



Oracle Application Server 10g  
J2EE Deployment and Administration

# Oracle Application Server 10g —— J2EE 部署和管理

Michael Wessler

Erin Mulder

Rob Harrop

Jan Machacek

狄东宁

著

译



清华大学出版社

# Oracle Application

## Server 10g

### J2EE 部署和管理

Michael Wessler  
Erin Mulder  
Rob Harrop  
Jan Machacek

(美) 著

狄东宁 译

清华大学出版社

北 京

EISBN: 1-59059-235-2

Oracle Application Server 10g: J2EE Deployment and Administration

Michael Wessler et al

Original English language edition published by Apress L. P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

Copyright ©2004 by Apress L.P. Simplified Chinese-Language edition copyright ©2004 by Tsinghua University Press.

All rights reserved.

本书中文简体字版由 Apress 出版公司授权清华大学出版社出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2005-4577

版权所有,翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

#### 图书在版编目(CIP)数据

Oracle Application Server 10g——J2EE 部署和管理/(美) 韦斯勒(Wessler, M.)等著; 狄东宁译。

—北京: 清华大学出版社, 2006.6

书名原文: Oracle Application Server 10g: J2EE Deployment and Administration

ISBN 7-302-12188-5

I.O… II.①韦…②狄… III.①关系数据库—数据库管理系统, Oracle ②JAVA 语言—程序设计

IV. ①TP311.138②TP312

中国版本图书馆 CIP 数据核字(2005)第 142517 号

出版者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社总机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 王 黎

封面设计: 康 博

版式设计: 康 博

印装者: 北京国马印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 28.75 字数: 736 千字

版 次: 2006 年 6 月第 1 版 2006 年 6 月第 1 次印刷

书 号: ISBN 7-302-12188-5/TP·7853

印 数: 1~4000

定 价: 58.00 元

# 作者简介

## Michael Wessler

Michael Wessler 获得印第安纳州西拉法叶城的大学计算机专业学士学位。他是 Oracle8 和 8i 的认证数据库管理员、Oracle9iAS 认证 Web 管理员和 Oracle 10g 数据库技术专家。他一直从事于 Windows NT、UNIX 和 Linux 等操作系统上包括分组的 Oracle 并行服务器(OPS)等工作环境的 Oracle 数据库管理工作。Michael 致力于多项 IT 工作, 范围小到 dot-com Internet 的启动, 大到政府机构和公司的这方面工作。目前正在美国国防部担任顾问的 Michael 也是 Perpetual 技术股份有限公司的技术经理。

除了担任 Oracle DBA 顾问之外, 他更多的是从事 Oracle9iAS 的 web 应用服务器管理员工作。目前, 他正在替美国国防部管理多个 web 应用, 并在多个政府机构和私营企业内担任顾问。他还经常就 Oracle9iAS 作专题讲座, 并讲授调整 Oracle 性能等内容。可以访问他们网站 <http://mike.wessler.name> 或通过他的 email: [mwessler@yahoo.com](mailto:mwessler@yahoo.com) 与他取得联系。

Michael 的其他著作: *Oracle DBA on Unix and Linux*(SAMS, 2002), 与人合著 *Oracle Unleashed, 2<sup>nd</sup> edition*(SAMS, 1997), *Unix Primer Plus, 3<sup>rd</sup> edition*(Waite Group Press, 2000), *COBOL Unleashed*(SAMS, 1998), *Unix Unleashed, 4<sup>th</sup> edition*(SAMS, 2002)。

Michael Wessler 撰写了本书第 9 章、第 15~21 章和附录。

## Erin Mulder

Erin Mulder 在 Java 刚诞生没多久就开始和它打交道。近些年来, 她帮助发布了多个成功的 Java 和 J2EE 应用。这些应用包括实时能源使用可视化、视频点播服务器的网络管理、金融市场分析、联邦政府成本核算和合同管理等。她将开放资源和商业应用软件捆绑在一起, 专门研究如何让它们更好的协调一致的工作。她也是多家刊物的技术撰稿人, 在 Java 用户组和 Java One 会议发表了许多专题文章。她还是 Chariot Solution 公司的软件架构师, 该公司是位于费城的一家 Java 技术咨询公司。

Erin Mulder 撰写了本书第 3、4 章, 并收集第 1~14 章的相关素材。

## Rob Harrop

Rob Harrop 是英国开发公司——Cake Solution 有限责任公司([www.cakesolutions.net](http://www.cakesolutions.net))的主要软件架构师。他在该公司领导了一个由 6 人组成的开发小组。该小组致力于为包括商贸部、都市警察和 NUS 服务有限责任公司在内的不同客户提供企业解决方案。Rob 和该公司专门致力于 .net 和基于 J2EE 的开发工作。他从测试阶段起就加入到 .net 的工作之中。

Rob Harrop 撰写过 *Pro Jakarta velocity: From professional to expert*, 并与其他人合作出版了 *Pro Jakarta Struts, 2<sup>nd</sup> edition*(Apress, 2004)、*Pro Visual Studio.NET*(Apress, 待出版)。

