lib/external` 目录中。

要使 Java 数据库连接 (Java Database Connectivity, JDBC™) API 驱动程序可供 Content Delivery Server 使用，必须执行此步骤。

6. 运行以下命令创建 Content Delivery Server 使用的用户帐户、创建模式并使用初始数据填充模式：

```
cdsi db init [-conf db-configuration-file] [-user db-user-file]
```

db-configuration-file 是数据库配置文件的名称，该文件包含用于创建模式的信息。如果未提供 *db-configuration-file*，则使用 `init_env.sh` 脚本中为 `DEFAULT_DB` 变量指定的值。*db-user-file* 是数据库用户文件的名称，该文件包含用于访问数据库的用户名和密码。如果没有提供 *db-user-file*，则使用 `init_env.sh` 脚本中为 `DEFAULT_SYS_USER` 变量指定的值。

注 – 如果不具有创建用户帐户的权限，则请让数据库管理员为您创建以下帐户：

```
catalog-prefix_ps_owner  
catalog-prefix_ps_app  
vending-prefix_vs_owner  
vending-prefix_vs_app
```

创建用户帐户后，请运行以下命令创建模式并使用初始数据填充这些模式，而不是运行

```
cdsi db init:  
cdsi db schemas [-conf db-configuration-file]  
cdsi db data [-conf db-configuration-file]
```

3.1.3 设置部署配置文件

设置数据库后，创建用于标识要部署的组件的部署配置文件。部署配置文件是每行都包含一个名称值对的标准属性文本文件，并用于将 Content Delivery Server 的设置传播到所有组件。附录 A 中介绍了部署配置文件。

Content Delivery Server 发行套件中包含可用于创建自己的文件的样例配置文件。这些样例配置文件位于 `$CDS_HOME/cfg` 目录中。与 Sun Java System Application Server 一起使用的样例文件是 `sn_sample.cfg`（请参见第 A-5 页上的第 A.2 节“与 Sun Java System Application Server 一起使用的样例文件”）。与 WebLogic Server 一起使用的样例文件是 `wl_sample.cfg`（请参见第 A-7 页上的第 A.3 节“与 WebLogic Server 一起使用的样例文件”）。

要将所有 Content Delivery Server 组件部署到一个服务器上，请按照下列步骤来创建配置文件：

1. 从某个样例文件中创建部署配置文件。

此文件的扩展名必须是 `.cfg`，而且必须位于 `$CDS_HOME/cfg` 目录中。

2. 设置常规属性。

有关这些属性的描述，请参见第 A-1 页上的第 A.1 节“配置属性”。

3. 确保未注释掉每个组件的 `prefix.include` 属性。

属性不能以井号 (#) 开头。

4. 将配置文件保存在 `$CDS_HOME/cfg` 目录中。

5. 编辑 `init_env.sh` shell 脚本，将 `DEFAULT_CONFIG` 变量设置为刚刚创建的部署配置文件的名称。

没有给命令（该命令接受文件名作为参数）提供文件名时，`DEFAULT_CONFIG` 变量指定所使用的默认部署配置文件。请参见第 3-2 页上的第 3.1.1 节“编辑 `init_env` Shell 脚本”。

3.1.4 部署

创建部署配置文件并设置环境变量后，请使用以下命令部署 Content Delivery Server 组件：

```
cdsi deploy [-deployment deployment-name] [-cfg cfg-file-name]
```

下表介绍了可选参数。

表 3-2 `deploy` 命令的可选参数

参数名	描述
<i>deployment-name</i>	用于标识部署的名称。将会在 <code>\$CDS_HOME/deployment</code> 目录下创建一个具有此名称的部署目录。该名称必须唯一。
<i>cfg-file-name</i>	用于说明部署的部署配置文件的名称。此文件必须位于 <code>\$CDS_HOME/cfg</code> 目录中。

注 – 如果省略这些参数，则使用 `init_env.sh` 脚本中为 `DEFAULT_DEPL` 和 `DEFAULT_CONFIG` 指定的值。

提示 – 要获得有关 `cdsi` 命令的其他信息，请输入 `cdsi help`。

部署过程将部署配置文件中标识的组件安装在指定的应用服务器上，并使用为部署指定的名称在 `$CDS_HOME/deployment` 目录下创建一个部署目录。例如，如果将部署命名为 `cds1`，则将创建 `$CDS_HOME/deployment/cds1` 目录。

3.1.5 设置应用服务器

成功部署 Content Delivery Server 后，要使用的应用服务器还需要其他一些配置。如果使用的是 Sun Java System Application Server，则必须设置 Message Queue 代理和应用服务器域。如果使用的是 WebLogic Server，则必须获取通过安全套接字层 (Secure Sockets Layer, SSL) 协议进行操作的证书以及更改应用服务器管理员的密码。

3.1.5.1 设置 Sun Java System Application Server

如果使用的是 Sun Java System Application Server，则必须设置 Message Queue 代理和应用服务器域。随 Content Delivery Server 提供了两个 shell 脚本供您使用：`deploymq.sh` 和 `deploy.sh`。这些脚本位于 `$CDS_HOME/deployment/deployment-name/sun/applications` 目录中。

要设置 Sun Java System Application Server，请执行以下操作：

1. 从命令行转到 `$CDS_HOME/deployment/deployment-name/sun/applications` 目录。
2. 要设置 Message Queue 代理，请输入 `deploymq.sh type`。
`type` 可以是 `server` 或 `client`。指定 `server` 可以创建新的 Message Queue 代理。指定 `client` 可以使用以前创建的 Message Queue 代理。
3. 输入提示您输入的信息。
默认值适用于没有专门要求的安装。要使用默认值，请在出现提示后按 `Enter` 键。下表说明了这些参数。

表 3-3 `deploymq.sh` 参数

提示	描述
Broker's Host Name (代理的主机名)	在其中运行 Message Queue 代理的主机的名称。默认值为 <code>localhost</code> 。
Broker's Port (代理的端口)	Message Queue 代理在其中进行监听的端口号。默认值为 <code>7676</code> 。
Broker's Name (代理的名称)	为 Message Queue 代理指定的实例名称。默认值为 <code>cdsbroker</code> 。 注 - 只有将 <code>server</code> 指定为 <code>type</code> 时才显示此提示。
Broker's Admin Name (代理的管理名称)	用于管理 Message Queue 代理的名称。默认值为 <code>admin</code> 。 注 - 只有将 <code>server</code> 指定为 <code>type</code> 时才显示此提示。
Broker's Admin Password (代理的管理密码)	用于管理 Messaging Queue 代理的密码。默认值为 <code>admin</code> 。 注 - 只有将 <code>server</code> 指定为 <code>type</code> 时才显示此提示。
Broker's User Name (代理的用户名)	用于访问 Message Queue 代理的名称。默认值为 <code>guest</code> 。
Broker's User Password (代理的用户密码)	用于访问 Message Queue 代理的密码。默认值为 <code>guest</code> 。

表 3-3 deploymq.sh 参数 (续)

提示	描述
Message Queue "bin" Directory Path (Message Queue "bin" 目录路径)	包含 imqbrokerd 和 imqcmd 命令文件的目录的全限定路径名。默认值为 \$SUN_AS_HOME/imq/bin。
Message Queue "lib" Directory Path (Message Queue "lib" 目录路径)	包含 fscontext.jar 和 imq.jar 文件的目录的全限定路径名。默认值为 \$SUN_AS_HOME/imq/lib。
Message Queue License Name (Message Queue 许可证名称)	正在使用的 Message Queue 版的许可证类型。有效值为： <ul style="list-style-type: none">• pe - 如果正在运行 Message Queue 的 Platform Edition。(这是与 Solaris 操作系统捆绑的版本。)• try - 如果正在运行 Message Queue 的 Platform Edition, 此版本支持在试用期内试用 Enterprise Edition 功能。• unl - 如果正在运行 Message Queue 的 Enterprise Edition。 <p>注 - 只有将 server 指定为 <i>type</i> 时才显示此提示。</p>

提示 - 要获得有关 deploymq.sh 命令的其他信息, 请输入: `deploymq.sh -h`。

除了设置 Message Queue 代理以外, 此脚本还将创建 broker.sh 脚本, 用于启动和停止 Message Queue 代理。此脚本位于 \$CDS_HOME/deployment/*deployment-name*/sun/applications 目录中。

4. 使用 \$CDS_HOME/deployment/*deployment-name*/sun/applications 目录中的 broker.sh start 命令启动代理。

5. 设置应用服务器域。

可以在创建应用服务器域时对使用 JavaServer Pages™ 技术 (JSP™ 页) 创建的页面进行预编译, 或在第一次访问该页面时对其进行编译。JSP 页预编译可能会使域创建过程增加一小时左右的时间。如果在第一次访问 JSP 页时对其进行编译, 则可能会延迟页面的初始显示过程。依据所需的 JSP 页编译时间, 使用以下某个命令设置应用服务器域:

- 要预编译 JSP 页, 请输入 `deploy.sh command`。
- 要在第一次访问页面时对其进行编译, 请输入 `deploy.sh command fast`。

command 是 create 或 recreate。如果以前没有为 Content Delivery Server 创建应用服务器域, 请指定 create 以创建新的域。指定 recreate 可替换以前为 Content Delivery Server 创建的现有应用服务器域。

6. 输入提示您输入的信息。

默认值适用于没有专门要求的安装。要使用默认值，请在出现提示后按 **Enter** 键。下表说明了这些参数。

表 3-4 `deploy.sh` 参数

提示	描述
Domain's Admin Name (域的管理名称)	用于访问 Sun Java System Application Server 管理控制台的名称。默认值为 <code>cdsadmin</code> 。
Domain's Admin Password (域的管理密码)	用于访问 Sun Java System Application Server 管理控制台的密码。默认值为 <code>cdsadmin</code> 。
Domains Master Password (域的主密码)	用于保护域的主密码。默认值为 <code>cdsmaster</code> 。
Domain's Admin Server Port (域的管理服务器端口)	Sun Java System Application Server 管理控制台在其中进行监听的端口号。默认值为 <code>4848</code> 。

提示 – 要获得有关 `deploy.sh` 命令的其他信息，请输入：
`deploy.sh -h`。

除了设置应用服务器域外，此脚本还将创建 `admin.sh` 脚本，用于启动和停止该域的 Sun Java System Application Server 管理服务器。此脚本位于 `$CDS_HOME/deployment/deployment-name/sun/applications` 目录中。

7. 使用 `$CDS_HOME/deployment/deployment-name/sun/applications` 目录中的 `broker.sh stop` 命令停止代理。

3.1.5.2 设置 WebLogic Server

如果使用的是 WebLogic Server，则必须获取通过 SSL 协议进行操作的证书以及更改应用服务器管理员的密码。如果将多个部署放在同一个主机上，则还需要更改所使用的 Cookie 名称以避免不正确的会话超时。

获取证书

Content Delivery Server 使用 SSL (SSL 要求证书) 连接到应用服务器。有关获得有效证书以用于此目的的信息，请参见 WebLogic Server 文档。

