



光盘内附书中范例

专门为 Delphi 使用者设计的深入编程经典书

掌握组件进行 Delphi 环境下的网络编程

充分领略 Delphi 中线程运用等的强大功能

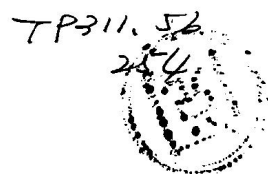


静海 著

深入 Delphi 6

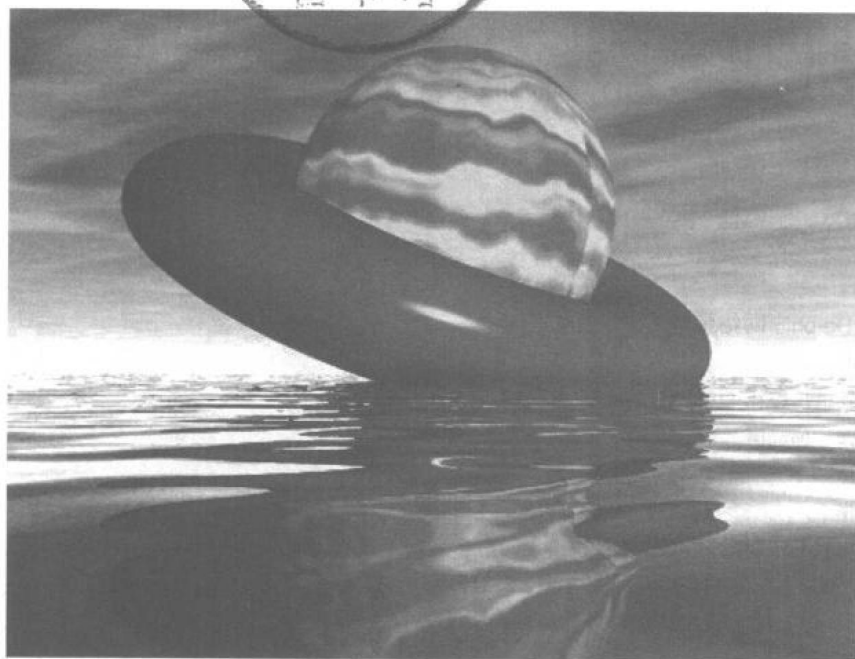
网络编程

00200939



深入 Delphi6

网络编程



JS865/04

中国铁道出版社

2001年·北京



北航

C0631680

(京)新登字 063 号

内 容 简 介

本书系统地介绍了编程工具 Delphi 及其网络组件、Delphi 中线程的应用、利用 WinSock API 进行网络编程等,并通过程序实例浅显易懂地说明各组件及 WinSock API 在 Delphi 环境下的网络编程,能够帮助读者清楚地掌握 Delphi 程序设计的精髓,书中配有范例光盘,以供读者学习参考。

图书在版编目(CIP)数据

深入 Delphi6 网络编程/静海著. —北京:中国铁道出版社, 2001.8

ISBN 7-113-04340-2

I. 深… II. 静… III. Delphi 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 061582 号

书 名:深入 Delphi6 网络编程

作 者:静 海

出版发行:中国铁道出版社(100054,北京市宣武区右安门西街 8 号)

策划编辑:苏 茜 郭晓溪

特邀编辑:陆奕伶

封面设计:冯龙彬

印 刷:北京市兴顺印刷厂

开 本:787×1092 1/16 印张:21.5 字数:515 千

版 本:2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

印 数:1~5000 册

书 号:ISBN 7-113-04340-2/TP·607

定 价:39.00 元

版权所有 盗印必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

前言

本书是针对 Delphi 程序员编写的，它讲解了在 Delphi 环境下进行网络编程的一些知识，特别是对基于 WinSock API 的底层网络程序编写给出了一些具体的实例，以方便大家对其内容的掌握。

但本书又不仅仅局限于单一的网络编程——它还涉及到和网络编程以及其他应用程序编写密切相关的内容，比如多线程的应用、Delphi 核心 VCL 组件源代码分析等等。本书主要是面向中、高级程序员，但是，如果你是一位 Delphi 的初学者，并对用 Delphi 进行网络编程怀有浓厚的兴趣，我想本书也一定不会让你失望，因为本书的每一个例子都讲解得很具体，并加有详细的注释，会使你受益非浅。

