

网络程序设计系列丛书

# Visual C++ .NET

## 网络编程 与互联网应用开发

文娟 萧秋水 编著



清华大学出版社  
<http://www.tup.tsinghua.edu.cn>



网络程序设计系列丛书

# Visual C++.NET 网络编程 与互联网应用开发

文 娟 萧秋水 编著

清华大学出版社

## (京)新登字 158 号

### 内 容 简 介

本书以微软推出的面向其 .NET 平台的集成开发环境 Visual Studio.NET 为开发工具,以其中最新版的 Visual C++.NET 为开发语言,介绍两个层次上的网络编程问题,即网络协议编程和面向企业集成的互联网应用开发。

本书分为三篇,分别讲解微软 .NET 战略、网络协议的 Win32 编程技术和 .NET 互联网应用技术。各篇及篇内各章之间循序渐进而又自成体系,读者可以顺序阅读,或者直接查阅感兴趣的章节。除了清晰地给出使用 Visual C++.NET 的各方面技术要点和理论知识之外,本书还针对各技术要点给出了丰富的编程实例,使读者针对某一项目可以依照实例中的步骤立即开始动手编程。

本书适用于使用 Visual C++.NET 进行网络协议编程和互联网应用开发的中、高级程序员。当然,初级读者也可以从实例开始入门并受益。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

### 图书在版编目(CIP)数据

Visual C++.NET 网络编程与互联网应用开发/文娟,萧秋水编著. --北京:清华大学出版社, 2002.12

(网络程序设计系列丛书)

ISBN 7-302-05329-4

I. V... II. ①文... ②萧... III. 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 085493 号

出 版 者: 清华大学出版社 (北京清华大学学研大厦, 邮编: 100084)

<http://www.tup.tsinghua.edu.cn>

<http://www.tup.com.cn>

责任编辑: 龙啟铭

印 刷 者: 北京密云胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 26 字数: 646 千字

版 次: 2002 年 12 月第 1 版 2002 年 12 月第 1 次印刷

书 号: ISBN 7-302-05329-4/TP·3129

印 数: 0001~5000

定 价: 36.00 元

# 前 言

未来是以网络为中心的世界。面对这个已经或即将来临的世界，每个 IT 巨人都在思考。.NET 是微软为 Web Service (Web 服务)——以统一的个性化的方式联系信息世界、设备和人的下一代互联网软件——提供的平台，它表明微软将以网络为中心，彻底转换产品研发、发布的方式，改变产品和服务的范围。.NET 是一项革命性的技术框架。.NET 的核心技术包括分布式计算、XML、组件技术、即时编译技术等。

对于开发人员来说，.NET 带来的冲击是十分巨大的，如何把现有的应用程序扩展到网络尤其是 Internet 上，如何为 Internet 开发出企业级的、高性能的以及可伸缩的应用程序，这些问题都可以在 .NET 及微软为其提供的集成开发环境 Visual Studio.NET 中找到答案。

本书以帮助开发人员转向 .NET 架构为目标，以 Visual Studio.NET 为开发工具，以其中最新版的 Visual C++ 为开发语言，分三篇介绍两个层次上的网络编程问题，这三篇包括微软 .NET 战略、网络协议的 Win32 编程技术和 .NET 互联网应用开发，而两个层次上的网络编程问题包括网络协议编程层次和面向企业集成的互联网应用层次，后者是构建在前者的基础之上的，而前者可以构建的应用程序决不仅仅囿于后者的范围之内，它可以说是现在整个 Internet 的基础设施，通过这个层次的编程我们可以实现与 .NET 不完全一样的体系结构，也可以实现异构平台之间的通信，如 Windows 与 Linux。

本书的各篇及篇内各章之间循序渐进而又自成体系，读者可以顺序阅读，或者直接查阅感兴趣的章节。除了清晰地给出使用 Visual C++ .NET 的各方面技术要点和理论知识之外，本书还针对各技术要点给出了丰富的编程实例，使读者针对某一项目可以依照实例中的步骤立即动手编程。而且读者可以利用本书原来的代码为基础，不用一切都从头开始。