更改管理员密码

部署过程中，将为 Content Delivery Server 创建用户名和密码，以便在访问应用服务器时使用。出于安全方面的考虑，可能会需要更改默认密码。

要更改默认密码，请执行以下操作：

1. 启动 Content Delivery Server。

请参见第 5-1 页上的第 5.1 节“启动 Content Delivery Server”。

2. 在浏览器中，转到 `https://hostname:port/console`。

`hostname` 是为 `app.server.admin.hostip` 属性指定的名称，`port` 是为部署配置文件中的 `app.server.admin.port` 属性指定的端口号。

3. 以管理员身份登录。

默认的用户名是 `system`。默认密码是 `system`。

4. 按照 WebLogic Server 文档中的说明更改管理员密码。

5. 编辑 `$CDS_HOME/deployment/deployment-name/conf/wladmin.prop` 文件以设置以下属性：

- `WL_ADMIN_USER`。将此属性设置为管理员的用户名。
- `WL_ADMIN_PASSWORD`。将此属性设置为管理员的密码。

更改 Cookie 名称

如果使用具有不同端口号的同一主机对 Content Delivery Server 进行多个部署，则需要更改在第一次部署后添加的部署的 Cookie 名称。否则，如果通过浏览器访问主机上的多个部署，则可能会与为每个会话创建的 Cookie 发生冲突，这可能会导致在部署之间切换时出现会话超时消息。

例如，如果将两个 Vending Manager 部署到同一个主机上并登录到每个 Vending Manager 的管理控制台上，则为第二个 Vending Manager 的会话创建的 Cookie 将覆盖第一个 Vending Manager 的会话的 Cookie。在切换回第一个 Vending Manager 的管理控制台时，将会出现会话超时消息，因为 Cookie 对该会话不再有效。

可以为主机（已包含一个部署）上的每个其他部署的 Cookie 提供不同的名称以防止发生冲突。要更改 Cookie 名称，请执行以下操作：

1. 停止该部署。

使用 `cdse stopserver -depl deployment-name` 命令停止所使用的部署中的所有组件。

2. 编辑包含部署中每个组件的 Cookie 名称的文件。

Cookie 名称是在以下文件中定义的，其中 *root* 是 `$CDS_HOME/deployment/deployment-name/weblogic/domains/server-domain/applications`；*server-domain* 是在部署配置文件中为 `app.server.domain` 属性指定的值：

- **Catalog Manager** - *root*/admin/WEB-INF/weblogic.xml
- **Developer Portal** - *root*/developer/WEB-INF/weblogic.xml
- **Vending Manager** - *root*/vsadmin/WEB-INF/weblogic.xml
- **Subscriber Portal** - *root*/subscriber/WEB-INF/weblogic.xml
- **Fulfillment Manager** - *root*/fs/WEB-INF/weblogic.xml
- **事件服务** - *root*/es/WEB-INF/weblogic.xml
- **消息传送服务** - *root*/ms/WEB-INF/weblogic.xml
- **通知服务** - *root*/ns/WEB-INF/weblogic.xml
- **后付费服务** - *root*/pps/WEB-INF/weblogic.xml

不需要更改未列出的组件，例如，目录搜索服务。

在部署的每个 `weblogic.xml` 文件中，请执行以下操作：

a. 找到以下语句：

```
<param-name>CookieName</param-name>
<param-value>SID</param-value>
```

b. 将为 `param-value` 元素指定的字符串更改为唯一的值。

部署中的每个文件可以使用相同的值，例如 `SID2`。

c. 保存所做的更改。

3. 重新启动部署。

使用 `cdse startserver -depl deployment-name` 命令启动所使用的部署中的所有组件。

3.1.6 准备 Content Delivery Server 的使用

使部署可用之前，请配置功能并设置要使用的计划、帐户和设备。本节介绍了所需的准备工作。

3.1.6.1 配置 Content Delivery Server

可以按照需要配置 Content Delivery Server 的很多功能。例如，可以设置页面上显示的项目数或者对 Catalog Manager、Vending Manager、Subscriber Portal 和 Developer Portal 而言缓存刷新的速率。

《Sun Java™ System Content Delivery Server 5.1 集成和配置指南》介绍了如何配置要使用的功能。使部署可供使用之前，请完成配置并准备组件。

3.1.6.2 准备 Catalog Manager

使用 Catalog Manager 管理控制台准备 Catalog Manager 的以下元素。有关启动 Catalog Manager 的信息，请参见第 5-1 页上的第 5.1 节“启动 Content Delivery Server”。有关使用管理控制台的信息，请参见《Sun Java™ System Content Delivery Server 5.1 参考手册》。

1. 准备内容。
 - a. 定义要支持的内容类型。
 - b. 定义要支持的 MIME 类型。
 - c. 为要保护的内容类型指定 DRM 方法。
 - d. 选择要使用的价格模型。
 - e. 创建价格选项。
 - f. 定义要使用的类别。
2. 准备设备。
 - a. 定义要使用的模板。
 - b. 添加要支持的设备。
 - c. 提交尚未在 Content Delivery Server 中并且是设备运行内容所需的库。
 - d. 指定默认设备。
 - e. 定义要支持的语言环境。例如，使用下表中的值来添加加拿大法语和简体中文支持。

表 3-5 样例语言环境

语言	语言代码	国家/地区代码
加拿大法语	fr	CA
简体中文	zh	CN

3. 设置 Vending Manager Server 帐户。

注 – 对于基本部署，Catalog Manager 通过部署配置文件中的信息来创建 Vending Manager Server 帐户。对于 Vending Manager 部署，您必须创建 Vending Manager Server 帐户。在成功创建 Vending Manager Server 帐户之前，Vending Manager 将同步错误记录在 Content Delivery Server 日志中。

- a. 选择要使用的售卖计划。
- b. 设置默认的售卖计划。

注 – Content Delivery Server 附带的默认售卖计划只允许访问分配给“特色”类别的内容。

- ### 4. 准备开发者帐户。
- a. 添加要使用的开发者计划和配置文件。
 - b. 设置默认的开发者计划。

注 – Content Delivery Server 附带的默认开发者计划不限制开发者在提交的应用程序中可以使用的 API。

- c. 添加所需的开发者帐户。

3.1.6.3 准备 Vending Manager

使用 Vending Manager 管理控制台准备 Vending Manager 的以下元素。有关启动 Catalog Manager 的信息，请参见第 5-1 页上的第 5.1 节“启动 Content Delivery Server”。有关使用管理控制台的信息，请参见《Sun Java™ System Content Delivery Server 5.1 参考手册》。

1. 准备内容。
 - a. 定义要使用的类别。
 - b. 设置首选项。
 - c. 设置储存选项。
 - d. 使用语言环境的价格来设置价格选项。
2. 准备订户帐户。
 - a. 添加要使用的订户计划。

- b. 设置默认的订户计划。

注 – Content Delivery Server 附带的默认订户计划允许访问所有类别中的内容。

- c. 添加所需的任何订户帐户。

3.2 Catalog Manager 部署

本节介绍部署 Catalog Manager 的步骤。部署 Catalog Manager 时，还会部署事件服务、消息传送服务、监视服务以及目录搜索服务。继续操作之前，请确保按照第 2 章中所述设置环境。

3.2.1 编辑 `init_env` Shell 脚本

按照第 3-2 页上的第 3.1.1 节“编辑 `init_env` Shell 脚本”中所述设置 `init_env.sh` 脚本。

`init_env.sh` 脚本包含默认部署、默认数据库配置文件和默认部署配置文件的设置。如果主机包含多个部署，则必须确定将哪个部署指定为默认部署。通过默认设置，可以运行 Content Delivery Server `execute` 命令 (`cdse`) 和 Content Delivery Server `install` 命令 (`cdsi`)，而无需输入所有参数。如果找不到所需的参数，则使用在 `init_env.sh` 脚本中指定的默认值。

3.2.2 设置数据库

要部署 Catalog Manager，请为 Catalog Manager 及其关联的事件服务创建模式。按照第 3-2 页上的第 3.1.2 节“设置数据库”中所述，设置数据库配置文件和数据库用户文件。需要 DBServer 和 Catalog 元素。还需要使用与系统中的任一 Vending Manager 对应的 Vending 元素，才能成功设置 DB 连接池。

可以使用以下命令为 Catalog Manager 创建模式：

```
cdsi db init [-conf db-configuration-file] [-user db-user-file] -cs
```

`db-configuration-file` 是数据库配置文件的名称，该文件包含用于创建模式的信息。如果未提供 `db-configuration-file`，则使用 `init_env.sh` 脚本中为 `DEFAULT_DB` 变量指定的值。`db-user-file` 是数据库用户文件的名称，该文件包含用于访问数据库的用户名和密码。如果没有提供 `db-user-file`，则使用 `init_env.sh` 脚本中为 `DEFAULT_SYS_USER` 变量指定的值。

注 – 如果不具有创建用户帐户的权限，则请让数据库管理员为您创建以下帐户：
catalog-prefix_ps_owner
catalog-prefix_ps_app

创建用户帐户后，请运行以下命令创建模式并使用初始数据填充这些模式，而不是运行

```
cdsi db init;  
cdsi db schemas -conf db-configuration-file -cs  
cdsi db data -conf db-configuration-file -cs
```

3.2.3 设置部署配置文件

设置数据库后，创建用于标识要部署的组件的部署配置文件。有关配置文件的概述，请参见第 3-7 页上的第 3.1.3 节“设置部署配置文件”。

可以使用以下指导来创建部署配置文件。部署 Catalog Manager 的同时，还会部署事件服务、消息传送服务以及监视服务的实例。

1. 从某个样例文件中创建部署配置文件。

此文件的扩展名必须是 .cfg，而且必须位于 \$CDS_HOME/cfg 目录中。

2. 设置常规属性。

有关这些属性的描述，请参见第 A-1 页上的第 A.1 节“配置属性”。

3. 确保未注释掉以下属性，即这些属性名不能以井号 (#) 开头：

- cs.include
- dp.include
- es.include
- ms.include
- mos.include
- css.include

4. 确保注释掉以下属性，即属性名以井号 (#) 开头：

- vs.include
- sp.include
- fs.include
- ns.include
- pps.include
- vss.include

5. 将配置文件保存在 \$CDS_HOME/cfg 目录中。

6. 编辑 `init_env.sh` shell 脚本，将 `DEFAULT_CONFIG` 变量设置为刚刚创建的部署配置文件的名称。

如果要将 Catalog Manager 部署到以前部署了其他 Content Delivery Server 组件的主机上，则将 `DEFAULT_CONFIG` 设置为要用作默认值的配置文件的名称。

没有给命令（该命令接受文件名作为参数）提供文件名时，`DEFAULT_CONFIG` 变量指定所使用的默认部署配置文件。请参见第 3-2 页上的第 3.1.1 节“编辑 `init_env Shell` 脚本”。

3.2.4 部署

有关部署 Catalog Manager 和支持组件的说明，请参见第 3-8 页上的第 3.1.4 节“部署”。

3.2.5 设置应用服务器

按照第 3-8 页上的第 3.1.5 节“设置应用服务器”中所述，设置所使用的应用服务器。如果使用的是 Sun Java System Application Server，并且在以前的部署中创建了 Message Queue 代理，请在运行 `deploymq.sh` 命令时输入 `client` 作为 `type`，从而创建此部署的 Message Queue 代理客户机。

