



全国高等职业教育规划教材

Java Web设计实用教程

——基于Eclipse环境的应用开发

主 编 孔 昊

副主编 卢 霞



电子课件下载网址 www.cmpedu.com



机械工业出版社
CHINA MACHINE PRESS

全国高等职业教育规划教材

Java Web 设计实用教程

——基于 Eclipse 环境的应用开发

主 编 孔 昊

副主编 卢 霞

参 编 刘新辉 胡银保 李向伟

主 审 赵锡英



机械工业出版社

本书以 Eclipse 工具作为开发环境,讲解了 Java Web 设计理论和设计方法,是长期在企业从事 Eclipse 软件开发的工程师与高校从事相关课程教学的老师紧密合作的成果。本书内容包括 Eclipse 基础、Web 开发环境搭建、Web 项目开发、MVC 框架设计、Struts 框架设计、Hibernate 框架设计、JUnit 测试、CVS 版本控制等内容。每章最后均安排了实训题,作为对教材的补充、强化和引导,便于读者上机练习。书中内容理论与实践相结合,注重基本知识的理解与基本技能的培养,是一本实用性较强的教材。

本书既可作为高职高专计算机及相关专业的 Java Web 设计课程教材,也非常适合 Eclipse 开发学习和参考。

本书配有授课电子课件和已调试成功的项目,需要的教师可登录 www.cmpedu.com 免费注册、审核通过后下载,或联系编辑索取(QQ: 1239258369, 电话: 010-88379739)。

图书在版编目(CIP)数据

Java Web 设计实用教程——基于 Eclipse 环境的应用开发/孔昊主编. —北京:机械工业出版社,2012.2
全国高等职业教育规划教材
ISBN 978-7-111-37298-1

I. ① J… II. ① 孔… III. ① JAVA 语言-程序设计-高等职业教育-教材 IV. ① TP312

中国版本图书馆 CIP 数据核字(2012)第 014208 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:鹿征

责任印制:乔宇

三河市宏达印刷有限公司印刷

2012 年 4 月第 1 版·第 1 次印刷

184mm×260mm·17.75 印张·438 千字

0001-3000 册

标准书号:ISBN 978-7-111-37298-1

定价:34.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010)88361066

门户网:<http://www.cmpbook.com>

销售一部:(010)68326294

教材网:<http://www.cmpedu.com>

销售二部:(010)88379649

读者购书热线:(010)88379203

封面无防伪标均为盗版

全国高等职业教育规划教材 计算机专业编委会成员名单

主 任 周智文

副 主 任 周岳山 林 东 王协瑞 张福强
陶书中 龚小勇 王 泰 李宏达
赵佩华

委 员 （按姓氏笔画排序）

马 伟	马林艺	万雅静	万 钢
卫振林	王兴宝	王德年	尹敬齐
史宝会	宁 蒙	刘本军	刘新强
刘瑞新	余先锋	张洪斌	张 超
李 强	杨 莉	杨 云	罗幼平
贺 平	赵国玲	赵增敏	赵海兰
钮文良	胡国胜	秦学礼	贾永江
徐立新	唐乾林	陶 洪	顾正刚
康桂花	曹 毅	眭碧霞	梁 明
黄能耿	裴有柱		

秘 书 长 胡毓坚

出版说明

根据《教育部关于以就业为导向深化高等职业教育改革的若干意见》中提出的高等职业院校必须把培养学生动手能力、实践能力和可持续发展能力放在突出的地位，促进学生技能的培养，以及教材内容要紧密结合生产实际，并注意及时跟踪先进技术的发展等指导精神，机械工业出版社组织全国近 60 所高等职业院校的骨干教师对在 2001 年出版的“面向 21 世纪高职高专系列教材”进行了全面的修订和增补，并更名为“全国高等职业教育规划教材”。

本系列教材是由高职高专计算机专业、电子技术专业和机电专业教材编委会分别会同各高职高专院校的一线骨干教师，针对相关专业的课程设置，融合教学中的实践经验，同时吸收高等职业教育改革的成果而编写完成的，具有“定位准确、注重能力、内容创新、结构合理和叙述通俗”的编写特色。在几年的教学实践中，本系列教材获得了较高的评价，并有多品种被评为普通高等教育“十一五”国家级规划教材。在修订和增补过程中，除了保持原有特色外，针对课程的不同性质采取了不同的优化措施。其中，核心基础课的教材在保持扎实的理论基础的同时，增加实训和习题；实践性较强的课程强调理论与实训紧密结合；涉及实用技术的课程则在教材中引入了最新的知识、技术、工艺和方法。同时，根据实际教学的需要对部分课程进行了整合。