笔者始终认为，编程是一门艺术，它同绘画、雕塑一样，任何一件杰出作品的诞生，都是其创作者智慧的结晶，都凝聚着他们汗水和心血。

一个真正和程序员，会不断探索编程过程中出现的各种问题，研究其中的奥秘。他们可以几天几夜不睡，只为了解决一个技术上的难题，同时他们在这种发现与创造的过程中享受无穷的乐趣，体现自己的价值。他们会尽全力使自己的作品更完满、更出色。

一个优秀的程序员，必定是一个真正的程序员，此外，还有一个基本的标准，那就是：他必须善于和别人合作，并且乐于同大家一起分享他的劳动成果。就像一位画家，技术高超而且创造力丰富，但如果他只把他的画摆在自己的卧室中孤芳自赏的话，他永远也成不了一名杰出的艺术家。

我只是一名普通的程序员，但是我渴望成为一名优秀的程序员。我知道要达到这一境界，还有很长的路要走。但是我很希望能把自己在学习过程中遇到的一些问题，取得的一些经验拿出来与大家一起研讨，并期望通过这种方式能给大家一点点帮助，哪怕是少走一些弯路也好。同时，我也希望在和大家交流的过程中，自身有所提高。简而言之，我希望这本书能起到抛砖引玉的作用，使我们一起百尺竿头，更进一步。这就是我写这本书的目的。

本书的组织

本书由七个章节组成，主要包括的内容有：

- * 用组件进行 Delphi 环境下的网络编程
- * 基于 WinSock API 的 Delphi 环境下网络编程
- * Delphi 中线程运用的深入分析
- * 与网络安全有关的网络编程应用

第 1 章介绍了我们基本的编程工具——Delphi，并将其与 Visual C++ 进行了一些比较。我想，这对那些还拿不定主意学习哪种语言的读者会有帮助。同时，这一章还提到了一些学习后续章节所需具备的知识，为读者快速进入状态打下了一定的基础。

深入 Delphi6 网络编程

第2章介绍了 Delphi 中的网络组件。这本书绝对不是组件参考大全或类参考手册，所以我只精选了一些常用的组件，力求基本，并且这些组件在第3章的例子程序中都会用到。

第3章是本书的重点章节。它通过举例子，详细地说明了在 Delphi 中如何用组件进行网络程序的编写。每一个例子都很具体，其中不仅有网络编程方面的知识，还介绍了一些小技巧，你扩展一下就可以写一个小应用程序了。

第4章介绍了 Delphi 中线程的应用。表面上看，它与网络编程关系不大，但是实际上，它对后面章节中利用 WinSock API 进行编程至关重要。即使你不将它用在网络编程方面，你也会在其他 Windows 程序编写过程中，体会精通了它之后的巨大作用。在本书中，将针对 Delphi 5、Delphi 6 中线程类的源代码进行深入分析，帮助大家理解其实质。本章节难度偏大，但我相信大家会受益非浅的。

第5章介绍了利用 WinSock API 进行网络编程相关的一些基本知识和概念，并列举说明了一些常用函数。同样，你不能期望它像 API 参考手册那样详尽，但它们都将在后面的例程中被应用。值得一提的是，在本章中，还分析了 Delphi 对 WinSock API 函数的封装，特别是 TServerSocket、TClientSocket 两个组件源代码的剖析，对我们理解其内部机制大有帮助。但该部分难度偏大，大家应该循序渐进地进行学习。

第6章也是本书的重点章节。它用举例子的方式详细地说明了 Delphi 中对 WinSock API 函数的调用，你可以慢慢体会其中所运用的方法和所体现的思想。

第7章介绍了网络安全方面的一些知识，包括密码学基础和一些应用协议。同时，在前面章节的基础上，尝试着将网络编程技术向安全方面的应用靠近。也许在形式上和实例上它们都是很粗浅的，但是从思维方式上讲，它们会对你有所启迪，可能还会影响深远。