3.2.6 准备 Catalog Manager 以供使用

有关配置要使用的功能的信息，请参见《Sun Java™ System Content Delivery Server 5.1 集成和配置指南》。使用 Catalog Manager 之前，必须部署至少一个 Vending Manager 并创建一个 Vending Manager Server 帐户。

有关 Catalog Manager 所需的其他准备工作，请参见第 3-14 页上的第 3.1.6.2 节“准备 Catalog Manager”。

3.3 Vending Manager 部署

本节介绍部署 Vending Manager 的步骤。部署 Vending Manager 时，还会部署 Fulfillment Manager 以及除目录搜索服务以外的所有服务。

部署 Vending Manager 之前，必须部署 Catalog Manager。可以将 Catalog Manager 作为基本部署（请参见第 3-1 页上的第 3.1 节“基本部署”）、Catalog Manager 部署（请参见第 3-16 页上的第 3.2 节“Catalog Manager 部署”）或高级部署（请参见第 3-22 页上的第 3.4 节“高级部署”）的一部分进行部署。

可以在已部署 Content Delivery Server 其他组件的主机或者在未部署任何其他组件的主机上部署 Vending Manager。如果主机包含现有部署，则使用现有安装和 CDS_HOME 目录。如果主机不包含现有部署，则请确保在继续操作之前按照第 2 章中所述设置环境。

3.3.1 编辑 init_env Shell 脚本

如果要将 Vending Manager 部署到未部署其他组件的主机上，请按照第 3-2 页上的第 3.1.1 节“编辑 init_env Shell 脚本”中所述设置 init_env.sh 脚本。

如果要将 Vending Manager 部署到已经部署了其他组件的主机上，请确认 \$CDS_HOME/bin 目录中的 init_env.sh shell 脚本包含 SUN_AS_HOME、JAVA_HOME、BEA_HOME 和 WL_HOME 变量的正确设置。该主机上的所有部署都使用这些设置。

init_env.sh 脚本还包含默认部署、默认数据库配置文件、默认数据库用户文件以及默认部署配置文件的设置。如果主机包含多个部署，则必须确定将哪个部署指定为默认部署。通过默认设置，可以运行 Content Delivery Server execute 命令 (cdse) 和 Content Delivery Server install 命令 (cdsi)，而无需输入所有参数。如果找不到所需的参数，则使用在 init_env.sh 脚本中指定的默认值。

3.3.2 设置数据库

要部署 Vending Manager，需要为该 Vending Manager 创建模式。此模式可以与 Catalog Manager 模式位于相同的数据库服务器上，也可以位于不同的服务器上。

如果在设置基本部署的数据库时为当前正在部署的 Vending Manager 创建了模式，则无需对此数据库进行进一步操作。继续执行第 3-20 页上的第 3.3.3 节“设置部署配置文件”。

如果没有为 Vending Manager 创建模式，则按照第 3-2 页上的第 3.1.2 节“设置数据库”中所述设置数据库配置文件和数据库用户文件。仅需要 DBServer 和 Vending 元素。

可以使用以下命令为 Vending Manager 创建模式：

```
cdsi db init [-conf db-configuration-file] [-user db-user-file] -vs vending-name
```

db-configuration-file 是数据库配置文件的名称，该文件包含用于创建模式的信息。如果未提供 *db-configuration-file*，则使用 init_env.sh 脚本中为 DEFAULT_DB 变量指定的值。*db-user-file* 是数据库用户文件的名称，该文件包含用于访问数据库的用户名和密码。如果没有提供 *db-user-file*，则使用 init_env.sh 脚本中为 DEFAULT_SYS_USER 变量指定的值。开关 -vs 表示仅创建指定 Vending Manager 的模式。

注 – 如果不具有创建用户帐户的权限，则请让数据库管理员为您创建以下帐户：
vending-prefix_vs_owner
vending-prefix_vs_app

创建用户帐户后，请运行以下命令创建模式并使用初始数据填充这些模式，而不是运行 `cdsi db init`：

```
cdsi db schemas -conf db-configuration-file -vs vending-name  
cdsi db data -conf db-configuration-file -vs vending-name
```

3.3.3 设置部署配置文件

设置数据库后，创建用于标识要部署的组件的部署配置文件。有关配置文件的概述，请参见第 3-7 页上的第 3.1.3 节“设置部署配置文件”。

如果以前部署了 Catalog Manager 并希望将其他 Vending Manager 添加到配置中，请按照下列步骤创建部署配置文件。部署 Vending Manager 的同时，还会部署 Fulfillment Manager 和所有 Content Delivery Server 服务的实例。

1. 从某个样例文件中创建部署配置文件。

此文件的扩展名必须是 `.cfg`，而且必须位于 `$CDS_HOME/cfg` 目录中。如果要部署到包含其他部署的主机，则请为此部署创建新的配置文件以避免混淆。

2. 设置常规属性。

有关这些属性的描述，请参见第 A-1 页上的第 A.1 节“配置属性”。

3. 确保未注释掉以下属性，即这些属性名不能以井号 (#) 开头：

- `vs.include`
- `sp.include`
- `fs.include`
- `es.include`
- `ms.include`
- `ns.include`
- `mos.include`
- `pps.include`
- `vss.include`

4. 确保通过在属性名前面添加井号 (#) 来注释掉以下属性：

- `cs.include`
- `dp.include`
- `css.include`

5. 将配置文件保存在 `$CDS_HOME/cfg` 目录中。
6. 编辑 `init_env.sh` shell 脚本，将 `DEFAULT_CONFIG` 变量设置为刚刚创建的部署配置文件的名称。

如果要将 Vending Manager 部署到已部署了其他 Content Delivery Server 组件的主机上，请将 `DEFAULT_CONFIG` 设置为要用作默认值的配置文件的名称。

没有给命令（该命令接受文件名作为参数）提供文件名时，`DEFAULT_CONFIG` 变量指定所使用的默认部署配置文件。请参见第 3-2 页上的第 3.1.1 节“编辑 `init_env` Shell 脚本”。

3.3.4 部署

有关部署 Vending Manager 和支持组件的说明，请参见第 3-8 页上的第 3.1.4 节“部署”。

3.3.5 设置应用服务器

按照第 3-8 页上的第 3.1.5 节“设置应用服务器”中所述，设置所使用的应用服务器。如果使用的是 Sun Java System Application Server，并且在以前的部署中创建了 Message Queue 代理，请在运行 `deploymq.sh` 命令时输入 `client` 作为 `type`，从而创建此部署的 Message Queue 代理客户机。

3.3.6 准备 Vending Manager 以供使用

有关配置要使用的功能的信息，请参见《Sun Java™ System Content Delivery Server 5.1 集成和配置指南》。

使用刚刚部署的 Vending Manager 之前，必须在 Catalog Manager 中为其创建一个 Vending Manager Server 帐户。有关信息，请参见 Catalog Manager 管理控制台联机帮助中的“添加 Vending Manager Server 帐户”主题。创建 Vending Manager Server 帐户时，必须将帐户 ID 设置为用于部署配置文件中 `vs.account.name` 属性的值。帐户密码必须设置为用于部署配置文件中 `vs.account.password` 属性的值。这些值还必须与为 Vending Manager 创建数据库结构时使用的名称和密码相匹配。

注 – 在 Catalog Manager 中成功创建 Vending Manager Server 帐户之前，Vending Manager 将同步错误记录在 Content Delivery Server 日志中。

有关 Vending Manager 所需的其他准备工作，请参见第 3-15 页上的第 3.1.6.3 节“准备 Vending Manager”。

3.4 高级部署

对一个服务器上的所有 Content Delivery Server 组件执行了基本部署后再执行一个或多个其他的 Vending Manager 部署，应可以满足大多数企业的需要。如果有特殊需要（例如限制部署方式的有限资源或安全性要求），则 Content Delivery Server 支持高度定制部署。

通过高级部署，可以在配置中有选择地安装满足需要的组件。由于高级部署具有独特的特性，因此本节只提供了设置部署的指导，而不是具体说明。

3.4.1 编辑 `init_env` Shell 脚本

如果要将组件部署到未部署其他组件的主机上，请设置 `init_env.sh` 脚本，如第 3-2 页上的第 3.1.1 节“编辑 `init_env` Shell 脚本”中所述。

如果要将组件部署到已经部署了其他组件的主机上，请确认 `$CDS_HOME/bin` 目录中的 `init_env.sh` shell 脚本包含 `SUN_AS_HOME`、`JAVA_HOME`、`BEA_HOME` 和 `WL_HOME` 变量的正确设置。该主机上的所有部署都使用这些设置。

`init_env.sh` 脚本还包含默认部署、默认数据库配置文件和默认的部署配置文件的设置。如果主机包含多个部署，则必须确定将哪个部署指定为默认部署。通过默认设置，可以运行 Content Delivery Server `execute` 命令 (`cdse`) 和 Content Delivery Server `install` 命令 (`cdsi`)，而无需输入所有参数。如果找不到所需的参数，则使用在 `init_env.sh` 脚本中指定的默认值。

3.4.2 设置数据库

为配置中的 Catalog Manager 和每个 Vending Manager 创建所需的模式。根据需要，数据库可以共享数据库服务器或使用单独的服务器。

要设置数据库，请执行以下操作：

1. 按照第 3-2 页上的第 3.1.2 节“设置数据库”中所述，创建数据库配置文件和数据库用户文件。

为所有要部署的 Vending Manager 创建条目。

注 – 数据库配置文件和数据库用户文件包含未加密的密码。请设置文件权限以限制对这些文件的访问。

2. 将数据库配置文件和数据库用户文件复制到要部署 Content Delivery Server 组件的所有主机。

将此文件放在每个主机的 `$CDS_HOME/cfg` 目录中。

3. 在每个主机上的 `init_env.sh` shell 脚本中设置以下变量：

- `DEFAULT_DB`。在步骤 1 中创建的数据库配置文件的名称。没有给命令（该命令接受文件名作为参数）提供文件名时，此变量指定所使用的默认数据库配置文件。
- `DEFAULT_SYS_USER`。在步骤 1 中创建的数据库用户文件的名称。没有给命令（该命令接受文件名作为参数）提供文件名时，此变量指定所使用的默认数据库用户文件。

有关此 shell 脚本的详细信息，请参见第 3-2 页上的第 3.1.1 节“编辑 `init_env Shell` 脚本”。

4. 将 `ojdbc14.jar` 文件从数据库服务器复制到 `$CDS_HOME/dist/cds/lib/external` 目录中。

要使 JDBC API 驱动程序可供 Content Delivery Server 使用，必须执行此步骤。

5. 创建在数据库配置文件中定义的模式。

从要部署 Catalog Manager 的主机运行以下命令。

```
cdsi db init [-conf db-configuration-file] [-user db-user-file]
```

`db-configuration-file` 是数据库配置文件的名称，该文件包含用于创建模式的信息。如果未提供 `db-configuration-file`，则使用 `init_env.sh` 脚本中为 `DEFAULT_DB` 变量指定的值。`db-user-file` 是数据库用户文件的名称，该文件包含用于访问数据库的用户名和密码。如果没有提供 `db-user-file`，则使用 `init_env.sh` 脚本中为 `DEFAULT_SYS_USER` 变量指定的值。

