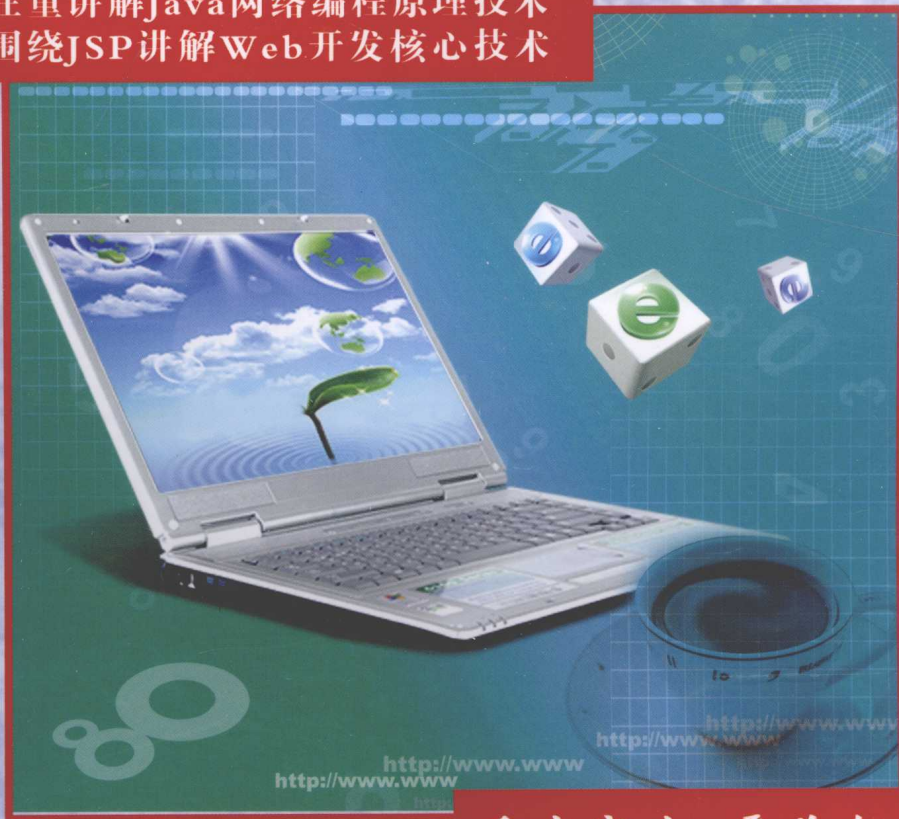


21世纪高等院校计算机专业规划教材

Java 网络编程原理与 JSP Web 开发核心技术

马晓敏 肖 明 姜远明 齐永波 编著

注重讲解Java网络编程原理技术
围绕JSP讲解Web开发核心技术



重点突出 覆盖全面



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

21 世纪高等院校计算机专业规划教材

Java 网络编程原理与 JSP Web 开发核心技术

马晓敏 肖 明 姜远明 齐永波 编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书是为 Java 学习者在了解 Java 编程的基础知识之后,进一步学习 Java 网络编程原理和 JSP Web 技术而编写的。本书清晰地介绍了计算机网络连接和通信的基本原理、Java 网络编程及各种应用协议的实现技术;并且详细介绍了 JSP Web 的各种核心技术及应用模式。

本书内容共分为两部分。第一部分讲解 Java 网络编程原理,包括网络程序设计基础、Java 的多线程机制、Socket 网络编程、Java 网络编程、应用协议的实现等内容。第二部分围绕 JSP 讲解 Web 开发核心技术,包括 JSP 基础知识、Java Servlet 技术、JavaBean 组件、JSP 数据库编程技术、JSP 相关技术与 MVC 模式,以及 Java EE 技术与 Java ME 移动编程。

本书侧重于 Java 网络编程原理和应用协议的实现以及 JSP Web 开发相关核心技术;同时力求重点突出,覆盖面广;各章均提供了丰富的实例和练习。全书内容由浅入深,实例生动,易学易用,可以满足不同层次读者的需求。

本书可作为普通高等院校本科计算机及信息相关专业的课程教材,也可作为软件开发人员的自学参考书。

图书在版编目(CIP)数据

Java 网络编程原理与 JSP Web 开发核心技术/马晓敏
等编著. —北京:中国铁道出版社,2010.1
21 世纪高等院校计算机专业规划教材
ISBN 978-7-113-10981-3

