



李媛媛

飞思科技产品研发中心

编著
监制

C/C++

开发专家

专业人士
权威经典

Visual C++

网络通信开发入门 与编程实践



5 STAR

超长时间多媒体有声讲解视频

全书以通信协议层的技术为主线

按网络体系结构从应用层到数据链路层逐一讲解

实例完美融合知识点、技巧、行业知识与成功经验



光盘包含视频讲解与
书中实例文件



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

http://www.phei.com.cn

李媛媛
飞思科技产品研发中心

编著
监制

C/C++
开发专家

Visual C++

网络通信开发入门 与编程实践



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内容简介

本书从介绍通信基础知识和 VC++ 基本编程模型开始, 通过众多的小实例来贯穿讲解晦涩的基础知识; 然后按照通信协议层展开, 将通信协议层和实际应用结合, 让读者在学习“基础”后学习中、高级应用, 最终提高实际应用水平和独立编程技能; 本书最后部分的综合案例, 可以让读者对前面学习的内容融会贯通, 以便深刻理解与实践应用。

本书的特点是: 以通信协议层的技术为主线, 在此基础上讲解各技术的应用范围, 再通过众多小、中、大型实例来全面而生动地讲解 VC++ 的知识, 既能让读者全面地学习 VC++ 技术, 又能让读者联系实际, 从而摆脱单纯讲解软件功能的枯燥学习模式。本书在讲解 VC++ 的过程中还穿插大量提示和技巧, 并对复杂和容易忽略的问题进行单独说明。书中的实例制作深入浅出, 步骤详细清晰, 可以帮助读者轻松、快速地学习 VC++, 能够保证对 VC++ 不了解的读者也可以轻松学习。同时对实例进行细致的选择, 使本书将知识点、技巧、行业知识和成功经验完美地融合到实例中, 也让中级读者感觉物有所值。

本书所配光盘中包含多媒体视频教学和实例源文件。

本书适合 VC++ 初、中级自学用户及 VC++ 设计爱好者, 同时也可作为计算机技能中级培训教材。

未经许可, 不得以任何方式复制或抄袭本书的部分或全部内容。

版权所有, 侵权必究。

图书在版编目(CIP)数据

Visual C++ 网络通信开发入门与编程实践 / 李媛媛编著. —北京: 电子工业出版社, 2008.10

(C/C++ 开发专家)

ISBN 978-7-121-07391-5

I.V… II.李… III.C 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字 (2008) 第 141673 号

责任编辑: 王树伟 田 蕾

印 刷: 北京智力达印刷有限公司

装 订: 北京中新伟业印刷有限公司

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: 32.25 字数: 825.6 千字

印 次: 2008 年 10 月第 1 次印刷

印 数: 5 000 册 定价: 55.00 元 (含光盘 1 张)

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。

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。

服务热线: (010) 88258888。

出版说明

“聪明的人使用 Delphi，真正的程序员使用 C++。”

时至今日，这句曾经在业内广为流行的话语又增添了更丰富的内涵。

脚本语言、Java、.NET 等正在争夺更大的天地。

然而，C/C++ 仍不失为最好、最纯粹的编程语言。

——“C/C++ 开发专家”引导你成为真正的程序员

C/C++ 的发展

作为一种结构化的中、高级编程语言，C 语言具有功能齐全、适用范围广的优势，一直为很多程序员所钟爱，并被视为最佳的编程入门语言，拥有着庞大的使用和学习人群。C++ 是在 C 语言基础上开发的一种集面向对象编程、通用编程和传统的过程化编程于一体的编程语言，是目前业界广泛使用的一种编程语言。然而，软件产业的规模和环境发展到今天，已经发生了深刻的变化。如今企业级应用整合与开发的任务主要由 Java、基于 .NET 平台的 C# 及各种新型动态语言来承担。C++ 的应用场合有所收缩，不再像之前那样从上到下包打天下，呈现出鲜明的行业应用特色。未来 C++ 主要在系统级复杂应用程序，高性能、实时中间件和嵌入式领域发挥所长。随着多核 CPU 的普及和网络安全重要性的空前提升，在并发程序设计、系统安全及视频处理、嵌入式开发方面，C++ 将获得新的应用空间。在大规模、高性能计算，游戏开发、嵌入式实时应用开发方面，以及一些传统的客户端软件和构件开发中，C++ 也将继续保持其稳定的地位。

C/C++ 的图书现状

C++ 的教学和使用具有其复杂性，而传统图书和学习方法的各种弊端更加剧了这一现象，使 C++ 成为不少人望而生畏的难学、难用的“专家语言”。虽然国内的 C/C++ 图书并不缺乏，但大多只适合有一定经验的程序员提升功力之用，而内容全面准确、讲解循序渐进、学习简明易懂的原创图书并不多见。近期 C/C++ 图书市场存在如下特点：

1. 国外经典图书全面翻新。近年来国外一些书商根据 C++ 所发生的变化，不断地进行版本升级或全面改写书稿，推出新的力作。
2. 国内原创图书缺乏力作。近年来国内虽然有一批令人耳目一新的 C++ 好书面世，但在技术层面上对实践的关注略显不足，难解读读者之渴。
3. 关键性图书存在空白。基于组件的软件开发、复杂网络应用，以及热度尚在的 COM 开发等方面的图书有待开发。

基于上述现状，我们组织 C/C++ 各应用领域的作者，推出本丛书“C/C++ 开发专家”，力求从新的、实用的、全面的角度介绍 C/C++，使其紧密地跟踪当前国内最实用、最热门的编程技术。我们期望通过这套丛书，能够提高各位读者的 C/C++ 开发水平及编程的实践能力，为我国计算机产业奉献一份微薄之力。

