

<<Java TCP/IP Socket编程>>

图书基本信息

书名：<<Java TCP/IP Socket编程>>

13位ISBN编号：9787111257561

10位ISBN编号：7111257561

出版时间：2009-1

出版时间：机械工业出版社

作者：（美）卡尔弗特，（美）多纳霍 著

页数：172

译者：周恒民

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

前言

如今，TCP / IP已成为计算机网络协议事实上的标准，而Java凭借其跨平台特性和对网络编程的强大支持能力，在网络应用中已占据了主导地位。

本书基于TCP / IP套接字的相关原理，对如何在Java中进行套接字编程作了深入浅出的介绍。

本书内容简明扼要，条理清晰，并在讲解相应的概念或编程技巧时列举了大量的示例程序，能够使读者在动手过程中加深理解，而每章结束时的练习可以帮助读者检查自己对已学知识的掌握程度，因此非常适合作为Java套接字编程的入门教程。

虽然本书专注于介绍如何使用Java进行TCP / IP套接字编程，但其涉及的套接字相关概念和基本原理与具体编程语言无关，从而使读者能够抓住套接字编程的本质，并轻松地转向其他编程语言。

译者在翻译本书时尽量忠实于原文，必要时对原书中提到的概念作了一定的解释，并力求做到言简意赅。

限于水平，翻译过程中难免有疏漏之处，敬请广大读者批评指正。

<<Java TCP/IP Socket编程>>

内容概要

在TCP/IP Socket首次发布时，Java作为一个新的未被关注的编程语言刚出现。然而，自2001年以来，Java的网络编程能力已经得到了很大程度的扩展，而且持续的应用优势已经使其成为了一个占主导地位的平台。考虑到Java受欢迎程度和近期在形式上和功能上令人兴奋的创新，本书的新版涵盖了Java的当前最新版本的内容（7.5版至1.7版），以及一些新的重要的抽象，包括NetworkInterface，InterfaceAddress，Inet4 / 6Address抽象、SocketAddress InetSocketAddress]Executor等。现在几乎所有互联网应用程序都使用Java Socket进行通信，而本书专注的、教导式的方法，使读者能够快速掌握所有计算机通信的任务和技术要点。

本书特点 对Socket编程的重点进行了集中简明的介绍，以使读者能够迅速掌握；并且使工程师们能够迅速地将所学的技能应用到他们的日常工作中。

对最新的Java平台（1.7）进行了简明而紧随其发展的介绍，使开发者即使在最新的技术面前也能处理大量未探究的陷阱。

配套网站（[books.elsevier.com / companionS / 9780123742551](http://books.elsevier.com/companionS/9780123742551)）为所有的示例程序提供了源码，从而使读者能够在上下文中查看重要的对象和方法，并理解每行代码的作用。

