

袁梅宇 著
王海瑞

Java EE

企业级编程 开发实例详解

• 探寻Java EE最新版本的革新技术 • 内容涵盖JPA 2.0 + EJB 3.1 + JSF 2.0 • 综合案例接近实战



清华大学出版社

Java EE 企业级编程开发实例详解

袁梅宇 王海瑞 著



清华大学出版社
北京

内 容 简 介

Java EE 一直是企业级应用开发的主流技术标准和流行的软件开发架构,最新的 Java EE 6.0 致力于简化复杂技术,大大降低了开发难度,使用户容易理解和上手。

本书带领读者由浅入深地学习 Java EE 开发的持久化层、业务逻辑层和 UI 用户表现层的知识,系统地介绍称为 Java EE “三剑客”的 JPA 2.0、EJB 3.1 和 JSF 2.0 开发。全书共 5 章,主要内容包括 Java EE 开发基础、Java 持久化 API、EJB 3.1、JSF 2.0、Java EE 应用安全和综合示例。

本书适合软件开发人员作为技术参考书,也适合作为计算机专业高年级本科生和研究生的教材或参考书。

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 企业级编程开发实例详解/袁梅宇,王海瑞著. --北京:清华大学出版社,2013
ISBN 978-7-302-31491-2

I. ①J… II. ①袁… ②王… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 023859 号

责任编辑:魏莹 桑任松
装帧设计:杨玉兰
责任校对:周剑云
责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 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

课 件 下 载: <http://www.tup.com.cn>,010-62791865

印 刷 者:清华大学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:27.5 字 数:668 千字

版 次:2013 年 3 月第 1 版 印 次:2013 年 3 月第 1 次印刷

印 数:1~4000

定 价:49.00 元

产品编号:049600-01

前 言

在当今发达的商业世界中，企业需要使用各种应用软件以帮助提升核心竞争力，而应用软件需要进行数据访问、实施业务逻辑、构筑增进用户体验的表示层，并与外部系统通信。公司需要在尽可能减少开支的同时，选择使用标准的、可靠的技术去处理日益加重的信息传输和信息处理的负荷。选择适合企业的工作平台的重要性日益突出。

Java EE(Java Enterprise Edition, Java 企业版)是一个开放的、基于标准的平台，用于开发、部署和管理面向 Web，以服务器为中心的企业级应用。

自 20 世纪 90 年代后，Java EE 一直是企业级应用开发的主流技术标准和流行的软件开发架构，并一直承受各种开源框架的竞争，每一次 Java EE 新版本的推出，都会给用户带来学习新知识的冲击。

长期以来，Java EE 一直被技术人员认为是难以理解、容易误用的“高雅”技术，其前身 J2EE 常常被业界视为重量级的技术(太过于强大而难以使用)——即使实现一个简单的 J2EE 程序，也需要大量的配置文件，尽管这些配置文件并不是必需的。

因为 J2EE 难以学习和掌握，使得很多开发人员转而寻求其他解决方案，各种以 SSH(Struts + Spring + Hibernate)为代表的轻量级架构大行其道。

吸取了轻量级架构的优点，Java EE 的“体重”在 2006 年 5 月正式发布 Java EE 5.0 规范时成功地“减肥”了。最新的 Java EE 6.0 则致力于简化复杂技术，大大降低了开发的难度，使用户容易理解和上手了。

Java EE 主要的优势在于它是开放的技术标准。使用 JPA、EJB、JMS、SOAP 及 RESTful Web 服务技术编写的应用软件容易在不同的应用服务器间移植。

Java EE 的另一优势是开源，大多数 Java EE 6 的参考实现都采用开源许可证，如 GlassFish、EclipseLink、Mojarra、OpenMQ、Metro 及 Jersey 等。

Java EE 并不仅仅是一门技术，而且还是一系列的技术标准和规范。本书探寻了 Java EE 新版本的革新技术，研究了一系列规范和如何将这些技术组织起来开发应用系统。

Java EE 6 包含近 30 项规范，内容极其庞杂，涵盖企业级开发的方方面面，主要包括 EJB 3.1、JPA 2.0 企业层、Servlet 3.0、JSF 2.0 Web 层、SOAP Web 服务、RESTful 服务的互操作规范。想完全掌握 Java EE 的每项技术，既耗时，也无必要。因此，本书主要涵盖称为开发“三剑客”的 JPA 2.0、EJB 3.1 和 JSF 2.0，分别用于构建企业 Web 应用的持久化层、业务逻辑层和表现层，深入浅出地叙述了各项技术的背景、要点和难点、注意事项，并且附上大量的典型示例供读者参考。本书既可以作为教学参考书，供计算机相关专业的本科生和研究生使用，也可以作为技术手册，在遇到类似的技术问题时查阅。