以后要部署其他 Vending Manager，请按照第 3-19 页上的第 3.3.2 节“设置数据库”中所述创建所需的其他模式。

3.4.3 设置部署配置文件

设置数据库后，创建用于标识要部署的组件的部署配置文件。有关配置文件的概述，请参见第 3-7 页上的第 3.1.3 节“设置部署配置文件”。

部署配置文件用于标识要部署的组件。为避免混淆，请为每个部署创建一个单独的配置文件。要创建配置文件，请执行以下操作：

1. 从某个样例文件中创建部署配置文件。

此文件的扩展名必须是 `.cfg`，而且必须位于 `$CDS_HOME/cfg` 目录中。

2. 设置常规属性。

有关这些属性的描述，请参见第 A-1 页上的第 A.1 节“配置属性”。

3. 对于部署中包含的每个组件，请确保未注释掉 `prefix.include` 属性。
属性不能以井号 (#) 开头。
4. 对于部署中未包含的每个组件，请确保注释掉 `prefix.include` 属性。
属性必须以井号 (#) 开头。
5. 将配置文件保存在 `$CDS_HOME/cfg` 目录中。
6. 编辑 `init_env.sh` shell 脚本，将 `DEFAULT_CONFIG` 变量设置为刚刚创建的部署配置文件的名称。
如果要将组件部署到已部署了其他 Content Delivery Server 组件的主机上，请将 `DEFAULT_CONFIG` 设置为要用作默认值的配置文件的名称。
没有给命令（该命令接受文件名作为参数）提供文件名时，`DEFAULT_CONFIG` 变量指定所使用的默认部署配置文件。有关此 shell 脚本的信息，请参见第 3-2 页上的第 3.1.1 节“编辑 `init_env` Shell 脚本”。

3.4.4 部署

有关部署的说明，请参见第 3-8 页上的第 3.1.4 节“部署”。

3.4.5 设置应用服务器

按照第 3-8 页上的第 3.1.5 节“设置应用服务器”中所述，设置所使用的应用服务器。如果使用的是 Sun Java System Application Server，请根据以下指导设置 Message Queue 代理：

- 部署 Catalog Manager 时，应在运行 `deploymq.sh` 命令时输入 `server` 作为 `type`，从而创建新的 Message Queue 代理。
- 如果独立于 Catalog Manager 部署 Vending Manager 或服务，则该组件可以共享现有的 Message Queue 代理或创建一个新的代理。
要共享现有的 Message Queue 代理，请在运行 `deploymq.sh` 命令时输入 `client` 作为 `type`。要创建新的 Message Queue 代理，请指定 `server`。

3.4.6 准备高级部署以供使用

有关配置所部署组件的功能的信息，请参见《Sun Java™ System Content Delivery Server 5.1 集成和配置指南》。

部署 Catalog Manager 之后，请参见第 3-14 页上的第 3.1.6.2 节“准备 Catalog Manager”以了解设置信息。部署 Vending Manager 之后，请参见第 3-21 页上的第 3.3.6 节“准备 Vending Manager 以供使用”。

第 4 章

配置访问

部署 Content Delivery Server 后，应配置对系统的访问权限。您可能需要在防火墙后面运行某些组件，或者阻止外部系统进行访问。您可能还需要支持容易记住的 URL。

本节包括以下主题：

- 针对防火墙配置 Content Delivery Server
- 控制外部系统的访问
- 配置别名转发
- 控制对 Sun Java System Application Server 的访问

4.1 针对防火墙配置 Content Delivery Server

Content Delivery Server 将会生成 URL，以便从外部实体（如订户设备）访问其组件。如果在防火墙后面运行 Content Delivery Server，则必须能够从防火墙外部访问这些 URL。使用的初始值是在部署配置文件中为 `prefix.external.uri` 属性设置的值，其中 `prefix` 是用于标识组件的字符串。如果在部署前未设置这些属性，请在 `$CDS_HOME/deployment/deployment-name/conf` 目录下的 `CDS.properties` 文件中设置以下属性。

- `default.external.catalogmanager.uri`。将此属性设置为 Catalog Manager Web 服务器的已发布的外部 URI，例如：

```
default.external.catalogmanager.uri=http://webhost.domain.com:80/admin/
```

- `default.external.developerportal.uri`。将此属性设置为 Developer Portal Web 服务器的已发布的外部 URI，例如：

```
default.external.developerportal.uri=http://webhost.domain.com:80/developer/
```

- `default.external.fulfillmentmanager.uri`。将此属性设置为 Fulfillment Manager Web 服务器的已发布的外部 URI，例如：

```
default.external.fulfillmentmanager.uri=http://webhost.domain.com:80/fs/
```

- `default.external.vendingmanager.uri`。将此属性设置为 Vending Manager Web 服务器的已发布的外部 URI，例如：

```
default.external.vendingmanager.uri=http://webhost.domain.com:80/vsadmin/
```

- `default.external.subscriberportal.uri`。将此属性设置为 Subscriber Portal Web 服务器的已发布的外部 URI，例如：

```
default.external.subscriberportal.uri=http://webhost.domain.com:80/subscriber/
```

此外，还必须将 Vending Manager Server 帐户的网络 ID 设置为运行 Web 服务器的主机的内部 IP 地址。使用 Catalog Manager 管理控制台可以更新 Vending Manager Server 帐户。有关详细信息，请参见联机帮助中的“添加 Vending Manager Server 帐户”主题。

提示 – 要验证 Catalog Manager 是否可以访问 Vending Manager，请在运行 Catalog Manager 的主机上启动浏览器。将浏览器指向 `http://vending-manager-ip`。如果无法显示页面，则必须更正为 Vending Manager Server 帐户（与要访问的 Vending Manager 相关联）指定的域名。

4.2 控制外部系统的访问

通过由内部和外部 URL 属性（在部署配置文件中设置）生成的 URL，可以访问 Content Delivery Server 组件（请参见附录 A）。要禁止未经授权的访问，请在访问控制文件中设置规则，以定义允许什么样的访问权限。默认访问控制文件允许所有请求者进行访问。

注 – 无需停止服务器即可更新访问控制文件。如果在 Content Delivery Server 运行时更改了该文件，将在下一次向服务器发出请求时强制重新装入该文件。新的规则将立即生效。

4.2.1 访问控制文件

`$CDS_HOME/deployment/deployment-name/conf` 目录中的 `access_control.config` 文件用于确定何时允许访问 Content Delivery Server 组件。此文件包含以下类型的语句：

- **访问控制列表** - 与请求源地址相匹配的一个或多个地址模式。

访问控制列表是由关键字 `acl` 和唯一标识列表的名称标识的。该列表包含一个或多个地址模式。语句的格式为 `acl acl-name {address-patterns}`。

`acl-name` 是由数字和字母组成的字符串（包括下划线），并且不能与任何语句中使用的任何关键字相匹配。`address-patterns` 是一个或多个主机名、IP 版本 4 (IPv4) 地址掩码或 IP 版本 6 (IPv6) 地址掩码。有关对 Backus Naur Form (BNF) 表示法中此语句的完整描述，请参见附录 B。

- **URI 组** - 与请求目标相匹配的 Content Delivery Server 组件的一个或多个相对路径。

URI 组是由关键字 `uri_group` 和唯一标识该组的名称标识的。该组包含一个或多个相对路径。语句的格式为 `uri_group group-name {paths}`。

`group-name` 是由数字和字母组成的字符串（包括下划线），并且不能与任何语句中使用的任何关键字相匹配。`paths` 是包含 Servlet 上下文名称的一个或多个绝对 URI，这些 URI 表示为正则表达式。有关对 BNF 表示法中此语句的完整描述，请参见附录 B。

- **规则** - 定义允许访问和拒绝访问的请求的语句。

规则是由关键字 `access` 标识的，并且包含以下信息：

- 请求访问的 Content Delivery Server 资源的相对路径
- 请求中使用的协议
- 请求中使用的方法
- 从中接收请求的 IP 地址
- 请求的状态（允许或拒绝）

引用 URI 组的规则的格式为：

`access group group-name protocol protocol method method {control-list}`。
`group-name` 是为文件中的某个 URI 组指定的名称。`protocol` 是请求中使用的协议，它是 `http`、`https` 或 `any`。`method` 标识所请求的操作，它是 `get`、`post`、`put` 或 `any`。`control-list` 是一个语句列表，用于表示允许访问和拒绝访问的地址。

有关对 BNF 表示法中此语句的完整描述，请参见附录 B。有关如何解释规则的信息，请参见下一节。

4.2.2 访问控制进程

收到要求访问 Content Delivery Server 组件的请求时，将使用访问控制文件中的规则来确定是否允许访问。

如果满足以下条件，则与规则相匹配：

- 请求访问的组件与为规则所引用的 URI 或 URI 组指定的路径相匹配。
- 请求中使用的协议与规则中指定的协议相匹配。
- 请求中使用的方法与规则中指定的方法相匹配。
- 请求源与控制列表中引用的地址相匹配。

将按照规则在文件中出现的顺序来处理它们。除非遇到关键字 `final`，否则会将请求与文件中的所有规则进行比较。最后一个匹配的规则确定是否允许访问。通常，您需要在文件结尾处放置限制性较强的规则。

`final` 关键字将停止匹配进程。此关键字可以出现在规则开头，也可以出现在规则中任何控制列表的结尾。如果此关键字出现在规则的开头，则当请求源与规则的任何控制列表中的地址相匹配时，都会停止匹配进程。如果此关键字出现在某个控制列表的结尾，则仅当请求源与该控制列表中的地址相匹配时，才会停止匹配进程。

当匹配进程停止时，将由包含请求源地址的控制列表中的关键字来确定请求状态。如果关键字为 `allow`，则允许访问。如果关键字为 `deny`，将拒绝访问并返回错误代码 `404`。

如果没有匹配的规则，将允许访问。要拒绝所有与任何规则均不匹配的请求进行访问，请将以下规则添加到规则的开头：

```
access uri /\.*' protocol any method any {deny any;}
```

4.3 配置别名转发

别名转发是一种技术，用于接受 URL 并将用户重定向到其他位置。通过此功能，可以为订户提供容易记忆的 URL，并可将请求重定向到正确位置。还可以使用此功能来截取请求，并将其重定向以进行特殊处理。

Content Delivery Server 的 URL 映射是在 `aliasforwarding.xml` 文件中定义的。该文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。要使用其他文件，请在 `$CDS_HOME/deployment/deployment-name/conf` 目录下的 `DefaultWebApp.properties` 文件中设置 `filter.AliasForwarding.file` 属性。

别名转发文件是一个过滤器，它使用 XML 来描述收到 URL 时执行的重定向。下表描述了此文件中的元素。

表 4-1 别名转发文件元素

元素	描述
AliasForwarding	包装整个文件。文件中只显示此元素的一个实例，该实例包含所有其他元素。
AliasMap	描述单个映射。此元素包含 <code>URLPattern</code> 、 <code>TargetContext</code> 和 <code>TargetPath</code> 元素。

