

“十五”国家重点电子出版物规划项目 计算机知识普及和软件开发系列

Java 应用技术丛书

3

JDBC 3.0

数据库开发与设计

北京希望电子出版社 总策划
梁陈剑 张威 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

“十五”国家重点电子出版物规划项目 计算机知识普及和软件开发系列

Java 应用技术丛书

JDBC 3.0

数据库开发与设计

北京希望电子出版社 总策划
梁陈剑 张威 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

JDBC 是目前使用很广泛的基于 Java 的数据库开发技术,随着 Java 语言使用的越来越普遍性, JDBC 的使用也越来越流行。利用 JDBC 可以提高数据库应用程序的开发效率,缩短开发周期,并且便于系统的维护和升级。

本书由 9 章组成,主要内容包括 JDBC 概述, JDBC 开发环境及配置, JDBC3.0 API 数据库开发的初、高级操作,二层结构数据库应用程序的开发,三层结构数据库应用程序的开发,综合开发实例分析, Java 与 XML 数据转换, JDBC 程序开发高级技术,并着重讲解了 JDBC 在开发三层结构数据库程序方面的技术。

本书语言浅显易懂,通过大量的实例一步一步地引导读者进入 JDBC 数据库应用程序开发的殿堂,读者可以通过模仿书中实例的开发过程,开发出自己的应用程序。

本版 CD 内容:全部实例源代码。

盘书系列名 : “十五”国家重点电子出版物规划项目·计算机知识普及和软件开发系列
Java 应用技术丛书(3)

盘 书 名 : JDBC 3.0 数据库开发与设计

总 策 划 : 北京希望电子出版社

文本著作者 : 梁陈剑 张威 编写

责 任 编 辑 : 柴 东

CD 制 作 者 : 希望多媒体开发中心

CD 测 试 者 : 希望多媒体测试部

出版、发行者 : 北京希望电子出版社

地 址 : 北京市海淀区知春路甲63号卫星大厦三层 100080

网址: www.bhp.com.cn

E-mail: lwm@bhp.com.cn

电 话: 010-62520290, 62521724, 62528991, 62630301, 62524940, 62521921, 82610344

(发行) 010-82675588-202 (门市), 010-82675588-501, 82675588-201 (编辑部)

经 销 : 各地新华书店、软件连锁店

排 版 : 希望图书输出中心 马伟科

文本印刷者 : 北京广益印刷有限公司

开本 / 规格 : 787 毫米×1092 毫米 1/16 33.625 印张 780 千字

版次 / 印次 : 2003 年 3 月第 1 版 2003 年 3 月第 1 次印刷

印 数 : 0001~3000 册

本 版 号 : ISBN 7-89498-130-3

定 价 : 50.00 元(本版 CD)

说明: 凡我社产品如有残缺,可执相关凭证与本社调换。

前 言

Java 数据库连接 (Java Database Connectivity, JDBC) API 是作为 Java 2 标准版 (Java 2 Standard Edition, J2SE) 和 Java 2 企业版 (Java 2 Enterprise Edition, J2EE) 平台的一个关键部分出现的。JDBC™ API 向 Java 编程语言提供了统一的数据访问方法。使用 JDBC 3.0 API, 开发者能访问从关系型数据库到表单到一般文件的几乎任何一种数据源。JDBC 技术同时提供了一个公共的基础, 其他工具和可替代的接口可建立在这个基础上。

本书读者对象

本书的阅读对象是所有对 JDBC 技术感兴趣的读者, 特别是希望能把握移动互联网所带来机遇的应用开发者。

JDBC 是使用 Java 开发数据库应用程序必须的技术, 本书假定读者已经有了一定 Java 语言基础。如果没有 Java 语言基础, 可以参见附录或找任何一本 Java 语言学习书籍掌握以下知识即可:

- Java 语言的基本数据类型及其运算。
- Java 语言基本控制语句。
- Java 类和对象的概念。

本书章节安排

本书章节安排如下:

第 1 章 JDBC 概述 主要介绍了 JDBC 相关的基本概念, 目的是让读者了解 JDBC 的规范、特点以及 JDBC 中的类和接口以及 JDBC 驱动程序等。

第 2 章 JDBC 开发环境及配置 主要介绍 JDK 的基本配置、JDBC 开发中需要用到的开发工具、数据库结构和 Web 服务器等, 并对一些基本技术进行初步的讲解。目的是让读者能对 JDBC 数据库应用开发前提有一个必要的了解。