Rob 喜欢在有限的业余时间里研究不同的技术。当前他的研究兴趣点在 Groovy 和面向方面编程。Rob 是开源 Spring 工程([www.springframework.org](http://www.springframework.org))的成员。该工程是一个根据依赖性注射原则构建的 Java 和 .NET 的应用框架。在不和计算机打交道的时候, Rob 通常扎在书堆中, 喜欢模仿特里·普莱切特的音乐。

Rob Harrop 撰写了本书的第 6、7、8 章和第 13 章，并收集第 1、2、5 章的相关素材。

### **Jan Machacek**

Jan Machacek 是英国软件开发公司——Cake Solution 有限责任公司([www.cakesolutions.net](http://www.cakesolutions.net))的骨干程序员。他在此帮助不同的美国和英国客户设计和实现企业级应用。他空闲的时候热衷于钻研软件体系结构、非过程性和人工智能编程，以及摆弄计算机硬件。

作为一个彻头彻尾的计算机怪才，Jan Machacek 非常喜欢“星球大战”和“指环王”系列电影。他居住在英国的曼彻斯特，email: [jan@cakesolutions.net](mailto:jan@cakesolutions.net)。

Jan Machacek 撰写了本书第 12 章，并收集第 10、13、14 章的相关素材。

# 目 录

<b>第 1 章 Oracle 的特性和版本概述</b> ..... 1	<b>第 2 章 使用 Oracle 10g AS 开发</b>
1.1 选择版本..... 1	应用程序..... 13
1.2 正确选择发行版本..... 3	2.1 为开发人员选择合适的版本..... 13
1.2.1 OAS..... 4	2.1.1 考虑应用程序需求..... 13
1.2.2 iAS..... 4	2.1.2 考虑开发人员的经验..... 14
1.2.3 9iAS v1..... 4	2.1.3 设置开发环境..... 14
1.2.4 9iAS v2..... 4	2.2 选择测试使用的版本..... 15
1.2.5 10g AS..... 4	2.2.1 单元测试..... 15
1.3 J2EE 特性..... 5	2.2.2 集成测试..... 15
1.3.1 Servlet 和 JSP..... 5	2.2.3 性能测试..... 15
1.3.2 EJB(Enterprise JavaBeans)..... 6	2.2.4 环境分析..... 15
1.3.3 JMS(Java Message Service)..... 6	2.3 开发使用的策略..... 16
1.3.4 JDBC 和 J2EE Connector..... 6	2.4 测试策略..... 17
1.3.5 事务..... 6	2.5 部署策略..... 17
1.3.6 安全性..... 6	2.6 访问 J2EE 应用程序..... 18
1.3.7 其他 J2EE 服务..... 6	2.7 小结..... 18
1.4 非 J2EE 组件..... 7	<b>第 3 章 OC4J 的安装和运行行为</b> ..... 19
1.4.1 OHS(Oracle Http Server)..... 7	3.1 安装 Standalone..... 19
1.4.2 Web Cache..... 7	3.1.1 下载和安装 OC4J Standalone..... 20
1.4.3 ASC..... 8	3.1.2 测试安装..... 20
1.4.4 SSO..... 8	3.1.3 服务器目录结构..... 22
1.4.5 Content Management SDK..... 9	3.2 基本操作..... 23
1.4.6 集成..... 9	3.2.1 启动服务器..... 23
1.4.7 OID(Oracle Internet Directory)..... 9	3.2.2 终止服务器..... 24
1.4.8 JDeveloper..... 9	3.2.3 重新启动服务器..... 25
1.4.9 TopLink..... 10	3.2.4 脚本范例..... 25
1.4.10 Portal..... 10	3.2.5 作为 Windows 服务运行..... 26
1.4.11 Discoverer..... 10	3.2.6 作为 Unix 服务运行..... 26
1.4.12 Reports..... 11	3.3 运行时行为..... 26
1.4.13 Forms..... 11	3.3.1 服务器体系结构..... 26
1.4.14 Personalization..... 11	3.3.2 Classloading..... 28
1.4.15 Wireless..... 11	3.3.3 端口和协议..... 32
1.5 小结..... 11	3.3.4 性能和内存使用..... 32
	3.3.5 编译器..... 33

3.4	小结	33
<b>第 4 章</b>	<b>通用的服务器配置</b>	<b>34</b>
4.1	配置文件的使用	34
4.1.1	配置应用级资源	34
4.1.2	编写配置文件	34
4.1.3	管理多项配置	35
4.1.4	多个 OC4J 目录树	35
4.1.5	带有多配置目录的单一 OC4J 目录树	35
4.1.6	单个目录下的多版本配置文件	36
4.2	查找正确的配置文件	36
4.3	配置日志	38
4.3.1	配置日志文件名和路径	38
4.3.2	标准输出和错误流的重新定向	39
4.3.3	使用 OC4J 调试参数	39
4.4	配置文件位置	40
4.5	配置端口	40
4.5.1	HTTP	41
4.5.2	HTTPS	41
4.5.3	JMS	42
4.5.4	RMI	42
4.5.5	IIOP	42
4.6	配置编译器	43
4.6.1	进程中编译	43
4.6.2	使用 jikes 编译器编译	44
4.7	配置事务	44
4.7.1	事务 Timeout	44
4.7.2	事务恢复	44
4.8	配置其他 J2EE 服务	45
4.9	配置 HTTP 连接限制	45
4.10	设置部署行为	46
4.10.1	application-auto-deploy 目录	46
4.10.2	application 目录	46
4.10.3	deployment 目录	47
4.10.4	connector 目录	47
4.10.5	自启动应用	47
4.11	配置应用	47