作为 Java EE 的爱好者和实践者，作者深知：如果没有对大量实例的学习和练习，想学好这些内容是不可能的。大量训练和模仿才是通向成功的正确道路。秉承这个思想，书中安排了大量的实例程序，每个实例都经过作者仔细挑选和构建，并经过认真调试，确认没有问题后才写入书中。



作者专门为本书设置了读者 QQ 群，群号是 **245295017**，欢迎读者加群，下载和探讨书中的实例，抒写读书心得，寻求技术难题解答等。

本书承蒙很多朋友、同事的帮助才得以成文。特别感谢同事尚振宏博士的热心帮助，他提出了不少意见和建议，对本书的改进帮助甚大；衷心感谢周兰江副教授和研究生何佳，难以忘记他们的真诚帮助；感谢朋友李海军博士、史博嘉先生，没有他们的帮助本书难以面世；感谢清华大学出版社的编辑老师在内容组织、排版以及出版方面提出的建设性意见和给予的无私帮助；感谢理解和支持我的家人，他们是我写作的坚强后盾。感谢购买本书的朋友，欢迎大家批评指正，所提出的问题将会受到重视，并在再版中改进。

编 者

目 录

第 1 章 Java EE 开发基础.....1	2.4.1 实体关联基本概念..... 77
1.1 Java EE 6 简介.....2	2.4.2 一对一单向关联..... 78
1.1.1 学习 Java EE 的意义.....2	2.4.3 一对一双向关联..... 82
1.1.2 Java EE 6 版本介绍.....2	2.4.4 一对多单向关联..... 84
1.2 Java EE 6 的开发环境.....4	2.4.5 多对一单向关联..... 88
1.2.1 Java EE 6 开发环境的 安装和配置.....4	2.4.6 多对一双向关联..... 90
1.2.2 Eclipse 集成 GlassFish.....12	2.4.7 多对多双向关联..... 93
1.2.3 Eclipse 集成数据库.....13	2.4.8 有额外字段的多对多映射 关联表..... 96
1.2.4 EJB 容器数据源的配置.....17	2.5 关联关系再讨论..... 101
1.3 Java EE 安全.....21	2.5.1 为关联关系选择集合类..... 101
1.3.1 Java EE 基本安全概念.....21	2.5.2 对关联实体进行排序..... 101
1.3.2 使用 GlassFish 文件域.....24	2.5.3 延迟加载 vs 即时加载..... 104
1.3.3 基于表单的认证.....30	2.6 继承关系..... 106
1.3.4 使用 GlassFish 证书域.....33	2.6.1 概述..... 106
第 2 章 Java 持久化 API.....39	2.6.2 继承关系映射示例..... 106
2.1 JPA 介绍.....40	2.7 JPQL..... 112
2.1.1 持久化概念.....40	2.7.1 JPQL 语句基础..... 112
2.1.2 JPA 2.0 概述.....41	2.7.2 JPQL 查询..... 116
2.2 JPA 环境设置.....43	2.7.3 命名查询..... 120
2.2.1 简单 JPA 程序的完整开发 过程.....43	2.7.4 原生查询..... 122
2.2.2 使用 JUnit 测试 JPA.....51	2.7.5 Criteria API..... 124
2.2.3 persistence.xml 文件配置.....55	2.8 JPA 高级话题..... 128
2.3 JPA 实体映射基础.....57	2.8.1 处理并发..... 128
2.3.1 JPA 管理实体基本概念.....57	2.8.2 实体的生命周期..... 132
2.3.2 实体与数据库表的映射关系.....59	2.8.3 JPA 回调方法..... 142
2.3.3 采用 XML 文件管理对象关系 映射.....63	第 3 章 EJB 3.1..... 145
2.3.4 定义复合主关键字.....66	3.1 EJB 3.1 介绍..... 146
2.3.5 定制属性的映射关系.....70	3.1.1 EJB 3.1 简介..... 146
2.3.6 JPA 的两种访问类型.....75	3.1.2 简单 EJB 的完整开发过程..... 149
2.4 JPA 实体关联映射.....76	3.2 依赖注入和 JNDI..... 156
	3.2.1 基本概念..... 156
	3.2.2 JNDI 示例..... 159



