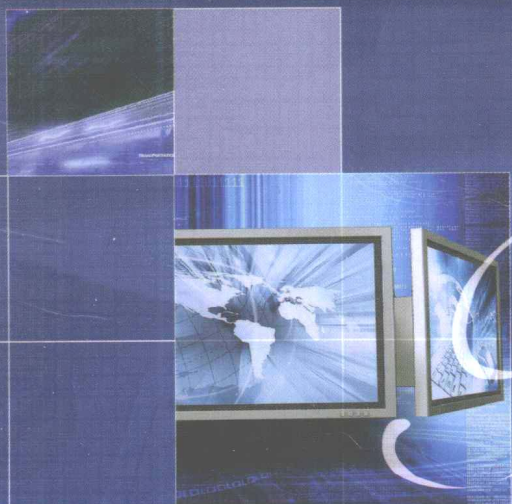


C#程序设计

胡艳菊 主 编
申 野 杜云奇 孙建伟 副主编



Programming

- 提供所有题目解决方案
- 案例化教学实践性演练
- C#程序员速成技术宝典



北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国应用型本科计算机案例型规划教材

C#程序设计

主 编 胡艳菊

副主编 申 野 杜云奇 孙建伟



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书共分 12 章, 包括 .NET 基本知识、C# 基础知识、C# 实现面向对象、C# 中的继承、C# 高级面向对象、数组和集合对象、C# 中的文件处理、WinForms 基础知识、ADO.NET、ASP.NET、Windows 应用程序项目开发案例和宠物网站的功能设计。本书由浅入深、结构完整、内容精要, 结合连续、深入的案例, 讲解了 .NET 框架、C# 语言语法、WinForms 程序设计和 ASP.NET 程序设计。通过突出实用知识点的学习和有效的学习过程设计, 以快速培养 C# 程序员。

本书既可作为高等学校相关课程和社会培训机构的教学用书, 也可作为有基本程序设计基础的程序爱好者的自学教材。

图书在版编目(CIP)数据

C#程序设计/胡艳菊主编. —北京: 北京大学出版社, 2012.9

(21 世纪全国应用型本科计算机案例型规划教材)

ISBN 978-7-301-16528-7

I. ①C… II. ①胡… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 197077 号

书 名: C#程序设计

著作责任者: 胡艳菊 主编

策划编辑: 郑 双

责任编辑: 郑 双

标准书号: ISBN 978-7-301-16528-7/TP · 1239

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn> <http://www.pup6.cn>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667 出版部 62754962

电子信箱: pup_6@163.com

印 刷 者: 三河市博文印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

经 销 者: 787 毫米×1092 毫米 16 开本 21.5 印张 491 千字

经 销 者: 2012 年 9 月第 1 版 2012 年 9 月第 1 次印刷

定 价: 40.00 元

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究

举报电话: 010-62752024

电子邮箱: fd@pup.pku.edu.cn

信息技术的案例型教材建设

(代丛书序)

刘瑞挺

北京大学出版社第六事业部在 2005 年组织编写了《21 世纪全国应用型本科计算机系列实用规划教材》，至今已出版了 50 多种。这些教材出版后，在全国高校引起热烈反响，可谓初战告捷。这使北京大学出版社的计算机教材市场规模迅速扩大，编辑队伍茁壮成长，经济效益明显增强，与各类高校师生的关系更加密切。

2008 年 1 月北京大学出版社第六事业部在北京召开了“21 世纪全国应用型本科计算机案例型教材建设和教学研讨会”。这次会议为编写案例型教材做了深入的探讨和具体的部署，制定了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向应用、能力驱动、精选案例、严把质量；在风格上力求文字精练、脉络清晰、图表明快、版式新颖。这次会议吹响了提高教材质量第二战役的进军号。

案例型教材真能提高教学的质量吗？

是的。著名法国哲学家、数学家勒内·笛卡儿(Rene Descartes, 1596—1650)说得好：“由一个例子的考察，我们可以抽出一条规律。(From the consideration of an example we can form a rule.)”事实上，他发明的直角坐标系，正是通过生活实例而得到的灵感。据说是在 1619 年夏天，笛卡儿因病住进医院。中午他躺在病床上，苦苦思索一个数学问题时，忽然看到天花板上有一只苍蝇飞来飞去。当时天花板是用木条做成正方形的格子。笛卡儿发现，要说出这只苍蝇在天花板上的位置，只需说出苍蝇在天花板上的第几行和第几列。当苍蝇落在第四行、第五列的那个正方形时，可以用(4, 5)来表示这个位置……由此他联想到可用类似的办法来描述一个点在平面上的位置。他高兴地跳下床，喊着“我找到了，找到了”，然而不小心把国际象棋撒了一地。当他的目光落到棋盘上时，又兴奋地一拍大腿：“对，对，就是这个图”。笛卡儿锲而不舍的毅力，苦思冥想的钻研，使他开创了解析几何的新纪元。千百年来，代数与几何，井水不犯河水。17 世纪后，数学突飞猛进的发展，在很大程度上归功于笛卡儿坐标系和解析几何学的创立。