4.12	小结	48
<b>第 5 章</b>	<b>配置 JNDI 和 JDBC 服务</b>	<b>49</b>
5.1	JNDI	49
5.1.1	理解 JNDI 位置	50
5.1.2	得到 JNDI 环境	50
5.2	通过 J2EE 应用程序访问数据库	50
5.2.1	非 J2EE 环境中简单的数据库访问	51
5.2.2	J2EE 环境中的数据库访问	51
5.3	JDBC 驱动程序	53
5.4	选择数据源类型	56
5.5	配置数据源	57
5.5.1	定制数据源属性	59
5.5.2	定制数据源特性	61
5.5.3	部署数据源	62
5.6	使用数据源	62
5.6.1	确定正确的 JNDI 位置	62
5.6.2	重载默认的用户名和密码	63
5.6.3	使用拓展的 Oracle JDBC	63
5.7	配置两阶段提交	63
5.8	小结	64
<b>第 6 章</b>	<b>配置 Java Message Service</b>	<b>65</b>
6.1	Oracle 10g AS 中的 JMS	65
6.2	配置和使用 Oracle 10g AS 的 JMS	66
6.2.1	标准配置	66
6.2.2	构建和配置应用程序	69
6.2.3	配置文件持久性	75
6.2.4	配置主机和端口	76
6.2.5	配置日志记录	76
6.3	配置 Oracle JMS	77
6.3.1	设置队列	77
6.3.2	配置 Oracle 10g AS	78
6.3.3	配置 Hello World 应用程序	78
6.4	配置第三方 JMS	79
6.5	选择供应商	81
6.6	小结	81

<b>第 7 章 安全性</b> .....	<b>82</b>
7.1 J2EE 安全性简介 .....	82
7.2 Java 应用认证和授权服务简介.....	83
7.2.1 JAAS 供应商 .....	83
7.2.2 JAAS 和 J2EE .....	84
7.3 Oracle Application Server	
安全性概述 .....	84
7.3.1 JAZNUserManager .....	85
7.3.2 XMLUserManager .....	85
7.3.3 DataSourceUserManager.....	85
7.4 创建和配置安全的应用程序.....	86
7.4.1 JVM 配置参数.....	86
7.4.2 HelloServlet 应用.....	86
7.4.3 配置 J2EE 安全性.....	88
7.4.4 配置 XML 供应商 .....	90
7.4.5 配置 jazn-data.xml 文件中	
的用户和角色 .....	90
7.4.6 将 J2EE 角色映射到	
JAAS 角色 .....	93
7.4.7 测试应用程序 .....	93
7.5 使用 JAZN ADMINTOOL	
管理 HelloServlet 应用 .....	95
7.6 使用 JAZNUserManager LDAP	
供应商 .....	96
7.6.1 配置 OID(Oracle Internet	
Directory) .....	96
7.6.2 配置 HelloServlet 应用程序.....	98
7.7 使用 DataSourceUserManager .....	99
7.7.1 DataSourceUserManager 概述.....	99
7.7.2 创建表 .....	99
7.7.3 创建数据源 .....	100
7.7.4 配置 HelloServlet 应用程序.....	100
7.8 定制安全性 .....	101
7.8.1 定制的 Usermanager.....	101
7.8.2 定制的 LoginModules.....	101
7.8.3 UserManager 或	
LoginModule .....	102
7.9 选择安全性方法 .....	102
7.10 总结 .....	102

<b>第 8 章 使用 JavaMail 和 Java API</b>	
<b>开发 XML</b> .....	<b>104</b>
8.1 JavaMail.....	104
8.1.1 使用 JavaMail 和 JNDI	
构建应用程序.....	104
8.1.2 配置 Mail 会话 .....	104
8.1.3 创建本地资源引用 .....	106
8.1.4 Mail Form.....	107
8.1.5 构建 JavaMailServlet .....	108
8.1.6 全局命名空间和本地命名	
空间的选择 .....	111
8.2 适合 XML 的 Java API.....	112
8.2.1 确定 JAX 的版本.....	112
8.2.2 重载 JAX 的实现.....	113
8.2.3 测试 JAX 的重载.....	113
8.3 小结 .....	114
<b>第 9 章 部署 J2EE 应用</b> .....	<b>115</b>
9.1 J2EE 体系结构.....	115
9.2 配置 .....	116
9.2.1 OC4J 服务器配置文件.....	116
9.2.2 应用配置文件 .....	118
9.3 组织和打包 .....	118
9.3.1 JAR 文件.....	119
9.3.2 WAR 文件 .....	120
9.3.3 EAR 文件 .....	124
9.4 部署 .....	126
9.4.1 部署步骤 .....	126
9.4.2 部署说明 .....	128
9.4.3 强制进行的自动化重部署.....	130
9.5 小结 .....	132
<b>第 10 章 Web 应用程序</b> .....	<b>133</b>
10.1 构建和部署应用程序 .....	133
10.1.1 WAR 应用程序 .....	133
10.1.2 EAR 应用程序 .....	138
10.1.3 EAR 部署技巧 .....	141
10.2 配置 orion-web.xml 文件 .....	143
10.2.1 初识 orion-web.xml .....	144
10.2.2 重载语境参数和环境变量.....	146
10.2.3 类加载 .....	148



