

业内一线专家倾心巨献!

用案例学 Java Web整合开发

——Java+Eclipse+Struts 2+Ajax

雷之宇 郑圣蒲 孙皓 编著

Eclipse		Java	Ajax
Hibernate		全覆盖	
	Oracle		Struts 2
Tomcat	零基础		



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

用案例学 Java Web整合开发

——Java+ Eclipse+ Struts 2+ Ajax

雷之宇 郑圣蒲 孙皓 编著

Eclipse		Java	
Hibernate		全覆盖	
	Oracle		
Tomcat	零基础		

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

Java 作为当前最炙手可热的网络编程语言,已经带动了相关衍生产品的流行。从开发工具 Eclipse + MyEclipse 和 Jbuilder,到服务器端的框架 Struts、Hibernate、Spring 的持续升温,再加上相关的服务器软件 Tomcat、Weblogic 等,Java 已经发展成为一个完整的产品系列。

本书的目标就是带领你逐个了解、熟悉、使用这些 Java 系列的产品,从一个零基础的 Java 菜鸟成为一个合格的 Java 程序员。

本书可以作为菜鸟级读者的入门教程。我们从系统环境的安装、配置、创建工程、调试、编译,到系统的发布和部署,一步步、手把手地教你实现。

而对于有经验的程序员,本书也是一本“企业级案例”的参考手册,本书中的“组织机构和用户管理、权限管理、合同管理、项目管理、工作流引擎、全文检索引擎、数据导出组件”等功能模块可以直接应用在真实的系统中。

最后,希望本书的阅读能够带给你一次愉快的、神奇的 Java 编程体验!

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

用案例学 Java Web 整合开发: Java+Eclipse+Struts 2+Ajax / 雷之宇,郑圣蒲,孙皓编著. —北京:电子工业出版社,2009.4
ISBN 978-7-121-08234-4

I. 用… II. ①雷… ②郑… ③孙… III. ①JAVA 语言—程序设计 ②软件工具—程序设计 IV. TP312 TP311.56

中国版本图书馆 CIP 数据核字(2009)第 015891 号

责任编辑:江 立

印 刷:北京天宇星印刷厂

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:860×1092 1/16 印张:38 字数:957 千字

印 次:2009 年 4 月第 1 次印刷

印 数:4000 册 定价:75.00 元(含光盘 1 张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前言

Java 从 1995 年正式发布，几乎是满载着赞誉一路而来的，受到了数以千万计的程序员的推崇，逐渐发展成为使用最广泛的编程语言之一。

Java 的流行已经带动了相关衍生产品的流行。从开发工具 Eclipse+MyEclipse 和 Jbuilder，到服务器端的框架 Struts、Hibernate、Spring 的持续升温，再加上相关的服务器软件 Tomcat、Weblogic 等，Java 已经发展成为一个完整的产品系列。

Java 所引发的风暴，每一个程序员都有自己亲身的体会。很多软件开发工程师都投入很多时间和精力不断地学习 Java 及其相关产品（Struts、Hibernate、Spring），对于他们而言，这可能意味着未来技术的方向、更多的就业机会和更高的薪酬。

Java 究竟有着什么样的魔力，能拥有这么多的追随者？

“Write once run anywhere（编写一次，随处运行）”也许是最好的答案，但已经不能够代表 Java 的全部。

Java 作为网络系统中最主要的编程语言，除了良好的可移植性外，更值得称道的还有其“开放性”。也正是由于 Java 的开放性，才得以吸引大量的第三方厂商开发出了各种 Java 的衍生产品，最终形成了今天完整的 Java 家族产品系列。

本书讲了些什么

章 节	内容预览
第 1 章	介绍与 Java 相关的概念、与 Java 相关工具的使用以及本书所使用的环境
第 2 章	主要介绍了系统开发环境的安装和配置，并介绍了安装 Oracle 数据库后的一些问题及解决办法
第 3 章	教你如何创建第一个 Struts 工程
第 4 章	介绍了数据库的连接和操作方法，教你如何创建第一个数据库工程
第 5 章	介绍组织机构和用户管理。把公司按职能部门进行划分，并对部门内的员工进行管理
第 6 章	介绍权限管理的几个概念、权限管理的工作原理，以及权限管理的实现
第 7 章	介绍合同分类的管理、合同台账的管理、审批流程的管理，以及合同和项目的关联管理
第 8 章	以“业务审批系统”为实例，讲述了 workflow 引擎的工作原理及 workflow 管理系统的实现过程。workflow 管理系统不仅能够用于业务审批的工作，同样可以用于其他需要审批流程化的办公系统

第 9 章	以项目管理子系统这个实例介绍了信息的管理，信息的管理主要是信息的录入、修改、删除、查看等操作。在项目基本信息管理模块中，基于 Struts 的 MVC 模式详细阐述了信息录入的具体操作
第 10 章	介绍全文检索的原理及如何利用 Lucene 实现全文检索
第 11 章	介绍传统的绘图技术，绘制图形组件的各种方法以及组件使用中碰到的一些问题及解决办法，以及定式报表和自定义报表的技术实现过程以及效果
第 12 章	教你如何通过调用 Java 方法实现数据的导出和页面功能的打印
第 13 章	介绍系统的安装与部署，其中介绍了部署时经常发生的错误及发生的原因和如何避免这些错误
第 14 章	介绍安全解决方案，教你如何解密加密文件，以及如何加强对自身电脑网络系统的安全防护

