

Java EE实用教程

- ◆ Java EE概述
- ◆ Java EE实训介绍
- ◆ 超文本标记语言HTML
- ◆ Servlet、JDBC与JSP
- ◆ JavaBean与JavaScript
- ◆ 过滤器与监听器
- ◆ Struts 2、Spring与Hibernate
- ◆ AJAX 简介
- ◆ 使用SVN协同开发



王映龙 刘春阳 熊曾刚 编著

高等学校计算机应用规划教材

Java EE 实用教程

王映龙 刘春阳 熊曾刚 编著

清华大学出版社

北 京

内 容 简 介

本书介绍了从 Java EE 入门到当前流行的 Struts 2、Spring、Hibernate、AJAX 的主要内容,在介绍时并非简单地罗列 Java EE 的技术及内容,而是希望通过对本书的学习,学生能够动手实践 Java EE,理解 Java EE 技术的核心,对 Java EE 编程产生兴趣。本书以一个小组管理的例子贯穿所有内容,由浅入深,循序渐进,可使学生从易到难,逐步理解并掌握。

本书结构清晰,案例丰富,技术讲解独到,可作为高等学校计算机相关专业 Java EE 课程的教材,也可作为 Java EE 实训教材。

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。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java EE 实用教程/王映龙,刘春阳,熊曾刚 编著. —北京:清华大学出版社,2011.7
(高等学校计算机应用规划教材)

ISBN 978-7-302-26207-7

I. J… II. ①王… ②刘… ③熊… III. JAVA 语言—程序设计—高等学校—教材
IV. TP312

中国版本图书馆 CIP 数据核字(2011)第 129449 号

责任编辑:刘金喜

装帧设计:孔祥丰

责任校对:胡花蕾

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×260 印 张:14.25 字 数:356 千字

版 次:2011 年 7 月第 1 版 印 次:2011 年 7 月第 1 次印刷

印 数:1~4000

定 价:22.00 元

产品编号:042726-01

前 言

本书根据作者多年为江西农业大学软件学院的学生在本校和北京实训基地(与中科院计算所共建)进行实训的授课经验编写而成。本书并非简单地罗列 Java EE 的技术,而是希望通过对本书的学习,学生能够动手实践 Java EE,理解 Java EE 技术的核心,对 Java EE 编程产生兴趣,具有面对问题的勇气,并不断提升解决问题的能力。

本书通过一个小组管理的例子贯穿所有内容,这是因为实训是分组的,学生对小组管理的内容比较熟悉。由于始终是一个例子,学生更容易体会各种技术的特点、采用框架带来的好处。在 Java EE 实训的教学实践中,作者也是采用这个例子来讲解 Java EE 中的各种技术的。主要的技术都讲解完后,进入项目阶段,这个阶段以学生为主,老师起辅助作用,每个组自己讨论确定一个感兴趣的项目,使用已经掌握的技术进行开发。

本书包含了从 Java EE 入门直到当前流行的 Struts 2、Spring、Hibernate、AJAX 的精华,但不包含细枝末节的内容,因为繁琐的技术细节很容易让学生产生畏难情绪。对于已经领会这些核心内容想进一步学习技术细节的学生,可以从网上或其他书籍获取这些资源,寻找资源的能力也是学生应该掌握的。

本书的内容安排是层层递进的,对于没有学过 Java EE 的学生,可以由浅入深地逐步学习;对于已经掌握了一些 Java EE 知识的学生,学习这些内容时可以当做一次提纲挈领的复习,因为人是会遗忘的。

本书涵盖了 HTML、Servlet、JSP、JavaScript、Struts、Spring、Hibernate、AJAX 等主题,将每个主题展开都可以单独成书,因此在内容的选择上经过了反复思考,既体现了每个主题的核心技术和思想,又不能占用过多的篇幅。