I. ①J… II. ①马… III. ①
JAVA 语言—程序设计—高等学校—教材②
JAVA 语言—主页制作—程序设计—高等学校—教材 IV.
①TP312②TP393.092

中国版本图书馆 CIP 数据核字 (2010) 第 011141 号

书 名: Java 网络编程原理与 JSP Web 开发核心技术
作 者: 马晓敏 肖 明 姜远明 齐永波 编著

策划编辑: 秦绪好 周海燕

责任编辑: 秦绪好

编辑助理: 张 丹

封面设计: 付 巍

编辑部电话: (010) 63560056

封面制作: 白 雪

责任印制: 李 佳

出版发行: 中国铁道出版社(北京市宣武区右安门西街8号 邮政编码: 100054)

印 刷: 北京京海印刷厂

版 次: 2010 年 3 月第 1 版 2010 年 3 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 21.25 字数: 523 千

印 数: 4 000 册

书 号: ISBN 978-7-113-10981-3

定 价: 32.00 元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

凡购买铁道版图书,如有印制质量问题,请与本社计算机图书批销部联系调换

Java 诞生后的几十年中, 已经发展成为 Internet 时代伟大的计算机语言之一, 它具有跨平台、直接面向对象、适合于单机和网络编程等诸多优点。对于 B/S 模式和 C/S 模式的应用开发, Java 语言提供了简单而卓有成效的解决方案。Java 技术也逐渐成为网络编程开发的主流技术之一。

本书内容分为两部分。第一部分是 Java 网络编程原理与技术。Java API 提供了 TCP 协议套接字、UDP 协议套接字和组播套接字等多种用于网络通信的类, 也包括数据包和数据流, 可以用来实现 SMTP 和 POP3、FTP、HTTP 等高层应用协议以及 C/S 模式。第二部分是 JSP Web 开发核心技术, 包括 JSP、Java Servlet、JavaBean、JSP 数据库技术、JSP 扩展技术、MVC 模式、Java EE 以及 Java ME 等技术。

目前 Java 技术框架非常庞大, MVC 模式(模型—视图—控制器)是 Xerox PARC 在 20 世纪 80 年代为编程语言 Smalltalk-80 开发的一种软件设计模式, 至今已被广泛使用。本书就基于 MVC 模式的开发框架 Struts 技术的应用也进行了简要介绍。

本书的特点是将 Java 网络编程与 JSP Web 开发技术有机地结合起来进行介绍, 使读者对于在 Internet 上从底层通信、数据传输、应用协议实现, 到高级应用层面的 JSP Web 开发的核心技术有一个全面的了解。

全书分为 11 章, 涉及内容如下:

第 1 章介绍了网络程序设计的基础知识, 包括网络基本概念和结构、OSI 七层网络体系结构和 TCP/IP 协议集。第 2 章介绍了 Java 多线程机制的原理及实现。第 3 章介绍了 C/S 设计模式以及 Socket 网络编程原理。第 4 章介绍了用 Java 技术实现的网络编程技术、过程和实例。第 5 章介绍了各种网络应用协议的基本原理以及 Java 编程实现。第 6 章介绍了 JSP 技术的基础知识和体系结构, 包括 JSP 的工作原理、基本语法及内置对象的使用。第 7 章介绍了 JavaServlet 的基本概念、生命周期、Servlet 常用类和接口的使用以及 Servlet 配置和调用。第 8 章介绍了 JavaBean 的基本概念, 介绍如何编写、编译和部署 JavaBean, 以及如何在 JSP 页面中对 JavaBean 进行调用和应用。第 9 章介绍了 JDBC 的工作原理, JDBC API 中相关接口和类的使用, 并介绍了 JSP 的数据库连接、高级操作和应用。第 10 章介绍了 JSP 技术的扩展应用, 着重介绍了 MVC 模式的概念、结构及原理, 基于 MVC 模式的框架技术 Struts 的原理及实现。第 11 章介绍了 Java EE 相关技术规范、应用层次与组件技术、Java EE 容器体系结构、EJB 的基本概念及工作原理, 以及 Java ME 技术的基本结构和开发手机游戏的基本过程。

本书内容新颖, 知识涵盖全面。同时力求重点突出, 层次清晰, 语言通俗易懂。本书可作为高等院校本科计算机及信息相关专业的课程教材, 也适合于编程开发人员培训、广大计算机技术爱好者自学使用。