第 3 章 JDBC 3.0 API 数据库基本操作 主要介绍 JDBC 3.0 API 数据库的一些基本操作, 包括程序与数据库的连接、操作数据库数据、获取数据库信息等内容, 目的是让读者能对 JDBC 操作数据库的基本技术有一个系统的了解, 并为后面章节的开发打下基础。

第 4 章 JDBC 3.0 API 数据库高级操作 本章和第 3 章紧密衔接, 主要讲述了 JDBC 对数据库的一些高级操作, 如存储过程、行集合、成批更新、JNDI、连接池等技术, 目的是让读者掌握 JDBC 开发数据库的更深层的技术。本章和第 3 章是后面各个章节学习的基础。

第 5 章 二层结构数据库开发实例 介绍了几个具体的二层数据库开发的实际例子。通过这些例子的介绍, 可以让读者能够了解到 JDBC 数据库开发的实际过程。

第 6 章 三层结构数据库开发实例 本章是在第 5 章的基础上进行的三层数据库开发实例介绍。通过本章的介绍, 读者应该可以对三层数据库开发的概念、模式以及常用方法

有一个比较完善的了解，并可以进行实际的开发。

第7章 综合开发实例分析 主要介绍了使用 UML 分析和建模并进行实际开发的实例。本章的侧重点在于使用 UML 的图形建模的过程。

第8章 Java 与 XML 数据转换 本章对 XML 的基础知识进行讲解，并用实例分析了 Java 和 XML 之间的一些数据信息交换、动态生成页面、制作动态的 XML 文档等内容进行了详细的介绍，以帮助读者了解 Java 和 XML 结合的数据库应用开发过程。

第9章 JDBC 程序开发高级技术 主要介绍了对 JDBC 开发中常见的一些高级开发技术及技巧以程序代码的形式，以帮助读者在实际开发中解决一些复杂的编程问题。

主要作者

本书主要作者：梁陈剑（有丰富的应用程序和数据库设计经验，曾参与过“教育管理信息系统”、“基于 Web 的专业防火墙”、“网络化异地制造系统”等多个项目的设计和开发工作）、张威。

致谢

参加本书编写工作的还有吴学跃、张臻、刘喆、王铁军、李伟、黄宇、蒲俊、潘陈佳、庞庆、刘祥、李晓、刘勇、王楠、施建中、李雪峰、李勇、唐峰、董威、郭伟、姚俊杰、吴波、胡平、崔炎、胡林、韦斌圣、王帅、苏斌、李杰、刘明、毛袁胜、张海、许健等。由于作者水平有限，时间仓促，错误和疏漏之处在所难免，恳请广大读者批评指正。

作 者

2002 年 11 月

只要您点击 www.bhp.com.cn 决不会空手而归!

北京希望电子出版社网站“书目信息”服务办法

为了帮助您尽早了解我社的新书出版信息,我社充分发挥自身计算机软件企业的技术优势,在本社的网站(www.bhp.com.cn)上开设了书目信息的服务项目,它可以使您每天与出版社同步得到新书出版信息,也可以为您提供本社数年来的全部书盘目录。

在网站查找、下载本社书盘目录的方法如下:

1、在网站的首页查找新书。在浏览器地址栏处输入 www.bhp.com.cn 之后便进入我社网站的首页,首页的右侧有“每周新书”栏目,该栏目每天更新,增添我社的当日新书。在首页上方的栏目条里,有“最近新书”栏目,包括近两个月的新书介绍。

2、在“书盘目录”栏目里,有本社图书光盘的分类目录。通过分类目录,可以找到同类图书光盘的目录和系列书目录,最后到达的是书的详细介绍,包括书的版权页信息、本书内容提要、目录等等。

3、在“经销商园地”栏目里,设有“书目下载”专栏。内容包括:1995年——2000年的书盘目录下载;1999年书盘目录下载;2000年书盘目录下载;2001年内书盘目录按出版月份下载。书目下载是专门为希望电子图书经销商,书店提供的信息服务,与我社的图书目录单内容相同,是图书目录单的电子版,您可按此书目向我社定购图书。

书目下载文件为 zip 格式的压缩文档,解压缩后为 Word 格式。下载的书目操作步骤是:

(1) 进入本社的网站(www.bhp.com.cn)“经销商园地”/“书目下载”栏目。

(2) 点击所选取的连接,出现对话框,选取“将该文件保存到磁盘”,点击“确定”。

(3) 在“另存为”对话框选择存放文件的路径,点击“保存”按钮后计算机便开始下载。

(4) 下载完毕,启动自己的“zip 软件”,沿着前面文件的存放路径找到该文件,解开压缩,打开文件阅读。

如果您的计算机里还没有 zip 软件,可以在我们的网站下载一个。地址是:首页/技术支持/工具箱/常用工具/Winzip80

《希望书盘交流俱乐部》会员须知

一、北京希望电子出版社的书盘立意新颖、风格迥异，受到广大读者喜爱。为了给长期热心支持和选购希望电脑书盘的朋友更多的回报，我社决定扩大《希望书盘交流俱乐部》，为此对读者入会条件和优惠政策作出如下调整：

1. 用户在本社一次性购买 100.00 元以上的书盘时，即可成为本俱乐部的会员，并在今后的购书中本市会员予以 八八折 优惠，外地会员购书 免加邮费同时优惠九折。会员将在本俱乐部建有个人档案。
2. 会员购书时必须出示此卡以便打折并累计金额。外地会员邮购图书时请把卡号写在汇款单的附言条上，以便累计。
3. 俱乐部会员投稿优先刊登本社报刊。
4. 俱乐部不定期组织会员参加各项活动。
5. 会员可优先得到我社的新书资料和信息。
6. 我社长期征稿，欢迎所有会员投稿。（题目为“我（不）喜爱的一本希望图书”、“我读 XX 书后的感想”，或对我社的图书选题有何感想都可写稿寄来。如果来稿被采用，便予以刊登并同时得到一份纪念品。）

二、会员卡另有储值功能，会员可随意存储金额，如需订购图书只需拨打订购电话（010-62613322-210）经确认预留金额后即可发书。这项功能缩短了会员邮寄图书的周期，使会员在最短的时间内收到图书。会员可随时查询余额或续款。

三、会员若连续一年未购书盘，会员卡自动取消，需按入会资格重新入会。

四、会员卡遗失后，由该卡的指定联系人办理补卡手续。

五、持卡人或联系人的通讯地址及联系方式发生变动时，请及时与俱乐部联系。

声明：2000 年 12 月 31 日以前拥有的有效〈希望交流俱乐部会员卡〉可持续使用，按原会员章程执行，不进行新卡替换（直到不履行会员须知，旧卡被合理取消为止）。2001 年 1 月 1 日〈希望交流俱乐部〉将启用新会员卡，按俱乐部新章程执行。

此卡最终解释权在北京希望电子出版社。

请用正楷认真填写此表，以便我们准确记录您的信息，与您及时联系

《希望书盘交流俱乐部》会员回执表

姓 名		年 龄		职 业				
工作单位					学 历			
通讯地址				邮 编			性 别	
				电 话				
E-Mail				卡 号				

北京希望电子出版社邮购部

目 录

第 1 章 JDBC 概述	1
1.1 JDBC 简介.....	1
1.2 JDBC3.0 规范.....	2
1.3 JDBC3.0 API 的新特点.....	4
1.3.1 JDBC 3.0 API 的一致性	5
1.3.2 不赞成的 API.....	5
1.4 JDBC 3.0 中的类和接口.....	6
1.4.1 java.sql 包中的类和 接口及其使用.....	6
1.4.2 javax.sql 包中所含内容 及其使用.....	8
1.5 JDBC 驱动程序简介.....	11
1.5.1 JDBC-ODBC 桥和 ODBC 驱动程序.....	12
1.5.2 本地 API 部分 Java 驱动程序.....	12
1.5.3 JDBC 网络纯 Java 驱动程序	13
1.5.4 本地协议纯 Java 驱动程序	13
1.5.5 JDBC 驱动程序的比较	14
1.6 本章小结	15
第 2 章 JDBC 开发环境及配置	16
2.1 数据库系统	16
2.1.1 SQL 数据库数据库基本 操作语句.....	16
2.1.2 Microsoft SQL Server 数据库设计及 ODBC 配置...	23
2.1.3 Oracle 数据库的基本 说明及其使用简介.....	32
2.1.4 MySQL 的安装及使用简介 ..	38
2.2 使用 JDBC 连接数据库	45
2.2.1 SQLJ 基本知识	46
2.2.2 用 PL/SQL 和 Java 建立应用 程序的基本知识	47
2.2.3 PL/SQL 和 Java 的特性	52
2.2.4 PL/SQL 和 Java 的实际开发应	