下面对本书的内容做一个简单概述，以方便读者的查找。

“第一篇 微软 .NET 战略”从总体上介绍了微软 .NET 平台的整体思想和框架及为 .NET 设计的 Visual Studio.NET 的集成开发环境的使用，它包括第 1 章和第 2 章。

“第 1 章 微软 .NET 体系结构”解释了 .NET 到底是什么，它的体系结构和组成包括哪些部分，以及其核心部件 .NET 框架 (.NET Framework) 的相关概念和技术，并指出了 .NET 体系结构的重大意义。

“第 2 章 Visual Studio.NET 集成开发环境”介绍了微软为 .NET 架构设计的优秀开发工具 Visual Studio.NET，主要包括对其开发环境各个组成要素的介绍，并探讨了其中我们将要使用的编程语言——Visual C++ .NET 的沿袭与演变。

“第二篇 Win32 网络编程技术”从网络协议编程的层次分 8 章讲解了使用 Visual C++ .NET 和 Win32 编程技术进行网络编程的各方面细节。这一篇的内容十分丰富，包括从第 3 章~第 10 章的所有内容。

“第 3 章 计算机网络与协议”为后面的章节做铺垫，引入了许多关于计算机网络的基础理论知识，包括计算机网络的拓扑结构、网络通信的标准及标准化组织、网络的开放

系统互连 (OSI) 模型以及由此引入的计算机网络的重要协议。

“第 4 章 WinSock 网络编程”介绍了一种历史极为悠久也非常重要的网络编程接口，那就是 Socket 接口，它在 Win32 网络编程技术中的实现就是这一章具体介绍的 WinSock。本章通过两个实例（网络聊天程序和获取主机名称及 IP 的程序）讲解并演示了 WinSock 的基本通信机理及其数据库例程的使用方法。

“第 5 章 NetBIOS 与 WNet API 网络编程”介绍了 WinSock 之外 Win32 下进行网络编程的另外两种网络接口，即 NetBIOS 和 WNet API 接口，并通过两个实例程序（获取网络物理地址和枚举网上邻居）示范了这两种网络编程接口的使用技巧。

“第 6 章 ICMP 与 TELNET 协议编程”引出了重要的 Internet 控制报文协议 ICMP 和远程登录协议 TELNET，诠释了协议的目的、种类和应用，并给出了使用这两种协议进行开发的基本原理，手把手地引导读者创建一个 PingPlus 实用工具。

“第 7 章 电子邮件与网络新闻”大致分为三个部分。第一部分给出了电子邮件收发程序的基本编程机理——SMTP 和 POP3 协议的细节及具体的实现方法，其中给出的实例源码很有参考价值。第二部分介绍了 Windows 下一个简单的电子邮件编程接口，即 MAPI，使用它，只需很少量的代码即可完成邮件的收发。第三部分讲解了网络新闻传输协议并实现了一个客户端。

“第 8 章 拨号上网”为读者讲解了 RAS（远程访问服务）客户机的编程原理，它是拨号上网的各种编程方式中最为简单也是最为有效的一种方法，是拨号编程必须掌握的一项技能。除了 RAS，在该章的最后还进一步提及了更深一层次的电话 API (TAPI) 的一些基本原理。

“第 9 章 FTP 协议编程”阐述了 Internet 上文件交换的原理和协议标准，并为 Internet 的文件传输协议实现了一个客户端软件，使用这个工具，你可以从 Internet 的 FTP 服务器上自由地下载想要的文件。

“第 10 章 HTTP 与 Finger 协议编程”借助一个简单版本的 Internet Explorer（即 Windows 自带的浏览器）演示程序的编写，引导读者进入 HTTP 和 Web 的世界。事实上，WWW 已经成为当前 Internet 发展最快的一个热点。而使用 Finger 协议则可以获取特定服务器上的用户信息。

“第三篇 .NET 互联网应用开发”立足于面向企业集成的互联网应用层次，向读者展示了 Visual C++.NET 的崭新应用领域，它从典型的互联网多层应用程序的结构入手，分 3 章介绍了 .NET 分布式应用的典型构建方法。

“第 11 章 使用组件技术封装业务规则”从多层应用程序的中间业务层入手，引入了微软的组件技术 COM+ 的各种细节及其在 .NET 中的应用，并给出了一个使用 COM+ 组件封装业务规则的实例。