本书各章主要内容如下。

第 1 章(Java EE 概述)让学生对 Java EE 有一个整体的了解。

第 2 章(Java EE 实训介绍)分两节进行介绍,第一节是作者多年实训授课的经验和心得。第二节介绍了实训中要用到的几个软件的获取和安装方法。

第 3 章(超文本标记语言 HTML)介绍了 HTML 文件结构、页面部署到服务器、基本标签和表单。这些都是 Java EE 开发中必须掌握的。对于以前学过 HTML 语言的学生,这是对 HTML 语言的一次复习,也是 Java EE 实训的热身;对于以前没有学过 HTML 语言的学生,可以通过这一章学会 HTML 语言的主要内容,能够进行随后的 Java EE 实训,对 HTML 语言的其他细节具备自己查找资料学习的能力。

第 4 章(Servlet)介绍了 Servlet 技术的主要内容。Servlet 技术是 Java EE 的核心,掌握了 Servlet 技术,才能理解 Java EE 的其他技术。

第 5 章(JDBC)介绍了 JDBC 以及在 Servlet 中如何使用 JDBC。因为 Java EE 应用中很多

时候都会用到数据库，所以本书在这里讲解 JDBC 的主要内容。本章还使用 Servlet 和 JDBC 开发了小组管理系统。

第 6 章(JSP)介绍了 JSP 的主要内容。JSP 技术是 Java EE 开发离不开的。

第 7 章(JavaBean)介绍了 JavaBean、模式 1 和模式 2，并分别用模式 1 和模式 2 实现了小组管理系统，让学生对比体会 MVC 模式的优点。

第 8 章(JavaScript)介绍了在 Java EE 开发中经常用到的 JavaScript 技术。介绍 JavaScript 既是因为在 Java EE 开发中会经常用到 JavaScript，也是为后面介绍 AJAX 做准备。

第 9 章(过滤器与监听器)介绍了过滤器与监听器的相关知识，在 Java EE 开发中，Struts 2 是基于过滤器的，Spring 是基于监听器的。

第 10 章(Struts 2)、第 11 章(Spring)、第 12 章(Hibernate)、第 13 章(AJAX)介绍了近几年 Java EE 开发中非常流行的技术，每一章都讲解了该技术最本质的内容，并通过小组管理系统这个例子让学生实践体会。

第 14 章(使用 SVN 协同开发)介绍了一个团队如何使用版本控制软件 SVN 进行协同开发。

本书结构清晰，案例丰富，技术讲解独到，可作为高等学校计算机相关专业的教材，也可作为 Java EE 实训教材。

参加本书编写的有江西农业大学软件学院王映龙教授，中科院计算所刘春阳高级工程师，湖北孝感学院计算机与信息科学学院熊曾刚博士。

本书的编写得到江西农业大学软件学院和北京实习基地的所有领导、老师、工作人员以及学生的大力帮助，在此一并向他们表示诚挚的感谢！

尽管作者在写作过程中投入了大量的时间和精力，但由于水平有限，错误和不足之处仍在所难免，敬请读者批评指正。

作者
2011 年 4 月

目 录

第 1 章	Java EE 概述	1
1.1	Java EE 的概念	1
1.2	Java EE 简史	1
1.3	Java EE 的优势	2
1.4	Java EE 的四层模型	3
1.5	Java EE 的结构	6
1.5.1	容器服务	6
1.5.2	容器类型	7
第 2 章	Java EE 实训介绍	8
2.1	Java EE 实训中的几个问题	8
2.1.1	分组	8
2.1.2	写日志	9
2.1.3	面对出错, 不要失去信心	9
2.1.4	不断重复很重要	10
2.1.5	要能够坚持	10
2.1.6	学不下去怎么办	10
2.2	实训中用到的软件	10
2.2.1	Java 开发工具包 JDK	10
2.2.2	Java EE 服务器 Tomcat	14
2.2.3	开发平台 Eclipse	19
2.2.4	数据库 MySQL	22
2.2.5	数据库图形界面工具 Navicat	31
2.3	练习题	32
第 3 章	超文本标记语言 HTML	33
3.1	HTML 文件结构	33
3.2	部署到服务器	34
3.3	基本标签	36
3.3.1	分段	36
3.3.2	字体	37