这个故事，听起来与阿基米德在浴池洗澡而发现浮力原理，牛顿在苹果树下遇到苹果落到头上而发现万有引力定律，确有异曲同工之妙。这就证明，一个好的例子往往能激发灵感，由特殊到一般，联想出普遍的规律，即所谓的“一叶知秋”、“见微知著”的意思。

回顾计算机发明的历史，每一台机器、每一颗芯片、每一种操作系统、每一类编程语言、每一个算法、每一套软件、每一款外部设备，无不像闪光的珍珠串在一起。每个案例都闪烁着智慧的火花，是创新思想不竭的源泉。在计算机科学技术领域，这样的案例就像大海边上的贝壳，俯拾皆是。

事实上，案例研究(Case Study)是现代科学广泛使用的一种方法。Case 包含的意义很广：包括 Example 例子，Instance 事例、示例，Actual State 实际状况，Circumstance 情况、事件、境遇，甚至 Project 项目、工程等。

我们知道在计算机的科学术语中，很多是直接来自日常生活的。例如 Computer 一词早在 1646 年就出现于古代英文字典中，但当时它的意义不是“计算机”而是“计算工人”，即专门从事简单计算的工人。同理，Printer 当时也是“印刷工人”而不是“打印机”。正是

由于这些“计算工人”和“印刷工人”常出现计算错误和印刷错误，才激发查尔斯·巴贝奇(Charles Babbage, 1791—1871)设计了差分机和分析机，这是最早的专用计算机和通用计算机。这位英国剑桥大学数学教授、机械设计专家、经济学家和哲学家是国际公认的“计算机之父”。

20 世纪 40 年代，人们还用 Calculator 表示计算机器。到电子计算机出现后，才用 Computer 表示计算机。此外，硬件(Hardware)和软件(Software)来自销售人员。总线(Bus)就是公共汽车或大巴，故障和排除故障源自格瑞斯·霍普(Grace Hopper, 1906—1992)发现的“飞蛾子”(Bug)和“抓蛾子”或“抓虫子”(Debug)。其他如鼠标、菜单……不胜枚举。至于哲学家进餐问题，理发师睡觉问题更是操作系统文化中脍炙人口的经典。

以计算机为核心的信息技术，从一开始就与应用紧密结合。例如，ENIAC 用于弹道曲线的计算，ARPANET 用于资源共享以及核战争时的可靠通信。即使是非常抽象的图灵机模型，也受到二战时图灵博士破译纳粹密码工作的影响。

在信息技术中，既有许多成功的案例，也有不少失败的案例；既有先成功而后失败的案例，也有先失败而后成功的案例。好好研究它们的成功经验和失败教训，对于编写案例型教材有重要的意义。

我国正在实现中华民族的伟大复兴，教育是民族振兴的基石。改革开放以来，我国高等教育在数量上、规模上已有相当的发展。当前的重要任务是提高培养人才的质量，必须从学科知识的灌输转变为素质与能力的培养。应当指出，大学课堂在高新技术的武装下，利用 PPT 进行的“高速灌输”、“翻页宣科”有愈演愈烈的趋势，我们不能容忍用“技术”绑架教学，而是让教学工作乘信息技术的东风自由地飞翔。

本系列教材的编写，以学生就业所需的专业知识和操作技能为着眼点，在适度的基础知识与理论体系覆盖下，突出应用型、技能型教学的实用性和可操作性，强化案例教学。本套教材将会有机融入大量最新的示例、实例以及操作性较强的案例，力求提高教材的趣味性和实用性，打破传统教材自身知识框架的封闭性，强化实际操作训练，使本系列教材做到“教师易教，学生乐学，技能实用”。有了广阔的应用背景，再造计算机案例型教材就有了基础。