本书第 1 章由齐永波编写, 第 2、6、10 章由马晓敏编写, 第 3~5 章由肖明编写, 第 7~9 章由姜远明编写, 第 11 章由齐永波和姜远明编写, 全书由肖明和马晓敏统稿、定稿。

在编写过程中，我们力求精益求精，投入了大量的时间和精力，但由于水平有限，难免存在一些不足之处，欢迎读者在使用本书时，不吝提出宝贵意见，请发 E-mail 至 netprogram2009@yeah.net。

编 者

2010 年 1 月

目 录

CONTENTS >>>

第 1 章 网络程序设计基础.....	1
1.1 网络程序设计简介.....	1
1.2 计算机网络体系结构.....	2
1.2.1 OSI 分层结构.....	2
1.2.2 Internet 体系结构.....	4
1.2.3 传输控制协议 TCP.....	5
1.2.4 网际协议 IP.....	7
1.3 Java 支持的网络程序技术.....	9
1.3.1 使用数据报传输.....	10
1.3.2 Java 的安全性.....	10
1.4 网络程序设计架构.....	12
1.4.1 C/S 体系结构.....	12
1.4.2 B/S 体系结构.....	13
1.4.3 发展与前景.....	14
小结.....	14
习题一.....	14
第 2 章 Java 的多线程机制.....	15
2.1 Java 中的线程与多线程概述.....	15
2.2 Java 中线程类、接口和方法及多线程实现.....	16
2.2.1 用 Thread 类创建多线程应用程序.....	16
2.2.2 用 Runnable 接口创建多线程应用程序.....	17
2.3 线程优先级.....	19
2.3.1 分配线程优先级.....	19
2.3.2 设置和获取当前线程优先级.....	20
2.4 线程的生命周期.....	21
2.4.1 线程的生命周期.....	21
2.4.2 线程的等待/通知机制.....	22
2.5 线程池.....	25
2.5.1 为什么使用线程池.....	25
2.5.2 线程池的开销.....	26
2.5.3 线程池的实现.....	26
小结.....	29
习题二.....	29

第 3 章 Socket 网络编程	30
3.1 套接字 Socket 和 Socket 通信基础	30
3.1.1 套接字 Socket 的定义	30
3.1.2 面向客户端/服务器结构的套接字 Socket	32
3.1.3 套接字 Socket 最基本的函数与函数调用流程	33
3.2 Windows Sockets 网络编程	35
3.2.1 Windows Sockets 编程概述	35
3.2.2 客户端和服务端主要函数描述	35
3.3 Windows Sockets 网络编程实例	42
3.3.1 构建编程环境	42
3.3.2 TCP 服务器和客户端编程实例	42
3.3.3 UDP 服务器和客户端编程实例	45
小结	47
习题三	47
第 4 章 Java 网络编程	48
4.1 数据流	48
4.1.1 数据流与网络技术的联系	48
4.1.2 流的工作方式	48
4.1.3 过滤器流、读取器和写入器	49
4.1.4 对象持久性和对象序列化	57
4.2 基于 TCP 协议的套接字通信	60
4.2.1 TCP 的套接字	61
4.2.2 Socket 类	63
4.2.3 ServerSocket 类	64
4.2.4 异常处理	66
4.2.5 TCP 协议网络通信实例	67
4.3 基于 UDP 协议的数据报和套接字	69
4.3.1 UDP 概述	70
4.3.2 DatagramPacket 类	70
4.3.3 DatagramSocket 类	71
4.3.4 构建 UDP 客户端/服务器	71
4.3.5 UDP 网络应用实例	73
4.4 组播套接字	77
4.4.1 组播传播方式	77
4.4.2 组播 Socket 类	79
4.4.3 组播 Socket 的实例	80
小结	82
习题四	83