另外，本书还配有配套光盘，其中包含了本书所有例程的源代码，以及一些常用的文档和工具软件。

我衷心希望本书能成为你学习 Delphi 编程的得力助手，并相信它不会令你失望。

相互讨论

我已尽力保证本书及配套光盘内容准确无误，但由于本人水平有限，加之时间仓促，书中内容难免有不当之处。如果你发现了其中的错误，或有好的建议，或者希望就技术等问题与我交流，请发邮件给我。我会虚心听取您的意见和建议。邮件请用纯文本方式。

邮件地址是：goldensplit@263.net

感谢

首先，我要感谢带我走入 Delphi 殿堂的赵燕军、陈浩，是她们使我认识了 Delphi，并给了我许多的帮助和指导。其次，赵东华、蔡宝忠、郭晓溪、赵爽、冯泽波、张群在编写此书的过程中给了我许多的支持和鼓励，没有他们，我不可能这样快的出版此书。另外，王春、李小凤、高沿、周子中、陈笑、刘恒、张宁、吴天乐、杨坡、田波、梁超、李晓含、姜伟等参与了此书的编辑、整理工作，在此一并表示感谢。

作者

2001年7月

编 委 会

编委 赵燕军 陈 浩 赵东华 蔡宝忠
 赵 爽 冯泽波 张 君 静 海
 郑 芳 许卓鲲

★ 作者 E-Mail: pubkj_008@263.net



目 录

第 1 章 引言与概述.....	1
1-1 网络编程需求.....	2
1-2 Delphi 与 Visual C++.....	2
1-2-1 相似点	2
1-2-2 不同点	3
1-2-3 Visual C++ 的优势	3
1-2-4 Delphi 的优势	3
1-2-5 如何取长补短	5
1-3 如何快速进入状态.....	6
1-3-1 不要误解标题	6
1-3-2 要做的准备工作	6
1-4 Delphi 编写网络程序的几种方式.....	7
1-4-1 利用 Delphi 的网络组件	7
1-4-2 使用第三方网络组件	9
1-4-3 使用 WinSock API.....	10
1-4-4 利用自己编写的网络组件	10
1-4-5 利用 ActiveX.....	10
1-5 小结	11
第 2 章 Delphi 的网络组件.....	13
2-1 网络组件概述.....	14
2-2 常用网络组件精解.....	15
2-2-1 TClientSocket 与 TServerSocket 组件.....	15
2-2-2 NMPop3 、NMSMTP 与 NMUUPProcessor 组件.....	23
2-2-3 到 TWebModule 与 TPageProducer 组件.....	25
2-3 网络组件使用技巧.....	30
2-3-1 正确的使用帮助	30
2-3-2 阅读 Delphi 组件源代码	32
2-3-3 在实践中摸索	34
2-3-4 在理论中升华以及完善	34
2-4 使用第三方组件.....	34
2-4-1 为什么会存在第三方组件	35

深入 Delphi6 网络编程

.....

2-4-2 如何正确使用第三方组件	35
2-4-3 知名第三方网络组件 ICS 简介	38
2-5 小结	39

第 3 章 基于组件的 Delphi 网络编程

41

3-1 概述	42
3-2 起步——分析一个简单的网络聊天程序	42
3-3 进一步——自己写一个聊天程序	45
3-3-1 导读	45
3-3-2 明确目的	46
3-3-3 回顾 TServerSocket 和 TClientSocket 组件	46
3-3-4 动手过程	48
3-3-5 不足之处	74
3-4 进一步——写一个 HTTP 协议分析器	74
3-4-1 导读	74
3-4-2 明确目的	75
3-4-3 整体结构分析	75
3-4-4 动手过程	75
3-4-5 不足之处	105
3-4-6 动脑时间	105
3-5 应用——CGI、NSAPI 与 ISAPI	106
3-5-1 导读	106
3-5-2 一点说明	106
3-5-3 CGI、NSAPI 与 ISAPI 比较	106
3-5-4 回顾 TWebModule 与 TPageProducer 组件	107
3-5-5 动手写一个 ISAPI 应用程序	108
3-5-6 程序的调试	128
3-5-7 动脑时间	129
3-6 完善——用用第三方组件	130
3-6-1 导读	130
3-6-2 明确目的:写一个 FTP 客户端程序	130
3-6-3 ICS 的 Ftp 组件	130
3-6-4 动手过程	130
3-6-5 动脑时间	146
3-7 小结	146