我相信北京大学出版社在全国各地高校教师的积极支持下，精心设计，严格把关，一定能够建设出一批符合计算机应用型人才培养模式的、以案例型为创新点和兴奋点的精品教材，并且通过一体化设计、实现多种媒体有机结合的立体化教材，为各门计算机课程配齐电子教案、学习指导、习题解答、课程设计等辅导资料。让我们用锲而不舍的毅力，勤奋好学的钻研，向着共同的目标努力吧！

刘瑞挺教授 本系列教材编写指导委员会主任、全国高等院校计算机基础教育研究会副会长、中国计算机学会普及工作委员会顾问、教育部考试中心全国计算机应用技术证书考试委员会副主任、全国计算机等级考试顾问。曾任教育部理科计算机科学教学指导委员会委员、中国计算机学会教育培训委员会副主任。PC Magazine《个人电脑》总编辑、CHIP《新电脑》总顾问、清华大学《计算机教育》总策划。

前 言

本书是响应教育部教学改革号召，以培养应用型人才为目的，注重理论联系实践，面向高校计算机本科专业的 C#程序设计教材。通过学习本书，学生在掌握基本理论知识的同时与实践相结合，能够尽快将所学知识应用于程序设计中，从事 C#程序设计有关的实际工作。

本书由浅入深，从最基本的语法开始讲起，逐步深入到面向对象、Windows 程序设计、数据库、网络等高级编程方法，结构完整、内容精要。本书的读者可以是程序设计的入门者，即使没有编程经验，通过学习本书的实例并借助 C#语言的强大功能，也可以很快学习到如何进行数据库、网络等复杂程序设计，快速成为 C#程序员。

C#是由 C 和 C++发展而来的一种“简单、高效、面向对象、类型安全”的程序设计语言，综合了 Visual Basic 的高效率和 C++的强大功能。C#是 .NET 的关键语言，是整个.NET 平台的基础。.NET 是一种以服务方式递交软件的策略，是 Microsoft 公司的新战略，所有 Microsoft 的产品都围绕此战略开发。.NET 能使用户通过 Web 与众多的智能设备交互，同时确保用户而不是应用程序控制此交互。.NET 使得用户对应用程序、服务、个性化设备的体验简单、一致且安全。

本书由吉林化工学院胡艳菊老师担任主编，共分为 12 章，带*章节为选修内容，其中，第 6、7 章由北京科技大学孙建伟老师编写，第 12 章由首钢胜利机械厂技工学校杜云奇老师编写，其余各章由胡艳菊老师编写，北华大学申野老师负责对全书的图、表进行加工处理以及全书的格式修改和内容校正。在本书的写作过程中，得到了家人以及吉林化工学院艾学忠、胡忆洧和季玉茹教授的大力支持和帮助，在此表示衷心的感谢！

在参考文献中罗列的参考书非常有限。有些知识点的获得很偶然，有些是来自某次的网络资料阅读，有些是来自某次朋友的聊天。当然之所以会有这本书的知识架构和知识点是来自于编者多年的经验积累。经验证明，这样的架构安排可在有限的学时内，给学生进行最快、最有效、最实惠的教学。

所有的知识都是人类文明成果的体现，是不断传承、发扬和进步的。在这里感谢所有在本书编写过程中给予帮助和建议的朋友及那些曾经交流过的良师益友。参考文献所列有限，未能一一列出，但是这里仍然要深深感谢他们，如有未尽事宜，尚且谅解。

由于编者水平有限，书中难免存在不当之处，恳请读者批评指正。

编 者

2012 年 6 月

目 录

第 1 章 .NET 基本知识	1	3.1.1 面向对象简介	48
1.1 .NET 简介	2	3.1.2 C#中的对象和类	50
1.1.1 .NET 出现的历史背景	2	3.2 构造函数和析构函数	53
1.1.2 .NET 的概念定位	5	3.2.1 构造函数简介	53
1.1.3 .NET 的发展	5	3.2.2 无参构造函数	53
1.2 .NET 平台	6	3.2.3 有参构造函数	54
1.2.1 .NET 平台概述	6	3.2.4 析构函数	56
1.2.2 .NET 框架	8	3.3 成员函数	56
1.3 .NET 应用程序	9	3.4 命名空间	63
1.4 集成开发环境	11	小结	65
1.4.1 集成开发环境简介	11	课后题	66
1.4.2 开发简单应用项目	14		
小结	15	第 4 章 C#中的继承	67
课后题	15	4.1 继承	68
第 2 章 C#基础知识	17	4.2 在 C#中实现类继承	69
2.1 C#程序结构	18	4.2.1 简单继承	69
2.2 数据类型	19	4.2.2 base 关键字	72
2.2.1 变量	19	4.2.3 覆盖	76
2.2.2 数据类型转换	20	4.2.4 重写	79
2.2.3 值类型与引用类型	26	4.3 抽象类和抽象方法	83
2.2.4 常值变量	28	4.4 接口	85
2.3 运算符和表达式	29	4.4.1 接口简介	85
2.4 程序流程控制结构	33	4.4.2 接口的多继承	89
2.4.1 选择结构	33	4.4.3 显式接口实现	92
2.4.2 循环结构	36	小结	94
2.5 数组	41	课后题	95
2.6 枚举	42		
2.7 结构体	43	第 5 章 C#高级面向对象	96
小结	45	5.1 属性	97
课后题	45	5.1.1 属性简介	97
第 3 章 C#实现面向对象	47	5.1.2 属性类型	103
3.1 C#的类和对象	48	5.2 索引器	108
		5.3 委托	123