“第 12 章 .NET 的数据访问”介绍了 .NET 分布式多层应用中的另一个重要领域，即分布式的数据访问技术，重点讲解了 .NET 中的重要组成部分——ADO.NET，它与 XML 一起构成了 .NET 平台中数据访问技术的支柱。

“第 13 章 构建 Web Service”诠释了 .NET 平台中的一项核心应用技术，即 Web Service（Web 服务）。这一章为读者解释了到底什么是 Web Service，Web Service 的协议栈及其工

作流程，最后给出了使用 Visual C++.NET 构建并发布一个 Web Service 的完整实例。

本书适用于使用 Visual C++.NET 进行网络协议编程和互联网应用开发的中高级程序员。当然，初级读者也可以从实例开始入门并受益。

本书是集体智慧的结晶，主要由文娟、萧秋水编著，蒋永波、刘鑫、刘政伟、唐明参与了书稿的编写工作，袁昌、宋志刚、李必刚、张程等为本书编写了部分实例代码。由于作者水平及视野所限，难免在内容选材和叙述上有不当之处。欢迎广大读者对本书提出批评和建议。

联系方式：longqm@tup.tsinghua.edu.cn。

编者

2002.10

# 目 录

## 第一篇 微软.NET 战略



第 1 章 微软.NET 体系结构.....	1
1.1 .NET 概述.....	1
1.1.1 .NET 是什么.....	1
1.1.2 .NET 的组成部分.....	3
1.1.3 .NET 的价值.....	5
1.2 .NET 框架.....	6
1.2.1 .NET 框架的组成.....	6
1.2.2 公共语言运行库 (Common Language Runtime) .....	7
1.2.3 .NET 框架类库.....	9
1.2.4 .NET 框架的优点.....	11
第 2 章 Visual Studio.NET 集成开发环境.....	13
2.1 Visual Studio.NET 安装简介.....	13
2.2 Visual Studio.NET 集成开发环境概述.....	14
2.2.1 起始页.....	15
2.2.2 解决方案资源管理器.....	15
2.2.3 增强的工具箱.....	16
2.2.4 服务器资源管理器.....	16
2.2.5 任务列表.....	17
2.2.6 动态帮助.....	18
2.2.7 增强的智能感知.....	18
2.2.8 Visual Studio.NET 中的数据访问.....	18
2.3 Visual C++.NET 的沿袭与演变.....	21
2.3.1 Visual Studio.NET 中的标准 C++.....	21
2.3.2 Visual Studio.NET 中的受管 C++.....	23

## 第二篇 Win 32 网络编程技术

第 3 章 计算机网络与协议.....	25
3.1 计算机网络的拓扑结构.....	25
3.1.1 局域网基本拓扑结构.....	26
3.1.2 局域网混合拓扑结构.....	28

3.1.3 广域网拓扑结构 .....	31
3.2 标准和标准化组织 .....	34
3.2.1 对标准的需要 .....	34
3.2.2 制订标准的机构 .....	34
3.3 开放系统和开放系统互连 (OSI) 模型 .....	36
3.3.1 物理层 .....	38
3.3.2 数据链路层 .....	38
3.3.3 网络层 .....	39
3.3.4 传输层 .....	39
3.3.5 会话层 .....	39
3.3.6 表示层 .....	40
3.3.7 应用层 .....	40
3.3.8 应用 OSI 模型 .....	40
3.4 网络协议 .....	42
3.4.1 协议介绍 .....	42
3.4.2 TCP/IP .....	43
3.4.3 NetBIOS 和 NetBEUI .....	48
3.4.4 IPX/SPX .....	49
<b>第 4 章 WinSock 网络编程 .....</b>	<b>51</b>
4.1 WinSock 简介 .....	51
4.2 WinSock API 主要函数的使用 .....	51
4.3 MFC 对 WinSock API 的封装 .....	58
4.4 WinSock 控件 .....	58
4.4.1 WinSock 控件的主要属性 .....	59
4.4.2 WinSock 控件的主要方法 .....	60
4.4.3 控件的主要事件 .....	61
4.5 实例一 网上聊天 .....	61
4.5.1 聊天服务器的实现 .....	62
4.5.2 聊天客户端的实现 .....	74
4.6 WinSock 的数据库例程 .....	81
4.7 实例二 获取本地计算机的主机名和 IP 地址 .....	84
4.7.1 建立一个新项目 .....	84
4.7.2 CIPEnum 类及其实现 .....	84
4.7.3 CMyIPEnum 类及其实现 .....	88
4.7.4 程序的主函数 .....	89
<b>第 5 章 NetBIOS 与 Wnet API 网络编程 .....</b>	<b>90</b>
5.1 NetBIOS 网络编程接口 .....	90
5.2 NetBIOS 的基本概念 .....	90