3.3.3	图片	40
3.3.4	链接	41
3.3.5	表格	41
3.3.6	层	42
3.4	表单	43
3.5	练习题	45
第 4 章	Servlet	46
4.1	工作原理	46
4.2	简单例子	47
4.2.1	创建 Java Web 项目	47
4.2.2	配置 Web 服务器	49
4.2.3	添加 servlet-api.jar	51
4.2.4	编写 Servlet 代码	52
4.3	生命周期	57
4.4	处理表单	58
4.5	使用 Eclipse 进行跟踪调试	63
4.6	使用 session	65
4.7	ServletConfig 与 ServletContext	66
4.7.1	ServletConfig	66
4.7.2	ServletContext	68
4.8	转发与重定向	69
4.8.1	转发	69
4.8.2	重定向	70
4.9	练习题	71
第 5 章	JDBC	72
5.1	在 MySQL 中创建数据库和表	72
5.1.1	启动 MySQL 服务器	72
5.1.2	使用数据库图形界面工具 Navicat 创建数据库和表	73
5.2	JDBC 常用的几个类	77
5.3	简单例子	79
5.3.1	MySQL 的 JDBC 驱动程序	79
5.3.2	增加数据的例子	82
5.3.3	查询数据的例子	83
5.4	在 Servlet 中使用 JDBC	85
5.4.1	小组管理系统	85
5.4.2	保存表单数据	88

5.4.3 显示数据	92
5.5 练习题	95
第 6 章 JSP	96
6.1 JSP 简介	96
6.1.1 一个小例子	96
6.1.2 工作原理	100
6.2 基本语法	101
6.2.1 指令	101
6.2.2 声明	101
6.2.3 显示的表达式	101
6.2.4 Java 程序片段	101
6.3 处理表单	102
6.4 内置对象	104
6.4.1 request 对象	104
6.4.2 response 对象	105
6.4.3 session 对象	105
6.5 练习题	106
第 7 章 JavaBean	107
7.1 JavaBean 简介	107
7.2 在 JSP 中使用 JavaBean	109
7.3 模式 1	112
7.4 模式 2	114
7.4.1 增加小组	115
7.4.2 显示小组	118
7.5 练习题	121
第 8 章 JavaScript	122
8.1 JavaScript 简介	122
8.2 表单对象与事件	124
8.2.1 表单对象	124
8.2.2 事件	125
8.3 使用自定义函数	128
8.3.1 自定义函数	129
8.3.2 JavaScript 文件	130
8.4 练习题	132

第 9 章 过滤器与监听器	133
9.1 过滤器	133
9.1.1 工作原理	133
9.1.2 Filter 接口	134
9.1.3 一个例子	134
9.2 监听器	136
9.3 练习题	140
第 10 章 Struts 2	141
10.1 Struts 2 简介	141
10.1.1 工作原理	141
10.1.2 Struts 2 的软件包	142
10.2 创建一个 Struts 2 项目	143
10.3 Struts 2 标签	147
10.3.1 显示小组	147
10.3.2 修改小组	149
10.4 ActionSupport	153
10.4.1 静态常量	153
10.4.2 校验	154
10.5 在 Struts 2 中使用 session	155
10.6 拦截器	159
10.6.1 面向方面编程(AOP)	160
10.6.2 拦截器程序 PrivilegeInterceptor	160
10.6.3 在 struts.xml 中设置	161
10.7 练习题	163
第 11 章 Spring	164
11.1 Spring 简介	164
11.1.1 依赖注入(DI)	164
11.1.2 Spring 的软件包	164
11.2 简单例子	167
11.3 Struts 2 与 Spring 相结合	170
11.4 练习题	175
第 12 章 Hibernate	176
12.1 Hibernate 简介	176
12.1.1 对象关系映射(ORM)	176
12.1.2 Hibernate 的软件包	176
12.2 使用 Hibernate 实现增删改查	179