第 4 章 线程应用

147

4-1 概述	148
4-2 为什么要写多线程的程序	148



4-3 利用 Windows API 创建多线程程序	149
4-4 Delphi 中的 TThread 类	151
4-4-1 Thread Object——Delphi 的同步对象	151
4-4-2 TThread 类源代码分析	153
4-5 进程同步	161
4-5-1 简介	161
4-5-2 临界区应用	161
4-5-3 互斥元 (Mutex) 应用	163
4-5-4 信号量应用	165
4-5-5 事件应用	166
4-6 使用 Synchronize 方法	167
4-7 Delphi 6 中线程类的一些改动	170
4-8 动脑时间	179
4-9 小结	179
第 5 章 WinSock API 与 Delphi	181
5-1 概述	182
5-2 WinSock 基础	182
5-2-1 TCP、UDP 和 IP 协议	182
5-2-2 套接口 (Socket) 和 Winsock API	182
5-2-3 面向连接和无连接	183
5-2-4 客户/服务器模式	184
5-2-5 套接口类型	184
5-2-6 使用面向连接的协议时套接口的调用	184
5-2-7 使用无连接的协议时套接口的调用	186
5-2-8 字节顺序问题	187
5-3 WinSock API 常用函数	189
5-3-1 accept 函数	189
5-3-2 bind 函数	190
5-3-3 closesocket 函数	191
5-3-4 connect 函数	191
5-3-5 htons 函数	191
5-3-6 inet_addr 函数	192
5-3-7 listen 函数	192
5-3-8 recv 函数	193
5-3-9 select 函数	193
5-3-10 send 函数	194
5-3-11 socket 函数	195
5-3-12 gethostbyname 函数	195

5-3-13	WSACleanup 函数.....	196
5-3-14	WSAStartup 函数.....	197
5-4	WinSock API——Delphi 组件的基础.....	199
5-4-1	回顾 TServerSocket 和 TClientSocket 组件.....	199
5-4-2	分析 TServerSocket 和 TClientSocket 的源代码.....	199
5-4-3	关于使用 WinSock API 写网络组件.....	206
5-5	Delphi 对 WinSock API 的封装.....	206
5-6	小结.....	207
第 6 章	基于 WinSock API 的网络编程.....	209
6-1	概述.....	210
6-2	起步——写一个文件传送程序.....	210
6-2-1	导读.....	210
6-2-2	整体结构分析.....	210
6-2-3	动手过程.....	211
6-2-4	不足之处.....	227
6-2-5	动脑时间.....	228
6-3	提高——写个简单的 HTTP 代理程序.....	228
6-3-1	导读.....	228
6-3-2	明确目的.....	228
6-3-3	整体结构及原理分析.....	229
6-3-4	动手过程.....	229
6-3-5	不足之处.....	246
6-3-6	动脑时间.....	247
6-4	应用——写一个 WWW 服务器.....	248
6-4-1	导读.....	248
6-4-2	明确目的.....	248
6-4-3	整体结构分析.....	248
6-4-4	动手过程.....	249
6-4-5	动脑时间.....	275
6-5	完善——网络编程容错处理.....	276
6-6	小结.....	277
第 7 章	网络编程与网络安全.....	279
7-1	网络安全与网络编程的关系.....	280
7-2	起步——一些密码学基本概念.....	280
7-3	提高——了解目录服务.....	281
7-4	进一步——了解一下 X.509 和 SSL.....	284
7-5	回顾——综合运用所学到的知识.....	290

7-6 应用——写个实用的 HTTP 代理.....	290
7-6-1 导读	290
7-6-2 明确目的和思路	290
7-6-3 整体结构分析	290
7-6-4 动手过程	291
7-6-5 不足之处	309
7-6-6 动脑时间	310
7-7 完善——给代理加上保密功能.....	313
7-7-1 导读	313
7-7-2 明确目的	313
7-7-3 ISAPI 过滤器.....	314
7-7-4 动手过程——让 Delphi 与 VC 合作	315
7-7-5 不足之处	326
7-7-6 动脑时间	326
7-8 小结	326
附录 A 参考资料	327
附录 B 配套光盘内容介绍	329