3.3 EJB 调试技术.....162	4.4.2 托管 Bean 的范围..... 254
3.3.1 嵌入式技术调试 EJB.....162	4.5 JSF 导航..... 257
3.3.2 Web 服务测试 EJB.....163	4.5.1 页面导航..... 257
3.3.3 GlassFish 应用程序客户端.....166	4.5.2 动作和动作监听器..... 260
3.4 会话 Bean.....169	4.5.3 参数传递..... 263
3.4.1 无状态 EJB.....169	4.6 JSF 转换器..... 268
3.4.2 有状态 EJB.....172	4.6.1 简单的转换器..... 269
3.4.3 单体 EJB.....176	4.6.2 数值转换器格式化输出..... 271
3.4.4 单体 EJB 初始化与并发.....179	4.6.3 日期时间转换器..... 273
3.4.5 EJB 部署描述文件.....182	4.6.4 用户定制的转换器..... 276
3.5 EJB 进阶.....185	4.7 JSF 验证器..... 280
3.5.1 异步调用 EJB.....185	4.7.1 标准的数据验证器..... 280
3.5.2 定时服务.....188	4.7.2 定制验证器的错误信息..... 282
3.5.3 会话 Bean 的生命周期.....193	4.7.3 用户定制的数据验证器..... 286
3.5.4 EJB 拦截器.....198	4.8 JSF 表格..... 288
3.6 EJB 事务.....204	4.8.1 h:dataTable 标签..... 288
3.6.1 EJB 事务处理.....204	4.8.2 ui:repeat 标签..... 292
3.6.2 EJB 支持的事务.....204	4.8.3 在表格中显示多种组件..... 294
3.6.3 容器管理的事务.....206	4.8.4 编辑表格数据..... 296
3.6.4 Bean 管理的事务.....212	4.8.5 添加删除表格数据..... 299
3.7 EJB 安全.....214	4.8.6 表格数据排序..... 303
3.7.1 EJB 安全概念.....214	4.8.7 表格分页显示..... 308
3.7.2 EJB 声明式安全策略.....216	4.9 Ajax..... 314
3.7.3 EJB 编程式安全策略.....220	4.9.1 使用简单的 Ajax..... 315
第 4 章 JSF 2.0.....223	4.9.2 组件的动态联动..... 318
4.1 JSF 2.0 介绍.....224	4.9.3 构建人性化 Web 应用..... 321
4.2 JSF 的运行环境.....226	4.10 Facelets..... 325
4.2.1 JSF 运行环境的配置.....226	4.10.1 Facelets 标签..... 325
4.2.2 第一个简单的 JSF 程序.....228	4.10.2 使用 Facelets 模板..... 326
4.2.3 JSF 与可视化网页开发.....234	4.11 JSF 进阶..... 328
4.2.4 配置 web.xml 文件.....234	4.11.1 JSF 复合组件..... 328
4.3 JSF 标准组件.....235	4.11.2 JavaScript 复合组件..... 332
4.3.1 JSF 标准组件示例.....236	4.11.3 JSF 的生命周期..... 334
4.3.2 JSF 2.0 标签库.....238	4.11.4 定制 JSF 安全..... 339
4.3.3 JSF 常用标准组件介绍.....240	4.11.5 显示动态图像..... 345
4.4 托管 Bean.....252	4.11.6 文件上传..... 348
4.4.1 JSF 托管 Bean.....252	4.11.7 PrimeFaces 多文件上传..... 355
	4.11.8 在 JSF 中使用 EJB..... 358

4.12 国际化.....	361	5.6.3 排序和分页处理.....	399
4.12.1 国际化 JSF 项目.....	361	5.7 管理员使用的功能.....	402
4.12.2 页面语言功能选项.....	365	5.7.1 添加新用户等.....	402
第 5 章 综合示例.....	369	5.7.2 用户管理.....	404
5.1 系统功能概述.....	370	5.7.3 班级管理.....	410
5.2 数据库设计.....	370	5.7.4 课程管理.....	411
5.3 企业 Java Bean 层.....	378	5.8 教师使用的功能.....	412
5.4 用户认证与授权.....	384	5.8.1 录入成绩.....	412
5.5 JSF 模板.....	386	5.8.2 查询成绩.....	417
5.5.1 基本模板.....	386	5.9 学生使用的功能.....	418
5.5.2 管理员模板.....	387	附录 A 源代码使用说明.....	425
5.5.3 教师、学生模板.....	391	附录 B 章节与 Java EE 项目名称	
5.6 公用功能.....	394	对照表.....	426
5.6.1 登出.....	395	参考文献.....	429
5.6.2 修改密码.....	396		



