



J2EE

项目开发实用案例

探索J2EE开发精髓
体验敏捷开发实践

陈鹏 程勇 编著



科学出版社
www.sciencep.com

J2EE 项目开发实用案例

陈 鹏 程 勇 编著

科 学 出 版 社

北 京

内 容 简 介

本书汇集了大量企业级应用项目开发的经验,归纳了不同项目中使用的框架、模式、设计方法、开发方法,形成一个完整的技术体系和指导方法,以供用 J2EE 进行项目开发的初级人员和有一定经验的高级开发人员阅读。

本书分为三部分,共 8 章,分别阐述了 J2EE 的技术体系结构、开发环境;以 PetStore 为例,探讨了使用 EJB 架构下的 J2EE 应用的方案;以敏捷的开发方法与架构思想重构第 2 部分的项目。总结了敏捷开发方法、开发规模、节奏和质量,降低开发风险等问题。

本书适合用 J2EE 进行项目开发的初学者阅读,对有 J2EE 开发经验的读者也有一定的参考价值。特别对大学高年级学生、研究生做毕业设计,本书有借鉴之用。

图书在版编目(CIP)数据

J2EE 项目开发实用案例/陈鹏,程勇编著. —北京:科学出版社,2006

ISBN 7-03-017241-8

I. J … II. ①陈… ②程… III. JAVA 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 048869 号

责任编辑:王淑兰 李太铎/责任校对:刘彦妮

责任印制:吕春珉/封面设计:耕者设计工作室

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2006 年 7 月第 一 版 开本:787×1092 1/16

2006 年 7 月第一次印刷 印张:22 1/4

印数:1—4 000 字数:530 000

定价:35.00 元(含光盘)

(如有印装质量问题,我社负责调换<环伟>)

销售部电话 010-62136131 编辑部电话 010-62130750 (H01)

前 言

最近几年是 J2EE 技术创新最多的时期, J2EE 轻量级开发架构的出现给整个 J2EE 开发的体系带来了更多革新性的思想。随着国内企业都面临着巨大的发展和挑战, 如何使用 J2EE 技术快速、可靠地构建大规模企业应用系统已成为大家关注的焦点。因此, 新的轻量级 J2EE 框架应运而生。与传统的重量级 J2EE 架构相比, 轻量级 J2EE 框架具备了快速、高效、易于使用的特点。J2EE 架构向 5.0 这个新的台阶的迈进, 以及近年日益红火的 Struts、Spring、Hiberante 轻量级开发架构也正反映了这一趋势。

本书汇集了作者大量的企业级应用项目开发与实施的体验和开放源码开发人员的实践经验。通过总结这些不同项目中使用到的框架、模式、设计方法、开发方法, 将那些支离的片断归纳升华, 形成一个完整的技术体系和指导方法, 并以图书的形式呈现给读者, 以帮助那些正进行 J2EE 应用开发的初级开发人员学会如何使用这些开发框架, 同时帮助那些已经具备一定开发经验的高级开发人员在整体架构设计上有更大的提升——这就是写这本书的一个初衷。

本书内容分为三个部分: 第 1 部分是叙述性的章节, 主要对 J2EE 的技术体系结构进行一个初步的介绍, 并对后面部分所需要使用的开发环境做了相应的描述。第 2 部分以 J2EE 的经典项目案例 PetStore 为例, 探讨了使用 EJB 架构下的 J2EE 应用实现的具体方案, 整个项目的实现按照软件工程开发的方法, 从需求分析到系统涉及, 由详细设计到具体实现, 最后完成项目的构建、安装与部署。第 3 部分是以敏捷的开发方法与架构思想对第 2 部分的项目进行了重构, 此部分采用了敏捷的 J2EE 架构设计, 引入了基于微内核结构的 J2EE 架构, 更加灵活和清晰地将复杂的 J2EE 应用化繁为简, 使应用开发人员能够从中体会 J2EE 应用开发的更多乐趣。在完成案例重构的同时, 我们还引入了对敏捷开发方法的一些思考, 希望通过敏捷开发方法的指导, 能够更加稳定地控制企业级 J2EE 项目开发的规模、节奏和质量, 以降低企业级应用开发的风险。

