

Java网络编程

核心技术详解



孙卫琴 编著



- 服务器和客户程序通信内幕 · 线程池和异步通信
- 分布式软件架构 · 对象的序列化 · 访问数据库
- 访问邮件服务器 · 基于Spring等的Web服务
- 灵活运用设计模式 · 经典实用编程范例和练习题

Java 网络编程

核心技术详解



孙卫琴 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书结合大量典型的实例，详细介绍了用Java来编写网络应用程序的技术。本书内容包括：Java网络编程的基础知识、套接字编程、非阻塞通信、创建HTTP服务器与客户程序、数据报通信、对象的序列化与反序列化、Java反射机制、RMI框架、JDBC API、MVC设计模式、JavaMail API、安全网络通信、XML数据处理和Web服务。

阅读本书，读者不仅可以掌握网络编程的实用技术，还可以进一步提高按照面向对象的思想来设计和开发Java软件的能力。本书适用于所有Java编程人员，包括Java初学者及资深Java开发人员。本书还可作为高校的Java教材，以及企业Java培训教材。

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

Java 网络编程核心技术详解：视频微课版/孙卫琴编著. —北京：
电子工业出版社，2020.3
ISBN 978-7-121-38315-1

I. ①J… II. ①孙… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字（2020）第 010548 号

责任编辑：孙学瑛

印 刷：山东华立印务有限公司

装 订：山东华立印务有限公司

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：787×1092 1/16 印张：37.5 字数：960 千字

版 次：2020 年 3 月第 1 版

印 次：2020 年 3 月第 1 次印刷

定 价：129.00 元

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）88254888，88258888。

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

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

推荐序

在 IT 行业，大多数 Java 程序员都看过孙卫琴老师的书。孙老师的书，清晰严谨，把复杂的技术架构层层剖析，结合典型的实例细致讲解，读者只要静下心来好好品读，就能深入 Java 技术的殿堂，领悟其中的核心思想，并掌握开发实际应用的种种技能。

读好书，犹如和名师面对面交流，可以全面地学习和传承名师在这个技术领域里的经验和学识。孙老师及其同仁孜孜不倦地钻研 Java 技术，紧跟技术前沿，传道授业、著书立说。无数程序员从中受益，从 Java 小白成长为 Java 大牛。

Oracle 作为 Java 领域的技术引领者和规范制定者，非常欢迎中国的作者把最新的 Java 技术介绍给广大 Java 开发人员，孙老师的书刚好满足了这一需求。如今，Java 在网络应用开发领域得到了非常广泛的运用，这本书深入浅出地介绍了套接字编程、非阻塞通信、数据报通信、RMI 框架、安全网络通信和 Web 服务等网络编程技术。读者如果希望成为高级 Java 开发人员，本书是必备的参考手册和学习宝典。

甲骨文人才产业基地作为 Oracle 在中国的业务拓展公司，非常欣赏这本书，许多老师和学员都用本书作为首选的 Java 网络编程参考书。相信读者能够从中受益匪浅，轻松上手，循序渐进，最后精通技术。

王正平

甲骨文人才产业基地教育产品部总监

前 言

Java 语言是第 1 个完全融入网络的语言。Java 语言之所以适合编写网络应用程序，归功于它的以下三方面的优势。

(1) Java 语言与生俱来就是与平台无关的。Java 程序能够运行在不同的平台上，运行在不同平台上的 Java 程序之间能够方便地进行网络通信。

(2) Java 语言具有完善的安全机制，可以对程序进行权限检查，这对网络程序至关重要。

(3) JDK 类库提供了丰富的网络类库（如套接字 API、JavaMail API 和 JDBC API 等），大大简化了网络程序的开发过程。

本书将展示如何利用 Java 网络类库来快速便捷地创建网络应用程序，致力于完成以下任务。

- 实现访问 HTTP 服务器的客户程序。
- 实现 HTTP 服务器。
- 实现多线程的服务器，以及非阻塞的服务器。
- 解析并展示 HTML 页面。
- 通过 JDBC API 访问数据库。
- 通过 JavaMail API 接收和发送电子邮件。
- 利用 RMI 框架实现分布式的软件系统。
- 进行安全的网络通信，对数据加密，验证身份，保证数据的完整性。
- 运用第三方开源软件框架，如 Axis、Spring 和 CXF，开发 Web 服务，实现分布式的软件系统。