---

5.3	NetBIOS 编程.....	92
5.4	使用 NetBIOS 获取网络参数.....	98
5.5	实例一 获取网卡物理地址.....	100
5.5.1	创建一个新项目.....	101
5.5.2	枚举可用的 MAC 地址.....	101
5.6	WNet API.....	102
5.7	WNet API 中的函数.....	104
5.8	实例二实现网上邻居枚举工具.....	109
5.8.1	建立一个新项目.....	109
5.8.2	添加 CNetwork 类对 WNet API 进行包装.....	109
5.8.3	创建模板类 CNetSearch.....	117
5.8.4	在对话框中完成枚举任务.....	118
<b>第 6 章</b>	<b>ICMP 与 Telnet 协议编程.....</b>	<b>120</b>
6.1	Internet 控制报文协议 ICMP.....	120
6.1.1	ICMP 报文格式.....	121
6.1.2	ICMP 报文的分类.....	121
6.2	实例一 PingPlus 实用工具.....	125
6.2.1	建立一个新项目.....	127
6.2.2	添加并实现 Cicmp 类.....	127
6.2.3	添加并实现 RegKey 类.....	136
6.2.4	添加并实现 SysTCPIP 类.....	139
6.2.5	实现具体的用户操作过程.....	142
6.3	Telnet (远程登录).....	147
6.4	NT 服务程序编程原理.....	149
6.5	实例二 Telnet 服务器.....	154
6.5.1	建立一个新项目.....	154
6.5.2	添加 CNTService 类包装 NT 服务器程序.....	154
6.5.3	添加 SMRemoteService 类实现 Telnet 服务程序.....	163
6.5.4	服务程序的启动细节.....	168
6.6	实例三 Telnet 客户机.....	168
6.6.1	建立一个新项目.....	169
6.6.2	实现服务器登录.....	169
6.6.3	与服务器远程交互.....	170
<b>第 7 章</b>	<b>电子邮件与网络新闻.....</b>	<b>174</b>
7.1	电子邮件.....	174
7.2	SMTP 协议简介.....	174
7.2.1	SMTP 协议扩展.....	176
7.2.2	多用途网际邮件扩充.....	176

7.3	POP3 协议简介 .....	178
7.4	任务条通知区的编程原理.....	180
7.5	实例一 SMTP 邮件发送程序 .....	182
7.5.1	建立一个新项目 .....	183
7.5.2	程序的整体结构 .....	183
7.5.3	CSMTP 类对 SMTP 协议的包装 .....	184
7.5.4	邮件的包装类 .....	190
7.5.5	MIME 邮件格式.....	194
7.5.6	编码与解码 .....	197
7.5.7	在界面上为邮件添加附件.....	197
7.5.8	发送邮件的过程 .....	199
7.6	实例二 POP3 邮件接收程序 .....	200
7.6.1	建立一个新项目 .....	200
7.6.2	CPop 类对 POP3 协议的实现.....	201
7.6.3	使用 CTrayIcon 管理任务条通知区.....	208
7.6.4	在程序中监测服务器上的邮件.....	212
7.7	MAPI 编程与消息应用程序.....	214
7.7.1	MAPI 简介 .....	215
7.7.2	扩展 MAPI .....	217
7.7.3	简单 MAPI 与 CMC 的差别.....	217
7.7.4	消息应用程序 .....	219
7.7.5	MAPI 控件 .....	219
7.8	实例三 包装 MAPI 进行邮件发送.....	222
7.8.1	CMapiMessage 的定义 .....	222
7.8.2	CMapiSession 的定义 .....	223
7.9	网络新闻组 .....	224
7.9.1	Usenet 新闻组 .....	224
7.9.2	网络新闻传输协议.....	225
7.10	实例四 网络新闻客户端.....	226
7.10.1	建立一个新项目 .....	228
7.10.2	实现主窗口的分割.....	228
7.10.3	设定参数 .....	229
7.10.4	登录到 NNTP 服务器并获取新闻组列表.....	230
7.10.5	检索新闻组的文章.....	233
7.10.6	读取特定的文章 .....	236
7.10.7	投递自己的文章 .....	237
<b>第 8 章</b>	<b>拨号上网.....</b>	<b>241</b>
8.1	RAS 简介.....	241