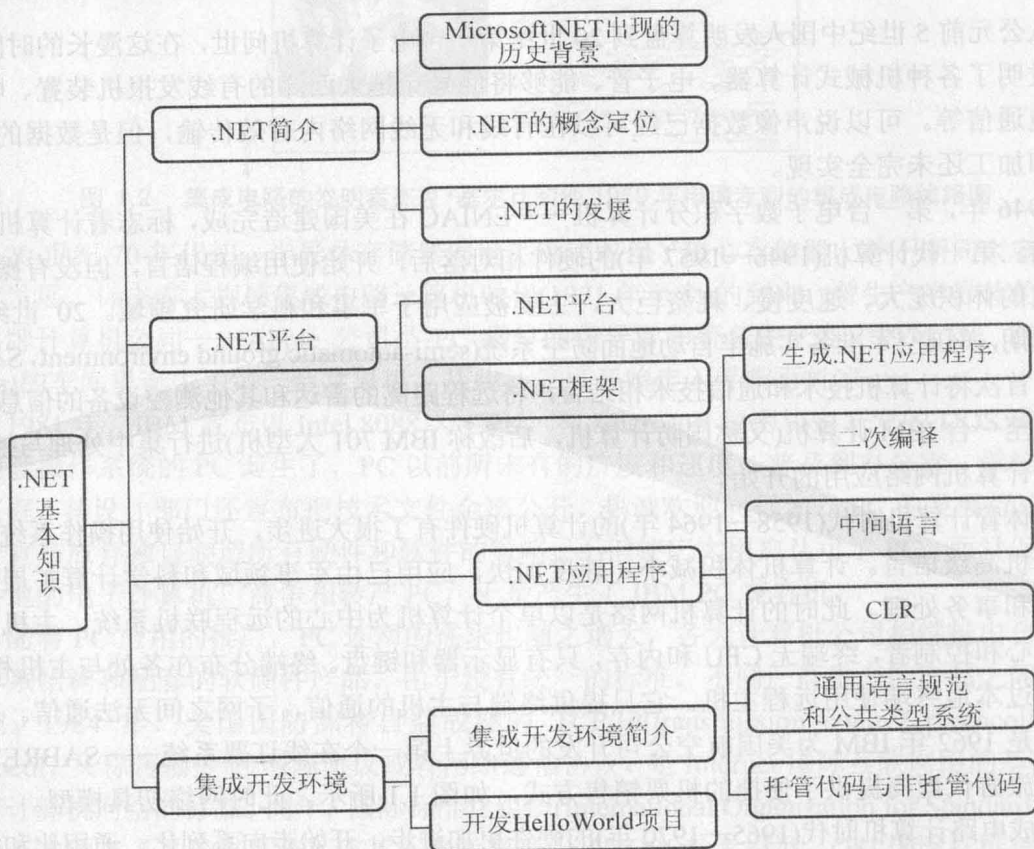
5.4 事件	125	课后题	218
小结	128		
课后题	128		
第6章 数组和集合对象*	129	第9章 ADO.NET	219
6.1 数组和 System.Array 对象	130	9.1 ADO.NET 组成及工作原理	221
6.2 System.Collections 命名空间	138	9.2 Connection 对象	222
6.2.1 Hashtable 类	138	9.2.1 自动生成连接字符串	222
6.2.2 ArrayList 类	140	9.2.2 手写代码	223
6.2.3 其他集合类	142	9.3 Command 对象	224
小结	143	9.3.1 自动填充 Command 对象	224
课后题	144	9.3.2 手写代码	225
第7章 C#中的文件处理	145	9.4 DataReader 对象	225
7.1 BinaryReader 类和 BinaryWriter 类	146	9.5 DataAdapter 和 DataSet 对象	226
7.2 Stream 类	147	9.6 .NET 事务处理	227
7.2.1 MemoryStream 类	147	9.7 综合实例	228
7.2.2 BufferedStream 类	148	9.7.1 自动生成数据访问	228
7.2.3 FileStream 类	150	9.7.2 手写代码访问数据库	229
7.2.4 CryptoStream 类	151	小结	232
7.3 Directory 类和 File 类	152	课后题	232
小结	157		
课后题	157	第10章 ASP.NET	234
第8章 WinForms 基础知识	158	10.1 ASP.NET 简介	235
8.1 WinForms	159	10.2 VS.NET 的安装	235
8.1.1 Windows 应用程序	159	10.3 ASP.NET 的开发	236
8.1.2 窗体	162	小结	250
8.1.3 this 关键字	165	课后题	250
8.1.4 事件函数	166		
8.2 消息框	166	第11章 Windows 应用程序项目 开发案例*	251
8.3 控件	169	11.1 需求分析	252
8.3.1 基础控件	171	11.2 可行性分析	253
8.3.2 LinkLabel	175	11.3 系统框图设计	254
8.3.3 简易资源管理器的制作	177	11.4 数据库设计	254
8.3.4 度量专题	186	11.4.1 E-R 图	254
8.3.5 选择专题	190	11.4.2 表字段分析	255
8.3.6 制作文本编辑器*	191	11.4.3 关系图设计	262
小结	217	11.4.4 存储过程设计	262
		11.5 系统功能设计	264
		11.5.1 登录界面	264
		11.5.2 系统主界面	264