10.2.4	文件类型、位置和基于类型的过滤器	151	12.4.5	配置连接池(connection pooling)	201
10.2.5	群集功能	154	12.4.6	配置安全性	202
10.2.6	基于 URL 失效期策略	154	12.4.7	配置事务	205
10.2.7	会话配置、会话和请求跟踪	155	12.4.8	配置日志记录	205
10.2.8	解析资源引用和 EJB 引用	157	12.5	部署连接器	206
10.2.9	安全性配置和解析安全性规则	158	12.6	小结	210
10.2.10	主机和基于网络的访问控制	160	<b>第 13 章</b>	<b>配置 Web 服务</b>	<b>211</b>
10.2.11	重载 web.xml 设置小结	160	13.1	选择 Web 服务实现类型	211
10.3	使用 ASC	161	13.2	创建 Echo Web 服务	211
10.4	小结	162	13.2.1	构建 Web 服务接口	212
<b>第 11 章</b>	<b>企业级 JavaBeans(EJB)</b>	<b>163</b>	13.2.2	构建无状态的实现	212
11.1	基本的部署步骤	163	13.2.3	构建有状态的实现	212
11.2	关于 orion-ejb-jar.xml	163	13.2.4	配置和部署 Web 服务	213
11.3	配置 EJB 部署	165	13.2.5	使用无状态 Session Beans	214
11.3.1	会话 Bean 的部署头文件	165	13.3	测试 Web 服务	216
11.3.2	实体 Bean 的部署头文件	167	13.4	创建 Web 服务的客户端	219
11.3.3	消息驱动 Bean 的部署头	183	13.4.1	构建 Apache Axis 客户端	219
11.3.4	解析 EJB 引用	184	13.4.2	构建 proxy 客户端	220
11.3.5	解析资源引用	185	13.4.3	构建 .NET 客户端	221
11.3.6	解析资源 env 引用	186	13.5	小结	223
11.3.7	重载 Environment 项	187	<b>第 14 章</b>	<b>应用程序客户端</b>	<b>224</b>
11.3.8	通过 AC4J 进行的异步调用	187	14.1	选择客户端应用类型	224
11.3.9	RMI/IIOP 安全性互操作性	189	14.2	从 J2SE 客户端应用程序进行连接	225
11.4	解析安全性角色	190	14.2.1	客户端应用程序的实现	225
11.5	小结	191	14.2.2	配置 JNDI	225
<b>第 12 章</b>	<b>J2EE 连接器</b>	<b>192</b>	14.2.3	使用 JNDI 进行验证	226
12.1	J2EE Connectors 的概念	192	14.2.4	使用 JAAS 进行验证	226
12.2	OC4J 对连接器的支持	193	14.2.5	构建客户端应用程序	227
12.3	查找连接器	193	14.2.6	运行客户端应用程序	228
12.4	配置连接器	199	14.3	从 J2EE 客户端应用程序进行连接	229
12.4.1	基本的打包和配置	200	14.3.1	实现客户端应用程序	229
12.4.2	OC4J 连接器描述符	200	14.3.2	配置 JNDI	230
12.4.3	配置名称和 JNDI 位置	201	14.3.3	实现 JAAS 的 Callback Handler(回调句柄)	230
12.4.4	从 ra.xml 重载属性值	201	14.3.4	服务器端配置	231
			14.3.5	运行客户端应用程序	232

