



安装指南

Sun Java™ System Content Delivery Server

版本 2004Q1

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, California 95054
U.S.A.
1-800-555-9SUN 或 1-650-960-1300
文件号码: 817-7361
2004 年 8 月

版权所有 © 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 保留所有权利。

对于本文档中介绍的产品，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。

此分发可能包括由第三方开发的材料。

Sun、Sun Microsystems、Sun 徽标、Java、JavaServer Pages、JDBC 和 Solaris 是 Sun Microsystems, Inc. 在美国和其他国家 / 地区的商标或注册商标。

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。

Netscape 和 Netscape Navigator 是 Netscape Communications Corporation 在美国和其他国家 / 地区的商标或注册商标。Adobe 徽标是 Adobe Systems, Incorporated 的注册商标。

本产品包含由 Apache Software Foundation (<http://www.apache.org/>) 开发的软件。版权所有 (c) 1999-2003 The Apache Software Foundation. 保留所有权利。

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。

本文档按“原样”提供，对所有明示或默示的条件、陈述和担保，包括对适销性、适用性和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。



请重复
利用



Adobe PostScript

目录

前言	xiii
适用读者	xiii
范围	xiii
组织	xiv
相关文档	xiv
惯例	xv
访问联机的 Sun 文档	xv
Sun 欢迎您提出意见	xvi
1. 概述	1
1.1	体系结构 1
1.2	组件 3
1.2.1	Catalog Manager 3
1.2.2	Vending Manager 4
1.2.3	Fulfillment Manager 4
1.2.4	数据库 4
1.2.5	Content Delivery Server 服务 5
1.3	用户角色 6
1.4	部署方案 7
1.4.1	基本配置 7
1.4.2	多售卖配置 7
1.4.3	高级配置 8

1.5	计划注意事项	8
1.5.1	服务器布局	8
1.5.2	可缩放性	9
1.5.3	容量	9
2.	设置环境	11
2.1	验证先决条件	11
2.1.1	硬件要求	12
2.1.2	软件要求	12
2.1.3	数据库要求	13
2.2	创建安装目录	14
2.3	解压缩 Content Delivery Server 文件	14
2.4	设置环境变量	15
3.	部署 Content Delivery Server	17
3.1	基本部署	17
3.1.1	第 1 步: 编辑基本部署的 init_env Shell 脚本	18
3.1.2	第 2 步: 设置基本部署的数据库	18
3.1.3	第 3 步: 设置基本部署的部署配置文件	21
3.1.4	第 4 步: 部署	22
3.1.5	第 5 步: 设置应用程序服务器	23
3.1.6	第 6 步: 准备 Content Delivery Server 的使用	26
3.2	Vending Manager 部署	28
3.2.1	第 1 步: 编辑 Vending Manager 部署的 init_env Shell 脚本	29
3.2.2	第 2 步: 设置 Vending Manager 部署的数据库	29
3.2.3	第 3 步: 设置 Vending Manager 部署的部署配置文件	30
3.2.4	第 4 步: 实施 Vending Manager 部署	32
3.2.5	第 5 步: 设置 Vending Manager 部署的应用程序服务器	32
3.2.6	第 6 步: 准备 Vending Manager 以供使用	32
3.3	高级部署	32
3.3.1	第 1 步: 编辑高级部署的 init_env Shell 脚本	33
3.3.2	第 2 步: 设置高级部署的数据库	33
3.3.3	第 3 步: 设置高级部署的部署配置文件	34

3.3.4	第 4 步：部署高级部署	35
3.3.5	第 5 步：设置高级部署的应用程序服务器	35
3.3.6	第 6 步：准备高级部署以供使用	35
4.	配置功能	37
4.1	配置 Content Delivery Server 服务	37
4.1.1	配置消息传送服务	37
4.1.2	配置推送监听器服务	40
4.1.3	配置后付费服务	41
4.1.4	配置监视服务	42
4.1.5	配置确认监听器服务	44
4.2	配置 SMS 推送消息	44
4.3	配置 MMS 支持	45
4.4	配置接口组件	47
4.4.1	设置通用属性	47
4.4.2	设置 Developer Portal 属性	48
4.4.3	设置 Subscriber Portal 属性	48
4.4.4	设置 Catalog Manager 属性	53
4.4.5	设置 Vending Manager 属性	53
4.5	为防火墙配置 Content Delivery Server	54
4.6	配置价格模型	55
4.7	配置提交检验器 workflow	56
4.7.1	Java 应用程序的 workflow	56
4.7.2	iAppli 应用程序的 workflow	60
4.7.3	外部托管受版权保护内容的工作流	61
4.7.4	DRM 消息中打包的静态内容的工作流	61
4.7.5	要在 DRM 消息中打包的静态内容的工作流	61
4.7.6	缺省 workflow	62
4.7.7	指定 workflow 标准	62
4.8	对 Subscriber Portal 进行本地化	63
4.8.1	设置缺省语言	63
4.8.2	提供已本地化的字符串	64

4.8.3	提供已本地化的模板和图像	64
4.8.4	设置货币符号	65
4.9	配置定制的市场活动	66
4.9.1	配置邮件服务	66
4.9.2	设置 Subscriber Portal URL	66
4.10	配置订户帐户管理	67
4.11	配置对外部托管的内容的支持	69
4.12	配置对定制报告的支持	70
4.13	设置对 iAppli 应用程序的支持	71
4.13.1	将 DoJa 库添加到数据库	71
4.13.2	提交 DoJa 库	71
4.14	设置发送给开发者的通知	75
4.15	设置订户通知的缺省选项	78
5.	操作 Content Delivery Server	79
5.1	启动 Content Delivery Server	79
5.1.1	启动 Content Delivery Server 管理器	80
5.1.2	启动 Content Delivery Server 服务	80
5.2	停止 Content Delivery Server	81
5.2.1	停止 Content Delivery Server 管理器	81
5.2.2	停止 Content Delivery Server 服务	82
5.2.3	停止 Sun Java System Message Queue	82
6.	卸载 Content Delivery Server	83
6.1	卸载部署	83
6.1.1	卸载 WebLogic Server 上的部署	83
6.1.2	卸载 Sun Java System Application Server 上的部署	84
6.2	卸载当前版本的 Content Delivery Server	85
6.3	卸载以前版本的 Content Delivery Server	85
A.	部署配置文件	87
A.1	常规属性	87
A.2	管理器属性	90

A.3	服务属性	91
A.4	与 BEA WebLogic Server 一起使用的样例部署配置文件	92
A.5	与 Sun Java System Application Server 一起使用的样例部署配置文件	95
B.	调节性能	101
B.1	调节配置变量	101
B.1.1	设置数据库连接池大小	101
B.1.2	设置 Java 内存设置	102
B.2	调节 Oracle 数据库	102
B.3	调节应用程序服务器	102
B.3.1	调节 Sun Java System Application Server	102
B.3.2	调节 WebLogic Server	103
C.	系统维护	107
C.1	系统监视	107
C.1.1	应用程序监视代理	107
C.1.2	日志文件	108
C.2	与外部系统的交互	109
C.2.1	记帐系统工作流	109
C.2.2	订户数据库工作流	110
C.2.3	Content Delivery Server 数据库工作流	111
C.3	数据库维护	111
C.3.1	事务表	111
C.3.2	实体表	112
C.4	错误消息	113
C.4.1	Content Delivery Server 日志	113
C.4.2	消息传送服务日志	115
C.4.3	推送监听器服务日志	116
C.4.4	确认监听器服务日志	116
C.4.5	通知服务日志	117
D.	报告	119
	索引	121



-
- 图 1 Sun Java System Content Delivery Server 2
 - 图 2 功能体系结构 3
 - 图 3 用户角色位于 Content Delivery Server 6
 - 图 4 基本配置 7
 - 图 5 多售卖配置 8

表

表 1	Content Delivery Server 组件	12
表 2	软件要求	12
表 3	在 <code>init_env</code> 中设置的环境变量	18
表 4	部署命令的可选变量	22
表 5	<code>deploymq.sh</code> 的参数	23
表 6	<code>deploy.sh</code> 的参数	24
表 7	监视服务属性	43
表 8	MMS 消息传送属性	46
表 9	<code>CommonConsole.properties</code> 文件中的属性	47
表 10	<code>DeveloperPortal.properties</code> 文件中的属性	48
表 11	<code>SubscriberPortal.properties</code> 文件中的属性	49
表 12	<code>AdminConsole.properties</code> 文件中的属性	53
表 13	<code>VSAdminConsole.properties</code> 文件中的属性	53
表 14	主机和端口属性	54
表 15	DRM 代理	57
表 16	工作流变量	58
表 17	本地化属性	64
表 18	货币属性	65
表 19	开发者通知的属性	75
表 20	<code>cdse</code> 命令的服务组件值	81
表 21	常规属性	88

表 22	与 WebLogic Server 一起使用的常规属性	89
表 23	Content Delivery Server 管理器组件前缀	90
表 24	管理器属性	90
表 25	Content Delivery Server 服务前缀	91
表 26	服务属性	91
表 27	状态通知	108
表 28	Content Delivery Server 生成的警报	108
表 29	报告数据	119

前言

本指南提供了关于安装、部署和配置 Sun Java™ System Content Delivery Server 的信息。还介绍了 Java System Content Delivery Server 的体系结构以帮助您制订精明的部署和容量计划决策。

适用读者

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

范围

本指南介绍了如何安装、配置和启动 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。还介绍了配置基本功能的步骤。《*Sun Java System Content Delivery Server 集成指南*》中介绍了与外部系统的集成以及定制集成。

组织

本指南的组织结构如下：

- [第 1 章 “概述”](#) 介绍了 Content Delivery Server 的体系结构和组件。还介绍了部署方案，并概述了一些计划注意事项。
- [第 2 章 “设置环境”](#) 介绍了部署 Content Delivery Server 的硬件和软件要求。还介绍了如何从发行套件文件提取文件以及如何为 Content Delivery Server 设置环境。
- [第 3 章 “部署 Content Delivery Server”](#) 提供了安装 Content Delivery Server 系统和部署组件的说明。
- [第 4 章 “配置功能”](#) 提供了设置要使用的 Content Delivery Server 功能的说明。
- [第 5 章 “操作 Content Delivery Server”](#) 介绍了如何启动和停止 Content Delivery Server 管理器和服务。还提供了卸载 Content Delivery Server 的说明。
- [第 6 章 “卸载 Content Delivery Server”](#) 提供了卸载单个 Content Delivery Server 部署以及卸载全部 Content Delivery Server 的说明。
- [附录 A “部署配置文件”](#) 介绍了部署和配置工具所使用的配置文件格式。还展示了两个样例配置文件：一个用于与 WebLogic Server 一同运行，另一个用于与 Sun Java System Application Server 一同运行。
- [附录 B “调节性能”](#) 提供了一些关于调节 Content Delivery Server 系统性能的建议。
- [附录 C “系统维护”](#) 提供了关于维护 Content Delivery Server 的信息。还介绍了可能记录的错误消息。
- [附录 D “报告”](#) 介绍了生成下载报告时可用的信息。

相关文档

下列文档提供了关于 Sun Java System Content Delivery Server 的信息：

- *Sun Java System Content Delivery Server 容量计划指南*

本指南适用于负责购买和管理用于 Sun Java System Content Delivery Server 的设备的计划人员和 IT 人员。它提供的指导用于确定所需的硬件和软件。

- *Sun Java System Content Delivery Server 安装指南*

本指南适用于负责部署和维护 Sun Java System Content Delivery Server 的系统管理员。它提供了有关安装和部署 Content Delivery Server 的信息。

- *Sun Java System Content Delivery Server 集成指南*

本指南适用于负责配置 Sun Java System Content Delivery Server 以使其在现有基础结构中进行工作的系统管理员。它介绍了用于将 Content Delivery Server 与现有记帐系统、用户数据、WAP 网关及推送系统集成的适配器。还介绍了创建 Subscriber Portal 的设备特定版本的框架。

- *Sun Java System Content Delivery Server 署名指南*

本指南适用于可视内容设计人员。它介绍了如何针对自己的企业来定制 Sun Java System Content Delivery Server 的 Subscriber Portal 和 Developer Portal 组件的外观。

■ *Sun Java System Content Delivery Server 定制指南*

本指南适用于负责创建将 Sun Java System Content Delivery Server 与现有基础结构集成时使用的定制适配器的程序员。

■ *Sun Java System Content Delivery Server 管理员指南*

本指南适用于负责管理 Catalog Manager 和 Vending Manager 的系统管理员。它介绍了如何管理内容以及如何定义支持的设备和设备功能。本指南还提供一些说明，用于管理开发者、Vending Manager 和订户对 Sun Java System Content Delivery Server 的访问。

惯例

本指南中使用了下列印刷惯例：

字样	含义	实例
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	输入的内容（相对于计算机屏幕输出信息）。	% su Password:
<i>AaBbCc123</i>	书名、新词或术语、要强调的词。	阅读《 <i>用户指南</i> 》中的第 6 章。 这些称为类选项。 必须是超级用户才能执行此操作。
	命令行变量；使用真实名称或值替换。	要删除文件，请键入 <code>rm filename</code> 。

访问联机的 Sun 文档

通过 Sun Product Documentation Web 站点可以在 Web 上访问 Content Delivery Server 文档：

<http://docs.sun.com>

Sun 欢迎您提出意见

我们希望能够改进自己的文档，欢迎您提出意见和建议。请将您的意见通过以下地址发送给我们：

<http://docs.sun.com/db/form/comments>

当您提供意见和建议时，可能需要在表单中提供文档英文版本的标题和文件号码。本文档英文版本的文件号码和标题是：817-4835, Installation Guide。

概述

Sun Java System Content Delivery Server 是一种高性能软件解决方案，使运营商可以管理向订户无线 (OTA) 传送几乎所有类型的无线内容。

本节包括以下主题：

- [体系结构](#)
- [组件](#)
- [用户角色](#)
- [部署方案](#)
- [计划注意事项](#)

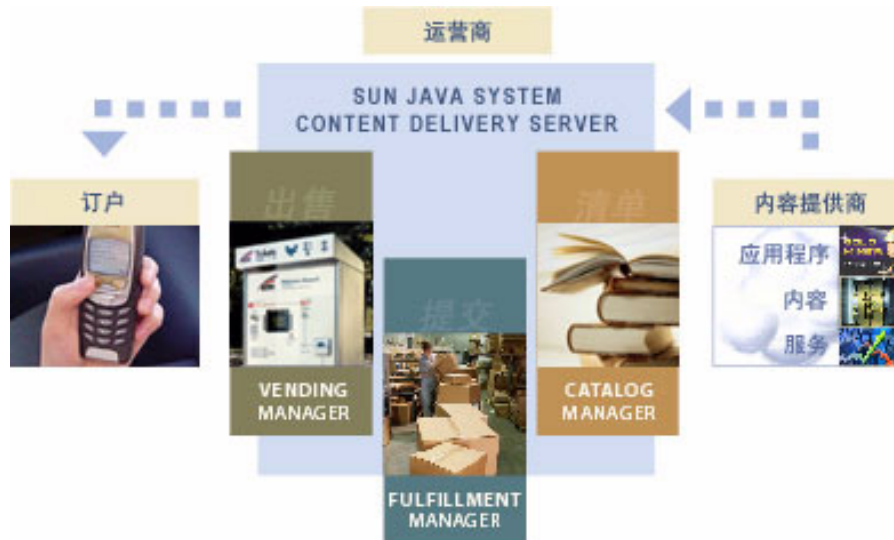
1.1 体系结构

Java System Content Delivery Server 将内容聚集、展示和传送分为如下几个独立的管理器— Catalog Manager、Fulfillment Manager 和 Vending Manager。利用这些管理器，运营商可以集中管理内容、定制多种订户界面，并可以提供快速下载。

Content Delivery Server 是一个灵活的系统，其中包含可以通过各种方式根据具体要求进行部署的多个组件。Content Delivery Server 是使用商业 Web 服务器、应用程序服务器、Oracle 数据库和 Java 技术的经过证实的 Internet 传送平台，并在多 CPU 的 UNIX 平台（如 HP-UX 和 SolarisTM）上得到了证实。

总体系统提供了内容提供者（也称为开发者）与订户之间的端对端连接。图 1 提供了贯穿 Content Delivery Server 的内容路径的高级概述。

图 1 Sun Java System Content Delivery Server



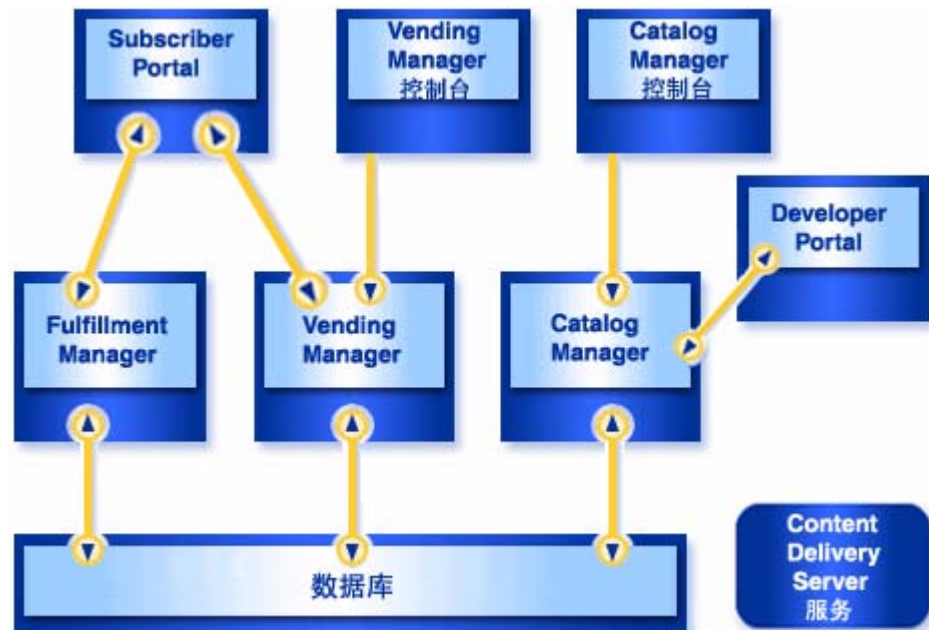
1. 内容提供者创建某些类型的内容（例如基于 Java 技术的应用程序，即“Java 应用程序”），并使用 Developer Portal 将其提交给 Catalog Manager。
2. Catalog Manager 自动验证内容并在通过验证后从开发者那里接受该内容。
3. Catalog Manager 管理员发布内容以便一个或多个 Vending Manager 使用。
4. Vending Manager 管理员储存 Catalog Manager 发布的内容以便订户使用。
5. 订户检查自己可用的内容和价格选项并使用 Subscriber Portal 下载所需的内容。

Content Delivery Server 体系结构的组织方式要求 Catalog Manager 必须核实内容提供者提交的所有内容。Vending Manager 可以通过 Vending Manager 的自动储存选项接受发布来的所有内容，并将内容存储在订户站点上。可以禁用此选项，以允许 Vending Manager 管理员手动选择要为订户储存的内容。

1.2 组件

图 2 阐述了 Content Delivery Server 的功能体系结构。

图 2 功能体系结构



Content Delivery Server 体系结构具有高度可缩放性。当订户群增长时，可以添加其他组件实例。Content Delivery Server 包括下列功能组件：

- Catalog Manager
- Vending Manager
- Fulfillment Manager
- 数据库
- Content Delivery Server 服务

1.2.1 Catalog Manager

Catalog Manager 是内容聚集器或中央仓库—它管理来自开发团体的内容。Catalog Manager 对内容进行授权之后，Catalog Manager 的管理员将该内容发布到 Vending Manager。每一个 Catalog Manager 都可以为一个或多个 Vending Manager 提供服务。Catalog Manager 包括 Developer Portal，即内容提供者的 Web 接口。

通过基于 Web 的 Catalog Manager 管理控制台来访问和管理 Catalog Manager。有关管理 Catalog Manager 的信息，请参见《Sun Java System Content Delivery Server 管理员指南》。

1.2.2 Vending Manager

Vending Manager 负责内容的出售和定价。Vending Manager 管理由 Catalog Manager 发布的内容，并使订户可以下载该内容。Vending Manager 控制订户访问权限和购买价格，还提供每日统计报告。Vending Manager 包括 Subscriber Portal，Subscriber Portal 将为订户提供从移动设备或 PC 对 Content Delivery Server 进行访问的权限。

可以在系统中安装多个 Vending Manager。可以将所有 Vending Manager 配置为从 Catalog Manager 接收相同的内容，也可以将各个 Vending Manager 配置为有选择地接收内容。通过将各个 Vending Manager 配置为有选择地接收内容，可以为不同的读者定制 Vending Manager。

对 Vending Manager 组件有下列要求：

- 每个 Vending Manager 都绑定到特定的 Catalog Manager。
- 每个 Vending Manager 必须拥有其自己的 Fulfillment Manager 实例。
- 每个 Vending Manager 对各种 Content Delivery Server 服务都拥有其自己的服务实例。
- Vending Manager 与 Catalog Manager 是否位于同一主机均可。
- 每个 Vending Manager 有特定的订户群。为了处理日益增长的需求，可以在订户群增长时添加 Vending Manager。可以将多个 Vending Manager 分配给一个 Catalog Manager。

通过基于 Web 的 Vending Manager 管理控制台来访问和管理 Vending Manager。有关管理 Vending Manager 的信息，请参见《*Sun Java System Content Delivery Server 管理员指南*》。

1.2.3 Fulfillment Manager

Fulfillment Manager 控制传送给订户的内容，并向订户传送内容。为了获得更好的性能，内容将在 Fulfillment Manager 中进行本地高速缓存。如果尚未更改数据库中的版本，则应使用本地副本满足随后的下载请求。Fulfillment Manager 还管理内容的数字权限。启动经过程序校验的应用程序时，该应用程序将与 Fulfillment Manager 联系，以验证订户是否有权继续使用该应用程序。

1.2.4 数据库

数据库用于：

- 跟踪和报告订户的使用行为
- 配置内容
- 管理设备功能
- 注册和存储内容
- 定义安全策略
- 管理用户配置文件

在数据库中，Content Delivery Server 初始部署中部署的 Catalog Manager、Fulfillment Manager 和 Vending Manager 将共享一个模式。其他每个 Vending Manager 都有自己的模式。每个事件服务都共享部署时使用的 Vending Manager 的模式。

1.2.5 Content Delivery Server 服务

Content Delivery Server 包括多个执行 Content Delivery Server 专用功能的服务。

1.2.5.1 消息传送服务

消息传送服务使应用程序可以通过短消息服务 (SMS)、无线应用协议 (WAP)、多媒体消息服务 (MMS) 或 Internet 电子邮件来发送数据，具体传送方式取决于订户的设备类型。此服务提供了将消息传送给设备或用户指定的目标（例如电子邮件地址）的环境。

1.2.5.2 事件服务

事件服务提供了用于发布和传送事件的框架。此框架使其他部分的 Content Delivery Server 可以将事件发送给系统，然后将相应的消息传送给用户。

1.2.5.3 推送监听器服务

推送监听器服务使 Content Delivery Server 可以从短消息服务中心 (SMSC) 接收移动始发 (MO) 推送消息。然后 Content Delivery Server 通过向始发设备发送请求的数据对消息作出响应。每个 Vending Manager 都需要自己的推送监听器服务实例。

1.2.5.4 通知服务

通知服务执行由 Vending Manager 管理员建立的广告活动。此服务将生成要向其发送通知的订户列表，并将消息提交给消息传送服务进行传送。

1.2.5.5 后付费服务

后付费服务处理记帐事件并向外部记帐系统提供信息。记帐系统可以使用此信息向订户收费或退费。

1.2.5.6 确认监听器服务

Content Delivery Server 支持内容一步下载，不支持 MIDlet。对于 UDP 消息，确认监听器服务将监听特定端口，以便一步下载成功后获取发自设备的下载确认消息。此服务还用于处理 MMS 确认消息。

1.2.5.7 监视服务

监视服务由应用程序监视代理用于监视 Content Delivery Server 和发布状态消息及报警。监视服务支持简单网络管理协议 (SNMP)。有关详细信息，请参见第 107 页上的“应用程序监视代理”。

1.3 用户角色

图 3 阐明了使用 Content Delivery Server 的各种用户类型。开发者通过 Developer Portal 将内容提交给 Catalog Manager。Catalog Manager 管理员将接受的内容发布到一个或多个 Vending Manager。Vending Manager 管理员储存选定的内容，并使订户可以使用该内容。订户通过 Subscriber Portal 从所提供的内容中进行选择。

图 3 用户角色位于 Content Delivery Server



一个部署方案包括一个作为 Catalog Manager 管理员的发布者，该管理员接受和拒绝内容提供者提交内容。每个 Vending Manager 都可能代表不同的运营商，其 Vending Manager 管理员负责选择订户可使用的内容。

如果 Catalog Manager 和 Vending Manager 共享一个管理员，则可以在 Vending Manager 管理控制台中启用自动储存功能。自动储存功能将自动储存所有 Catalog Manager 批准的、Vending Manager 可以使用的内容。