第 1 章



Java EE 开发基础

本章介绍 Java EE 6 的版本及特色，以及 Java EE 6 开发环境安装和配置的各种常见问题，最后讨论 Java EE 的安全，并附有详细的实例来说明如何使用安全域进行验证和授权，以确保应用程序的安全。





1.1 Java EE 6 简介

了解一项技术平台是选择和使用这个平台的首要条件。只有在了解 Java EE 技术结构和特点之后, Java EE 爱好者才能根据自身需求, 考虑学习和使用哪些技术。

1.1.1 学习 Java EE 的意义

假如程序员要处理对象集合, 除非专门为了学习数据结构, 显然不会从头来构建自己的链表、队列和堆栈, 而是直接使用编程语言提供的集合 API。

同样道理, 假如程序员要开发支持事务、安全、互操作和分布式的应用软件系统, 也不会去重新开发全部的底层 API, 而是直接使用 Java EE 提供的框架。

道理很简单——站在巨人的肩膀上看得更远。就像 Java SE(Java Standard Edition, Java 标准版)提供处理集合的 API 一样, Java EE 提供标准的方式去处理企业级应用, 例如, 提供 JTA(Java Transaction API, Java 事务 API)以处理事务, 提供 JMS(Java Message Service, Java 消息服务)以处理消息, 提供 JPA(Java Persistence API, Java 持久化 API)以处理持久化, 等等。Java EE 就是专门为企业级应用开发提供的一组规范, 可以视其为 Java SE 的扩展, 有助于分布式、高可靠性、强大的、高可用性应用系统的开发。

Java EE 提供 Web 服务、组件模型、管理和通信 API 等一系列完整的解决方案, 用于实现企业级的面向服务体系架构(Service Oriented Architecture, SOA)和构建 Web 2.0 应用程序。

自 20 世纪 90 年代开始, Java 就为用户广泛接受。Java 技术推动了 Web 的迅速发展。由于 Java 出现得较早, 并且能持续地推出创新技术, 因此被广泛应用于很多领域, 拥有跨平台、动态 Web 以及分布式的计算能力, 使学习和使用 Java 成为一种时尚。

综上所述, 学习和使用 Java EE, 可以使开发人员能够紧跟 IT 技术的潮流, 充分地享受先进技术带来的强大功能、高生产率和低复杂性, 在未来的竞争中立于不败之地。

1.1.2 Java EE 6 版本介绍

Java EE 从版本 1.0 开始, 最初的版本称为 JPE(Java Professional Edition, Java 专业版), 于 1998 年 5 月发布, 至今已经走过十多年了, 经历了多次变更, 每次都增加了很多新的规范, 最新版是 Java EE 6, 于 2009 年 12 月发布。各版本的发布时间以及规范概要如图 1.1 所示。

Java EE 6 包括 28 个规范和如图 1.2 所示的 4 种不同功能的容器(Container), 涵盖了企业级开发技术的方方面面。在上一个版本基础上, Java EE 6 取得了不少进展, 添加了大量的新技术, 相对于以往的 Java EE 版本, Java EE 6 进一步简化了平台, 扩展了可用性。

据 Oracle 公司提供的材料, Java EE 6 比 Java EE 5 更加简化了编程。在基于 POJO、JPA、REST 的简单用例中, 与 Java EE 5 相比, Java EE 6 的 Java 类减少了 25%, 代码行减少了 50%, XML 码行减少了 80%。



cron 风格的定时服务，支持通过使用 `@Asynchronous` 标注来对会话 Bean 方法进行异步调用。EJB 3.1 的新特性还包括直接将 EJB 打包成 WAR 文件，可以运行在嵌入式容器中，便于在 Java SE 环境进行 JUnit 测试，使用统一的标准化全局 JNDI 命名方式。

- JSF 2.0: JSF 2.0 正式将 Facelets 作为默认的视图技术，为适应 Ajax 而改变了自身的生命周期，使得 JSF 与 Ajax 组合更加自然。
 - Servlet 3.0: 提供异步处理方式，大量使用标注，简化了 XML 部署描述文件。
- 总之，Java EE 6 简化了平台的使用，但功能更为强大。

1.2 Java EE 6 的开发环境

本节介绍 Java EE 6 开发环境的安装和配置，如何在 Eclipse 中集成 GlassFish 和数据库，以及 EJB 容器数据源的配置。

1.2.1 Java EE 6 开发环境的安装和配置

使用 Java EE 6 进行开发，首要的任务是能够安装和配置开发环境。虽然 EJB 容器、集成开发环境和数据库系统单独的安装配置并不困难，但三者的联合工作还需要一定的技巧，有时一个小小的问题都会让开发人员困惑很长时间。