8.2	拨号与挂断 .....	242
8.3	连接管理 .....	252
8.4	电话簿管理 .....	255
8.5	实例 · RAS 拨号客户程序 .....	265
8.5.1	建立一个新项目 .....	266
8.5.2	枚举计算机上已有的拨号连接 .....	266
8.5.3	异步地启动拨号连接 .....	268
8.5.4	处理拨号事件 .....	269
8.5.5	终止拨号 .....	273
8.6	TAPI 提供的服务 .....	273
8.6.1	TAPI 的分级 .....	274
8.6.2	通信过程描述 .....	275
<b>第 9 章</b>	<b>FTP 协议编程 .....</b>	<b>279</b>
9.1	FTP 协议及其实现方法 .....	279
9.1.1	使用 Microsoft Internet Transfer 控件 .....	281
9.1.2	直接使用 WinInet API .....	285
9.1.3	使用 MFC WinInet 类 .....	290
9.2	实例一 FTP 客户程序 .....	303
9.2.1	建立一个新项目 .....	303
9.2.2	添加代码连接到 FTP 服务器/断开连接 .....	304
9.2.3	添加代码定位/读写远程文件 .....	307
<b>第 10 章</b>	<b>HTTP 与 Finger 协议编程 .....</b>	<b>312</b>
10.1	WWW 与 HTTP 协议 .....	312
10.1.1	统一资源定位符 .....	313
10.1.2	超文本传送协议 .....	316
10.1.3	使用 Microsoft WebBrowser 控件 .....	319
10.1.4	使用 CHtmlView 类 .....	322
10.2	实例一 一个简单的 WWW 浏览器 .....	323
10.2.1	建立一个新项目 .....	323
10.2.2	在项目中添加 WebBrowser 控件 .....	324
10.2.3	在程序窗口创建时创建 WebBrowser 控件 .....	324
10.2.4	完成程序的浏览任务 .....	324
10.3	Finger 协议 .....	325
10.4	实例二 实现 Finger 客户机 .....	327
10.4.1	建立一个新项目 .....	327
10.4.2	添加并实现 CFingerSocket 类 .....	328
10.4.3	添加并实现 CFingerThread 类 .....	332
10.4.4	响应 Finger 按钮 .....	336

## 第三篇 .NET 互联网应用开发

第 11 章 使用组件技术封装业务规则 .....	338
11.1 理解 COM+ 机理 .....	338
11.1.1 COM 回顾 .....	338
11.1.2 COM+ 的引入 .....	340
11.1.3 COM+ 基本结构 .....	340
11.1.4 COM+ 组件的对象环境 .....	342
11.1.5 COM+ 系统服务介绍 .....	344
11.1.6 COM+ 应用开发 .....	350
11.1.7 COM+ 小结 .....	350
11.1.8 .NET 框架与 COM+ .....	351
11.1.9 MSMQ (微软消息队列) .....	352
11.1.10 .NET 框架对 MSMQ 提供的支持 .....	352
11.2 实例一 COM+ 组件的实现 .....	356
11.2.1 创建 ATL 项目 .....	356
11.2.2 添加 ATL COM+ 组件 .....	357
11.2.3 为 COM+ 组件添加方法 .....	359
11.2.4 部署 COM+ 应用程序 .....	361
第 12 章 .NET 的数据访问 .....	365
12.1 ADO.NET 概述 .....	365
12.1.1 System.Data 命名空间 .....	366
12.1.2 OLE DB 简介 .....	366
12.1.3 ADO.NET 架构 .....	367
12.1.4 ADO.NET 的特征和优点 .....	369
12.1.5 ADO.NET 中的 XML .....	371
12.2 使用 Connection 对象连接数据源 .....	371
12.3 从数据源取回数据 .....	373
12.3.1 Command 对象简介 .....	373
12.3.2 DataReader 对象简介 .....	375
12.4 使用 DataSet 对象处理数据集 .....	376
12.4.1 数据集的基本概念 .....	376
12.4.2 处理数据集 (带有数据) 的一般步骤 .....	377
12.4.3 编程创建数据集 .....	377
12.4.4 给数据集添加一个数据表 .....	378
12.4.5 在两个表之间添加关系 .....	378
12.4.6 利用 DataRelation 从一个表导航至另一个表 .....	379