“C/C++ 开发专家”助你成为真正的程序员

“C/C++ 开发专家”的读者定位是：C/C++ 初学者，需要提升应用开发能力的程序员，具有实际开发经验的中、高级程序员。对阅读本丛书的读者建议如下：

➤ 面向 C/C++ 初学者

本丛书通俗易懂，并自成体系。丛书全面介绍 C/C++ 及 Visual C++ 的编程技术和实践操作。通过学习，初学者可快速地掌握涉及 OOP、STL、泛型编程等标准 C/C++ 的内容，对 C/C++ 技术应用有更深刻的理解。

➤ 面向需要提升应用开发能力的程序员

对于那些急需提升应用开发能力的程序员来说，本丛书是再好不过的专家向导。丛书除全面介绍标准

C/C++的内容外,还涉及数字图像处理、流媒体、网络通信和嵌入式开发等多个领域,可以为从事相关领域开发的程序员提供有益的帮助和参考。

➤ 面向具有实际开发经验的中、高级程序员

本丛书同样适合于具有实际开发经验的中、高级程序员。书中列举的大部分实例具体翔实,非常值得广大高级程序员学习和借鉴。

“C/C++开发专家”为程序员量身打造

本套丛书通过不同种类的图书来满足读者的需求。

➤ 语言入门

C/C++是一门优秀的高级语言。它绝不像一些传统图书所述是一门晦涩难懂、高深莫测的“专家语言”。本丛书的语言入门分支面向初学者,以通俗易懂的语言,介绍标准的C/C++语言知识,以及Visual C++编程技术;在保证知识体系的完整性的同时,在语言、体例上更贴近程序员的学习心理需求。

➤ 应用实践

如果脱离了具体的应用背景,任何一门计算机语言的学习都是“纸上谈兵”。如果程序员没有真正掌握面向应用的实践开发技能,那么很有可能面临来自就业的压力。本丛书的应用实践分支面向数字图像处理、流媒体、网络通信、嵌入式开发等不同的行业应用方向,介绍C/C++应用技术。目标是努力将读者培养成具有实际开发能力的从业人员。

➤ 开发详解

只让人阅读一遍的书很难说是一本好书。任何一本书在读者的眼中总会经历“厚→薄→厚”的过程。同样,C/C++语言会耐人寻味,但真正理解C/C++一般性内容需要花时间,而要做到融会贯通则更要下工夫。本丛书的开发详解分支针对C/C++语言及Visual C++中的高级特性,进行深入的剖析和讲解。C/C++程序员一旦掌握更高级的编程技巧,且对C/C++的语言内涵及开发技术有更为深入的理解,就能得心应手地运用这门语言。

➤ 技巧集锦

从大规模的并行计算到嵌入式系统开发,C/C++的应用领域非常广泛。即便是世界上最厚的一本书,也无法介绍所有的C/C++技术。针对这一特点,本丛书的技巧集锦分支对程序员经常遇到的问题进行解答和分析,并注重举一反三,启发读者思考。通过对这一话题的讨论,给正在从事或即将从事C/C++开发的程序员以最大的启迪。

“C/C++开发专家”丛书特色

本丛书具有如下特色。

➤ 由浅入深,通俗易懂

实际上,根本就不存在只面向纯粹的初学者的C/C++书籍。原因很简单:C/C++就不是初级的语言。初学者选择C/C++的时候,除了有足够的兴趣之外,还要有足够的耐心和恒心。为此,本丛书在保持完整性的同时注重语言的通俗性和知识的趣味性,避免了较为复杂的理论概念,取而代之的是常见的编程技巧和实际例子,力求由浅入深,通俗易懂,充分调动读者的阅读兴趣。

➤ 案例为主,内容生动

如果没有“案例”,C/C++的学习可能非常枯燥无趣;如果没有合适、有趣的“案例”,C/C++的学习仍会枯燥无趣。与以往的风格不同,本丛书强调编程实践,提供了大量的实例及源代码。这些案例均由作者从实际开发工作中设计的原型案例精简加工而成,形式丰富多样,具有很好的实用价值。

➤ 倡导正确的编程思想

“授之以鱼,不如授之以渔。”本丛书并非按部就班地完成知识传授,而是在介绍知识的同时倡导正确的学习思想和方法。如:倡导OOP思想、泛型编程、流行的设计模式、不断的重构理念和开源精神等。读者在阅读本书的同时,会接触到这些新的理念和方法。在某些开放性话题上,本丛书一反以往一些图书的“专家”面孔,更加贴近读者,从各个角度与读者展开交流和探讨。

飞思科技产品研发中心

0.1 编写目的

欢迎阅读《Visual C++网络通信开发入门与编程实践》一书。Microsoft 推出的 Visual Studio C++和网络通信的完美结合成就了本书。本书全面介绍 VC++网络编程的知识,语言本身不难,易于学习,并且是一门典型的面向对象的编程技术,用 VC++开始您的网络通信编程是绝佳方式。

0.2 本书特色

网络通信作为一门独立的学科,其内容较为繁杂,有许多彼此交织的概念和协议。基础篇中涉及到 VC++工具的使用及网络应用在 VC++中的实现,协议原理全部放在提高篇中讲解。提高篇中按照网络的体系结构从应用层到数据链路层逐一讲解,这种独特的讲解方式使读者能够清晰地认识网络通信中的各个层次,从网络的最顶层应用层开始认识,在应用层有很多读者每天都会接触到的网络通信应用,譬如电子邮件、远程登录等。这样可以激发读者的兴趣,从而开展后续更深入的学习。读者不仅可以看到网络应用程序和工作原理的结合,还能通过典型的实例创建自己的网络通信应用程序,从而不再使读者迷惘于网络通信原理和开发工具之间。进阶篇用几个大型的实例来进行演练,进行 VC++网络通信的全面编程实践。

0.3 内容导读

本书共 13 章,在结构上分为三大部分:基础篇,提高篇,进阶篇。

前 4 章为基础篇,主要讲解 Visual C++网络通信基础、Windows 编程模型及网络的基本应用。涵盖计算机网络体系结构中各种基本概念、Windows Sockets 和 MFC;详细讲解对初学者非常重要的一些编程规则和编程方法;并且在 VC++中实践网络基本应用,如获取 MAC 地址等。