14.4	小结	233	17.5	小结	306
<b>第 15 章</b>	<b>体系结构和部署拓扑结构</b>	<b>234</b>	<b>第 18 章</b>	<b>部署应用程序</b>	<b>307</b>
15.1	概述	234	18.1	开发概述	307
15.2	体系结构	238	18.2	创建专用的 OC4J 实例	308
15.2.1	术语	238	18.2.1	优点	308
15.2.2	基础结构	240	18.2.2	步骤	308
15.2.3	中间层	245	18.3	使用 ASC 部署和撤销应用程序	311
15.3	部署的拓扑结构	247	18.3.1	部署步骤	311
15.3.1	带有 Web Cache 的 单机 J2EE	248	18.3.2	重新部署的步骤	315
15.3.2	Business Intelligence 和 Forms	249	18.3.3	撤销部署的步骤	316
15.3.3	带有 Web Cache 的 J2EE Clusters	251	18.4	手动部署和撤销应用程序	317
15.4	小结	252	18.4.1	部署步骤	317
<b>第 16 章</b>	<b>安装和管理</b>	<b>253</b>	18.4.2	重新部署的步骤	318
16.1	安装	253	18.4.3	撤销部署的步骤	318
16.1.1	安装前的要求	254	18.5	使用 ASC 改动应用程序配置	318
16.1.2	安装步骤	263	18.5.1	默认的应用程序设置	318
16.2	日常管理	272	18.5.2	具体的应用程序设置	319
16.2.1	管理职责	272	18.6	小结	322
16.2.2	ASC 工具包	273	<b>第 19 章</b>	<b>配置 web Cache</b>	<b>323</b>
16.2.3	命令行工具包	278	19.1	概述	323
16.3	小结	285	19.1.1	体系结构	323
<b>第 17 章</b>	<b>配置 J2EE 服务</b>	<b>286</b>	19.1.2	优点	326
17.1	J2EE 概述	286	19.1.3	缺点	327
17.1.1	部署层	286	19.2	配置 Web Cache	328
17.1.2	中间层代码类型	287	19.2.1	管理工具	328
17.1.3	版本信息	288	19.2.2	配置步骤	333
17.2	OC4J 体系结构	289	19.2.3	配置属性	338
17.2.1	结构	289	19.3	日志记录和监测	341
17.2.2	处理过程	291	19.3.1	事件和访问日志	341
17.3	配置工具和选项	292	19.3.2	Web Cache Statistics 页面	342
17.4	用于监测和调整的工具	300	19.3.3	Origin Server Statistics 页面	343
17.4.1	Application Server 控制日志记录工具(ASC)	300	19.3.4	Popular Requests	343
17.4.2	AggreSpy 和 dmstool	302	19.3.5	Web Cache Detailed Statistics Monitor	344
17.4.3	iHAT (Hi-AV)工具	304	19.4	缓存和失效规则设置	345
			19.4.1	默认的缓存设置	345
			19.4.2	超期终止策略	347

19.4.3	失效方法	348	20.5.1	用户管理问题	371
19.5	小结	349	20.5.2	身份管理解决方案	372
<b>第 20 章</b>	<b>配置企业安全性</b>	<b>350</b>	20.5.3	身份管理的实现	372
20.1	威胁和受影响的当事人	350	20.6	小结	375
20.1.1	威胁	350	<b>第 21 章</b>	<b>配置 Clustering 和失效切换</b>	<b>376</b>
20.1.2	人人都会受影响	351	21.1	Clustering 概述	376
20.2	Web 安全的基础知识	352	21.1.1	可升级的性能	376
20.2.1	加固每一层	352	21.1.2	高可用性	377
20.2.2	网络设计和防火墙	355	21.1.3	Clustering 定义	378
20.2.3	关闭不必要的服务 和端口	357	21.2	10g AS Clustering	379
20.2.4	打补丁	357	21.3	创建 Farm	381
20.2.5	定期检查安全性警告	359	21.3.1	安装基础结构	381
20.2.6	使用常识判断并保持警醒	359	21.3.2	加入基础结构	385
20.2.7	安全性的成本	361	21.3.3	安装中间层	391
20.3	确保 ASC 工具安全	361	21.4	创建 Cluster	392
20.3.1	更改默认的 ASC 端口	361	21.4.1	创建 Cluster 实例	392
20.3.2	更改 ASC 的 ias_admin 密码	362	21.4.2	向 Cluster 添加成员	393
20.3.3	使用 SSL 对 ASC 通信加密	363	21.4.3	配置 Web Cache	396
20.4	SSL 和 OWM	365	21.5	部署集群的应用	401
20.4.1	密钥	366	21.6	实现 Failover	402
20.4.2	证书	367	21.7	小结	405
20.4.3	OCA	367	<b>附录 A</b>	<b>服务器配置文件和文档 类型声明</b>	<b>406</b>
20.4.4	OWM	367	<b>附录 B</b>	<b>部署描述符和文档类型声明</b>	<b>425</b>
20.4.5	设置 SSL	370	<b>附录 C</b>	<b>命令行工具使用方法</b>	<b>431</b>
20.5	身份管理	371	<b>附录 D</b>	<b>调试技巧</b>	<b>443</b>

# 第 1 章 Oracle 的特性和版本概述

当用户准备开始评估 Oracle Application Server 10g 时，就会发现自己已经迷失在 Oracle 版本、发行版和非 J2EE 特性的海洋之中。本章将介绍每个 Oracle 版本的具体内容，主要包括以下几方面：

- 如何在 Java、Standard Edition 和 Enterprise Edition 之间选择
- 识别发行版本号的方法
- 在全部 Application Server 中哪种 J2EE 技术是标准的
- Oracle 10g AS 的特性
- Oracle 10g AS 中包含的诸如 Oracle HTTP 服务器(OHS)、Oracle Internet 目录(OID)、Single Sign-On( SSO)、Web Cache 和 Clustering 等非 J2EE 特性

## 1.1 选择版本

同大多数 Web Application Server 一样，Oracle 10g AS 有许多版本可供使用，小到仅用于开发的 J2EE Server 版本，大到包罗万象的 Enterprise 安装版本。所有这些中都包括了基本的 J2EE 容器(也称为 Oracle 10g AS Container for J2EE，或 OC4J)，它支持 Servlet、JSP、EJB 和其他标准 J2EE 技术。其他功能更完善和费用更高的版本包括附加的非 J2EE 特性，诸如 Oracle 改进的 Apache Web Server 之类早已面世的 Oracle 产品如 Portal、Form、Report 和 Discoverer，Oracle 的身份管理产品如 Oracle InternetDirectory(OID)、SSO 以及高性能高可用性站点的高级特性诸如 WebCache 和 Clustering。没有任何一种产品可以满足全部要求的，因而重点是定制合适的版本，以满足应用的要求。下文列出了一些可用的 10g AS 版本：