---

12.4.7	为数据集添加约束.....	380
12.4.8	处理数据集事件.....	380
12.4.9	处理带类型的数据集.....	381
12.5	数据集中的数据表操作.....	382
12.5.1	什么是数据表.....	382
12.5.2	创建数据表.....	382
12.5.3	给数据表添加列.....	383
12.5.4	表达式列.....	384
12.5.5	自动增量列.....	384
12.5.6	为表创建一个主关键字.....	385
12.5.7	给表添加数据.....	385
12.5.8	列状态.....	386
12.5.9	从表中删除或移除一个数据行.....	386
12.5.10	处理表的数据.....	387
12.5.11	添加和读取行错误信息.....	388
12.5.12	接受或拒绝数据表行的改变.....	389
12.5.13	处理 DataTable 事件.....	390
12.5.14	对表中的行进行过滤和排序.....	391
<b>第 13 章</b>	<b>构建 Web Service.....</b>	<b>392</b>
13.1	Web Service 概述.....	392
13.1.1	什么是 Web Service.....	393
13.1.2	Web Service 协议栈.....	393
13.1.3	Web Service 工作流程.....	395
13.2	实例一 构建 Web Service.....	396
13.2.1	新建 Web Service 项目.....	396
13.2.2	实现 Web Service.....	397
13.2.3	测试 Web Service.....	399
13.2.4	发布 Web Service.....	400

## 第 1 章 微软 .NET 体系结构

### 1.1 .NET 概述

#### 1.1.1 .NET 是什么

.NET 是微软的 XML Web 服务平台。微软 .NET 平台包括用于创建和操作新一代服务的基础结构和工具、建立和运行基于 XML 的软件所需要的全部部件、可以启用大量客户机的 .NET User Experience、用于建立新一代高度分布式的数以百万计的 .NET 积木式组件服务，以及用于启用新一代智能互联网设备的 .NET 设备软件。

微软 .NET 产品和服务包括 Windows.NET、建立积木式服务的核心集成套件、MSNTM.NET、个人订购服务、Office.NET、Visual Studio.NET，以及用于 .NET 的 bCentral™。

.NET 的整体架构如图 1-1 所示，它以 Web Services 为中心，将服务器、种类繁多的客户端、面向开发人员的开发工具、用户体验和企业级解决方案等有机地整合到一起，从而构建了一个崭新的、完整的、集成的分布式计算模式。