归纳起来，本系列教材具有以下特点：

- 1) 围绕培养学生的职业技能这条主线来设计教材的结构、内容和形式。
- 2) 合理安排基础知识和实践知识的比例。基础知识以“必需、够用”为度，强调专业技术应用能力的训练，适当增加实训环节。
- 3) 符合高职学生的学习特点和认知规律。对基本理论和方法的论述要容易理解、清晰简洁，多用图表来表达信息；增加相关技术在生产中的应用实例，引导学生主动学习。
- 4) 教材内容紧随技术和经济的发展而更新，及时将新知识、新技术、新工艺和新案例等引入教材。同时，注重吸收最新的教学理念，并积极支持新专业的教材建设。
- 5) 注重立体化教材建设。通过主教材、电子教案、配套素材光盘、实训指导和习题及解答等教学资源的有机结合，提高教学服务水平，为高素质技能型人才的培养创造良好的条件。

由于我国高等职业教育改革和发展的速度很快，加之我们的水平和经验有限，因此在教材的编写和出版过程中难免出现问题和错误。我们恳请使用这套教材的师生及时向我们反馈质量信息，以利于我们今后不断提高教材的出版质量，为广大师生提供更多、更适用的教材。

机械工业出版社

前 言

2004年2月，IBM把Eclipse转化为一个独立的实体，也就是我们现在所熟知的Eclipse基金会，这是继Apache软件基金会之后的又一个开源软件团体。发展至今，Eclipse为IDE平台开启了一个新的时代，它的应用已遍布在世界的每个角落。

作为Java集成开发环境（IDE），Eclipse所提供的Java编写、编译、调试在易用性、便捷性及效率方面极具优势。除此之外，Eclipse平台又是一个成熟的、精心设计的、可扩展的平台，众多插件的支持使得Eclipse拥有其他功能相对固定的IDE软件很难具有的灵活性。在它的支持下，基于Java的Web设计更显优势。

本书将开发技术与开发环境相结合，既要讲明Web设计的过程，又要突出Eclipse开发工具对Web设计的支持，所以从全书布局上体现了以下设计思路：在熟悉Eclipse开发环境和掌握Web设计基本技术的基础上，进一步加强和学习Web架构的设计，学习测试理论和测试方法，掌握在开发过程中进行版本控制的意义和实现手段。利用Eclipse所具备的丰富的开发内涵进行Web设计是一个循序渐进并不断学习的过程，所以本书在内容的选取和安排上，不但考虑教学要求，更遵循以基础为重、不断拓展的原则。

本书分为9章。第1章介绍了Eclipse的历史发展、体系结构、安装及汉化，同时讲解了插件的安装方法。第2章在介绍了开发环境及基本操作之后，开始创建Java项目，并围绕构建系统、程序运行和调试等主题进行了详细讲解。第3章是进行Web开发的基础，从Web开发环境的分析开始，介绍了应用服务器Tomcat的安装与配置、MyEclipse插件的安装与配置、MySQL的安装与配置，为后继的开发做好了准备工作。第4章引出了Web开发的一些经典问题，如Servlet机制及其处理过程、JavaBean机制及其处理过程、如何进行数据库连接操作、如何调试Web项目等，并进行了详细描述和示范操作。第5章通过一个实例加强Web开发训练，实例以MVC结构为设计思路，遵循MVC的协作与处理规则，将视图、控制器、模型三者有机地结合起来，形成清晰、明确的层次结构，突出可扩展功能的设计优势。第6章介绍了Struts框架设计，通过实例演示了Struts结构的设计和实现，并利用这个技术实现了第5章实例的一个子模块，使读者体会到Struts框架设计带来的开发效率。第7章从项目开发过程中的数据层着手，引出了数据持久化概念，并通过实例细致地讲解了数据持久化技术的实现，即Hibernate框架技术的实现。第8章对JUnit单元测试进行了讲解，包括JUnit插件及其安装、编写测试单元、调试测试用例等，是保证项目顺利开发的重要手段。第9章以团队开发为主题，着眼于CVS版本控制与管理软件，讲解了关于CVS服务器及客户端的安装与配置、CVS项目的落实与更新、冲突的产生与解决、打包版本与分支操作等知识点，是团队开发和阶段性开发的有力武器。

本书的特点如下。

(1) 基础部分以Eclipse开发环境为主，突出介绍工作台应用、项目创建与配置、代码编写与调试、项目构建与运行等基本概念和使用技巧，为后继的开发工作打好基础。

(2) 重点介绍Eclipse环境中Web开发的过程，分两个环节展开：

- MVC 模式的建立和开发，突出 Jsp + JavaBean + Servlet 的应用方式；
- Struts 框架及 Hibernate 框架的设计和实现。

(3) 作为对项目开发的支持，介绍 JUnit 测试理论及 CVS 版本控制理论。

(4) 对学习了 Java 语言的初学者来说，很容易进入到基础部分的学习；对有一定 JSP 基础的学习者来说，Web 设计就变得容易了。通过学习本书，读者既可以掌握新的开发工具、学习新的知识点，又可以把已经学过的知识再次运用和拓展，从而提高了开发效率，是工具应用与开发工作相结合的一个实践过程，为以后进一步进行 Web 开发奠定了良好的学习基础。