本书的组织结构和主要内容

本书结合大量典型的实例，详细介绍了用 Java 来编写网络应用程序的技术。本书内容包括：Java 网络编程的基础知识、套接字编程、非阻塞通信、创建 HTTP 服务器与客户程序、数据报通信、对象的序列化与反序列化、Java 反射机制、RMI 框架、JDBC API、MVC 设计模式、JavaMail API、安全网络通信、XML 数据处理和 Web 服务。图 1 展示了本书各章之间的循序渐进关系。

从图 1 可以看出，本书第 1 章介绍了分层的网络体系结构，Java 网络程序位于最上层——应用层，并且通过套接字访问底层网络，也可以说，套接字为应用层封装了底层网络传输数据的细节。套接字（Socket）是 Java 网络编程的基础，第 2 章和第 3 章分别详细介绍了 Socket 与 ServerSocket 的用法。Java 网络程序都采用客户/服务器模式，客户端发出获得

特定服务的请求，服务器接收请求，执行客户端所请求的操作，然后向客户端发回响应。本书在介绍服务器端编程时，探讨了服务器端实现并发响应多个客户请求的两种方式：一种方式是运用线程池（第3章），还有一种方式是采用非阻塞通信（第4章）。在介绍客户端编程时，介绍了JDK提供的一种通用的客户端协议处理框架（第6章）。