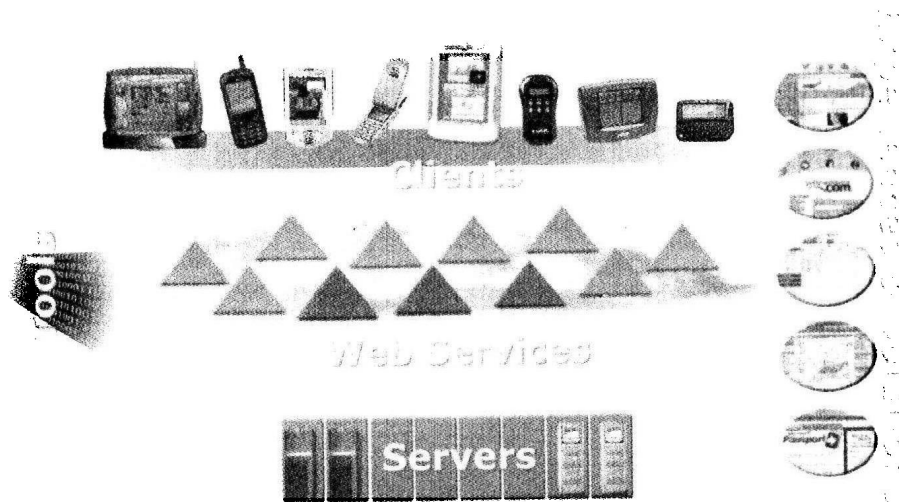


图 1-1 .NET 整体架构

微软 .NET 解决了以下一些当今软件开发中的核心问题。

- 互操作性 (Interoperability)、集成性 (Integration) 和应用程序的可扩展性 (extensibility) 太难实现而且代价很高。微软 .NET 依靠 XML (一个由 W3C 管

理的开放标准)消除了数据共享和软件集成的障碍。

- 无数具有相当竞争力的私有软件技术使得软件的集成变得非常复杂。而微软 .NET 建立在一个开放的标准上,它包含了所有编程语言。
- 当终端用户使用软件时,他们总觉得不够简便。有时甚至感到很沮丧,因为他们无法在程序之间方便地共享数据或是无法对能访问的数据进行操作。XML 使数据交换变得容易了,并且 .NET 软件可以使得用户只要一得到数据就能对它们进行操作。
- 终端用户在使用 Web 的时候,无法对自己的个人信息和数据进行控制,这导致了个人隐私和安全泄漏问题。而微软 .NET 提供了一套服务,使用户可以管理他们的个人信息,并且控制对这些信息的访问。
- .COM 公司和 Web 站点开发者们很难为用户提供足够的有价值的信息,至少有一部分原因是由于他们的应用程序和服务无法很好地和其他程序和服务合作,只是一个不和外界连接的信息孤岛。而微软 .NET 的设计宗旨就是为了使来自于多个站点和公司的数据或服务能够整合起来。

如同微软以前推出的 MS-DOS 和 Windows 一样, .NET 将大大改变我们的计算领域。MS-DOS 使得个人电脑在商业和家庭中广为接受; Windows 增强了用户的图形界面,使其成为首选的与软件交互方式,最终使得图形界面成为个人电脑的主流;而 .NET 则要把 XML Web 服务变成日后的主流计算模式。XML Web 服务是建立在 XML 数据交换基础上的软件模型,它帮助应用程序、服务和设备一起工作。用 XML 进行共享的数据,彼此之间独立,但同时又能够松散耦合地连接到一个执行某特定任务的合作组中。

整个 .NET 的结构非常复杂,但其核心则非常简单,主要包括以下内容:

- 软件即服务 (Software-As-Services)

软件即服务的思想是 .NET 研发的初衷,它是 .NET 的核心思想,一个典型的例子就是微软提供的 MSN 服务。

- 基于 XML 技术 (Base On XML Technology)

XML 是一个开放的工业标准,它被 .NET 采纳以作为整个体系结构中的数据交换的标准格式,在 .NET 中起着举足轻重的作用。

- 整合 Web 服务 (Integrate Web Services)

Web 服务的概念将“软件即服务”的思想做了进一步拓展,将因特网尤其是 Web 也整合到 .NET 大家庭中来。

.NET 这种集成的崭新的计算模式有着广泛的应用前景,可以被应用到很多重要经济领域中去,例如:

- 无线网络服务

提供无线上网手机、掌上电脑等服务。

- 网络服务集成

MSN 或一些不同行业网站服务的整合。

- 促进新设备的诞生和推广

一些很好的实例包括掌上电脑、智能电话、WebTV、书写板电脑 (Tablet PC) 等,如图 1-2 所示。



图 1-2 .NET 促进新设备的诞生和推广

总之，正如史蒂夫·鲍尔默所说，微软.NET 代表了一个集合、一个环境、一个可以作为平台支持下一代 Internet 的可编程结构。它的最终目的就是让用户在任何地方、任何时间，以及利用任何设备都能访问他们所需要的信息、文件和程序。