(5) 以经典案例配合教学点实施讲解，全线贯穿，以求结构完整、思路清晰。

(6) 采用图解方式讲解、演示和分析，便于理解和上机调试。

(7) 在每一节中，根据需要及时安排一些诸如“温馨提示”、“操作向导”等细节，提高学习者的兴趣，从而更快地掌握一些开发经验，提高学习速度。

(8) 每章配备实训题目，与教学点相得益彰，有利于知识点的理解和掌握。

本书由孔昊任主编，卢霞任副主编。书中第 1、5 章由孔昊编写；第 2、8 章由刘新辉编写；第 3 章由李向伟编写；第 4、7 章由卢霞编写；第 6、9 章由胡银保编写。兰州工业学院赵锡英教授认真细致地审阅了全书，提出了许多宝贵意见，编者深表感谢。

由于作者水平有限，不当之处在所难免，恳请读者批评指正。

编 者

目 录

出版说明

前言

第 1 章 Eclipse 概述及安装 1

1.1 认识 Eclipse 1

1.1.1 Eclipse 的历史 1

1.1.2 Eclipse 的体系结构及插件机制 2

1.2 安装 Eclipse 5

1.3 安装 Eclipse 语言包 9

1.4 小结 11

1.5 实训 12

第 2 章 Eclipse 的集成开发环境 16

2.1 Eclipse 界面 16

2.2 常用视图 17

2.2.1 视图操作 17

2.2.2 几种常见视图 18

2.3 透视图 23

2.3.1 编辑器 23

2.3.2 Java 透视图 25

2.3.3 资源透视图 25

2.3.4 Java 浏览透视图 26

2.3.5 Java 类型层次结构透视图 26

2.3.6 调试透视图 28

2.4 Java 程序编写 29

2.4.1 创建 Java 项目 30

2.4.2 创建 Java 类 31

2.4.3 Java 编辑器 32

2.5 编译和运行 35

2.5.1 Java 构建器 35

2.5.2 代码编译 35

2.5.3 代码运行 40

2.6 Java 代码调试 43

2.6.1 调试参数的设置 43

2.6.2 调试代码 44

2.7 小结 48

2.8 实训 48

第 3 章 Eclipse 中搭建 Web 开发

环境 52

3.1 Eclipse 中 Web 开发环境的
分析 52

3.2 应用服务器 Tomcat 的安装与
配置 53

3.2.1 Tomcat 简介 54

3.2.2 Tomcat 的安装与配置 54

3.3 MyEclipse 插件的安装与配置 57

3.4 数据库的安装与配置 60

3.4.1 MySQL 与 Navicat MySQL 简介 60

3.4.2 MySQL 与 Navicat MySQL 的安装与
配置 61

3.5 小结 65

3.6 实训 65

第 4 章 Eclipse 中开发 Web 项目 67

4.1 Eclipse 中创建 Web 工程 67

4.1.1 一个 Web 工程的创建 67

4.1.2 Web 工程的部署 71

4.1.3 完成用户登录功能 73

4.2 建立 Servlet 处理方式 77

4.2.1 Servlet 的工作流程 77

4.2.2 建立 Servlet 78

4.2.3 利用 Servlet 技术实现用户登录
功能 83

4.3 建立 JavaBean 处理方式 85

4.3.1 建立 JavaBean 85

4.3.2 利用 Servlet + JavaBean 技术实现
用户登录功能 87

4.4 数据库开发 90

4.4.1 利用 DB 视图测试数据库连接 90

4.4.2 编程完成数据库操作 94

4.4.3 用户登录功能的再设计 98

4.5	Eclipse 中调试 Web 工程	103	6.6	实训	184
4.5.1	配置调试环境	103	第 7 章	Hibernate 实现数据持久化	189
4.5.2	调试 Servlet 和 JavaBean 文件	104	7.1	持久化技术	189
4.5.3	调试 JSP 页面	106	7.1.1	建立持久层应用模型	189
4.6	小结	108	7.1.2	持久层技术的实现	190
4.7	实训	108	7.2	Eclipse 中开发 Hibernate	191
第 5 章	Eclipse 开发实例——电子		7.2.1	对项目添加 Hibernate 支持	191
	会议厅	111	7.2.2	配置 hibernate.cfg.xml 文件	197
5.1	MVC 设计模式	111	7.2.3	创建映射文件	199
5.2	系统分析	112	7.3	Hibernate 项目的设计	204
5.2.1	任务描述	112	7.3.1	数据处理的基本实现方式	204
5.2.2	功能分析	113	7.3.2	数据处理的 DAO 模式实现	205
5.3	数据库设计	114		方式	205
5.3.1	数据库需求分析	114	7.4	利用 Hibernate 实现“班级事务	
5.3.2	数据库逻辑结构设计	115		浏览”模块	210
5.4	框架结构设计	116	7.4.1	准备项目结构	210