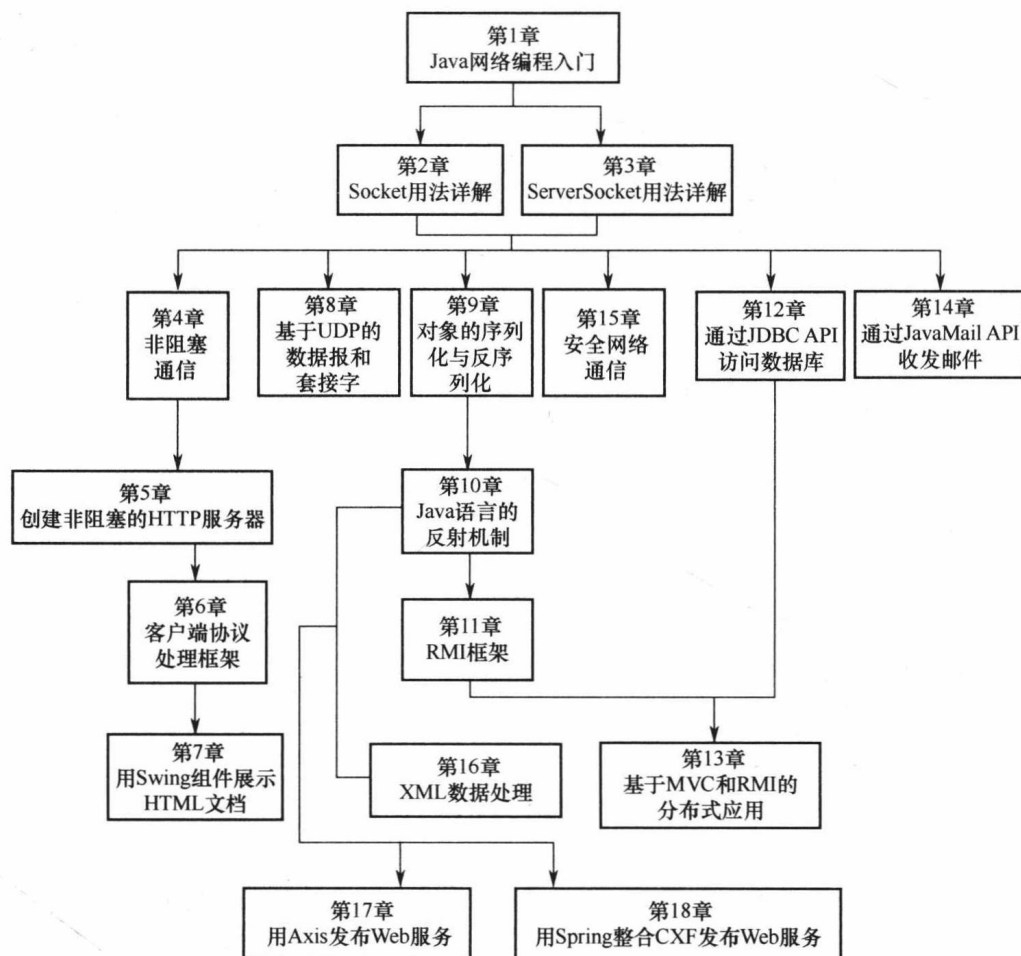


图1 本书各章之间的顺序渐进关系

利用Java网络API，可以实现基于各种应用层协议（比如HTTP和FTP）的服务器程序与客户程序，本书侧重介绍了HTTP服务器（第5章）与HTTP客户程序（第7章）的实现方法，HTTP客户程序也被称为浏览器。

本书还介绍了两种分布式的软件架构：RMI（第11章）和Web服务（第17章和第18章）。这些分布式架构主要解决的问题是，如何让客户端调用服务器端的远程对象。RMI是JDK自带的，它要求客户端与服务器端都是Java程序，而Web服务允许用任意编程语言编写的客户程序与服务器程序之间能够通信。本书详细介绍了RMI框架的用法。RMI框架在其实现中封装了用套接字通信的细节，此外，RMI框架的实现会把客户端的方法调用请求信息序列化为字节序列，把它发送给服务器端，然后在服务器端再通过反序列化把字节序列还原为方法调用请求。RMI框架还运用了动态代理机制，为客户端提供了远程对象的代理。客户端实际上直接访问的是远程对象的代理。为了帮助读者理解RMI框架的

实现原理，本书第 9 章和第 10 章分别介绍了 Java 序列化以及反射机制。在介绍反射机制时，还介绍了动态代理。

本书第 17 章和第 18 章介绍了在开源软件框架 Axis、Spring 和 CXF 中创建和发布 Web 服务的方法。这些框架软件封装了客户端和服务端底层通信的细节，使得开发人员只要利用框架软件的 API、注解和配置文件，就能方便地编写与具体业务领域相关的服务程序和客户端程序。

本书还介绍了两个常用的客户端的网络 API：JDBC API（第 12 章）和 JavaMail API（第 14 章），这两个 API 分别用于访问数据库服务器和邮件服务器，在它们的实现中都封装了用套接字与服务器通信的细节。Java 客户端程序可以通过 JDBC API 来访问各种数据服务器，还可以通过 JavaMail API 来访问各种邮件服务器。

本书第 13 章介绍了一个运用了 MVC 设计模式和 RMI 框架的综合应用。MVC 设计模式把实际的软件应用分为视图、控制器和模型 3 个层次，每个层次相对独立。本书的范例把模型作为远程对象放到 RMI 的服务器端，把视图和控制器放到 RMI 的客户端。

本书第 15 章介绍了 SSLSocket，它支持 SSL (Server Socket Layer) 协议和 TLS (Transport Layer Security) 协议。运用 SSLSocket，可以实现安全的网络通信，网络上传输的是被加密的数据，并且通信两端还能验证对方的身份。

本书在介绍以上技术时，采用 UML 建模语言中的类框图和时序图来展示对象模型，以及类与类之间的协作关系。此外，本书还把一些常见的设计模式，如静态代理模式、动态代理模式和 MVC 设计模式等运用到实际例子中。阅读本书，读者不仅可以掌握 Java 网络编程的实用技术，还可以进一步提高按照面向对象的思想来设计和编写 Java 软件的能力。