12.3	一对多	184
12.4	练习题	189
第 13 章	AJAX	190
13.1	AJAX 简介	190
13.1.1	用 JavaScript 更新层的内容	190
13.1.2	AJAX 工作原理	191
13.2	编写 AJAX 程序	192
13.3	练习题	196
第 14 章	使用 SVN 协同开发	197
14.1	SVN(Subversion)简介	197
14.1.1	SVN 服务器软件 Visual SVN Server	197
14.1.2	Eclipse 的 SVN 插件 Subclipse	199
14.2	在 Eclipse 中使用 SVN	202
14.2.1	Visual SVN Server 的设置	202
14.2.2	在 Eclipse 中使用 SVN 协同开发	207
14.3	练习题	216

第1章 Java EE概述

Java EE 是当前的一个主流开发平台，本章是对 Java EE 的简单介绍，让大家对 Java EE 的全貌有一个了解。

本章重点内容：

- Java EE 的概念。
- Java EE 的优势。
- Java EE 的四层模型。
- Java EE 的结构。

1.1 Java EE 的概念

Java EE(Java Platform, Enterprise Edition) 技术是 Java 语言平台的扩展，支持开发人员创建可伸缩的强大的可移植企业应用程序。它为应用程序组件定义了 4 种容器：Web、Enterprise JavaBean(EJB)、应用程序客户机和 Applet。一个应用服务器规范详细描述了这些容器和它们必须支持的 Java API，这促使 Java EE 产品之间相互竞争，同时又保证符合这个规范的应用程序可以在服务器之间移植。

1.2 Java EE 简史

- J2EE 1.2(于 1999 年 12 月发布)：这是第一个 J2EE 版本，随着 Java 2 Standard Edition(J2SE) 发布。它包含 10 个规范和 API，针对企业应用程序所需的常用的 Web 层、业务逻辑、表示层和消息传递服务。
- J2EE 1.3(于 2001 年 9 月发布)：通过 JSR 58 发布，这个版本改进了半数 J2EE 1.2 规范并引入了 XML API、连接器架构和安全框架。
- J2EE 1.4(于 2003 年 11 月发布)：这个版本改进了 J2EE 1.3 中 13 种技术中的 9 种，并引入了新的 Web 服务和安全支持。

- **Java EE 5**(于 2006 年 5 月发布): Java EE 5 主要关注提高开发人员的生产率, 它提供了更简单的编程模型, 而没有牺牲平台的任何功能。更简单的开发模型主要由两个机制提供——Java 注解和更好的默认行为。其主要的功能性改进包括改进了 Web 服务支持以及在平台中集成了 Java Server Faces(JSF)和 Java Standard Tag Library (JSTL)。

J2EE 1.4 之后就是 Java EE 5, 因为 Sun 已经从名称中去掉了“2”(这是从 Java 1.2 被称为“Java 2”时开始的做法), 而且现在在技术的简写名中使用单词“Java”而不是“J”。现在的标准版是 Java SE 6(而不是 J2SE 1.6), 企业版是 Java EE 5(而不是 J2EE 1.5)。

1.3 Java EE 的优势

Java EE 为搭建具有可伸缩性、灵活性、易维护性的系统提供了良好的机制。

(1) 保留现存的 IT 资产

由于企业必须适应新的商业需求, 利用已有的企业信息系统方面的投资, 而不是重新制定全盘方案就变得很重要。这样, 一个以渐进的(而不是激进的, 全盘否定的)方式建立在已有系统之上的服务器端平台机制是公司所需求的。Java EE 架构可以充分利用用户原有的投资, 如一些公司使用的 BEA Tuxedo、IBM CICS、IBM Encina、Inprise VisiBroker 以及 Netscape Application Server。这之所以成为可能, 是因为 Java EE 拥有广泛的业界支持和一些重要的企业计算领域供应商的参与。每一个供应商都对现有的客户提供了不用废弃已有投资, 进入可移植的 Java EE 领域的升级途径。由于基于 Java EE 平台的产品几乎能够在任何操作系统和硬件配置上运行, 现有的操作系统和硬件也能被保留使用。

(2) 高效的开发

Java EE 允许公司把一些通用的、很繁琐的服务端任务交给中间件供应商去完成。这样开发人员可以集中精力于如何创建商业逻辑, 相应地缩短了开发时间。高级中间件供应商提供以下复杂的中间件服务。