5.4.1	框架结构分析	117	7.4.2	具体设计	211
5.4.2	Web 项目建立与目录结构		7.5	项目的打包与发行	219
	设计	118	7.6	小结	220
5.5	功能设计与实现	121	7.7	实训	221
5.5.1	班级事务发布功能	121	第 8 章	JUnit 单元测试	226
5.5.2	班级事务维护功能	131	8.1	关于 JUnit	226
5.5.3	班级事务浏览功能	145	8.1.1	测试驱动开发	226
5.5.4	班级事务评论功能	148	8.1.2	JUnit 插件	227
5.5.5	班级事务查询功能	154	8.2	编写和运行 JUnit 测试	229
5.6	小结	162	8.2.1	JUnit 4.0 + 应用中的特性	229
5.7	实训	162	8.2.2	编写测试用例	232
第 6 章	Struts 实现 MVC 设计	164	8.2.3	运行测试用例	237
6.1	实例介绍	164	8.2.4	调试测试用例	239
6.2	Struts 框架结构	165	8.3	JUnit 测试案例	241
6.2.1	Struts 的工作流程	165	8.3.1	异常测试	241
6.2.2	Struts 的组件结构	166	8.3.2	字符串截取测试	243
6.3	Struts 实例设计	167	8.4	小结	245
6.4	利用 Struts 实现“班级事务发布”		8.5	实训	245
	模块	178	第 9 章	使用 CVS 进行版本控制	249
6.4.1	准备项目结构	178	9.1	版本管理与 CVS	249
6.4.2	获取事务分类信息的设计	179	9.1.1	版本管理	249
6.4.3	班级事务发布的设计	181	9.1.2	关于 CVS	250
6.5	小结	184	9.1.3	Eclipse 对 CVS 的集成	251

9.2	CVS 服务器的安装与配置	252	9.4.3	文件的提交与更新	263
9.2.1	安装 CVSNT	252	9.4.4	冲突的产生与解决	265
9.2.2	配置 CVSNT	254	9.5	标记版本	267
9.3	CVS 客户端的配置	257	9.6	小结	270
9.4	CVS 项目管理	259	9.7	实训	270
9.4.1	将 Eclipse 项目提交到 CVS	259	参考文献		274
9.4.2	导入 CVS 服务器上的项目	261			

第 1 章 Eclipse 概述及安装

第一章是学习的起点，我们将认识一种新的编程工具——Eclipse 集成开发环境。Eclipse 极大地改善了开发环境，使开发者得到了有力的帮助。通过学习，读者将体会到 Eclipse 为什么会受到广大开发者的青睐，而且，随着更深层次地应用，读者对它的认识，也会从最初的编程工具逐步转换成开发工具。

在学习 Eclipse 之前，开发者有必要了解它的成长背景和发展前途，更要掌握它的发展现状，以便根据自己所需选择合适的版本和插件进行安装，搭建一个理想的开发平台。在这一章我们将围绕下面的主题进行必要的阐述、说明和指导，为后面章节的开展做好准备工作。

本章内容包括：

- Eclipse 的历史及发展过程
- Eclipse 的体系结构
- Eclipse 的安装
- Eclipse 插件的安装

1.1 认识 Eclipse

1.1.1 Eclipse 的历史

Eclipse 起源的一个非常重要的原因是：IBM 的开发工具 Visual Age for Java 和 WebSphere Studio 很难集成到一起，而且底层的技术比较脆弱，进一步发展非常艰难，所以无法满足业界应用开发的需求。因此，1998 年，IBM 成立了一个项目开发小组，探索下一代开发工具，并希望这个项目能够吸引开发人员，发展起一个强大而又充满活力的商业合作伙伴社区。从商业目标考虑，通过开源的方式，IBM 推出了 Eclipse 试用计划，并在开发社区广泛开展起来。

2001 年 12 月，IBM 向世界宣布了两件事：第一件事是创建开源项目，即 IBM 捐赠价值 4 千万美元的源码给开源社区；另外一件事是成立 Eclipse 协会（Eclipse Consortium），这个协会由一些成员公司组成，主要任务是支持并促进 Eclipse 开源项目。自此 Eclipse 在开发者社区得到飞速发展。后来基于促进 Eclipse 发展的目的，IBM 与其他成员公司合作起草了管理条例，于 2004 年初成立了 Eclipse 基金会（Eclipse Foundation）。这样，Eclipse 的完全开放性使得任何人都可以参与，而参与的最简单方法是关注它的新闻组。用户可以选择一个感兴趣的技术领域，然后下载代码，自己做些实验，并在新闻组上回答一些问题，这样随着技术的提高，就可以提交 bug 和修复包，或去修复其他 bug。如果得到项目工作组的认可，便可以承担项目的一些任务，成为项目开发组的真正成员。由于 Eclipse 是开放源代码的免费软件，程序员们可以阅读到最优秀的代码，学到最优秀的设计理念，推动 Eclipse 蓬勃发展。