这本书是否适合您

阅读本书，要求读者已经具备了 Java 编程的基础知识。对于不熟悉 Java 语言的读者，建议先阅读本书作者的另一本书《Java 面向对象编程》，本书是它的姊妹篇。《Java 面向对象编程》自 2006 年 7 月出版后，一直畅销至今，受到了广大 IT 读者的欢迎。本书围绕着网络编程，进一步介绍了 Java 语言的一些高级特性，这些特性是作为一个高级 Java 开发人员必须知晓的。深入了解这些高级特性，有助于开发人员熟练地开发分布式的软件系统，或者轻松地学习和掌握现有的分布式软件架构。

本书一方面由浅入深地组织内容，满足 Java 网络编程初学者的需求，另一方面与实际项目紧密结合，介绍了线程池、非阻塞通信和动态代理等高级话题，可作为 Java 开发人员的参考手册。本书还可以作为高校的 Java 教材，以及企业培训教材。

致谢

本书在编写过程中得到了 Oracle 公司在技术上的大力支持。此外，JavaThinker.net 网站的网友为本书的编写提供了有益的帮助，在此表示衷心的感谢！尽管我们尽了最大努力，

但本书难免会有不妥之处，欢迎各界专家和读者朋友批评指正。以下网址是作者为本书提供的技术支持网址，读者可通过它下载与本书相关的资源（如源代码、软件安装程序和视频课程、PPT 讲义等），还可以与其他读者交流学习心得，以及对本书提出宝贵意见。

<http://www.javathinker.net/javanet.jsp>



读者服务

微信扫码：38315

- 获取博文视点学院 20 元付费内容抵扣券
- 获取本书配套 500+分钟的视频微课，以及配套源代码
- 获取免费增值资源
- 加入读者交流群，与更多读者互动
- 获取精选书单推荐

目 录

第 1 章	Java 网络编程入门	1
1.1	进程之间的通信	1
1.2	计算机网络的概念	3
1.3	OSI 参考模型	5
1.4	TCP/IP 参考模型和 TCP/IP	7
1.4.1	IP	10
1.4.2	TCP 以及端口	14
1.4.3	RFC 简介	15
1.4.4	客户/服务器通信模式	16
1.5	用 Java 编写客户/服务器程序	17
1.5.1	创建 EchoServer	18
1.5.2	创建 EchoClient	20
1.6	小结	23
1.7	练习题	24
第 2 章	Socket 用法详解	26
2.1	构造 Socket	26
2.1.1	设定等待建立连接的超时时间	27
2.1.2	设定服务器的地址	28
2.1.3	设定客户端的地址	28
2.1.4	客户连接服务器时可能抛出的异常	29
2.1.5	使用代理服务器	33
2.1.6	InetAddress 地址类的用法	33
2.1.7	NetworkInterface 类的用法	36
2.2	获取 Socket 的信息	36
2.3	关闭 Socket	39
2.4	半关闭 Socket	40
2.5	设置 Socket 的选项	45
2.5.1	TCP_NODELAY 选项	46
2.5.2	SO_REUSEADDR 选项	46
2.5.3	SO_TIMEOUT 选项	47
2.5.4	SO_LINGER 选项	50

2.5.5	SO_RCVBUF 选项	53
2.5.6	SO_SNDBUF 选项	53
2.5.7	SO_KEEPALIVE 选项	53
2.5.8	OOBINLINE 选项	54
2.5.9	IP 服务类型选项	54
2.5.10	设定连接时间、延迟和带宽的相对重要性	55
2.6	发送邮件的 SMTP 客户程序	56
2.7	小结	61
2.8	练习题	62
第 3 章	ServerSocket 用法详解	64
3.1	构造 ServerSocket	64
3.1.1	绑定端口	64
3.1.2	设定客户连接请求队列的长度	65
3.1.3	设定绑定的 IP 地址	68
3.1.4	默认构造方法的作用	68
3.2	接收和关闭与客户的连接	68
3.3	关闭 ServerSocket	69
3.4	获取 ServerSocket 的信息	70
3.5	ServerSocket 选项	72
3.5.1	SO_TIMEOUT 选项	72
3.5.2	SO_REUSEADDR 选项	73
3.5.3	SO_RCVBUF 选项	74
3.5.4	设定连接时间、延迟和带宽的相对重要性	75
3.6	创建多线程的服务器	75
3.6.1	为每个客户分配一个线程	76
3.6.2	创建线程池	78
3.6.3	使用 JDK 类库提供的线程池	84
3.6.4	向线程池提交有异步运算结果的任务	85
3.6.5	使用线程池的注意事项	89
3.7	关闭服务器	91
3.8	小结	95
3.9	练习题	96
第 4 章	非阻塞通信	98
4.1	线程阻塞的概念	98
4.1.1	线程阻塞的原因	98
4.1.2	服务器程序用多线程处理阻塞通信的局限	99
4.1.3	非阻塞通信的基本思想	101