用实例.....	55
2.2.5 Java 连接各类数据库的 程序代码.....	56
2.3 Web 应用服务器.....	60
2.3.1 Tomcat 的安装及系统文件 设置.....	60
2.3.2 WebSphere 简介	71
2.3.3 WebLogic 的安装和配置 ...	76
2.3.4 Resin 的安装与配置	98
2.4 Java 开发环境.....	101
2.4.1 JDK 设置	101
2.4.2 JBuilder 的配置	104
2.4.3 Cafe 的配置	110
2.4.4 JDeveloper 的配置	114
2.4.5 FORTE FOR JAVA 简介	116
2.5 JDBC 开发调试技巧.....	118
2.6 本章小结	122
第 3 章 JDBC3.0 API 数据库开发 基本操作	123
3.1 数据库用例说明	123
3.2 与数据库建立连接	124
3.2.1 JDBC URL.....	124
3.2.2 连接事件 (Connection Event)	125
3.2.3 连接事件监听器 (Connection EventListener)	125
3.3 获取数据库信息	126
3.3.1 获取数据库详细信息.....	126
3.3.2 获取数据库基本信息.....	127
3.3.3 获取数据库中表的信息...	129
3.3.4 获取表中各列的信息.....	131
3.3.5 获取索引的信息.....	133
3.3.6 获取数据库中各种信息 的实例.....	134
3.4 Statement 对象.....	140

3.4.1	Statement 对象使用基础	141
3.4.2	获取所有记录	143
3.4.3	获取指定记录	145
3.5	修改数据库表	146
3.5.1	修改数据库表本身	146
3.5.2	修改数据库表记录	149
3.6	结果集	152
3.6.1	结果集的基本操作	153
3.6.2	获取结果集的信息	153
3.6.3	使用流获取大型数据	155
3.6.4	可滚动结果集	156
3.6.5	可更新结果集	158
3.7	SQL 和 Java 数据类型转换	162
3.8	本章小结	172
第 4 章	JDBC 3.0 API 数据库开发	
	高级操作	173
4.1	预编译语句	173
4.1.1	PreparedStatement 对象的创建	173
4.1.2	传递 IN 参数	174
4.1.3	传递对象参数	175
4.1.4	传递大型数据参数	176
4.2	调用存储过程对象	177
4.2.1	CallableStatement 对象的创建	178
4.2.2	IN、OUT 及 INOUT 参数 的使用	178
4.2.3	执行 CallableStatement Object 对象	180
4.2.4	CallableStatement 对象 使用实例	182
4.2.5	SQL Server 存储过程编程 经验技巧	186
4.3	成批更新 (BatchedUpdate)	187
4.3.1	成批更新所使用的对象	187
4.3.2	使用成批更新的实例	190
4.4	行集合对象	192
4.4.1	设计时行集合	192
4.4.2	运行时行集合	193

4.4.3	非标准 JDBC API 类的行集	193
4.4.4	行集合的接口	198
4.5	事务	208
4.5.1	事务提交 (Commit)	208
4.5.2	事务使用实例	211
4.6	JNDI 与 JDBC API	213
4.6.1	JDBC 数据源	213
4.6.2	实例	215
4.7	连接池	218
4.7.1	连接池数据源 (Connection PoolDataSource)	219
4.7.2	连接池处理事件	220
4.7.3	三层环境下的连接池 操作	221
4.7.4	连接池和 DataSource 实现	221
4.7.5	包含连接池的数据源配置	222
4.7.6	池连接对象对于语句的 重新利用	223
4.7.7	关闭池连接语句	224
4.7.8	连接池使用实例	225
4.8	分布式事务处理	243
4.8.1	XADataSource 接口和 XAConnection 接口	244
4.8.2	事务管理器	247
4.8.3	关闭连接	248
4.9	例外处理	249
4.9.1	数据访问警告 (SQLWarning)	249
4.9.2	数据截断 (DataTruncation)	249
4.10	高级数据类型	250
4.10.1	Java 对象类型	250
4.10.2	SQL3 类型	251
4.10.3	定制类型	255
4.10.4	实例	257
4.11	本章小结	264
第 5 章	二层结构数据库开发实例	265
5.1	二层结构原理	265