11.5.3 人员信息卡片式维护界面 ...	265	12.4.1 E-R 图分析	294
11.5.4 工资管理	268	12.4.2 数据库设计	295
11.5.5 医疗保险管理	270	12.5 公共类的实现	299
11.5.6 活动管理	272	12.5.1 Customer 类	299
11.5.7 经费管理	273	12.5.2 DB 类	300
11.5.8 统计	276	12.5.3 DBCustomer 类	300
11.5.9 报表	277	12.5.4 Order 类	301
11.5.10 系统管理	278	12.5.5 DBOrder 类	302
11.6 系统功能实现	281	12.5.6 Pet 类	303
11.6.1 登录	281	12.5.7 DBPet 类	304
11.6.2 系统主界面	282	12.5.8 PetDetail 类	304
11.6.3 人员信息卡片式维护	283	12.5.9 DBPetDetail 类	305
11.6.4 显示数据	285	12.5.10 Supply 类	305
11.6.5 教师姓名与编号绑定	286	12.5.11 DBSupply 类	305
11.6.6 报表功能	287	12.5.12 Images 类	306
11.6.7 数据备份功能	287	12.6 页面设计及相关代码分析	306
第 12 章 宠物网站的功能设计*	290	12.6.1 宠物网站的自定义 控件设计	306
12.1 网站简介	291	12.6.2 首页设计及其代码分析	307
12.2 需求分析	292	12.6.3 个人用户界面及其 代码分析	311
12.2.1 理解需求	292	12.6.4 供应商界面及其代码分析 ...	317
12.2.2 分析需求	293	12.6.5 管理员界面及其代码分析 ...	323
12.3 模块关系图	293	参考文献	326
12.4 数据库分析	294		

第 1 章

.NET 基本知识

知识结构图



学习目标

- (1) 了解 Microsoft.NET 的历史背景。
- (2) 了解 .NET 的发展。
- (3) 了解 .NET 框架。
- (4) 认识 .NET 应用程序。
- (5) 认识 .NET 集成开发环境及开发过程。

1.1 .NET 简介

Microsoft.NET(简称.NET)是一个新的开发平台,这个平台融合了从操作系统到程序设计语言等方面的内容,概念繁多。本章的学习目的是了解.NET 的出现和发展,明确其组织架构,并且知道一个.NET 应用程序的开发、编译和执行过程的原理和实际应用。

1.1.1 .NET 出现的历史背景

.NET 的出现不是计算机历史上的空穴来风,而是由其时的计算机硬件水平、软件发展决定的,尤其是网络硬件、软件和应用市场的繁荣发展对 Microsoft 公司提出此新理念起到了关键作用。