4.2	非阻塞通信 API 的用法	102
4.2.1	缓冲区	103
4.2.2	字符编码 Charset	106
4.2.3	通道	106
4.2.4	SelectableChannel 类	108
4.2.5	ServerSocketChannel 类	109
4.2.6	SocketChannel 类	109
4.2.7	Selector 类	112
4.2.8	SelectionKey 类	114
4.2.9	Channels 类	116
4.2.10	Socket 选项	117
4.3	服务器编程范例	119
4.3.1	创建阻塞的 EchoServer	120
4.3.2	创建非阻塞的 EchoServer	123
4.3.3	在 EchoServer 中混合用阻塞模式与非阻塞模式	131
4.4	客户端编程范例	135
4.4.1	创建阻塞的 EchoClient	135
4.4.2	创建非阻塞的 EchoClient	137
4.5	异步通道和异步运算结果	141
4.6	在 GUI 中用 SwingWorker 实现异步交互	147
4.6.1	SwingWorker 类的用法	150
4.6.2	用 SwingWorker 类来展示进度条	152
4.6.3	用 SwingWorker 类实现异步的 AsynEchoClient	154
4.7	小结	155
4.8	练习题	156
第 5 章	创建非阻塞的 HTTP 服务器	159
5.1	HTTP 简介	159
5.1.1	HTTP 请求格式	160
5.1.2	HTTP 响应格式	162
5.1.3	测试 HTTP 请求	163
5.2	创建非阻塞的 HTTP 服务器	168
5.2.1	服务器主程序: HttpServer 类	168
5.2.2	具有自动增长的缓冲区的 ChannelIO 类	170
5.2.3	负责处理各种事件的 Handler 接口	172
5.2.4	负责处理接收连接就绪事件的 AcceptHandler 类	172
5.2.5	负责接收 HTTP 请求和发送 HTTP 响应的 RequestHandler 类	173
5.2.6	代表 HTTP 请求的 Request 类	175

5.2.7	代表 HTTP 响应的 Response 类	178
5.2.8	代表响应正文的 Content 接口及其实现类	180
5.2.9	运行 HTTP 服务器	183
5.3	小结	183
5.4	练习题	185
第 6 章	客户端协议处理框架	186
6.1	客户端协议处理框架的主要类	186
6.2	在客户程序中运用协议处理框架	187
6.2.1	URL 类的用法	187
6.2.2	URLConnection 类的用法	189
6.3	实现协议处理框架	194
6.3.1	创建 EchoURLConnection 类	195
6.3.2	创建 EchoURLStreamHandler 及工厂类	196
6.3.3	创建 EchoContentHandler 类及工厂类	197
6.3.4	在 EchoClient 类中运用 ECHO 协议处理框架	200
6.4	小结	201
6.5	练习题	202
第 7 章	用 Swing 组件展示 HTML 文档	204
7.1	在按钮等组件上展示 HTML 文档	205
7.2	用 JEditorPane 组件创建简单的浏览器	207
7.2.1	处理 HTML 页面上的超级链接	208
7.2.2	处理 HTML 页面上的表单	209
7.2.3	创建浏览器程序	211
7.3	小结	215
7.4	练习题	216
第 8 章	基于 UDP 的数据报和套接字	218
8.1	UDP 简介	218
8.2	DatagramPacket 类	222
8.2.1	选择数据报的大小	223
8.2.2	读取和设置 DatagramPacket 的属性	223
8.2.3	数据格式的转换	224
8.2.4	重用 DatagramPacket	225
8.3	DatagramSocket 类	227
8.3.1	构造 DatagramSocket	227
8.3.2	接收和发送数据报	228
8.3.3	管理连接	229