5.2	Applet 访问数据库实例.....	266
5.3	Application 访问数据库实例....	268
5.4	使用 Jdevelpoer 工具的数据库 开发实例	269
5.5	二层结构开发数据库的优缺点 ...	291
5.6	本章小结	291
第 6 章	三层结构数据库开发实例 ..	292
6.1	三层结构原理	292
6.2	JSP 数据库操作实例.....	293
6.3	Servlet 数据库操作实例.....	314
6.4	JavaBeans 操作数据库实例.....	337
6.5	三层结构开发数据库的优缺点 ...	342
6.6	本章小结	343
第 7 章	综合开发实例分析	344
7.1	UML 要点与应用.....	344
7.1.1	UML 定义及组织	344
7.1.2	UML 的概念模型	345
7.1.3	UML 的应用	347
7.2	用 UML 建模开发网上商店实例 ...	348
7.2.1	对网上商店的 UML 建模 分析.....	349
7.2.2	网上商店数据库设计.....	363
7.2.3	网上商店的程序实现.....	364
7.3	用 UML 建模开发图书馆系统实例 .	383
7.4	本章小结	392
第 8 章	JAVA 与 XML 数据转换	393
8.1	XML 的基础知识.....	393
8.1.1	XML 的特点	393
8.1.2	XML 文档的结构	397
8.1.3	XML DTD 的建立与使用 ...	400
8.1.4	文档显示与样式单.....	403
8.1.5	XML 链接语言	409
8.1.6	实用 XML 工具概览.....	413
8.2	XML 中数据与信息交换.....	415
8.2.1	XML 数据源	415
8.2.2	XML 在数据库中的应用 模式.....	416
8.2.3	XML 数据交换技术及应用 .	416
8.3	使用 JSP 和 XML 两种技术建设动态	

网站.....	418
8.3.1 XML 存储数据的优点.....	418
8.3.2 XML 文档中的 Picture 对象定义.....	418
8.3.3 JSP 文档与 XML 的结合...	420
8.4 用 XML 和 XSL 生成动态页面.....	422
8.4.1 建立标准的数据格式.....	423
8.4.2 生成动态 XML.....	423
8.4.3 用 XSL 作为模板语言.....	425
8.4.4 生成 HTML.....	427
8.5 从 XML 到 Java 代码的数据绑定 ..	427
8.5.1 访问方式选择.....	427
8.5.2 数据绑定.....	428
8.5.3 约束数据.....	429
8.5.4 从 XML 获得 Java.....	431
8.5.5 生成 Java 代码.....	433
8.5.6 完成包.....	439
8.6 使用 JSP 技术生成动态 XML.....	442
8.6.1 JSP 技术概述.....	442
8.6.2 使用 JSP 技术生成动态 XML 文档的结构.....	443
8.7 本章小结.....	448
第 9 章 JDBC 程序开发高级技术....	449
9.1 中文处理技术.....	449
9.1.1 Java 中文处理的理论知识	449
9.1.2 中文处理的解决方案.....	451
9.1.3 中文处理技术程序实例...	456
9.2 数据查询结果的分页显示技术...	459
9.2.1 使用 JavaBean 和连接池 实现数据分页显示.....	459
9.2.2 不用连接池不用 JavaBean 的分页显示.....	473
9.2.3 使用 JavaBean 不使用连接 池的分页显示.....	477
9.2.4 用 JSP 和 Servlet 配合实现 分页显示.....	484
9.3 数据库图形文件的存取技术.....	498
9.3.1 向数据库中存储图形.....	499
9.3.2 将数据库的图象二进制数据	

显示成图片.....	501		
9.4 Java 开发中的其他技术.....	503		
9.5 本章小结	510		
附录 A JSP 基本语法参考	511		
A.1 注释语句	511		
A.1.1 在客户端显示注释.....	511		
A.1.2 隐藏注释.....	511		
A.2 表达式	511		
A.3 声明语句	511		
A.4 代码段	512		
A.5 指令	512		
		A.5.1 include 指令	512
		A.5.2 page 指令	512
		A.5.3 taglib 指令	514
		A.6 XML 格式的 JSP 动作语法.....	515
		A.6.1 <jsp:include />.....	515
		A.6.2 <jsp:forward />.....	515
		A.6.3 <jsp: useBean />.....	519
		A.6.4 <jsp:setProperty />....	520
		A.6.5 <jsp:getProperty />....	521
		A.6.6 <jsp:plugin />.....	524

第 1 章 JDBC 概述

本章重点:

- JDBC 简介
- JDBC3.0 规范
- JDBC3.0 API 新特点
- JDBC3.0 中的类和接口
- JDBC 驱动程序简介

在本章中将对 JDBC 的历史、特点及一些基本概念等进行介绍。通过对本章的阅读,基本可以对 JDBC 的来历和功能等内容有一个较为全面的了解,这对理解和掌握本书后面章节的内容均有较大的帮助。