本书每个章节都是一个独立的小案例，针对每个案例，我们通过原理介绍、设计思路、知识点、语法、核心代码一步步展现给读者，让读者真正理解并掌握每个案例。

谁需要这本书

本书是一本有别于普通实例教学的参考书，作者希望通过具体的实例，让 Java 应用的开发过程更简单。无论你是刚开始学习 Java 的菜鸟或是一个经验丰富的资深 Java 程序员，都可以成为本书的读者。本书能够帮助前者尽快上手，并能够帮助后者缩短开发周期。

尽管作者和出版社可以凭空臆想每一个程序员朋友都可能是本书的读者，但从营销管理学的角度来看，“一个产品的市场定位如果是所有人，这就意味着这个产品没有市场定位”。所以，我们有必要对本书面向的读者进行一个定义：

1. 程序员（有一些编程经验，Windows 或 Web 均可）；
2. 工作在第一线，以解决问题或完成任务为目的的程序员朋友；
3. 没有任何 Java 编程经验的读者。

本书的读者应该是对 Java 有所了解，但没有任何 Java 编程经验的程序员。

本书强调读者对象的零基础、注重过程的描述。手把手帮助初学者一步步从系统环境、开发环境的搭建，安装、配置、创建数据库，程序编码、测试、发布、安装、部署逐一进行详细的描述，最终完成一个构架先进、设计缜密、功能完善的信息系统。

在这个过程中，本书结合目前流行的 JDK 1.5 + Struts 2.0 + Hibernate 进行整合开发，使用 Eclipse 3.1 作为开发工具，使用 Tomcat 5.5 作为应用服务器，帮助读者在实现功能的同时熟悉这些工具。

对于初学者来说，本书可以作为你的 Java 实践教材，带你领略 Java 技术的魅力，帮你实践那些激动人心的功能。

而对于那些有经验的 Java 程序员，本书中关于工作流引擎、全文检索、Tree view 等技术的实现，以及组织机构、用户、权限、合同、项目、统计、安全管理等真实应用案例的代码解析，相信可以对你的工作提供一些参考价值。

本书每个章节都是一个独立的、成熟的案例。针对每个案例，我们通过原理介绍、设计思路、知

识点、语法、核心代码、常见问题解答一步步抽丝剥茧般地展现给读者，让读者真正理解、掌握每一个案例。

在本书出版之际，我想将它献给我的好朋友：曾经和我一起工作、奋斗、成长并不断给我支持和鼓励的马文汀、张育、夏雨春。

想起当初大家一起在小茶馆中规划着公司（北京杰软科技有限公司）的未来，那场景着实让人怀念。现在张育已经远赴美国定居，马文汀、夏雨春也有了各自的发展。我想他们看到本书的出版也会开心不已。

在本书编撰之初，正值盛夏，在没有中央空调的周末，和几个同事在办公室里挥汗如雨，而当本书终稿确认时，已是数九寒天（2009 年年初），大家都穿上了臃肿的羽绒服，准备去八达岭滑雪，想想时间过得真快，又是一年了。

本书的另外两位作者：郑圣蒲（北京杰软科技有限公司政府事业部经理）、孙皓（Oracle 中国高级顾问）参与编写了本书的大部分章节。郑圣蒲负责全书的编排和整理，孙皓则对全书的内容进行校对和编审，尤其是针对企业级 Oracle 数据库的设计和程序优化实现给出了很多建设性的意见，在这里表示感谢。

参与本书编撰的还有我的同事：马霞、杨子华、陈湘萍、陈舟、张志宁、王峰林、杨方方、张松箭、宁兆航、肖康，他们的敬业精神和责任心让我钦佩（每次收到他们的邮件都是在深夜发出的，这很让我感动）。没有他们的全力配合，本书不可能完成。

特别要感谢的是电子工业出版社博文视点资讯有限公司的李冰小姐和江立小姐，她们本着严谨、负责的态度，对本书提出很多修改建议，并对书稿逐字逐句地进行编审，为本书的出版付出了辛勤的劳动。