从公元前 5 世纪中国人发明算盘到 20 世纪第一台电子计算机问世,在这漫长的时间里,人们发明了各种机械式计算器、电子管、能够将信号穿越大西洋的有线发报机装置、电话、无线电通信等。可以说声像数据已经可以在有线和无线网络内有效传输,但是数据的自动计算和加工还未完全实现。

1946 年,第一台电子数字积分计算机——ENIAC 在美国建造完成,标志着计算机时代的开始。第一代计算机(1946—1957 年)的硬件相对落后,开始使用编程语言,但没有操作系统。它的体积庞大、速度慢、耗资巨大,主要被应用于军事和科学研究领域。20 世纪 50 年代中期,美国空军准备实施半自动地面防空系统(semi-automatic ground environment, SAGE)计划,首次将计算机技术和通信技术相结合,将远程距离的雷达和其他测控设备的信息经线路汇集至一台 IBM 计算机(又称国防计算机,后改称 IBM 701 大型机)进行集中处理与控制,标志着计算机网络应用的开始。

晶体管计算机时代(1958—1964 年)的计算机硬件有了很大进步,开始使用操作系统和各种计算机高级语言。计算机体积减小、速度加快,应用已由军事领域和科学计算扩展到数据处理和事务处理。此时的计算机网络是以单个计算机为中心的远程联机系统。主机是网络的中心和控制者,终端无 CPU 和内存,只有显示器和键盘。终端分布在各处与主机相连,用户通过本地终端使用远程主机。它只提供终端与主机的通信,子网之间无法通信。其典型代表是 1962 年 IBM 为美国航空公司开发的世界上第一个在线订票系统——SABRE,为乘客及旅游代理商提供了便捷的机票销售方式,如图 1.1 所示。此时网络初具模型。

集成电路计算机时代(1965—1970 年)的硬件更加进步,开始走向系列化、通用化和标准化,操作系统进一步完善,高级语言数量更多。计算机体积、质量减小,速度和可靠性进一步提高,主要被应用于科学计算、数据处理和过程控制。此时的计算机网络是利用多个主机通过通信线路互连,为用户提供服务。主机之间不直接用线路连接,而由接口报文处理机(IMP)转接后互连。IMP 和它们之间互连的通信线路一起负责主机间的通信任务,构成通信子网。通信子网互连的主机负责运行程序,实现资源共享,组成资源子网。计算机网络概念初步形成,典型代表是美国国防部高级研究计划局协助开发的 ARPANET——Internet 的前身。1969 年 12 月,ARPANET 投入运行,标志着计算机网络的兴起。集成电路的发明者杰克·基尔比(Jack Kilby)和他 1959 年申请专利的集成电路线路图如图 1.2 所示。



图 1.1 1962 年的 SABRE 预定系统

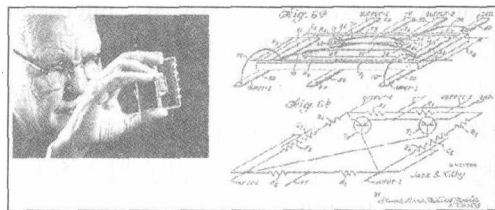


图 1.2 集成电路的发明者杰克·基尔比和他 1959 年申请专利的集成电路线路图

20 世纪 70 年代初, 半导体存储器问世, 迅速取代了磁心存储器, 并不断向大容量、高速度发展, 标志着大规模集成电路计算机时代(1971 年至今)的到来。曾生产半导体的美国著名品牌计算机公司——Apple 公司是 PC 最早的倡导者和著名生产商。1976 年, 在乔布斯(Jobs)的车库里诞生了第一台苹果机, 从此计算机开始走入寻常百姓家。

1981 年, IBM 宣布以 Intel 8088 为 CPU, 与 Microsoft 秘密协定开发的 DOS 操作系统为配套操作系统的 PC 诞生了, PC 以前所未有的广度和速度, 普及到办公室、学校、商店和家庭, 其设计部门还宣布把技术文件全部公开, 热诚欢迎同行加入 PC 的发展行列。IBM 公开宣布放弃独自制造所有硬件和软件策略, 不但使广大用户认可了 PC, 而且促使全世界的电子计算机厂商争相转产 PC, 于是产生了 IBM PC 兼容机。