1.1 JDBC 简介

JDBC 是一种用于执行 SQL 语句的 Java API。JDBC 本身是个商标名而不是一个缩写字,但在通常情况下, JDBC 被认为是“Java 数据库连接 (Java Database Connectivity)”的缩写。

JDBC 由一组用 Java 编程语言编写的类和接口组成,由于 Java 编程语言具有坚固、安全、易于使用、易于理解和易于从网络自动下载等特性,Java 编程语言正逐渐成为编写数据库应用程序的大众编程语言。

JDBC API 向 Java 编程语言提供了统一的数据访问方法。使用 JDBC3.0 API,开发者能访问从关系型数据库到表单到一般文件的几乎任何一种数据源。JDBC 技术同时提供了一个公共的基础,其他工具和可替代的接口都建立在这个基础上。

JDBC 为数据库开发人员提供了一个标准的 API,使开发人员能够用纯 Java API 来编写数据库应用程序。通过使用 JDBC,在程序中实现向各种关系数据库发送 SQL 语句已经变成一个很容易的过程。换言之,有了 JDBC API,就不必为访问 Sybase 数据库专门写一个程序,为访问 Oracle 数据库又专门写一个程序,为访问 Informix 数据库再写另一个程序。开发人员只需用 JDBC API 写一个程序,就可以向任何相应的数据库发送 SQL 语句。同时,使用 Java 编程语言编写的应用程序可以跨平台使用,即在操作系统改变时,无需重新为新的操作系统编写不同的应用程序,这也使得采用 Java 和 JDBC 结合的方式进行开发的程序员只需写一遍程序就可让该程序在任何平台上运行。

随着越来越多的程序员开始使用 Java 编程语言,对 Java 访问数据库的便捷性要求也在日益增加。JDBC 扩展了 Java 的这个功能。JDBC API 使得程序员能够用 Java 语言来访问关系型数据库。通过使用 JDBC API,用 Java 编写的应用程序能够执行 SQL 语句、检索结果,并把修改后的结果回存到底层的数据源。使用 JDBC API 也能在分布式的异质环境下与多个数据源交互。JDBC API 基于 X/Open SQL CLI (Calling Level Interface: 调用级接口),这也是 ODBC 的基础。JDBC 提供了从 Java 语言到 X/Open CLI 和 SQL 标准中定义

的抽象类型的自然而便于使用的映射关系。

有了这样的功能,Java 编程语言的功能得到更大的增强。例如,用 Java 和 JDBC API 可以发布含有 Applet 的网页,而该 Applet 使用的信息可能来自远程数据库;企业也可以用 JDBC 通过 Intranet 将所有机器连到一个或多个内部数据库中(即使企业职员所用的计算机有 Windows, Macintosh 和 UNIX 等各种不同的操作系统)。

MIS 管理员们也喜欢将 Java 和 JDBC 结合使用,因为这样可以使信息传播变得容易且合理。通过这种结合方式,企业可以不必重新安装数据库而继续使用以前已安装好的数据库,并能同以前一样方便、快捷地存取信息,即使这些信息储存在不同数据库管理系统上。同时,通过这种结合方式,新程序的开发期可以得到较大的缩短,安装和版本控制将大为简化。程序员可只编写一遍应用程序或只更新一次,然后将改后的程序放到服务器上,随后就可以让任何人都能得到最新版本的应用程序。对于商务上的销售信息服务,Java 和 JDBC 还可为外部客户提供获取信息更新、更好的方法。

1. JDBC 3.0 API 所包括的内容

JDBC 3.0 API 包括 java.sql 包(JDBC 核心 API 包)和 javax.sql 包(JDBC 可选 API 包)。完整的 JDBC API 包括在 Java 2 SDK、标准版(J2SE1.4)中。javax.sql 包把 JDBC API 的功能从客户端的 API 扩展到服务器端的 API,是 Java 2 SDK、企业版(J2EE)技术的一个基本部分。

☛注意: J2EE 平台也包括完整的 JDBC API。JDBC 3.0 API 包括在 J2EE 1.3 版本中。

2. JDBC3.0 API 版本简介

由于 JDBC3.0 的一致性特点, JDBC 3.0 API 中集成了 JDBC API 的所有旧版本。这些版本包括如下一些:

- JDBC 2.1 Core API。
- JDBC 2.0 Optional Package API。
- JDBC 1.2 API。
- JDBC 1.0 API。