最后，虽然作者在本书的编著过程中力求完美，但由于能力所限，疏漏和错误在所难免，请广大读者不吝赐教。

编 者
2009 年 1 月

目 录

第 1 章 引言	1	第 3 章 创建第一个 Struts 工程 “Hello World”	31
1.1 和 Java 相关的几个概念	2	3.1 创建一个新的 Java Web 项目	31
1.1.1 什么是 Java	2	3.1.1 创建 Web 工程	31
1.1.2 JSP、Java、JavaBean 与 JavaScript	3	3.1.2 编辑 Profile 配置文件	37
1.1.3 MVC 与 Struts	4	3.1.3 增加 Struts 功能	38
1.2 和 Java 相关的工具介绍	5	3.1.4 增加 Hibernate 功能	41
1.2.1 JDK 1.6	5	3.1.5 配置 Formbean 和 Action 操作	45
1.2.2 Tomcat	6	3.1.6 配置 Eclipse 和 Tomcat 的连接	45
1.2.3 BEA WebLogic	7	3.1.7 编译和运行	47
1.2.4 Oracle iAS	7	3.2 代码详解	48
1.2.5 Eclipse 和 MyEclipse	8	3.3 常见问题经验谈	50
1.2.6 JBuilder	9	3.4 本章小结	50
1.3 本书所使用的环境	9	第 4 章 创建第一个数据库工程——“BookManage”	51
1.4 本章小结	10	4.1 数据库配置	51
第 2 章 搭建开发系统环境	11	4.1.1 配置表空间	52
2.1 安装和配置 JDK 1.5	11	4.1.2 创建用户	54
2.1.1 安装 JDK 1.5	11	4.1.3 创建表（用 UI 界面或 Script）	55
2.1.2 设置环境变量	13	4.1.4 手工添加几条数据	57
2.2 安装和配置 Tomcat 5	14	4.1.5 SQL Plus 的介绍	57
2.2.1 安装 Tomcat 5	15	4.2 第一次用 Java 访问数据库	62
2.2.2 如何启动及关闭 Tomcat	17	4.2.1 JDBC 技术基础	63
2.3 安装 MyEclipse	17	4.2.2 DBHandle 组件介绍	68
2.4 安装和配置 Oracle 10g 数据库	20	4.3 创建数据库工程——“BookManage”	71
2.4.1 安装 Oracle 10g 数据库	20	4.3.1 向数据库中插入新的数据	72
2.4.2 创建数据库实例	23	4.3.2 在页面中显示数据列表	78
2.4.3 配置监听器（Listener）	28	4.3.3 实现删除的功能（支持复选）	82
2.5 常见问题经验谈	29	4.4 常见问题经验谈	84
2.5.1 Oracle 数据库无法启动的几种情况	29		
2.5.2 Oracle 安装后 Eclipse 无法启动	30		
2.6 本章小结	30		

4.5 本章小结	85	6.3.3 权限元素分配管理	269
第 5 章 组织机构和用户管理	86	6.3.4 权限元素验证	294
5.1 功能描述	86	6.4 常见问题经验谈	307
5.2 TreeView 基础	86	6.5 本章小结	307
5.2.1 几种常见的 TreeView 实现方法	87	第 7 章 合同管理子系统	308
5.2.2 用 XML 实现 (优点、缺点)	95	7.1 合同分类的管理 (树形结构)	308
5.2.3 用 JavaScript 动态获取下一级 节点 (优点、缺点)	95	7.1.1 数据结构	309
5.2.4 TreeView 的数据结构	95	7.1.2 合同分类树的显示组件 (XML)	309
5.2.5 TreeView 组件介绍	96	7.1.3 树节点的管理	311
5.3 实现组织机构树的管理	110	7.2 合同台账 (查询统计) 的管理	320
5.3.1 数据结构	110	7.2.1 数据结构详解 (ER 图)	321
5.3.2 组织机构树的展示	114	7.2.2 合同基本信息的管理	322
5.3.3 添加树节点	134	7.2.3 客户管理	338
5.3.4 修改树节点	142	7.2.4 合同的查询和统计	338
5.3.5 删除树节点	148	7.2.5 审批流程的管理	339
5.4 实现用户的管理	153	7.3 合同和项目的关联管理	339
5.4.1 数据结构	154	7.4 常见问题经验谈	340
5.4.2 显示节点下的用户列表	155	7.5 本章小结	340
5.4.3 添加用户	164	第 8 章 Workflow 工作流引擎	341
5.4.4 删除用户	178	8.1 工作流基础	341
5.4.5 用户转移部门	185	8.1.1 工作流的概念	341
5.5 常见问题经验谈	195	8.1.2 为什么要使用工作流	347
5.6 本章小结	196	8.1.3 常见的工作流实现方式	348
第 6 章 权限管理	197	8.2 工作流实现原理	349
6.1 权限管理的几个概念	197	8.2.1 Workflow 引擎设计综述 (WF、 Node、Instance、History)	349
6.1.1 用户组	197	8.2.2 Workflow 引擎所需的数据结构	354
6.1.2 权限元素与角色	197	8.3 工作流代码解析	356
6.1.3 授权与赋权	197	8.3.1 Workflow 和 Node 的定义	356
6.1.4 横向权限与纵向权限	198	8.3.2 为 Node 绑定角色	372
6.2 理解权限管理的工作原理	198	8.3.3 本部门领导的处理	374
6.2.1 数据结构	198	8.3.4 工作流执行情况的监控 HistoryRecord 类	376
6.2.2 权限校验组件 (HaveRight) 原理	202	8.3.5 Workflow 执行	379
6.3 权限管理的实现	203	8.4 工作流实例: “合同申请”	393
6.3.1 权限元素的管理	203	8.4.1 填写合同的基本信息	393
6.3.2 角色的管理	239		