随着 PC 应用的推广, PC 连网的需求也随之增大, 各大计算机公司相继推出自己的网络体系结构和配套的软硬件产品。由于没有统一的标准, 不同厂商生产的产品之间互连很困难。1984 年, 美国国防部将日益成熟的 TCP/IP(transmission control protocol/Internet protocol, 又称传输控制/网际协议或者网络通信协议, 是 Internet 国际互联网络的基础)作为所有计算机网络的标准。同年, 国际标准化组织(International Organization for Standardization, ISO)正式颁布了“开放系统互连基本参考模型”, 即 OSI 参考模型。OSI 国际标准使计算机网络体系结构实现了标准化。计算机开始能够互通互连。

这个时期(1980—1990 年), 各种基于 PC 互连的微机局域网纷纷出台, 局域网系统的典型结构是在共享介质通信网平台上共享文件服务器, 即为所有连网 PC 设置一台专用的、可共享的网络文件服务器。PC 面向用户, 微机服务器专用于提供共享文件资源。它是客户机/服务器(C/S)模式。

美国国家科学基金会(National Science Foundation, NSF)鼓励并资助美国各科研机构、大学共享其 4 个计算机主机, 后又在美国 5 所著名大学建立 5 个超级计算中心, 并避开 DARPA NET (ARPANET 的分支之一)受控于政府的军用主干网, 遵循 TCP/IP 协议, 出资自建 NSFNET 广

域网。截止到20世纪80年代末,并入NSFNET网络的大学、科研机构、学术团体、开发商的资源子网有3000多个,共享网络的主机近10万台。此时,Internet雏形具备。

进入20世纪90年代,光纤使数据远距离高速有线传播成为现实。NSFNET连接全美上千万台计算机,拥有几千万个用户,是Internet最主要的成员网,美洲以外的网络也逐渐接入NSFNET主干或其子网。Internet开始进入商业化运作模式。

20世纪90年代初,虽然人们希望利用网络进行学习和交流,但是连入Internet还是非常专业和复杂的。1991年8月6日,英国科学家蒂姆·伯纳斯-李(Tim Berners-Lee)提出万维网(world wide web, WWW)设想,他和同事为方便共享研究成果,在欧洲粒子物理研究所(CERN)一部NeXT(Apple公司打造)主机上架设了人类历史上的第一个互联网网站。

万维网被公认为促使Internet迅速发展的重要发明。Internet的诞生比万维网早15年,但起初因使用技术复杂难以普及。万维网借助超文本链接,把不同计算机上的文本、图像、声音等文档链接在一起,使人们不必受计算机操作系统类别和地域等限制,即可自由浏览和分享信息,Internet的操作因而大大简化。万维网问世后广为采用,普及速度惊人。正如蒂姆·伯纳斯-李所说:“起飞原因在于全球的人可随意融入参与。”

1993年美国宣布建立国家信息基础设施(NII)后,全世界许多国家纷纷制定和建立本国的NII,使计算机网络进入了一个崭新的高速发展阶段。1995年Sun推出的Java语言第一次可以开发出能运行在所有操作系统之上的程序代码。Java技术使开发人员能为所有计算机只编写一次程序。1996年,Sun开发XML语言,使关系数据库数据可以在网络内按照XML标准快速传输。同年Sun向所有主流硬件和软件厂商许可Java。

美国政府于1996年开始研究发展更加快速可靠的互联网2(Internet 2)。它具有更大的地址空间和网络规模,接入网络的终端种类和数量更多,网络应用更广泛;并且具有更快、更安全、更及时、更方便、更可管理、更有效的特点。之后,高速网络在世界各国迅速普及,人们越来越希望网络带给自己更加便捷的学习、工作和生活方式。能够跨平台的Java语言成为较热门的语言之一。2000—2001年,在Internet创业公司和大公司的助力下,Sun价格高昂的服务器计算机需求强劲,在2000年9月将Sun的股价推到历史最高点,即258.75美元。至此,网络的商业利益达到一个顶点。这给其时的Microsoft公司造成了巨大的压力。

Microsoft公司于1981年成立,同年,IBM推出带有Microsoft 16位操作系统MS-DOS 1.0的PC。随着IBMPC的大获全胜,Microsoft迅速占领了PC主要的操作系统市场。之后,Microsoft又陆续推出了与同期PC配套的Windows操作系统系列,Windows以其图形化界面、方便快捷的操作等优点,被广大非计算机专业用户接受,Microsoft成了PC操作系统的霸主。在1990—1995年期间,Microsoft除了不断完善自己的操作系统外,还与众多合作伙伴合作,开发Office办公软件、各种程序设计语言及其快速可视化集成开发环境、SQL Server数据库等,并且使用其Visual C++(以下简称VC++)等程序设计语言,已经能够开发出功能强大的C/S结构的程序,巩固了其在计算机软件市场的地位。1995年之前的计算机网络对全世界普通PC用户来说还很遥远。因此,Microsoft没有把精力过多地投入到网络技术开发,而是放在了竞争非常激烈的操作系统开发上。