表 1-1 表示了 Eclipse 产品从诞生以来，目前已有的版本代号。

表 1-1 Eclipse 的发布历史

版本代号	发行日期	平台版本
	2001 年 11 月 7 日	1.0
	2002 年 6 月 28 日	2.0
	2003 年 3 月 28 日	2.1
	2004 年 6 月 25 日	3.0
IO (伊奥)	2005 年 6 月 27 日	3.1
Callisto (卡利斯托)	2006 年 6 月 30 日	3.2
Europa (欧罗巴)	2007 年 6 月 25 日	3.3
Ganymede (伽倪墨得斯)	2008 年 6 月 25 日	3.4
Galileo (伽利略)	2009 年 6 月 24 日	3.5
Helios (赫利俄斯)	2010 年 6 月 23 日	3.6
Indigo (靛青)	2011 年 6 月	3.7



温馨提示:

从 Eclipse 3.1 版本开始到 Eclipse 3.4，形成了一个与木星卫星名称相关的系列。由于木卫 1 ~ 木卫 4 分别被赋予了 4 个神话传说中人物的名字，所以版本代号也随之起名为表中的名称，并遵循这 4 颗卫星距离木星由近到远的顺序，将其名称依次指定为 Eclipse 的发布版本：3.1、3.2、3.3 和 3.4。

1.1.2 Eclipse 的体系结构及插件机制

1. 体系结构

Eclipse 是一种普遍适用的开放式扩展 IDE。所谓普遍适用就是它不仅仅可以用来开发 Java 程序，也可以用来开发 C/C++、PHP；所谓开放式扩展就是任何人都可以扩展 Eclipse 的功能。这源于 Eclipse 拥有一个优良的体系结构，该体系结构由一个微小核心及众多插件组成。通过特有的插件开发结构，可以使其支持任何语言，甚至成为绘制图片的工具箱。图 1-1 展示了 Eclipse 的体系结构。

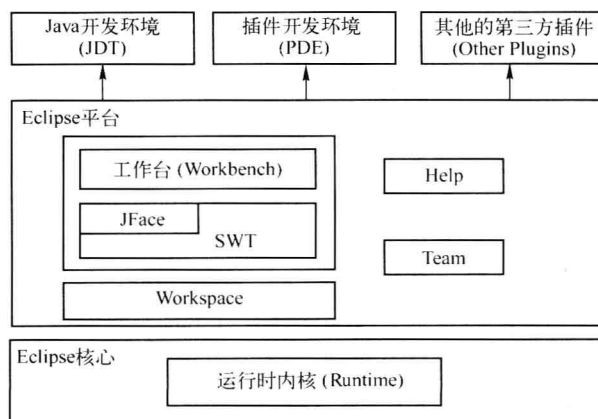


图 1-1 Eclipse 体系结构

图中结构说明如下。

- 运行时内核：负责所有已安装插件的发现和启动，并且创建关于插件的注册信息。Eclipse对这些插件是动态载入并动态调用的。所谓动态就是指，Eclipse启动后要真正用到某个插件时，该插件才会被调入内存，当该插件不再被使用时，它就会在适当的时候被清除出内存。因此即使安装了一大堆插件在Eclipse中，也不必担心某些不常用的插件会白白消耗内存。
- 工作空间：是一个管理用户资源的插件。这些资源包括用户创建的项目和其中的文件，以及文件变更和其他资源。工作空间还负责通知其他插件关于资源变更的信息，如文件创建、删除或更改。
- 工作台：是指桌面开发环境，包含一个或多个透视图，其目标是获得无缝的工具集成。工作台是使用标准窗口工具包（Standard Widget Toolkit, SWT）和一个更高级的API（Application Programming Interface, 应用程序编程接口）——JFace来构建的。
- SWT/JFace：是Eclipse为开发人员提供的、生产大量GUI（Graphical User Interface, 图形用户界面）驱动工具和应用程序的基础。SWT（Standard Widget Toolkit）是Java的Swing/AWT非标准替代者，JFace则建立在SWT基础上，向SWT应用程序中添加大量服务，简化了基于SWT的应用程序的构建。它们的应用可以从两个方面阐释，既是用于GUI开发的API，开发用于Eclipse环境的图形用户界面，又可以将它们用于开发高效率的桌面应用。
- Eclipse的帮助功能：提供了一个附加的导航结构，帮助的内容很全面，并且允许工具以HTML文件的形式添加文档。如果安装了多国语言包后，其帮助文件也可实现全部中文化。
- 团队支持：随着项目复杂度的提高，开发人员数目的增多和工作独立性要求的提高，版本控制在团队开发中是必不可少的，团队支持组件负责提供版本控制和配置管理支持。软件配置管理是软件开发过程中一种按规则实施的、管理软件开发和维护过程以及软件产品的方法，在软件质量管理中起着重要作用。而能否实现有效的版本管理，已经成为判断软件组织是否专业化和正规化的重要标准。版本管理的主要任务是：新版本的生成、过程管理和版本追踪，这些任务贯穿工程配置的整个过程。Eclipse根据用户需要提供了支持小组开发的透视图以及各种版本和配置管理的透视图，使用户可以与其所使用的任何版本控制系统交互。