第 1 章讲解 Visual C++网络通信基础,涵盖计算机网络的组成、计算机网络的分层体系结构,认识 Winsock 编程,学习 Winsock 的相关概念和技术特点及理解 Winsock 的编程原理,最后利用 MFC 进行网络编程,利用 CSocket 类制作网络聊天室。

第 2 章主要介绍 Windows 编程模型,从最简单的如何命名开始介绍匈牙利命名法;对于如何使用 Visual C++集成开发环境,则通过简单的 Windows 程序做出演示;在 Windows 编程模型中,需要关注如何处理消息事件,深刻理解处理消息循环的方式,还应该掌握使用 Visual C++集成环境下提供的高级资源,为后续在 VC++下使用网络编程做良好的铺垫。

第 3 章讲解网络通信代码编程的理论基础,介绍用 VC++实现网络基本应用的几个主题,包括 NetBIOS 编程接口的基本概念和编程基础方法;如何利用 NetBIOS 获取一个网卡或者多个网卡的 MAC 地址的操作实现,涉及到网络的 IP 地址及网络相关的关键概念;超级链接程序的原理和实现;如何获取域名和网卡类型等。

提高篇中包括第 5 章至第 8 章。提高篇中按照体系结构将网络按照层次展开,层层剖析,理论联系实际,从最熟知的应用层讲起,再到传输层、网络层和数据链路层。

第 5 章介绍应用层协议的体系结构,使读者从整体上了解和理解应用层协议。作为本书的核心章节之一,本章目的是向读者详细介绍基于 TCP/IP 的网络应用层协议的工作原理和功能,以便在用户网络应用程序的设计中使用这些标准的应用层协议。本章通过介绍应用层相关协议的原理和相关应用层程序的实例,使读者更容易理解和体会应用层协议是如何工作的。具体介绍一些应用层协议和实例,如 HTTP 协议及 HTTP 客户端和服务器的实例、FTP 协议及 FTP 客户端实例、POP3 协议及邮件客户端实例、SMTP 协议和 Telnet 协议等。

第 6 章主要介绍传输层和应用层之间的关系,针对网络通信中经常涉及的传输层中的面向链接的传输和面向无连接的传输做详细的讲解。在网络体系结构中,传输层是唯一一层用于总体的数据传输和数据控制的,并提供端对端的交换数据机制,对网络层提供可靠的目的地站点信息。还详细讲解 TCP 和 UDP 协议的工作原理和协议的报文格式,基于协议原理在 VC++环境下实现两种不同传输方式的网络聊天实例程序。

第7章中讲解的网络层是协议栈中最复杂的层次之一,并且将研究网络体系结构中为网络层提供服务的数
据链路层。本章列举的网络层各项功能及相关协议有助于理解网络层是如何实现主机到主机通信服务的,包括
网络层的转发和选路功能、著名的互联网协议、网络层编址等。针对数据链路层主要涉及 PPP 协议和广播链
路协议、网际控制报文 ICMP,最后用 ICMP 在 VC++环境下实现 Traceroute 程序,深入地运用 ICMP 报文的
格式理论。

进阶篇中的第9章到第13章均采用软件工程的方式进行实践开发,对于选择本书的读者,通过进阶篇的
学习不仅可以将网络通信知识和 VC++开发环境的使用融会贯通,而且还可以了解开发一个实用软件的流程。
了解实例原理就是需要掌握这个软件的需求是什么,只有了解需求,才可以量身定做实用的软件;设计程序结
构和流程就是为程序搭建合理的框架,就像建筑离不开框架一样,实用软件也需要合理的框架来维持;实例设
计这一步比较详细地设计各个类之间的接口等;最后的代码设计,即代码的详细实现,具体到每行的编码。

第9章是在 VC++下使用 Windows API 编程控制串口通信,通过虚拟终端的程序,从键盘输入的字符串传
送到串口上,从串口中获得的字符显示在屏幕上。从原理开始以流程图作为向导,对串口进行基本配置,完成
程序的编写,在实例设计中从函数的原型到函数的运用,对 Windows API 通信函数进行充分的学习,并且复
习串口的通信知识,以及串口的各项通信参数。

第10章介绍在 VC++环境下的基于对话框的多线程文件传输程序,通过实例原理、程序结构、程序流程、
程序设计、代码实现和程序演示等多个步骤,详细介绍实例的编写过程。实例中利用5个线程完成文件的传输,
一个监视线程监视文件传输,读者可以修改本实例代码来设置传输文件线程的数量,还可以修改传输文件的缓
冲区大小,加深读者对多线程网络文件传输的理解。

第11章是防火墙的设计与实现。介绍一个基于 Filter-hook drive 设计的数据包过滤防火墙,结合 Filter-Hook
Driver 的工作原理和特点,在 Windows XP 下使用 Filter-Hook Driver 创建过滤函数,利用设定好的过滤规则对
访问的数据包进行过滤,是这个防火墙的特点之一,可以阻止特定的 IP 地址和端,并且可以选择需要阻止的
协议,从而达到保护系统安全的功能。

第12章是在 VC++ 6.0 下成功地设计实现一个 Foxmail 邮件转发器。在程序设计的过程中,除了注重对
SMTP 协议原理的运用,同时还很好地利用 MFC 和 VC++提供的一些特性,通过对实例的跟踪学习达到对第5
章讲解到的 SMTP 协议规范的进一步熟悉,并能够熟练在 VC++ 6.0 环境下进行编程。

第13章是通过实例基于 Telnet 的 BBS 客户端的设计,从而实践 Telnet 的通信原理。从远程登录的基本原
理,代码结构的框架设计,代码的详细设计逐步深入到核心代码内部进行剖析,其中对主程序控制代码将进行
详细的讲解。