1.4 部署方案

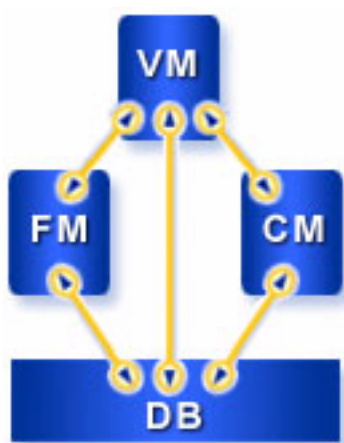
Content Delivery Server 体系结构支持各种部署方案：从在同一台服务器上部署所有组件到将各个组件分别部署在其专用的服务器上。为每个部署创建的部署配置文件用于确定如何部署组件。

注—讨论 Content Delivery Server 部署时，“服务器”是指应用程序服务器实例。“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，一个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。

1.4.1 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，并且一般是最初部署时使用的配置。下图阐明了基本配置。

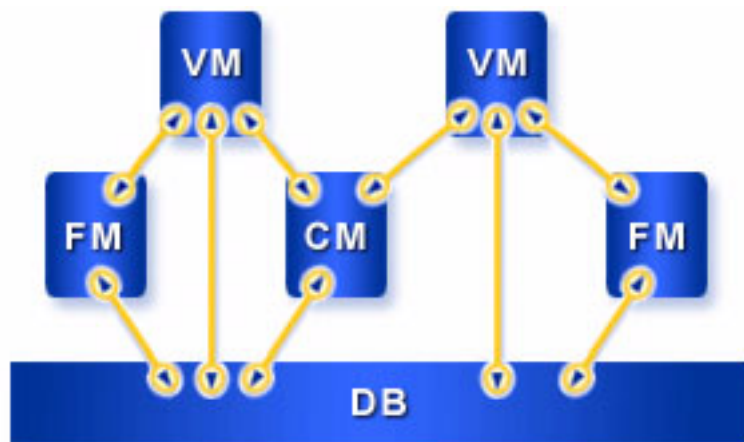
图 4 基本配置



1.4.2 多售卖配置

Content Delivery Server 的一个优点是能够使用定制的 Vending Manager 为各种用户群提供服务。在该配置中部署和配置了一个或多个附加 Vending Manager，以便从现有 Catalog Manager 接收内容。每个 Vending Manager 都可以有自己的记帐管理器、后付费处理程序和用户群。使用其他的 Vending Manager 部署 Fulfillment Manager 和每个 Content Delivery Server 服务的实例。下图阐明了多售卖配置。

图 5 多售卖配置



1.4.3 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 和 Fulfillment Manager，然后在第三台服务器上部署服务组件。下列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。

1.5 计划注意事项

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

1.5.1 服务器布局

可以在一个或多个应用程序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、应用程序服务器和数据库可以提高容错能力并改善系统性能。

开始计划的一种方法是确定要部署的 Catalog Manager 和 Vending Manager 的数量。可以将系统集中在一个 Catalog Manager，也可以选择部署多个独立的 Catalog Manager。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，或者通过为每个 Catalog Manager 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 4 向 UNIX 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，请参见正在使用的应用程序服务器的文档。

1.5.2 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。

1.5.3 容量

要开始 Content Delivery Server 的容量计划，先从所需的 Catalog Manager 和 Vending Manager 数开始。在容量计划中，必须考虑日后的增长。订户群可能会成指数增长。此外，单个造成问题的 MIDlet 或其他内容可能表明了会加重系统负担的使用峰值。

请从下列因素考虑容量估计值：

- 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、应用程序服务器和 Oracle 数据库的性能。因为要维护数据库中的订户状态，所以 Content Delivery Server 类似于其他系统，实际上几乎没有事务时间消息传送开销。

有关容量计划的其他信息，请参见《Sun Java System Content Delivery Server 容量计划指南》。

设置环境

要部署 Sun Java System Content Delivery Server，必须首先设置环境。

本节包括以下主题：

- [验证先决条件](#)
- [创建安装目录](#)
- [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且每个要在其中部署 Content Delivery Server 组件的服务器上都安装了所需的软件。

本节介绍以下要求：

- [硬件要求](#)
- [软件要求](#)
- [数据库要求](#)

2.1.1 硬件要求

下表显示了要在其中部署 Content Delivery Server 组件的所有主机所需的硬件。

表 1 Content Delivery Server 组件

组件	要求
平台	下列平台之一： <ul style="list-style-type: none">• HP-UX 11i• Sun Solaris 9
内存	一台服务器的内存至少为 512 MB
硬盘驱动器	5 GB 磁盘空间，用于系统安装、数据库和日志文件

2.1.2 软件要求

下表显示了要在其中部署 Content Delivery Server 组件的主机所需的软件。

表 2 软件要求

组件	要求
数据库	Oracle9i, 版本 9.0.1 或更高版本 只有一台主机上需要数据库。要支持英语以外的其他语言，字符集应该是 AL32UTF8。
应用程序服务器	下列应用程序服务器之一： <ul style="list-style-type: none">• WebLogic Server 7.0• Sun Java System Application Server, Standard Edition 7, Update 3 (安装 Web Server、Application Server 和 Message Queue)
Bash Shell	Bash Shell 版本 2.0.5 或更高版本 http://www.gnu.org/software/bash/bash.html 注意: Bash Shell 必须安装在 /bin 目录中。
Java 开发工具包 (JDK)	对于 WebLogic Server: Java 2 SDK, Standard Edition 1.3.1_10。 对于 Sun Java System Application Server: Java 2 SDK, Standard Edition 1.4.2_04 注意: 请使用正在与应用程序服务器一同使用的 JDK。

表 2 软件要求 (续)

组件	要求
设备库	包含应用程序可以使用的 API 的 Java 类库。 如果需要使用 iAppli 应用程序, 则需要 DoJa 库的版本 2.2。可以从 https://www.nttdocomo.co.jp/java/dl.html 获得该库 只有在其中部署 Catalog Manager 的主机才需要设备库。
浏览器	对于 Developer Portal: <ul style="list-style-type: none"> • Internet Explorer 5.5 或更高版本 • Netscape Navigator™ 4.7.4 或更高版本 对于 Subscriber Portal: <ul style="list-style-type: none"> • Internet Explorer 5.5 或更高版本 • Netscape Navigator 4.7.4 或更高版本 • 任何支持 WML 1.0、cHTML 或 XHTML 的设备浏览器。 对于 Catalog Manager 和 Vending Manager 管理控制台: <ul style="list-style-type: none"> • Internet Explorer 6.0 或更高版本 • Netscape™ 7.0 或更高版本
解压缩	解压缩 Zip 文件的实用程序。 请访问 http://www.pkware.com/downloads/ 。

2.1.3 数据库要求

Content Delivery Server 需要 Oracle 数据库 (Oracle9i 版本 9.0.1 或更高版本)。要设置 Oracle 数据库:

1. 验证 Oracle 数据库的安装是否有效。

Oracle 文档 (包括安装和升级详细资料) 可以从以下 Oracle 的 TechNet Web 站点获得: <http://technet.oracle.com/>。要支持英语以外的其他语言, 数据库的字符集应该是 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。以后配置数据库时需要此信息。

- 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

2.2 创建安装目录

可以在您选择的任何目录中安装 Content Delivery Server。如果主机中已安装了某个版本的 Content Delivery Server，请将此版本安装在其他目录。在要托管 Content Delivery Server 组件的每台计算机上都创建一个安装目录。

使用下列命令创建目录：

```
cd /  
mkdir cds-install
```

其中 *cds-install* 是您具有写入权限的任何目录的子目录。例如，`/home/cds`。

提示—要重新使用包含 Content Delivery Server 以前版本的目录，必须先卸载以前的版本。请参见第 85 页上的第 6.2 节“[卸载当前版本的 Content Delivery Server](#)”。

2.3 解压缩 Content Delivery Server 文件

发行套件 CD 包含下列文件和目录：

- `LicenseReadme`：包含与此产品相关的许可证文本文件的目录。
- `ReleaseNotes`：包含此产品发行说明的目录。应查看该发行说明以了解该产品的最新信息。
- `Proprietary_Rights_Notice.txt`：包含有关专用权限信息的文件。
- `Sun_Java_System_Content_Delivery_Server_2004Q1.zip`：包含产品文件和第三方源代码的文件。

如果收到的产品是 Zip 文件而不是 CD，则应在解压缩 Zip 文件后查看上述文件和目录。

要安装 Content Delivery Server，请使用 unzip 命令将 Sun_Java_System_Content_Delivery_Server_2004Q1.zip 中的文件解压缩到您在第 14 页上的“创建安装目录”中创建的安装目录中。例如：

```
unzip Sun_Java_System_Content_Delivery_Server_2004Q1.zip -d cds-  
install
```

此命令将解压缩文件并在 `cds-install` 目录下创建以下目录：

```
/Sun_Java_System_Content_Delivery_Server_2004Q1  
  /bin  
  /cfg  
  /deployment  
  /dist  
  /Documentation  
  /external  
  /javadoc  
  /LicenseReadme  
  /ReleaesNotes  
  /tools
```

2.4 设置环境变量

Content Delivery Server 要求在每台要托管 Content Delivery Server 组件的计算机上都正确设置环境变量：

1. 设置名为 `CDS_HOME` 的变量以引用包含产品文件的目录。

此目录即在第 14 页上的“解压缩 Content Delivery Server 文件”中解压缩产品文件时创建的目录。例如：

```
CDS_HOME=cds-install/  
Sun_Java_System_Content_Delivery_Server_2004Q1  
export CDS_HOME
```

2. 将 `/bin` 子目录添加到路径中。

例如：

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


部署 Content Delivery Server

本章提供了部署 Content Delivery Server 组件的说明。继续操作之前，请确保按照第 2 章“设置环境”说明正确设置您所在的环境。

本节介绍了以下部署类型：

- 将所有组件都部署到同一主机的基本部署。这是可实现运行的最简单配置。
- 将 Vending Manager 和支持组件部署到主机的 Vending Manager 部署。如果希望设置多售卖配置，则应进行一个基本部署和一个或多个 Vending Manager 部署。
- 使您可以创建完整定制配置并有选择地将组件部署到不同主机的高级部署。如果基本配置或多售卖配置无法符合需要，则只能执行高级部署。

本节包括以下主题：

- [基本部署](#)
 - [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 Content Delivery Server 入门的第一个部署。如果正在使用单个 Vending Manager 运行简单配置，则基本部署是唯一需要完成的部署。如果正在使用多个 Vending Manager 运行比较复杂的配置，则基本部署是需要完成的第一个部署。

3.1.1 第 1 步：编辑基本部署的 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 在 `init_env` 中设置的环境变量

变量名	说明
BEA_HOME	WebLogic Server 起始目录。将其设置为包含 <code>license.bea</code> 文件的目录。 注意： 只有通过 WebLogic Server 运行时才需要此变量。
SUN_AS_HOME	Sun Java System Application Server 的安装目录。 例如： <code>/opt/SUNWappserver7</code> 。 注意： 只有通过 Sun Java System Application Server 运行时才需要此变量。
SUN_AS_JAVA_HOME	Sun Java System Application Server 使用的 Java 主目录。必须正确设置此变量才能运行 Content Delivery Server Installation 命令 (<code>cdsi</code>) 和 Content Delivery Server Execution 命令 (<code>cdse</code>)。 注意： 只有通过 Sun Java System Application Server 运行时才需要此变量。
WL_HOME	WebLogic Server 安装目录下的 <code>server</code> 目录。将其设置为包含 <code>/bin</code> 和 <code>/lib</code> 目录的目录。例如： <code>/usr/local/bean/weblogic700/server</code> 。 注意： 只有通过 WebLogic Server 运行时才需要此变量。
WL_JAVA_HOME	WebLogic Server 使用的 Java 主目录。必须正确设置此变量才能运行 Content Delivery Server Installation 命令 (<code>cdsi</code>) 和 Content Delivery Server Execution 命令 (<code>cdse</code>)。 注意： 只有通过 WebLogic Server 运行时才需要此变量。

3.1.2 第 2 步：设置基本部署的数据库

您应已经设置了 Oracle 数据库实例，并且对此实例的下列信息有所了解：

- 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、Fulfillment Manager、每个事件服务和每个 Vending Manager 的数据库模式。Fulfillment Manager 的所有实例都必须引用同一 Fulfillment Manager 模式。

模式通过创建时提供的前缀进行识别。通常，一起部署的组件将使用同一前缀。例如，对于基本部署，所有模式的前缀都指定为同一值。

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应创建的模式。

进行基本部署时，需要为 Catalog Manager、Fulfillment Manager、Vending Manager 和事件服务创建模式。要创建基本部署模式：

1. **创建数据库配置文件。**可以将位于 `$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 1 样例数据库配置文件

```
<CDSDatabase>
  <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>

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

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

  <!-- Additional Vending Managers, if any
  <Vending name="suncds.vending.server.2">
    <Prefix>CDS_2</Prefix>
    <Password>dbpassword</Password>
  </Vending>
-->
</CDSDatabase>
```

注— 该文件包含未加密的密码。确保将文件权限设置成限制对该文件的访问。

a. 将 **Hostname**、**Port** 和 **SID** 元素设置为 **Content Delivery Server** 的 Oracle 数据库实例的主机名、端口号和数据库 **SID**。

b. 将 **Tablespace** 和 **TempTablespace** 元素设置为 **Content Delivery Server** 的 Oracle 数据库实例的名称。

如果希望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**: 该组包含不会发生更改的小型表。
- **catalog**: 该组包含从小型到中型的不经常更改的表。
- **content**: 该组包含当内容更改时经常更改的从小型到大型的表。
- **event**: 该组包含了当事务发生时发生更改的中型表。
- **subscriber**: 该组包含了当添加订户时发生更改的从小型到中型的表。
- **vending**: 该组包含了不经常更改的从小型到大型的表。

c. 将 **Catalog** 元素的 **Prefix** 和 **Password** 元素设置为用于标识和访问该模式的名称。例如，**CDS**。

Fulfillment Manager 的模式使用您为 **Catalog** 元素指定的同一前缀和密码。

d. 将 **Vending** 元素的 **Prefix** 和 **Password** 元素设置为用于标识和访问此模式的名称。例如，**CDS**。

事件服务的模式使用您为 **Vending** 元素指定的同一前缀和密码。

提示—如果知道还需要部署其他 **Vending Manager**，则可以在创建基本部署模式的同时创建所需的其他模式。对于要部署的每个其他 **Vending Manager**，请另外添加一个 **Vending** 元素，并为名称特性和 **Prefix** 元素提供唯一值。

e. 将创建的文件保存在 `$CDS_HOME/cfg` 目录中。

2. 编辑 `init_env.sh` shell 脚本，将 `DEFAULT_DB` 变量设置为刚刚创建的数据库配置文件名称。

`DEFAULT_DB` 变量指定没有为接受文件名作为参数的命令提供文件名时所使用的缺省数据库配置文件。有关此 shell 脚本的信息，请参见第 18 页上的第 3.1.1 节“第 1 步：编辑基本部署的 `init_env` Shell 脚本”。

3. 如果希望支持 **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，请参见第 71 页上的第 4.13.1 节“将 **DoJa** 库添加到数据库”。

4. 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 system-user system-password [db-configuration-file]
```

其中，`system-user` 和 `system-password` 是用于访问数据库的用户名和密码，`db-configuration-file` 是包含模式创建信息的数据库配置文件的名称。如果没有提供 `db-configuration-file`，则使用 `init_env.sh` 脚本中为 `DEFAULT_DB` 变量指定的值。

注—如果不具有创建用户帐户的权限，请让数据库管理员为您创建下列帐户：

```
catalog-prefix_ps_owner
catalog-prefix_ps_app
catalog-prefix_fs_owner
catalog-prefix_fs_app
vending-prefix_es_owner
vending-prefix_es_app
vending-prefix_vs_owner
vending-prefix_vs_app
```

然后，不要运行 `cdsi db init`，而是运行：

```
cdsi db schemas [db-configuration-file]
cdsi db data [db-configuration-file]
```

以创建模式并使用初始数据填充这些模式。

3.1.3 第 3 步：设置基本部署的部署配置文件

对数据库进行设置后，下一步就是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 “部署配置文件”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 WebLogic Server 一起使用的样例文件是 `wl_sample.cfg`（请参见第 92 页上的第 A.4 节“与 BEA WebLogic Server 一起使用的样例部署配置文件”）。与 Sun Java System Application Server 一起使用的样例文件是 `sn_sample.cfg`（请参见第 95 页上的第 A.5 节“与 Sun Java System Application Server 一起使用的样例部署配置文件”）。

要将所有 Content Delivery Server 组件部署到一个服务器，请使用下列指导来创建配置文件：

1. 从某个样例文件创建部署配置文件。
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。
有关这些属性的说明，请参见第 87 页上的第 A.1 节“常规属性”。
3. 对于每个管理器组件：
 - a. 确保未注释 `mgr_prefix.include` 属性。
 - b. 设置组件的所有属性。
有关这些属性的说明，请参见第 90 页上的第 A.2 节“管理器属性”。
4. 确保未注释 Vending Manager 部分中的 `sp.include` 属性。
5. 确保未注释 Catalog Manager 部分中的 `dp.include` 属性。

6. 对于每个服务组件：
 - a. 确保未注释 `service_prefix.include` 属性。
 - b. 设置组件的所有属性。
有关这些属性的说明，请参见第 91 页上的第 A.3 节“服务属性”。
7. 将配置文件保存在 `$CDS_HOME/cfg` 目录中。
8. 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。
`DEFAULT_CONFIG` 变量指定没有为接受文件名作为参数的命令提供文件名时所使用的缺省部署配置文件。有关此 shell 脚本的信息，请参见第 18 页上的第 3.1.1 节“第 1 步：编辑基本部署的 `init_env` Shell 脚本”。

3.1.4 第 4 步：部署

创建部署配置文件并设置环境变量后，使用以下 Content Delivery Server Install (`cdsi`) 命令来部署 Content Delivery Server 组件：

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

下表介绍了可选变量。

表 4 部署命令的可选变量

变量名	说明
<code>deployment-name</code>	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。
<code>cfg-file-name</code>	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。

注意 如果省略这些变量，将使用 `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 -h`

部署过程将部署配置文件中标识的组件安装在您指定的应用程序服务器上，并通过您为 `$CDS_HOME/deployment` 目录下的部署指定的名称创建部署目录。例如，如果将部署命名为 `cds1`，则将创建 `$CDS_HOME/deployment/cds1` 目录。

3.1.5 第 5 步：设置应用程序服务器

成功部署 Content Delivery Server 后，要使用的应用程序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 代理和应用程序服务器域。如果使用的是 WebLogic Server，则应获取通过安全套接字层 (SSL) 协议进行操作的证书、更改应用程序服务器管理员的密码和设置用于监视 Java Message Server (JMS) 连接的脚本。

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 代理和应用程序服务器域。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：

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。要使用缺省值，请在出现提示后按 `Enter`。表 5 对各个参数进行了说明。

表 5 deploymq.sh 的参数

参数	说明
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。
Broker's Port (代理的端口)	Message Queue 代理在其中进行监听的端口号。
Broker's Name (代理的名称)	希望分配给 Message Queue 代理的实例名。 注意： 只有输入 <code>server</code> 作为 <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。 注意： 只有输入 <code>server</code> 作为 <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。 注意： 只有输入 <code>server</code> 作为 <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。
Broker's User Password (代理的用户密码)	用于访问 Message Queue 代理的密码。

表 5 deploymq.sh 的参数 (续)

参数	说明
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。
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。
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"> • cdse (如果正在运行 Message Queue 的 Platform Edition)。(这是与 Solaris 9 捆绑的版本。) • try (如果正在运行 Message Queue 的 Platform Edition, 此版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)。 注意: 只有输入 server 作为 <i>type</i> 时才显示此提示。

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. 要设置应用程序服务器域, 请输入: `deploy.sh command`

其中 *command* 是 create 或 recreate。指定 create 创建新的应用程序服务器域。指定 recreate 替换现有的应用程序服务器域。

5. 输入提示您输入的信息。要使用缺省值, 请在出现提示后按 Enter。表 6 对各个参数进行了说明。

表 6 deploy.sh 的参数

参数	说明
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。
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。
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。

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

除了设置应用程序服务器域外，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。

3.1.5.2 设置 WebLogic Server

如果使用的是 WebLogic Server，则应获取通过 SSL 协议进行操作的证书、更改应用程序服务器管理员的密码和设置用于监视 Java Message Server (JMS) 连接的脚本。

获取证书

Content Delivery Server 使用 SSL（SSL 要求证书）连接到应用程序服务器。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，以便在访问应用程序服务器时使用。出于安全方面的考虑，可能会需要更改缺省的密码。

要更改缺省的密码：

1. 启动 Content Delivery Server。

请参见第 79 页上的第 5.1 节“启动 Content Delivery Server”。

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

其中 `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。

缺省的用户名是 `system`。缺省的密码是 `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` 文件以设置下列属性：

- `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。

设置监视脚本

通过 WebLogic Server 运行时，事件服务偶尔会停止接收事件。所提供的 `wljms.sh` 公用程序用来切断不传递消息的 JMS 连接。此公用程序位于 `$CDS_HOME/deployment/deployment-name/weblogic/domains/server-domain` 目录中。

要安排此公用程序在每天的午夜、早晨 6:00、中午和下午 6:00 运行，请将以下行（没有换行符）添加到 `crontab` 文件。

```
00 0,6,12,18 * * * cd cds-home-dir && ( CDS_HOME=`pwd`  
    PATH=$PATH:`pwd`/bin `pwd`/deployment/deployment-name/weblogic/  
    domains/server-domain/wljms.sh autofix ; CDS_HOME=`pwd`  
    PATH=$PATH:`pwd`/bin `pwd`/deployment/deployment-name/weblogic/  
    domains/server-domain/wljms.sh destroy )
```

其中 `cds-home-dir` 是指向 `CDS_HOME` 目录的绝对路径，`deployment-name` 是部署的名称，`domain-name` 是应用程序服务器域的名称。

要定期运行此公用程序：

1. 停止正在运行的服务。管理器应保持运行状态。
2. 输入：`wljms.sh destroy`
3. 重新启动服务。

要确定 JMS 连接是否正在传递消息，请输入：

```
wljms.sh status JMSConsumerRuntime
```

如果待传递消息数不断增长，且此连接不传递消息，您就应该切断该连接。

3.1.6 第 6 步：准备 Content Delivery Server 的使用

使部署可用之前，应配置功能并设置要使用的计划、帐户和设备。本节介绍了可能需要进行的一些准备。

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。可以设置 Subscriber Portal 使用的缺省的语言和货币符号。

[第 4 章“配置功能”](#) 描述了如何配置要使用的功能。确保在部署可用之前完成了配置。

3.1.6.2 准备 Catalog Manager

使用 Catalog Manager 管理控制台准备 Catalog Manager 的下列元素。有关启动 Catalog Manager 的信息，请参见第 80 页上的第 5.1.1 节“启动 Content Delivery Server 管理器”。有关使用管理控制台的信息，请参见《Sun Java System Content Delivery Server 管理员指南》。

注—项目 3a 是关键。如果没有设置 Vending Manager 的网络 ID，则无法将内容发布给它。

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. 定义要使用的类别。
2. 准备设备。
 - a. 确保已经提交所需的库。
 - b. 定义要使用的模板。
 - c. 添加要支持的设备。
 - d. 指定缺省设备。
3. 准备 Vending Manager 帐户。
 - a. 设置 Vending Manager 帐户的“网络 ID”字段。
 - b. 添加要使用的售卖计划。
 - c. 设置缺省的售卖计划。

注—Content Delivery Server 附带的缺省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. 设置缺省的开发者计划。

注—Content Delivery Server 附带的缺省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 的下列元素。有关启动 Catalog Manager 的信息，请参见第 80 页上的第 5.1.1 节“启动 Content Delivery Server 管理器”。有关使用管理控制台的信息，请参见《Sun Java System Content Delivery Server 管理员指南》。

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。
2. 准备订户帐户。
 - a. 添加要使用的订户计划。
 - b. 设置缺省的订户计划。

注— Content Delivery Server 附带的缺省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 Vending Manager 部署

本节介绍部署 Vending Manager 的步骤。部署 Vending Manager 时，还应部署 Fulfillment Manager 和所有支持该 Vending Manager 的服务。为了支持多售卖配置，需要在完成基本部署后再完成 Vending Manager 部署。

部署 Vending Manager 之前，必须先部署 Catalog Manager 和 Fulfillment Manager。可以将 Catalog Manager 和 Fulfillment Manager 作为基本部署的一部分（请参见第 17 页上的第 3.1 节“基本部署”）或高级部署的一部分（请参见第 32 页上的第 3.3 节“高级部署”）来部署。

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，请使用现有安装和 CDS_HOME 目录。如果主机不包含现有部署，请确保在继续操作之前按照第 2 章“设置环境”中所述设置环境。

3.2.1 第 1 步：编辑 Vending Manager 部署的 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，请设置 `init_env.sh` 脚本，如第 18 页上的第 3.1.1 节“第 1 步：编辑基本部署的 `init_env` Shell 脚本”中所述。

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 `BEA_HOME`、`JHOME`、`SUN_AS_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` 脚本还包含缺省部署、缺省数据库配置文件和缺省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缺省部署。通过缺省设置，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` 脚本中指定的缺省值。

3.2.2 第 2 步：设置 Vending Manager 部署的数据库

应已使用 Catalog Manager 和 Fulfillment Manager 的数据库模式对 Oracle 数据库实例进行了设置。要部署 Vending Manager，需要为该 Vending Manager 及其关联的事件服务创建模式。

如果在设置 Catalog Manager 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。如果没有为其他 Vending Manager 创建模式，请继续下列步骤。

要创建 Vending Manager 部署的模式：

1. 如果正在将 Vending Manager 部署到不包含其他部署的主机上，请从部署 Catalog Manager 的主机中复制数据库配置文件。将此文件保存在部署 Vending Manager 的主机上的 `$CDS_HOME/cfg` 目录中。

如果要将 Vending Manager 部署到以前部署了其他 Content Delivery Server 组件的主机上，请使用以前部署时所使用的数据库配置文件。

注—数据库配置文件包含未加密的密码。确保将文件权限设置成限制对该文件的访问。

2. 编辑数据库配置文件：
 - a. 添加名称特性和 Prefix 元素值都唯一的另一个售卖元素。另外，设置用于访问此模式的密码。

事件服务的模式使用您为 Vending 元素指定的同一前缀和密码。

注—必须对下一步（第 30 页上的第 3.2.3 节“第 3 步：设置 Vending Manager 部署的部署配置文件”）设置的部署配置文件中的 `vs.account.name` 属性使用您分配的名称。在您使用 Catalog Manager 管理控制台创建 Vending Manager 帐户时，还必须使用此名称作为帐户 ID。

b. 不要更改 Catalog 和其他售卖元素的 Prefix 元素的值。

c. 将该文件保存在 `$CDS_HOME/cfg` 目录中。

3. 编辑 `init_env.sh` shell 脚本，将 `DEFAULT_DB` 变量设置为刚刚创建的数据库配置文件的名称。

如果要将 Vending Manager 部署到以前部署了其他 Content Delivery Server 组件的主机上，请将 `DEFAULT_DB` 设置为要用作缺省值的配置文件的名称。

`DEFAULT_DB` 变量指定没有为接受文件名作为参数的命令提供文件名时所使用的缺省数据库配置文件。有关此 shell 脚本的信息，请参见第 18 页上的第 3.1.1 节“第 1 步：编辑基本部署的 `init_env` Shell 脚本”。

4. 运行以下命令以便为要部署的 Vending Manager 和事件服务创建模式：