- **状态管理服务**——让开发人员写更少的代码, 不用关心如何管理状态, 这样能够更快地完成程序开发。
- **持续性服务**——让开发人员不用对数据访问逻辑进行编码就能编写应用程序, 能生成更轻巧, 与数据库无关的应用程序, 这种应用程序更易于开发与维护。
- **分布式共享数据对象 CACHE 服务**——让开发人员编制高性能的系统, 极大提高整体部署的伸缩性。
- **支持异构环境**——Java EE 能够开发部署在异构环境中的可移植程序。基于 Java EE 的应用程序不依赖任何特定操作系统、中间件、硬件。因此设计合理的基于 Java EE

的程序只需开发一次就可部署到各种平台。这在典型的异构企业计算环境中是十分关键的。Java EE 标准也允许客户订购与 Java EE 兼容的第三方的现成组件，把它们部署到异构环境中，节省了由自己制订整个方案所需的费用。

(3) 可伸缩性

企业必须要选择一种服务器端平台，这种平台应能提供极佳的可伸缩性去满足那些在系统上进行商业运作的大批新客户。基于 Java EE 平台的应用程序可被部署到各种操作系统上。例如可被部署到高端 UNIX 与大型机系统，这种系统单机可支持 64 至 256 个处理器。Java EE 领域的供应商提供了更为广泛的负载平衡策略。能消除系统中的瓶颈，允许多台服务器集成部署。这种部署可达数千个处理器，实现可高度伸缩的系统，满足未来商业应用的需要。

(4) 稳定的可用性

一个服务器端平台必须能全天候运转以满足公司客户、合作伙伴的需要。因为 Internet 是全球化的、无处不在的，即使在夜间按计划停机也可能造成严重损失。若是意外停机，那会有灾难性后果。Java EE 部署到可靠的操作环境中，它们支持长期的可用性。一些 Java EE 部署在 Windows 环境中，客户也可选择健壮性能更好的操作系统如 Sun Solaris、IBM OS/390。最健壮的操作系统的可用性可达到 99.999% 或每年只需 5 分钟停机时间。这是实时性很强的商业系统的理想选择。

1.4 Java EE 的四层模型

Java EE 使用多层的分布式应用模型，应用逻辑按功能划分为组件，各个应用组件根据它们所在的层分布在不同的机器上。事实上，SUN 设计 Java EE 的初衷正是为了解决两层模式(Client/Server)的弊端，在传统模式中，客户端担当了过多的角色而显得臃肿。在这种模式中，第一次部署的时候比较容易，但难于升级或改进，可伸展性也不理想，而且经常基于某种专有的协议(通常是某种数据库协议)。它使得重用业务逻辑和界面逻辑非常困难。现在 Java EE 的多层企业级应用模型将两层化模型中的不同层面切分成许多层。一个多层化应用能够为不同的每种服务提供一个独立的层，以下是 Java EE 典型的四层结构，如图 1-1 所示。

- 运行在客户端机器上的客户层组件。
- 运行在 Java EE 服务器上的 Web 层组件。
- 运行在 Java EE 服务器上的业务逻辑层组件。
- 运行在 EIS 服务器上的企业信息系统(Enterprise Information System)层软件。