- OC4J Standalone(Oracle 10g AS Container for J2EE)。只需支付一定的费用就可以得到这个纯 java、面向开发的，支持全部的 J2EE 技术的版本。该产品未使用正式的管理或监控工具，而采用命令行和基于文件的配置。其安装容量和内存需求都较小，因而适合作为开发人员工作站的理想平台。但并不适合产品系统。为此，仍须购买下文列出 3 个版本之一。该版本是本书开发和配置章节(第 2~14 章)中主要讨论的内容。
- Java 版。这款针对低端用户的商业版包括 OC4J 的全部 Standalone 特性并添加了一个基于 Web 的管理工具、一个更为健壮的 HTTP Server(基于开源 Apache WebServer)，并支持通过 Toplink 进行非 J2EE 的对象关系映射，提供 JDeveloper 的 5 个许可，一个功能完整的 Java 集成开发环境。
- Standard Edition。该版本包括了 Java 版中的全部特性，并添加了 Portal、SSO 和 Content Management 功能。由于 Java 版适合开发独立的 J2EE 应用，而 Standard Edition 中更高级的特性使开发人员可以在企业系统内将大量不同的应用组合到一起，为用户提供一个通用的界面(通常通过 Enterprise Information Portal/EIP 实现)。

- Enterprise Edition。这款针对高端用户的版本包括 Standard Edition 中的各种功能并添加了 We Cache(Oracle Web Cache)、企业集成功能(包括 Oracle 的 SSO)、目录服务(OID)、数据分析和报表功能(包括使用 Oracle Report 和 Discoverer 实现的 Business Intelligence)。此外,该版本提供了通过 WebCache 提供的聚集了 J2EE 实例的功能。Web Cache 确保了最优的负载均衡和高可用性。这些特性使得诸如 Form、Report 和 Discoverer 等 Oracle 产品在长时间运行时具有额外的可升级性、互操作性并可以进行实时分析。通常 Enterprise Edition 需要一个基础结构数据库(在安装产品时,会提供该数据库),用以支持这些高级功能。本书的 Oracle10g AS 管理部分(15~21 章)将重点介绍该版本内容。

表 1-1 给出了各个版本在费用、安装容量和功能特性上的比较。比较时间截止于本书撰稿时间:2004 年 6 月。本章将对列举的每种特性作深入的阐述。

表 1-1 Oracle 10g AS 不同版本之间的比较

特 性	OC4J Standalone	Java Edition	Standard Edition	Enterprise Edition
报价	包含在其他版本报价中	\$5,000/CPU	\$10,000/CPU	\$20,000/CPU
大小	28 MB	超过 500MB	个人不能下载	超过 2GB
配置类型	手动配置	企业管理人员	企业管理人员	企业管理人员
J2EE 容器	√	√	√	√
HTTP 服务器		√	√	√
JDeveloper		√	√	√
Toplink		√	√	√
Application Server Control(ASC)		√	√	√
Portal			√	√
Single Sign-On(SSO)			√	√
内容管理			√	√
Web Caching				√
集成				√
Oracle Internet Directory				√
Discoverer				√
报表服务				√
个性化功能				√
无线功能				√

## 确定所需性能

最后, Standard Edition 和 Enterprise Edition 适合于大量额外的(非 J2EE)插件产品。因此,在选择任何一个版本的时候,首先要决定真正需要的插件产品。然后将绑定的 Oracle 插件和其

他开发商开发的候选产品做比较，以确定哪个产品最能满足需要，性价比最好。

例如，如果在 Oracle 10g AS 中运行多个内联网应用程序，并希望用户对每个应用程序都能使用相同的用户名和密码，那么您肯定就会对 SSO 功能感兴趣了。这种情况下可以选择购买 Oracle 10g AS Standard Edition，使用 Oracle 10g AS 内置的 SSO 功能。然而，如果 Oracle SSO 不能满足您的要求，那么可以选择使用 Oracle 10g AS JavaEdition，并购买诸如 RSA ClearTrust 这样的第三方 SSO 集成产品即可。

您必须考虑一下使用 Oracle 10g AS 的总投资。纯 J2EE 应用非常方便，但当您开始集成其他的 Oracle 产品时，就会被“困在”Oracle 10g AS 中了，并且发现将它移植到其他 Application Server Application Server 比较难。如果认为最终可能会选择使用其他供应商(如 BEA 或 IBM)的产品，那么可以选择能够很好集成在多个 Application Server 中的第三方插件产品。另一方面，如果确定应用程序很长时间要运行在 Oracle 上的话，那么选择集成和打折处理的 Oracle 插件产品就会节约大量的时间和金钱。这对那些将 Oracle 作为自己公司数据库的公司尤其如此。如果已经使用 Oracle 作为自己的数据库的话，那么选择 Oracle 10g AS 的一个明显的争议就是数据库和 Application Server 的紧密集成，由此可以带来性能的提高和管理的便捷。