1. Java EE 开发环境的组成

使用 Java EE 进行开发，离不开三大组成部分——EJB 容器、集成开发环境 IDE 和数据库系统。

(1) EJB 容器按是否收费，分为收费版和免费版。收费版主要是各种商用应用服务器，如 WebLogic、WebSphere、Oracle Application Server 等。收费版昂贵，且不开源，不适合初学者；免费版主要有 JBoss AS 和 GlassFish，两者都是开源的 Java EE 应用服务器，前者目前市场占有率居第一位，后者由 Oracle 旗下的 Sun 主推，有很好的发展前景，因此本书使用 GlassFish 3.1 作为 EJB 容器。注意，广受欢迎的 Tomcat 和 Resin 并不是 EJB 容器，只是一般的 Java Web 应用服务器，只支持 JSP 和 Servlet 开发，不支持 EJB 开发。

(2) 集成开发环境 IDE 也有多种选择，尽管 Oracle 公司发布的 Java EE 6 示例程序都直接支持 NetBeans，但由于历史原因，多数用户更为钟爱 Eclipse，因此本书选用 Eclipse 作为集成开发环境。

(3) 数据库系统已经非常成熟，如果使用 JPA 2.0 作为持久化方案，数据库系统间的差异大部分都已经被屏蔽，使用何种数据库已经基本没有差别。本书选用用户喜爱的 MySQL 和 JDK 自带的 JavaDB 作为示例数据库。

2. 典型 Java EE 开发环境的安装配置

(1) EJB 容器的安装配置

首先到 Oracle 官方网站下载新版 Java EE 6 SDK，本书下载的是 Java EE 6 SDK Update

3 with JDK 7, 网址是 <http://www.oracle.com/technetwork/java/javasee/downloads/index.html>。注意不要选择 Web Profile 版本, 要选择完整版。

根据操作系统, 选择要下载的文件, 本书假定使用 Windows 操作系统, 所做的选择如图 1.3 所示。

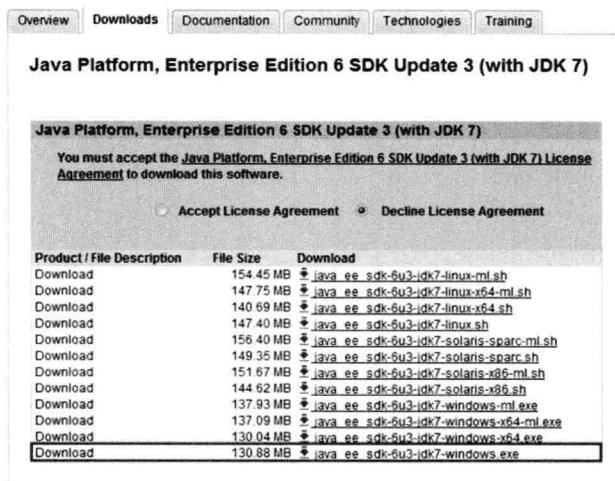


图 1.3 下载 Java EE 6 SDK

运行下载的安装文件, 一步一步地按照提示做, 就能顺利安装, 由于较为简单, 就不再赘述。安装向导如图 1.4 所示。

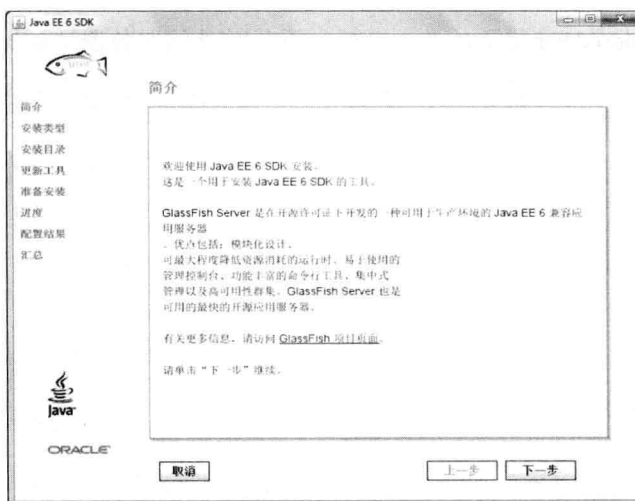


图 1.4 安装向导

注意, 完整版本的 Java EE 6 SDK 附带 JavaDB、JDK 7 等工具, 不用另外专门安装。

(2) Eclipse 安装配置

到 Eclipse 官方网站(<http://www.eclipse.org/downloads/>)下载最新的 Eclipse IDE for Java EE Developers, 这里下载的是 eclipse-jee-indigo-win32.zip 文件, 共 212MB。下载后, 用解