如何阅读本书? 这本书能给读者带来什么? 对于刚开始学习 J2EE 技术的读者, 我们建议从本书的第 1 部分开始阅读, 这样能对 J2EE 的整个体系架构有一个完整的认识, 以便循序渐进地把握 J2EE 开发的不同模式与方法。对于具备实际开发经验的高级开发人员, 可以从第 2 部分开始阅读, 通过经典的 J2EE 应用案例逐步深入 EJB 开发, 掌握 EJB 开发的核心设计原则。如果读者已经迫不及待希望看到敏捷 J2EE 技术所带来的新的革命性思想, 那就马上从第 3 部分开始阅读, 并针对企业需求进行一些思考, 归纳和总结出有益于企业自身实际项目开发的 J2EE 应用框架, 帮助企业构建面向快速变化的 IT 应用的基础架构。

致 谢

本书是我们在过去几年工作中参与多个 J2EE 项目和学习相关技术的一个小结，很多的朋友为本书中的内容提供了有益的帮助，在这里，我们要向他们表示衷心的感谢：

首先感谢的是本书作者之一，我的合作伙伴，陈鹏博士，他是我在北京航空航天大学做关于开放源码的校园行活动时认识的好朋友，他为本书的完成做出了巨大的贡献，完成了本书第 1 部分的编写和整本书的规划工作。在坦诚、愉快的沟通与交流中使得这本书设计的表现形式和结构更贴近读者。因为，我们想使本书能够真正引领读者进行实际有效的 J2EE 项目开发。

还要感谢曾经一起工作过的东软移动互联网事业部的同事：徐晓飞、孙毅夫、王胜波、文礼、张峰、郑晔等，是他们帮助我一步步走向成熟。还要特别感谢徐景辉和张春民，他们认真和严谨的工作态度，帮助我们在 J2EE 框架的设计、开发和测试上取得了飞速的进步。

徐娇是我在东软移动互联网事业部工作时的另外一位同事，也是本书的最佳审阅者，她牺牲了大量的业余休息时间校对本书稿，是她的认真与仔细让我们对本书有了更多的信心。

此外，还要感谢与我一直致力于开源 J2EE 应用服务器 JFox 开发的杨泳，以及他的女朋友 Lisa，他们的支持与帮助使得我更有信心与勇气继续努力向前。

最后，感谢我的父母，他们在家乡辛勤的工作才换来我现在的一切，我将永记在心。

程 勇

目 录

第 1 部分 J2EE 的技术体系和开发环境

第 1 章 J2EE 概述	2
1.1 J2EE1.4 体系结构	2
1.2 J2EE 分布式应用	3
1.2.1 J2EE 组件	4
1.2.2 J2EE 客户端	4
1.2.3 Web 组件	5
1.2.4 业务组件	6
1.2.5 企业信息系统层	6
1.2.6 组件打包	7
1.2.7 角色分配	7
1.3 J2EE 核心技术	7
1.3.1 JDBC	8
1.3.2 JNDI	16
1.3.3 RMI	18
1.3.4 JMS	23
1.3.5 EJB	37
1.3.6 Java Servlet	47
1.3.7 JSP	49
1.3.8 JAF	55
1.3.9 JavaMail	55
1.3.10 Java IDL	57
1.3.11 XML	57
1.3.12 JTA	58
1.3.13 JTS	58
1.4 J2EE 5.0 新功能	58
1.5 进一步阅读	59
第 2 章 开发环境	60
2.1 JDK——开发包	60
2.1.1 什么是 JDK	60
2.1.2 JDK 的安装	60
2.1.3 JDK 的使用	64

2.1.4	进一步阅读.....	65
2.2	Eclipse——集成开发环境.....	65
2.2.1	Eclipse 简介.....	65
2.2.2	Eclipse 的安装.....	65
2.2.3	Eclipse 的使用.....	68
2.2.4	进一步阅读.....	71
2.3	JBoss——应用服务器.....	71
2.3.1	JBoss4.0 简介.....	71
2.3.2	JBoss 的安装.....	71
2.3.3	进一步阅读.....	73
2.4	Lomboz——调试工具.....	73
2.4.1	Lomboz 简介.....	73
2.4.2	Lomboz 的安装.....	73
2.4.3	Lomboz 插件的激活.....	76
2.4.4	Lomboz 的配置.....	78
2.4.5	进一步阅读.....	85
2.4.6	jboss400.server 文件.....	85
2.5	Maven——项目构建工具.....	89
2.5.1	什么是 Maven.....	89
2.5.2	Maven 体系结构.....	90
2.5.3	使用 Maven 进行项目管理.....	96
2.5.4	在 Eclipse 中使用 Maven.....	103
2.5.5	新的里程碑——Maven 2.0.....	109
2.5.6	进一步阅读.....	109
2.6	Lunbuild——持续集成服务器.....	110
2.6.1	持续集成.....	110
2.6.2	持续集成工具.....	111
2.6.3	Lunbuild 介绍.....	111
2.6.4	进一步阅读.....	117