第 5 章 应用协议的实现	84
5.1 概述	84
5.2 应用协议规范	84
5.2.1 基于 TCP 传输协议的应用层协议	84
5.2.2 基于 UDP 传输协议的应用层协议	85
5.2.3 RFC	86
5.3 SMTP 通信协议	86
5.3.1 SMTP 介绍	86
5.3.2 SMTP 客户端——Java 应用程序	88
5.3.3 SMTP 客户端——Java Applet	92
5.4 POP3 通信协议	95
5.4.1 POP3 介绍	95
5.4.2 POP3 客户端	96
5.5 FTP 协议	100
5.5.1 FTP 协议体系结构	100
5.5.2 FTP 工作模式	103
5.5.3 服务器端 FTP 程序	104
5.5.4 客户端 FTP 程序	110
5.6 HTTP 协议	113
5.6.1 HTTP 协议概述	113
5.6.2 HTTP 协议内容	114
5.6.3 HTTP 服务器实现	117
5.7 统一资源定位器 URL 类与 HTTP 协议	121
5.7.1 Java URL 主类	121
5.7.2 URLConnection 抽象类	123
5.7.3 用 URL 和 URLConnection 获取资源	127
小结	129
习题五	130
第 6 章 JSP 基础知识	131
6.1 JSP 概述	131
6.1.1 JSP 的工作原理	131
6.1.2 JSP 的特点	133
6.1.3 JSP 与传统 CGI 的比较	134
6.1.4 JSP 与 ASP 的比较	134
6.1.5 JSP 与 PHP 的比较	135
6.2 建立 JSP 运行环境	136
6.2.1 JSP 运行环境要求	136
6.2.2 JDK 的安装与配置	137

6.2.3	JSP 服务器配置	137
6.2.4	一个典型的 JSP 文件	139
6.3	JSP 基本语法	140
6.3.1	通用的语法规则	140
6.3.2	JSP 的脚本元素	141
6.3.3	JSP 指令	143
6.3.4	JSP 动作元素	145
6.4	JSP 的内置对象	150
6.4.1	内置对象介绍	150
6.4.2	application 对象	152
6.4.3	config 对象	153
6.4.4	exception 对象	155
6.4.5	out 对象	156
6.4.6	page 对象	158
6.4.7	pageContext 对象	158
6.4.8	request 对象	158
6.4.9	reponse 对象	161
6.4.10	session 对象	163
6.5	JSP 与 HTML 表单	165
6.5.1	表单概述	165
6.5.2	建立表单	165
6.5.3	按钮标识	166
6.5.4	复选框标识	168
6.5.5	单选按钮标识	169
6.5.6	列表框标识	170
6.5.7	文本区域标识	172
6.5.8	隐藏域标识	173
6.6	JSP 的异常处理	174
	小结	175
	习题六	175
第 7 章	JSP 核心技术之 Java Servlet	176
7.1	Servlet 概述	176
7.1.1	Servlet 容器与 Tomcat 服务器	176
7.1.2	Servlet 的特点	177
7.1.3	Servlet 的生命周期	178
7.1.4	Servlet API 简介	179
7.2	编写简单的 Servlet	182
7.2.1	Servlet 基本结构	182

7.2.2 输出纯文本的 Servlet	182
7.2.3 编译运行 Servlet	183
7.2.4 输出 HTML 的 Servlet	183
7.2.5 Servlet 的配置	184
7.2.6 Servlet 的调用	186
7.3 Servlet 应用	187
7.3.1 获取表单数据	187
7.3.2 读取 HTTP 请求头	188
7.3.3 处理 Cookie	189
7.3.4 Servlet 中的 HTTP 会话管理	190
7.3.5 创建 Web 监听	191
7.3.6 创建过滤程序	195
小结	199
习题七	199
第 8 章 JSP 核心技术之 JavaBean 组件	200
8.1 JavaBean 概述	200
8.1.1 什么是 JavaBean	200
8.1.2 JavaBean 的属性	200
8.1.3 编写简单的 JavaBean	202
8.1.4 编译和部署 JavaBean	203
8.2 在 JSP 中应用 JavaBean	203
8.2.1 在 JSP 中调用 JavaBean	203
8.2.2 通过动作指令获取与设置 JavaBean 属性值	205
8.2.3 JSP+Servlet + JavaBean 结合使用	208
8.3 JavaBean 应用实例	211
8.3.1 计数器 JavaBean	211
8.3.2 在线投票 JavaBean	211
小结	215
习题八	215
第 9 章 JSP 数据库编程技术	216
9.1 JDBC 概述	216
9.1.1 JDBC 工作原理	216
9.1.2 JDBC API 简介	217
9.2 MySQL 数据库的安装与配置	221
9.3 连接数据库	225
9.3.1 连接数据库的一般过程	225
9.3.2 使用 JDBC-ODBC 驱动连接 Access 数据库	226
9.3.3 使用 JDBC 驱动连接 MySQL 数据库	228