压缩软件释放到读者想要的文件夹中，例如，解压缩到“C:\jee6learning\eclipse”文件夹中。然后双击 eclipse.exe 文件启动，如图 1.5 所示。

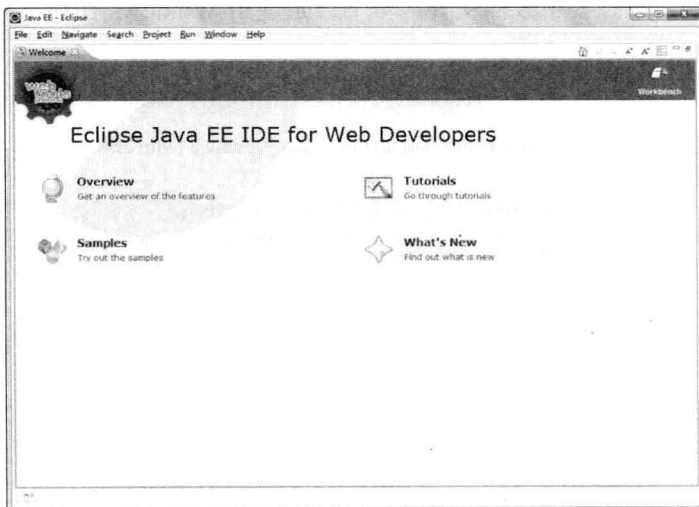


图 1.5 启动 Eclipse

单击欢迎页面右上角的  进入 Eclipse 工作台。

Eclipse 默认没有配置对 GlassFish 服务器的支持，只能手动进行配置。注意确保计算机已连上 Internet。首先在 Eclipse IDE 中，选择 File→New→Other，在弹出的向导窗口中选择 Server→Server，然后单击 Next 按钮，如图 1.6 所示。在出现的 New Server 界面中，单击 Download additional server adapters 链接，下载额外的服务器适配器，如图 1.7 所示。

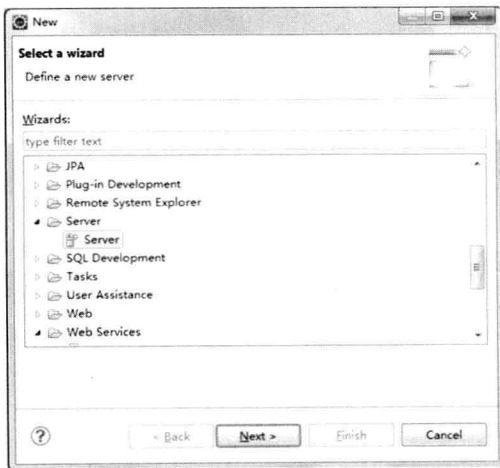


图 1.6 安装 GlassFish 服务器支持工具



图 1.7 下载额外的服务器适配器

选择欲安装的服务器扩展，即 Oracle GlassFish Server Tools，如图 1.8 所示。然后，选中第一个单选按钮，同意安装服务器的许可协议，如图 1.9 所示。