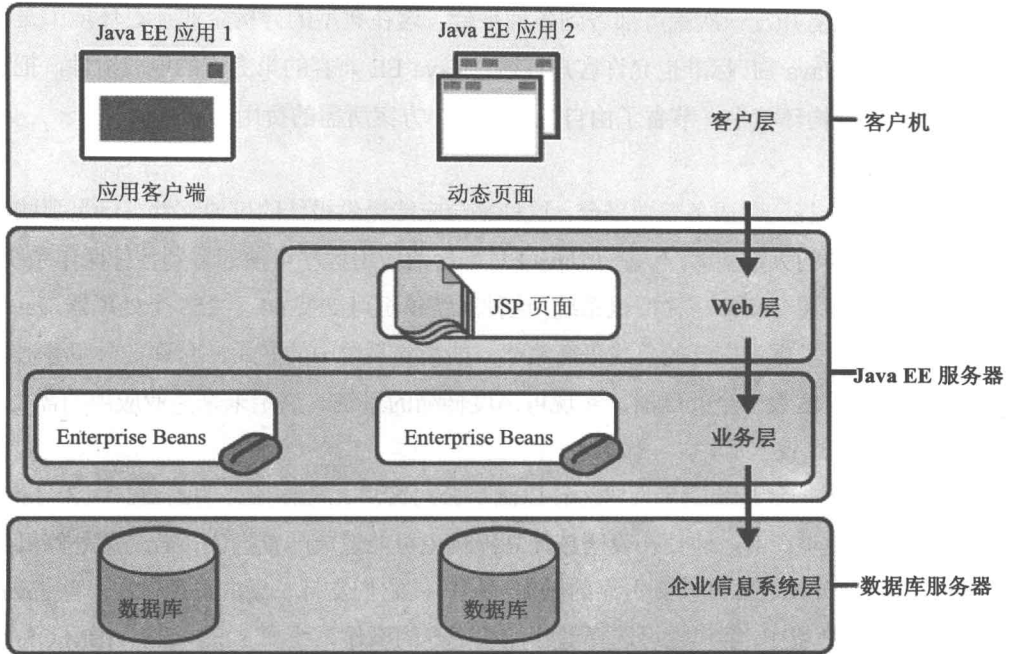


图 1-1 多层应用

1. Java EE 应用程序组件

Java EE 应用程序是由组件构成的。Java EE 组件是具有独立功能的软件单元，它们通过相关的类和文件组装成 Java EE 应用程序，并与其他组件交互。Java EE 说明书中定义了以下的 Java EE 组件。

- 应用客户端程序和 Applets 是客户层组件。
- Java Servlet 和 Java Server Pages(JSP)是 Web 层组件。
- Enterprise JavaBeans(EJB)是业务层组件。

2. 客户层组件

Java EE 的客户端可以是 Web 客户端或应用客户端。

3. Web 层组件

Java EE Web 层组件可以是 JSP 页面或 Servlets。按照 Java EE 规范，静态的 HTML 页面和 Applets 不算是 Web 层组件。

正如图 1-2 所示的客户层那样，Web 层可能包含某些 JavaBean 对象来处理用户输入，并把输入发送给运行在业务层上的 Enterprise Bean 来进行处理。

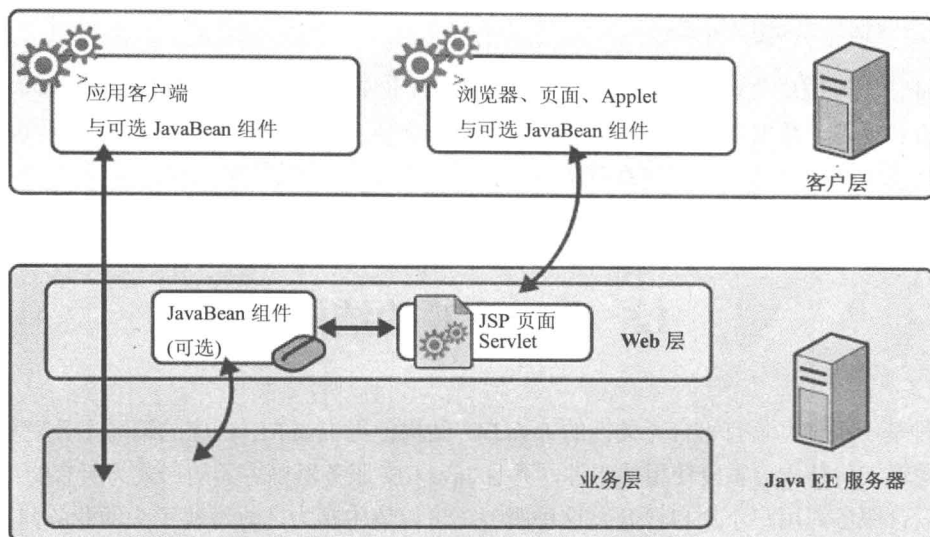


图 1-2 Web 层和 Java EE 应用

4. 业务层组件

业务层代码的逻辑用来满足银行、零售、金融等特殊商务领域的需要，由运行在业务层上的 Enterprise Bean 进行处理。图 1-3 表明了一个 Enterprise Bean 是如何从客户端程序接收数据，并进行处理(如果必要的话)，最后发送到 EIS 层储存的，这个过程也可以逆向进行。