9.3.4	通过 JSP 页面连接数据库	229
9.3.5	通过 JavaBean 连接数据库	230
9.3.6	通过 Servlet 连接数据库	233
9.4	数据库高级操作	235
9.4.1	使用存储过程	235
9.4.2	事务操作	237
9.4.3	操作二进制数据	237
9.5	数据库连接池技术	238
9.5.1	连接池与数据源概述	238
9.5.2	使用连接池连接 MySQL 数据库	241
9.6	应用实例	242
小结		246
习题九		246
第 10 章	JSP 相关技术与 MVC 模式	247
10.1	JSP 的文件操作	247
10.1.1	JSP 的目录操作	247
10.1.2	JSP 的文本文件操作	249
10.1.3	JSP 的流文件操作	251
10.1.4	文件的上传和下载	253
10.2	JSP 的邮件发送技术	261
10.2.1	JavaMail API 简介	261
10.2.2	JavaMail API 结构	261
10.2.3	在 JSP 页面中使用 JavaMail API 发送 E-mail	264
10.2.4	在 JSP 页面中发送包含附件的 E-mail	267
10.3	Ajax 技术	270
10.3.1	Ajax 概述	270
10.3.2	Ajax 的关键技术	271
10.3.3	Ajax 框架	277
10.3.4	在 JSP 中应用 Ajax 技术	278
10.4	MVC 设计模式与应用	283
10.4.1	MVC 设计模式	283
10.4.2	Struts 框架结构	285
10.4.3	Struts 配置文件	288
10.4.4	Struts 标签库	291
10.4.5	Struts 框架实现	291
小结		299
习题十		300

第 11 章 Java EE 技术与 Java ME 移动编程	301
11.1 Java EE 技术	301
11.1.1 Java EE 简介	301
11.1.2 Java EE 应用层次与组件	302
11.1.3 Java EE 容器体系结构	303
11.1.4 EJB 简介	304
11.1.5 Java EE 应用程序的打包与部署	306
11.2 Java ME 移动编程	307
11.2.1 Java ME 移动编程概述	308
11.2.2 Java ME 技术框架	308
11.2.3 Java ME 应用项目开发过程	309
小结	314
习题十一	314
附录 A Struts 核心类介绍	315
附录 B Struts 常用标签库	319
附录 C web.xml 文件	322
附录 D 本书所用资源	327
参考文献	328

第 1 章 | 网络程序设计基础

把处于不同地理位置、具有独立功能的计算机通过通信设备及传输媒体互连起来，在通信软件和网络协议的支持下，实现计算机之间资源共享、信息交换或协同工作的系统，称为计算机网络，简称网络。

在网络中，不同设备之间的通信是通过一组预先定义好的规则进行的，这组规则就是通常所说的网络协议。网络协议描述了计算机发送信息所应具有的模式，以及对收到的信息应作出何种响应。网络协议的存在，使得开发人员能够使用不同的程序设计语言，以不同的数据结构和命令代码，独立地开发和实现分布式应用程序的不同部分。在 Internet 中，最主要的网络协议是 TCP (Transmission Control Protocol) 协议和 IP (Internet Protocol) 协议。有了协议才能有网络的正常通信，最终才能进行网络编程实现网络应用。

1.1 网络程序设计简介

上网时打开浏览器，输入一个 URL 地址，这个地址指向的网页就会从远程 Web 服务器传到客户端，然后这个网页按协议要求在浏览器中显示出来。由此可知，网络编程的基本任务就是开发像浏览器一样的客户端程序，以及像 Web 服务器这样的服务器程序，并且使两者按一定的协议连接、通信，才能有条不紊地交换数据，实现一定的功能。

例如，甲给乙邮寄一封信，甲不必亲自把信送到乙家里，送信的任务由邮政网络来完成。甲只须提供乙的地址，邮政网络就会准确地把这封信送达目的地。在这个过程中，各方都必须遵循一定的规则，即协议，否则信件就不一定到达目的地。同样，服务器程序与客户端程序只关心发送什么样的数据给对方，而不必考虑如何把这些数据传输给对方。传输数据的任务由计算机网络来完成的，如图 1-1 所示。这个过程中同样要遵守一定的协议，工作才能做好。