第 2 部分 典型的 J2EE 项目案例

第 3 章	Java 宠物店项目需求分析与系统设计.....	120
3.1	需求分析.....	121
3.1.1	业务分析.....	121
3.1.2	用例分析.....	126
3.1.3	模块分析.....	130
3.2	系统设计.....	131
3.2.1	设计策略.....	131

3.2.2	项目约束.....	134
3.2.3	架构设计.....	142
第 4 章	Java 宠物店项目设计与实现.....	145
4.1	控制模块.....	145
4.2	安全模块.....	156
4.3	购物车模块.....	161
4.4	订单处理模块.....	165
4.5	目录管理模块.....	168
4.6	客户管理模块.....	173
第 5 章	Java 宠物店项目构建 安装与运行.....	178
5.1	构建宠物店项目.....	178
5.2	安装与运行.....	179
第 6 章	J2EE 项目风险.....	183
6.1	分析阶段.....	183
6.1.1	阶段风险及规避.....	184
6.1.2	最佳实践.....	184
6.2	设计与开发阶段.....	186
6.2.1	阶段风险及规避.....	186
6.2.2	最佳实践.....	189
6.3	开发生命周期管理阶段.....	191
6.3.1	阶段风险及规避.....	191
6.3.2	最佳实践.....	191
6.4	总结.....	194
6.5	进一步阅读.....	195

第 3 部分 敏捷 J2EE 开发实践

第 7 章	敏捷 J2EE.....	198
7.1	敏捷开发方法.....	198
7.1.1	敏捷宣言.....	199
7.1.2	敏捷方法.....	199
7.1.3	敏捷的思考.....	205
7.1.4	进一步阅读.....	207
7.2	敏捷 J2EE 框架.....	207
7.2.1	插件式体系结构.....	208
7.2.2	微内核.....	211
7.2.3	表现层框架.....	223
7.2.4	持久层框架.....	236
7.2.5	将组件集成到一起.....	247

7.2.6 进一步阅读.....	286
7.3 敏捷开发实践.....	286
7.3.1 最佳实践.....	286
7.3.2 支持平台.....	292
7.3.3 进一步阅读.....	295
第8章 轻量级宠物店系统实现——JPetstore.....	296
8.1 架构设计.....	296
8.1.1 JPetstore 系统架构.....	296
8.1.2 编程模型与包结构.....	297
8.1.3 核心设计模式.....	299
8.2 详细设计与实现.....	301
8.2.1 搜索目录.....	301
8.2.2 浏览种类.....	308
8.2.3 查看产品详细信息.....	311
8.2.4 查看产品条目详细信息.....	314
8.2.5 更新购物车.....	316
8.2.6 创建账户.....	321
8.2.7 更新账户.....	328
8.2.8 系统登录.....	333
8.2.9 提交订单.....	337
8.3 JPetstore 构建 安装与运行.....	346
8.3.1 运行环境配置.....	346
8.3.2 JPetstore 应用部署.....	347
8.3.3 运行 JPetstore.....	348

第 1 部分

J2EE 的技术体系 和开发环境



本部分的内容在全书中的使命是奠定一个技术基础，以及建立相关的开发、运行环境。

第 1 章概述性地介绍了 J2EE 技术，希望读者能够对 J2EE 有一个基本了解，能够掌握 J2EE 这个庞杂体系的技术骨架。如果读者希望深入地掌握 J2EE 知识，建议读者在掌握技术脉络的前提下，再有针对性地学习，补充相关的 J2EE 知识。如果读者对第 1 章介绍的内容已经掌握，那么，就可以跳过这章，直接阅读后面相关的章节。

第 2 章介绍了本书中将要涉及到的各种开发、运行环境，希望读者能够搭建起一整套 J2EE 开发、运行环境，以便进一步地学习。

第1章 J2EE 概述

J2EE 即 Java 2 Enterprise Edition。

如果问 J2EE 到底是什么？可能会获得许多不同的答案。我们认为这个问题并没有一个标准答案，正所谓“横看成岭侧成峰”，对 J2EE 的理解和认识取决于看待 J2EE 的角度。从一个程序员的角度，我们认为：