温馨提示：

最常用的版本管理系统是CVS（Concurrent Versions System，并发版本系统），它是一个开放源代码的项目，是基本的软件工程配置管理工具之一。Eclipse本身内置了CVS客户端，只要再建立一个CVS服务器就可以使用这一功能强大的版本控制系统了。随着软件的更新和发展，Subversion、Git等一系列版本控制软件发展很快，应用前景很好。

Eclipse良好的体系结构非常值得借鉴，尤其是在开发高扩展性、高通用性的应用程序、软件以及系统的过程中更是如此。

2. 插件机制

Eclipse 的设计思想是“一切皆为插件”。

(1) 插件模型

Eclipse 由一个很小的核心和大量建立在核心之上的插件组成，在运行时可以动态发现插件。Eclipse 核心非常小，仅负责基本环境的后台处理工作和提供标准的用户导航模型，而其他功能都基于核心写成插件，只专注于自身任务的处理。这样的插件在图 1-1 中看得非常清楚，如：Eclipse 的图形 API（称为 SWT/JFace）、Java 开发环境插件（简称 JDT）、插件开发环境（简称 PDE）等，Eclipse 对这些插件的协同工作提供了良好的支持，不仅安装简单，还可以无缝结合。

对于 Eclipse 插件架构，一个很好的比喻是 Web 浏览器：如果你想在浏览器中既能浏览多媒体动画，又能浏览 PDF 文件，那么你就需要两个不同的插件，才能保证两种内容都能在 Web 浏览器中显示。Eclipse 也一样，通过它的插件模式我们可以安装不同的插件来进行 HTML 编辑、数据库设计、Java 开发、Web 开发、C++ 开发等等，所有这些开发工作完全在一个 Eclipse 平台上进行。

(2) 插件扩展机制

可以利用 Eclipse 的插件开发环境开发自己的 Eclipse 插件，随己所需地扩展 Eclipse 的功能。同时还可以创建自己的扩展点，即插件模式的扩展和扩展点是递归的。

Eclipse 开发最初的决定之一是：“We will eat our own dog food”。（谚语，指一个公司在日常运行中使用自己的产品），即 Eclipse 团队要使用 Eclipse 进行 Eclipse 的开发。于是，随着使用 Eclipse 构建 Eclipse 插件的经验累积，这种插件模式已变得日臻成熟起来。

(3) 插件的商业效益

由于插件可以使用商业许可证发行，开发人员可以直接为 Eclipse 构建插件，然后出售，以获取商业利润。Eclipse 的所有成员公司基本都是商业软件提供商，也就证明了这个问题的本质。目前全球有上百万开发人员在使用 Eclipse，因此中国的开发人员完全可以构建有价值的插件和工具，并进行销售。

(4) 开发插件的标准

最初，Eclipse 的插件机制是自己设计的，在 OSGI（Open Service Gateway Initiative）成为动态插件机制方面的标准后，Eclipse 按 OSGI 标准重新实现了自己的插件机制，并形成了一个独立的项目 Equinox，它是 OSGI 标准的一个实现。由于 Equinox 有 Eclipse IDE 这么个成功案例，反映出了 Equinox 作为 OSGI 框架的优势。



温馨提示：

OSGI 标准还在发展的过程中，属于比较前沿的技术，关于 OSGI 更详细的信息，可参阅国内开放文档组织 OpenDoc 的《OSGI 实战》，其正式版的下载地址为：http://www.bluedavy.com/opendoc/OSGI_Opendoc.rar。压缩包中包含了 OSGI Opendoc 的 PDF、随文发布的代码以及可运行包。

1.2 安装 Eclipse

Eclipse 几乎在所有主流操作系统上都可以运行，考虑到教学和初次学习的方便性，这里以 Windows XP 中文版操作系统为例。

Eclipse 的发布版本已经到 3.6 版本了，考虑到软件的运行速度以及汉化后带给学习的便捷性，本书采用了 Eclipse 3.2.1 版本。虽然每个版本的 Eclipse 变化并不大，但是就是那一点点变化而导致的使用差异，对于不熟悉 Eclipse 的初学者来说却可能是难以逾越的障碍。

安装 Eclipse 非常简单，它属于绿色安装：不需要运行安装程序，不需要往 Windows 注册表写入信息，只需将下载的 ZIP 压缩包解压就可以运行了。



follow me!