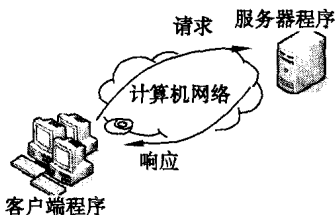


图 1-1 客户端和服务器的通信过程

以上说明三个问题：网络应用程序设计需要建立在计算机网络基础上；计算机网络要遵循一定的网络协议；客户端与服务要建立连接，实现通信。

目前网络程序设计比较成熟的语言和技术有 Java 技术和 .net 技术，分别属于 Sun 公司和微软公司。网络程序设计一般采用客户端与服务器 (C/S) 或浏览器与服务器 (B/S) 结构，这些结构将在 1.4 节“网络程序设计架构”中介绍；客户端与服务器结构的网络编程将分别在第 2~5 章介绍；浏览器与服务器 (B/S) 结构的网络程序设计将在第 6 章介绍，其核心技术将在第 7~10 章介绍，也称之为 JSP Web 开发核心技术。

计算机网络协议是采用分层结构，将在 1.2 节“计算机网络体系结构”中介绍。

客户端与服务器建立连接，并实现通信是采用抽象网络套接字（Socket）对象的方式实现连接和通信。将在第 3 章的“3.1 套接字 Socket 和 Socket 通信基础”节中介绍。套接字（Socket）封装了复杂的 TCP/IP 协议簇，形成了 Socket 抽象层，由它去应对复杂的网络间通信，并符合指定的协议，实现对应的连接和通信，这样对用户来说非常方便。

1.2 计算机网络体系结构

近代计算机网络采用分层的体系结构，把一个复杂的问题分解成若干个简单又易于处理的问题，使之易于实现。在这种分层结构中，每层都是建筑在它的前一层基础上，每层之间有相应的通信协议，相邻层之间的通信约束称为接口。在分层处理后，相似的功能出现在同一层内，每一层仅与其相邻上、下层通过接口进行通信，使用下层提供的服务，并为上层提供服务。上、下层之间的关系是下层对上层的服

务，上层是下层的用户。计算机网络的各层和在各层上使用的全部协议的集合称为网络的体系结构。网络体系结构是比较抽象的概念，可以使用不同的硬件和软件来实现这样的结构。世界上第一个网络体系结构是 IBM 公司于 1974 年提出的 SNA。在此之后，许多公司纷纷提出了各自的网络体系结构，如 DEC 公司提出的 DNA、Honeywell 公司提出的 DSA 以及 Internet 上使用的 TCP/IP 等。这些著名的网络体系结构的共同之处在于它们都采用了分层技术，但层次划分、功能分配与采用技术术语各不相同，结果导致了不同网络之间难以互连。因此，1977 年国际标准化组织（ISO）成立了计算机网络标准分委员会，负责研究网络的标准化问题。该委员会于 1983 年提出了一个开放式系统互连参考模型（Open System Interconnection, OSI）。直到 20 世纪 90 年代初期，整套 OSI 国际标准才被制定出来，该模型已被世界各国所承认。

1.2.1 OSI 分层结构

1. OSI 参考模型

OSI 参考模型的结构如图 1-2 所示。OSI 共划分为七层，每一层都具有一定的独立性，它们各自完成各自的任



图 1-2 OSI 参考模型

在七层结构中，每一层都为上一层提供服务。因此，越向上层其功能越具有综合性和宏观控制特性，下层承担具体的基础工作。通信时，数据从上层逐层传递到下层，在物理层连接到通道

的物理介质上,完成通信;接收方则将物理层的数据依次向上传递,方向与发送方相反。在这七层中,1~4层为低层,是面向通信的;5~7为高层,是面向信息处理的。

划分为层次后,由于各层相对独立,可以使层间的接口更加标准化,且易于变更和扩充。各层次需要解决层间的服务规范和层内的通信协议规范。各层的基本功能说明如下:

(1) 物理层 (Physical Layer)

物理层是 OSI 参考模型的最低层,利用物理传输介质为其上的数据链路层提供比特流传输。该层的主要任务是在通信线路上传输数据比特的电信号。物理层协议主要规定了计算机、终端和通信设备之间的接口标准,它包含接口的机械、电气、功能和规程方面的特性。物理层传送的基本单位是比特 (bit)。

(2) 数据链路层 (Datalink Layer)

数据链路层的功能是向网络层提供无差错的传输服务。

数据链路层的主要功能有建立和拆除数据链路;将信息按一定格式组装成帧,以便无差错地传送。此外还具有处理应答、差错控制、信息流控制和顺序控制等功能。

数据链路层传送的基本单位是帧,帧的结构如图 1-3 所示。soh 为帧的开始符, eot 为帧的结束符。使用帧可以解决长时间延迟和计算机崩溃情况下出现的数据传输不完整性问题,如 eot 没有到达,表示此帧不完整。但由于使用帧多传了开始符和结束符,也增加了网络开销。