- ① J2EE 是一种架构，它定义了一种基于组件开发的多层分布式应用架构。
- ② J2EE 是一套体系，它涵盖了十几种核心技术规范，同时将这些技术有机地融为一个体系。
- ③ J2EE 是一个开发平台，它为多层分布式企业级大型应用项目的开发提供应用开发接口（API）。
- ④ J2EE 是一个运行平台，它为 J2EE 项目的运行提供环境及其他标准服务。

参考

Java 的官方网站 (<http://java.sun.com>) 上给出了 J2EE 的描述：“J2EE 定义了一种开发多层企业应用的标准。基于标准化、模块化的组件，通过为这些组件提供一系列的服务以及自动地处理应用中的许多细节问题，J2EE 简化了企业级应用的整个开发过程。”

本章将从体系结构、应用架构、核心技术立体视图熟悉、了解 J2EE。

说明

目前，J2EE 的最新版本已升级到了 J2EE5.0，但考虑到适用的广泛性，我们的介绍内容所依托的主要版本为 J2EE1.4。

1.1 J2EE1.4 体系结构

“不识庐山真面目，只缘身在此山中”，在了解和掌握 J2EE 的过程中，如果我们只将自己深深地陷于 J2EE 所提供的纷繁芜杂的开发接口中时，我们将不能真正掌握 J2EE，更不能游刃有余地使用它。

因此，我们应该高屋建瓴，从体系架构的角度认识 J2EE，如图 1-1 所示。

J2EE 体系结构中包含了多个应用程序组件。J2EE 应用程序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 J2EE 应用程序，并与其他组件交互。在 J2EE 说明书中定义了以下的 J2EE 组件：

- ① 应用客户端程序。应用客户端程序是采用 Java 语言开发、主要在桌面计算机上运行的 GUI 程序。对于用户而言它是一个能够访问 J2EE 中间层服务的本地程序。
- ② Applets。Applets 主要是在 Web 浏览器中运行，同时也能够在其他支持 Applets 编程模型的应用和设备上运行。Applets 为 J2EE 的应用提供强大的用户界面。