☛注意: JDBC 2.1 Core API 和 JDBC 2.0 Optional Package API 统称为 JDBC 2.0 API。

1.2 JDBC3.0 规范

从 1997 年 1 月第一次被引入至今, JDBC API 已经发展成为一种成熟的技术。在最初的发行版中, JDBC API 集中于提供一种基本的针对 SQL 数据库的调用级接口。JDBC 2.1 规范和 2.0 可选择包的规范扩大了 API 的范围,以包含进对更高级的应用程序和应用服务器代表它们的应用程序来管理 JDBC API 使用所需特点的支持。JDBC 3.0 规范的整体目的是加入一些小范围的未曾考虑的功能。这也是 JDBC API 的设计原则和 JDBC3.0 API 的特别之处。其中包括以下一些内容:

- 适应 J2EE 和 J2SE 平台。JDBC API 是 Java 平台的一个技术组成部分, JDBC 3.0 API 应该与 J2EE 和 J2SE 平台的整体方向保持一致。

- 与 SQL99 保持一致。JDBC API 提供用 Java 语言访问标准 SQL 的通路。在 JDBC 2.0 API 处于开发阶段时, SQL99 规范还没有一个固定的目标。SQL99 现在已经是经发布的标准, 其中包括受到数据库管理系统生产厂商广泛支持的特点。同时它也包括其他一些只受少数生产厂商支持的特点。JDBC 3.0 API 的目的是提供访问 SQL99 子集特点的方法, 而这些特点将会在未来的 5 年内得到广泛的支持。
- 融入先前的规范。该文档集成以前 3 种 JDBC 规范的内容, 以提供一个独立的 JDBC API 规范。
- 提供厂商独立的共同特点。JDBC API 致力于在不同的厂商中提供受到公共支持的、高带宽的访问特点。这一目标是为了提供一种在访问时能够达到与本地应用访问特点相同的访问特点。这一 API 必须是常用并且足够灵活的, 以允许广泛的实现形式。
- 继续以和 SQL 技术的配合为主要开发目标。JDBC API 的主要功能在于使用 Java 语言来访问关系型数据, 这一特点在 JDBC 3.0 API 中被继续保持下来。JDBC 3.0 API 并不排除与其他技术的交互, 比如 XML, CORBA 或者非关系型数据, 但主要的目标还是关系型数据和 SQL 技术。
- 为工具和高层 API 提供一个基础。JDBC API 代表了一个标准的、能访问不同底层数据源的 API。实现对不同底层数据源访问时所使用的不同技术在 JDBC API 的抽象上是透明的, 这给那些想要生产便携工具和应用程序的生产商们提供了一个很有价值的目标平台。因为它是一个从 Java 语言到 SQL 的调用级的接口, JDBC API 也可作为高层工具, 比如 EJB (Enterprise JavaBeans 2.0) 和 SQLJ 等的底层。
- 保持简单。JDBC API 应该是一个简单易用的直接接口, 在它上面可以构建更加复杂的实体。这一目标通过定义许多紧凑的、单一用途的方法, 而不是通过少量的、复杂的、多用途的、带标志位参数的方法来实现。
- 增强可靠性、有效性和可缩放性。可靠性、有效性和可缩放性是 J2EE 和 J2SE 平台的主题以及将来 Java 平台的方向。JDBC 3.0 API 通过增强对资源管理、跨越逻辑连接的预编译语句的重用及错误处理等范围的支持来保持这些主题。
- 保持对现有应用程序和驱动程序的向下兼容。这一特点是保证现有 JDBC 驱动程序和使用这些驱动的应用程序能够在支持 JDBC 3.0 API 的虚拟机上继续运行。那些仅使用早期版本 JDBC API 的应用程序 (不包括在 JDBC 2.0 中不赞成使用的 API), 不需要为了能继续运行而进行修改, 现有应用程序应该能够直接移植到 JDBC 3.0。
- 允许使用连接器 (Connector) 进行向上兼容。连接器体系结构定义了一个标准的方式来打包和部署一个资源适配器, 该适配器允许 J2EE 容器集成它的连接、事务、安全管理和外部资源, JDBC 3.0 API 提供了从 JDBC 驱动到连接器体系结构的移植路径。生产厂商有可能逐步地把 JDBC 技术的产品转移到用连接器 API 来实现。期望这些实现将会在现有的数据源实现上写上“资源管理包装”, 这样就可以在连接器框架上进行重用。
- 明确指明要求。JDBC 的要求应该是明确而易于识别的, JDBC 3.0 规范和 API 文

档将会说明这些要求的特点及哪些是可选的。

1.3 JDBC3.0 API 的新特点

同以往的 JDBC API 相比, JDBC 3.0 API 引入了一些更新的内容和特点。这些内容和特点主要如下:

- 对存储点 (Savepoint) 的支持。增加的存储点接口能够设置、释放或者回滚一个事务到指定的存储点上。
- 通过连接池对预编译语句进行重用。增加了控制预编译语句, 并使之被池化的功能, 该功能还可描述被池化的预编译语句是如何被连接重用的。
- 连接池配置。为 `ConnectionPoolDataSource` 接口定义一定数量的属性, 这些属性能够用来描述数据源对象创建的 `PooledConnection` 对象如何被池化。
- 数据元参数的检索, 增加的 `ParameterMetaData` 接口描述预编译语句中参数的数量、类型和属性。
- 对自动生成键的检索, 增加了一种从包含了自动生成值的列中检索值的方法。
- 保持多个打开记录集对象的能力。增加的新方法: `getMoreResults (int)`, 该方法中, 由参数指定 `Statement` 对象返回的记录集对象在返回任何一个连续的记录集对象前是否需要关闭。
- 通过名字向 `CallableStatement` 对象传递参数, 增加的方法允许以一个字符串来识别将要设定的 `CallableStatement` 对象的参数。
- 支持可保持的游标, 增加了指定一个记录集对象的可保持性功能。
- 增加 `BOOLEAN` 型数据类型。增加数据类型 `java.sql.Types.BOOLEAN`, `BOOLEAN` 在以往的 JDBC API 中是不被支持的, 在 JDBC3.0 中, 其在逻辑上与 `BIT` 等同。
- 增加在 `Blob` 和 `Clob` 对象中进行内部修改的方法, 增加的方法允许 `Blob` 和 `Clob` 对象中的数据可以被修改。
- 通过 `Ref` 对象来检索和更新它所引用的对象。增加方法来检索 `Ref` 对象引用的对象, 也增加了通过 `Ref` 对象来更新它所引用的对象的功能。
- 更新包含有 `Blob`, `Clob`, `Array` 和 `REF` 类型的列。在 `ResultSet` 接口中加入了 `updateBlob`, `updateClob`, `updateArray` 和 `updateRef` 方法。
- 增加 `DATALINK/URL` 数据类型。增加的 `java.sql.Types.DATALINK` 数据类型允许 JDBC 驱动程序存储和检索外部数据的引用。
- 转换组和类型映射。描述转换组的影响, 并描述该影响是如何在元数据中体现出来的。
- JDBC SPI (Service Provider Interface) 和连接器 (Connector) 体系结构的关系, 描述 JDBC SPI 和连接器体系结构之间的关系。
- `DatabaseMetadata` APIs, 增加元数据检索 `SQL` 类型的层次。

注意: JDBC3.0 中许多新特点是可选的,这就导致不同 JDBC 驱动程序在实现上会有一些的差别。这些不同的 JDBC 驱动程序所支持的特点也不同,具体的特点请查看所使用的驱动程序文档。

除了以上新特点和内容外,作为一种新推出的升级产品,JDBC3.0 API 还具有以下两个较为突出的特性。

1.3.1 JDBC 3.0 API 的一致性

与 JDBC 3.0 API 一致的驱动程序必须做到以下几点:

(1) 与 JDBC2.0 API 的要求一致。

(2) 包括下列接口:

- java.sql.ParameterMetaData。
- java.sql.Savepoint。

(3) 完全实现 DatabaseMetaData 接口,包括下列 JDBC 3.0 API 中增加的方法:

- supportsSavepoints。
- supportsNamedParameters。
- supportsMultipleOpenResults。
- supportsGetGeneratedKeys。
- getSuperTypes。
- getSuperTables。
- getAttributes。
- getResultSetHoldability。
- supportsResultSetHoldability。
- getSQLStateType。
- getDatabaseMajorVersion。
- getDatabaseMinorVersion。
- getJDBCMajorVersion。
- getJDBCMinorVersion。

1.3.2 不赞成的 API

不赞成指的是在将来的版本中不再存在,所以不推荐使用的类、方法、接口、构造器等。下面所列出的构造器和方法是在 JDBC 2.0 API 中不赞成使用的。

- java.sql.CallableStatement.getBigDecimal (int,int) 。
- java.sql.Date (int,int,int) 。
- java.sql.Date.getHours () 。
- java.sql.Date.getMinutes () 。
- java.sql.Date.getSeconds () 。
- java.sql.Date.setHours (int) 。
- java.sql.Date.setMinutes (int) 。
- java.sql.Date.setSeconds (int) 。