共有三种企业级 Bean: 会话(Session)Beans、实体(Entity)Beans 和消息驱动(Message-Driven) Beans。会话 Bean 表示与客户端程序的临时交互。当客户端程序执行完后，会话 Bean 和相关数据就会消失。相反，实体 Bean 表示数据库表中一行永久的记录，当客户端程序中止或服务关闭时，就会有潜在的服务保证实体 Bean 的数据得以保存。消息驱动 Bean 结合了会话 Bean 和 JMS 的消息监听器的特性，允许一个业务层组件异步接收 JMS 消息。

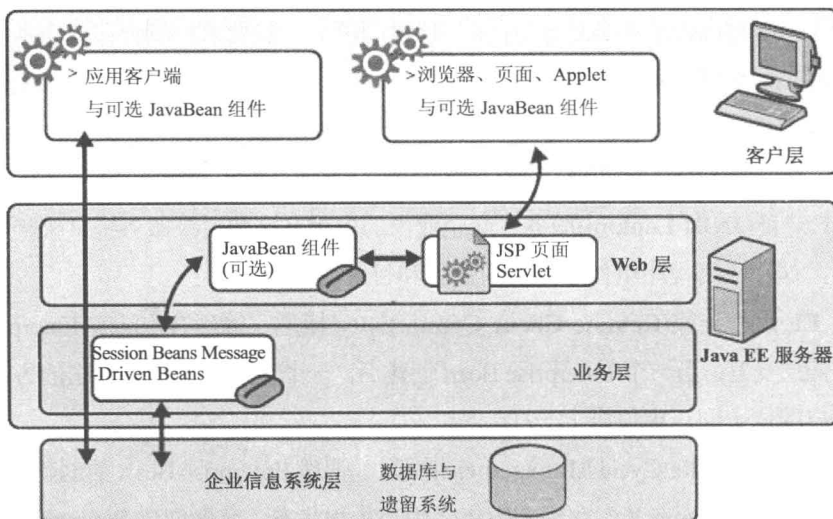


图 1-3 业务层和企业信息系统层

5. 企业信息系统(EIS)层

企业信息系统层处理企业信息系统软件,包括企业基础建设系统、企业资源计划 (ERP)、大型机事务处理、数据库系统和其他的遗留信息系统等。例如,Java EE 应用组件可能为了数据库连接需要而访问企业信息系统。

业务层和企业信息系统层如图 1-3 所示。

1.5 Java EE 的结构

这种基于组件、具有平台无关性的 Java EE 结构使得 Java EE 程序的编写十分简单。因为业务逻辑被封装成可重复使用的组件,并且 Java EE 服务器以容器的形式为所有的组件类型提供后台服务。用户不用自己开发这种服务,可以集中精力,只解决手头的业务问题。

1.5.1 容器服务

容器设置定制了 Java EE 服务器所提供的内在支持,包括安全、事务管理、JNDI(Java Naming and Directory Interface)寻址、远程连接等服务。以下列出最重要的几种服务。