0.4 适用读者

本书是一本学习网络通信非常有价值的书。如果读者是编程的初学者,请从头学习本书,本书将引领初学
者一步一个脚印、扎实地学习 VC++网络通信。如果读者有过一定的面向对象的编程经验,可以在大致浏览本
书的基础篇后,快速跳到提高篇中去,提高篇是本书的核心。如果您有过 VC++的编程经验,请直接阅读提高
篇和进阶篇。对于想快速寻找一个网络通信方面实例的读者,请参考每章的实例部分及进阶篇中的实例。本书
可以作为一本完整的教程,也可以作为一本参考资料放在手边查阅。

编 著 者

联系方式

咨询电话: (010) 68134545 88254160

电子邮件: support@fecit.com.cn

服务网址: <http://www.fecit.com.cn> <http://www.fecit.net>

通用网址: 计算机图书、飞思、飞思教育、飞思科技、FECIT

目 录

第 1 章 Visual C++网络通信基础	1	3.2.1 原理	78
1.1 计算机网络的组成及体系结构	2	3.2.2 实现程序	85
1.1.1 网络边缘	2	3.3 超链接程序的原理与实现	88
1.1.2 网络核心	4	3.3.1 原理	89
1.1.3 计算机网络的分层体系结构	6	3.3.2 实现程序	90
1.2 初识 Windows Sockets		3.4 获取域名和网卡类型的原理	
编程规范	7	和实现	96
1.2.1 Windows Sockets 的相关概念	8	3.4.1 原理	97
1.2.2 Winsock 技术特点	9	3.4.2 实现程序	100
1.2.3 Winsock 编程原理	10	3.5 小结	104
1.2.4 实例——Winsock 实现基于		第 4 章 串口通信及其实例	105
TCP 的客户端/服务器通信	17	4.1 串行通信原理	106
1.3 利用 MFC 网络编程	21	4.1.1 串行通信基本概念	106
1.3.1 CAsyncSocket 类的基本组成	21	4.1.2 单工、半双工和全	
1.3.2 CSocket 类的基本组成	24	双工定义	107
1.3.3 Wininet 类的基本组成	25	4.1.3 串行通信协议	108
1.3.4 实例——基于 CSocket		4.1.4 串行通信方式	110
的网络聊天室	27	4.2 MSComm 控件	111
第 2 章 认识 Windows 编程模型	38	4.2.1 VC++ 中的 MSComm 控件	111
2.1 Windows 初级编程模型	39	4.2.2 实例——MSComm 多串	
2.1.1 匈牙利命名法	39	口通信	117
2.1.2 Visual C++ 使用入门	41	4.3 Windows API 串口通信编程	124
2.1.3 最简单的 Windows 应用程序	43	4.3.1 Windows 串口通信 API	
2.2 Windows 应用程序剖析	45	函数	125
2.2.1 真正的 Windows 应用程序	45	4.3.2 VC++ 中的 CSerialPort 类	127
2.2.2 Windows 程序分析	50	4.3.3 实例——串口的多线程	
2.2.3 重要的消息事件处理	57	通信	138
2.2.4 Windows 控件的应用	59	4.4 小结	143
2.3 Windows 高级编程	62	第 5 章 应用层协议及编程实例	144
2.3.1 利用 Windows 菜单		5.1 应用层协议体系结构	145
中的位图资源	63	5.1.1 应用层协议原理	145
2.3.2 利用资源制作菜单	66	5.1.2 网络应用程序的体系结构	145
2.4 小结	67	5.2 HTTP 协议	146
第 3 章 网络基本应用在 VC++		5.2.1 HTTP 协议	146
中的实现	68	5.2.2 实例——基于 VC++ 的	
3.1 获取网卡的 MAC 地址	69	HTTP 客户端程序	151
3.1.1 原理	69	5.3 Email 协议及电子邮件	155
3.1.2 实现程序	74	5.3.1 SMTP 协议	155
3.2 获取网络中计算机的 IP		5.3.2 POP3 模型及会话过程	159
地址和计算机名	78	5.3.3 实例——Email 接收程序	163

5.4	FTP 文件传输协议	177	8.5	实时监测网络流量	338
5.4.1	FTP 工作原理	177	8.6	小结	353
5.4.2	FTP 服务器实例	185	第 9 章	基于 Windows API 的	
5.5	Telnet 协议	207		虚拟终端实现	354
5.5.1	Telnet 命令	208	9.1	实例原理	355
5.5.2	Telnet 操作	211	9.2	程序结构和流程图	360
5.6	实例——HTTP		9.3	实例设计	361
	服务器端程序	211	9.4	代码设计	371
5.7	小结	232	9.5	实例演示	385
第 6 章	传输层协议及编程实例	233	9.6	小结	388
6.1	计算机网络传输层概述	234	第 10 章	多线程网络文件传输	
6.1.1	传输层概述	234		的设计与实现	389
6.1.2	传输层的作用	236	10.1	实例原理	390
6.2	UDP 无连接的传输	237	10.2	程序结构和流程图	392
6.2.1	UDP 报文结构	237	10.3	实例设计	393
6.2.2	UDP 协议特点	238	10.4	代码实现	394
6.2.3	实例——基于 VC++ 用 UDP		10.5	实例演示	413
	实现点对点聊天程序	239	10.6	小结	415
6.3	TCP 面向连接的传输	248	第 11 章	防火墙的设计与实现	416
6.3.1	TCP 报文结构	248	11.1	实例原理	417
6.3.2	TCP 协议特点	250	11.2	程序结构与流程图	425
6.3.3	TCP 通信	252	11.3	实例设计	431
6.3.4	实例——基于 VC++		11.4	代码设计	435
	用 TCP 实现客户机		11.5	实例演示	451
	服务器模式的聊天程序	259	11.6	小结	453
6.4	小结	266	第 12 章	Foxmail 邮件转发器	
第 7 章	网络层协议和数据链路层	267		设计与实现	454
7.1	网络服务模型概述	268	12.1	实现原理	455
7.1.1	网络层概述	269	12.2	程序结构和流程图	456
7.1.2	网络层与传输层的关系	279	12.3	实例设计	457
7.2	数据链路层服务	283	12.4	代码设计	460
7.2.1	PPP 协议	285	12.5	实例演示	478
7.2.2	广播链路协议的划分	289	12.6	小结	479
7.3	ICMP 协议	295	第 13 章	基于 Telnet 的 BBS	
7.3.1	数据报格式	295		客户端的设计和实现	480
7.3.2	实例——用 ICMP		13.1	实现原理	481
	实现 Traceroute 程序	299	13.2	实例结构设计	484
7.4	小结	312	13.3	代码设计	485
第 8 章	Internet 通信原理及		13.4	实例演示	506
	编程实例	313	13.5	小结	508
8.1	WinInet 编程技术	314			
8.2	Web 浏览器的实现	317			
8.3	从 Internet 上下载文件	323			
8.4	强制过滤弹出的窗口	328			