在出现的界面中单击 OK 按钮确认安装服务器扩展，如图 1.10 所示。然后，Eclipse 开

始从网络下载并进行安装，如图 1.11 所示。



图 1.8 选择欲安装的服务器扩展

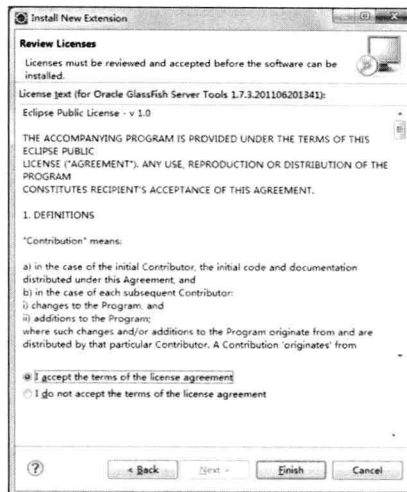


图 1.9 同意许可协议

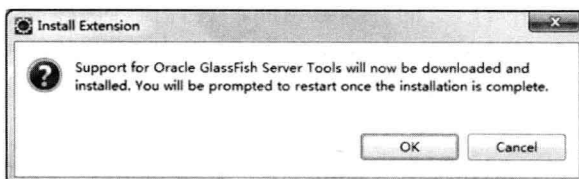


图 1.10 确认安装

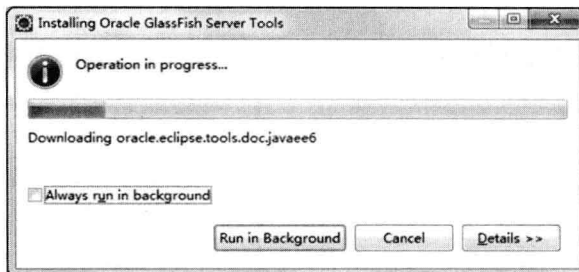


图 1.11 进行安装

安装完毕后，提示需要重新启动 Eclipse 以使安装生效，单击 Restart Now 按钮重新启动，如图 1.12 所示。

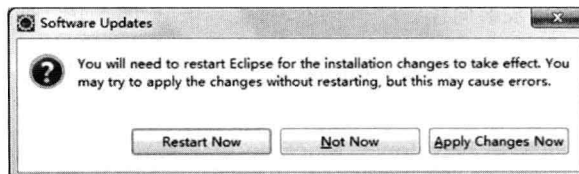


图 1.12 安装完毕重新启动



重新启动后，Eclipse 欢迎页面显示 GlassFish 应用服务器扩展的帮助信息，如图 1.13 所示。单击对应的超链接，可以查看 GlassFish 服务器扩展说明文档，如图 1.14 所示。

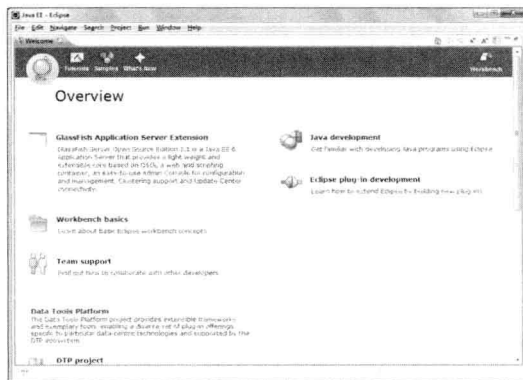


图 1.13 已安装的 GlassFish 服务器扩展

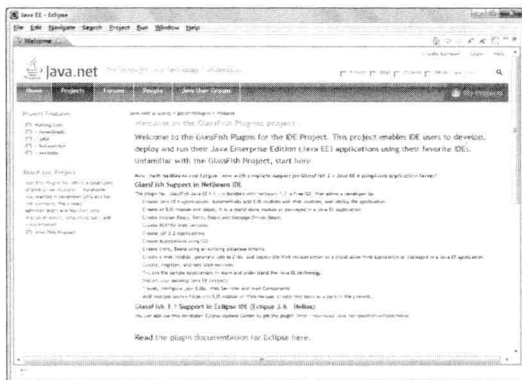


图 1.14 GlassFish 服务器扩展说明

关闭欢迎页面。至此，已经完成 GlassFish 应用服务器扩展的安装。

(3) 安装配置 MySQL

MySQL 数据库是应用最广泛的开源数据库，具有执行性能高、运行速度快、容易使用等特点，其官方主页地址为 <http://www.mysql.com/>，产品分为 MySQL Community Server (MySQL 社区版)和 MySQL Enterprise(MySQL 企业版)两类，前者免费开源，后者用于商业，需要付费。

在编写本书时下载的 `mysql-5.5.15-win32.msi` 为 Windows 环境下的完全安装版本。下载后双击该文件进行安装，在如图 1.15 所示的安装向导中，单击 Next 按钮进入下一步。然后，单击 Typical 按钮选择典型安装，如图 1.16 所示。