- **Java EE 安全(Security)模型** 可以让我们配置 Web 组件或 Enterprise Bean,这样只有被授权的用户才能访问系统资源。每一客户属于一个特别的角色,而每个角色只允许激活特定的方法。用户在 Enterprise Bean 的布置描述中声明角色和可被激活的方法。由于是这种声明性的方法,因此不必编写加强安全性的规则。
- **Java EE 事务管理(Transaction Management)模型** 让用户指定组成一个事务中所有方法间的关系,这样一个事务中的所有方法被当成一个单一的单元。当客户端激活一个 Enterprise Bean 中的方法时,容器介入一个管理事务。因有容器管理事务,则在 Enterprise Bean 中不必对事务的边界进行编码。一般要求控制分布式事务的代码会非常复杂,但在 Enterprise Bean 中,用户只需在布置描述文件中声明 Enterprise Bean 的事务属性,而不用编写并调试复杂的代码。容器将自动读取此文件,并为用户处理此 Enterprise Bean 的事务。
- **JNDI 寻址(JNDI Lookup)服务** 向企业内的多重名字和目录服务提供了一个统一的接口,这样应用程序组件可以访问名字和目录服务。
- **Java EE 远程连接(Remote Client Connectivity)模型** 管理客户端和 Enterprise Bean 间的低层交互。当一个 Enterprise Bean 创建后,一个客户端可以调用它的方法就像它和客户端位于同一虚拟机上一样。
- **生存周期管理(Life Cycle Management)模型** 管理 Enterprise Bean 的创建和移除。一个 Enterprise Bean 在其生存周期中将会历经几种状态。容器创建 Enterprise Bean,并在可用实例池与活动状态中移动它,而最终将其从容器中移除。即使可以调用

Enterprise Bean 的 create 及 remove 方法，容器也会在后台执行这些任务。

- **数据库连接池(Database Connection Pooling)模型** 该模型是一个有价值的资源。获取数据库连接是一项耗时的工作，而且连接数非常有限。容器通过管理连接池来缓和这些问题。Enterprise Bean 可从池中迅速获取连接。在 Bean 释放连接后，可为其他 Bean 使用。

1.5.2 容器类型

Java EE 应用组件可以安装部署到以下几种容器中去，如图 1-4 所示。

- **EJB 容器** 管理所有 Java EE 应用程序中企业级 Bean 的执行。Enterprise Bean 和它们的容器运行在 Java EE 服务器上。
- **Web 容器** 管理所有 Java EE 应用程序中 JSP 页面和 Servlet 组件的执行。Web 组件和它们的容器运行在 Java EE 服务器上。
- **应用程序客户端容器** 管理所有 Java EE 应用程序中应用程序客户端组件的执行。应用程序客户端和它们的容器运行在 Java EE 服务器上。
- **Applet 容器** 该容器是运行在客户端机器上的 Web 浏览器和 Java 插件的结合。

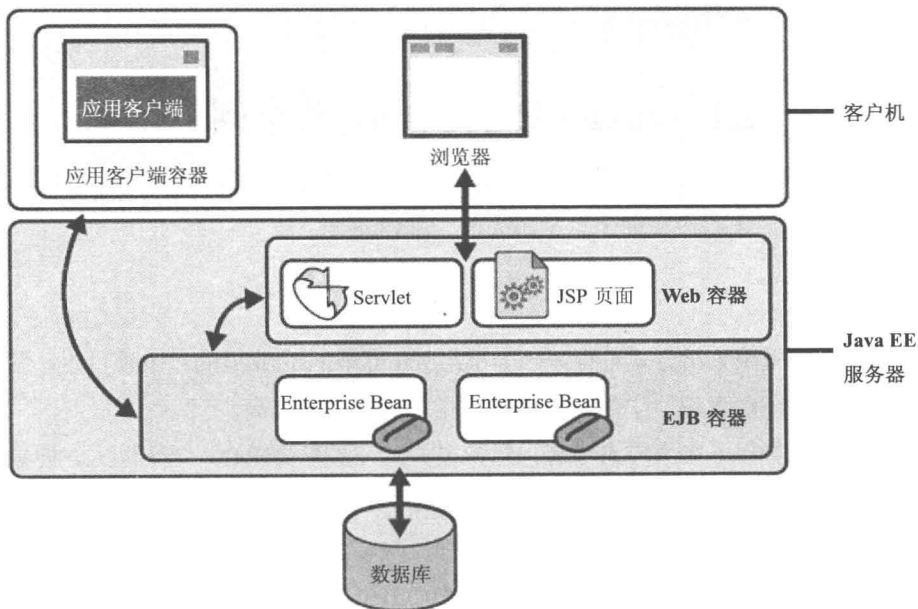


图 1-4 Java EE 服务器和容器