表 1-2 列出的一些有助于确定所需性能准则。

表 1-2 选择 Oracle 10g AS 不同版本的提示

开发人员构建和配置的应用	推荐版本
仅 J2EE 应用	JavaEdition
多个未关联的 J2EE 应用	Java Edition
公司的内联网或门户	Standard Edition(或第三方门户网站和 SSO 产品)
与多个外部系统或业务伙伴集成的应用	Enterprise Edition(或第三方集成产品)
Oracle PL/SQL WebToolkit	Standard Edition 或 Enterprise Edition
Oracle Portal	Standard Edition 或 Enterprise Edition
Oracle Forms、Reports、Discoverer	使用 Oracle business intelligence 选项的 Enterprise Edition
高流量电子商务站点	具有 WebCache 功能的 Enterprise Edition
高可用性站点	具有 Clustering 功能的 Enterprise Edition
需要集成企业范围的身份管理系统的應用	Enterprise Edition(或第三方 LDAP(轻量级目录访问协议)和 SSO 产品)
无线设备上可使用的應用	Enterprise Edition(或第三方无线网关)

## 1.2 正确选择发行版本

Oracle Application Server 发行的版本号并不遵循常规。尽管多数 Application Server 开发商公布的主版本号是以大数递增的，但 Oracle 公司是跨整个产品线调整版本的。这意味着对 J2EE 容器做的主要修整工作只会对 Oracle 10g AS 平台版本做少量的改动(例如从 9.0.2 到 9.0.3)。因此，如果认为这些次版本号不同的版本之间 J2EE 配置文件兼容，就会非常危险。

**提示:**

9.0.2 版只支持 J2EE1.2 规范(除极个别情况外), 而 9.0.3 版与 J2EE1.3 兼容。当前发行的 9.0.4 版也与 J2EE1.3 兼容。

此外, 某些版本只包含了 Oracle 10g AS 的部分服务。例如 9.0.3 版只更新了 J2EE1.2 和 Toplink, 并要求已经安装的 9.0.2 版包含诸如 Clustering 和高级安全性等企业特性在内。

本书主要关注 9.0.4 版。该版本是一个全新平台。其官方名称为“Oracle 10g Application Server(9.0.4 版), 简称为“Oracle 10g AS”。虽然名称是 Oracle 10g AS, 但产品版本号却是 9.0.4, 这很容易让人误解。下面是对该产品的一个历史回顾, 这对于了解贯穿本书介绍全部更新特性很有帮助。

### 1.2.1 OAS

用 Oracle 编写的 Oracle Application Server 的最后一个版本为 OAS 4.0.8.2。该产品存在很多问题, 因而最终停产了。

### 1.2.2 iAS

Oracle Internet Application Server 的第一个发行版本。该版本基于 Oracle 8i 技术, 并使用了 Jserv 的 Apache Web Server。该产品对 OAS 进行了重新开发。Oracle 公司精明之处在于将成功的产品(如 Apache)集成在 iAS 中而不是“重新开发新产品”。在该产品中使用 JServ 实现 Java 全部功能。

### 1.2.3 9iAS v1

这是基于 Oracle 8i 技术和 Oracle Developer 6i 的 Internet Application Server 的第一个发行版。虽然该产品仍然使用 Apache 和 Jserv, 但是通过 Apache Proxypass 命令可以将 Internet Application Server 和 OC4J Standalone 集成在一起。该产品是第一个得到广泛使用的产品。由于它的 1.0.2 版和 1.0.2.2.2 版支持 Oracle Developer 6i 应用, 所以目前仍在使用之中。

### 1.2.4 9iAS v2

该版本的产品体系结构发生了根本性的变化。首先它在产品内完全集成了 OC4J, 可以直接支持 J2EE 应用, 并作为诸如 Oracle Form 和 Report 等其他组件的一种间接支持手段。虽然以前版本的产品都在尝试着将部分组件集成在一起, 但直到 9iAS v2 才真正地将诸如 ASC、OHS 和 OC4J 这类组件集成在一起。但同大多数产品一样, 该产品也存在许许多多问题。9i AS v2 有严格的安装要求, 对 ASC 响应速度慢, Oracle Reports 的不稳定性, OID 问题以及各种 bug。从某种程度上讲, 虽然该产品对生产系统很有用也有自己的优点, 但它就是出问题, 尤其是对第一次使用它的工程更是如此。该产品包含两个版本: 9.0.2 版和 9.0.3 版。虽然新项目已经更新到了 Oracle 10g AS 并且当前项目也将更新到 Oracle 10g AS, 但是到本书编写的时候为止, 这两个版本的产品仍在使用中。