8.4.2 信息确认, 并创建工作流实例	394	10.4 常见问题经验谈	445
8.4.3 显示“原型表单”并进行审批 (同意、驳回)	396	10.4.1 中文分词问题	445
8.4.4 添加附件	397	10.4.2 超大/加密文件的处理	445
8.4.5 查看历史审批记录	397	10.5 本章小结	446
8.4.6 提交至下一个节点, 并在首页 提醒	398	第 11 章 综合统计汇总子系统	447
8.5 常见问题经验谈	398	11.1 功能描述	447
8.6 本章小结	399	11.2 绘图组件介绍	447
第 9 章 项目管理子系统	400	11.2.1 绘图组件简介	448
9.1 项目基本信息管理	400	11.2.2 组件安装配置以及注意事项	456
9.1.1 数据结构	400	11.2.3 绘制柱状图	457
9.1.2 项目信息管理	401	11.2.4 绘制饼图	461
9.1.3 项目相关附件管理	413	11.2.5 绘制折线图/点线图	466
9.2 项目过程管理	424	11.3 定式统计汇总	485
9.2.1 任务下达	424	11.3.1 几种常见的合同统计报表	485
9.2.2 实施大纲	424	11.3.2 统计报表的技术实现	486
9.2.3 项目执行与进度管理	425	11.3.3 为统计报表增加图形表现形式	495
9.2.4 验收申办、项目归档	425	11.3.4 定式统计报表经验谈	497
9.2.5 项目的查询和统计	426	11.4 自定义统计汇总	497
9.3 常见问题经验谈	426	11.4.1 什么是自定义报表	497
9.4 本章小结	427	11.4.2 三步完成自定义报表	497
第 10 章 利用 Lucene 实现全文检索	428	11.4.3 自定义报表的实现	498
10.1 功能描述(为项目的附件增加 全文检索功能)	428	11.4.4 自定义统计报表经验谈	521
10.2 全文检索的原理介绍	429	11.5 常见问题经验谈	521
10.2.1 索引项	429	11.6 本章小结	522
10.2.2 全文检索评价标准	429	第 12 章 数据的导出和打印	523
10.2.3 全文检索基本模型	430	12.1 统计报表的打印	523
10.3 Lucene 介绍	430	12.1.1 报表模板介绍(水科院 WF、 广电报表)	523
10.3.1 Lucene 概述	430	12.1.2 用 HTML 及 Script 实现报表的 打印	524
10.3.2 中文分词介绍(中科院分词 ICTCLAS)	432	12.1.3 页面保存、预览、打印代码介绍	525
10.3.3 为文件建立索引	437	12.2 统计数据导出 Excel 文件	528
10.3.4 检索过程的实现	442	12.2.1 两种方法导出成为 Excel 文件 (优缺点比较)	528
10.3.5 其他(字体加亮显示、链接等)	445	12.2.2 页面导出法(整个页面、部分 表格)	532

12.2.3 用 Java 程序导出后下载	534	第 14 章 安全解决方案	560
12.3 格式文本生成 PDF 文件	538	14.1 软件部分	560
12.3.1 PDF 组件简介	538	14.1.1 系统软件	560
12.3.2 添加格式文本	543	14.1.2 应用软件	565
12.3.3 插入图片	544	14.1.3 主动安全	567
12.3.4 添加表格	545	14.1.4 被动安全	582
12.4 本章小结	548	14.2 硬件部分	583
第 13 章 系统的安装与部署	549	14.2.1 RAID5 磁盘阵列	583
13.1 安装部署前的准备工作	549	14.2.2 Cluster 集群服务	584
13.1.1 需要准备的系统软件清单	549	14.2.3 应急预案系统	584
13.1.2 对应用系统、数据库进行备份	549	14.3 网络部分	586
13.1.3 其他	550	14.3.1 CA 安全身份确认 (USB Key)	586
13.2 搭建系统软件/环境	550	14.3.2 SSL 网络安全协议	587
13.3 应用系统的安装与部署	550	14.4 管理与制度	588
13.4 部署自动备份任务	555	14.5 本章小结	588
13.5 系统常用的安全设置	557	附录 1 解决方案的设计原则	589
13.6 常见问题经验谈	559	附录 2 系统的体系结构	592

引言

Java 语言（简称 Java）诞生于 20 世纪 90 年代中期，在过去的十多年里，Java 像一个耐力持久的慢热型长跑选手，越跑越快，在计算机软件领域里不断掀起一次次的技术革命，最终使它一跃成为在万维网（WWW）上开发软件的主要工具。可以说，Java 语言的发展过程就是 IT 技术在一个时间段内发展历程的缩影，Java 技术以各种形式出现、发展，甚至对未来 IT 产业的技术发展方向也产生一定的影响。

现在，Java 已经成为使用最广泛、最主流的网络编程语言。有的 IT 媒体甚至称“不知道 Java 的程序员可能比在北京生活却没有坐过地铁的人还少”，还有人说“精通 Java 的程序员往往意味着优厚的薪酬待遇和令人尊敬的技术专家的头衔”。虽然媒体的说辞略显夸张，但 Java 作为软件开发的一种革命性的技术，其地位已被确立。

想要真正成为一位精通 Java 的高级程序员却并不容易。因为今天的 Java 已经不是 10 年前我们所说的那个由 Sun 公司发布的 JDK 软件包，Java 已经成为一个软件产品系列的代名词。