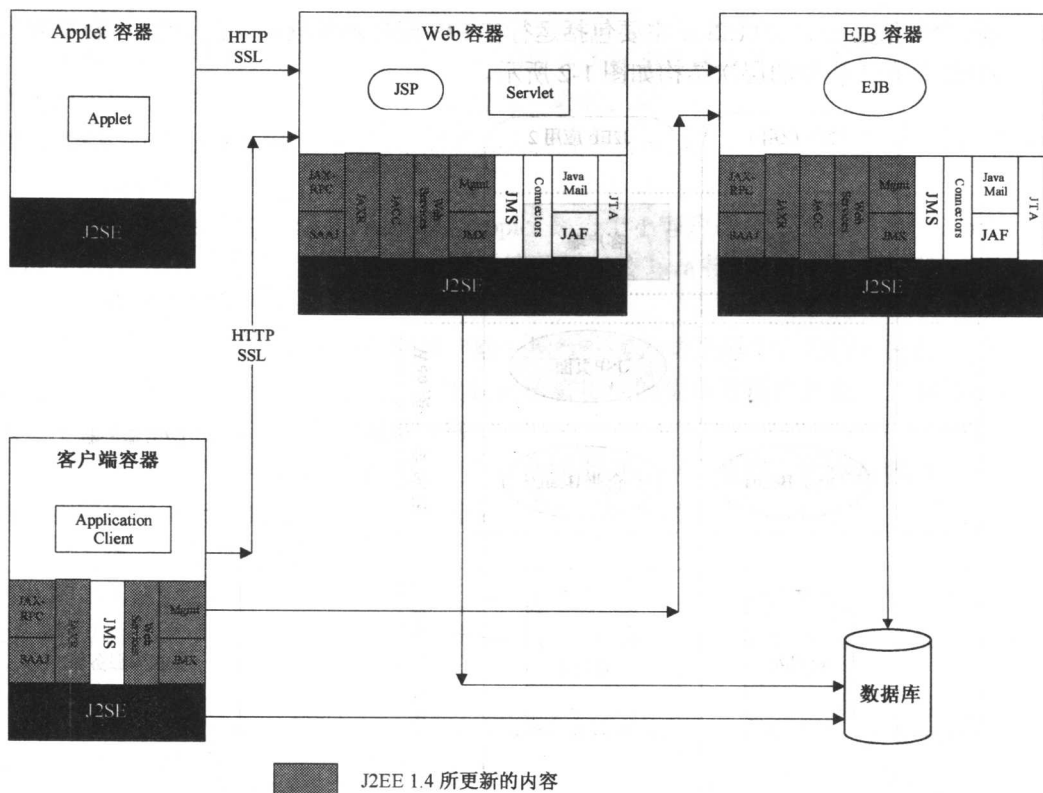


图 1-1 J2EE1.4 体系结构框图

③ Servlets、JSP、过滤器以及 Web 事件监听器。这些组件主要是在 Web 容器中运行，用于响应 Web 客户端的 HTTP 请求。Servlets、JSP 能够产生 HTML、XML 以及其他格式的数据供应用程序组件使用。有种 Servlets 能通过使用 SOAP/HTTP 协议支持 Web Services。Servlets、JSP、Web 过滤器以及 Web 事件监听器合称为 Web 组件。一个 Web 应用由 Web 组件以及其他数据（比如 HTML 页面）组成。

④ 企业 JavaBeans (EJB)。EJB 组件运行在一个支持事务的可管理的容器中。在一个 J2EE 应用程序中，往往将业务逻辑包含到企业 Bean 中。企业 Bean 能够通过 SOAP/HTTP 协议直接提供 Web Services。

1.2 J2EE 分布式应用

基于 J2EE 平台的典型应用应该是多层分布式应用模型。在应用系统架构过程中，依照逻辑功能划分可将应用程序划分成不同的组件，而这些组件能对应于 J2EE 分布式模型中的各不同层。

J2EE 分布式模型主要包括如下四层：

- ① 客户层：主要包括运行在客户计算机上的组件。
- ② Web 层：主要包括运行在 J2EE 服务器上的 Web 组件。
- ③ 业务层：主要包括运行在 J2EE 服务器上的 EJB 组件。

④ 企业信息集成层(EIS): 主要包括运行在 EIS 服务器的软件系统(例如, 数据库)。J2EE 分布式模型的层次结构如图 1-2 所示。

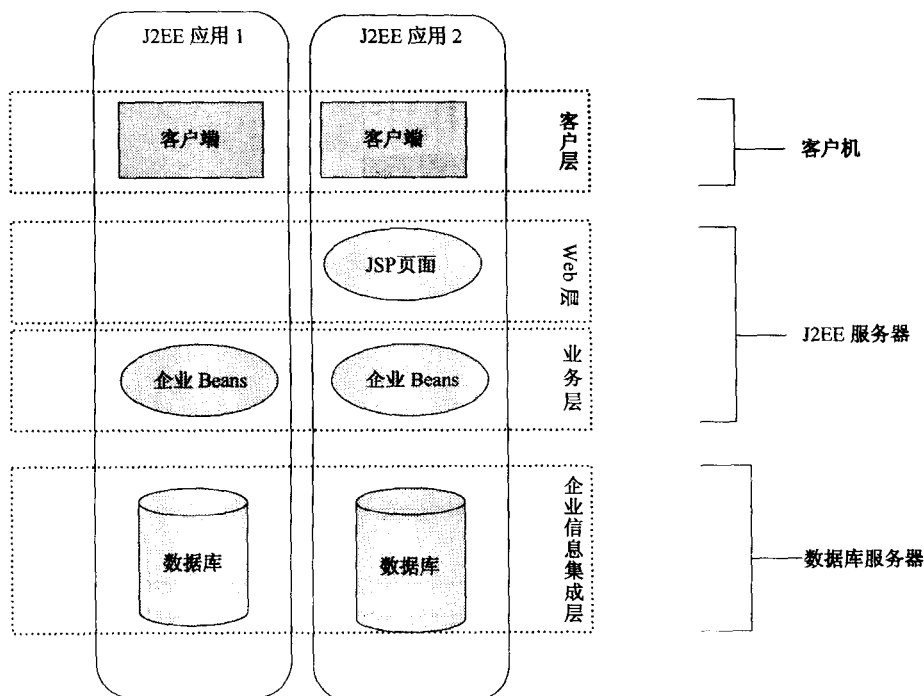


图 1-2 J2EE 分布式模型的层次结构图

J2EE 应用服务器整合 Web 层和业务层。从软件架构的角度, 区别于传统的客户和服务模型, J2EE 应用程序亦称为三层应用结构: 客户计算机、J2EE 服务器以及后台的数据库。

1.2.1 J2EE 组件

J2EE 组件是一个封装了功能的软件单元, 能够与相关的一些类和文件一起组成 J2EE 应用程序, 并且可以与其他组件通信。J2EE 规范对 J2EE 组件定义如下:

- ① 应用程序客户和 Applet 是指运行在客户计算机上的组件。
- ② 基于 Java Servlet 和 JSP 技术的组件叫 Web 组件, 它们可以运行在服务器上。
- ③ 企业 JavaBean (EJB) 组件 (也称为企业 Bean) 叫业务组件, 同样运行在服务器上。

1.2.2 J2EE 客户端

J2EE 客户端可以是一个 Web 客户端, 也可以是一个应用程序客户端。

(1) Web 客户端

Web 客户端包含以下两部分:

- ① 动态 Web 页面以及不同的标记语言 (HTML、XML 等)。动态页面是由 Web 层的 Web 组件产生。

② Web 浏览器。Web 浏览器呈现从服务器端接收到的页面。

Web 客户端也称为瘦客户端。瘦客户端通常不做查询数据库、执行复杂商业规则、连接旧应用程序等工作。当你使用一个瘦客户端时，很多复杂的操作（如上所述）被部署到 J2EE 的业务组件中。

(2) Applets

Web 页面能够包含嵌入的 Applet。Applet 是一个小客户端应用程序。为使 Applet 可以成功的在 Web 浏览器上执行，客户端系统需要安装 Java 插件和一个安全策略文件。

(3) 应用程序客户端

J2EE 应用程序客户端（也称胖客户端）运行在客户端机器中，为用户提供了处理任务的另一种方式。这些任务需要一个比标记语言提供的更丰富用户界面。一般地，它提供一个用 Swing 和 Awt 产生的图形界面。

应用程序客户端直接访问运行在业务层的企业 Beans。然而，如果需要的话，J2EE 应用程序客户端可以打开一个 HTTP 连接和运行在服务器端的 Servlet 建立会话。

(4) 客户端与服务端的通信

图 1-3 显示了组成客户层的不同元素以及客户端和 J2EE 服务器之间的通信示意，例如一个 Web 客户端可以通过访问 JSP 页面或运行在 Web 层的 Servlets 从而实现对业务组件的访问。

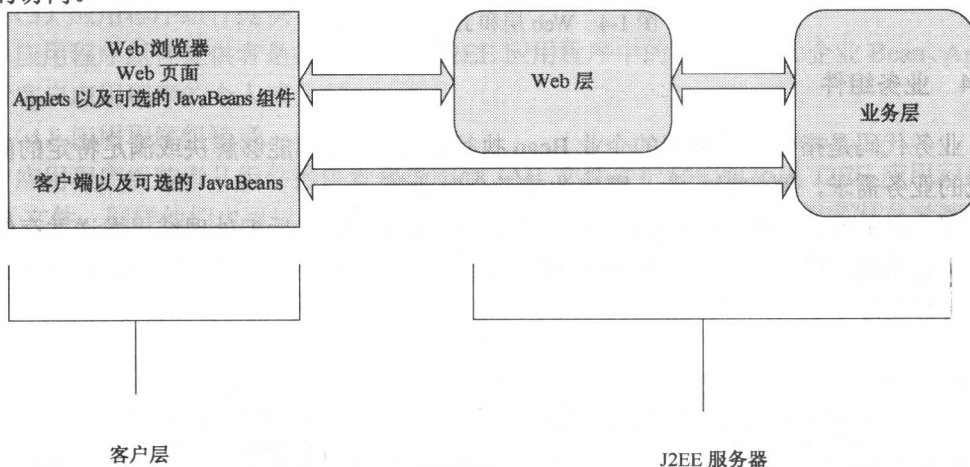


图 1-3 客户端与服务端的通信示意图

在应用系统的架构设计过程中，选择使用瘦客户端（Web 客户端）还是胖客户端取决于对业务功能的部署分配，应该权衡如下因素：

① 瘦客户的优势在于客户端实现简单，应用程序的分布、部署、配置、管理简单。

② 胖客户的优势在于保持了客户端的功能性，使应用程序更亲近用户，有利于察觉用户的体验。

1.2.3 Web 组件

J2EE 的 Web 组件可以是 Servlet 或 JSP。Servlet 是用 Java 语言开发的类，它能动态处理请求并构造响应。JSP 页面则是基于文本的，也能像 Servlet 一样被执行，更多用于