```
cdsi db init system-user system-password db-configuration-file
vending-name
```

其中，`system-user` 和 `system-password` 是用于访问数据库的用户名和密码，`db-configuration-file` 是标识此数据库的配置文件的名称，和 `vending-name` 是为此 Vending Manager 指定的名称。

注—如果不具有创建用户帐户的权限，请让数据库管理员为您创建下列帐户：

```
vending-prefix_es_owner
vending-prefix_es_app
vending-prefix_vs_owner
vending-prefix_vs_app
```

然后，不要运行 `cdsi db init`，而是运行：

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

以创建模式并使用初始数据填充这些模式。

3.2.3 第 3 步：设置 Vending Manager 部署的部署配置文件

对数据库进行设置后，下一步就是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 21 页上的第 3.1.3 节“第 3 步：设置基本部署的部署配置文件”。

如果以前部署了 Catalog Manager 并希望将其他 Vending Manager 添加到配置中，请使用下列指导创建部署配置文件。建议部署 Vending Manager 的同时部署 Fulfillment Manager 和所有 Content Delivery Server 服务的实例。

1. 从某个样例文件创建部署配置文件。

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，则应为此部署创建新的配置文件。对同一主机上的不同部署，建议不要使用同一配置文件。

2. 设置常规属性。

有关这些属性的说明，请参见第 87 页上的第 A.1 节“常规属性”。

3. 对于 Catalog Manager 组件，请在相应行的开始位置添加 # 来注释 cs.include 属性。

4. 注释 Catalog Manager 部分中的 dp.include 属性。

5. 对于 Vending Manager 和 Fulfillment Manager:

a. 确保未注释 mgr_prefix.include 属性。

b. 将 vs.account.name 属性设置成您为您在第 29 页上的第 3.2.2 节“第 2 步: 设置 Vending Manager 部署的数据库”中设置的数据库配置文件中的 Vending 元素的名称特性指定的值。

如果在数据库配置文件中指定了多个 Vending 元素，请使用相应于当前部署的 Vending Manager 的名称。

c. 设置组件的所有属性。

有关这些属性的说明，请参见第 90 页上的第 A.2 节“管理器属性”。

6. 确保未注释 Vending Manager 部分中的 sp.include 属性。

7. 对于每个服务组件:

a. 确保未注释 service_prefix.include 属性。

b. 设置组件的所有属性。

有关这些属性的说明，请参见第 91 页上的第 A.3 节“服务属性”。

8. 将配置文件保存在 \$CDS_HOME/cfg 目录中。

9. 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。

如果要部署 Vending Manager 到以前部署了其他 Content Delivery Server 组件的主机上，请将 DEFAULT_CONFIG 设置为要用作缺省值的配置文件的名称。

DEFAULT_CONFIG 变量指定没有为接受文件名作为参数的命令提供文件名时所使用的缺省部署配置文件。有关此 shell 脚本的信息，请参见第 18 页上的第 3.1.1 节“第 1 步: 编辑基本部署的 init_env Shell 脚本”。

3.2.4 第 4 步：实施 Vending Manager 部署

有关部署 Vending Manager 和支持组件的说明，请参见第 22 页上的第 3.1.4 节“第 4 步：部署”。

3.2.5 第 5 步：设置 Vending Manager 部署的应用程序服务器

按照第 23 页上的第 3.1.5 节“第 5 步：设置应用程序服务器”中所述设置要使用的应用程序服务器。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，则应在运行 `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 第 6 步：准备 Vending Manager 以供使用

有关配置要使用的功能的信息，请参见第 4 章“配置功能”。

在系统中使用刚刚部署的 Vending Manager 之前，必须在 Catalog Manager 中创建 Vending Manager 帐户。有关信息，请参见《*Sun Java System Content Delivery Server 管理员指南*》。创建 Vending Manager 帐户时，使用的名称必须是创建数据库模式时 Vending Manager 所使用的帐户 ID。

有关应为 Vending Manager 所做的其他准备，请参见第 28 页上的第 3.1.6.3 节“准备 Vending Manager”。

3.3 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。

3.3.1 第 1 步：编辑高级部署的 `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 18 页上的第 3.1.1 节“第 1 步：编辑基本部署的 `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 `BEA_HOME`、`JHOME`、`SUN_AS_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` 脚本还包含缺省部署、缺省数据库配置文件和缺省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缺省部署。通过缺省设置，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` 脚本中指定的缺省值。

3.3.2 第 2 步：设置高级部署的数据库

无论配置的复杂程度如何，所有 `Content Delivery Server` 组件都使用同一数据库。必须在数据库中为配置中的 `Catalog Manager`、`Fulfillment Manager` 以及每个 `Vending Manager` 和事件服务创建模式。

要设置数据库：

1. 按照第 18 页上的第 3.1.2 节“第 2 步：设置基本部署的数据库”中所述创建数据库配置文件。

为所有要部署的 `Vending Manager` 创建条目。

注—数据库配置文件包含未加密的密码。确保将文件权限设置成限制对该文件的访问。

2. 将数据库配置文件复制到要部署 `Content Delivery Server` 组件的所有主机。
此文件必须位于每个主机的 `$CDS_HOME/cfg` 目录中。
3. 从要部署 `Catalog Manager` 的主机运行以下命令：

```
cdsi db init system-user system-password
```

其中 `system-user` 和 `system-password` 是用于访问数据库的用户名和密码。

如果以后要部署其他 `Vending Manager`，请按照第 29 页上的第 3.2.2 节“第 2 步：设置 `Vending Manager` 部署的数据库”中的说明创建所需的其他模式。

3.3.3 第 3 步：设置高级部署的部署配置文件

对数据库进行设置后，下一步就是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 21 页上的第 3.1.3 节“第 3 步：设置基本部署的部署配置文件”。

部署配置文件用于标识要部署的组件。每个部署都应有自己的配置文件。要创建配置文件：

1. 从某个样例文件创建部署配置文件。
此文件的扩展名必须是 `.cfg`，而且必须位于 `$CDS_HOME/cfg` 目录中。
2. 设置常规属性。
有关这些属性的说明，请参见第 87 页上的第 A.1 节“常规属性”。
3. 对于要在部署中包括的每个管理器组件：
 - a. 确保未注释 `mgr_prefix.include` 属性。注释行以 `#` 开头。
 - b. 设置组件的所有属性。
有关这些属性的说明，请参见第 90 页上的第 A.2 节“管理器属性”。
4. 如果要包括 Developer Portal，请确保未注释 Catalog Manager 部分中的 `dp.include` 属性。
5. 如果要包括 Subscriber Portal，请确保未注释 Vending Manager 部分中的 `sp.include` 属性。
6. 对于部署中不包括的各个管理器组件，请在相应行的开始位置添加 `#` 来注释 `mgr_prefix.include` 属性。
7. 如果不需要包括 Developer Portal，请注释 Catalog Manager 部分中的 `dp.include` 属性。
8. 如果不需要包括 Subscriber Portal，请注释 Vending Manager 部分中的 `sp.include` 属性。
9. 对于要在部署中包括的每个服务组件：
 - a. 确保未注释 `service_prefix.include` 属性。注释行以 `#` 开头。
 - b. 设置组件的所有属性。
有关这些属性的说明，请参见第 91 页上的第 A.3 节“服务属性”。
10. 对于部署中不包括的各个服务组件，请在相应行的开始位置添加 `#` 来注释 `service_prefix.include` 属性。
11. 将配置文件保存在 `$CDS_HOME/cfg` 目录中。

12. 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以前部署了其他 Content Delivery Server 组件的主机上，请将 `DEFAULT_CONFIG` 设置为要用作缺省值的配置文件的名称。

`DEFAULT_CONFIG` 变量指定没有为接受文件名作为参数的命令提供文件名时所使用的缺省部署配置文件。有关此 shell 脚本的信息，请参见第 18 页上的第 3.1.1 节“第 1 步：编辑基本部署的 `init_env` Shell 脚本”。

3.3.4 第 4 步：部署高级部署

有关部署的说明，请参见第 22 页上的第 3.1.4 节“第 4 步：部署”。

3.3.5 第 5 步：设置高级部署的应用程序服务器

按照第 23 页上的第 3.1.5 节“第 5 步：设置应用程序服务器”中所述设置要使用的应用程序服务器。如果使用的是 Sun Java System Application Server，请根据下列指导设置 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。
- 如果部署独立于 Catalog Manager 的 Vending Manager 或 Fulfillment Manager，应在运行 `deploymq.sh` 命令时输入 **client** 作为 `type`，从而设置 Message Queue 代理。
- 如果在一个部署中只部署一个或多个服务，且不部署任何管理器，则不要运行 `deploymq.sh` 命令。对于此部署，只需运行 `deploy.sh` 命令。

3.3.6 第 6 步：准备高级部署以供使用

有关配置已部署的组件的功能的信息，请参见第 4 章“配置功能”。

部署 Catalog Manager 之后，请参见第 27 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 32 页上的第 3.2.6 节“第 6 步：准备 Vending Manager 以供使用”。

配置功能

部署 Content Delivery Server 后，请设置需要支持的功能。确保在继续配置之前已成功部署了所需的所有 Content Delivery Server 组件，请参见第 3 章“部署 Content Delivery Server”。

本节包括以下主题：

- 配置 Content Delivery Server 服务
- 配置 SMS 推送消息
- 配置 MMS 支持
- 配置接口组件
- 为防火墙配置 Content Delivery Server
- 配置价格模型
- 配置提交检验器 workflow
- 对 Subscriber Portal 进行本地化
- 配置定制的市场活动
- 配置订户帐户管理
- 配置对外部托管的内容的支持
- 配置对定制报告的支持
- 设置对 iAppli 应用程序的支持
- 设置发送给开发者的通知
- 设置订户通知的缺省选项

4.1 配置 Content Delivery Server 服务

Content Delivery Server 服务需要配置和环境信息才能正常运行。可以通过设置每个服务使用的属性文件中的属性来提供此信息。在部署此服务的主机上设置这些属性。

4.1.1 配置消息传送服务

如果希望 Content Delivery Server 支持短消息服务 (SMS) 和无线应用程序协议 (WAP) 推送消息传送格式，则需要配置消息传送服务。

有关配置系统以支持推送传送的其他信息，请参见《*Sun Java System Content Delivery Server 集成指南*》。

4.1.1.1 配置 WAP 推送的消息传送服务

要配置 WAP 推送支持的消息传送服务，请执行以下操作：

1. 打开 `MsgService.properties` 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。

2. 设置环境所需的 WAP 和 PPG 属性。例如：

```
#WAP and PPG properties
asyncmsg.wap.ppg=
asyncmsg.wap.id=CDS
asyncmsg.wap.ip.bearer=
asyncmsg.wap.priority=high
asyncmsg.wap.bearer_type=SMS
asyncmsg.wap.bearer_required=false
asyncmsg.wap.delivery_method=unconfirmed
asyncmsg.wap.network_type=GSM
asyncmsg.wap.netwok_required=true
```

4.1.1.2 配置 SMS 的消息传送服务

Content Delivery Server 支持以下 SMS 推送协议：SMPP（对等短消息）、SMS HTTP 和 CIMD2。为使用的协议配置消息传送服务。

要配置 SMPP 推送支持的消息传送服务，请执行以下操作：

1. 打开 `MsgService.properties` 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。

2. 设置环境所需的 SMS 属性。

下面是 SMPP 推送适配器的样例属性：

```
smsc.hostname = 127.0.0.1
smsc.port = 11111
esme.system_id=user1
esme.password = usrpw
esme.system_type =
esme.destination=
smsc.gsm.ton = 2
smsc.gsm.npi = 0
```

3. 设置环境所需的推送发送器属性。

下面是 SMPP 推送适配器的样例属性：

```
pushsender.send.keep_alive=true
pushsender.keep_alive.milliseconds=30000
```

4. 设置可用于对从服务器传出的消息进行分类的属性。

是否设置这些属性均可。SMSC 可以将其用于记帐或其他目的。如果没有为任何服务类型输入值，将使用 `default.service_type` 的值。

```
subscriber_detail_url.service_type=  
mobile_originated_detail_url.service_type=  
gifting_detail_url.service_type=  
content_sharing_detail_ur.service_type=  
password_reminder.service_type=  
campaign_message.service_type=  
default.service_type=
```

要配置 CIMD2 推送支持的消息传送服务：

1. 打开 `MsgService.properties` 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。
2. 设置环境所需的 SMS 属性。

下面是 CIMD2 推送适配器的样例属性：

```
smsc.hostname = 127.0.0.1  
smsc.port = 11111  
esme.system_id=user1  
esme.password = usrpw
```

要配置 SMS HTTP 推送支持的消息传送服务：

1. 打开 `MsgService.properties` 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。
2. 设置环境所需的 SMS 属性。

下面是 SMS HTTP 推送适配器的样例属性：

```
# SMS HTTP properties  
sms.http.serverurl=  
sms.http.user=  
sms.http.password=  
sms.http.from=CDS
```

注—对于 SMS HTTP，参数名与提供的缺省值不同。另外，可能还需要编写自己的 SMS HTTP 推送适配器。要将系统配置为使用正确的适配器，请参见《*Sun Java System Content Delivery Server 集成指南*》。要编写自己的适配器，请参见《*Sun Java System Content Delivery Server 定制指南*》中的“消息传送 API”信息。

4.1.2 配置推送监听器服务

要使 Content Delivery Server 支持移动起始的推送消息，请配置推送监听器服务。支持以下两种 SMS 推送协议：SMPP 和 CIMD2。

可以将推送监听器服务配置成支持多个到 SMSC 的连接。使用的连接取决于请求的数据类型。已经定义了下列数据类型：

- 常规内容：通过标准购买过程传送的内容。
- 活动：订户请求的活动信息。

要配置推送监听器服务：

1. **编辑** `PushListener.properties` 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。
2. **设置环境所需的连接属性。**

这些属性用于所有连接。下面是 SMPP 推送监听器适配器的样例设置：

```
pushlistener.smsc.hostname = 127.0.0.1
pushlistener.smsc.port = 11111
pushlistener.esme.system_id.regular_content = user1
pushlistener.esme.password.regular_content = usrpw
pushlistener.esme.system_type=
pushlistener.esme.destination=
pushlistener.smsc.gsm.ton = 2
pushlistener.smsc.gsm.npi = 0
```

下面是 CIMD2 推送监听器适配器的样例设置：

```
pushlistener.smsc.hostname = 127.0.0.1
pushlistener.smsc.port = 11111
pushlistener.esme.system_id.regular_content = user1
pushlistener.esme.password.regular_content = usrpw
```

3. **设置要支持的连接的属性。**

要支持常规内容，需设置下列属性：

```
pushlistener.esme.system_id.regular_content =
pushlistener.esme.password.regular_content =
```

要支持活动，需设置下列属性：

```
pushlistener.esme.system_id.campaign =
pushlistener.esme.password.campaign =
```

要用单个连接支持所有数据类型：

- a. 通过在每行的开始位置加上 `#` 来注释与特定的连接类型相关的所有属性。

- b. 通过删除每行开始位置的 `#` 来取消对下列行的注释。

```
pushlistener.esme.system_id.all =
pushlistener.esme.password.all =
pushlistener.cds_mopush_url.all = mo_push.do
```

- c. 将 `pushlistener.esme.system_id.all` 和 `pushlistener.esme.password.all` 属性设置为系统所要求的值。

- d. 设置 `$CDS_HOME/deployment/deployment-name/conf` 目录下 `SubscriberPortal.properties` 文件中的下列属性。

```
mopush.regular_content.prefix=  
mopush.campaign.prefix=
```

指定的前缀必须包括在发送的 SMS 消息中。例如，如果内容 ID 是 1000，`mopush.regular_content.prefix` 是 RC，则处理程序所期望的“smsmsg”消息格式是 RC1000。

4.1.3 配置后付费服务

如果希望 Content Delivery Server 支持后付费记帐模型，则需要配置后付费服务。要配置后付费服务：

1. 打开 `PostpaidService.properties` 文件进行编辑。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。
2. 记帐事件的属性指定要处理的记帐事件。将要处理的事件设置为 `true`。将不希望处理的事件设置为 `false`。如果要处理免费试用的记帐事件，请将 `postpaid.billingevent.process_free_downloads` 设置为 `true`。

下面是记帐事件的样例设置：

```
postpaid.billingevent.type.download_initiated=false  
postpaid.billingevent.type.download_install_notified=true  
postpaid.billingevent.type.download_error=true  
postpaid.billingevent.type.content_purchased=true  
postpaid.billingevent.type.content_refunded=true  
postpaid.billingevent.type.subscription_purchased=true  
postpaid.billingevent.type.subscription_cancelled=true  
postpaid.billingevent.type.usage_purchased=true
```

```
postpaid.billingevent.process_free_downloads=false
```

3. 如果记帐系统要求不使用 Content Delivery Server 内容标识符来标识内容，请执行以下任务：
 - a. 将 `common.external_content_id.enable` 属性设置为 `true`。
如果此属性为 `true`，存储内容时管理员将收到提供记帐系统已知的内容 ID 和组 ID 的提示。
 - b. 打开 `external_content_id_selection.xml` 文件进行编辑。此文件位于 `$CDS_HOME/deployment/deployment-name/conf/resources` 目录中。

- c. 要向管理员提供已知内容 ID 的列表，请为 `content_id` 元素下的每个已知内容 ID 添加 `entry` 元素。例如：

```
<content_id>
  <entry default="true">ID-1A</entry>
  <entry>ID-1B</entry>
  <entry>ID-1C</entry>
</content_id>
```

对于自动存储的内容，外部内容 ID 将被设置为包含特性 `default="true"` 的 `entry` 元素的值。

- d. 要向管理员提供已知组 ID 的列表，请为 `group_id` 元素下的每个已知组 ID 添加 `entry` 元素。例如：

```
<group_id>
  <entry>Games</entry>
  <entry>Pictures</entry>
</group_id>
```

4. 如果不希望为记帐记录生成归档文件，请按以下方式设置下列属性：

```
postpaid.output.refresh.frequency=
postpaid.output.refresh.size=0
```

使用这些设置，可将记帐记录添加到单个文件，此文件将不断增长，直到服务停止。如果希望在编写一定数量的记录后创建归档文件，请将 `postpaid.output.refresh.size` 设置为此记录数。要每天都创建归档文件，请将 `postpaid.output.refresh.frequency` 设置为 `daily`。

5. 如果希望恢复过去的记帐记录，请设置下列属性：

- `postpaid.recovery.starting.point`：将此属性设置为表示恢复记录的时间段起始点的时间戳。必须将此值的格式指定为 `mm-dd-yyyy hh:mm:ss`，例如，
`01-01-2004 00:00:01`。
- `postpaid.recovery.stopping.point`：将此属性设置为表示恢复记录的时间段结束点的时间戳。必须将此值的格式指定为 `mm-dd-yyyy hh:mm:ss`，例如，
`01-01-2004 23:59:59`。

下次启动后付费服务时，指定时间段的记帐记录将写入 `$CDS_HOME/deployment/deployment-name/conf/Postpaid.recover` 文件。完成恢复过程后，文件将被重命名为 `Postpaid.recover.timestamp`。

如果不希望向此文件写入任何过去的记录，请将停止点的时间设置为比开始点的时间早。

4.1.4 配置监视服务

要将应用程序监视代理与网络监视系统集成，需要配置监视服务。可以将监视服务配置为只发出您感兴趣的警报，禁用您不感兴趣的警报。第 107 页上的第 C.1.1 节“应用程序监视代理”中介绍了监视代理发出的状态和警报。

要配置监视服务，请编辑 `$CDS_HOME/deployment/deployment-name/conf` 目录中的 `CDSsnmp.properties` 文件。表 7 介绍了这些属性。

表 7 监视服务属性

属性	说明
snmp.trap.disabled	表示是否应禁用所有陷阱。将其设置为 true 可以禁用所有陷阱并忽略各个陷阱的 snmp.trap.alarm.enabled 属性。将其设置为 false 可以接受各个陷阱的设置。
snmp.trap.v1	表明是生成 SNMP v1 陷阱还是生成 SNMP v2 陷阱。将此属性设置为 true 将生成 SNMP v1 陷阱。设置为 false 将生成 SNMP v2 陷阱。
snmp.agent.address	运行监视服务的主机的 IP 地址。
snmp.agent.port	运行监视服务的主机的端口号。
snmp.manager.address	运行网络管理系统的主机的 IP 地址。
snmp.manager.port	运行网络管理系统的主机的端口号。
snmp.oid.base	根节点的绝对对象标识符 (OID) 值。其他 Content Delivery Server OID 将指定为此基值的相对值。

标量对象定义

status 是提供状态的组件的名称。例如，eventServiceStatus。有关有效状态的信息，请参见第 107 页上的第 C.1.1 节“应用程序监视代理”。

snmp.object.status.oid	此对象的 MIB 标识符。 注意： 请不要更改此值。
snmp.object.status.handler	此对象的处理程序。 注意： 请不要更改此值。
snmp.object.status.attr. pidfile	包含此对象进程 ID 的文件的名称和位置。 注意： 请不要更改此值。

陷阱定义

alarm 是发出的警报的名称。例如，downloadFailure。有关有效警报的信息，请参见第 107 页上的第 C.1.1 节“应用程序监视代理”。

snmp.trap.alarm.oid	此对象的 MIB 标识符。 注意： 请不要更改此值。
snmp.trap.alarm.severity	此警报的严重程度： <ul style="list-style-type: none"> • 1 = 高 • 2 = 中 • 3 = 低
snmp.trap.alarm.enabled	表明警报是否发出。将其设置为 true 以允许发出警报。将其设置为 false 以防止发出警报。缺省值为 true。如果 snmp.trap.disabled 为 true，则忽略此设置。

4.1.5 配置确认监听器服务

如果希望 Content Delivery Server 在一步成功下载内容后处理来自设备的用户数据报协议 (UDP) 确认消息, 则需要配置确认监听器设备。要配置此设备:

1. 编辑 `ConfirmListener.properties` 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。
2. 将 `udp.confirm_listener.port` 属性设置为设备将确认消息发送到的端口号。此端口号由设备制造商和无线运营商指定。缺省值为 2003。

例如, `udp.confirm_listener.port=2003`

有关处理 MMS 确认消息的信息, 请参见《*Sun Java System Content Delivery Server 集成指南*》。

4.2 配置 SMS 推送消息

缺省的 SMS 推送消息包含来自 Content Delivery Server 的消息和要下载的内容的 URL。例如:

Download:Application Name

`http://servername:port/subscriber/main/ddd?subid=101&riid=115`

如果发送的消息长度存在问题, 则可以将此消息限制为仅为来自 Content Delivery Server 的消息或仅为 URL。编辑 `$CDS_HOME/deployment/deployment-name/conf` 目录中的 `sms_push_msg_template.xml` 模板。

以下代码显示了一个模板样例。