Java 的风潮已经带动了相关衍生产品的流行，从开发工具 Eclipse+MyEclipse 和 JBuilder 到服务器端的框架 Struts、Hibernate、Spring 的持续升温，再加上相关的服务器软件 Tomcat、Weblogic、Websphere 等，Java 已经发展成为一个完整的产品系列。

本书的目标就是带领你了解、熟悉、使用这些 Java 系列的产品，从一个零基础的 Java 菜鸟成为一个真正的 Java 程序员。

事实上，从 Java 诞生之日起，介绍 Java 的图书和资料就没有缺少过。随着 Struts 和 Hibernate 的流行以及 Eclipse 的逐渐普及，相关的书籍也是层出不穷。书店里介绍 Java 及其相关技术的图书往往摆满了好几个书架，并放在醒目的位置。尽管如此，我们公司刚毕业的新同事却总抱怨找不到一本合适、实用的入门教材，原因何在？

1. 术业有专攻

Java 相关技术的涉及面很广，例如：Struts、Hibernate 仔细研究起来都可以专门作为一门课程来介绍，而图书的作者往往都是某一个领域的专家，因此 Java 的相关图书按照专题介绍的不少，而面面俱到的综合类图书则较少。

2. 原理与实例

由于 Java 技术的多样性，它的每一种相关技术的产生、发展的过程及其原理都是相当复杂的。理解和掌握这些原理，对程序员进行深入的高级编程无疑是大有裨益的，因此很多图书的提纲选排都是以介绍知识点、讲解原理为主，辅以实例，帮助读者在理解的基础上做些小练习加以实践。

这样的选排虽然完美，但是对于一些没有基础的程序员来说，他可能连 Eclipse 的环境变量都不会配置，那些练习根本无从实践。而那些急于上手的初级程序员则可能会有“远水难救近火”的困惑。

除去 Java 技术本身的复杂性不说，由于 Java 良好的开放性，它的每一种相关技术都是由不同的企业或机构开发并提供的。把这么多各不相同的技术搭建、配置、整合在一起，也是一个细致而复杂的过程。因此多数 Java 图书对读者的基础都有一定的要求。

本书针对这些问题，结合目前主流的 JDK+Struts+Hibernate+Ajax 组合方式，配合开发工具 Eclipse+MyEclipse，手把手帮助初学者一步步从系统环境、开发环境的搭建，安装、配置、创建数据库，程序编码、测试、发布、安装、部署逐一进行详细的描述，最终完成一个构架先进、设计缜密、功能完善的信息系统。甚至对系统的自动备份、安全防护等实用编程技巧也涵盖在本书的范围中。

本书不是一本内容全面的 Java 教材，也不完全属于实例类的参考书，如果硬要将本书归类的话，应该算是“工程实践类”的图书。因此本书的目的不是要教会读者 Java 的相关技术及工具，而是带领读者使用这些工具，在一个软件开发系统中，充分应用这些 Java 技术，并从头到尾地完成一个软件开发的全过程。

这个类别的图书有一个特点就是“轻理论、重实践”。

从现实意义的角度来讲，最终客户（软件的需求提供方）往往并不关心程序员所使用的编程语言及系统构架是什么，采用了多少先进的技术，他最关心的问题依次是：

1. 技术上是否能完全实现我的需求（功能和性能两个方面的要求）；
2. 技术的稳定性（新技术往往稳定性较差，成本也较高）；
3. 综合成本（开发成本与维护成本）；
4. 技术的领先性（能够确保在一段时间内不被淘汰）。

因此，本书将以客户的需求为导向，以实践和具体操作过程为主线，抽丝剥茧般逐步向你展示 Java 系列技术的魅力和价值所在。帮助你在最短的时间内，尽快学会 Java Web 编程。

1.1 和 Java 相关的几个概念

1.1.1 什么是 Java

Java 到底是一种什么样的语言呢？Java 是一种简单的、面象对象的、分布式的、解释的、健壮的、安全的、结构中立的、可移植的、性能优异的、多线程的、动态的语言。

这句话太长，虽然它几乎包含了 Java 的所有特征，但长得让所有人都记不住这些特征。我认为 Java 作为一种编程语言从 1995 年诞生发展至今，其最重要的特性是“可移植性”。

Sun 公司的一句口号“Write once, run anywhere（编写一次，到处运行）”成为 Java 最不可替代的价值标识，开始对微软的系列编程语言（C++、VB 等）发起挑战，并发展成为今天网络世界中首选的

编程语言。

Java 诞生之初，是作为嵌入式芯片的编程环境存在的，时至今日 Java 已经拥有多个分支（变种）并在其各自的领域内得到了广泛的发展。JSP、Java、JavaBean、JavaScript 每一个名字我们都耳熟能详，下面我们逐个了解一下这些成员。

1.1.2 JSP、Java、JavaBean 与 JavaScript

JSP

JSP (Java Server Pages) 是由 Sun MicroSystem 公司在 Java 语言上开发出来的一种动态网页制作技术，其能够使你在 HTML 页面中插入 Java 程序实现动态页面，并且支持和 HTML 的混合编写，不需要编译直接执行。你可以使用普通的文本编辑工具来书写 HTML 语句，然后将动态部分用特殊的标记嵌入 Java 程序即可。