创建一些静态的内容。

图 1-4 显示了 Web 层可能包含的 JavaBean 组件来管理用户输入，把输入发送到运行在业务层上的企业 Bean 中进行处理的过程。

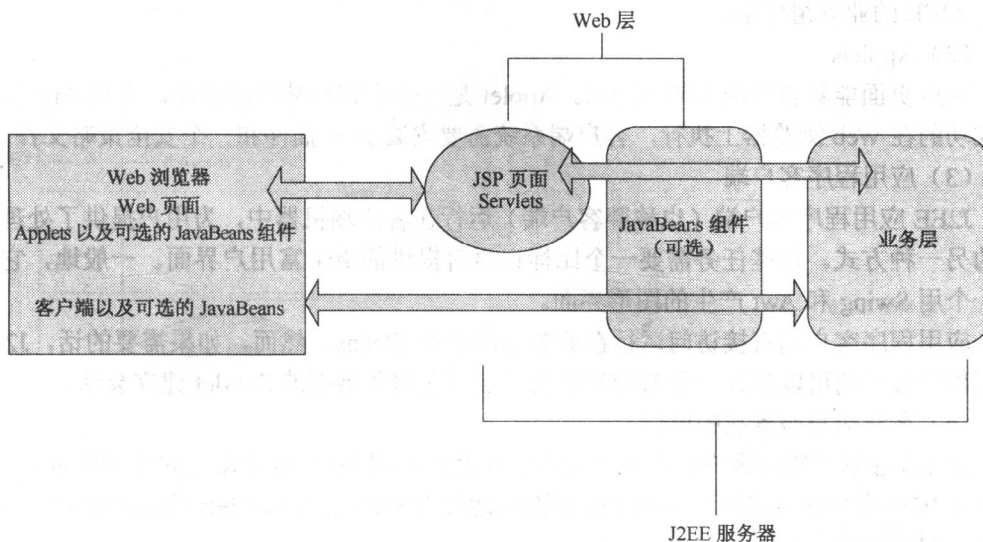


图 1-4 Web 层和 J2EE 应用程序

1.2.4 业务组件

业务代码是指位于业务层的企业 Bean 执行的逻辑，它们能够解决或满足特定的商业领域的业务需求，例如银行业、零售业或者金融业等。

图 1-5 显示了企业 Bean 从客户程序接收数据并处理，然后把处理结果发送并存储到企业信息系统层的过程。企业 Bean 也可以从存储层中读取数据并完成处理，然后把结果发回客户程序中。

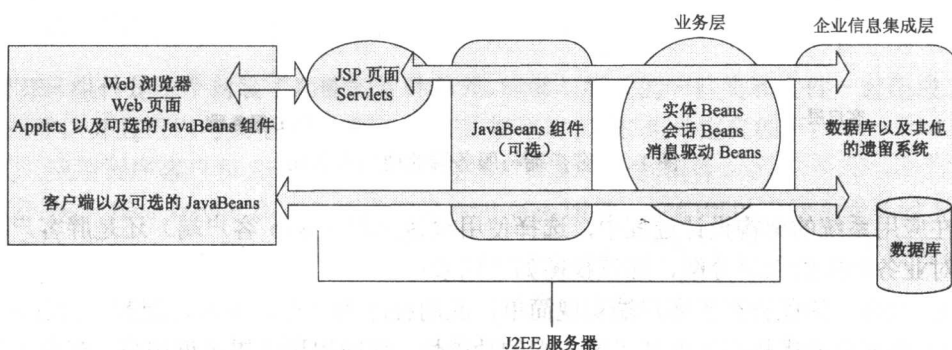


图 1-5 业务层和 EIS 层

1.2.5 企业信息系统层

企业信息系统层包括处理企业信息系统的软件，例如企业资源计划（ERP）、大型机的事务处理、数据库系统以及其他历史上遗留下来的信息系统等。

1.2.6 组件打包

J2EE 组件可以分别打包然后再封装成 J2EE 应用程序来发布。所有组件和其他相关的文件，如 GIF 图片、HTML 页面、服务器端的工具类以及所需的部署描述文件一起组装成一个模块，然后加入 J2EE 应用程序中。J2EE 应用程序是由一个或多个企业 Bean、Web 或者应用程序客户等组件模块构成。

1.2.7 角色分配

采用可复用的模块使应用程序的开发和部署过程可以分配给不同的角色来完成，不同的人或公司可以承担整个开发过程中的不同部分。

在 J2EE 应用开发过程中共有五个角色。

(1) J2EE 产品提供者

J2EE 产品提供者设计并提供可供购买的 J2EE 平台，实现了包括各种相关的 API 以及 J2EE 规范所定义的特征，他们可以是个人或者公司。

(2) 工具提供者

工具提供者是指一些人或者公司，他们创建了开发、组装和打包阶段使用的各种工具并提供给组件开发者、组件组装者以及组件发布者。

(3) 应用程序组件提供者

应用程序组件提供者是指那些创建 J2EE 应用程序中的 Web 组件、企业 Bean、Applet 或者客户应用程序的个人或公司。

(4) 应用程序组装者

应用程序组装者从组件提供者那里收到 JAR 文件并把它们组装成 J2EE 应用程序的 EAR 文件，同样他们也可以是个人或公司。应用程序组装者或是应用程序发布者都可以重新编辑部署描述文件，他们可以直接编辑文件也可以通过一些工具来实现。