第 1 章

Visual C++网络通信基础

自 20 世纪 60 年代出现计算机网络开始，计算机和网络通信的出现给人们的生活带来了革命性的变化。从网络聊天、网上购物、网络游戏到几乎每个办公桌上的 PC，都具有了连网功能。计算机的硬件离不开软件，通信软件则使计算机网络硬件的具体功能得以实现。网络通信软件是网络通信的灵魂。Microsoft Visual C++是由微软开发的可视化软件开发平台，具有强大面向对象功能，在此平台下可以利用微软已经封装好的基本类库，来开发网络通信软件。

下面将向读者介绍网络通信的基本理论知识，结合理论知识，再联系 Visual C++平台开发出一系列实例，读者学习这些简单易懂的实例，能够深入浅出地理解网络通信的原则，并通过理论和实例的结合更加深刻地认识网络通信的实质，最后可以举一反三地利用 Visual C++编写出自己需要的网络通信软件。

本章概述 Visual C++网络通信基础，其中包括计算机网络的组成、计算机网络的分层体系结构、认识 Winsock 编程、学习 Winsock 的相关概念和技术特点，以及理解 Winsock 的编程原理，最后利用 MFC 进行网络编程，利用 CSocket 类制作网络聊天室。

1.1 计算机网络的组成及体系结构

计算机网络是一个比较复杂的系统，这里所指定的计算机网络为因特网。本节开始从日常使用的计算机开始，即网络边缘，然后深入到网络的核心，最后讲解计算机网络的分层体系结构。

1.1.1 网络边缘

因特网遍布世界各地，是一个世界范围的计算机网络。由因特网互连的计算机设备有传统的个人电脑、办公电脑，有基于 UNIX 的工作站及用于存储的 Web 页面的网络服务器。当然随着计算机网络的发展，诸如移动计算机、PDA 个人数字助手、无线传感器、网络电视等设备，也可以由网络连接起来供人们使用。以上提到的这些设备，由于处于因特网的边缘，因而被称之为端系统，如图 1-1 所示。

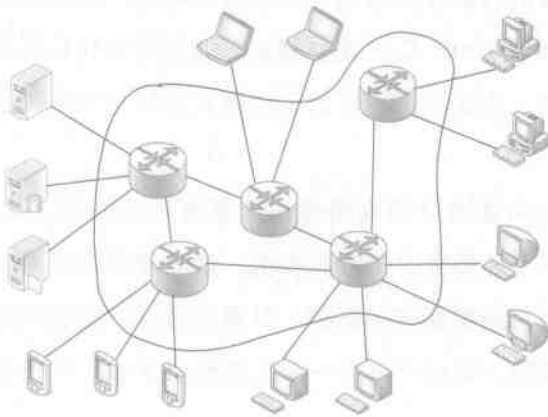


图 1-1 端系统示意图

图 1-1 中所表示的网络符号对应表 1-1。

表 1-1 网络符号对照表

表示含义	路 由 器	服 务 器	PDA	终 端	PC	便 携 电 脑
网络符号						

端系统通过因特网服务提供商 Internet Service Provider (ISP) 连接入因特网，不同的因特网服务提供商为处于网络边缘的端设备提供不同类型的网络服务，这种不同体现在网络速度、网络内容及网络接入形式。

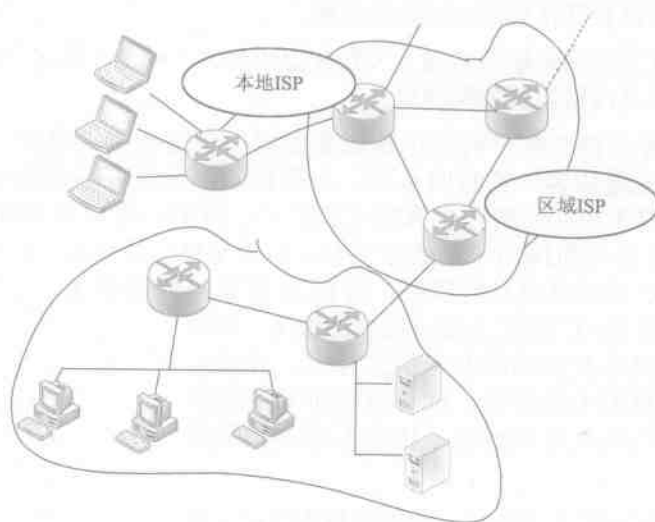


图 1-2 网络端系统和 ISP 示意图

端系统在不同的情况下也有不同的称呼。当端系统作为运行 Web 浏览器、Web 服务器程序、电子邮件阅读程序时，端系统称之为主机。一般情况下主机和端系统也可以通用。主机还可以进一步划分为客户机 Client 和服务端 Server。