表 4-1 别名转发文件元素（续）

元素	描述
URLPattern	用于匹配传入 URL 的模式。如果字符串 /* 出现在模式的开头或结尾，则会将其解释为通配符。
TargetContext	标识 Web 应用程序的上下文根路径，请求将被重定向到该 Web 应用程序。指定的值必须是现有的上下文根路径，它是随部署了 Content Delivery Server 的应用服务器一起部署的。默认情况下，Content Delivery Server 将应用程序部署在以下上下文根路径中： <ul style="list-style-type: none">• /DefaultWebApp• /admin• /vsadmin• /developer• /subscriber• /fs 如果没有为此元素提供值，则使用根 URI。
TargetPath	位于上下文根路径下的路径，请求将被重定向到该路径。此值必须为绝对路径，并且必须对指定的上下文根路径有效。

以下代码示例显示了一个样例映射文件。

代码示例 4-1 样例 aliasForwarding.xml 文件

```
<AliasForwarding>
  <AliasMap>
    <URLPattern>/dev</URLPattern>
    <TargetContext>/developer</TargetContext>
    <TargetPath>/main</TargetPath>
  </AliasMap>

  <AliasMap>
    <URLPattern>/sub</URLPattern>
    <TargetContext>/subscriber</TargetContext>
    <TargetPath>/desktop_home.do</TargetPath>
  </AliasMap>
</AliasForwarding>
```

别名转发过滤器将部署到上下文根路径 /DefaultWebApp 中，Content Delivery Server 将该根路径设置为 Content Delivery Server 请求的默认应用程序。不包含与已知上下文根路径相匹配的 URL 的请求将被发送到默认应用程序，并由该过滤器进行处理。包含与已知上下文根路径相匹配的 URL 的请求将直接发送到应用程序，不由此过滤器进行处理。例如，基于代码示例 4-1 中的映射，对 /sub 的请求将由此过滤器进行处理，然后重新路由到 /subscriber。但是，对 /subscriber 的请求将直接发送到应用程序，而不由此过滤器进行处理。

当 Content Delivery Server 的默认应用程序收到请求时，会根据以下规则将传入 URL 与别名转发文件中的每个 URLPattern 元素进行匹配：

- 如果传入 URL 未以正斜杠 (/) 开头，则假定以正斜杠开头。
- 如果模式或传入 URL 未以正斜杠 (/) 或文件扩展名结尾，则假定以正斜杠结尾。
- 如果模式为 /*，则传入 URL 始终是匹配项。
- 如果模式多于三个字符，以 / 开头并以 /* 结尾，则正斜杠之间的字符串将与传入 URL 的开头相匹配。例如，模式 /PATH/* 与 URL /PATH 和 /PATH/anything 相匹配。
- 如果模式以 /*. 开头，则句点后面的字符串将与传入 URL 的结尾相匹配。例如，模式 /*.EXT 将与任何以 .EXT 结尾的 URL 相匹配。
- 如果模式以 *. 开头，则句点后面的字符串将与传入 URL 的结尾相匹配。例如，模式 *.EXT 将与任何以 .EXT 结尾的 URL 相匹配。
- 如果模式以 / 开头并以 / 结尾，则正斜杠之间的字符串将与传入 URL 相匹配。例如， /PATH/ 与 URL /PATH 和 /PATH/ 相匹配。
- 如果模式以 / 开头，则正斜杠后面的字符串将与传入 URL 相匹配。例如， /PATH 与 URL /PATH 和 /PATH/ 相匹配。

找到匹配项后，进程将停止。然后，请求将被重定向到通过附加 TargetContext 和 TargetPath（与传入请求匹配的 URLPattern 相关联）值得到的位置。如果未找到匹配项，则将请求返回到应用服务器。

根据定义映射的方式，在找到匹配项时，新位置也可能由过滤器进行处理。请确保未创建循环映射，此类映射只能生成由过滤器处理的 URL，而不会生成被重定向到实际位置的 URL。

4.4 控制对 Sun Java System Application Server 的访问

Sun Java System Application Server 包含一个安全管理器，它可以阻止未在安全策略中定义所需权限的应用程序进行访问。此策略是一个名为 `server.policy` 的文件；在进行部署后，它位于 `$CDS_HOME/deployment/deployment-name/sun/domains/server-domain/config` 目录中。

如果使用《Sun Java™ System Content Delivery Server 5.1 定制指南》中介绍的 API 编写您自己的适配器，单单使用安全策略中的一组默认权限，并不足以保证适配器能够成功运行。请编辑 `server.policy` 文件以添加适配器所需的任何定制权限，然后重新启动 Content Delivery Server。有关安全策略的详细信息，请参见 Sun Java System Application Server 文档。

第 5 章

操作 Content Delivery Server

Sun Java System Content Delivery Server 可以为运营商提供持续且最省心的服务。部署和配置 Content Delivery Server 后，请按本章所述使用 Content Delivery Server execute (cdse) 命令来启动和停止这些组件。

本节包括以下主题：

- [启动 Content Delivery Server](#)
- [停止 Content Delivery Server](#)

5.1 启动 Content Delivery Server

必须手动启动 Content Delivery Server 的组件。如果使用的是 Sun Java System Application Server，则必须先使用以下命令启动 Sun Java System Message Queue 代理：

```
$CDS_HOME/deployment/deployment-name/sun/applications/broker.sh start
```

cdse startserver 命令用于启动部署中的所有组件。如果不想启动部署中包含的服务，则请在启动 Content Delivery Server 前，使用应用服务器的管理控制台来禁用该服务。只有在要使用以下服务所支持的功能时，才需要启动这些服务：

- 只有在需要向订户发送关于产品促销通知时才需要启动通知服务。
- 只有在记帐系统使用由 Content Delivery Server 生成的电话数据记录 (Call Data Record, CDR) 或者要将事务存储在报告数据库中时，才需要启动后付费服务。
- 只有在将 SNMP 网络管理站连接到 Content Delivery Server 时，才需要启动监视服务。

可以使用以下命令来启动在主机上部署的 Content Delivery Server 的所有组件：

```
cdse startserver [-depl deployment-name] [-dbconf database-configuration-file]  
[-log log-file] [-spawn]
```

所有参数都是可选的。如果省略部署参数 `-depl`，则使用 `init_env.sh` 脚本中为 `DEFAULT_DEPL` 所指定的值。如果省略数据库配置文件参数 `-dbconf`，则使用 `init_env.sh` 脚本中为 `DEFAULT_DB` 所指定的值。如果省略日志参数 `-log`，则使用 `$CDS_HOME/cfg/logging.properties`。如果指定衍生参数 `-spawn`，则会衍生该进程并将其从 `shell` 中分离出来。

提示 – Catalog Manager 和 Vending Manager 管理控制台的默认用户名是 `admin`。默认密码是 `admin`。首次登录时必须更改密码。Subscriber Portal 和 Developer Portal 都没有默认帐户。

5.2 停止 Content Delivery Server

可以使用以下命令来停止部署中的所有组件：

```
cdse stopserver [-depl deployment-name] [-force]
```

所有参数都是可选的。如果省略部署参数 `-depl`，则使用 `init_env.sh` 脚本中为 `DEFAULT_DEPL` 所指定的值。如果指定强制参数 `-force`，则会强制停止该进程，无论该进程处于何种状态。

如果使用的是专用于 Content Delivery Server 的 Sun Java System Application Server，则在停止 Content Delivery Server 后再停止 Sun Java System Message Queue。可以使用以下命令来停止 Message Queue：

```
$CDS_HOME/deployment/deployment-name/sun/applications/broker.sh stop
```

第 6 章

卸载 Content Delivery Server

如果不再需要某个 Content Delivery Server 实例，请从安装该实例的主机上将其删除。可以仅删除一个部署，也可以删除整个产品。

本节包括以下主题：

- [卸载部署](#)
- [卸载整个产品](#)

6.1 卸载部署

如果仅需要删除某个部署，则无需从主机删除整个产品。如果主机上还存在其他部署或希望在删除现有部署后进行新的部署，则可以仅删除不再需要的部署。

6.1.1 卸载 Sun Java System Application Server 上的部署

要删除 Sun Java System Application Server 上的部署，请执行以下操作：

1. 转到 `$CDS_HOME/deployment/deployment-name/sun/applications` 目录，其中 *deployment-name* 是要删除的部署的名称。
2. 停止 Content Delivery Server。
有关说明，请参见第 5-2 页上的第 5.2 节“停止 Content Delivery Server”。
3. 停止 Message Queue 代理。
运行 `broker.sh stop` 命令。
4. 如果部署了 Message Queue 代理服务器，请使用以下命令删除服务器：

```
deploymq.sh server delete
```

5. 使用以下命令删除应用服务器域:

```
deploy.sh delete
```

6. 使用以下命令删除数据库结构:

```
cdsi db delete [-conf database-configuration] [-user db-user-file]
```

database-configuration 是用于创建结构的数据库配置文件的名称。如果省略 *database-configuration*, 则使用为 *init_env.sh* 脚本中的 `DEFAULT_DB` 所指定的值。*db-user-file* 是数据库用户文件的名称, 该文件包含用于访问数据库的用户名和密码。如果没有提供 *db-user-file*, 则使用 *init_env.sh* 脚本中为 `DEFAULT_SYS_USER` 变量指定的值。

7. 转到 `$CDS_HOME` 目录。

8. 使用以下命令删除包含部署的目录:

```
rm -rf $CDS_HOME/deployment/deployment-name
```

deployment-name 是要删除的部署的名称。

6.1.2 卸载 WebLogic Server 上的部署

要删除 WebLogic Server 上的部署, 请执行以下操作:

1. 停止 Content Delivery Server。

有关说明, 请参见第 5-2 页上的第 5.2 节 “停止 Content Delivery Server”。

2. 使用以下命令删除数据库结构:

```
cdsi db delete [-conf database-configuration] [-user db-user-file]
```

database-configuration 是用于创建结构的数据库配置文件的名称。如果省略 *database-configuration*, 则使用为 *init_env.sh* 脚本中的 `DEFAULT_DB` 所指定的值。*db-user-file* 是数据库用户文件的名称, 该文件包含用于访问数据库的用户名和密码。如果没有提供 *db-user-file*, 则使用为 *init_env.sh* 脚本中的 `DEFAULT_SYS_USER` 变量指定的值。

3. 转到 `$CDS_HOME` 目录。

4. 使用以下命令删除包含部署的目录:

```
rm -rf $CDS_HOME/deployment/deployment-name
```

deployment-name 是要删除的部署的名称。

6.2 卸载整个产品

如果不再需要与主机上的 Content Delivery Server 相关联的所有文件，请执行以下操作来删除整个产品：

1. 卸载该主机上的所有部署。

请参见第 6-1 页上的第 6.1 节“卸载部署”。

2. 使用以下命令删除主目录：

```
rm -rf cds-home
```

cds-home 是包含产品文件的目录的名称。

3. 删除 `$CDS_HOME` 环境变量。

4. 从 `PATH` 环境变量删除 `$CDS_HOME/bin` 目录。

5. 删除 `/tmp` 目录中的内容以删除由 Content Delivery Server 创建的所有临时文件。

附录 A

部署配置文件

本附录介绍了用于部署 Content Delivery Server 组件的部署配置文件的格式。此文件格式被设计成可通过 ANT replace 命令进行操作。有关使用配置文件的信
息，请参见第 3-7 页上的第 3.1.3 节“设置部署配置文件”。