Sun MicroSystem 公司首先推出 Servlet，其功能强大，体系设计也很先进，但是其输出 HTML 语句的方式仍然沿用了传统 CGI 程序的做法：用命令逐行输出。这使得在 Servlet 动态生成 HTML 比较麻烦，因此微软的 ASP 一度成为动态网页编程的首选语言。

这促使 Sun MicroSystem 很快推出了类 ASP 的 Java 嵌套型语言：JSP。

JSP 作为一种全新的网页编程技术，不仅提供了一种方便有效的动态网页制作方法，而且作为 Java 家族的一分子，它还具有一般 Java 应用良好的可移植性，这使得 JSP 页面可以在任何支持 Java 的应用服务器中运行。

JSP 语言通过在 HTML 语法中镶嵌 JSP 的 Tag 标签，从而大大简化和方便了网页的设计和修改，这使得源于 Java 的 JSP 语言很快地被广泛应用。然而对于多数大型 Web 应用程序的开发仍然需要 Servlet 和 JSP 搭配使用。

Java

Java 是介于解释型和编译型的一种语言。用 Java 编写的程序都先被编译成字节码，再通过 JVM (Java 虚拟机) 解释执行。由于依赖解释执行的 Java 字节码在运行速度上比可以直接编译成机器码的语言（如 C/C++）缓慢得多，所以大多数用户认为 Java 程序运行较慢。

字节码是造成 Java 运行速度缓慢的主要原因，但是它也是让 Java 可以跨越计算机平台使用的重要因素之一。

字节码是 Java 专有的一种中间码，必须通过 JVM 来解释才能运行。在不同的操作平台上，Java 有不同的 JVM，使 Java 字节码可以跨越平台的限制，在不同的环境上运行。Java 当初在设计时便考虑到跨平台的问题，因此字节码的产生正是为了达到跨平台的功能，这是 Java 有别于其他语言的主要原因。

然而在性能方面显而易见的缺陷，又使得 Java 的竞争力有些削弱。因此，人们在提高 Java 性能方面所作的工作从未停止过。

Sun 公司及 Java 的支持者们为提高 Java 的运行速度已经做出了许多努力，其中大多数集中在程序设计的方法和模式选择方面。由于算法和设计模式的优化是通用的，对 Java 有效的优化算法和设计模式，对其他编译语言也基本适用，因此不能从根本上改变 Java 程序与编译型语言在执行效率方面的差异。

Java 语言有多种表现形式, Java Applet、Java Application、Java Bean、JSP 等, 分别应用在不同的体系结构中, 例如: Java Applet (也称 Java 小程序) 用于开发基于浏览器的小程序, 以提高 HTML 页面的高可交互性, 在这一点上它和 JavaScript 存在的目的是一致的。

JavaBean

ASP 通过 COM 来扩充复杂的功能, 如文件上载、发送 E-mail 以及将业务处理或者复杂计算分离出来成为独立可重复利用的模块。JSP 则通过 JavaBean 完成同样的功能扩充。

本书中的大部分组件都是通过 JavaBean 的方式来提供的。读者通过简单的调用就可以完成各种强大、复杂的功能。

在开发的复杂程度上, COM 的开发远比 JavaBean 复杂和烦琐, 学会 ASP 不难, 但学会开发 COM 却不容易。相比较而言, 使用 JavaBean 对系统功能进行扩充则较容易实现。而且在后期维护的时候, COM 组件必须在服务器上注册后方可使用, 如果对 COM 组件进行了更新, 就必须在注册表中对组件进行重新注册。更新 COM 组件的注册过程, 对很多程序员来说都是一段不愉快的经历。

所幸的是 JavaBean 并不需要注册, 只要将其放置在 CLASSPATH 所包含的目录中就行了。对 JavaBean 进行了修改和更新, 对绝大多数应用服务器 (如 JSWDK 和 Tomcat) 来说, 需要重新启动服务, 就可以载入最新版的组件。

JavaBean 是完全的面向对象程序 (OOP), 可以针对不同的业务处理功能方便地建立一整套可重复利用的对象库, 例如用户权限控制、E-mail 自动回复等。

JavaScript

首先需要明确的是: JavaScript 不是 Java 大家庭的成员。

我们在这里对 JavaScript 进行介绍的主要原因是:

1. JavaScript 在 B/S 结构的浏览器中被广泛应用;
2. 很多初学者将 Java 和 JavaScript 混淆不清。

Applet、Servlet、JSP 和 JaveBean 都是 Java 在不同环境中的表现形式, 尽管它们对标准 Java 的支持程度各有不同, 但它们都是 Java 的一种分支, 秉承着 Java 的全部特性, 支持同样的标准, 共同由 Java 虚拟机 JVM 解释执行。