<<Java TCP/IP Socket编程>>

作者简介

Kenneth L.Calvert，肯塔基大学（University of Kentucky）的教授，他的教学和研究方向是计算机网络系统。

他已经有近20年的TCP / IP Socket编程经验。

Michael J.Donahoo，贝勒大学（Baylor University）的副教授，为大学生或大学研究生讲授计算机网络课程。

他已经编著了在各种编程语言下进行Socket编程的多本书籍和一本有关SQL的书籍。

书籍目录

译者序前言第1章 简介1.1 计算机网络、分组报文和协议1.2 关于地址1.3 关于名字1.4 客户端和服务器1.5 什么是套接字1.6 练习第2章 基本套接字2.1 套接字地址2.2 TCP套接字2.2.1 TCP客户端2.2.2 TCP服务器端2.2.3 输入输出流2.3 UDP套接字2.3.1 DatagramPacket类2.3.2 UDP客户端2.3.3 UDP服务器端2.3.4 使用UDP套接字发送和接收信息2.4 练习第3章 发送和接收数据3.1 信息编码3.1.1 基本整型3.1.2 字符串和文本3.1.3 位操作：布尔值编码3.2 组合输入输出流3.3 成帧与解析3.4 Java特定编码3.5 构建和解析协议消息3.5.1 基于文本的表示方法3.5.2 二进制表示方法3.5.3 发送和接收3.6 结束3.7 练习第4章 进阶4.1 多任务处理4.1.1 Java多线程4.1.2 服务器协议4.1.3 一客户一线程4.1.4 线程池4.1.5 系统管理调度：Executor接口4.2 阻塞和超时4.2.1 accept ()、read () 和receive () 4.2.2 连接和写数据4.2.3 限制每个客户端的时间4.3 多接收者4.3.1 广播4.3.2 多播4.4 控制默认行为4.4.1 Keep-Alive4.4.2 发送和接收缓存区的大小4.4.3 超时4.4.4 地址重用4.4.5 消除缓冲延迟4.4.6 紧急数据4.4.7 关闭后停留4.4.8 广播许可4.4.9 通信等级4.4.10 基于性能的协议选择4.5 关闭连接4.6 Applet4.7 结束4.8 练习第5章 NIO5.1 为什么需要NIO5.2 与Buffer一起使用Channel5.3 Selector5.4 Buffer详解5.4.1 Buffer索引5.4.2 创建Buffer5.4.3 存储和接收数据5.4.4 准备Buffer：clear ()、flip () 和rewind () 5.4.5 压缩Buffer中的数据5.4.6 Buffer透视：duplicate () 和slice () 等5.4.7 字符编码5.5 流 (TCP) 信道详解5.6 Selector详解5.6.1 在信道中注册5.6.2 选取和识别准备就绪的信道5.6.3 信道附件5.6.4 Selector小结5.7 数据报 (UDP) 信道5.8 练习第6章 深入剖析6.1 缓冲和TCP6.2 死锁风险6.3 性能相关6.4 TCP套接字的生存周期6.4.1 连接6.4.2 关闭TCP连接6.5 解调多路复用揭秘6.6 练习

章节摘录

通常情况下，在程序中使用套接字是因为需要向其他程序提供信息，或使用其他程序提供的信息。这并不是什么魔法：任何要交换信息的程序之间在信息的编码方式上必须达成共识（如将信息表示为位序列），以及哪个程序发送信息，什么时候和怎样接收信息都将影响程序的行为。程序间达成的这种包含了信息交换的形式和意义的共识称为协议，用来实现特定应用程序的协议叫做应用程序协议。前面章节中的反馈程序示例中的应用程序协议都过于简单：客户端和服务器的行为都不受它们之间所交换的信息内容的影响。而在绝大部分实际应用中，客户端和服务器的行为都要依赖于它们所交换的信息，因此应用程序协议通常更加复杂。

TCP / IP协议以字节的方式传输用户数据，并没有对其进行检查和修改。这个特点使应用程序可以非常灵活地对其传输的信息进行编码。大部分应用程序协议是根据由字段序列组成的离散信息定义的，其中每个字段中都包含了一段以位序列编码的特定的信息。应用程序协议中明确定义了信息的发送者应该怎样排列和解释这些位序列，同时还要定义接收者应该怎样解析，这样才使信息的接收者能够抽取出每个字段的意义。TCP / IP协议的唯一约束是，信息必须在块（chunk）中发送和接收，而块的长度必须是8位的倍数，因此，我们可以认为在TCP / IP协议中传输的信息是字节序列。鉴于此，我们可以进一步把传输的信息看作数字序列或数组，每个数字的取值范围是0 ~ 255。这与8位编码的二进制数值范围是一致的：00000000代表0，00000001代表1，00000010代表2等，最多到11111111，即255。

如果你建立了一个程序使用套接字与其他程序交换信息，通常符合下面两种情况之一：要么是你设计和编写了套接字的客户端和服务端，这种情况下你能够随心所欲地定义自己的应用程序协议；要么是你实现了一个已经存在的协议，或许是一个协议标准。

媒体关注与评论

我觉得本书做了很好的工作，包括简明扼要地抓住关键问题，提供了非常好的实例，因此可以作为重要主题的参考资料。

我认为本书适合希望快速地掌握Java网络编程的开发者，也可以作为学生的补充读物。

——Robere Brunner，美国国家超级计算应用中心（NCSA）

<<Java TCP/IP Socket编程>>

编辑推荐

《Java TCP/IP Socket编程（原书第2版）》适合作为Java Socket编程的入门教程，也可供从事网络相关专业的技术人员参考。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>