第1章

引言与概述

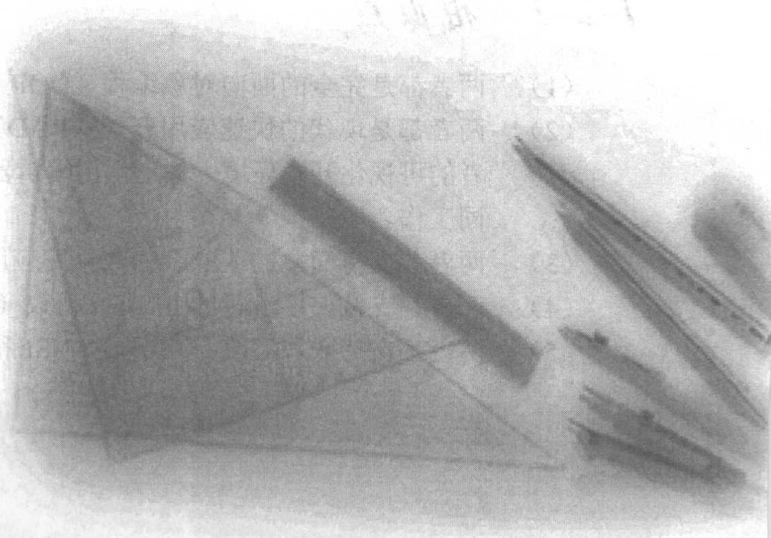
本章内容:

网络编程需求

比较 Delphi 与 Visual C++

帮你快速进入状态

Delphi 网络编程的几种方式



1-1 网络编程需求

网络是一个奇迹，它将我们带入一个丰富多彩的新世界，使我们能更方便，快捷的沟通与交流。随着计算机网络技术的飞速发展，越来越多的人意识到网络编程的重要性，在诸如电子商务等许多实际应用中，都要求程序员通过网络编程为用户提供可靠、高效、丰富的信息内容。即使你是一名数据库程序员，你也应该对网络编程有较为详尽的了解。

实际的网络编程是具有平台性的。因为各种操作系统下有关的应用程序编程调用接口是不尽相同的，而这一层次是我们编写大多网络应用程序的基础。如 UNIX 下网络编程的基础是 Berkeley 套接口，而 Windows 下则主要是 WinSock API。不过，网络编程所需的基础知识和概念在很大程度上并不依赖于操作系统。

那么，什么是网络编程呢？W.Richard.Stevens 在他所著的《UNIX 网络编程》一书中对此是这样解释的：“网络编程，即是编写通过计算机网络与其他程序进行通信的这类程序”。

我们这里讨论的网络编程是针对 Windows 平台的。Windows 操作系统以它亲切友好的界面和灵活方便的操作方式在国内赢得了越来越多的用户，尽管在许多程序员眼中，它的安全性和稳定性有待提高，但不可否认的是：中国的广大用户需要它。

“存在的就是有道理的”，因此，开发 Windows 平台下的网络程序就成为一种需求，而掌握 Windows 平台下的网络编程技术也自然成为程序员的需要。

1-2 Delphi 与 Visual C++

在 Windows 平台下，有着各种各样的开发工具，如 Microsoft 公司的 Visual C++、Visual Basic 及 Borland 公司的 Delphi、C++ Builder 等等，同时还有各种 Java 语言工具。

其中，Visual C++ 和 Delphi 因为功能强大，性能稳定，界面友好受到广大程序员的青睐，各有各的支持者，在市场上竞争也很激烈。有一句为程序员所熟知的形象地说明了这一点：“真正的程序员用 C，聪明的程序员用 Delphi”。这也使许多初学者在选择开发工具时拿不定主意：我到底该做一个聪明的程序员呢，还是做一个真正的程序员呢？

其实，对一个“真正的程序员”的概念，我有着自己的观点——这并不决定于他使用哪种开发工具，有兴趣的读者可以翻回前言看看。在这里，我只想从不同方面，不同角度客观地比较一下这两种开发工具，给大家以参考。