(5) 应用程序部署者和管理员

应用程序部署者和管理员是一些个人和公司，他们配置、部署 J2EE 应用程序，并且管理 J2EE 应用程序运行着的基本计算机和网络结构，同是也负责监视 J2EE 应用程序的运行环境。他们的职责包括设置事务控制、安全属性和定义数据库链接等。

1.3 J2EE 核心技术

在了解 J2EE 体系概貌后，我们就需要深入 J2EE 的核心——各种技术规范和细则。真正地掌握 J2EE 需要熟悉 J2EE 的各种核心技术。

J2EE 的核心技术如表 1-1 所示。

表 1-1 J2EE 核心技术

J2EE 技术	描 述
JDBC	Java 数据库连接技术 (Java database connectivity)
JNDI	Java 命名和目录服务接口 (Java naming and directory interface)
RMI	远程过程调用 (remote method invocation)

J2EE 技术	描 述
JMS	Java 消息服务 (Java messaging service)
EJB	企业级 Java Bean (enterprise Java Beans)
Java Servlet	在服务端执行的 Java 小程序
JSP	Java 脚本 (Java server pages)
JavaMail	Java 邮件服务
JAF	JavaBeans 激活架构 (JavaBeans activation framework)
XML	扩展的标记语言(extensible markup language)
JTA/JTS	Java 事务框架/Java 事务服务 (Java transaction architecture / Java transaction service)

1.3.1 JDBC

JDBC 是一种用于执行 SQL 语句的 Java API (JDBC 常被认为是代表 Java 数据库连接 (Java database connectivity))。JDBC 由一组用 Java 编程语言编写的类和接口组成。通过使用 JDBC, Java 应用程序能够执行 SQL 语句, 检索数据库, 更新修改数据库。JDBC API 和 ODBC 一样为开发人员隐藏了不同数据库的不同特性。另外, 由于 JDBC 建立在 Java 的基础上, 因此还提供了数据库存取的平台独立性。

目前 JDBC 的版本已经更新到 JDBC4.0 (参考<http://java.sun.com/products/jdbc/>)。J2EE1.4 体系包括的 JDBC 版本为 JDBC3.0 及 JDBC2.1。在本部分内容中, 主要介绍 JDBC3.0 相关的技术。

1. JDBC 应用模型

JDBC 常用的模型包括两层模型和三层模型。

两层模型按功能可分为客户层和服务层, 如图 1-6 所示。

客户层包括应用程序和一个或多个 JDBC 驱动, 应用程序负责处理表示逻辑、业务逻辑、多指令 (multiple-statement) 事务或分布事务的事务管理、资源管理等。

在两层模型中, 应用程序直接与 JDBC 驱动交互, 包括建立和管理物理连接及对底层数据源进行细节处理。应用程序可以利用数据库的非标准特性实现它的专有功能或进行执行优化。该模型存在着不足, 体现为:

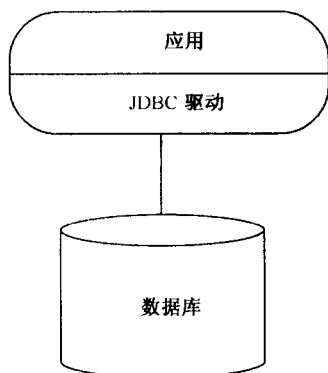


图 1-6 JDBC 两层模型

① 在基础构架和系统级功能上, 混杂了表示逻辑和业务逻辑。这为建立可维护代码带来了障碍。

② 由于针对专门的数据库进行了优化, 使得应用程序缺乏可移植性。应用程序如需连接多种数据库必须考虑不同厂商间实现的差异。

③ 有限的可扩展性。典型的应用程序会对数据库保持一个或多个物理连接, 直到其结束。在两层模型中, JDBC 驱动来负责性能、可伸缩性和可用性问题以及对底层数据源的响应, 这限制了应用程序的可扩展性。

三层模型引入了中间层服务。如图 1-7 所示。