A.1 配置属性

下表说明了与 Sun Java System Application Server 和 WebLogic Server 一起使用的配置文件中的常规属性。必须对这些属性进行设置。表 A-2 中说明了仅由 WebLogic Server 使用的属性。

表 A-1 常规属性

属性	描述
server.type	指定所使用的应用服务器的类型。对于 Sun Java System Application Server, 请使用 sun; 而对于 WebLogic Server, 请使用 weblogic, 例如: server.type=sun 注 - 不要更改此属性的值。
java.mem	为应用服务器启动 Java 虚拟机时所使用的字符串, 它将决定 Java 虚拟机的内存设置, 例如: java.mem=-ms512m -mx1024m -XX:MaxPermSize=256m
app.server.domain	应用服务器的域名, 例如: app.server.domain=cdsdomain
app.server.name	应用服务器使用的名称, 例如: app.server.name=cdsserver
app.server.portnumber	应用服务器在其上监听 HTTP 请求的端口, 例如: app.serverportnumber=8080
app.server.ssl.port	应用服务器在其上监听 SSL HTTP 请求的端口号, 例如: app.server.ssl.port=8081

表 A-1 常规属性 (续)

属性	描述
<code>jndi.factory</code>	要使用的 Java Naming and Directory Interface™(JNDI) API 工厂类的名称。它必须与在所使用的应用服务器的样例配置文件中使用的值相同。请指定全限定类名。 <ul style="list-style-type: none"> 对于 Sun Java System Application Server, 请指定 <code>com.sun.jndi.fscontext.RefFSContextFactory</code> 对于 WebLogic, 请指定 <code>weblogic.jndi.WLInitialContextFactory</code>
<code>jndi.url</code>	JNDI API 的 URL。 <ul style="list-style-type: none"> 对于 Sun Java System Application Server, 请设置为在样例配置文件中使用的值。 对于 WebLogic Server, 请指定服务器的主机名。该值必须以字符串 <code>t3://</code> 开头, 例如: <code>t3://host1.central.com:8080</code> <p>注 – 不要使用 <code>localhost</code>。</p>
<code>cs.internal.uri*</code>	Content Delivery Server 中使用的 Catalog Manager 的 URI, 例如: <code>cs.internal.uri=http://host1.central.com:8080/admin/</code>
<code>dp.internal.uri*</code>	Content Delivery Server 中使用的 Developer Portal 的 URI, 例如: <code>dp.internal.uri=http://host1.central.com:8080/developer/</code>
<code>fs.internal.uri*</code>	Content Delivery Server 中使用的 Fulfillment Manager 的 URI, 例如: <code>fs.internal.uri=http://host1.central.com:8080/fs/</code>
<code>vs.internal.uri*</code>	Content Delivery Server 中使用的 Vending Manager 的 URI, 例如: <code>vs.internal.uri=http://host1.central.com:8080/vsadmin/</code>
<code>sp.internal.uri*</code>	Content Delivery Server 中使用的 Subscriber Portal 的 URI, 例如: <code>sp.internal.uri=http://host1.central.com:8080/subscriber/</code>
<code>es.internal.uri*</code>	Content Delivery Server 中使用的事件服务的 URI, 例如: <code>es.internal.uri=http://host1.central.com:8080/es/</code>
<code>ms.internal.uri*</code>	Content Delivery Server 中使用的消息传送服务的 URI, 例如: <code>ms.internal.uri=http://host1.central.com:8080/ms/</code>
<code>ns.internal.uri*</code>	Content Delivery Server 中使用的通知服务的 URI, 例如: <code>ns.internal.uri=http://host1.central.com:8080/ns/</code>
<code>pps.internal.uri*</code>	Content Delivery Server 中使用的后付费服务的 URI, 例如: <code>pps.internal.uri=http://host1.central.com:8080/pps/</code>
<code>mos.internal.uri*</code>	Content Delivery Server 中使用的监视服务的 URI, 例如: <code>mos.internal.uri=http://host1.central.com:8080/mos/</code>
<code>css.internal.uri*</code>	Content Delivery Server 中使用的目录搜索服务的 URI, 例如: <code>css.internal.uri=http://host1.central.com:8080/css/</code>
<code>vss.internal.uri*</code>	Content Delivery Server 中使用的售卖搜索服务的 URI, 例如: <code>vss.internal.uri=http://host1.central.com:8080/vss/</code>

表 A-1 常规属性 (续)

属性	描述
<code>cs.external.uri*</code>	外部实体访问 Catalog Manager 时所使用的 URI, 例如: <code>cs.external.uri=http://webhost.domain.com:8080/admin/</code>
<code>dp.external.uri*</code>	外部实体访问 Developer Portal 时所使用的 URI, 例如: <code>dp.external.uri=http://webhost.domain.com:8080/developer/</code>
<code>fs.external.uri*</code>	外部实体访问 Fulfillment Manager 时所使用的 URI, 例如: <code>fs.external.uri=http://webhost.domain.com:8080/fs/</code>
<code>vs.external.uri*</code>	外部实体访问 Vending Manager 时所使用的 URI, 例如: <code>vs.external.uri=http://webhost.domain.com:8080/vsadmin/</code>
<code>sp.external.uri*</code>	外部实体访问 Subscriber Portal 时所使用的 URI, 例如: <code>sp.external.uri=http://webhost.domain.com:8080/subscriber/</code>
<code>vs.account.name</code>	当前部署的 Vending Manager 的名称, 例如, <code>suncds.vending.server.1</code> 。此名称必须与为数据库配置文件中的相应 Vending 元素指定的名称属性相匹配。在 Catalog Manager 中创建 Vending Manager Server 帐户时, 请将此名称用作帐户 ID。 注 - 如果部署了 Vending Manager 或 Catalog Manager, 则需要此属性。
<code>vs.account.password</code>	用于访问当前部署的 Vending Manager 的密码。此密码必须与为数据库配置文件中的相应 Vending 元素指定的密码相匹配。在 Catalog Manager 中创建 Vending Manager Server 帐户时, 请将此密码用作帐户密码。 注 - 如果部署了 Vending Manager 或 Catalog Manager, 则需要此属性。
<code>ms.mail.smtp.host</code>	能够接受未经身份验证的电子邮件的邮件服务器名称。 注 - 只有在部署消息传送服务时, 才需要此属性。
<code>ms.mail.smtp.sender</code>	电子邮件的回复地址。 注 - 只有在部署消息传送服务时, 才需要此属性。

* 指定的值必须是主机名和端口号, 或者是 IP 地址和端口号。不要使用 `localhost`。无论使用哪种格式, 都必须将其用于对该组件的每个引用。例如, 如果指定主机名和端口号, 则无法使用 IP 地址来引用组件。在使用 SSL 时, 必须将 URI 属性设置为部署该组件的主机的数字证书所使用的主机名。

下表说明了仅与 WebLogic Server 一起使用的属性。如果未使用 WebLogic Server，则不需要设置这些属性。

表 A-2 与 WebLogic Server 一起使用的常规属性

属性	描述
<code>app.server.threadcount</code>	应用服务器处理 HTTP 请求时使用的线程数，例如， <code>app.server.threadcount=10</code>
<code>app.server.ssl.enable</code>	SSL 支持的状态。将此属性设置为 <code>false</code> 可以禁用应用服务器的 SSL，例如： <code>app.server.ssl.enabled=false</code> 默认值为 <code>true</code> 。
<code>app.server.ssl.port</code>	SSL 所监听的端口，例如： <code>app.server.ssl.port=81</code>
<code>app.server.hostip</code>	部署服务器的主机的主机名或 IP 地址，例如： <code>app.server.hostip=host1.central.com</code> 注 – 不要使用 <code>localhost</code> 。
<code>app.server.admin.port</code>	WebLogic Server 管理控制台的端口号，例如： <code>app.server.admin.port=7800</code>

除了常规属性外，配置文件还包含以下格式的属性，这些属性用于标识部署中包含的组件：

`prefix.include`

下表列出了每个组件的前缀。

表 A-3 组件前缀

前缀	组件
<code>cs</code>	Catalog Manager
<code>dp</code>	Developer Portal
<code>vs</code>	Vending Manager
<code>sp</code>	Subscriber Portal
<code>fs</code>	Fulfillment Manager
<code>es</code>	事件服务
<code>ms</code>	消息传送服务
<code>ns</code>	通知服务
<code>pps</code>	后付费服务

表 A-3 组件前缀

前缀	组件
mos	监视服务
css	目录搜索服务
vss	售卖搜索服务

如果属性前面没有井号 (#)，则会在部署中包括某个组件。例如，如果部署配置文件中包含下列语句，则部署中包括 Catalog Manager 和 Developer Portal，而不包括 Vending Manager 和 Subscriber Portal。

```
cs.include
dp.include
#vs.include
#sp.include
```

A.2 与 Sun Java System Application Server 一起使用的样例文件

下列代码示例是一个与 Sun Java System Application Server 一起使用的样例部署配置文件。此文件显示了本附录中所描述的配置设置。此文件位于 \$CDS_HOME/cfg 目录中，名为 sn_sample.cfg。

请在使用该文件前编辑这些属性，或者使用设置创建一个新文件。样例设置通常不适用于您的环境。非超级用户无法访问小于 1024 的端口号，因此，如果没有以超级用户身份运行 Content Delivery Server，则确保将端口号更改为大于 1024 的数字。

注意，以井号 (#) 开始的行为注释，Content Delivery Server 将忽略这些行。

代码示例 A-1 与 Sun Java System Application Server 一起使用的样例部署配置文件

```
#####
#
# Configuration File
#
# This Java properties file defines all of the required properties
# that are used to deploy and configure a server.Your deployment
# can consist of any combination of the following components:
#
# CATALOG MANAGER           (cs)
# DEVELOPER PORTAL          (dp)
# VENDING MANAGER           (vs)
# SUBSCRIBER PORTAL         (sp)
```