而 JavaScript 除了在编程语法上和 Java 的书写格式有很大的相似性, 以及命名中包含了 Java 字样外, 和 Java 并无关系。

JavaScript 是一种脚本语言, 它无法编译成字节码, 自然也不需要 JVM 的支持, 是由浏览器对其解释执行的。

1.1.3 MVC 与 Struts

MVC 是 Model (模型) /View (视图) /Controller (控制) 的缩写。在我们真正了解 MVC 之前, 我们先来看看为什么需要使用 MVC 构架, 以及 MVC 能给我们带来哪些令人激动的新特性。

在 MVC 的概念深入人心和 Struts 广为流传之前, 大部分 Web 应用程序都是用 ASP、PHP、JSP 这样的过程化语言来创建的。它们将从数据库中检索出来的数据和 HTML 这样的表示层代码混在一起编写后, 由浏览器展示给最终客户。

这样编写的程序代码像面条一样，将服务器端的程序代码和表示层的标签混在一个文件中编写，为程序的调试和修改带来很大的麻烦。甚至美工对页面布局的轻微调整都会导致大量代码的重新编写和调试。经验丰富的开发者能够区分出文件中哪些是数据，哪些是表示层的标签，但这通常很难做到，它需要精心的计划和不断的尝试。而 MVC 的出现则从根本上强制性地将它们分开。

尽管 MVC 的概念很早就有人提出，但是直到像 Struts、JSF 之类的 MVC 构架产品逐渐成熟后，MVC 的模型才真正在软件系统的开发过程中得到广泛应用。

如果你已经对 JSP 的“面条式”编程感到失望，或者你需要维护其他项目组用 JSP 编写的应用系统，正面对着一大堆 JSP 程序而无从下手，又或者你作为一位资深的项目经理，需要带领团队开发一个大型应用系统，而你正对选择什么样的构架设计煞费苦心时，再或者你需要完成一个界面多样化（可以随心定义，甚至可以通过 Wap 为手机用户提供服务）的应用系统，那么建议你采用 MVC 构架。

尽管构造 MVC 应用程序需要一些额外的工作，但是它所带来的好处是毋庸置疑的。

需要说明的是：作为一个 MVC 的框架，Struts 并不是唯一的 Java 语言对 MVC Framework 的实现。但 Struts 作为 Jakarta 计划的一部分，无论从易用性、可靠性，还是拥有用户的数量上都是目前行业中的佼佼者。因此我们将通过 Struts 来领略一下 MVC 的魅力所在。

1.2 和 Java 相关的工具介绍

1.2.1 JDK 1.6

JDK 是（Java 开发包 Java Development Kit 的缩写），也就是通常所说的 Java。大家所说的 Java 的版本就是指 JDK 的版本。作为 Java 家族最核心的部分，JDK 中包含了 Java 所有的类库、编译工具、解释环境等，由 Sun MacroSystem 提供升级、支持和服务。

JDK 1.6 是目前 Java 的最新版本。这个版本的一个重要主题就是通过新增一些特性来简化开发，这些特性包括泛型、For-Each 循环、自动装包/拆包、枚举、可变参数、静态导入。使用这些特性有助于我们编写更加清晰、精练、安全的代码。下面我们简单介绍一下这些新特性。

1) 泛型（Generic）

我们在编写程序时，经常遇到两个方法的功能非常相似，只是一个处理 int 数据，另一个是处理 string 数据，或其他自定义的数据类型，但我们没有办法，只能分别写多个方法处理每个数据类型。如果能够在方法中传入通用的数据类型，这样就可以用一个方法实现了。

泛型的出现就是专门解决这个问题的。JDK 1.6 中增加的泛型类型，这可以说是 Java 语言在类型安全方面的一个重要的改进。

2) For-Each 循环

For-Each 循环的加入简化了集合的遍历，避免了强制类型转换。

3) 自动装包/拆包（Autoboxing/unboxing）

自动装包/拆包大大方便了基本类型数据和它们包装类的使用。

自动装包：基本类型自动转为包装类（int >> Integer）；

自动拆包：包装类自动转为基本类型（Integer >> int）；

在 JDK 1.6 之前，我们总是对集合不能存放基本类型而耿耿于怀，现在自动转换机制解决了我们的问题。

4) 枚举 (Enums)

JDK 1.6 加入了一个全新类型的“类”——枚举类型。为此 JDK 1.6 引入了一个新关键字 `enum`。枚举类型还提供了两个有用的静态方法 `values()` 和 `valueOf()`。我们可以很方便地使用它们。

5) 可变参数 (Varargs)

在 JDK 1.6 之前，我们可以用重载来实现，但是这样就需要写很多的重载函数，显得不是很有效。如果使用可变参数的话，我们只需要一个函数就行了，引入可变参数以后，Java 的反射包也更加方便使用了。

6) 静态导入 (Static Imports)

使用静态导入可以使被导入类的所有静态变量和静态方法在当前类直接可见，使用这些静态成员无须再给出它们的类名。不过，过度使用这个特性也会在一定程度上降低代码的可读性。

1.2.2 Tomcat

Tomcat 是在 Sun 的 JSWDK 的基础上发展起来的另一个优秀的 JSP 服务器，它不但支持 Servlet，还提供了 Web 服务器的功能。可以毫不夸张地说，Tomcat 已经成长是目前最主流、使用最广泛、拥有用户数最多的 JSP 服务器。最重要的是，它还是一个开源和免费的产品。