请跟着小向导做一做！

1) Eclipse 是运行在 JDK (Java Development Kit, Java 开发工具包) 之上的，但它本身并不包含 Java 运行时的环境，所以首先要在本机安装 JRE (Java Runtime Environment, Java 运行环境) 或 JDK。

2) 登录 Eclipse 官方下载网站 (<http://archive.eclipse.org/eclipse/downloads/>)，如图 1-2 所示。这里面包含了 Eclipse 1.0 到 Eclipse 3.6.1 的所有版本，单击“3.2.1”超级链接进行下载。

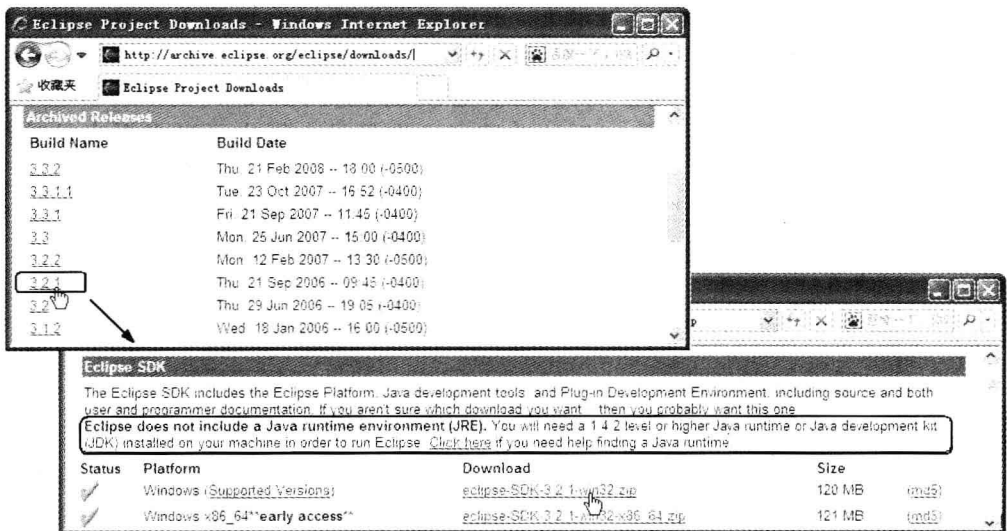


图 1-2 Eclipse 官方下载网站

图中有两点说明如下。

- 页面中圈起来的文字说明：Eclipse 本身是不包含 JRE 的，所以要在 Eclipse 中编译并运行 Java 文件，必须要事先搭建这个平台，这就是我们前面所做的安装 JDK 的工作。由于安装 JDK 的过程包含了 JRE 的安装，又考虑到在某些环境下要配

置环境变量，所以工作流程就变成了安装 JDK 工具。

- 根据提示可以知道：Eclipse 3.2.1 版本所需要的 JDK 支持至少在 1.4.2 版本以上，这就说明 Eclipse 是需要相应版本的 JDK 支持和配合的。在 Eclipse 3.2.1 以上版本的安装中，尽量使用新版本的 JDK 是没有错的。

3) 将下载的压缩包 eclipse-SDK-3.2.1-win32.zip 解压缩到本地路径（这里设为 C:\），解压后的目录结构如图 1-3 所示。

4) 双击 eclipse.exe 可执行文件来启动 Eclipse，如图 1-4 所示。

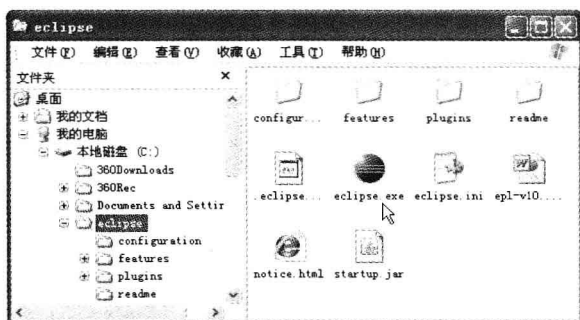


图 1-3 解压后的目录结构

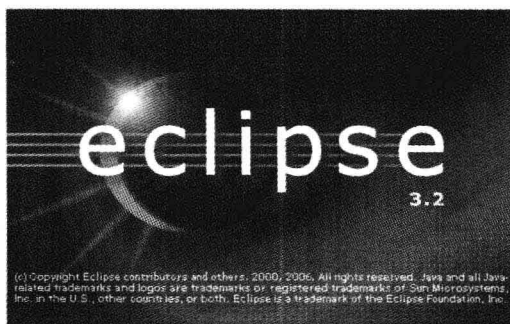


图 1-4 Eclipse 3.2 启动界面

在第一次启动的时候，会弹出一个选择工作空间的对话框，如图 1-5 所示。工作空间用来存放项目源代码和一些 Eclipse 个性化配置文件，在默认情况下，通过 Eclipse 创建的所有项目都会被存放在这个目录下。在备份时，可以备份整个目录。根据需要可以创建多个工作空间，但是为了便于维护和管理，最好只创建一个。图 1-5 中的工作空间目录被修改为“D:\Eclipse 3.2”，由于笔者的计算机中还有 MyEclipse 7.1 版本的软件，为了对二者的工作空间作区别，选用了这个目录，用户可以根据自己的情况设置工作空间目录。