代码示例 A-1 与 Sun Java System Application Server 一起使用的样例部署配置文件 (续)

```
# FULFILLMENT MANAGER      (fs)
# EVENT SERVICE            (es)
# MESSAGING SERVICE        (ms)
# NOTIFY SERVICE           (ns)
# POSTPAID SERVICE         (pps)
# MONITORING SERVICE       (mos)
# CATALOG SEARCH SERVICE   (css)
# VENDING SEARCH SERVICE   (vss)
#
# NOTE:Be careful not to leave trailing spaces.....
#
#####

#=====
# General Properties
#=====

# Specifies your application server to be Sun Java System (DO NOT CHANGE)
server.type=sun

# ----- Java Settings ----- #
java.mem=-Xms512m -Xmx1024m -XX:MaxPermSize=256m

# ----- App Server Settings ----- #
app.server.domain=cdsdomain
app.server.name=cdsserver
app.server.portnumber=80
app.server.ssl.port=81

# ----- JNDI Server Settings ----- #
# Specifies the JNDI factory (DO NOT CHANGE)
jndi.factory=com.sun.jndi.fscontext.RefFSContextFactory
# Specifies the JNDI URL (DO NOT CHANGE)
jndi.url=file://[[deployment]]/sun

# ----- URI Settings ----- #
cs.internal.uri=http://webhost.domain.com:80/admin/
dp.internal.uri=http://webhost.domain.com:80/developer/
fs.internal.uri=http://webhost.domain.com:80/fs/
vs.internal.uri=http://webhost.domain.com:80/vsadmin/
sp.internal.uri=http://webhost.domain.com:80/subscriber/
es.internal.uri=http://webhost.domain.com:80/es/
ms.internal.uri=http://webhost.domain.com:80/ms/
ns.internal.uri=http://webhost.domain.com:80/ns/
pps.internal.uri=http://webhost.domain.com:80/pps/
mos.internal.uri=http://webhost.domain.com:80/mos/
css.internal.uri=http://webhost.domain.com:80/css/
```


代码示例 A-1 与 Sun Java System Application Server 一起使用的样例部署配置文件（续）

```
vss.internal.uri=http://webhost.domain.com:80/vss/

cs.external.uri=http://webhost.domain.com:80/admin/
dp.external.uri=http://webhost.domain.com:80/developer/
fs.external.uri=http://webhost.domain.com:80/fs/
vs.external.uri=http://webhost.domain.com:80/vsadmin/
sp.external.uri=http://webhost.domain.com:80/subscriber/

# ----- Vending Settings ----- #
vs.account.name=cds.vending.server.1
vs.account.password=vending

# ----- Mail Settings ----- #
ms.mail.smtp.host=mailhost.domain.com
ms.mail.smtp.sender=info@domain.com

#=====
# Included Managers, Portals, & Services
#=====
# Make any of the following lines a comment if the manager, portal, or service
# should not be deployed
cs.include
dp.include
vs.include
sp.include
fs.include
es.include
ms.include
ns.include
pps.include
mos.include
css.include
vss.include
```

A.3 与 WebLogic Server 一起使用的样例文件

下列代码示例是一个与 WebLogic Server 一起使用的样例配置文件。此文件显示了本附录中所描述的配置设置。此文件位于 \$CDS_HOME/cfg 目录中，名为 wl_sample.cfg。

请在使用该文件前编辑这些属性，或者使用设置创建一个新文件。样例设置通常不适用于您的环境。非超级用户无法访问小于 1024 的端口号，因此，如果没有以超级用户身份运行 Content Delivery Server，则确保将端口号更改为大于 1024 的数字。

注意，以井号 (#) 开头的行为注释，Content Delivery Server 将忽略这些行。

代码示例 A-2 与 WebLogic Server 一起使用的样例部署配置文件

```
#####  
#  
# Configuration File  
#  
# This Java properties file defines all of the required properties  
# that are used to deploy and configure a server.Your deployment  
# can consist of any combination of the following components:  
#  
# CATALOG MANAGER          (cs)  
# DEVELOPER PORTAL         (dp)  
# VENDING MANAGER          (vs)  
# SUBSCRIBER PORTAL        (sp)  
# FULFILLMENT MANAGER      (fs)  
# EVENT SERVICE            (es)  
# MESSAGING SERVICE        (ms)  
# NOTIFY SERVICE           (ns)  
# POSTPAID SERVICE         (pps)  
# MONITORING SERVICE       (mos)  
# CATALOG SEARCH SERVICE   (css)  
# VENDING SEARCH SERVICE   (vss)  
#  
# NOTE:Be careful not to leave trailing spaces.....  
#  
#####  
  
#=====
```

General Properties

```
#=====
```

Specifies your application server to be BEA WebLogic (DO NOT CHANGE)
server.type=weblogic

----- Java Settings ----- #
java.mem=-Xms512m -Xmx1024m -XX:MaxPermSize=256m

----- App Server Settings ----- #
app.server.domain=cdsdomain
app.server.name=cdsserver
app.server.portnumber=80
 # Number of listening threads
app.server.threadcount=10
 # true for SSL support
app.server.ssl.enabled=true
 # SSL listen port
app.server.ssl.port=81

代码示例 A-2 与 WebLogic Server 一起使用的样例部署配置文件 (续)

```
# Host Name or IP address of the system where server is deployed
# You should change it if you are running with multiple IP addresses
app.server.hostip=webhost.domain.com
# WL administration console port, ONLY IF SSL IS ENABLED
app.server.admin.port=7800

# ----- JNDI Server Settings ----- #
# Specifies the JNDI factory (DO NOT CHANGE)
jndi.factory=weblogic.jndi.WLInitialContextFactory
# Specifies the JNDI URL (MUST BEGIN WITH t3://)
jndi.url=t3://webhost.domain.com:80

# ----- URI Settings ----- #
cs.internal.uri=http://webhost.domain.com:80/admin/
dp.internal.uri=http://webhost.domain.com:80/developer/
fs.internal.uri=http://webhost.domain.com:80/fs/
vs.internal.uri=http://webhost.domain.com:80/vsadmin/
sp.internal.uri=http://webhost.domain.com:80/subscriber/
es.internal.uri=http://webhost.domain.com:80/es/
ms.internal.uri=http://webhost.domain.com:80/ms/
ns.internal.uri=http://webhost.domain.com:80/ns/
pps.internal.uri=http://webhost.domain.com:80/pps/
mos.internal.uri=http://webhost.domain.com:80/mos/
css.internal.uri=http://webhost.domain.com:80/css/
vss.internal.uri=http://webhost.domain.com:80/vss/

cs.external.uri=http://webhost.domain.com:80/admin/
dp.external.uri=http://webhost.domain.com:80/developer/
fs.external.uri=http://webhost.domain.com:80/fs/
vs.external.uri=http://webhost.domain.com:80/vsadmin/
sp.external.uri=http://webhost.domain.com:80/subscriber/

# ----- Vending Settings ----- #
vs.account.name=cds.vending.server.1
vs.account.password=vending

# ----- Mail Settings ----- #
ms.mail.smtp.host=mailhost.domain.com
ms.mail.smtp.sender=info@domain.com

#=====
# Included Managers, Portals, & Services
#=====
# Make any of the following lines a comment if the manager, portal, or service
# should not be deployed
cs.include
dp.include
```

代码示例 A-2 与 WebLogic Server 一起使用的样例部署配置文件（续）

```
vs.include  
sp.include  
fs.include  
es.include  
ms.include  
ns.include  
pps.include  
mos.include  
css.include  
vss.include
```

访问控制定义

`$CDS_HOME/deployment/deployment-name/conf` 目录中的 `access_control.config` 文件用于确定何时允许访问 Content Delivery Server 组件。本节使用 BNF 表示法来描述访问控制文件中使用的语句。可以在 <http://rfc.net/rfc2234.html> 中找到此表示法的描述。

该访问控制文件包含访问控制列表、URI 组以及规则。以下代码样例显示了每种语句类型的示例。

```
acl sun {
    host '*.sun.com';
    ip6 ::1;
}

uri_group vending_service {
    uri '/vsadmin/main/sync_to_catalog.*';
    uri '/subscriber/mo_push.*';
}

access uri '/*' protocol any method any {
    deny any;
}

final access group vending_service protocol any method any {
    allow sun;
}
```

有关设置访问控制文件的信息，请参见第 4-2 页上的第 4.2 节“控制外部系统的访问”。

要创建访问控制语句，请使用以下规则：

代码示例 B-1 访问控制语句的 BNF 规则

```
acl_def = "acl" acl_name "{ 1*source }"

source = host_name [ nummask ] ";"
host_name = ip4addr | ip6addr | dnsname | "any"

nummask = "/" decnumber

ip4addr = "ip4" hostnum "." hostnum "." hostnum "." hostnum
ip6addr = "ip6" ipv6_address

hostnum = digit [ digit [ digit ] ]

decnumber = digit [ decnumber ]

digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

dnsname = "host" "'" hostname_string "'"

rule = [ "final" ] "access" uri "protocol" 1#protocols
      "method" 1#methods "{ 1*control }"

uri = "uri" "'" uri_regexp "'" | "group" uri_group_name

uri_group = "uri_group" uri_group_name "{ 1*group_def }"
group_def = "uri" "'" uri_regexp "'" ";"

control = ( "allow" | "deny" ) ( acl_name | "any" ) [ "final" ] ";"

protocols = "http" | "https" | "any"

methods = "get" | "post" | "put" | "any"

config = *acl *uri_group *rule
```

以下标记不是使用 BNF 表示法定义的：

- `hostname_string` - 允许用户使用星号 (*) 进行二义性比较的域名系统 (Domain Name System, DNS) 主机名，例如，`*.sun.com` 或 `cds*.sun.com`。

- `ipv6_address` - 符合 RFC 2373 的 IPV6 地址, <http://rfc.net/rfc2373.html> 中介绍了该规范。

支持压缩和完整的形式, 例如:

```
# 12AB:0000:0000:CD30:0000:0000:0000:0000 OR
# 12AB::CD30:0:0:0:0
# 12AB:0:0:CD30::
```

- `uri_regexp` - 用于匹配请求 URI 的正则表达式

传递给过滤器的 URI 始终是完整的 URI, 如 `/admin/main` 或 `/developer/submit_form`。不需要输入任何名称或查询参数。以下语句是以正则表达式表示的 URI 的几个示例:

- `/*`
- `/admin/main/$`
- `acl_name` 和 `uri_group_name` - 字符串文字

名称可以包含字母字符、数字或下划线 (`_`), 并且不能以数字开头。配置定义中使用的字符串文字不能用作名称。

附录 C

X11 库

X11 是 X 服务器的一种实现，它为其他系统提供显示资源。X11 是随 Solaris 操作系统 (Operating System, OS) 分发的。请将 X11 的安装目录添加到 `LD_LIBRARY_PATH` 变量中。如果使用的是 Solaris 9 操作系统，则需要 X11R6.6.1。如果使用的是 Solaris 10 操作系统，则需要 X11R6.6.2。下表指明了要添加到系统中的软件包。

表 C-1 所需的 X11 库

软件包	描述	说明
SUNWxwacx	AccessX 客户端程序	
SUNWxwcfnt	X 窗口系统通用（非必需）字体	
SUNWxwcs1	X 窗口系统显示 Postscript CID 支持库	
SUNWxwdem	X 窗口系统演示程序	
SUNWxwdim	X 窗口系统演示图像	
SUNWxwdv	X 窗口系统窗口驱动程序	
SUNWxwdvx	X 窗口系统窗口驱动程序（64 位）	仅限 Solaris 9 操作系统
SUNWxwdx	DPS motif 库	
SUNWxwfa	X 窗口系统字体管理器	
SUNWxwfont	X 窗口系统平台所需的字体	
SUNWxwfs	X 窗口系统字体服务器	
SUNWxwhl	<code>/usr/include</code> 中的 X 窗口系统和图形标题链接	
SUNWxwice	X 窗口系统客户端间交换 (Inter-Client Exchange, ICE) 组件	
SUNWxwicx	X 窗口系统 ICE 库（64 位）	仅限 Solaris 9 操作系统
SUNWxwinc	X 窗口系统包含文件	