1-2-1 相似点

- (1) 两者都是完全的面向对象编程 (OOP) 工具。
- (2) 两者都是现代的快速应用开发 (RAD) 工具，并都具有可视化的开发环境。两者的可视化开发环境大体上都由编辑器，调试器和窗体设计器三部分组成，并协同工作。
- (3) 两者都支持简单的或复杂的数据库应用。
- (4) 两者都支持许多流行的 Internet 协议（通过组件方式或 MFC 类库）。
- (5) 两者都支持嵌入汇编，多线程，ANSI 和 Unicode 字符串，支持变体 (Variant) 类型。

1-2-2 不同点

- (1) Delphi 使用的是 Object Pascal 语言。Pascal 这种大学中的教学语言，历来以结构清晰，语法严谨而著称。而 Object Pascal 功能更为强大，应用更为灵活。Visual C++ 的语言基础是由 C 语言扩展而来的 C++，加之微软又为之写了功能强大的基础类库（MFC），使之如虎添翼。
- (2) 两者虽然都为面向对象编程工具，但构造程序的基础不同。Delphi 用自己的 VCL，而 Visual C++ 用 MFC。

1-2-3 Visual C++ 的优势

Visual C++ 自从“出道”以来，深受广大程序员的喜爱，它的优势集中体现在如下几个方面。

- (1) 它是微软的产品。这是很大的优势：因为微软控制着 Windows 操作系统，它的开发工具必然和 Windows 结合得更为紧密、深入，而且操作系统的最新技术也会迅速地反应到开发工具中去。这点 Borland 公司可总是要慢上半拍的。
- (2) Visual C++ 有着强大的技术支持。各种软件开发工具包（SDK），设备驱动程序开发包（DDK），MSDN 里厚厚的参考资料和丰富的例子，能为学习和编程提供详细的参考。Borland 公司的帮助系统和支持虽然也很强大，但和微软比较而言，还是略逊一筹。
- (3) 这一优势应该算在 C (C++) 语言上。C (C++) 有着广泛的、良好的应用基础：UNIX 是用 C 语言写的，UNIX 和 Linux 下的大多数程序也是用 C (C++) 开发的，所以 C (C++) 的用户和支持者很多，构成了 Visual C++ 庞大的用户群。特别是在涉及到程序移植时（如从 UNIX 下移植到 Windows 中），将 C (C++) 源程序用 Visual C++ 来改写，比起改写到符合 Object Pascal 语法的 Delphi 下要方便多了。很可惜，Borland C++ 风行一时的时代一去不复了。Borland 公司失去了推出 Borland C++ Builder 的好机会。
- (4) Visual C++ 的适应性很广，从高层的数据库应用到底层的设备驱动程序编写，它都有一手。但如果用 Delphi 写设备驱动的话——那你必须是个高手，并且同时熟悉 C (C++) 语言，因为你要看的好多例子都是用 C (C++) 或者 Visual C++ 写的。同时不容忽视的一点是，Visual C++ 继承了 C (C++) 语言运行效率高的优点，特别是在针对操作系统进行了特殊的编译优化时，这点表现的更为明显。
- (5) Visual C++ 的应用工具十分丰富。比如 Visual Studio 6.0 中常规安装就附带了二十几个小工具，其中 Windiff、Spy++、Depends 等都是十分有用的工具，给编程带来了很大的方便。

1-2-4 Delphi 的优势

说了许多 Visual C++ 的闪光点，那么 Delphi 有什么引以为荣的东西呢？

- (1) Delphi 有着针对 Windows 平台的最快的高级语言本地代码编译器。

Pascal 编译器最著名的特点就是快，而 Delphi 正是建立在这种编译器的基础之上的。虽然 C++ 编译器在近年来取得了很大的进步（如采用缓存策略，增加链接等方法），但其速度还是比 Delphi 的编译器慢了几倍。别小看这一点，它可以使你逐步递进地开发软件，经常地修改源代码，重新编译、测试，再修改、编译、测试……形成一个良好的开发循环。比较而言，如果编译速度很慢，开发者就不得不分批地修改代码，每次编译前进行多处修改以期更快地完成工作。同时 Delphi 和 C++ Builder 共享同一种编译器后端，它生成的可执行代码效率也很高。