使用不同的物理介质可以将端系统连接在一起，物理介质包括同轴电缆、双绞线和光线等。不同物理介质的传输速度不同适用的范围也不同。

同轴电缆可分为两类：粗缆和细缆，这种电缆在实际应用中很广，比如有线电视网，就是使用同轴电缆。不论是粗缆还是细缆，其中央都是一根铜线，外面包有绝缘层。同轴电缆由内部导体环绕绝缘层及绝缘层外的金属屏蔽网和最外层的护套组成。这种结构的金属屏蔽网可防止中心导体向外辐射电磁场，也可用来防止外界电磁场干扰中心导体的信号。双绞线（TP:Twisted Pairwire）是布线工程中最常用的一种传输介质。双绞线是由相互按一定扭距绞合在一起的类似于电话线的传输媒体，每根线加绝缘层并有色标来标记。成对线的扭绞旨在使电磁辐射和外部电磁干扰减到最小。目前，双绞线可分为非屏蔽双绞线（UTP:Unshielded Twisted Pair）和屏蔽双绞线（STP:Shielded Twisted Pair）。我们平时一般接触比较多的就是 UTP 线。光缆不仅是目前可用的媒体，而且是今后若干年后将会继续使用的媒体，其主要原因是这种媒体具有很大的带宽。光缆是由许多细如发丝的塑胶或玻璃纤维外加绝缘护套组成，光束在玻璃纤维内传输，防磁防电，传输稳定，质量高，适于高速网络和骨干网。光纤与电导体构成的传输媒体最基本的差别是，它的传输信息是光束，而非电气信号。因此，光纤传输的信号不受电磁的干扰。

用物理介质连接起来的端系统形成一个通信链路。从发送端系统到接受端系统，数据经由的一系列的设备称之为网络路径。

系统之间的通信犹如人类之间的交互活动，需要使用一定的沟通标准，而这种沟通的标准无处不在。例如，两个人之间的对话，需要使用相同的语言，否则就产生了沟通的障碍，只有使用相同的语言才能够使对话成功。在对话过程中，对于不同的回答需要产生不同的动作回应。这也是协议的一个要素。在计算机网络中，网络协议定义了两个或者多个实体之间交换报文的格式和次序，并且在报文传输的过程中传输或者接受其他事件所采取的动作。在

本书后续章节将对协议及其使用进行详细的讲解。

对于端系统彼此之间的通信因特网上大致分为两种：一种是面向连接的，一种是面向无连接的。这两种方法是网络实现的两种途径。

面向连接的服务是指当客户机程序和服务器程序发送数据的过程中要彼此发送控制分组。通过握手机制通知客户机和服务器，为数据的到达做好应有的准备，当握手的过程结束，那么两个端系统之间的连接就建立了起来。例如，假设 A 和 B 两个人需要发送信件，信件就类似于计算机网络中的数据信息。如果是把信件交给一位可信的朋友 C，由 C 来传送信件，当 C 把信件成功传送后，告知 A 和 B 信件已经发送。在这种方法中，传送的两端都保持着联系，C 提供了面向连接的服务。在面向连接的方法中，网络负责顺序发送报文分组，并且以一种可靠的方法检测丢失和冲突，同时提供可靠的数据传输服务。发送方与接收方保持联系协调发送和接受数据信息。

面向连接服务要经过 3 个阶段：数据传送前先建立连接，连接建立后再传输数据，数据传送完后释放连接。面向连接服务，可确保数据传送的次序和传输的可靠性。



图 1-3 面向链接的服务示意图

在面向无连接的方法中，网络只需要将报文分组发送到接收点，检错与流控由发送方和接收方处理。这种方法被称为“最佳工作（best-effort）”或“无应答（unacknowledged）”的传输协议所使用。假设 A 需要邮寄若干封信件，在信封上注明地址并将它们投进邮局就可以了，但是 A 并没有得到保证说每封信都会达到目的地的承诺。如果都到达了，这若干封信件也可能是在不同的时间到达并且不是连续的，这就像一个网络中的无连接服务。

无连接服务的特点是：不需要建立连接，直接开始数据传送阶段，消除了除数据通信外的其他开销。只要存在发送端就可以发送，不要获得接收端的认可。无连接服务的优点是灵活方便、传送速度快，特别适合于传送少量的琐碎数据信息，但是无连接服务不能防止信息在传递过程中的丢失、重复或顺序错乱。

在面向链接的服务和面向无连接的服务区分中可以用比较形象的例子来做类比。面向链接的服务，就好像是在打电话，如果达到两端通话的目的，就要首先建立链接（即拨号），然后等待对方应答后才开始传递信息（即开始通话）；最后就是信息传递完毕需要结束通话（挂电话）。而面向无连接的服务就好像是写信，写信只要将姓名地址写清楚投递到邮箱里面就可以了，面向无连接的服务就是只要发送端将发送的信息发送出去即可。

面向链接的服务主要应用于，远程登录 TelNet、文件传输 FTP 及 HTTP 中，面向无连接的服务主要用于网络电话和视频会议等。

1.1.2 网络核心

数亿的端系统通过路由器连接起来，这个由路由器或者交换机连接起来的网状网络，称之为网络核心，如图 1-4 所示。

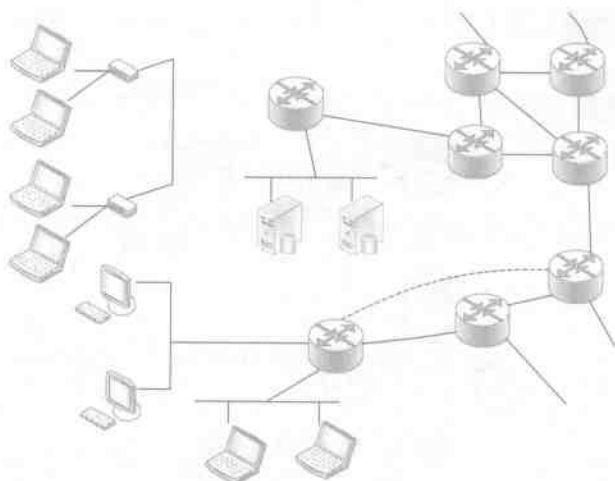


图 1-4 网络核心示意图

如何建立网络核心呢？建立网络核心有两种方法：一种是电路交换，一种是分组交换。电路交换是指当两台主机需要通信时，该网络在两台主机之间创建的一条专用的连接，就好像我们现在用的电话的通信方式。这种方式和 1.1.1 节中所讲到的面向连接的服务相比，电路交换是一种实际的连接，需要线路上的发送方和接受方的路径保持连接状态。如图 1-5 所示，计算机 A 为了和计算机 B 通信，在网络上的两台主机之间先创建了一条专用的通路，这种方式就是电路交换。