8.3.4	关闭 DatagramSocket	229
8.3.5	DatagramSocket 的选项	230
8.3.6	IP 服务类型选项	232
8.4	DatagramChannel 类	232
8.4.1	创建 DatagramChannel	232
8.4.2	管理连接	232
8.4.3	用 send()方法发送数据报	233
8.4.4	用 receive()方法接收数据报	233
8.4.5	用 write()方法发送数据报	238
8.4.6	用 read()方法接收数据报	238
8.4.7	Socket 选项	242
8.5	组播	243
8.5.1	MulticastSocket 类	245
8.5.2	组播 Socket 的范例	248
8.6	小结	251
8.7	练习题	251
第 9 章	对象的序列化与反序列化	254
9.1	JDK 类库中的序列化 API	254
9.1.1	把对象序列化到文件	256
9.1.2	把对象序列化到网络	258
9.2	实现 Serializable 接口	260
9.2.1	序列化对象图	263
9.2.2	控制序列化的行为	265
9.2.3	readResolve()方法在单例类中的运用	273
9.3	实现 Externalizable 接口	275
9.4	可序列化类的不同版本的序列化兼容性	277
9.5	小结	280
9.6	练习题	280
第 10 章	Java 语言的反射机制	283
10.1	Java Reflection API 简介	283
10.2	在远程方法调用中运用反射机制	289
10.3	代理模式	294
10.3.1	静态代理类	294
10.3.2	动态代理类	296
10.3.3	在远程方法调用中运用代理类	300
10.4	小结	305
10.5	练习题	307

第 11 章 RMI 框架	309
11.1 RMI 的基本原理	310
11.2 创建第 1 个 RMI 应用	312
11.2.1 创建远程接口	312
11.2.2 创建远程类	313
11.2.3 创建服务器程序	315
11.2.4 创建客户程序	318
11.2.5 运行 RMI 应用	320
11.3 远程对象工厂设计模式	323
11.4 远程方法中的参数与返回值传递	328
11.5 回调客户端的远程对象	332
11.6 远程对象的并发访问	337
11.7 分布式垃圾收集	341
11.8 远程对象的 equals()、hashCode()和 clone()方法	346
11.9 使用安全管理器	346
11.10 RMI 应用的部署以及类的动态加载	348
11.11 远程激活	350
11.12 小结	356
11.13 练习题	357
第 12 章 通过 JDBC API 访问数据库	359
12.1 JDBC 的实现原理	360
12.2 安装和配置 MySQL 数据库	362
12.3 JDBC API 简介	364
12.4 JDBC API 的基本用法	368
12.4.1 处理字符编码的转换	372
12.4.2 把连接数据库的各种属性放在配置文件中	374
12.4.3 管理 Connection、Statement 和 ResultSet 对象的生命周期	377
12.4.4 执行 SQL 脚本文件	382
12.4.5 处理 SQLException	385
12.4.6 输出 JDBC 日志	386
12.4.7 获得新插入记录的主键值	386
12.4.8 设置批量抓取属性	387
12.4.9 检测驱动器使用的 JDBC 版本	388
12.4.10 元数据	388
12.5 可滚动以及可更新的结果集	390
12.6 行集	398
12.7 调用存储过程	405