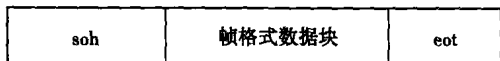


图 1-3 帧结构示意图

(3) 网络层 (Network Layer)

网络层的基本任务是将数据移到某个特定的网络位置,其关键是路由选择。

网络层将从高层传送下来的数据打包,再进行必要的路由选择、差错控制、流量控制及顺序检测等处理,使发送站传输层所传下来的数据能够准确无误地按照地址传送到目的站,并交付给传输层。

当异构网络互连时,网络层需要解决寻址方式、分组长度及不同协议间可能出现的问题。数据链路层是将数据发给连到网络上的所有设备,由接收设备决定是否接收这个数据。而与此不同,网络层则是选择一个特定的路由,避免将数据发给无关的网络。网络通过交换、网络层寻址及路由选择算法来达到这一点。网络层所传送信息的基本单位称为包。网络层采用的协议是 X.25 分组级协议。

(4) 传输层 (Transport Layer)

传输层的主要作用是保证通信的质量,如消除通信过程中产生的错误等。它负责使点 A 到点 B (可能属于也可能不属于同一网络段) 的数据传输是可靠、无错的。在发送报文时,传输层将报文分解成若干数据块,向下层传输;而接收时负责将数据块组合成报文,提交给上层。传输层信息传送的基本单位称为报文。传输层所采用的协议是近年来才标准化的 ISO8072/3。

(5) 会话层 (Session Layer)

用户或进程间的一次连接称为一次会话。会话层负责提供建立、维护和拆除两个进程间的会话连接,对会话进行有效管理。当连接建立后,会话层负责控制数据交换、限制或同步两进程之间的数据操作。会话层传送信息的基本单位也是报文,但它与传输层的报文有本质的不同。会话层的协议是 ISO832/7。

(6) 表示层 (Presentation Layer)

表示层主要功能是处理两个通信系统中信息的表示方法。完成数据格式的转换,对数据进行加密和解密、压缩和恢复等。表示层传送信息的基本单位也是报文。该层使用的协议是 ISO8822/3/4/5。

(7) 应用层 (Application Layer)

应用层是 OSI 模型的最高层,它负责应用程序与网络操作系统之间的联系,为用户提供各种服务,包括 WWW 浏览服务、文件传送、远程登录、电子邮件及网络管理、News 新闻组讨论、DNS 域名服务、NFS 网络文件系统等。在应用层包含了许多协议,如 WWW 浏览服务的 HTTP 协议、文件传送的 FTP 协议、远程登录的 Telnet 协议、电子邮件及网络管理的 SMTP 协议等。

应用层传送的是用户数据报文。该层典型的协议有 ISO8571/1/2/3/4 用于文件传送、存取和管理;ISO9040/1 用于虚终端;ISO8831/2,作业传送与操纵协议;ISO8649/50,公共应用服务元素等。

2. OSI 参考模型的主要特征

OSI 参考模型的主要特征如下:

- ① OSI 是一种将异构系统互连的分层结构,它提供了控制互连系统交互规则的标准框架,定义了抽象结构而并非具体实现的描述。
- ② 对等层之间的通信必须遵循相应层的协议,如网络层协议、传输层协议等。
- ③ 相邻层之间的接口定义了原语操作和下层向上层提供的服务。
- ④ 所提供的公共服务是面向连接的或无连接的数据通信服务。

OSI 只是一个参考模型,其作用只是提供概念性和功能性结构,同时确定研究和改进标准的范围,并为维持所有有关标准的一致性提供共同的参考。因此,OSI 参考模型及其有关标准只是技术规范,而不是工程规范。实际实现操作中,一般只取 OSI 中的一部分并有所变化,很少有一个与 OSI 完全一致的体系得以实现。这正说明了这个标准的开放性和它的优越性。

1.2.2 Internet 体系结构

1974 年 Vinton Cerf 和 Robert Kahn 开发了 TCP/IP 协议,Internet 采用是 TCP/IP 协议。随着 Internet 的飞速发展,TCP/IP 协议也因此成为事实上的工业标准。TCP/IP 协议实际上是一组协议,因其两个著名的协议 TCP (Transmission Control Protocol, 传输控制协议) 和 IP (Internet Protocol, 网际协议) 而得名。TCP/IP 协议体系和 OSI 参考模型一样,也是一种分层结构。它由基于硬件层次上的四个概念性层次构成,即网络接口层、网际层、传输层和应用层。如图 1-4 所示表示了 TCP/IP 协议体系与 OSI 参考模型的对应关系。