### 1.2.5 10g AS

Oracle 10g AS 的最初发行版实际上是盼望已久的 9.0.4 打补丁后发行的版本。虽然它包含

了许多新特性，尤其在安全性和身份管理方面，但是该产品没有实质性的变化，只是向前发展了一些。虽然 9iAS 第 2 版存在自身的不足，但 Oracle 10g AS 却非常可靠。它修补了 9iAS 第 2 版中存在的问题并采用了 Java1.4。虽然它看上去很稳定、可靠、比较好用，但这一切的前提是开发人员必须很好的了解 Java、Oracle 和 Web server 管理。本书是针对 9.0.4 版(即 Oracle 10g AS)编写的。

#### 提示：

怎样才能成为一个 Enterprise Edition 10g AS 管理员？显而易见，很大程度上取决于工程 (J2EE、Oracle、Apache 等) 的本质。但是到目前为止，需要的是复合型管理员，他要么是了解 Apache、J2EE 和 Orion、Oracle 开发产品、中间级 OracleDatabase Administration(DBA)、安全性、系统体系结构和规划以及网络通信，要么可以很快学会这些技术。看上去内容很多，但实际上有许多专题只需要理解。管理 10g AS 需要具备 Oracle DBA、Web 管理员、开发人员 (J2EE 或 PL/SQL)、系统管理员和网络管理员的技能。由于产品涉及到如此众多的不同领域，所以这些技能必不可少。绝大多数选中的人要么从事过系统管理工作，要么从事过数据库管理工作，但最重要在于他们具备学习新技术和解决问题的能力。

## 1.3 J2EE 特性

每个 J2EE Application Server 都要提供 “J2EE Container”，这有助于开发人员构造和配置 Web 应用、EJB、J2EE Connector 和客户端应用。它还必须支持 J2EE 的全部技术，这包括 JMS (Java Message Service)、JDBC (Java Database Connection)。Oracle 10g AS 9.0.4 支持 J2EE 1.3 版。而 J2EE 1.3 版需要得到表 1-3 所列举的规范的支持。

表 1-3 J2EE1.3 技术

规 范	版 本
Servlet	2.3
JSP	1.2
EJB	2.0
J2EE Connector	1.0
JavaTransaction API	1.0.1b
JDBC	2.0
JMS	1.0.2b
JNDI (Java Naming and Directory Interface)	1.2.1
JavaMail	1.2

下面简要介绍了 J2EE 的主要技术。后面的章节将详细讨论这些组件的配置和部署。

### 1.3.1 Servlet 和 JSP

Java Servlet 属于服务器范畴，响应 HTTP 请求。JSP 则合并了 Java Script lets、自定义标签



和标记(如 HTML、WML 和 XML 等)以表现对 HTTP 的响应。这两种技术经常一起使用,用于构建基于 Model-View-Controller(MVC)的 Web 应用。在这个常见的设置(也称为“模型 2 体系结构”)中,Servlet 解析到达的请求,执行业务逻辑,更新数据对象并将控制传递给 JSP,由 JSP 显示最终结果。第 10 章将介绍 Web 应用程序的配置和部署。

### 1.3.2 EJB(Enterprise JavaBeans)

EJB 是使用由容器管理的安全性、事务、持久性、远程以及其他 J2EE 服务的商业组件。有三类 EJB: Entity Bean、Session Bean 和 message-driven beans。Entity Bean 对应于数据库中的数据行。严格的讲,它常用于数据访问。Session Bean 包含了复杂的业务逻辑,通常用于管理一个或多个 entity Bean 或数据库表。Message-driven beans 响应异步传递给 JMS topic 或 queue 的消息。本书第 11 章将介绍 EJB 的配置和部署。

### 1.3.3 JMS(Java Message Service)

JMS 允许 J2EE 组件之间通过 topic(公布/订购)和 queue(点对点)进行异步通信。这对控制请求、剔除长时间运行的进程或响应异步请求非常有用。第 6 章将介绍 JMS 的配置和部署。

### 1.3.4 JDBC 和 J2EE Connector

开发人员通过 JDBC Datasource 可以和关系型数据库方便的交互。J2EE Connector 则可以确保和外部其他类型的系统进行交互。这两种技术都可以进行事务管理并确保多个系统事务的一同提交或回滚。第 5 章将讨论 JDBC 数据源的配置和使用。第 12 章将介绍 J2EE Connector 的配置、部署和使用。

### 1.3.5 事务

事务可以确保对数据库和其他系统的相关变化一起提交或回滚,这使得开发人员能够防止数据被破坏。J2EE 提供了 Java Transaction API(JTA)用于编程实现事务的启动,它还可以在 EJB 的部署描述符内声明性地配置 EJB 事务设置。第 11 章将介绍配置声明性事务的相关内容。

### 1.3.6 安全性

J2EE 既可以编程实现安全性(从代码内部手动检查许可权限),也可以声明安全性(设置哪些角色可以访问组件和方法)。绝大多数应用程序都要具备这两种安全性能,并且这两种安全性都取决于相同的潜在的安全性体系结构,这些内容在第 7 章介绍。第 20 章将深入讨论 Application Server 的安全性问题。

### 1.3.7 其他 J2EE 服务

J2EE 包含了大量其他服务和 API,它们可以简化编程工作,而不需要进行显式的服务器配置。第 5 章将专门讨论 JNDI,第 8 章将会介绍 JavaMail 和 XML 编程使用的 Java APIs(JAX)。