```
1. <?xml version="1.0" ?>
2. <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
3. version="1.0">
4. <xsl:output method="text" indent="yes" />
5. <xsl:template match="/">
6. <xsl:value-of select="/SMSMessage/MESSAGE" />
7. <xsl:value-of select="/CDSSMSMessage/HREF" />
8. </xsl:template>
9. </xsl:stylesheet>
```

要将消息限制为仅为来自 Content Delivery Server 的消息, 请删除第 7 行。要将消息限制为仅为 URL, 请删除第 6 行。

4.3 配置 MMS 支持

要使 Content Delivery Server 通过 MMS 传送内容，必须配置 VSAdminConsole.properties、SubscriberPortal.properties 和 MsgService.properties 文件中的 MMS 消息传送属性。MMSC（多媒体消息服务中心）根据您为属性输入的值来处理 MMS 消息。第 46 页的表 8 介绍了 MMS 消息传送属性及其值。

1. **编辑** VSAdminConsole.properties 和 SubscriberPortal.properties 文件。这两个文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。
2. 在每个文件中设置下列 MMS 消息传送属性。每个文件中的值都必须匹配。

下面是 MMS 支持的样例设置：

```
mms.smil.template.filename=mms_smil_template.xml
mms.message.class=AUTO
mms.message.priority=NORMAL
mms.message.sender.visibility=SHOW
mms.message.read_report_required=true
mms.message.delivery_report_required=true
admin.mms.from.address=address@host.com
```

3. 此外还需要设置 SubscriberPortal.properties 文件中的以下属性。样例设置是：

```
mms.fallback.pushtype=sms
```

注意，也必须为 Content Delivery Server 配置您指定的服务。有关配置 SMS 或 WAP 的详细信息，请参见第 37 页上的第 4.1.1 节“配置消息传送服务”。

4. **编辑** MsgService.properties 文件。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。

5. 设置 MMS 发送器类的完整类名:

此属性的样例设置是:

```
mms.senderclass=com.sun.content.server.server.msgserver.push.TestM  
MSSenderImpl
```

表 8 MMS 消息传送属性

MMS 属性名	说明
admin.mms.from.address	用于发送 MMS 内容的电子邮件或电话号码。
mms.fallback.pushtype	标识在不使用 MMS 时推送消息的替代机制。有效值是: <ul style="list-style-type: none">• sms (缺省) — 短消息服务• wap — 无线应用程序协议 假定为 Content Delivery Server 配置了替代机制。有关详细信息, 请参见第 37 页上的第 4.1.1 节“配置消息传送服务”。
mms.message.class	为 MMSC 提供消息类别。有效值是: <ul style="list-style-type: none">• 广告• 自动 (缺省)• 信息• 个人
mms.message.delivery_report_required	表明是否应从订户设备得到传送报告。有效值是: <ul style="list-style-type: none">• true — (缺省) 要求应从订户设备收到传送报告。• false — 不要求传送报告。例如, 作为促销活动一部分发送给订户的消息可以将此属性设置为 false。
mms.message.priority	为 MMSC 提供消息的优先级。有效值是: <ul style="list-style-type: none">• 高• 一般 (缺省)• 低
mms.message.read_report_required	表明是否应从订户设备获得表明消息已被阅读的消息。有效值是: <ul style="list-style-type: none">• true — (缺省) 要求应该从订户设备获得已阅读的报告。• false — 不要求已阅读的报告。例如, 作为促销活动一部分发送给订户的消息可以将此属性设置为 false。
mms.message.sender.visibility	表明显示还是隐藏消息发送者的姓名。有效值是: <ul style="list-style-type: none">• HIDE — MMS 消息中不显示发送者的姓名。• SHOW — (缺省) MMS 消息中将显示发送者的姓名。
mms.senderclass	实现 MMSSender 类的类完整名称。MMSSender 类位于 cdsapi 模块中。
mms_smil.template.filename	用于创建同步多媒体集成语言 (SMIL) 数据的 XSL 模板的名称。此属性的值表示为模板文件及其路径的名称, 且与 conf 目录相关。

4.4 配置接口组件

Content Delivery Server 的接口组件是 Developer Portal、Subscriber Portal、Catalog Manager 管理控制台和 Vending Manager 管理控制台。Content Delivery Server 提供了很多属性，可以通过设置这些属性来配置这些接口使其符合要求。每个接口自身都具有包含该接口的属性的文件，还具有一个包含适用于所有接口的属性的通用文件。

4.4.1 设置通用属性

表 9 介绍了可以在 `$CDIS_HOME/deployment/deployment-name/conf` 目录的 `CommonConsole.properties` 文件中设置的属性。这些属性适用于所有 Content Delivery Server 接口。

表 9 CommonConsole.properties 文件中的属性

特性	说明	属性
最大文件大小	允许的最大文件大小。此大小适用于提交到 Content Delivery Server 的所有文件，包括提交软件包和库。	<code>web.common.upload_max_size</code> 将其设置为要接受的最大文件的大小（以 MB 为单位）。缺省值为 10。
最大的类别深度	Catalog Manager 和 Vending Manager 的类别分层结构中最大的级别数。例如，如果将此属性设置为 3，则主类别可以具有子类别，子类别还可以具有子类别。	<code>catalog.categories.max</code> 将其设置为要支持的类别级别数。缺省值为 10。
缺省的最大应用程序大小	定义设备时用作设备的最大应用程序大小的缺省大小。	<code>handset.default.maximum.application.size</code> 将其设置为要使用的大小（以字节为单位）。缺省值为 64000。

4.4.2 设置 Developer Portal 属性

表 10 介绍了可以在 `$CDS_HOME/deployment/deployment-name/conf` 目录的 `DeveloperPortal.properties` 文件中设置的属性。这些属性仅适用于 Developer Portal。

表 10 DeveloperPortal.properties 文件中的属性

特性	说明	属性
每页的项目数	使用 Developer Portal 时每页显示的最大项目数。	<code>developer.pagination.display.max</code> 将此属性设置为希望每页显示的项目数。缺省值为 20。
页面阈值	列表导航栏在页面的底端重复之前在此页面上必须显示的项目数。如果项目数少于指定的数目，列表导航栏将仅显示在页面顶端。	<code>developer.pagination.display.threshold</code> 将此属性设置为在页面的结尾位置显示列表导航栏之前在此页面上必须显示的项目数。缺省值为 15。

4.4.3 设置 Subscriber Portal 属性

表 11 介绍了可以在 `$CDS_HOME/deployment/deployment-name/conf` 目录的 `SubscriberPortal.properties` 文件中设置的属性。这些属性仅适用于 Subscriber Portal。

表 11 SubscriberPortal.properties 文件中的属性

特性	说明	属性
自动创建设备	使订户使用的未知设备自动添加到带有 HTTP 头信息的 Content Delivery Server 数据库。由于头信息可能不完整，因此将提供用于隔离该设备的选项，直到管理员完成定义。可以在新建设备时发送通知电子邮件。	<ul style="list-style-type: none"> • <code>autoCreate.newDevice</code> 将此属性设置为 true 可以允许自动将未知设备添加到数据库。将此属性设置为 false 可以防止自动添加设备。 • <code>quarantine.newDevice</code> 将此属性设置为 true 可以隔离新添加的设备。将此属性设置为 false 可以立即使用该设备。检测到的设备可能未完整定义。 • <code>newDevice.notification.emailAddress</code> 将此属性设置为应该将新设备的通知发送到的电子邮件地址。
自动创建订户	在未注册的订户从其设备访问 Subscriber Portal 时自动创建订户。是否将新订户添加到订户数据库取决于 <code>自动置备</code> 属性的设置。	<p><code>autoCreate.newUser</code> 将此属性设置为 true 可以创建订户。 将此属性设置为 false 可以防止新建订户。</p>
自动登录用户	根据 URL 参数而不是用户名和密码来启用设备登录。基于订户 ID 的自动登录将用于查找 URL 中的 <code>subId</code> 参数。基于用户名的自动登录用于查找 URL 中的用户参数。这两个选项主要用于演示系统。其他自动登录取决于 WAP 网关传递的头。	<p><code>autoLogin.uniqueId.enable</code> <code>autoLogin.mobileId.enable</code> <code>autoLogin.subscriberId.enable</code> <code>autoLogin.username.enable</code> 将这些属性设置为 true 可以启用设备上的自动登录功能。将这些属性设置为 false 可以禁用设备上的自动登录功能。</p>
自动置备	使订户不必先注册即可从 Content Delivery Server 下载内容。如果启用了自动置备功能且未注册的订户尝试下载内容，则使用 MSISDN 号创建订户帐户并允许进行下载。	<ul style="list-style-type: none"> • <code>auto_provision.unknown.user</code> 将此属性设置为 true 可以启用自动置备功能。将其设置为 false 可以禁用自动置备功能。 • <code>auto_provision.unknown.userpeer</code> 将此属性设置为 true，订户不必先通过 Content Delivery Server 注册即可访问系统。将此属性设置为 false 可以强制订户注册。
确认购买	指定订户在每次购买内容前是否必须同意一组已定义的条款和条件。	<p><code>device.confirmPurchase</code> 将此属性设置为 true 可以在购买过程中包括“条款和条件”页面。设置为 false 可以不包括“条款和条件”页面。</p>

表 11 SubscriberPortal.properties 文件中的属性 (续)

特性	说明	属性
动态标题	指定在桌面主页上显示的动态 HTML 的路径。	dynamic_banner.default.path 将此属性设置为在桌面主页上显示的动态 HTML 的路径。
编码	指定内容类型和输出流的编码类型，每个属性中的 <i>device-id</i> 标识的设备都支持该类型。要指定编码的缺省类型，请将 0 用作设备 ID。要获得特定设备的 ID，请将鼠标放在 Catalog Manager 管理控制台“设备管理”页中的模型名上。浏览器状态区域中显示的命令中将包含此 ID。	<ul style="list-style-type: none"> • descriptor.contentType.encoding.<i>device-id</i> 将此属性设置为下载 Java 应用程序描述符 (JAD) 文件时要包含在 Content-Type HTTP 头中的缺省字符集编码字符串，例如 ISO-8859-1。如果设备无法处理 Content-Type 头中的字符集包含，请将该属性设置为 none。 • descriptor.outputStream.encoding.<i>device-id</i> 将此属性设置为输出流所使用的编码类型。例如，ISO-8859-1。
隐藏或禁用注册链接	防止未知订户访问桌面门户上的注册页面和注册服务。	<ul style="list-style-type: none"> • desktop.signup.disable_link 将此属性设置为 true 可以禁用指向注册页面的链接。设置为 false 可以启用该链接。 • desktop.signup.hide_link 将此属性设置为 true 可以隐藏指向注册页面的链接。设置为 false 可以显示该链接。
每页的项目数	使用 WML Portal 时每页显示的最大项目数。	wml.display.no_of_items_per_page 将此属性设置为希望每页显示的项目数。缺省值为 5。
负载均衡器 cookie	如果要使用负载均衡器将通信流量路由到其他 CDS 服务器，则可向每个请求添加关键字 / 值对以确保订户会话通过负载均衡器“粘到”特定服务器上。	<ul style="list-style-type: none"> • lb.cookieName 将此属性设置为由负载均衡器评估的关键字名。 • lb.cookieValue 将此属性设置为与由负载均衡器评估的关键字关联的值。
“我的活动”中的项目数	订户的“我的活动”列表中显示的项目数。将显示最新添加的项目。如果还有其他项目没有显示出来，将提供查看所有项目的链接。	desktop.display.numberOfItemsInMyCampaigns 将此属性设置为最初显示的项目数。设置为 -1 可以显示所有项目。缺省值为 5。
桌面内容链接中的项目数	搜索或浏览结果中显示的项目数。如果还有其他项目没有显示出来，将提供查看整个结果集的导航选项。	desktop.display.numberOfItemsInContentList 将此属性设置为希望每个结果页显示的项目数。缺省值为 20。

表 11 SubscriberPortal.properties 文件中的属性 (续)

特性	说明	属性
“我的下载”中的项目数	订户的“我的下载”列表中显示的项目数。将显示最新下载的项目。如果还有其他项目没有显示出来，将提供查看所有项目的链接。	desktop.display.numberOfItemsInMyPurchases 将此属性设置为最初显示的项目数。设置为 -1 可以显示所有项目。缺省值为 5。
“我的礼品”中的项目数	订户的“我的礼品”列表中显示的项目数。将显示最新添加的项目。如果还有其他项目没有显示出来，将提供查看所有项目的链接。	desktop.display.numberOfItemsInMyGiftings 将此属性设置为最初显示的项目数。设置为 -1 可以显示所有项目。缺省值为 5。
“我的愿望列表”中的项目数	订户的“我的愿望列表”中显示的项目数。将显示最新添加的项目。如果还有其他项目没有显示出来，将提供查看所有项目的链接。	desktop.display.numberOfItemsInMyWishlist 将此属性设置为最初显示的项目数。设置为 -1 可以显示所有项目。缺省值为 5。
登录尝试次数	订户在帐户被锁定之前尝试登录和失败的次数。	user.security.max_login_attempts 将此属性设置为允许的登录尝试次数。设置为 0 可以允许无限次登录。缺省值为 3。
密码提醒	订户忘记密码时显示的选项。至少将一个选项设置为 true 才能使订户收到密码提醒。	<ul style="list-style-type: none"> • password_reminder.template.filename 将此属性设置为用于生成密码提醒消息的样式表的名称。 • password_reminder.firstnameenabled • password_reminder.lastnameenabled <p>将这些属性设置为 true 可以在密码提醒屏幕上看到这些字段。设置为 false 可以隐藏这些字段。</p> <ul style="list-style-type: none"> • password_reminder.emailenabled <p>将此属性设置为 true 可以为订户提供将密码提醒发送到其电子邮件中的选项。设置为 false 可以在订户设备中隐藏此选项。</p> <ul style="list-style-type: none"> • password_reminder.webalertenabled <p>将此属性设置为 true 可以为订户提供将密码提醒作为报警发送到其设备中的选项。设置为 false 可以在订户设备中隐藏此选项。</p>

表 11 SubscriberPortal.properties 文件中的属性 (续)

特性	说明	属性
推送	启用或禁用 SMS 和 WAP 推送消息的发送。设置推送消息的缺省类型。	wap.push.enable sms.push.enable 将这些属性设置为 true 可以启用消息推送。将这些属性设置为 false 可以禁用消息推送。 default.pushType 指定要发送的推送消息的缺省类型。有效值为 sms 和 wap 。
快速购买	通过使用订户在登录时收到的数据或检索的内容验证购买, 而非验证前更新数据, 启用快速购买过程。验证检查包括验证订户是否已被禁用或者内容是否尚未激活。	quick.purchase.enabled 将此属性设置为 false 可以在购买时间进行所有验证检查。将此属性设置为 true 可以使用订户登录或上次检索内容时所检索的数据。
快速浏览	每个用户会话仅查找一次类别中的项目数和某个项目的下载次数, 从而快速装入页面。	quick.browse.enabled 将此属性设置为 false 可以每次请求时都查询计数。将此属性设置为 true 可以每个用户会话查找一次计数。
共享内容到期时间	礼品到期之前可以兑换的天数。	share_content.gifting.expirationDays 将此属性设置为可兑换礼品的天数。
共享内容模板	用于生成礼品消息的模板。	share_content.template.filename 将此属性设置为从中生成礼品消息的样式表名称。
更新许可证	表明当许可证更改时应该对订户下载内容的许可证进行更新。此属性仅适用于指定每次下载时订户都需要付费的许可证。	subscriber.every_download.update_license 将此属性设置为 true 表明当记帐模型更改时应该根据新的记帐模型对订户进行收费。设置为 false 表明应根据原始记帐模型对订户进行收费。
双向推送消息传送	启用 Content Delivery Server 可以支持来自短消息服务中心 (SMSC) 的移动起始的 SMS 消息。还必须将消息传送服务配置为支持 SMS 推送 (请参见第 38 页上的第 4.1.1.2 节“配置 SMS 的消息传送服务”)。如果更改这些属性的值, 必须重新启动服务器。	<ul style="list-style-type: none"> • mopush.accept.external.sms 将此属性设置为 true 可以接受来自任何 SMSC 的消息。将其设置为 false 仅可以接受从在 mopush.accept.moblieid 属性中指定的 SMSC 发送的消息。 • mopush.accept.mobileid 如果 mopush.accept.external.sms 设置为 false, 则将其设置为您希望从中接受消息的 SMSC 的电话号码。

4.4.4 设置 Catalog Manager 属性

表 12 介绍了可以在 `$CDS_HOME/deployment/deployment-name/conf` 目录的 `AdminConsole.properties` 文件中设置的属性。这些属性仅适用于 Catalog Manager 管理控制台。

表 12 AdminConsole.properties 文件中的属性

特性	说明	属性
每页的项目数	使用 Vending Manager 管理控制台时每页显示的最大项目数。	<code>admin.pagination.display.max</code> 将此属性设置为希望每页显示的项目数。缺省值为 20。
页面阈值	列表导航栏在页面的底端重复之前在此页面上必须显示的项目数。如果项目数少于指定的数目，列表导航栏将仅显示在页面顶端。	<code>admin.pagination.display.threshold</code> 将此属性设置为在页面的结尾位置显示列表导航栏之前在此页面上必须显示的项目数。缺省值为 15。

4.4.5 设置 Vending Manager 属性

表 13 介绍了可以在 `$CDS_HOME/deployment/deployment-name/conf` 目录的 `VSAdminConsole.properties` 文件中设置的属性。这些属性仅适用于 Vending Manager 管理控制台。

表 13 VSAdminConsole.properties 文件中的属性

特性	说明	属性
每页的项目数	使用 Vending Manager 管理控制台时每页显示的最大项目数。	<code>admin.pagination.display.max</code> 将此属性设置为希望每页显示的项目数。缺省值为 20。
页面阈值	列表导航栏在页面的底端重复之前在此页面上必须显示的项目数。如果项目数少于指定的数目，列表导航栏将仅显示在页面顶端。	<code>admin.pagination.display.threshold</code> 将此属性设置为在页面的结尾位置显示列表导航栏之前在此页面上必须显示的项目数。缺省值为 15。

4.5 为防火墙配置 Content Delivery Server

如果 Content Delivery Server 在防火墙后面运行，则还需要对配置进行其他更改。

1. 如果希望强制所有绝对 URL 生成操作都使用特定的主机和端口，请设置下列属性：

表 14 主机和端口属性

属性	值
<code>alternate.dns.host</code>	Web 服务器的已发布的外部 IP 地址。
<code>alternate.dns.port</code>	Web 服务器的已发布的外部端口号。
<code>alternate.dns.scheme</code>	所使用的方案， <code>http</code> 或 <code>https</code> 。如果未指定值，将从传入的 HTTP 请求中确定方案。缺省情况下，不指定任何值。

- 对于 Subscriber Portal，请在 `SubscriberPortal.properties` 文件中设置这些属性。
- 对于 Developer Portal，请在 `DeveloperPortal.properties` 文件中设置这些属性。
- 对于 Catalog Manager 管理控制台，请在 `AdminConsole.properties` 文件中设置这些属性。
- 对于 Vending Manager 管理控制台，请在 `VSAdminConsole.properties` 文件中设置这些属性。

这四个文件位于

`$CDS_HOME/deployment/deployment-name/conf` 目录中。

2. 将初始 Vending Manager 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 帐户。有关详细信息，请参见《*Sun Java System Content Delivery Server 管理员指南*》。

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 帐户（与要访问的 Vending Manager 相关联）所指定的域名。

4.6 配置价格模型

价格模型用于确定如何针对订户下载内容收费。Content Delivery Server 支持下列价格模型：

- **Download:** 该模型根据订户下载项目的次数来对订户收费。所有内容都支持该模型。
- **Usage:** 该模型根据订户使用项目的次数来对订户收费。只有 Java MIDlet 应用程序支持该模型。
- **Time:** 该模型根据订户使用项目的时间段来对订户收费。只有 Java MIDlet 应用程序支持该模型。

Download 价格模型始终可用。可以配置使用其他模型以满足您的需要。可以将系统设置为只包括 download 价格模型，或者设置为支持与其他价格模型的任意组合一同使用 download 价格模型。

要配置价格模型，请设置 `$CDS_HOME/deployment/deployment-name/conf/CommonConsole.xml` 文件中的 `cds.billing.models` 属性。有效值为：

- **usage:** 指定支持基于使用的价格模型，使得 Developer Portal 和管理控制台可以使用价格的“每次使用”选项。该值还使 Vending Manger 管理控制台可以使用“试用”选项。
- **subscription:** 指定支持基于时间的价格模型，使得 Developer Portal 和管理控制台中可以使用“订阅 / 每个时段”选项。
- **none:** 指定不支持 download 模型以外的任何其他价格模型。

要同时指定 **usage** 和 **time**，请用逗号分隔值。例如：

```
cds.billing.models=usage,subscription
```

如果将系统设置为提交 Java 应用程序时对该应用程序进行程序校验，则必须确保将系统配置成仅支持您的程序校验所支持的价格模型。有关详细信息，请参见第 56 页上的第 4.7.1 节“Java 应用程序的工作流”。

4.7 配置提交检验器 workflow

提交检验器 workflow 指定提交内容时 Content Delivery Server 要采取的操作。执行的 workflow 由您指定的标准确定。不要求特殊处理的内容必须由缺省的 workflow 处理。

随 Content Delivery Server 在 `$CDS_HOME/deployment/deployment-name/conf/SubmissionVerifierWorkflows.xml` 文件中提供了下列内容类型的 workflow:

- 使用 MIDP 1.0 库的 Java 应用程序
- 使用 MIDP 2.0 库的 Java 应用程序
- 使用 DoJa 库的 iAppli 应用程序
- 受版权保护的外部托管内容
- DRM (数字权限管理) 消息中打包的静态内容 (应用程序以外的内容, 如图片或铃声)
- 要在 DRM 消息中打包的静态内容
- 其他内容

如果不希望执行 workflow 中提供的操作 (如 API 过滤), 可以注释该操作的步骤。如果希望为缺省标准指定的内容以外的内容执行 workflow, 则可以更改标准, 如第 62 页上的第 4.7.7 节“指定 workflow 标准”中所述。如果希望将步骤添加到 workflow 或者要新建 workflow, 请参见《*Sun Java System Content Delivery Server 集成指南*》。

注一 SubmissionVerifierWorkflows.xml 文件必须保存为 UTF-8。请确保使用的编辑器支持该格式。

4.7.1 Java 应用程序的 workflow

Content Delivery Server 支持使用 MIDP 1.0 库或 MIDP 2.0 库的 Java 应用程序。为使用 MIDP 1.0 的 Java 应用程序所提供的 workflow 将执行内容的 API 过滤、程序校验和存储。使用 MIDP 2.0 库的应用程序的 workflow 还将签署 MIDlet。对于每个 workflow, 还会提供不对 MIDlet 进行程序校验的替换版本。

可通过注释相关步骤来禁用操作。例如, 如果不希望执行 API 过滤, 请注释 workflow 的第 2 步。

不要注释名为 AddingDerivedEdition 的最后一步。此步骤将存储订户要下载的内容版本, 此步骤为必需。

4.7.1.1 设置数字权限管理程序校验

数字权限管理 (DRM) 是 Content Delivery Server 的重要组成部分。Java 应用程序的工作流执行的程序校验将添加代码 (称为 DRM 代理), 该代码将在允许应用程序运行前验证订户的许可证。注意, 如果进行程序校验的应用程序版本不符合连接受限设备配置 (CLDC) 的技术兼容性工具包 (TCK), 则程序校验可能失败或者应用程序可能不工作。

下表介绍了随 Content Delivery Server 提供的 DRM 代理。“大小”列表明添加到应用程序的 DRM 代理的平均大小。

表 15 DRM 代理

标题	说明	大小 (平均)
Disconnected Time Sensitive	支持 download 价格模型和基于时间 (定期订阅和每个时间段) 的价格模型。运行程序校验的内容时, 不必联系 Content Delivery Server 即可验证许可证。许可证到期时, 订户必须返回到 Subscriber Portal 以购买另外一个时间段才能再次下载应用程序。	3.0 KB
Disconnected Use Sensitive	支持 download 价格模型和基于使用 (试用和每次使用) 的价格模型。运行程序校验的内容时, 不必联系 Content Delivery Server 即可验证许可证。许可证到期时, 订户必须返回到 Subscriber Portal 以购买更多的使用权限才能再次下载应用程序。	4.4 KB
Small Connected Use and Time Sensitive	支持所有价格模型。运行程序校验的内容时, 将联系 Content Delivery Server 以验证许可证。许可证到期时, 订户可以更新许可证, 不必再次下载应用程序。 注意: 为了防止运行时错误, 该 DRM 代理不应该与执行下列操作的应用程序一起使用: <ul style="list-style-type: none">• 调用 startApp 方法前显示警报。• 列出其记录存储。• 修改或删除 Content Delivery Server 记录存储。	5.1 KB
Midsized Connected Use and Time Sensitive	支持所有价格模型。运行程序校验的内容时, 将联系 Content Delivery Server 以验证许可证。许可证到期时, 订户可以更新许可证, 不必再次下载应用程序。 注意: 为了防止运行时错误, 该 DRM 代理不应该与执行下列操作的应用程序一起使用: <ul style="list-style-type: none">• 调用 startApp 方法前显示警报。• 修改或删除 Content Delivery Server 记录存储。	5.8 KB
Standard Connected Use and Time Sensitive	支持所有价格模型。运行程序校验的内容时, 将联系 Content Delivery Server 以验证许可证。许可证到期时, 订户可以更新许可证, 不必再次下载应用程序。	6.9 KB
无	仅支持 download 价格模型。不添加任何代码。	0 KB

设置程序校验

所有 DRM 代理都受提供的工作流支持。使用的 DRM 代理将由传递给工作流的变量确定。这些变量将在用于确定执行哪个工作流的标准列表中设置。下表介绍了这些变量。

表 16 工作流变量

名称	说明
allowsetcurrent2params	一个标记，表明在调用 startApp 方法前显示警报的应用程序是否应该由 Content Delivery Server 接受。将该标记设置为 true 将接受这些应用程序。设置为 false 将拒绝这些应用程序。 注意： 使用 Small Connected Use and Time 代理或 Midsize Connected Use and Time 代理时，将此标记设置为 false 将避免接受代理所不支持的应用程序。
allowlistrecordstores	一个标记，表明列出记录存储的应用程序是否应该由 Content Delivery Server 接受。将此标记设置为 true 将接受这些应用程序。设置为 false 将拒绝这些应用程序。 注意： 使用 Small Connected Use and Time 代理时，将此标记设置为 false 可避免接受代理所不支持的应用程序。
drmjarfile	包含 DRM 代理的 Java 归档 (JAR) 文件的全限定路径和名称。
drmclass	DRM 代理的全限定类名。
drmsharedjarfile	包含 DRM 代理所要求的共享类的 JAR 文件的全限定路径和名称。
drmredirectionjarfile	包含 DRM 代理所要求的重定向类的 JAR 文件的全限定路径和名称。 注意： 对于 Small Connected Use and Time Sensitive 代理，该值应该为空。
editionnamesuffix	附加到用于标识版本的内容版的名称的后缀。后缀不能多于 64 个单字节字符。

将变量设置为下列值：

■ 对于 Disconnected Time Sensitive:

```
<argument name="allowsetcurrent2params" value="true"/>
<argument name="allowlistrecordstores" value="true"/>
<argument name="drmjarfile" value="/cds-home/deployment/deployment-
name/lib/cdscontent/drmagenttime.jar"/>
<argument name="drmclass"
value="com.sun.content.instrumentation.midp.cdsdrmagent.DrmAgentTime"/>
<argument name="drmsharedjarfile" value="/cds-home/deployment/
deployment-name/lib/cdscontent/drmagentshared.jar"/>
<argument name="drmredirectionjarfile" value="/cds-home/deployment/
deployment-name/lib/cdscontent/drmredirectdisplay.jar"/>
<argument name="editionnamesuffix" value="_WithDrmAgentTime"/>
```

注—确保将系统配置成仅提供基于时间的价格模型。请参见第 55 页上的第 4.6 节“配置价格模型”。

■ 对于 Disconnected Use Sensitive:

```
<argument name="allowsetcurrent2params" value="true"/>
<argument name="allowlistrecordstores" value="true"/>
<argument name="drmjarfile" value="/cds-home/deployment/deployment-
  name/lib/cdscontent/drmagentcount.jar"/>
<argument name="drmclass"
value="com.sun.content.instrumentation.midp.cdsdrmagent.DrmAgentCount"/>
<argument name="drmsharedjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmagentshared.jar"/>
<argument name="drmredirectionjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmredirectshared.jar"/>
<argument name="editionnamesuffix" value="_WithDrmAgentCount"/>
```

注—确保将系统配置成仅提供基于使用的价格模型。请参见第 55 页上的第 4.6 节“配置价格模型”。

■ 对于 Small Connected Use and Time Sensitive:

```
<argument name="allowsetcurrent2params" value="false"/>
<argument name="allowlistrecordstores" value="false"/>
<argument name="drmjarfile" value="/cds-home/deployment/deployment-
  name/lib/cdscontent/drmagentnetthin.jar"/>
<argument name="drmclass"
value="com.sun.content.instrumentation.midp.cdsdrmagent.DrmAgentNetThin"/>
<argument name="drmsharedjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmagentshared.jar"/>
<argument name="drmredirectionjarfile" value=""/>
<argument name="editionnamesuffix" value="_WithDrmAgentNetSmall"/>
```

■ 对于 Midsize Connected Use and Time Sensitive:

```
<argument name="allowsetcurrent2params" value="false"/>
<argument name="allowlistrecordstores" value="true"/>
<argument name="drmjarfile" value="/cds-home/deployment/deployment-
  name/lib/cdscontent/drmagentnetthin.jar"/>
<argument name="drmclass"
value="com.sun.content.instrumentation.midp.cdsdrmagent.DrmAgentNetThin"/>
<argument name="drmsharedjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmagentshared.jar"/>
<argument name="drmredirectionjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmredirectlistrecordstores.jar"/
>
<argument name="editionnamesuffix"
value="_WithDrmAgentNetMidSized"/>
```

- 对于 Standard Connected Use and Time Sensitive:

```
<argument name="allowsetcurrent2params" value="true"/>
<argument name="allowlistrecordstores" value="true"/>
<argument name="drmjarfile" value="/cds-home/deployment/deployment-
  name/lib/cdscontent/drmagentnet.jar"/>
<argument name="drmclass"
value="com.sun.content.instrumentation.midp.cdsdrmagent.DrmAgentNet"/>
<argument name="drmsharedjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmagentshared.jar"/>
<argument name="drmredirectionjarfile" value="/cds-home/deployment/
  deployment-name/lib/cdscontent/drmredirectshared.jar"/>
<argument name="editionnamesuffix" value="_WithDrmAgentNet"/>
```

删除程序校验

如果不希望对 MIDlet 进行程序校验，请使用已提供的可选 MIDP 1.0 和 MIDP 2.0 工作流。通过将 MIDlet 的标准更改成指向不同的工作流来指定可选工作流的使用。

- 对于使用 MIDP 1.0 的应用程序，标准的名称是 “isMIDletMIDP1”。将工作流 id 特性更改成 “1”。
- 对于使用 MIDP 2.0 的应用程序，标准的名称是 “isMIDletMIDP2”。将工作流 id 特性更改成 “2”。

注—如果不进行任何程序校验，则只能支持 download 价格模型。确保将系统配置成仅提供该模型。请参见第 55 页上的第 4.6 节 “配置价格模型”。

4.7.1.2 设置受信证书

MIDP 2.0 应用程序的工作流要求其中含有安装所需的私钥和受信证书的密钥存储文件。使用随 JDK 提供的 keytool 公用程序来创建该文件。必须为 keyalg 参数指定 **RSA**。有关信息，请参见 JDK 文档。

编辑 SubmissionVerifierWorkflows.xml 文件中的 MIDP 2.0 Java 应用程序工作流，并提供密钥存储文件的全限定路径和文件名作为工作流第 5 步中的 MIDletSigning.KeyStoreFilePath 的值。

4.7.2 iAppli 应用程序的工作流

Content Delivery Server 支持使用 DoJa 库的 iAppli 应用程序。iAppli 应用程序的缺省工作流执行 API 过滤并将内容存储为原始形式。

可通过注释相关步骤来禁用操作。不要注释名为 AddingDerivedEdition 的最后一步。此步骤将存储订户要下载的内容版本，此步骤为必需。

4.7.3 外部托管受版权保护内容的工作流

外部托管的受版权保护的內容的工作流將标识由于版权限制不应该被 Content Delivery Server 缓存的內容。

本工作流中的两个步骤都是必需的。名为 AddingDerivedEdition 的第一步将在目录中为內容创建一个条目。名为 PreventingCopies 的第二步将确保只为內容存储元数据，內容本身不存储。

如果希望 Content Delivery Server 接受这种类型的內容，请从名为“Copyrighted External Content Workflow”的工作流中和从定义內容（将为此內容执行工作流）的标准中删除开始和结束注释语句。

只能將静态內容标记为受版权保护。要指定將为其执行该工作流的內容，请按第 62 页上的第 4.7.7 节“指定工作流标准”中所述來设置标准。

4.7.4 DRM 消息中打包的静态內容的工作流

Content Delivery Server 可以接受符合 Open Mobile Alliance (OMA) Digital Rights Management 1.0 Enabler Release 定义的在 DRM 消息中打包的静态內容。在 DRM 消息中打包的静态內容的工作流將分析软件包以验证內容。

如果希望 Content Delivery Server 接受这种类型的內容，请从名为“OMA DRM Message Parser”的工作流中和从定义內容（将为此內容执行工作流）的标准中删除开始和结束注释语句。确保该标准定义了您希望由此工作流处理的內容。有关设置标准的信息，请参见第 62 页上的第 4.7.7 节“指定工作流标准”。

除了设置工作流和标准外，还必须确保 Content Delivery Server 中定义了 MIME 类型 application/vnd.oma.drm.message 和扩展名 .dm。有关將 MIME 类型和扩展名添加到系统的信息，请参见《Sun Java System Content Delivery Server 管理员指南》。

4.7.5 要在 DRM 消息中打包的静态內容的工作流

Content Delivery Server 可以使用 OMA Digital Rights Management 1.0 Enabler Release 中定义的转发锁方法来对 DRM 消息中已提交的静态內容进行打包。用于对静态內容进行打包的工作流將创建 DRM 消息并存储订戶將下载的內容版本。

如果希望 Content Delivery Server 將静态內容打包在 DRM 消息中，请从名为“OMA DRM Message wrapper”的工作流中和从定义內容（将为此內容执行工作流）的标准中删除开始和结束注释语句。确保该标准定义了您希望由此工作流处理的內容。有关设置标准的信息，请参见第 62 页上的第 4.7.7 节“指定工作流标准”。

除了设置工作流和标准外，还必须确保 Content Delivery Server 中定义了 MIME 类型 application/vnd.oma.drm.message 和扩展名 .dm。有关將 MIME 类型和扩展名添加到系统的信息，请参见《Sun Java System Content Delivery Server 管理员指南》。

4.7.6 缺省 workflow

对于所有与任何其他 workflow 的标准不匹配的内容都执行缺省的 workflow。缺省的 workflow 将内容存储为原始形式。

不要注释名为 `AddingDerivedEdition` 的步骤。此步骤将存储订户要下载的内容版本，此步骤为必需。

4.7.7 指定 workflow 标准

每个 workflow 都应该至少有一组标准以标识应该为其执行 workflow 的内容。此标准将通过 `$CDS_HOME/deployment/deployment-name/conf/SubmissionVerifierWorkflows.xml` 文件中的 workflow 进行定义。

一项内容只执行一个 workflow。执行的 workflow 将由内容匹配的第一组标准确定，因此标准的顺序很重要。如果一组标准中指定了多个标准，则对于要视为匹配的内容而言，所有标准都必须满足。

[代码示例 2](#) 显示了外部托管的受版权保护的内容 workflow 的样例标准。

代码示例 2 受版权保护的外部内容 workflow 的标准列表

```
<criteria id="4" name="isCopyrighted">
  <desc>
    A sample validation workflow for copyrighted external content.
    Note: criterion names and values are case insensitive.
    Note: for location-type, specify e for external
    and i for internal.
  </desc>
  <workflow-list>
    <workflow id="4"/>
  </workflow-list>
  <criteria name="location-type" value="e"/>
  <criteria name="content-type" value="ringtone"/>
  <criteria name="mime-type" value="audio/mp3"/>
  <criteria name="developer-plan" value="copyrightplan"/>
  <criteria name="developer" value="composer"/>
</criteria>
```

编辑现有标注集的标准或者创建其他的标注集以标识希望由 workflow 处理的内容。对于每一组新的标准：

- 为标准 id 特性提供唯一值。
- 为 workflow id 特性指定要分配的 workflow 的 ID。例如，对于受版权保护的外部内容 workflow 而言，该值为 4。
- 如果指定的 workflow 要求输入值，则 workflow 元素内需要包括变量列表并提供变量名称和值。

- 对文件内的标准集进行排序，将限制性最强的标准集排在首位。只要内容与一组标准匹配，就不再检查任何其他标准。最后一组标准应该指定缺省的工作流，不应该包含任何标准。

将下列特性的任意组合用作一组标准的标准。如果内容与指定的所有项目匹配，则执行工作流。

- `location-type`: 为外部托管的内容指定 **e**，为本地托管的内容指定 **i**。
- `content-type`: 指定受 Content Delivery Server 支持的内容类型。
- `mime-type`: 指定受 Content Delivery Server 支持的 MIME 类型。
- `developer-plan`: 指定开发者计划的名称。
- `developer`: 指定开发者的用户名。
- `MicroEdition-Configuration`: 指定 MIDlet 的 MicroEdition Configuration 版本，如 CLDC-1.0。
- `MicroEdition-Profile`: 指定 MIDlet 的 MicroEdition Profile 版本，如 MIDP-1.0。
- `ConfigurationVer`: 指定 iAppli 应用程序的 Configuration 版本，如 CLDC-1.0。
- `ProfileVer`: 指定 iAppli 应用程序的 Profile 版本，如 DOJA-1.0。

4.8 对 Subscriber Portal 进行本地化

Subscriber Portal 对订户显示的语言是为订户使用的浏览器设置的语言。如果 Content Delivery Server 不支持该语言，则 Subscriber Portal 将以所指定的缺省语言进行显示。可以按照需要设置缺省语言，定制字符串和图像，以及设置使用的货币符号。

通过 Catalog Manager 管理控制台定义订户可用的语言。有关添加语言环境的信息，请参见《*Sun Java System Content Delivery Server 管理员指南*》。

4.8.1 设置缺省语言

缺省情况下，Subscriber Portal 显示为英语。要更改缺省语言，请将 `vending.config` 文件中的 `subscriber.default.locale` 属性设置为正确的语言代码。此文件位于 `$CDS_HOME/deployment/deployment-name/conf` 目录中。例如：

```
subscriber.default.locale=fr
```

设置缺省语言后，必须重新启动服务器。

4.8.2 提供已本地化的字符串

Subscriber Portal 上使用的文本字符串是在 `$CDS_HOME/deployment/deployment-name/localization` 目录下的 `SubscriberPortalLocaleResource.properties` 文件和 `FulfillmentLocaleResource.properties` 文件中定义的。注意，无论指定了何种语言环境，管理员输入的元素（如类别名和价格）都将按输入时的样子显示。

如果需要更改显示的字符串或者需要定制语言：

1. 编辑现有文件，或者将相应的资源文件复制到新文件。如果要创建其他语言的文件，请将相应的语言代码附加到文件名中。例如：
`SubscriberPortalLocaleResource_fr_CA.properties`。
2. 翻译等号 (=) 右侧的文本字符串。确保不要更改等号左侧的任何字符串，并且不要删除现有字符串。

更改将在保存文件后生效。

4.8.3 提供已本地化的模板和图像

对于基于 PC 的版本，除了更改 Subscriber Portal 的文本外，还可以提供语言特定的页面模板和图像。

对于需要支持的每种语言环境，在 `$CDS_HOME/deployment/deployment-name/localization` 目录中都应有与之相应的 `SubscriberPortalLocaleResource.properties` 和 `FulfillmentLocaleResource.properties` 文件的语言特定版本。例如，要支持法语，则应具有名为 `SubscriberPortalLocaleResource_fr.properties` 和 `FulfillmentLocaleResource_fr.properties` 的文件。

在每个语言特定的 `SubscriberPortalLocaleResource.properties` 文件中设置下列属性。

表 17 本地化属性

属性名	说明
<code>desktop.path.templateUrl</code>	模板文件 <code>_tpl_main.jsp</code> 的位置。此位置应为 <code>/tpl</code> 目录下的子目录。例如： <code>desktop.path.templateUrl=/tpl/_tpl_main.jsp</code>
<code>localized.path.image_url</code>	已本地化的图像的位置。此位置应为 <code>/static/media</code> 目录下的子目录。例如： <code>localized.path.image_url=/static/media/fr</code>

Subscriber Portal 的源文件位于以下目录：

- 如果使用 Sun Java System Application Server，则位于：


```
$CDS_HOME/deployment/deployment-name/sun/domains/server-domain/server-name/applications/j2ee-modules/CDSSubscriberPortal_1
```

- 如果使用 WebLogic Server，则位于：

```
$CDS_HOME/deployment/deployment-name/weblogic/domains/server-domain/applications/subscriberportal
```

deployment-name 是部署 Vending Manager 时指定的名称。*server-domain* 是在 `app.server.domain` 属性的部署配置文件中指定的域名。*server-name* 是在 `app.server.name` 属性的部署配置文件中指定的服务器名。

为在 `SubscriberPortalLocaleResource.properties` 文件中指定的模板和图像创建子目录。例如，要支持英语和法语，则可以创建以下目录结构：

```
../CDSSubscriberPortal_1  
  /tpl  
    /english  
    /french  
  /static  
    /media  
      /en  
      /fr
```

每个模板子目录都应该包含模板文件 `_tpl_main.jsp` 的本地化副本。

每个图像子目录都应该包含 Subscriber Portal 所引用的所有本地化图像的副本。与语言无关的图像应保留在 `/static/media` 根目录中。包含 Subscriber Portal 中按钮图像的文件在《*Sun Java System Content Delivery Server 署名指南*》中进行了介绍。

在模板文件中，对于根图像目录中的图像，请使用图像路径变量 `<%=img_url%>`。对于语言特定的子目录中的图像，请使用图像路径变量 `<%=localized_image_url%>`。有关编辑这些文件的其他信息，请参见《*Sun Java System Content Delivery Server 署名指南*》。

4.8.4 设置货币符号

Content Delivery Server 系统中所使用的货币都在 `$CDS_HOME/deployment/deployment-name/conf` 目录的 `CDS.properties` 文件中进行了定义。只能指定一种货币，而且该货币可以用于所有组件。

要设置货币符号，需设置 `CDS.properties` 文件中的下列属性。

表 18 货币属性

属性	说明
<code>i18n.currency.locale</code>	要使用的货币的语言环境代码。例如，对于加拿大法语，则为 <code>fr_CA</code> 。

表 18 货币属性

属性	说明
<code>i18n.currency.symbol</code>	货币符号所使用的字符。例如，美元用 \$。
<code>i18n.currency.code</code>	用于货币的代码。例如，美元使用 USD。
<code>i18n.currency.position</code>	与单位相关的符号的位置。有效值为 before 和 after 。

注—改变货币设置不会改变系统中已有的数额。如果设置是美元时输入的价格为 \$1.50，则当设置更改成英镑时价格将是 £1.50。

4.9 配置定制的市场活动

Content Delivery Server 提供了可以将广告发送到选定订户的功能。消息可以通过 SMS、WAP、MMS 或 SMTP（电子邮件）发送。消息中可以包括指向促销项目的链接。

4.9.1 配置邮件服务

以电子邮件形式发送的活动使用为消息传送服务定义的 SMTP 邮件服务。根据您在部署配置文件为消息传送服务设置的属性部署消息传送服务时，将对邮件服务进行初始配置。

如果需要更改所使用的邮件服务，请修改 `$CDS_HOME/deployment/deployment-name/conf/MsgService.properties` 文件中的下列属性：

- `mail.smtp.host`：将此属性设置为要使用的邮件服务器的名称。邮件服务器必须可以接受未经身份验证的电子邮件。
- `mail.smtp.sender`：将此属性设置为电子邮件的答复地址。

更改这些属性后，必须重新启动消息传送服务。

4.9.2 设置 Subscriber Portal URL

可以在发送给订户的消息中包含指向促销项目的链接。链接指向的地址基于为部署配置文件中的 `sp.external.uri` 属性指定的值。

如果配置文件中的值不正确，请编辑 `$CDS_HOME/deployment/deployment_name/conf` 目录中的 `CDS.properties` 文件，并将 `default.external.subscriberportal.uri` 属性的值设置为正确的地址。如果 Content Delivery Server 在防火墙后面运行，请指定订户可以从防火墙外部访问的地址。

4.10 配置订户帐户管理

订户帐户包含标识用于访问 Content Delivery Server 的订户和设备的信息。订户帐户由 Vending Manager 管理员通过 Vending Manager 管理控制台进行管理。订户还可以通过 Subscriber Portal 访问自己的帐户。

可以配置为订户和管理员显示的选项。例如，可以让订户只查看自己的帐户，或者不允许管理员删除帐户。还可以配置订户或管理员可以编辑的字段。例如，可以不允许订户更改其移动电话号码，或者不允许管理员更改订户名。

要指定可以使用的选项和可以编辑的字段，请编辑 `$CDS_HOME/deployment/deployment-name/conf` 目录中的下列文件：

- `subsubmgr.xml`：配置此文件以指定订户可以通过 Subscriber Portal 进行的操作。
- `vsadminsubmgr.xml`：配置此文件以指定管理员可以通过 Vending Manager 管理控制台进行的操作。

管理订户帐户的选项是添加、编辑和删除。这些选项在上述文件中由 `<add>`、`<edit>` 和 `<delete>` 标记表示。要使选项可用，请将 `isEnabled` 属性设置为 `true`。要隐藏选项，请将 `isEnabled` 属性设置为 `false`。缺省值为 `true`。例如，要防止订户创建帐户，请在 `subsubmgr.xml` 文件中包括以下语句：

```
<add isEnabled="false?>
```

订户帐户中的每个字段都由文件中 `<edit>` 标记下的某个标记表示。要防止字段被更改，请将 `isReadOnly` 参数设置为 `true`。要使字段可以被更改，请将 `isReadOnly` 参数设置为 `false`。缺省值为 `false`。

注一 不要更改任何标记上的 `isRequired` 参数的设置。

以下代码显示了 subsubmgr.xml 样例文件的相关部分。

```
<subscriber>
  <subscriber>
    <!-- For adding -->
    <add isEnabled="false"/>
    <!-- For editing -->
    <edit isEnabled="true">
      <password isRequired="true"/>
      <mobile_id isRequired="true" isReadOnly="true"/>

      <first_name isRequired="true"/>
      <middle_name/>
      <last_name isRequired="true"/>
      <salutation/>
      <gender/>

      <street_1/>
      <street_2/>
      <city/>
      <state/>
      <postal_code/>
      <country_code/>

      <email isRequired="true"/>
      <contact_phone/>
    </edit>
  </subscriber>
</subscriber>
```

以下代码显示了 vsadminsubmgr.xml 样例文件的相关部分。

```
<vsadmin>
  <subscriber>
    <!-- For adding -->
    <add isEnabled="true"/>
    <!-- For editing -->
    <edit isEnabled="true">
      <password isRequired="true"/>
      <status isRequired="true"/>
      <mobile_id isRequired="true"/>

      <first_name isRequired="true"/>
      <middle_name/>
      <last_name isRequired="true"/>
      <salutation/>
      <gender/>

      <street_1/>
      <street_2/>
      <city/>
      <state/>
      <postal_code/>
      <country_code/>

      <email isRequired="true"/>
      <contact_phone/>
    </edit>
    <!-- For deleting -->
    <delete isEnabled="true"/>
  </subscriber>
</vsadmin>
```

4.11 配置对外部托管的内容的支持

如果选择支持外部托管的内容，则必须设置事件服务可以用于访问 Developer Portal 的 URL。如果未正确设置此属性，则无法获取对外部托管内容的更新。

要设置 Developer Portal URL:

1. 编辑 `$CDS_HOME/deployment/deployment-name/conf` 目录中的 `CDS.properties` 文件。
2. 将 `default.internal.developerportal.uri` 属性的值设置为 **Developer Portal** 的内部地址。缺省值为 `http://localhost/developer`

4.12 配置对定制报告的支持

除了由 Vending Manager 生成的报告外，还可以对 Content Delivery Server 进行配置以便将事务数据存储在数据库表中。然后可以使用此数据创建自己的报告。

要配置对定制报告的支持：

1. 编辑 `$CDS_HOME/deployment/deployment-name/conf` 目录中的 `PostpaidService.properties` 文件。将 `postpaid.handler.class` 属性设置为报告处理程序的名称。例如：

```
postpaid.handler.class=com.sun.content.server.postpaid.impl.ReportingHandler
```

此文件中可以包含多个 `postpaid.handler.class` 属性。

2. 编辑 `$CDS_HOME/deployment/deployment-name/conf` 目录中的 `ReportService.properties` 文件。设置下列属性：

- `reportdb.username`：用于访问报告数据库的用户名。这里是 `prefix_report`，其中 `prefix` 是为 `Prefix` 元素指定的值，`Prefix` 元素在用于创建数据库的数据库配置文件中位于 `Vending` 元素的下方。有关该文件的信息，请参见第 18 页上的第 3.1.2 节“第 2 步：设置基本部署的数据库”。
- `reportdb.password`：用于访问报告数据库的密码。这是为 `Password` 元素指定的值，`Password` 元素在数据库配置文件中位于 `Vending` 元素的下方。
- `reportdb.server.url`：数据库的连接 URL。该值指定为 `jdbc:oracle:thin:@dbhost.domain.com:1521:sid`。按照需要为数据库服务器设置 `dbhost.domain.com` 和 `sid`。

注—启动 Content Delivery Server 以获取存储在报告数据库中的事务时，必须启动后付费服务。

有关为每个事务存储的数据的信息，请参见附录 D “报告”。

4.13 设置对 iAppli 应用程序的支持

要支持 iAppli 应用程序，数据库需要有一条 DoJa 库条目，而且需要将该库提交到 Content Delivery Server。

4.13.1 将 DoJa 库添加到数据库

缺省情况下，在创建 Catalog Manager 模式时将向其中添加 DoJa 库的条目，除非在创建 Catalog Manager 模式时从 `$CDS_HOME/dist/cds/database/cs` 目录中删除了 `iappli.sql` 文件。如果创建 Catalog Manager 模式时包括了 iAppli 支持，则只需提交 DoJa 库，如第 4.13.2 节“提交 DoJa 库”中所述。

如果创建 Catalog Manager 模式时未包括 iAppli 支持，请在部署 Catalog Manager 的主机上执行下列步骤。

1. 从 `$CDS_HOME/dist/cds/database/cs` 目录中移除扩展名为 `.sql` 的所有文件。将这些文件保存在其他位置。
2. 将 `iappli.sql` 文件复制到 `$CDS_HOME/dist/cds/database/cs` 目录。
3. 运行以下命令：