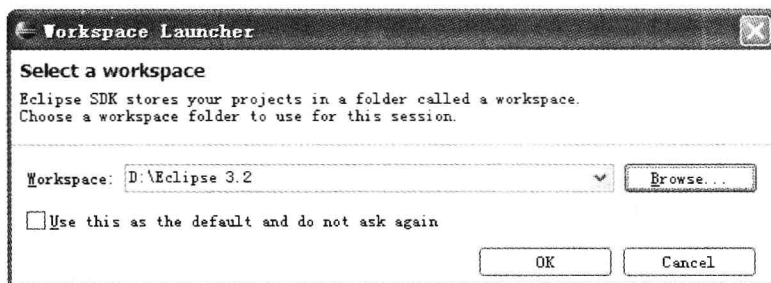


图 1-5 Eclipse 工作空间的设置

以后每次启动 Eclipse 都会出现这个对话框，用户可以勾选“Use this as the default and do not ask again”复选框，将不再询问。

5) 认识 Eclipse 的工作界面。

随着 Eclipse 的启动，出现图 1-6 所示的窗口，包含了一个“Welcome”视图，即一个功能窗口。用户可以点击它所提供的 5 个功能按钮，查看具体的细节。

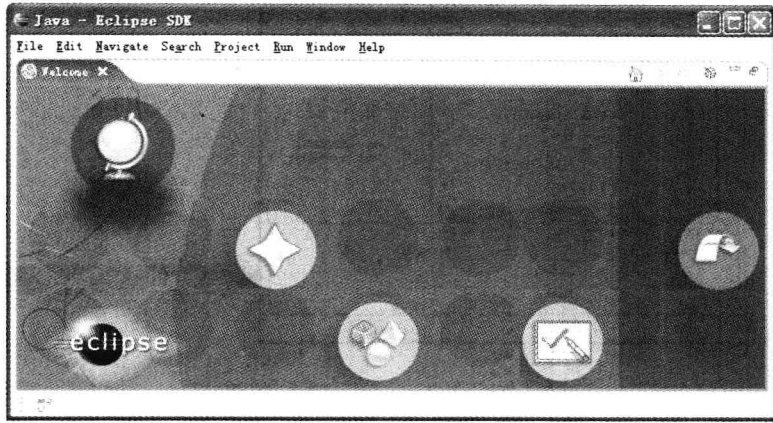


图 1-6 Eclipse 的欢迎界面

关闭“Welcome”视图后，出现了 Eclipse 的工作界面，称为 Eclipse 工作台，如图 1-7 所示。其中各个区域都可以被单独关闭、最大化、最小化和移动位置。

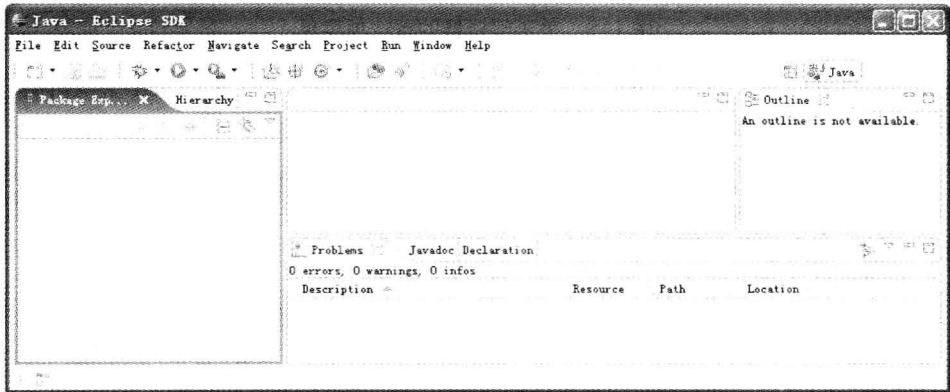


图 1-7 Eclipse 的工作台

从图 1-7 中可以看出：Eclipse 的界面非常简洁，这是用 Java 写的一个 IDE，准确地说，是用 SWT/JFace 开发而成的。SWT/JFace 直接调用本机界面组件，体现了快速并且界面风格和本机一致的特点。用户可以通过改变操作系统的主题风格观察 Eclipse 的界面变化，就可以体会到这一点。从这里也反映了 Eclipse 的一大特点：SWT/JFace 技术为 Eclipse 提供了优秀的图形 API。

6) 观察 Eclipse 与 JDK 的关系。

按照图 1-8 的引导就会进入 Eclipse 的“首选项”配置窗口。

- 在左边的项目栏中选择“Java→Compiler”，可以看到 Eclipse 所选用的编译器的版本。默认是 1.4 版本，根据我们所安装的 JDK 情况，将它改为 6.0，且勾选“Use default compliance settings”设置，这样便保证了 Eclipse 与 JDK 版本的一致性。