表 C-1 所需的 X11 库 (续)

软件包	描述	说明
SUNWxwman	X 窗口系统联机用户手册页	
SUNWxwmod	X 窗口系统内核模块	
SUNWxwmox	X 窗口系统内核模块 (64 位)	仅限 Solaris 9 操作系统
SUNWxwoft	X 窗口系统可选字体	
SUNWxwopt	X 窗口系统可选客户端	
SUNWxwpft	X 窗口系统打印机字体	
SUNWxwpl	特定于平台的 X 服务器辅助链接	
SUNWxwplt	X 窗口系统平台软件	
SUNWxwplx	X 窗口系统库软件 (64 位)	仅限 Solaris 9 操作系统
SUNWxwpmn	X 窗口系统联机程序员手册页	
SUNWxwplr	X 窗口系统平台软件配置	仅限 Solaris 10 操作系统
SUNWxwpsr	特定于 Sun4u 平台的 X 服务器辅助过滤器模块	
SUNWxwrtl	/usr/lib 中的 X 窗口系统和图形运行时库链接	
SUNWxwrtx	X 窗口系统运行时兼容性软件包 (64 位)	仅限 Solaris 9 操作系统
SUNWxwslb	X 窗口系统 static/lint 库	
SUNWxwslx	X 窗口系统 lint 库 (64 位)	仅限 Solaris 9 操作系统
SUNWxwsrc	X 窗口系统样例源	
SUNWxwsvr	X 窗口系统虚拟服务器	
SUNWxwsvr	XScreenSaver	仅限 Solaris 10 操作系统
SUNWxwxft	Xft (X FreeType) 库	仅限 Solaris 10 操作系统
SUNWxwxst	X 窗口系统 XST 扩展	仅限 Solaris 10 操作系统

索引

A

access_control.config 文件, 4-2, B-1
admin.sh 命令, 3-11
AliasForwarding 元素, 4-4
aliasforwarding.xml 文件, 4-4
AliasMap 元素, 4-4
安装目录, 删除, 6-3

B

Bash Shell, 2-2
BEA_HOME 环境变量, 3-2
broker.sh 命令, 3-10, 5-1, 5-2, 6-1
别名转发, 4-4
部署
 Catalog Manager, 3-16
 Catalog Manager 部署, 3-18
 高级, 3-22
 高级部署, 3-24
 基本, 3-1
 基本部署, 3-8
 Vending Manager, 3-18
 Vending Manager 部署, 3-21
 卸载
 Sun Java System Application Server, 6-1
 WebLogic Server, 6-2
 一个主机上的多个, 3-12
部署配置
 Catalog Manager, 1-3
 高级, 1-5
 基本, 1-3

Vending Manager, 1-4

部署配置文件
 默认, 3-8, 3-18, 3-21, 3-24
 设置
 Catalog Manager 部署, 3-17
 高级部署, 3-23
 基本部署, 3-7
 Vending Manager 部署, 3-20
属性, A-1
样例
 对于 Sun Java System Application Server, A-5
 对于 WebLogic Server, A-7
 sn_sample.cfg 文件, A-5
 wl_sample.cfg 文件, A-7

C

Catalog Manager
 管理, 4-2
 启动, 5-1
 停止, 5-2
 URI, A-2
Catalog Manager 部署
 部署, 3-18
 设置配置文件, 3-17
 设置数据库, 3-16
CDS_HOME 环境变量
 删除, 6-3
 设置, 2-5
cdse, 请参见 Content Delivery Server execute 命令

cdsi, 请参见 Content Delivery Server install 命令
CFG 文件, 请参见部署配置文件

cHTML, 2-3

Content Delivery Server

概述, 1-1

解压缩 Content Delivery Server 文件, 2-5

启动, 5-1

设置环境变量, 2-5

停止, 5-2

卸载, 6-3

Content Delivery Server execute 命令

启动服务器, 5-1

停止服务器, 5-2

Content Delivery Server install 命令

部署, 3-8

初始化数据库, 3-7, 3-16, 3-19, 3-23

创建数据库结构, 3-7, 3-17, 3-20

删除数据库, 6-2

填充数据库, 3-7, 3-17, 3-20

Cookie 名称, 3-12

D

DBConf.xml 文件, 3-3

DBSysUser.xml 文件, 3-5

DEFAULT_CONFIG, 3-24

DEFAULT_CONFIG 环境变量, 3-8, 3-18, 3-21

DEFAULT_DB 环境变量, 3-6, 3-23

DEFAULT_DEPL 环境变量, 3-8

DEFAULT_SYS_USER 环境变量, 3-6, 3-23

DefaultWebApp.properties 文件, 4-4

deploy.sh 命令, 3-10, 6-2

deploymq.sh 命令, 3-9, 6-1

Developer Portal, URI, A-2

DRM Fusion Toolkit, 2-4

端口号

Message Queue 代理, 3-9

SSL, A-1

WebLogic Server 管理控制台, A-4

应用服务器, A-1

短消息服务中心 (Short Messaging Service Center, SMSC), 2-3

多媒体消息服务中心 (Multimedia Message Service Center, MMSC), 2-4

F

filter.AliasForwarding.file 属性, 4-4

final 关键字, 4-4

Firefox, 2-3

Fulfillment Manager, URI, A-2

防火墙, 配置, 4-1

访问控制

定义, B-1

规则, 4-3

进程, 4-3

Sun Java System Application Server, 4-6

文件, 4-2

访问控制列表, 4-3

服务器布局, 1-1

服务器名称, A-1

G

高级部署

部署, 3-24

设置配置文件, 3-23

设置数据库, 3-22

工厂, JNDI, A-2

规则, 访问控制, 4-3

H

后付费服务

启动, 5-1

属性, A-4

停止, 5-2

URI, A-2

环境变量

Content Delivery Server, 2-5

init_env 文件, 3-2

I

iAppli 应用程序, 3-6

init_env.sh shell 脚本, 3-2

Internet Explorer, 2-3

J

Java 编程语言主目录, 3-2

Java 开发工具包, 2-3

Java 图像 I/O 框架, 2-4

JAVA_HOME 属性, 3-2

JNDI 工厂, A-2

基本部署

部署, 3-8

设置配置文件, 3-7

设置数据库, 3-2

规划注意事项, 1-1

监视服务

启动, 5-1

属性, A-5

停止, 5-2

URI, A-2

结构创建, 3-7, 3-17, 3-20

解压缩, 2-3, 2-5

K

可缩放性, 1-2

L

LDAP, 2-3

浏览器, 2-3

M

Message Queue 代理

启动, 5-1

删除, 6-1

设置, 3-9

停止, 5-2

ms.mail.smtp.host 属性, A-3

ms.mail.smtp.sender 属性, A-3

目录搜索服务, URI, A-2

N

内存

设置, A-1

要求, 2-2

P

password

Message Queue 代理, 3-9

WebLogic Server 管理, 3-12

配置属性, A-1

配置文件

部署

默认, 3-8, 3-18, 3-21, 3-24

Sun Java System Application Server 的样例, A-5

设置 Catalog Manager 部署, 3-17

设置 Vending Manager 部署, 3-20

设置高级部署, 3-23

设置基本部署, 3-7

属性, A-1

WebLogic Server 的样例, A-7

数据库

默认, 3-6, 3-23

设置基本部署, 3-3

为 Catalog Manager 部署设置, 3-16

为 Vending Manager 部署设置, 3-19

为高级部署设置, 3-22

样例, 3-3

样例位置, 3-7

Q

群集, 1-2

R

容量规划, 1-2

软件要求, 2-2

S

SafeNet DRM Fusion Toolkit, 2-4

server.policy 文件, 4-6

SMPP API 库, 2-3

sn_sample.cfg 文件, A-5

SSL, 3-11

Subscriber Portal, URI, A-2

Sun Java System Application Server

控制访问, 4-6

设置, 3-9

样例配置文件, A-5

SUN_AS_HOME 环境变量, 3-2

设备库, 2-3

事件服务

- 启动, 5-1
- 属性, A-4
- 停止, 5-2
- URI, A-2

售卖搜索服务, URI, A-2

数据库

- 初始化, 3-7, 3-16, 3-19, 3-23
- 结构创建, 3-7, 3-17, 3-20
- 删除, 6-2
- 设置
 - 高级部署, 3-22
 - 基本部署, 3-2
 - 为 Catalog Manager 部署, 3-16
 - 为 Vending Manager 部署, 3-19
- 填充, 3-7, 3-17, 3-20
- 要求, 2-4

数据库配置文件

- 默认, 3-6, 3-23
- 设置
 - 高级部署, 3-22
 - 基本部署, 3-3
 - 为 Catalog Manager 部署, 3-16
 - 为 Vending Manager 部署, 3-19
- 样例, 3-3

数据库用户文件, 3-5, 3-16, 3-19, 3-22

属性

- 后付费服务, A-4
- 监视服务, A-5
- 事件服务, A-4
- 通知服务, A-4
- 消息传送服务, A-4

T

TargetContext 元素, 4-5

TargetPath 元素, 4-5

通知服务

- 启动, 5-1
- 属性, A-4
- 停止, 5-2
- URI, A-2

U

URI 组, 4-3

URLPattern 元素, 4-5

V

Vending Manager

- 启动, 5-1
- 停止, 5-2
- URI, A-2
- 准备使用, 3-15

Vending Manager 部署

- 部署, 3-21
- 设置配置文件, 3-20
- 设置数据库, 3-19

Vending Manager Server 帐户, 4-2

vs.account.name 属性, 3-5, 3-21, A-3

vs.account.password 属性, 3-21, A-3

W

WAP 网关, 2-3

WebLogic Server

- 更改 Cookie 名称, 3-12
- 设置, 3-11
- 样例部署配置文件, A-7
- 证书, 3-11

weblogic.xml 文件, 3-13

WL_HOME 属性, 3-2

wl_sample.cfg 文件, A-7

WML, 2-3

X

X 服务器, 2-4, C-1

X11 库, 2-4, C-1

XHTML, 2-3

线程数, A-4

先决条件, 2-1

消息传送服务

- 启动, 5-1
- 属性, A-4
- 停止, 5-2
- URI, A-2

卸载

部署

Sun Java System Application Server, 6-1

WebLogic Server, 6-2

Content Delivery Server, 6-3

Y

样例

部署配置文件

对于 WebLogic Server, A-7

Sun Java System Application Server, A-5

数据库配置文件, 3-3

要求

Bash Shell, 2-2

Java 开发工具包, 2-3

解压缩, 2-3

浏览器, 2-3

软件, 2-2

数据库, 2-4

硬件, 2-2

应用服务器, 2-2

硬件要求, 2-2

应用服务器域

删除, 6-2

设置, 3-10

应用服务器

端口号, A-1

服务器名称, A-1

设置

Sun Java System Application Server, 3-9

WebLogic Server, 3-11

要求, 2-2

域名, A-1

用户名

Message Queue 代理, 3-9

WebLogic Server 管理, 3-12

邮件服务器, A-3

域名, A-1

Z

证书

对于 WebLogic Server, 3-11

主目录, 删除, 6-3