```
cdsi db data
```

要验证数据库中是否存在条目，请登录到 Catalog Manager 管理控制台并显示设备库列表。应该能够看到 DoJa 1.0 库的条目。

4.13.2 提交 DoJa 库

如果要支持 iAppli 应用程序，需要在完成安装后将 DoJa 库提交到 Content Delivery Server。

可以通过 Catalog Manager 管理控制台来提交库。有关启动 Catalog Manager 的信息，请参见第 80 页上的第 5.1.1 节“启动 Content Delivery Server 管理器”。

要提交 DoJa 库：

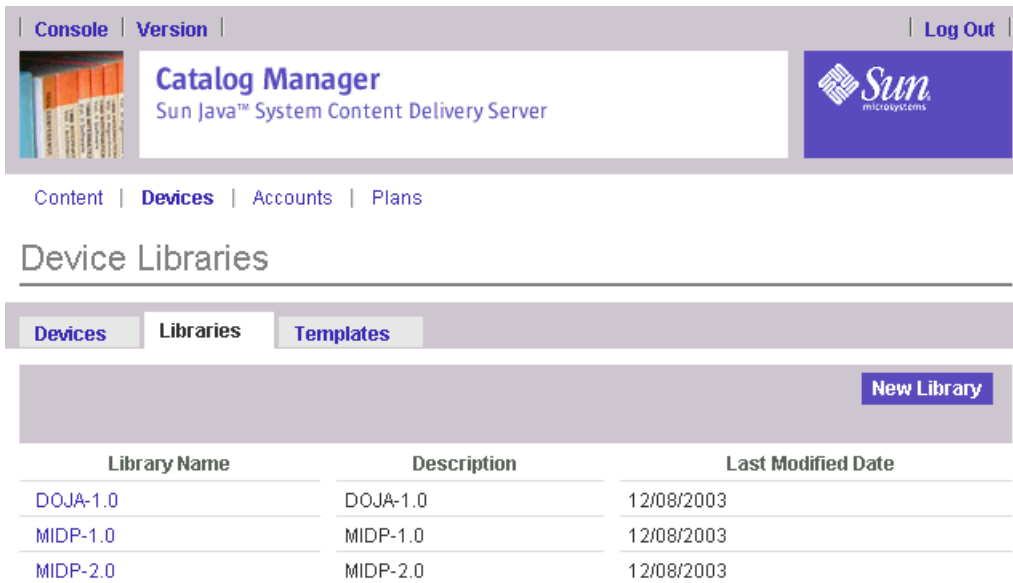
1. 通过键入以下地址从浏览器窗口中启动 Catalog Manager 管理控制台：
`http://host:port/admin/main`
此时会显示 Catalog Manager 登录页面。
2. 输入您的管理员名称和密码。
3. 单击“登录”。
将显示 Catalog Manager 主页。

4. 单击主菜单栏中的“设备”。

将显示“设备管理”页。

5. 单击“库”选项卡。

将显示“设备库”页。




The screenshot shows the Sun Catalog Manager web interface. At the top, there is a navigation bar with 'Console', 'Version', and 'Log Out'. Below this, the main header displays 'Catalog Manager' and 'Sun Java™ System Content Delivery Server' next to the Sun Microsystems logo. A secondary navigation bar includes 'Content', 'Devices', 'Accounts', and 'Plans'. The main content area is titled 'Device Libraries' and features three tabs: 'Devices', 'Libraries', and 'Templates'. A 'New Library' button is located in the top right of the content area. Below the tabs is a table listing device libraries.

Library Name	Description	Last Modified Date
DOJA-1.0	DOJA-1.0	12/08/2003
MIDP-1.0	MIDP-1.0	12/08/2003
MIDP-2.0	MIDP-2.0	12/08/2003

6. 单击“DOJA-1.0”。


显示“库定义”页。

Console | Version | Log Out



Catalog Manager

Sun Java™ System Content Delivery Server



Content | **Devices** | Accounts | Plans

Library Definition

DOJA-1.0 Properties

Upload JAR Edit Delete


Name:	DOJA-1.0
Description:	DOJA-1.0
Last Modified Date:	12/03/2003

Library Contents

OK


7. 单击“上载 Jar”。
- 将显示库的属性。

Console | Version | Log Out



Catalog Manager

Sun Java™ System Content Delivery Server



Content | **Devices** | Accounts | Plans

Library Definition

New Library Definition 1 of 2

Cancel Next

Name:	<input type="text" value="DOJA-1.0"/>
Description:	<input type="text" value="DOJA-1.0"/>
Select JAR File:	<input type="text"/> Browse...

Cancel Next

8. 在“选择 JAR 文件”中输入库文件的路径，或者单击“浏览”查找库文件。
- 不要更改库的名称。

9. 单击“下一步”。

将显示库中包含的可用项目。

Console | Version | Log Out

Catalog Manager
Sun Java™ System Content Delivery Server

Content | **Devices** | Accounts | Plans

Library Definition

New Library Definition 2 of 2

Cancel **Back** **Finish**

Name:

Description:

Select items to include in this Library:

Global Package

Cancel **Back** **Finish**

10. 选中“全局软件包”复选框。

11. 单击“确定”。

库将添加到数据库中。然后即可使用开发者计划中的库来限制开发者使用的 API。

12. 单击“确定”，关闭确认页。

注一重新安装数据库时必须重新提交库。

4.14 设置发送给开发者的通知

开发者可以通过 Developer Portal 将内容提交到 Content Delivery Server。可以将 Content Delivery Server 设置为只要对开发者提交的内容进行操作时就向其发送电子邮件通知。

要设置通知：

1. 确保 `$CDS_HOME/deployment/deployment-name/conf/CDS.properties` 文件中的 `default.external.developerportal.uri` 属性指向 Developer Portal 所在的位置。

此值的格式应为 `http://hostname:port/developer/`，其中 `hostname:port` 是运行 Developer Portal 的主机的名称和端口号。

2. 设置 `$CDS_HOME/deployment/deployment-name/conf/EventService.properties` 文件中的下列属性。

表 19 开发者通知的属性

属性	说明
<code>eventservice.developer.email.enabled</code>	电子邮件通知的状态。将此属性设置为 true 可以在对内容进行操作时自动发送通知。设置为 false 可以禁用发送通知。缺省值为 false 。
<code>eventservice.developer.email.from</code>	从中发送通知的地址。
<code>eventservice.developer.email.admin</code>	当开发者对通知有疑问时可以将响应发送到的地址。
<code>eventservice.developer.email.template.propertychanged</code>	XSL 文件（其中包含在更改任何内容属性时发送的通知说明）的全限定位置。
<code>eventservice.developer.email.template.submitted</code>	XSL 文件（其中包含当 Content Delivery Server 接收或拒绝内容时发送的通知说明）的全限定位置。
<code>eventservice.developer.email.template.statuschanged</code>	XSL 文件（其中包含在更改内容状态时发送的通知说明）的全限定位置。

3. 如果需要定制要发送的通知，请编辑随 Content Delivery Server 提供的缺省模板，或者创建自己的 XSL 文件。

代码示例 3 提供了应遵循的通知模板的定义。如果创建自己的文件，则必须更新上一步骤中说明的标识文件位置的属性。

代码示例 3 通知模板的定义

```
/*
 * Defintion of XML documents that are piped to the
 * XSL transformation sheets.
 *
 * <dn>
 *   <developer>Pawel S. Veselov</developer>
 *
 *   <submission> <!-- when content was submitted (even if failed) -->
 *     <succeeded> <!-- may be zero or more -->
 *       <name>Name used to submit</name>
 *       <id>ID used to submit</id>
 *       <url>URL this content is viewable at</url>
 *     </succeeded>
 *     <failed> <!-- may be zero or more -->
 *       <name>Name used to submit</name>
 *       <id>ID used to submit</id>
 *       <errmsg>Error message</errmsg>
 *       <errorlog>Lengthy error info</errorlog>
 *     </failed>
 *   </submission>
 *
 *   <statuschanged>
 *
 *     <!-- The possible status values are :
 *
 *     pending
 *     denied
 *     published
 *     deleted
 *     new
 *
 *     -->
 *
 *     <newstatus>status</newstatus>
 *
 *     <name>Named used to submit</name>
 *     <id>content id</id>
 *     <url>URL where this content is accessible</url>
 *     <message>Message left for developer</message>
 *   </statuschanged>>
 *
 *   <propertychanged>
 *     <name>Named used to submit</name>
 *     <id>content id</id>
 *     <url>URL where this content is accessible</url>
 *     <textproperty> <!-- zero or more of those -->
 *       <name>property name</name>
 *       <oldvalue>old property value</oldvalue>
 *       <newvalue>new property value</newvalue>
 *     </textproperty>
```

```

*
* <!-- binary property is special, since it doesn't have a
*      displayable value, so only name is mentioned.
* -->
* <binaryproperty>property name</binaryproperty> <!-- zero or
*      more -->
* <priceproperty> <!-- zero or one of this, only for suggested
*      price -->
* <!-- note the pricedata tag.It is used to unify the
*      structure to make XSLT transformation easier.
* -->
* <oldvalue><pricedata>
*   <billing1>
*     <!-- the following is $13.27, but helps localization
*           if broken in such a way.There is also a "number"
*           tag which gives the price in cents (coins)
*     -->
*     <price>
*       <number>1327</number>
*       <bills>13</bills>
*       <coins>27</coins>
*     </price>
*     <usage_count>usage_count</usage_count>
*     <usage_period_days>usage_period_days</usage_period_days>
*     <recurring>recurring</recurring>
*   </billing1>
*   <billing2>
*     <price><bills>13</bills><coins>27</coins></price>
*     <frequency>frequency</frequency>
*     <recurring>recurring</recurring>
*   </billing2>
*   <billing3>
*     <price><bills>13</bills><coins>27</coins></price>
*     <usage_count>usage_count</usage_count>
*   </billing3>
* </pricedata></oldvalue>
* <newvalue><!-- the same as for olvalue --> </newvalue>
* </priceproperty>
*
* </propertychanged>
*
* <adminemail>Email address developer can inquire at</adminemail>
* </dn>
*/

```

4.15 设置订户通知的缺省选项

Content Delivery Server 会将关于更新或促销的通知发送给订户。订户在设置帐户首选项时，可以通过设置选项来选择是否接收这些通知。可以将缺省值设置为 **opt-in** 或 **opt-out**。初始缺省值为 **opt-in**。

要更改缺省的选项，需要设置 `$CDS_HOME/deployment/deployment-name/conf/security.config` 文件中的 `user.profile.optin` 属性。将该属性设置为 **true** 表示将缺省值设置为 **opt-in**。将该属性设置为 **false** 表示将缺省值设置为 **opt-out**。

操作 Content Delivery Server

Sun Java System Content Delivery Server 旨在为运营商提供不间断的服务。正确配置每个部署后，启动 Content Delivery Server 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 的所有组件。与 Sun Java System Application Server 一起运行时，还必须启动 Sun Java System Message Queue 代理。

- [启动 Content Delivery Server 管理器](#)
- [启动 Content Delivery Server 服务](#)

5.1.1 启动 Content Delivery Server 管理器

使用 Content Delivery Server Execute 命令 (cdse) 启动 Catalog Manager、Vending Manager 和 Fulfillment Manager。如果与 Sun Java System Application Server 一起运行，则必须先使用以下命令启动 Sun Java System Message Queue 代理：

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

注—如果在 HP-UX 平台上运行，请确保在启动 Content Delivery Server 组件之前将打开文件的最大数目设置为 512 或更大的数目。要了解当前的限制，请输入：

```
ulimit -a
```

要对限制进行设置，请输入：

```
ulimit -n nn
```

其中 *nn* 是大于等于 512 的数。

要启动在主机上部署的 Content Delivery Server 管理器，请输入以下命令：

```
cdse startserver [deployment-name]
```

如果省略 *deployment-name*，将使用为 `init_env` 脚本中的 `DEFAULT_DEPL` 所指定的值。

启动管理器后，将收到以下响应之一，这取决于您使用的应用程序服务器：

- WebLogic Server 报告 `Listening to port nn`（其中 *nn* 是端口号）。不会返回命令行提示。
 - Sun Java System Application Server 报告 `Instance server-name started` 并返回命令行提示。
-

提示—Catalog Manager 和 Vending Manager 管理控制台的缺省用户名是 **admin**。缺省的密码是 **admin**。首次登录时必须更改密码。

Subscriber Portal 或 Developer Portal 没有缺省帐户。

5.1.2 启动 Content Delivery Server 服务

使用 Content Delivery Server Execute 命令 (cdse) 启动 Content Delivery Server 服务组件。只有在使用 CIMD2 推送适配器时才需要启动推送监听器服务。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。只有在支持一步下载后发送确认消息的设备时才需要启动确认监听器。只有在记帐系统使用由 Content Delivery Server 生成的电话数据记录 (CDR) 时才需要启动后付费服务。

要启动所有服务，请输入以下命令：

```
cdse spawnservice all [deployment-name]
```

要启动单个服务，请输入以下命令：

```
cdse startservice component [deployment-name] &
```

表 20 显示了 *component* 的有效值。

表 20 cdse 命令的服务组件值

服务组件值	说明
es	事件服务
ms	消息传送服务
pl	推送监听器服务
ns	通知服务
pp	后付费服务
cl	确认监听器服务
mo	监视服务

如果省略 *deployment-name*，将使用为 `init_env` 脚本中的 `DEFAULT_DEPL` 所指定的值。

5.2 停止 Content Delivery Server

使用 Content Delivery Server Execute (`cdse`) 命令以手动停止 Content Delivery Server 的所有组件。与 Sun Java System Application Server 一起运行时，还必须停止 Sun Java System Message Queue 代理。

- 停止 Content Delivery Server 管理器
- 停止 Content Delivery Server 服务
- 停止 Sun Java System Message Queue

5.2.1 停止 Content Delivery Server 管理器

使用 Content Delivery Server Execute 命令 (`cdse`) 停止在服务器上运行的所有管理器组件：

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

如果省略 *deployment-name*，将使用为 `init_env` 脚本中的 `DEFAULT_DEPL` 所指定的值。

5.2.2 停止 Content Delivery Server 服务

要停止所有服务，请输入以下命令：

```
cdse stopservice all [deployment-name]
```

要停止单个服务，请输入以下命令：

```
cdse stopservice component [deployment-name]
```

第 81 页的表 20 显示了 *component* 的有效值。

如果省略 *deployment-name*，将使用为 `init_env` 脚本中的 `DEFAULT_DEPL` 所指定的值。

如果与 WebLogic Server 一起运行并停止和重新启动事件服务或消息传送服务，则必须重新启动在该服务连接的应用程序服务器上安装的 Content Delivery Server 组件。

- 如果部署了基本部署（同一应用程序服务器上的所有 Content Delivery Server 组件），则必须在重新启动基本部署中部署的事件服务或消息传送服务后停止和重新启动 Content Delivery Server 管理器。
- 如果部署了 Vending Manager 部署，则必须在重新启动通过该 Vending Manager 部署的事件服务或消息传送服务后停止和重新启动 Vending Manager。

5.2.3 停止 Sun Java System Message Queue

如果与专用于 Content Delivery Server 的 Sun Java System Application Server 一起运行，则应在停止 Content Delivery Server 后停止 Sun Java System Message Queue（请参见第 81 页上的第 5.2 节“停止 Content Delivery Server”）。

要停止 Message Queue，请输入：

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

卸载 Content Delivery Server

如果不再需要某个 Content Delivery Server 实例，则可以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](#)
- [卸载当前版本的 Content Delivery Server](#)
- [卸载以前版本的 Content Delivery Server](#)

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 WebLogic Server 上的部署

要删除 WebLogic Server 上的部署，请执行以下操作：

1. 使用以下命令停止所有服务：

```
cdse stopservice all [deployment-name]
```

其中 *deployment-name* 是要删除的部署的名称。

2. 使用以下命令停止所有管理器：

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

其中 *deployment-name* 是要删除的部署的名称。

3. 使用以下命令删除数据库模式：

```
cdsi db delete system-user system-password [database-configuration]
```

其中 *system-user* 和 *system-password* 分别为用于访问数据库的用户名和密码，*database-configuration* 为用于创建模式的数据库配置文件的名称。

4. 转到 `$CDS_HOME` 目录。
5. 使用以下命令删除包含部署的目录：

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

其中 *deployment-name* 是要删除的部署的名称。
6. 删除为运行 `wljms.sh` 实用程序而为 `crontab` 文件创建的所有条目。

6.1.2 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. 使用以下命令停止所有服务：

```
cdse stopservice all [deployment-name]
```

其中 *deployment-name* 是要删除的部署的名称。
3. 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
```
4. 使用以下命令删除应用程序服务器域：

```
deploy.sh delete
```

注一必须是超级用户才能执行此命令。

5. 使用以下命令删除数据库模式：

```
cdsi db delete system-user system-password [database-configuration]
```

其中 *system-user* 和 *system-password* 分别为用于访问数据库的用户名和密码，*database-configuration* 为用于创建模式的数据库配置文件的名称。
6. 转到 `$CDS_HOME` 目录。
7. 使用以下命令删除包含部署的目录：

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

其中 *deployment-name* 是要删除的部署的名称。

6.2 卸载当前版本的 Content Delivery Server

如果不再需要任何与主机上的 Content Delivery Server 相关联的文件，则可以删除整个产品。

要卸载整个产品，请执行以下操作：

1. 卸载该主机上的所有部署。请参见第 83 页上的第 6.1 节“卸载部署”。
2. 使用以下命令删除在第 14 页上的第 2.2 节“创建安装目录”中创建的安装目录：

```
rm -rf cds-install
```

其中 *cds-install* 是安装目录的名称。
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` 目录的内容以清理可能由 Content Delivery Server 创建的所有临时文件。

6.3 卸载以前版本的 Content Delivery Server

要卸载以前版本的 Content Delivery Server，请执行以下操作：

1. 删除 `$PIXO_HOME` 目录及其文件和子目录。
2. 删除所有永久性环境变量设置以及引用 Content Delivery Server 的任何数据库或应用程序服务器配置文件更改。
3. 重新引导主机以确保删除由 Content Delivery Server 使用的所有临时文件。

部署配置文件

本附录详细介绍了用于部署 Content Delivery Server 组件的部署配置文件的格式。此文件格式旨在通过 ANT “替换”命令进行操作。有关使用配置文件的信息，请参见第 21 页上的第 3.1.3 节“第 3 步：设置基本部署的部署配置文件”。

属性按以下方式进行分组：

- 常规属性
- 管理器属性
- 服务属性

本附录还包括用于说明设置用法的样例部署配置文件。

- 与 BEA WebLogic Server 一起使用的样例部署配置文件
- 与 Sun Java System Application Server 一起使用的样例部署配置文件

A.1 常规属性

下表说明了在使用 WebLogic Server 或 Sun Java System Application Server 时所用配置文件中的常规属性。必须对这些属性进行设置。表 22 中说明了仅由 WebLogic Server 使用的属性。

表 21 常规属性

属性	说明
server.type	<p>指定要部署到的应用程序服务器的类型。对于 WebLogic Server, 请使用 <code>weblogic</code>; 而对于 Sun Java System Application Server, 请使用 <code>sun</code>。例如, <code>server.type=weblogic</code></p> <p>注意: 不要更改此属性的值。</p>
java.mem	<p>为应用程序服务器启动 JVM 时所使用的字符串, 它将决定 JVM 的内存设置。例如, <code>java.mem=-ms512m -mx1024m -XX:MaxPermSize=256m</code></p>
app.server.domain	<p>应用程序服务器的域名。例如, <code>app.server.domain=cdsdomain</code></p>
app.server.name	<p>应用程序服务器使用的名称。例如, <code>app.server.name=cdsserver</code></p>
app.server.portnumber	<p>应用程序服务器将在其上监听 HTTP 请求的端口。例如, <code>app.server.portnumber=80</code></p>
cs.internal.uri	<p>Content Delivery Server 中使用的 Catalog Manager 的 URI。例如, <code>cs.internal.uri=http://localhost:80/admin/</code></p>
dp.internal.uri	<p>Content Delivery Server 中使用的 Developer Portal 的 URI。例如, <code>dp.internal.uri=http://localhost:80/developer/</code></p>
fs.internal.uri	<p>Content Delivery Server 中使用的 Fulfillment Manager 的 URI。例如, <code>fs.internal.uri=http://localhost:80/fs/</code></p>
vs.internal.uri	<p>Content Delivery Server 中使用的 Vending Manager 的 URI。例如, <code>vs.internal.uri=http://localhost:80/vsadmin/</code></p>
sp.internal.uri	<p>Content Delivery Server 中使用的 Subscriber Portal 的 URI。例如, <code>sp.internal.uri=http://localhost:80/subscriber/</code></p>
cs.external.uri	<p>外部实体访问 Catalog Manager 时所使用的 URI。例如, <code>cs.external.uri=http://webhost.domain.com:80/admin/</code></p>
dp.external.uri	<p>外部实体访问 Developer Portal 时所使用的 URI。例如, <code>dp.external.uri=http://webhost.domain.com:80/developer/</code></p>

表 21 常规属性 (续)

属性	说明
<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:80/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:80/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:80/subscriber/</code>

下表说明了仅在使用 WebLogic Server 时才使用的属性。如果未使用 WebLogic Server，则不需要这些属性。

表 22 与 WebLogic Server 一起使用的常规属性

属性	说明
<code>app.server.threadcount</code>	应用程序服务器处理 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 false 可以禁用应用程序服务器的 SSL。例如， <code>app.server.ssl.enabled=false</code> 。 缺省值为 true 。
<code>app.server.ssl.port</code>	应用程序服务器将在其上监听 SSL HTTP 请求的端口号。例如， <code>app.server.ssl.port=81</code>
<code>app.server.hostip</code>	在其中部署服务器的主机的主机名或 IP 地址。例如， <code>app.server.hostip=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>
<code>jndi.host</code>	提供 JNDI 服务的主机的名称。这应是 IP 地址或可解析的 DNS 名称。例如， <code>jndi.host=localhost</code>
<code>jndi.port</code>	JNDI 主机在其上监听 JNDI 请求的端口号。指定为 <code>jndi.host</code> 指定的主机所使用的端口号。例如， <code>jndi.port=80</code>

A.2 管理器属性

三个 Content Delivery Server 管理器组件（Vending Manager、Catalog Manager 和 Fulfillment Manager）都有类似的属性集。特定管理器的属性由属性前缀进行标识。下表说明了这些前缀：

表 23 Content Delivery Server 管理器组件前缀

前缀	说明
vs	Vending Manager
cs	Catalog Manager
fs	Fulfillment Manager

下表介绍了管理器属性。

表 24 管理器属性

属性	说明
<code>prefix.include</code>	此属性用于确定部署的配置中是否包括由 <i>prefix</i> 标识的组件。如果组件不应部署到给定的部署，请在行的前面加上 # 前缀以注释相应的 <i>prefix.include</i> 属性。
<code>vs.account.name</code>	当前部署的 Vending Manager 的名称。例如， <code>suncds.vending.server.1</code> 。该名称必须与您为数据库配置文件中的相应 Vending 元素指定的名称特性匹配，而且当您在 Catalog Manager 中创建 Vending Manager 帐户时该名称必须用作帐户 ID。 此属性只适用于 Vending Manager。

A.3 服务属性

除了部署和配置 Vending Manager、Catalog Manager 及 Fulfillment Manager 外，还应部署 Content Delivery Server 服务。与上述的管理器属性一样，每个服务的所有属性都由唯一前缀来标识。下表介绍了当前的服务及其前缀。

表 25 Content Delivery Server 服务前缀

前缀	说明
es	事件服务
ms	消息传送服务
pl	推送监听器服务
ns	通知服务
pp	后付费服务
cl	确认监听器服务
mo	监视服务

下表介绍了服务属性。

表 26 服务属性

属性	说明
<code>prefix.include</code>	对于服务器组件，这些属性确定是否应部署指定的服务。如果不需要部署特定服务，请注释此属性。
<code>prefix.java.mem</code>	启动服务时使用的内存设置。例如， <code>es.java.mem=-ms64m -mx128m</code>
<code>prefix.jmx.port</code>	Java Management Extensions 代理在其上进行监听的端口。例如， <code>es.jmx.port=8082</code> 此属性只适用于事件服务和消息传送服务。 注意： 在同一主机上部署的每个服务的端口号必须各不相同。
<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。 此属性只适用于消息传送服务。
<code>ms.mail.smtp.sender</code>	电子邮件的回复地址。 此属性只适用于消息传送服务。