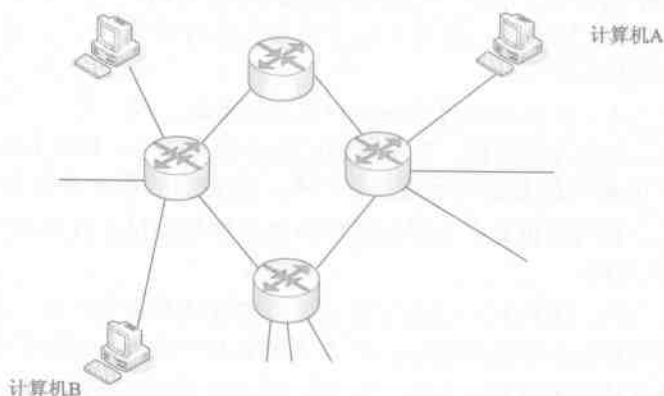


图 1-5 电路交换示意图

分组交换的方式是将发送的信息分成较小的数据块在发送方和接受方之间交换数据，这种被拆分的小的数据块称为分组，所以这种方式成为分组交换。如图 1-6 所示，计算机 A 和计算机 B 需要发送数据，则通过路由器或者分组交换机将数据存储，然后按照分组转发到目的地计算机 C，在网络核心中传送的数据被拆成分组，在计算机终端又被重新组合起来得到正确的信息。

我们所使用的通信网络要么是电路交换要么是分组交换。电路交换和分组交换各有特点，电路交换可以保证数据交换的稳定性，但是需要提前预留链路；而分组交换的更为简单，并且可以使数据共享网络带宽，但是可能会有一定的时间延迟。

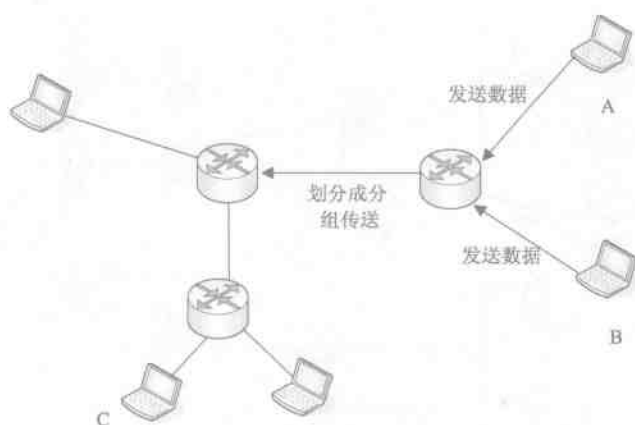


图 1-6 分组交换示意图

1.1.3 计算机网络的分层体系结构

计算机网络是一个比较复杂的系统，有大量的应用程序和协议，如果没有一个体系结构将不堪设想，统一的标准将应用程序和协议合理地划分开。因特网中综合考虑硬件和软件的因素，将计算机网络分为 5 层：应用层、运输层、网络层、链路层和物理层，如图 1-7 所示。

应用层处于体系结构的最上层，其包括的协议最多，如 HTTP 协议、SMTP 协议、Telnet 协议等，在第 5 章中将会有详细的讲解。处于应用层的信息称为报文。

传输层在应用层之下，负责应用程序的客户机和服务器之间的信息传递，传递的信息称为报文段。处于传输层的协议有两个：TCP 和 UDP。TCP 提供了面向链接的服务，UDP 提供的是面向无连接的服务。在本书的第 6 章将会有详细的讲解。

网络层的数据信息称为数据报。在因特网中网络层将数据报在两台主机之间传递。其中有一个非常著名的 IP 协议。

链路层的信息称为帧，因特网的链路层通过路由器存储和转发信息，链路层为网络层提供服务，网络层将数据报向下传给链路层。在本书的第 6 章也会做详细的讲解。

物理层处理体系结构的最低层，在这一层规定的协议和所用的物理介质有关。由于在因特网的分层结构中，处于各个网络层次中有不同的协议，所以也称之为协议层。知道了这 5 个网络层次，那么在各个层次之间的数据是如何传输的呢？看了下面的图 1-8 就会明白。



图 1-7 因特网分层体系结构



图 1-8 网络分层体系结构中的数据流

在网络的各个层次中的数据信息都可以统称为分组，不同的层次封装成不同的信息格式，从图中可以看出在每个层次上都加上了本层的首部信息，但是对于处于同一个层次的数据可以互相理解，每一个层次的数据都和上下层之间紧密联系。

在了解了计算机网络的组成和体系结构后，下面从 Windows Sockets 来开始我们的 Visual C++ 网络通信之旅吧！

1.2 初识 Windows Sockets 编程规范

网络中普遍的服务方式为客户机/服务器模式 (Client/Server model)，因此在网络编程中最常用的方案为 C/S 模型。在该服务模式中，客户端应用程序向服务器程序提出服务请求，而服务程序通常在一个开放的地址监听对各类服务的请求。可以这样理解：服务进程一直处于休眠状态，直到某一客户程序向这个服务的地址提出了连接请求，此时，服务程序被“惊醒”，并且为客户提供服务——对客户请求做出适当的反应。