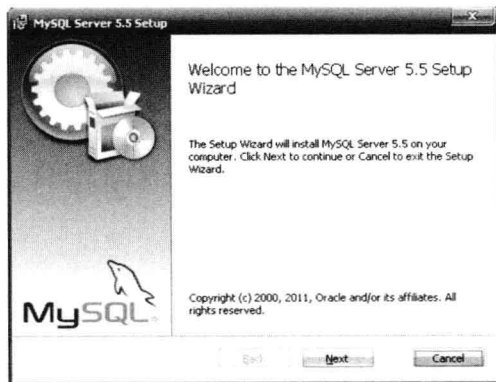


图 1.15 MySQL 安装向导



图 1.16 选择典型安装

选择详细配置，即选中 Detailed Configuration 单选按钮，如图 1.17 所示，单击 Next 按钮进入下一步。

选择开发者模式，即选中 Developer Machine 单选按钮，如图 1.18 所示，单击 Next 按钮进入下一步。

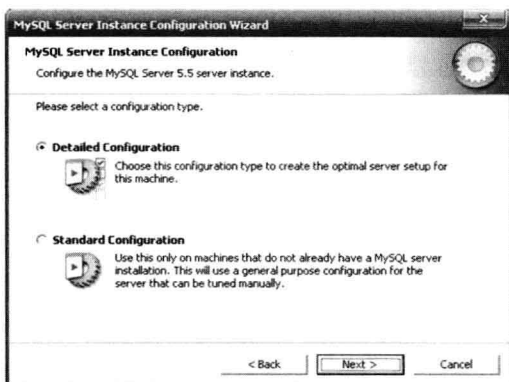


图 1.17 选择详细配置

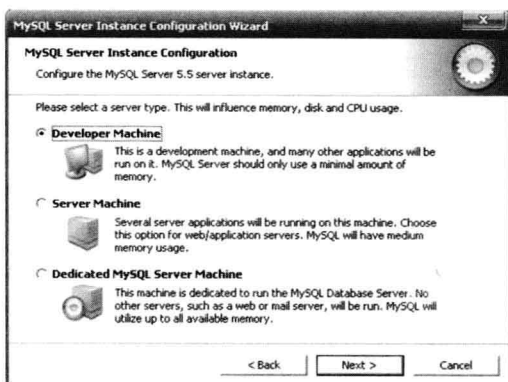


图 1.18 选择开发者模式

选择多功能数据库，即选中 Multifunctional Database 单选按钮，如图 1.19 所示，然后单击 Next 按钮进入下一步。然后，选择表空间存储位置，默认使用 C 盘存储表空间，保持默认设置不变，如图 1.20 所示，单击 Next 按钮。

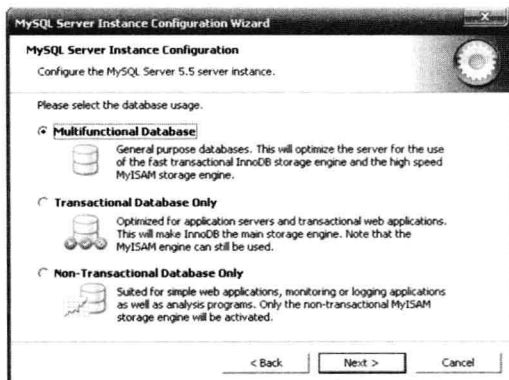


图 1.19 选择多功能数据库

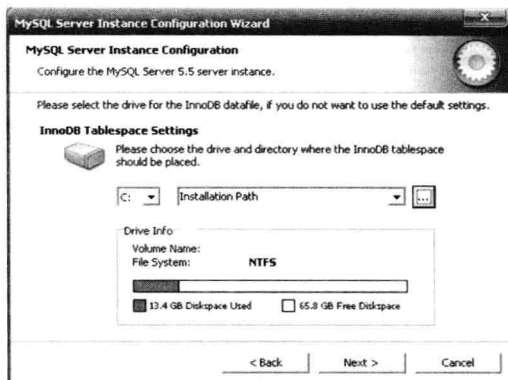


图 1.20 选择表空间存储位置

下一步选择“决策支持/OLAP”，即选中“Decision Support(DSS)/OLAP”单选按钮，如图 1.21 所示，然后单击 Next 按钮。设置网络端口号可以使用默认配置，只要不与本机端口号冲突，可以使用默认端口 3306，如图 1.22 所示，单击 Next 按钮继续。

下一步设置多语言字符集，MySQL 默认的字符集只适合英文和西欧语言，使用中文时需要设置多语言字符集，一般选用 UTF8 以支持多语言，确保选中 Best Support For Multilingualism 单选按钮，如图 1.23 所示，单击 Next 按钮继续。选中 Install As Windows Service 复选框，将 MySQL 安装为 Windows 服务；如果想自动加载 MySQL，应选中 Launch the MySQL Server automatically 复选框；默认的服务名为 MySQL，如果想改变，也在这一步进行设置，如图 1.24 所示，单击 Next 按钮继续。

下一步设置数据库管理员密码，确保两次输入的管理员密码是完全一致的，如图 1.25 所示，然后单击 Next 按钮。最后一步是执行数据库实例配置，等待四个配置步骤全部完成后，会出现四个蓝色的勾，单击 Finish 按钮结束 MySQL 的安装，如图 1.26 所示。