A.4 与 BEA WebLogic Server 一起使用的样例部署配置文件

这是显示本附录中所述配置设置的样例配置文件。此文件位于 `$CDS_HOME/cfg` 目录中，名为 `wl_sample.cfg`。使用该文件之前，应先编辑属性。很多样例设置对于您的环境都不正确。

注意，Content Delivery Server 将忽略以 `#` 开始的行。

代码示例 4 与 WebLogic 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:
#
# VENDING MANAGER          (vs)
# CATALOG MANAGER          (cs)
# FULFILLMENT MANAGER      (fs)
# EVENT SERVICE            (es)
# MESSAGING SERVICE        (ms)
# POSTPAID SERVICE         (pp)
# NOTIFY SERVICE           (ns)
# CONFIRMLISTENER SERVICE (cl)
# PUSHLISTENER SERVICE    (pl)
# MONITORING SERVICE       (mo)
#
# Each of these components has its own properties that are used to
# configure it.
#
# 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=-ms512m -mx1024m -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
    # Host Name or IP address of the system where server is
    # deployed
    # You should change it if you running multiple IP addresses
app.server.hostip=localhost
    # WL administration console port, ONLY IF SSL IS ENABLED
app.server.admin.port=7800

# ----- JNDI Server Settings ----- #
jndi.host=jndihost.domain.com
jndi.port=80

# ----- URI Settings ----- #
cs.internal.uri=http://localhost:80/admin/
dp.internal.uri=http://localhost:80/developer/
fs.internal.uri=http://localhost:80/fs/
vs.internal.uri=http://localhost:80/vsadmin/
sp.internal.uri=http://localhost:80/subscriber/
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/

#=====
# Catalog Manager & Developer Portal
#=====
# Make the following line a comment if the catalog manager is not to
# be deployed and configured
cs.include
# Make the following line a comment if the developer portal is not
# to be deployed and configured
dp.include

#=====
# Vending Manager & Subscriber Portal
#=====
# Make the following line a comment if the vending manager is not to
# be deployed and configured
vs.include
# Make the following line a comment if the subscriber portal is not
# to be deployed and configured
sp.include

```

```

#=====
# Fulfillment Manager
#=====
# Make the following line a comment if the fulfillment manager is
# not to be deployed and configured
fs.include

#=====
# Event Service
#=====
# Make the following line a comment if the event service is not to
# be deployed and configured
es.include

# ----- Java Settings ----- #
es.java.mem=-ms64m -mx128m

# ----- Service Settings ----- #
es.jmx.port=7801

#=====
# Messaging Service
#=====
# Make the following line a comment if the messaging service is not
# to be deployed and configured
ms.include

# ----- Java Settings ----- #
ms.java.mem=-ms64m -mx128m

# ----- Service Settings ----- #
ms.jmx.port=7802

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

#=====
# Postpaid Service
#=====
# Make the following line a comment if the postpaid service is not
# to be deployed and configured
pp.include

# ----- Java Settings ----- #
pp.java.mem=-ms64m -mx128m

#=====
# Notify Service
#=====
# Make the following line a comment if the notify service is not to

```

```

# be deployed and configured
ns.include

# ----- Java Settings ----- #
ns.java.mem=-ms64m -mx128m

#####
# ConfirmListener Service
#####
# Make the following line a comment if the confirmlistener service
# is not to be deployed and configured
cl.include

# ----- Java Settings ----- #
cl.java.mem=-ms64m -mx128m

#####
# PushListener Service
#####
# Make the following line a comment if the pushlistener service is
# not to be deployed and configured
pl.include

# ----- Java Settings ----- #
pl.java.mem=-ms64m -mx128m

#####
# Monitoring Service
#####
# Make the following line a comment if the monitoring service is not
# to be deployed and configured
mo.include

# ----- Java Settings ----- #
mo.java.mem=-ms64m -mx128m

```

A.5 与 Sun Java System Application Server 一起使用的样例部署配置文件

这是显示本附录中所述配置设置的样例部署配置文件。此文件位于 `$CDS_HOME/cfg` 目录中，名为 `sn_sample.cfg`。使用该文件之前，必须先编辑属性。很多样例设置对于您的环境都不正确。

注意，Content Delivery Server 将忽略以 `#` 开始的行。

代码示例 5 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:
#
# VENDING MANAGER          (vs)
# CATALOG MANAGER         (cs)
# FULFILLMENT MANAGER     (fs)
# EVENT SERVICE           (es)
# MESSAGING SERVICE       (ms)
# POSTPAID SERVICE        (pp)
# NOTIFY SERVICE          (ns)
# CONFIRMLISTENER SERVICE (cl)
# PUSHLISTENER SERVICE    (pl)
# MONITORING SERVICE      (mo)
#
# Each of these components has its own properties that are used to
# configure it.
#
# 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=-ms512m -mx1024m -XX:MaxPermSize=256m

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

# ----- URI Settings ----- #
cs.internal.uri=http://localhost:80/admin/
dp.internal.uri=http://localhost:80/developer/
fs.internal.uri=http://localhost:80/fs/
vs.internal.uri=http://localhost:80/vsadmin/
sp.internal.uri=http://localhost:80/subscriber/
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/

#####
# Catalog Manager & Developer Portal
#####
# Make the following line a comment if the catalog manager is not to
# be deployed and configured
cs.include
# Make the following line a comment if the developer portal is not
# to be deployed and configured
dp.include

#####
# Vending Manager & Subscriber Portal
#####
# Make the following line a comment if the vending manager is not to
# be deployed and configured
vs.include
# Make the following line a comment if the subscriber portal is not
# to be deployed and configured
sp.include

#####
# Fulfillment Manager
#####
# Make the following line a comment if the fulfillment manager is
# not to be deployed and configured
fs.include

#####
# Event Service
#####
# Make the following line a comment if the event service is not to
# be deployed and configured
es.include

# ----- Java Settings ----- #
es.java.mem=-ms64m -mx128m

# ----- Service Settings ----- #
es.jmx.port=7801

#####
# Messaging Service
#####
# Make the following line a comment if the messaging service is not
# to be deployed and configured
ms.include

```

```

# ----- Java Settings ----- #
ms.java.mem=-ms64m -mx128m

# ----- Service Settings ----- #
ms.jmx.port=7802

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

#####
# Postpaid Service
#####
# Make the following line a comment if the postpaid service is not
# to be deployed and configured
pp.include

# ----- Java Settings ----- #
pp.java.mem=-ms64m -mx128m

#####
# Notify Service
#####
# Make the following line a comment if the notify service is not to
# be deployed and configured
ns.include

# ----- Java Settings ----- #
ns.java.mem=-ms64m -mx128m

#####
# ConfirmListener Service
#####
# Make the following line a comment if the confirmlistener service
# is not to be deployed and configured
cl.include

# ----- Java Settings ----- #
cl.java.mem=-ms64m -mx128m

#####
# PushListener Service
#####
# Make the following line a comment if the pushlistener service is
# not to be deployed and configured
pl.include

# ----- Java Settings ----- #
pl.java.mem=-ms64m -mx128m

#####
# Monitoring Service

```

```
#=====
# Make the following line a comment if the monitoring service is not
# to be deployed and configured
mo.include

# ----- Java Settings ----- #
mo.java.mem=-ms64m -mx128m
```


调节性能

本附录介绍了一些调节建议，以帮助您从 Content Delivery Server 系统获得最佳性能。

本节包括以下主题：

- [调节配置变量](#)
- [调节 Oracle 数据库](#)
- [调节应用程序服务器](#)

B.1 调节配置变量

对于如何对 Content Delivery Server 所使用的配置变量进行设置以提高性能，本部分提供了一些指导性建议。

B.1.1 设置数据库连接池大小

如果使用 WebLogic Server 或 Sun Java System Application Server，则增加 Content Delivery Server 所使用的数据库连接数将允许提高订户的并发操作能力。

1. **编辑** `$CDS_HOME/deployment/deployment-name/conf/VAdminConsole.properties`。
2. **将** `db.pool.size.max` **property 设置为 30。**
例如，`db.pool.size.max=30`。
3. **编辑** `$CDS_HOME/deployment/deployment-name/conf/FulfillmentService.properties`。
4. **将** `db.pool.size.max` **property 设置为 30。**
例如，`db.pool.size.max=30`。

更改将在 Content Delivery Server 重新启动后生效。

B.1.2 设置 Java 内存设置

如果已使用 WebLogic Server 实施了基本部署，则 Content Delivery Server 在将主机内存的 50% 分配给 Java 虚拟机时工作良好。如果主机正在运行数据库或其他资源密集的进程，请减小此值。

1. **编辑** `$CDS_HOME/deployment/deployment-name/bin/start_server.sh` 文件。
2. **修改** `export JAVA_MEM="ms1024m -mx1024m"` 语句并将每个值都设置为系统内存大小的一半。

更改将在 Content Delivery Server 重新启动后生效。

注—还应在要用于部署 Content Delivery Server 的部署配置文件中更改每个组件的 `java.mem` 和 `prefix.java.mem` 属性。更新配置文件可以确保将在以后的部署中使用所需的值。

B.2 调节 Oracle 数据库

有关调节数据库性能的提示，请访问 Oracle Web 站点 (<http://technet.oracle.com>)。

B.3 调节应用程序服务器

还可以通过调节应用程序服务器来改善性能。本节包括以下主题：

- [调节 Sun Java System Application Server](#)
- [调节 WebLogic Server](#)

B.3.1 调节 Sun Java System Application Server

有关调节 Sun Java System Application Server 的提示，请访问 Sun Java System Application Server Web 站点 (<http://www.sun.com/software/sunone/>)。

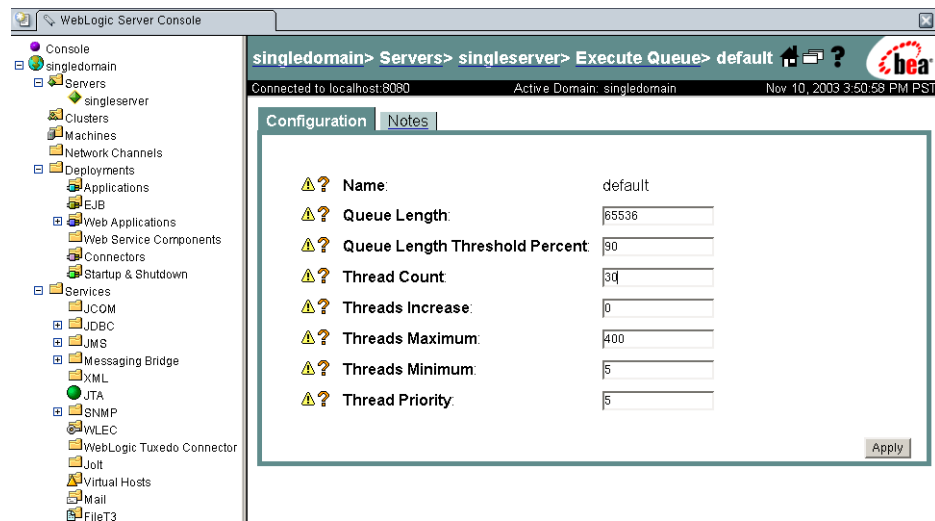
B.3.2 调节 WebLogic Server

本节介绍了一些参数，可设置为调节 WebLogic Server 的参数。有关调节 WebLogic Server 的其他提示，请访问 WebLogic Server Web 站点 (<http://www.weblogic.com/>)。

▼ 设置线程数

1. 登录到 WebLogic Server 控制台。
2. 在左侧窗格的目录树中，导航到 `cds-domain -> Servers -> deployment-name`。
3. 在右侧窗格中选择“监视”选项卡。
4. 单击“监视所有活动队列”。
5. 单击“配置执行队列”。
6. 单击“缺省”。

您应看到以下页面。



7. 将“线程数”设置为 30。
8. 单击“应用”。

更改将在 Content Delivery Server 重新启动后生效。

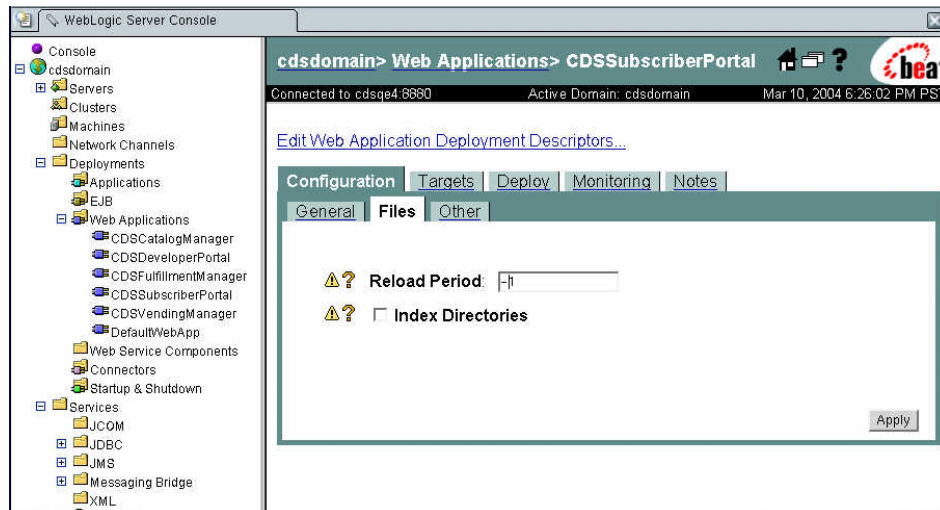
▼ 禁用动态 Servlet 类重新装入

1. 登录到 WebLogic Server 控制台。
2. 在左侧窗格的目录树中，导航到 `cds-domain -> Deployments -> Web Applications`。

3. 对于 Web 应用程序下列出的每个项目：

a. 选择“文件”选项卡。

您应看到以下页面。



b. 将“重新装入周期”设置为 -1。

c. 单击“应用”。

更改将在 Content Delivery Server 重新启动后生效。

▼ 使用 JavaServer Pages™ 技术禁用所创建网页的动态重新装入

1. 转到 `$CDS_HOME/deployment/deployment-name/weblogic/domains/domain-name/applications` 目录。
2. 编辑以下每个文件，并按代码示例 6 中所示将 `param-name element` `pageCheckSeconds` 设置为 -1。
 - `adminconsole/weblogic.xml`
 - `developerportal/weblogic.xml`
 - `subscriberconsole/weblogic.xml`
 - `vsadmin/weblogic.xml`

代码示例 6 更新为 weblogic.xml

```
<jsp-descriptor>
  <jsp-param>
    <param-name>pageCheckSeconds</param-name>
    <param-value>-1</param-value>
  </jsp-param>
  <jsp-param>
    <param-name>keepgenerated</param-name>
    <param-value>>false</param-value>
  </jsp-param>
</jsp-descriptor>
```

更改将在 Content Delivery Server 重新启动后生效。

系统维护

安装 Content Delivery Server 并将其投入生产后，需要确保它的执行情况符合要求。还需要能够分析所显示的信息并对问题做出响应。本节提供帮助您维护 Content Delivery Server 的信息。

C.1 系统监视

Content Delivery Server 提供了用于监视 Content Delivery Server 状态的应用程序监视代理。此代理支持简单网络管理协议 (SNMP) 并可以与现有网络监视系统集成。

Content Delivery Server 组件还在日志文件中记录信息，您可以使用这些信息查看系统状态。

C.1.1 应用程序监视代理

应用程序监视代理提供有关 Content Delivery Server 的状态信息。通过此代理，现有网络监视系统可以监视 Content Delivery Server 组件的可用性。您还可以监视 Content Delivery Server 与外部系统（例如记帐系统或订户数据库）之间的连接。

要激活该代理，请启动监视服务。请参见第 80 页上的第 5.1.2 节“启动 Content Delivery Server 服务”。

应用程序监视代理提供用来说明 Content Delivery Server 所发出的状态和警报的管理信息库 (MIB)。此信息包含在 `$CDS_HOME/deployment/deployment-name/conf` 目录的 `CDS36MibModule.mib` 文件中。将您的网络监视系统配置为使用此 MIB 规范。有关装入 MIB 的信息，请参见所使用的网络监视系统的产品文档。

表 27 显示了通过 MIB 提供的状态信息。网络监视系统可以查询应用程序监视代理的任何状态。状态 1 表示该组件正在运行。状态 0 表示该组件已停止工作。

表 27 状态通知

状态	说明
cdsServerStatus	Content Delivery Server 核心服务器的状态
eventServiceStatus	Content Delivery Server 事件服务的状态
messagingServiceStatus	Content Delivery Server 消息传送服务的状态
postPaidServiceStatus	Content Delivery Server 后付费服务的状态

表 28 显示检测到问题时 Content Delivery Server 发出的警报。通常情况下，警报表示特定的组件不可用。应重新启动所指示的组件，以确保 Content Delivery Server 继续正常运行。

表 28 Content Delivery Server 生成的警报

警报	说明
dbConnectionDown	数据库的连接已失败。
smscConnectionDown	SMSC 的连接已失败。
ppgConnectionDown	PPG 的连接已失败。
subscriberDirectoryConnectionDown	外部订户数据库的连接已失败。
downloadFailure	尝试为订户下载内容时出现下载故障。
httpPushConnectionDown	基于 HTTP 的 SMSC 的连接已停止。
missingExternalContent	无法再从远程位置获得外部托管的内容。

C.1.2 日志文件

执行 Content Delivery Server 时，组件将信息记录到以下日志文件（这些文件位于 `$CDS_HOME/deployment/deployment-name/logs` 目录中）：

- `EventService.log`：此日志中的条目是在下载内容和启动记帐功能时由事件服务生成的。
- `MessagingService.log`：此日志中的条目是在将内容推送到订户设备时由消息传送服务生成的。
- `NotifyService.log`：此日志中的条目是在处理通知并将其发送给订户时由通知服务生成的。
- `PostpaidService.log`：此日志中的条目是在处理后付费记帐时由后付费服务生成的。
- `PushListenerService.log`：此日志中的条目是在从订户设备收到消息时由推送监听器服务生成的。

- `ConfirmListenerService.log`: 此日志中的条目是在从订户设备收到确认消息时由确认监听器服务生成的。
- `SNMPService.log`: 此日志中的条目是在监视服务运行时由它生成的。
- `CDS.log`: 此日志中的条目是在用户访问管理控制台、`Subscriber Portal` 和 `Developer Portal` 时由 `Content Delivery Server` 的其余组件生成的。

请检查这些日志是否存在异常、标记为 `ERROR` 的行或其他问题指示。第 113 页上的第 C.4 节“错误消息”提供您可能收到的消息说明及您可以采取的应对措施建议。

当 `Content Delivery Server` 处于运行状态时，日志文件中将写入条目。缺省情况下，在 `Content Delivery Server` 停止之前，条目会不断写入每个日志文件。每个日志都将在重新启动 `Content Delivery Server` 后随之重新启动。如果要保留这些日志的历史记录，应在重新启动 `Content Delivery Server` 之前创建副本。

`Content Delivery Server` 使用日志软件包 `log4j`。可以根据 `log4j` 的功能，通过设置 `$CDS_HOME/deployment/deployment-name/conf` 目录下的 `logging.properties` 文件中的属性来管理日志文件。有关 `log4j` 的信息，请访问 Web 站点 <http://logging.apache.org/log4j/docs/manual.html>。

要控制写入日志的信息量，需要将 `logging.properties` 文件中的 `log4j.rootCategory` 属性设置成下列值之一：

- **DEBUG**: 调试消息和所有其他类型的消息都写入到日志。
- **INFO**: 除调试消息以外的所有类型的消息都将写入到日志。
- **WARN**: 只有警告、错误和致命的消息才将写入到日志。
- **ERROR**: 只有错误和致命的消息才写入到日志。
- **FATAL**: 只有致命的消息才写入到日志。

C.2 与外部系统的交互

`Content Delivery Server` 旨在与外部记帐系统及订户数据库进行交互。`Content Delivery Server` 还可以与自己的 `Oracle` 数据库进行交互。本节介绍组件之间的工作流。

有关将 `Content Delivery Server` 与外部系统集成的信息，请参见《*Sun Java System Content Delivery Server 集成指南*》。

C.2.1 记帐系统工作流

`Content Delivery Server` 支持预付费和后付费记帐模型。对于预付费记帐模式，在允许订户购买和下载内容之前需与记帐系统联系以获取授权。对于后付费记帐模式，事务发生后记帐系统将收到购买信息。

对于预付费记帐实现，Content Delivery Server 会在以下每个事务发生之前与记帐系统联系。记帐系统可以确定是允许还是拒绝该事务。对于后付费记帐实现，以下每个事务发生之后，Content Delivery Server 会生成一个 CDR，其中包含记帐系统根据自身的日程安排处理的信息。

- 订户查看详细资料。对于预付费记帐，记帐系统包含用来返回选定内容的定制价格的选项。
- 订户购买内容。
- 订户下载内容。
- 订户购买订阅。
- 订户取消订阅。
- 将退款发放给订户。
- 设备确认下载成功。
- 设备报告错误。

C.2.2 订户数据库 workflow

如果将 Content Delivery Server 配置为使用外部订户数据库，则在 Vending Manager 中为每个访问 Subscriber Portal 的订户保留的信息将仅限于订户登录 ID、移动电话号码、设备信息和订户计划。Vending Manager 还保留有关订户的下载历史的信息。要获取订户名和地址等配置信息，Content Delivery Server 将联系外部订户数据库。

如果外部数据库允许 Content Delivery Server 更改订户记录，则 Vending Manager 管理员可以通过 Vending Manager 管理控制台的帐户管理功能来编辑这些配置信息。如果外部数据库不允许 Content Delivery Server 进行更改，则 Vending Manager 管理员将只能查看订户的配置信息。

仅在订户访问 Subscriber Portal 之后才在 Content Delivery Server 中反映订户配置文件。此时，将在 Vending Manager 中创建订户帐户。Vending Manager 管理员可以在不与订户数据库交互的情况下查看所有订户帐户的列表。当管理员打开订户帐户时，Content Delivery Server 将联系订户数据库并检索配置信息。

Content Delivery Server 还将联系外部订户数据库以执行下列任务：

- 订户登录到 Subscriber Portal 时验证授权。
- 订户更改密码时更新帐户。
- 向忘记密码的订户发送提醒。
- 注册新订户。

C.2.3 Content Delivery Server 数据库 workflow

Content Delivery Server 使用 JDBC Type 4 驱动程序与 Oracle 数据库进行通信。除了错误日志与系统配置属性以外，所有 Content Delivery Server 数据都保留在该数据库中。

每次需要通过 Subscriber Portal 或 Developer Portal 向用户显示信息，或者需要通过 Catalog Manager 或 Vending Manager 管理控制台向管理员显示信息时，都将访问该数据库。要最大程度地减少对该数据库的访问，可以指定通过设置 `$CDS_HOME/deployment/deployment-name/conf/cache.config` 文件中的 `cache.enabled` 属性来缓存数据库信息。

Content Delivery Server 具有内置的容错功能，因此可以在数据库出现故障时继续运行。系统发出每个失败事务的错误消息并进行记录，同时 Content Delivery Server 将继续执行操作。

C.3 数据库维护

Content Delivery Server 数据库包含由 Content Delivery Server 执行的所有事务的记录。随着时间的推移，包含这些事务的表会变得很大，性能会降低。还应该定期清理数据库中的一些表，以删除不活动的项目。

本节标识了 Content Delivery Server 数据库中应该根据需要监视和删除其中某些项目的表。此外，还应该按照 Oracle 的建议对该数据库执行标准的维护步骤。

C.3.1 事务表

下表会随着 Content Delivery Server 中发生的事务数量的增加而增大。您可以对这些表中的某些数据进行归档，以使这些表的大小维持在易于管理的水平。

SUBSCRIBER_PURCHASE (售卖模式)

此表包含订户从特定的 Vending Manager 中购买的所有订阅。此表会相对于订户数量和他们的尝试购买的订阅数目的增加而相应增大。为确保使用此内容的订户许可证的完整性，应对该表进行维护。

当订户或他们已购买的内容不再处于活动状态时，可以从该表中清除相应的数据。

FULFILLMENT_REQUEST (实现模式)

此表包含代表任何 Vending Manager 发出的所有下载请求，并且需要对订户尝试的任何后续下载进行授权。由于一种实现模式为所有 Vending Manager 提供服务，因此该表将随着从任何 Vending Manager 购买的订阅数量的增加而增大。

当订户或他们已购买的内容不再处于活动状态时，可以从该表中清除相应的数据。

CDS_EVENT (事件模式)

此表包含 Content Delivery Server 已发出的所有事件。由于有很多事件类型，因此该表会迅速增大。此表中的事件用于报告历史记录。

当不再需要报告历史记录时，可以从该表中清除数据。

CAMPAIGN_PROCESSING (售卖模式)

该表包含向每个订户发送的所有活动消息。该表将按已执行的活动数量和已收到通知的订户数量的增加成比例增大。只要活动处于活动状态且订户可以从该活动中购买内容，就需要用到这些数据。

活动到期后，可以从该表中清除数据。

SUBSCRIBER_GIFT (售卖模式)

此表存储已由订户指定的礼品。

订户被停用后，可以从该表中清除数据。

C.3.2 实体表

下表保留了数据实体，例如订户和可下载内容。当实体处于非活动状态时，可以从这些表及其相关的表中删除不使用的记录。

SUBSCRIBER (售卖模式)