为了方便这种 Client/Server 模型的网络编程，20 世纪 90 年代初，由 Microsoft 联合其他几家公司共同制定了一套 Windows 下的网络编程接口，即 Windows Sockets 规范，简称为 Winsock，它不是一种网络协议，而是一套开放的、支持多种协议的 Windows 下的网络编程接口。该编程规范从 Berkeley Sockets 扩展而来，在继承 Berkeley Sockets 的基础上，又进行了新的扩充，主要是提供了一些异步函数，并增加了符合 Windows 消息驱动特性的网络事件异步选择机制。目前使用的 Winsock 已经基本上实现了与协议的无关性，可以使用 Winsock 来调用多种协议的功能，较常使用的是 TCP/IP 协议。

套接字是通信的基础，是支持 TCP/IP 协议的网路的基本操作单元，套接字实际在计算机中提供了一个通信端口，可以通过这个端口与任何一个具有套接字接口的计算机通信。应用程序在网络上传输、接收的信息都通过这个套接字接口来实现。在进行信息传输过程中，其中某个应用程序将需要传输的信息写入该应用程序所在系统的套接字中，套接字通过网络接口卡 (NIC 卡) 的传输媒介将该信息发送至另一系统的套接字接口中，通过该套接字接口将信息传递给对应的应用程序执行相关操作。

使用 Winsock 在 Internet 上传输数据和交换信息时，由于不需要关心网络连接的细节而受到广大网络编程人员的喜爱。网络应用程序调用 Winsock API 函数实现相互之间的通信，同时，Winsock 利用下层的网络通信协议和操作系统实现实际的通信，如图 1-9 所示。

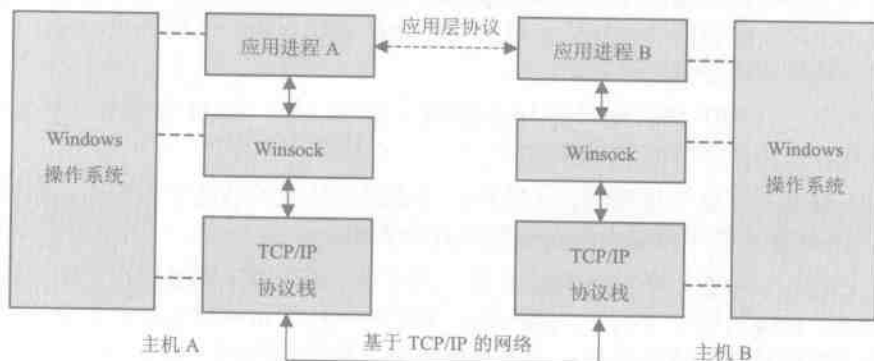


图 1-9 应用进程使用 Winsock 进行网络通信

1.2.1 Windows Sockets的相关概念

1. Windows Sockets 的版本

常用的 Windows Sockets 有两个版本: WinSock 1.1 网络编程接口和 Winsock 2.2 网络编程接口。WinSock 1.1 由动态链接库 WINSOCK.DLL 支持, 主要应用在 Windows 95 中, Winsock 2.2 由动态链接库 WINSOCK32.DLL 支持, 主要应用在 Windows 98 和 Windows 2000 中。Winsock 2.2 与低版本的 WinSock 1.1 相比, 主要对一些协议进行了扩充, 如 IPX、NETBIOS 等, 同时, Winsock 2.2 与 WinSock 1.1 的函数完全兼容, 因此低版本的 WinSock 应用程序可以在 Windock2 库的加载下正常运行。

2. 通信域

通信域是为了处理一般的线程通过套接口通信而引进的一种抽象概念, 套接字存在于通信域中, 套接字通常和处于同一个域中的套接字交换数据。可以这样理解, 套接字实际上是通过网络中的各类协议进行相互通信, 如果两台主机之间使用协议不一致, 通信必定出现故障, 为了保证通信的正常, 套接字所连接的两端需要使用相同的通信协议。Windows Sockets 规范支持单一的通信域, 即 Internet 域。各种进程使用这个域互相之间用 Internet 协议族来进行通信。需要说明的是, Windows Sockets 1.1 以上的版本支持其他的域。

3. 阻塞和非阻塞

套接字具备两种模式: 阻塞和非阻塞模式。当套接字处于阻塞模式时, 调用高模式的函数时, 需要消耗一定的时间来等待操作的完成, 而当套接字处于非阻塞模式时, 调用函数会立即返回, 但通常会出错。当一特定的 Winsock API 函数指出一个错误已经发生时, 需要获取对应的错误代码, 根据 Winsock 版本的不同, 包含在不同的头文件中。

4. 网络字节顺序

不同的主机对于字节的存储方式不同, 确切地说, 存储的字节顺序不同。如 Intel 处理器的字节顺序为小序在前 (Little-Endian), 而 Solaris 计算机则使用相反的大序在前 (Big-Endian) 的存储方式。小序在前指的是计算机在起始地址处存放整数的低序号字节, 大序在前则在起始地址处存放整数的高序号字节。

在计算机中, TCP/IP 协议使用的 16 位整数 (如端口号) 和 32 位整数 (如 IP 地址) 是按照计算机各自的主机字节顺序来表示的。

为了保证数据在网络中传输时的正确性, 需要确定特定的网络字节顺序, 按照“互联网网标准”, 网络字节顺序按照大序在前的形式来表示。

如果用户输入一个数, 并且指定使用某一端口号, 应用程序则必须在使用该端口号建立地址之前把它转换成网络字节顺序, 相应地, 如果应用程序希望显示包含于某一地址中的端口号, 则这一端口号就必须在被显示之前从网络字节顺序转换到主机字节顺序。

在很多情况下, 上述的转换是无法避免的, 下面的 API 函数将一个数从主机字节顺序转