12.8	处理 Blob 和 Clob 类型数据	407
12.9	控制事务	412
12.9.1	事务的概念	412
12.9.2	声明事务边界的概念	414
12.9.3	在 mysql.exe 程序中声明事务	416
12.9.4	通过 JDBC API 声明事务边界	417
12.9.5	保存点	419
12.9.6	批量更新	420
12.9.7	设置事务隔离级别	422
12.10	数据库连接池	424
12.10.1	创建连接池	424
12.10.2	DataSource 数据源	432
12.11	小结	434
12.12	练习题	435
第 13 章	基于 MVC 和 RMI 的分布式应用	439
13.1	MVC 设计模式简介	439
13.2	store 应用简介	441
13.3	创建视图	447
13.4	创建控制器	455
13.5	创建模型	458
13.6	创建独立应用	462
13.7	创建分布式应用	464
13.8	小结	466
13.9	练习题	467
第 14 章	通过 JavaMail API 收发邮件	470
14.1	E-mail 协议简介	470
14.1.1	SMTP	471
14.1.2	POP3	471
14.1.3	接收邮件的新协议 IMAP	471
14.1.4	MIME 简介	472
14.2	JavaMail API 简介	472
14.3	建立 JavaMail 应用程序的开发环境	474
14.3.1	获得 JavaMail API 的类库	474
14.3.2	安装和配置邮件服务器	475
14.4	创建 JavaMail 应用程序	477
14.5	身份验证	482
14.6	授权码验证	485

14.7	URLName 类	488
14.8	创建和读取复杂电子邮件	490
14.8.1	邮件地址	491
14.8.2	邮件头部	493
14.8.3	邮件标记	493
14.8.4	邮件正文	495
14.9	操纵邮件夹	501
14.10	小结	507
14.11	练习题	508
第 15 章	安全网络通信	510
15.1	SSL 简介	510
15.1.1	加密通信	511
15.1.2	安全证书	511
15.1.3	SSL 握手	512
15.1.4	创建自我签名的安全证书	514
15.2	JSSE 简介	516
15.2.1	KeyStore、KeyManager 与 TrustManager 类	519
15.2.2	SSLContext 类	520
15.2.3	SSLServerSocketFactory 类	521
15.2.4	SSLSocketFactory 类	521
15.2.5	SSLSocket 类	521
15.2.6	SSLServerSocket 类	525
15.2.7	SSLEngine 类	526
15.3	创建基于 SSL 的安全服务器和安全客户	531
15.4	小结	536
15.5	练习题	536
第 16 章	XML 数据处理	539
16.1	用 DOM 处理 XML 文档	539
16.2	用 SAX 处理 XML 文档	542
16.2.1	创建 XML 文档的具体处理类 CustomerHandler	544
16.2.2	创建 XML 文档的解析类 SaxDemo	546
16.3	用 JDOM 处理 XML 文档	547
16.4	用 DOM4J 处理 XML 文档	551
16.5	Java 对象的 XML 序列化和反序列化	554
16.6	小结	556
16.7	练习题	556

第 17 章 用 Axis 发布 Web 服务	558
17.1 SOAP 简介	558
17.2 建立 Apache Axis 环境	560
17.3 在 Tomcat 上发布 Apache-Axis Web 应用	561
17.4 创建 SOAP 服务	562
17.4.1 创建提供 SOAP 服务的 Java 类	562
17.4.2 创建 SOAP 服务的发布描述文件	562
17.5 发布和管理 SOAP 服务	563
17.5.1 发布 SOAP 服务	563
17.5.2 管理 SOAP 服务	565
17.6 创建和运行 SOAP 客户程序	567
17.7 小结	569
17.8 练习题	570
第 18 章 用 Spring 整合 CXF 发布 Web 服务	571
18.1 创建 Web 服务接口和实现类	571
18.2 在 Spring 配置文件中配置 Web 服务	572
18.3 在 web.xml 配置文件中配置 Spring 和 CXF	572
18.4 在 Tomcat 中发布 Web 服务	573
18.5 创建和运行客户程序	575
18.6 小结	576
18.7 练习题	576
附录 A 本书范例的运行方法	577
A.1 本书所用软件的下载地址	577
A.2 部分软件的安装	578
A.2.1 安装 JDK	578
A.2.2 安装 ANT	579
A.2.3 安装 Tomcat	579
A.3 编译源程序	580
A.4 运行客户/服务器程序	580
A.5 处理编译和运行错误	581