此表包含已使用 Content Delivery Server 的订户的所有订户帐户。如果 Content Delivery Server 与外部订户数据库集成，则对应信息子集将保留在该表中以便其他表（如 SUBSCRIBER_PURCHASE）引用。当订户处于非活动状态时，可以从以下各表中删除相关数据：

- CDS_SUBSCRIBER：名称、地址和其他配置信息。
- SUBSCRIBER_PLAN_MAP：为订户指定的订户计划。
- SUBSCRIBER_PURCHASE：订户已购买的所有内容。
- SUBSCRIBER_WISHLIST：订户在愿望列表中放置的项目。
- SUBSCRIBER_PREFERENCE：订户设置的显示首选项。

RESOURCE_CLASS (目录模式)

此表包含使用 Content Delivery Server 提交或注册的所有可下载内容。当项目处于非活动状态时，可以从以下各表中删除相关数据：

- RESOURCE_CLASS_LOCALE：相应语言环境的名称和说明。
- RESOURCE_CONTENT_DATA：内容的二进制文件。
- RESOURCE_INSTANCE：与设备有关的各版。
- RESOURCE_SCREENSHOT：内容的屏幕快照。
- INSTANCE_CAPABILITY_MAP：内容的功能要求。
- MODEL_CLASS_INSTANCE_MAP：内容到设备的映射。
- CAT_PRICING_MODEL_MAP：内容的价格模型。
- CATEGORY_RESOURCE_MAP：内容所在的类别。

CATEGORY_ITEM (售卖模式)

此表包含 Vending Manager 的内容。如果内容在目录中不可用，则可以为每个 Vending Manager 删除该内容，并从以下各表中删除相关数据：

- CATEGORY_ITEM_MAP：内容所在的类别。
- EXTERNAL_CONTENT_MAP：缓存的内容的文件位置。
- VEN_PRICING_MODEL_MAP：内容所使用的价格模型。
- HANDSET_LOCALE_EDITION_MAP：内容版所支持的语言环境。

C.4 错误消息

操作 Content Delivery Server 期间，日志文件中将写入消息。消息还将通过门户和管理控制台显示给用户。本节介绍 Content Delivery Server 发出的消息并给出应对措施建议。

有关日志文件的信息，请参见第 108 页上的第 C.1.2 节“日志文件”。

C.4.1 Content Delivery Server 日志

写入 Content Delivery Server 日志 (CDS.log) 的消息包括：

```
[DBPoolManager] ERROR Exception initializing connection
pool:com.sun.content.server.jdbc.ConnectionException:Error
getting jdbc connection in ConnectionManager.
```

说明：启动数据库时出现连接故障。

操作：确保 Oracle 正在运行且可以通过 Content Delivery Server 进行访问。确保数据库配置文件包含正确值。有关该文件的信息，请参见第 18 页上的第 3.1.2 节“第 2 步：设置基本部署的数据库”。

```
[DBPoolManager] WARN Removing failed
connection:oracle.jdbc.driver.OracleConnection@connectionId
java.sql.SQLException:Io exception:Connection reset by
peer:socket write error
```

说明：出现数据库连接故障。*connectionId* 将标识已失败的连接。

操作：确保 Oracle 正在运行且可以通过 Content Delivery Server 进行访问。确保数据库配置文件包含正确值。有关该文件的信息，请参见第 18 页上的第 3.1.2 节“第 2 步：设置基本部署的数据库”。

```
[EventService_JMS] ERROR Error when publishing event via JMS
Pool
```

说明：JMS 连接故障（事件）

操作：如果正在运行 Sun Java System Application Server，确保消息传送代理也正在运行。确保部署配置文件 (.cfg) 包含正确的 JMS 属性值。

```
[LDAPUserManager:ExecuteThread:'6' for queue:'default'] FATAL
Cannot create LDAP context.
java.net.UnknownHostException:loadtesting at
java.net.InetAddress.getAllByName0(InetAddress.java:566)
```

说明：Content Delivery Server 无法连接到外部 LDAP 用户系统。

操作：确保 LDAP 系统正在运行且可以访问，然后重试操作。

```
[Stock] ERROR failed to load url:http://vendingManager/
vsadmin/main/admin_autostock?...:vendingManager
java.net.UnknownHostException:vendingManager
```

说明：无法访问 Vending Manager *vendingManager*。

操作：验证为 Vending Manager 指定的 IP 地址是否正确以及是否可以从 Catalog Manager 进行访问。确保 Vending Manager 正在对网络请求做出响应。

```
[Task] ERROR permission denied:permissionError
```

说明：用户无权访问该页面。*permissionError* 将提供附加信息。

操作：重新登录并重试操作。

```
com.sun.content.server.server.web.util.WebMsgException:Could
not get UserManager.Could not load the usermanager class
```

说明：订户尝试登录到 Subscriber Portal，但 Content Delivery Server 无法连接至外部 LDAP 用户系统。

操作：确保 LDAP 系统正在运行且可以访问，然后重试操作。

C.4.2 消息传送服务日志

写入消息传送服务日志 (MessagingService.log) 的消息包括：

```
[MsgServer] ERROR Failed to initialize JMS Connection
java.net.ConnectException:t3://hostname:port:Destination
unreachable; nested exception
is:java.net.ConnectException:Connection refused:connect; No
available router to destination
[MsgServer] INFO Will Retry to initialize JMS Connection in
60 seconds
```

说明：Content Delivery Server 未正确启动或者消息传送服务正在运行时有人关闭了 Content Delivery Server。

操作：确保正确启动 Content Delivery Server。同时确保部署配置文件 (.cfg) 包含正确的 JMS 属性值。

```
[MsgServer] INFO JMS Connection Restored
```

说明：指示已成功恢复 JMS 连接的说明性消息。

操作：无需采取操作。

```
[MsgServer ] ERROR Exception caught
com.sun.content.server.server.msgserver.protocolPpgConnectionFa
iledException:Failed to connect to wap ppg server
```

说明：MsgService.properties 文件中的 WAP PPG 属性可能不正确，或者 PPG 已关闭。

操作：确保 MsgService.properties 文件中正确设置了 WAP PPG 属性，而后重新启动消息传送服务。同时确保 WAP PPG 处于打开状态且正在运行。

```
[MsgServer] ERROR Error while sending SMS Push message
java.net.ConnectException:Connection refused:connect
```

说明：MsgService.properties 文件中的推送发送器 SMSC 属性可能不正确或者 SMSC 已关闭。如果正在使用基于 HTTP 的 SMSC 服务发送 SMS，则 SubscriberConsole.properties 文件中的 SMSC 属性可能不正确。

操作：确保 MsgService.properties 文件中正确设置了 SMSC 属性。如果正在使用基于 HTTP 的 SMSC 服务来发送 SMS，确保

SubscriberConsole.properties 文件中正确设置了 SMSC 属性，而后重新启动消息传送服务。同时确保 SMSC 处于打开状态且正在运行。

C.4.3 推送监听器服务日志

写入推送监听器服务日志 (PushListenerService.log) 的消息包括:

```
[MsgServer] ERROR Connection could not be established
java.net.UnknownHostException
[MsgServer] ERROR FAILED TO CONNECT TO SMSC
[MsgServer] INFO Will try to reconnect after 60000
milliseconds
```

说明: PushListener.properties 文件中的推送监听器 SMSC 属性可能不正确或者 SMSC 已关闭。

操作: 确保 PushListener.properties 文件中正确设置了推送监听器 SMSC 属性, 而后重新启动推送监听器服务。同时确保 SMSC 处于打开状态且正在运行。

```
[MsgServer] ERROR CONNECTION BROKEN
```

说明: SMSC 的连接已中断或 SMSC 已关闭。

操作: 确保 SMSC 正在运行且可以访问。

```
[MsgServer] ERROR Invalid SMSC host name specified in the
properties file
```

说明: pushListener.properties 文件中没有正确指定 SMSC 主机名。

操作: 确保 PushListener.properties 文件中正确设置了 SMSC 主机名, 而后重新启动 PushListener 服务。

```
[MsgServer] ERROR Invalid SMSC port number specified in the
properties file
```

说明: pushListener.properties 文件中没有正确指定 SMSC 端口号。

操作: 确保 PushListener.properties 文件中正确设置了 SMSC 端口号, 而后重新启动 PushListener 服务。

C.4.4 确认监听器服务日志

写入确认监听器服务日志 (ConfirmListenerService.log) 的消息包括:

```
[ConfirmListener] ERROR Could not listen on port port-number
```

说明: ConfirmListener.properties 中的端口号可能不正确, 或该端口号正忙。

操作: 确保将确认监听器服务设置为通过正确的端口号进行监听, 且 ConfirmListener.properties 中指定的端口号可用。重新启动确认监听器服务。

C.4.5 通知服务日志

写入通知服务日志 (NotifyService.log) 的消息包括：

`Couldn't process subscriber login-name`

说明：处理订户登录名 *login-name* 时失败。可能的原因包括无法连接外部数据库或无法从中收到授权，或者订户的设备无法运行活动中包含的内容。未向此订户发送通知。

操作：如果数据库存在问题，请进行更正并重新提交活动。将会向满足条件的所有订户重新发送通知。

`SMS is too long. Truncating to 140 characters`

说明：指定的消息太长并且已由系统截断。订户将收到截断的消息。

操作：无需采取操作。

`Failed executing campaign for subscriber ID subscriber-ID`

说明：处理订户 ID *subscriber-ID* 时失败。可能的原因包括电话号码或电子邮件地址不正确，或存在网络问题。未向此订户发送通知。

操作：如果网络存在问题，请进行更正并重新提交活动。将会向满足条件的所有订户重新发送通知。

报告

Vending Manager 提供了每日统计报告，通过该报告可以查看和跟踪应用程序下载信息和使用情况统计。Content Delivery Server 还可以在数据库表中存储事务数据。可以使用此数据创建自己的报告。

每个购买和退款事务的记录都存储在数据库 *prefix_report*（其中 *prefix* 是安装报告数据库模式时输入的值）中的 REPORT_DOWNLOAD 表中。有关安装模式的信息，请参见第 70 页上的第 4.12 节“配置对定制报告的支持”。下表说明了为每个事务存储的数据。

注一可以在以后的版本中更改该表。

表 29 报告数据

名称	格式	说明
REPORT_DOWNLOAD_ID	数字	事务的唯一标识符。
TRANSACTION_DATE	日期	事务发生时的时间戳。
RESOURCE_INSTANCE_ID	数字	事务涉及的内容项目的版标识符。
RESOURCE_CLASS_ID	数字	事务涉及的内容项目的标识符。
CONTENT_NAME	字符串	事务涉及的内容的名称。
CONTENT_DESCRIPTION	字符串	内容的说明。
CONTENT_TYPE	字符串	内容的类型，如 <i>midlet</i> 。
CONTENT_PROVIDER_KEY	字符串	内容提供者赋予内容的标识符。
CONTENT_PROVIDER_ID	数字	内容提供者的标识符。
CONTENT_PROVIDER	字符串	内容提供者的名称。
CONTENT_COST	浮点	由内容提供者指定的价格。
PRICE	浮点	订户接受的价格。
FULFILLMENT_REQUEST_ID	数字	实现请求的标识符。

表 29 报告数据 (续)

名称	格式	说明
FULFILLMENT_STATUS	字符串	事务的状态, “已购买” 或 “已退款”。
BILLING_MODEL	字符串	内容的记帐模型。
SUBSCRIBER_ID	数字	订户的标识符。
MSISDN	字符串	与订户相关联的 MSISDN。
FIRST_NAME	字符串	订户的名字。
LAST_NAME	字符串	订户的姓氏。
MODEL	字符串	订户设备的名称。

为每个事务存储的信息都是事务发生时的实时信息, 并且不会被更改。例如, 如果订户更改所使用的设备, 则只有该订户的新事务才显示新设备。使用旧设备发生的事务将继续在 MODEL 字段中显示旧设备。

索引

A

admin.sh, 25
AdminConsole.properties, 53, 54
API 过滤, 56
Application Server
 要求, 12
安装目录
 创建, 14
 删除, 85

B

Bash Shell, 12
BEA_HOME, 18
broker.sh, 24, 80, 82
报告, 定制, 70
本地化, Subscriber Portal, 63 to ??
本指南的读者, xiii
本指南的适用范围, xiii
本指南的组织结构, xiv
编码, 50
部署
 高级部署, 35
 基本配置, 22
 Vending Manager 部署, 32
卸载
 Sun Java System Application Server, 84
 WebLogic Server, 83
部署配置文件
 缺省, 22, 31, 35
 设置
 高级部署, 34
 基本部署, 21
 Vending Manager 部署, 30

属性

常规, 87
服务, 91
管理器, 90

样例

对于 Sun Java System Application Server, 95
对于 WebLogic Server, 92
sn_sample.cfg, 95
wl_sample.cfg, 92

C

Catalog Manager

管理, 3, 54
管理控制台, 3
启动, 80
属性, 53, 90
说明, 3
停止, 81
URI, 88

CDS.properties, 69, 75

CDS_HOME

删除, 85
设置, 15

CDS36MibModule.mib, 107

cdse, 请参见 Content Delivery Server Execute 命令

cdsi, 请参见 Content Delivery Server Install 命令

CDSSnmp.properties, 42

CFG 文件, 请参见部署配置文件

cHTML, 13

CIMD2, 39

CommonConsole.properties, 47

ConfirmListener.properties, 44

Content Delivery Server

概述, 1
解压缩 Content Delivery Server 文件, 14

- 启动
 - 服务, 80
 - 管理器, 80
- 日志, 109, 113
- 设置环境变量, 15
- 数据库, workflow, 111
- 体系结构, 1
- 停止
 - 服务, 82
 - 管理器, 81
- 卸载, 85
- 组件, 3
- Content Delivery Server Execute 命令
 - 启动服务, 81
 - 启动服务器, 80
 - 停止服务, 82
 - 停止服务器, 81
 - 衍生服务, 81
- Content Delivery Server Install 命令
 - 部署, 22
 - 初始化数据库, 20, 30, 33
 - 创建数据库模式, 21, 30
 - 填充数据库, 21, 30
- crontab 文件, 26
- 程序校验
 - 从 workflow 中删除, 60
 - DRM 代理, 57
 - 设置, 57

D

- DBConf.xml, 19
- DEFAULT_CONFIG, 22, 31, 35
- DEFAULT_DB, 20, 30
- DEFAULT_DEPL, 22
- deploy.sh, 24
- deploymq.sh, 23
- Developer Portal
 - 浏览器要求, 13
 - 属性, 48
 - 说明, 3
 - URI, 88
 - URL, 69
- DeveloperPortal.properties, 48, 54
- DoJa 库
 - 提交, 71
 - 要求, 13
- DRM 代理, 57
- 到期时间, 共享内容, 52

- 登录尝试次数, 51
- 订户首选项, 78
- 订户数据库, workflow, 110
- 订户通知, opt-in, 78
- 订户帐户管理, 67
- 定制报告, 70
- 动态 Servlet 类重新装入, 103
- 动态标题, 50
- 端口号
 - JMX, 91
 - JNDI, 89
 - Message Queue 代理, 23
 - SSL, 89
 - WebLogic Server 管理控制台, 89
 - 应用程序服务器, 88
- 端口号, 确认监听器设备, 44
- 短消息服务, 37
- 短消息服务中心 (SMSC), 52

E

- EventService.properties, 75

F

Fulfillment Manager

- 启动, 80
- 属性, 90
- 说明, 4
- 停止, 81
- URI, 88
- 发行说明, 14
- 防火墙, 配置, 54
- 服务器布局, 8
- 服务器名称, 88
- 负载均衡器 cookie, 50

G

概述

- 体系结构, 1
- 组件, 3
- 高级部署
 - 设置配置文件, 34
 - 设置数据库, 33
- 高级配置
 - 部署, 35
- 共享内容到期时间, 52
- 共享内容模板, 52

workflows
Content Delivery Server 数据库, 111
订户数据库, 110
记帐系统, 109
请参见提交检验器 workflow
购买, 快速, 52
管理
Catalog Manager, 3
Vending Manager, 4
惯例, 印刷, xv
管理控制台
Catalog Manager, 3
浏览器要求, 13
Vending Manager, 4
管理信息库 (MIB), 107
关于本指南, xiii

H

后付费服务
配置, 41
启动, 81
日志, 108
属性, 91
说明, 5
停止, 82
环境变量
Content Delivery Server, 15
in init_env 文件, 18
货币符号, 65

I

iAppli 应用程序, 20, 60, 71
init_env shell 脚本, 18

J

Java 开发工具包, 12
Java 内存设置, 102
Java 起始目录, 18
基本部署
设置配置文件, 21
设置数据库, 18
基本配置
部署, 22
计划注意事项, 8
记帐记录, 42
记帐系统, workflow, 109

价格模型, 55
简单网络管理协议, 6
监视, 系统, 107
监视服务
配置, 42
启动, 81
日志, 109
属性, 91
说明, 6
停止, 82
监视脚本, WebLogic Server, 26
解压缩, 13, 14
警报, 43, 108
绝对 URL 生成, 54

K

开发者通知, 75
可缩放性, 9
快速购买, 52
快速浏览, 52

L

log4j, 109
logging.properties, 109
类别深度, 47
连接池大小, 101
浏览, 快速, 52
浏览器, 13

M

Message Queue 代理
启动, 80
设置, 23
停止, 82
MIDP 库, 56
MsgService.properties, 38, 39, 115
每页的项目数
Catalog Manager, 53
Developer Portal, 48
Subscriber Portal, 50
Vending Manager, 53
密码
Message Queue 代理, 23
WebLogic Server 管理, 25
密码提醒, 51
模板, 共享内容, 52

N

内存

- 调节, 102
- 设置, 88, 91
- 要求, 12

P

PostpaidService.properties, 41, 70

PushListener.properties, 40, 116

配置文件

部署

- 常规属性, 87
- 服务属性, 91
- 管理器属性, 90
- 缺省, 22, 31, 35
- Sun Java System Application Server 的样例, 95
- 设置 Vending Manager 部署, 30
- 设置高级部署, 34
- 设置基本部署, 21
- WebLogic Server 的样例, 92

数据库

- 缺省, 20, 30
- 设置基本部署, 19
- 样例, 19

样例

- 位置, 21

Q

启用的缓存, 111

签署, MIDlet, 60

确认购买, 49

确认监听器服务, 6

- 配置, 44
- 启动, 81
- 日志, 109, 116
- 属性, 91

缺省语言, 63

群集, 9

R

日志

Content Delivery Server, 109, 113

后付费服务, 108

监视服务, 109

确认监听器服务, 109, 116

事件服务, 108

通知服务, 108, 117

推送监听器服务, 108, 116

消息传送服务, 108, 115

日志文件, 108

容量计划, 9

软件要求, 12

S

SMPP, 38

SMS HTTP 适配器属性, 39

SMS 推送消息, 配置, 44

SMS, 请参见短消息服务

sms_push_msg_template.xml, 44

sn_sample.cfg, 95

SNMPService.log, 109

SSL, 25

Subscriber Portal

- 本地化, 63
- 浏览器要求, 13
- 属性, 48
- 说明, 4
- URI, 88

Subscriber Portal URL, 66

SubscriberPortal.properties, 41, 49

绝对 URL 生成, 54

Sun Java System Application Server

设置, 23

样例配置文件, 95

SUN_AS_HOME, 18

SUN_AS_JAVA_HOME, 18

设备库要求, 13

事件服务

- 启动, 81
- 日志, 108
- 属性, 91
- 说明, 5
- 停止, 82

受信证书, 60

数据库

- 调节, 102
- 设置
 - 高级部署, 33
 - 基本部署, 18
 - Vending Manager 部署, 29
- 要求, 13
- 用于, 4

数据库初始化, 20

数据库模式创建, 21, 30

- 数据库配置文件
 - 缺省, 20, 30
 - 设置
 - 基本部署, 19
 - 样例, 19
- 数据库填充, 21, 30
- 属性
 - Catalog Manager, 53, 90
 - 常规, 87
 - Developer Portal, 48
 - Fulfillment Manager, 90
 - 服务, 91
 - 管理器, 90
 - 后付费服务, 91
 - 监视服务, 43, 91
 - 配置, 47
 - 确认监听器服务, 91
 - Subscriber Portal, 48
 - 事件服务, 91
 - 通用, 47
 - 通知服务, 91
 - 推送监听器服务, 91
 - Vending Manager, 53, 90
 - 消息传送服务, 91
- 双向推送消息传送, 52

T

- 提交检验器 workflow
 - DRM 消息中打包的静态内容, 61
 - workflow 变量, 58
 - iAppli 应用程序, 60
 - Java 应用程序, 56
 - 缺省, 62
 - 提供的工作流, 56
 - 外部托管受版权保护的内容, 61
 - 要在 DRM 消息中打包的静态内容, 61
 - 指定标准, 62
- 体系结构, 1, 3
- 调节
 - 动态 Servlet 类重新装入, 103
 - Java 内存设置, 102
 - 连接池大小, 101
 - Oracle 数据库, 102
 - 配置变量, 101
 - Sun Java System Application Server, 102
 - WebLogic Server, 103
 - 网页的动态重新装入, 104
 - 线程数, 103

- 性能, 101
- 停止
 - 服务, 82
 - 管理器, 81
- 通知
 - 订户, opt-in, 78
 - 开发者, 75
- 通知服务
 - 启动, 81
 - 日志, 108, 117
 - 属性, 91
 - 说明, 5
 - 停止, 82
- 通知模板, 75
- 推送, 启用或禁用, 52
- 推送监听器服务
 - CIMD2 属性, 40
 - 配置, 40
 - 启动, 81
 - 日志, 108, 116
 - SMPP 属性, 40
 - 属性, 91
 - 说明, 5
 - 停止, 82
- 推送类型, 缺省类型, 52
- 推送适配器
 - CIMD2 属性, 39
 - SMPP 属性, 38
 - SMS HTTP 属性, 39
- 推送消息, 移动起始, 40

U

- ulimit, 80
- URL, Developer Portal, 69

V

- Vending Manager
 - 管理, 4
 - 管理控制台, 4
 - 启动, 80
 - 属性, 53, 90
 - 说明, 4
 - 停止, 81
 - URI, 88
 - 准备使用, 28
- Vending Manager 部署
 - 设置配置文件, 30

- 设置数据库, 29
- Vending Manager 配置
 - 部署, 32
- Vending Manager 帐户, 54
- vending.config, 63
- vs.account.name, 90
- VSAdminConsole.properties, 53, 54

W

- WAP 推送, 37
- WebLogic Server
 - 设置, 25
 - 样例部署配置文件, 92
- WL_HOME, 18
- WL_JAVA_HOME, 18
- wl_sample.cfg, 92
- wljms.sh, 26
- WML, 13
- WML Portal, 50
- 外部内容 ID, 41
- 外部托管的内容, 69
- 网页的动态重新装入, 104
- 维护, 系统, 107
- 文件, 最大大小, 47
- 文件的最大大小, 47
- 我的活动, 项目, 50
- 我的礼品, 项目, 51
- 我的下载, 项目, 51
- 我的愿望列表, 项目, 51

X

- XHTML, 13
- 系统监视, 107
- 系统维护, 107
- 线程数, 89, 103
- 陷阱, 43, 108
- 先决条件, 11
- 消息
 - Content Delivery Server 日志, 113
 - 确认监听器服务日志, 116
 - 通知服务日志, 117
 - 推送监听器服务日志, 116
 - 消息传送服务日志, 115
- 消息传送服务
 - 配置
 - SMS, 38
 - WAP 推送, 38

- 启动, 81
- 日志, 108, 115
- 属性, 91
- 说明, 5
- 停止, 82
- 卸载
 - 部署
 - Sun Java System Application Server, 84
 - WebLogic Server, 83
 - Content Delivery Server, 85
- 性能, 调节, 101

Y

- 样例
 - 部署配置文件
 - 对于 WebLogic Server, 92
 - Sun Java System Application Server, 95
 - SMS 推送消息模板, 44
 - subsubmgr.xml, 68
 - 数据库配置文件, 19
 - 提交验证 workflow 标准, 62
 - vsadminsubmgr.xml, 69
- 要求
 - Application Server, 12
 - Bash Shell, 12
 - Java 开发工具包, 12
 - 解压缩, 13
 - 浏览器, 13
 - 软件, 12
 - 数据库, 13
 - 硬件, 12
- 页面阈值
 - Developer Portal, 48
 - Vending Manager, 53
- 移动起始的推送消息, 40
- 隐藏或禁用注册链接, 50
- 印刷惯例, xv
- 硬件要求, 12
- 应用程序服务器
 - 端口号, 88
 - 服务器名称, 88
 - 设置
 - Sun Java System Application Server, 23
 - WebLogic Server, 25
 - 域名, 88
- 应用程序服务器域, 设置, 24
- 应用程序监视代理, 107 to 108
- 用户角色, 6

- 用户名
 - Message Queue 代理, 23
 - WebLogic Server 管理, 25
- 邮件服务, 配置, 66
- 邮件服务器, 91
- 域名, 88

Z

- 证书
 - 对于 WebLogic Server
 - WebLogic Server
 - 证书, 25
- 主目录
 - 删除, 85
 - 设置, 15
- 桌面内容链接, 项目, 50
- 自动储存, 6
- 自动创建订户, 49
- 自动创建设备, 49
- 自动登录用户, 49
- 自动置备, 49
- 组件, 3
- 最大的应用程序大小, 47