- (2) Delphi 的 IDE 中集成的窗体设计器使用方便，功能强大。

对于 RAD 工具来说，窗体设计器是一项独特的功能，而且它对开发工具的整体效率影响巨大。严格的说，Visual C++ 并没有将窗体设计器集成到开发流程中，所以，你不得不分别对各种资源进行编辑——对话框、菜单、工具条、图标等等，并且要在复杂的程序结构中添加并不很容易理解的代码，将其组合、集成到一起并显示出来。而在 Delphi 中，你只需要拖动和点击几下鼠标就可以了。同时必须强调的是，Delphi 和 Visual Basic 也是有所不同的——不单单是执行代码的效率上相差甚远。Delphi 是严格意义上的面向对象编程语言，而 Visual Basic 则不是，它所设计的窗体是没办法被继承，派生的。

- (3) Delphi 很好地把握了语言复杂性和功能性的平衡。

C++ 是一种极为有力的语言，它简洁、直观、灵活而且功能强大，但并不是每个开发者都会正确、严谨地利用这些功能。C++ 的灵活性也常常“引诱”着程序员朝着一种不好的设计方向前进——写出糟糕的 C++ 代码是最容易不过的事了。Delphi 在某些方面比 C++ 要“死板”得多，比如对声明变量的位置要求十分严格，不支持美观但危险的操作符重载等等。实际上，它是通过限制自身可用功能来强化开发者的逻辑设计思维。如果你比较过 Delphi 和 Visual C++ 编写出的程序，你就会发现，Delphi 代码在程序模块化设计和逻辑构成上要比 Visual C++ 生成的代码更为清晰、严谨。同时要指出的是，Visual C++ 由向导生成框架的程序代码，冗余是很大的。

- (4) Delphi 的 VCL 要比微软的 MFC 更容易使用。

VCL 是 Delphi 的重要组成部分，而 MFC 是 Visual C++ 庞大架构的基础，在这里，我并不想比较两者的功能。但是，VCL 会使得你更容易在设计时期所见即所得地搭建程序框架，并方便地设定组件属性，操纵组件行为。相比之下，如果要熟练、细致地操纵 MFC，则需要你有丰富的有关 MFC 内部结构和逻辑关系的相关知识，并且，如果没有 RAD 工具支持的话，其功能还会受到一定的限制。也许《深入浅出 MFC》一书会对你有所帮助。

不要被 Visual C++ 的 ActiveX 组件所迷惑，它的确具有和 VCL 组件相同的设计时期性能，但却不能被继承以创建一个具有其他不同行为的新类。

- (5) Delphi 在数据库支持方面具有明显的优势。

Borland 公司缺少自己的有影响力的数据库产品，目前，面对巨大的竞争压力，它甚至公开了其 InterBase 数据库的源代码。但也正因为如此，Delphi 保留了我

认为是所有工具中最具灵活性的数据库支持。它的 BDE 引擎功能强大，可以支持本地数据库，客户/服务器形式和 ODBC 方式的数据库平台。你也可以使用 ADO、第三方工具或者自己创建数据库访问类来进行数据库程序的编写。此外，Delphi 的 MIDAS 使对数据源的多层访问更易于实现，它所支持的 CORBA 功能也是十分强大的。Delphi 是我心中最理想的数据库前端开发工具，相比较而言，Microsoft 的工具，如 ODBC、OLE DB 等等，从逻辑上来说，更趋向于支持它自己的数据库访问解决方案。

- (6) Delphi 的网络编程开发工具功能强大，有很大的潜力可挖。

我不知道为什么，市面上论述 Delphi 下网络编程的书很少，有些参考资料即使提到一些，却也不够详细和深入。是因为前期的 Delphi 版本对网络支持很少使大家忽略了它，还是大家都已经习惯于用 C++ 来写网络相关的程序？其实 Delphi 5 自带了 30 多个网络编程的组件，而 Delphi 6 中，单单是 Indy 网络组件单元就包括了 60 多个强大的网络组件。同时，Delphi 还封装了 WinSock API 等底层网络函数，其在网络编程方面功能之强大，潜力之巨大，毫不逊色于任何一种开发工具。如果大家深入了解了这方面的知识并掌握了一定技能之后，将其与 Delphi 其他的优势相结合，我想，你会很快捷、方便地建立适合自己的、功能丰富的网络相关应用程序的。这也是我决心写这本书的目的之一。