作为一个开放源代码的软件，Tomcat 有着自己独特的优势，可以和目前大部分的主流服务器一起工作，而且有着相当高的运行效率。

Tomcat 本身也是一个功能强大的、支持 JSP 语言的 Servlet 容器（Servlet 容器是一个可以管理、激活、执行 Servlet 的程序）。

市场上主流的 Servlet 容器分为如下几类型：

(1) 独立的 Servlet 容器，内置有 Web 服务器的功能。

这种情况下 Servlet 容器作为一个 Web 服务器的角色出现。两者（Web 服务器和 Servlet 容器）合二为一，为基于 Java 语言的 web 应用充当服务器。Tomcat 的默认模式就是这样作为一个 Java 服务器，承担着 Web 服务器和 Servlet 容器双重身份。

然而，并不是所有的 Web 应用都使用 Java 编写而成。因此，出现如下两种容器的模式。

(2) 进程内的 Servlet 容器，Servlet 容器作为其他 web 服务器的插件出现。Web 服务器插件在内部地址空间打开一个 JVM（Java Virtual Machine）使 Java 容器得以在内部运行。例如：一个用 ASP 编写的 Web 应用，运行在微软的 Internet Information Server（微软的 Web Server，简称为 IIS）中，但其中的某个功能是用 JSP 或 Servlet 编写的，因此需要在 IIS 中内嵌一个 Servlet 容器插件，来执行这些对 JSP 或 servlet 的请求。

进程内容器对于多线程、单进程的服务器非常合适并且能够提供很好的运行速度，但伸缩性有所不足。

(3) 进程外的 Servlet 容器，Servlet 容器运行于 Web 服务器之外的地址空间且作为 Web 服务器的插件和 Java 容器实现的结合。Web 服务器插件和 Java 容器 JVM 使用 IPC 机制（通常是 TCP/IP）进行通信。当一个调用 Servlet 的请求到达时，插件将取得对此请求的控制并将其传递（使用 IPC 等）给 Java 容器进行处理。

进程外容器的反应时间或进程外容器引擎不如进程内容器,但进程外容器引擎具有更好伸缩性、稳定性。

Tomcat 既可作为独立的 Web 服务器和 Servlet 容器 (通常用于开发与调试或小型网站),也可以作为对其他 Web 服务器的附加组件 (如: Apache、IIS 和 Netscape 服务器)。所以在配置 Tomcat 时,必须决定如何应用它,如果选择第二或第三种模式,需要在 Web 服务器安装完成后进行配置。

1.2.3 BEA WebLogic

WebLogic 是一套基于 Java 功能强大的电子商务套件,提供了许多功能强大的中间件以方便编程人员编写 JSP、SERVLET 等电子商务应用,可以为企业提供一个完整的商务应用解决方案。对于开发人员来说,可以在 www.bea.com 免费下载一套完整的 WebLogic,并得到一个限制了 IP 的 license,用于学习和开发基于这个套件的代码。如果需要正式投入使用的话,那么就必须支付一定的费用获取没限制的 license。由于这个套件基于这种发布方式,对于一般网站开发人员可以很轻易地得到 WebLogic 用于学习开发,当然投入使用是另一回事。

BEA WebLogic Server 8.1 是业界最好的应用服务器,它能够使 IT 部门降低应用基础架构的复杂程度和成本,从而将 IT 资产转化成一种重要的业务优势。

BEA WebLogic Server 8.1 提供了一种全新的方法,以开发、集成、保护和管理服务器端的 Java 应用。

- 提高开发人员生产效率。
- 充分利用现有资源与现有基础结构即取即用的互操作性,既统一了企业环境,又使成本降低。
- 适应性强的安全架构,安全管理员无须应用编码,就能轻松的设计和实现各种运行时 (Runtime) 安全策略。
- 简化管理,应用管理员能够简化并实现自动化运行,从而能够观察和掌握应用和基础结构。
- 简单、快速部署。
- 每分钟、每秒钟可处理的事务更多。
- 良好的可扩展性。
- 更加详尽、完善的日志记录 (log)。

1.2.4 Oracle iAS

Oracle iAS 的英文全称是 Oracle Internet Application Server,即 Internet 应用服务器,Oracle iAS 是基于 Java 的应用服务器,通过与 Oracle 数据库等产品的结合,Oracle iAS 能够满足 Internet 应用对可靠性、可用性和可伸缩性的要求。

Oracle iAS 最大的优势是其集成性和通用性,它是一个集成的、通用的中间件产品。在集成性方面,Oracle iAS 将业界最流行的 HTTP 服务器 Apache 集成到系统中,集成了 Apache 的 Oracle iAS 通信服务层,可以处理多种客户请求,包括来自 Web 浏览器、胖客户端和手持设备的请求,并且根据请求的具体内容,将它们分发给不同的应用服务进行处理。在通用性方面,Oracle iAS 支持各种业界标准,包括 JavaBeans、CORBA、Servlets 以及 XML 标准等,这种对标准的全面支持使得用户很容易将在其他系统平台上开发的应用移植到 Oracle 平台上。