OSI模型		TCP/IP模型	信息格式
应用层	各种应用层协议,如Telnet/ SMTP/FTP/DNS/HTTP等	应用层 (应用程序)	报文流
表示层			
会话层			
传输层	传输层TCP/UDP协议	传输层	分组
网络层	IP/ARP/ICMP/IGMP协议	网际层	IP数据报
数据链路层	SLIP/PPP协议	网络接口层	帧层
物理层			

图 1-4 TCP/IP 协议与 OSI 参考模型的对比

TCP/IP 协议各层的主要功能如下:

① 网络接口层:称为数据链路层,是 TCP/IP 协议的最底层,但是 TCP/IP 协议并没有严格定义该层。它只是要求主机必须使用某种协议与网络连接,以便能在其上传递 IP 分组。

② 网际层:也称为 IP 层,处理计算机之间的通信。它接收来自传输层的请求,传输某个具有目的地址信息的分组。该层把分组封装到 IP 数据报中,填入数据报的头部(报头),使用路由算法选择是直接把数据报发送到目标机还是把数据报发送给路由器,然后把数据报交给下面网络接口层中的对应网络接口模块。该层还处理接收到的数据报,检验其正确性,使用路由算法决定对数据报是在本地进行处理还是进行继续向前传送的处理。

③ 传输层:传输层的基本任务是提供应用层之间的通信,即端到端的通信。传输层管理信息流,提供可靠的传输服务,以确保数据无差错地按序到达。为了这个目的,传输层协议软件要进行协商,让接收方回送确认信息或让发送方重发丢失的分组。传输层协议软件将要传送的数据流划分成分组,并把每个分组连同目的地址交给下一层去发送。

④ 应用层:这是最高层,用户调用应用程序来访问 TCP/IP 协议互连网络提供的多种服务。应用程序负责发送和接收数据。每个应用程序选择所需的传输服务类型,可以是独立的报文序列,也可以是连续的字节流。应用程序将数据按要求的格式传送给传输层。

1.2.3 传输控制协议 TCP

(1) TCP 协议概述

TCP 协议是 TCP/IP 协议体系中面向连接的传输层协议,它提供全双工的和可靠的服务。如图 1-5 所示是 TCP 协议发送报文段的过程示意图。图中是一个方向的数据流,实际上,只要建立了 TCP 协议连接,就能支持同时双向通信的数据流。

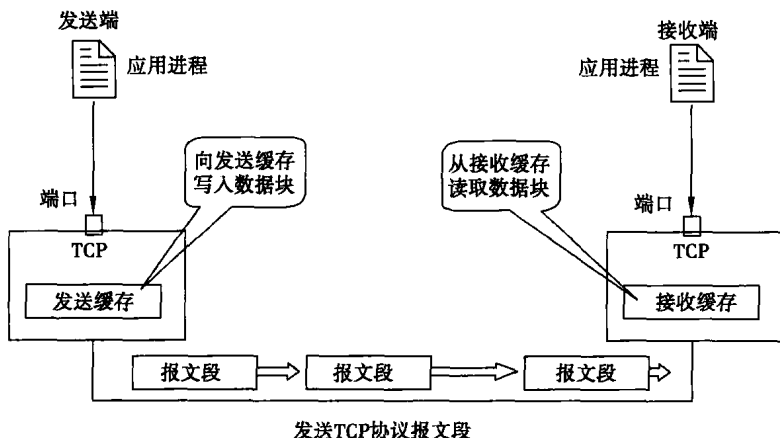


图 1-5 TCP 协议发送报文段的示意图

发送端的应用进程不断地将数据块陆续写入到 TCP 协议的发送缓存中,将其组成 TCP 协议报文段 (Segment), 逐个传送给网际层, 然后发送出去。接收端从网际层收到 TCP 协议报文段后, 先将其暂存在接收缓存中, 接收端的应用进程从接收端的缓存中逐个读取数据块。

(2) TCP 协议报文段的首部

一个 TCP 协议报文段分为首部和数据两部分,如图 1-6 所示。报文段首部的前 20B 是固定的,后面可以根据需要而增加,因此 TCP 协议的首部的最小长度是 20B。