1-2-5 如何取长补短

从上面的论述、分析中，读者也许已经了解了 Delphi 和 Visual C++ 各自的优势。那如何在实际编程应用中取长补短，快速地开发出高效、稳定、维护容易、扩展灵活的程序呢？在此，我给出以下建议：

- (1) 如果你的程序效率要求很高（比如写一些加密的软件算法等），并且不涉及复杂的界面构造，请选择 C (C++)，其实这种情况下，汇编更合适一些；
- (2) 如果你要对系统底层的硬件进行操作（比如写设备驱动程序等），请选择 Visual C++；
- (3) 如果你想使你的程序有着漂亮的界面，并力争避免书写大量繁琐复杂的代码，请选择 Delphi；
- (4) 如果你想写功能强大，方式灵活的数据库前端，请选择 Delphi；
- (5) 如果你不愿用方便，快速地通过点击鼠标和拖放组件的方式编程，而愿意小心修改用向导生成的含大量冗余信息的代码，请选择 Visual C++；
- (6) 如果你时间紧，任务重，要求开发周期短的话，请用 Delphi；

其实，我最支持的，也是强烈向大家推荐的编程方式是下面这种：

用 Visual C++(C、C++)编写与界面和数据库操作无关的底层函数，以求得更高的执行效率和更好的移植性，将其统一编译成动态链接库，在 Delphi 中进行调用。而 Delphi 则用来生成应用界面、数据库操作前端、CGI/ISAPI/NSAPI 等高层应用程序。

也许有的读者会笑。“你还是没有告诉我该选择哪种语言工具呢”？

笔者认为，一个好的程序员至少应该熟练掌握两种以上的编程工具，以便能互相促进，

取长补短。在此，笔者推荐你学习 Delphi 并深入理解 C++。请注意，我说的是 C++，而不一定是 Visual C++。当然，为了学好 C++，你可能要付出很大的努力。

1-3 如何快速进入状态

1-3-1 不要误解标题

千万不要误解本节的标题。“快速进入状态”并不意味着“有捷径可以使你快速掌握 Delphi 网络编程”。我只是努力为你顺利地阅读以后的章节做一些铺垫和提示——但是是否能掌握还要看你的理解能力。

实际编程过程中，是没有“捷径”可走的。你可以找到一本好的参考书，一个贴切易懂的好例子，但你永远都不会找到“捷径”。有的只是不断的学习、探索和积累。只有“厚积”才能“薄发”。

1-3-2 要做的准备工作

请不要为没有通向成功的“捷径”感到懊恼——如果你按下面的要求做了一定的准备工作，我相信，我们的进展一定比你想象的要顺利。

(1) 理解 Object Pascal 语言

请注意，我说的是“理解”，而不是“熟练掌握”。

Object Pascal 语言是 Delphi 的基础。即使你有使用 Pascal 的经验，你仍将发现，深入理解 Object Pascal 是非常必要的。

为了能为阅读下面的章节打下基础，你需要重点理解如下一些内容：

- (a) Delphi 中包的概念和使用；
- (b) 面向对象编程思想在 Delphi 中的具体体现；
- (c) Delphi 的结构化异常处理；
- (d) Delphi 常用的运行时库 (RTL, run time library) 和 VCL。

(2) 理解网络的一些基本概念

在我的思维里，网络是一个综合技术的产物，它涉及的技术范围无论从深度还是广度上讲，都是前所未有的。从硬件到软件，从通讯到安全，从协议制定到应用实现……庆幸的是，我们只需要综合、全面地理解它，而不需要面面俱到地掌握它。如果实际工作中，我们需要掌握某方面的知识了，那我们只要知道在什么地方能找到它就可以了。

你需要了解的基础知识有：

- (a) 计算机网络的定义；网络的拓扑结构和体系结构；需要着重理解一下 ISO/OSI 网络分层模型；
- (b) TCP/IP 协议族的产生及其重要性；各自的功能及简要原理；
- (c) 了解常用的应用协议，如 HTTP, SMTP, FTP 等等，并能直观地将它们与在现实 Internet 世界中的具体应用挂起钩来；最好能简单了解一下这些协议