### 1.1.2 .NET 的组成部分

.NET 是一种崭新的计算模式，是一个完整的机体，它主要包括两部分：

- .NET 平台 一套编程工具和基本构架，用来创建、发布、管理和整合 Web 服务。
- .NET 体验 终端用户用于和.NET 交互的手段。

#### 1.1.2.1 .NET 平台

Microsoft 的.NET 平台是由用于创建和运行 XML Web 服务的组件组成的。它包含了下面四个组件：

##### (1) .NET 框架和 Visual Studio.NET

这些是开发人员用来生成 XML Web 服务的工具。.NET 框架是 Microsoft .NET 平台核心的一套编程接口；Visual Studio.NET 是一套多语言的编程工具。

##### (2) 服务器基础设施 (Server Infrastructure)

.NET 的服务器基础设施是一系列用于生成、发布和操作 XML Web 服务的基础程序，包括 Windows 和各种.NET 企业服务器。主要的技术包括对 XML、scale-out 及跨进程和服务的商务流程 (business process orchestration) 的支持。这些服务器包括有：

- Application Center 2000 用于 scale-out 解决方案。
- BizTalk Server 2000 用于创建和管理基于 XML 的跨进程和服务的商务流程 (business process orchestration across applications and services)。
- Host Integration Server 2000 用来访问主机上的数据和应用程序



- **Mobile Information 2001 Server** 使一些移动设备（比如移动电话）也能使用这些应用程序。
- **SQL Server 2000** 储存和检索结构化的 XML 数据。

### (3) 块构建服务 (Building Block Services)

块构建服务是一套以用户为中心的 XML Web 服务, 它把用户数据的控制权从应用程序移到了用户手上, 使 Web 有了一个翻天覆地的变化, 做到了程序、服务和设备之间的简单性及一致性, 这保证了所有的事务都必须得到用户的同意。这些服务包含了 **Passport** (用于用户身份验证)、服务之间的消息传递、文件存储、用户个性设置的管理、日历管理和其他一些功能。Microsoft 将在那些对 .NET 基础设施起至关重要作用的领域内提供一些块构建服务。大量的合作伙伴和开发商将对这些块构建服务作重要的扩展。

### (4) 智能设备 (smart device)

.NET 利用软件使智能设备, 诸如手提电脑、轻便电话、游戏操纵台等都能够 .NET 世界中得以使用。

一个典型的智能设备应该具有如下特征:

- **用户接口智能化** 能根据用户的 .NET 身份、档案和有关数据简化用户的工作; 另外要对用户的存在与否保持足够的智能, 能根据用户的在线与不在线对通知作出调整。
- **网络接口智能化** 负责带宽的限制; 支持应用程序的在线和离线两种使用模式; 知道有哪些有效的服务。
- **信息接口智能化** 能在任何地方、任何时间访问、分析和操作数据。
- **设备接口智能化** 能发现和报告其他智能设备、服务和 Internet 的存在; 知道如何为其他设备提供服务; 能够灵活方便地从 PC 上访问信息。
- **软件和服务接口智能化** 能根据表单的情况, 最恰当地表现应用和数据; 为终端用户提供合适的输入方法和连接; 用 XML、SOAP 和 UDDI 来使用 Web 服务; 对开发者来说, 要具有可编程性和扩展性。

Microsoft 的一些软件能够在智能设备上运行, 它们包括 Windows XP、Windows Me、Windows CE、嵌入式 Windows、.NET 框架以及 .NET Compact 框架。

## 1.1.2.2 .NET 体验 (.NET Experiences)

终端用户是通过 .NET 体验访问 XML Web 服务的, 这和现有的独立应用程序有点类似, 但在下列这些重要的方面是不同的:

- .NET 体验可使用于多种设备

我们无需为可能使用的每一个设备编写一个不同 XML Web 服务和不同的 .NET 体验, .NET 体验能够读取用户选取设备的特征, 给出一个恰当界面。

- .NET 体验使用 XML Web 服务

当 .NET 体验连入网络后就能有效地利用 XML Web 服务为用户带来额外的价值, 以更好地解决问题。

- .NET 体验是以用户为中心的

.NET 体验的焦点在终端用户, 使用基于身份验证的块构建服务来为用户验证、参数设