在1995—2000年期间,随着PC硬件性能的不断提 高、软件技术的高速发展,高速互联网及其技术的迅速普及,Microsoft公司开始不断完善自己的网络操作系统和软件产品。2000年6月22日,Microsoft推出.NET战略,希望借助.Net战略的有效实施,使Microsoft

公司成为一个繁荣两个世纪的传世企业。.NET 出现的历史背景概括如表 1-1 所示。

表 1-1 .NET 出现的历史背景

时 代	硬 件	软 件	网 络	计算机应用
机械式、继电式计算机(公元前 5 世纪—1945 年)			发明有线信号、无线信号通信	计算、电话、电报
电子管(1946—1957 年)	体积庞大、速度慢、耗资巨大、可靠性差	开始使用编程语言,但没有操作系统	SAGE 计划的实施是计算机网络的开始	军事应用和科学研究
晶体管(1958—1964 年)	体积减小、速度加快、可靠性提高	开始使用操作系统和各种计算机高级语言	集中式计算,网络初具模型(诞生阶段)	已由军事领域和科学计算扩展到数据处理和事务处理
大规模集成电路(1965—1970 年)	计算机体积、质量减小、速度和可靠性进一步提高	操作系统进一步完善,高级语言数量更多	有资源子网(网络形成阶段)	科学计算、数据处理和过程控制
超大规模集成电路(1971 年至今)	大容量、高速度 PC 出现	功能更强大、种类更丰富、2000 年 .NET 出现	互通互连、局域网形成、广域网形成、网际网形成、发明万维网高速网形成	

1.1.2 .NET 的概念定位

.NET 战略是 Microsoft 的又一大胆构想。它包括 3 个主要方面:一是利用 PC 的强大性能使其在 Internet 中参与分布式计算,转变传统的集中式计算和 C/S 计算,在网络中发挥 PC 在软硬件资源上的优势;二是转变 Internet 的传统用途,使得 Internet 可以为用户提供更好的服务,包括为用户打造所有联网的智能设备都能访问的数据空间、为程序开发人员提供更方便的跨平台的站点开发和部署服务;三是将软件作为服务,使得应用程序开发中所需的组件在被 Internet “云”分隔的系统中获取,转变组件只能存储在本地的开发方法。因此,通过 .NET 可以将用户数据存放在网络上,随时随地地访问该数据。这是一种全新的平台,完全以 Internet 为中心。以这种模式创建的应用程序可以通过任何浏览器在任何设备上访问。这种应用程序可以充分地利用 Internet 的功能。

Microsoft 对 .NET 的概念定位是,.NET 本身是一系列技术方案和产品;它是以 Internet 为中心的一种全新的平台,也是一个开发和运行软件的新环境;它创建可以通过任何浏览器、任何设备访问的应用程序,用于构建、配置、运行 Web 服务和应用程序的多语言环境,即 “any time, any place and on any device”。

1.1.3 .NET 的发展

自 2000 年 .NET 战略提出以来,.NET 一共经历了 1.0~4.0 几个版本。其中,2003 年的 1.1 版本基本将其理念体现尽致,开发和文档资料也非常完整,尤其是其中文版本也比较稳定,因此受到广大开发者的喜爱。其后的各个版本在技术上进行了部分扩充和改进,使得 .NET 开发更加方便、快捷,并且支持向下兼容。尤其是 C#4.0 在语言和语法上的限制越



表 1-2 .NET 版本发展

版本	版本号	发布日期	Visual Studio	Windows 集成环境
1.0	1.0.3705.0	2002 年 2 月 13 日	Visual Studio .NET	
1.1	1.1.4322.573	2003 年 4 月 24 日	Visual Studio .NET 2003	Windows Server 2003
2.0	2.0.50727.42	2005 年 11 月 7 日	Visual Studio 2005	
3.0	3.0.4506.30	2006 年 11 月 6 日		Windows Vista、Windows Server 2008
3.5	3.5.21022.8	2007 年 11 月 19 日	Visual Studio 2008	Windows 7、Windows Server 2008 R2
4.0	4.0.30319.1	2010 年 4 月 12 日	Visual Studio 2010	

1.2 .NET 平台

.NET 主要由 3 个组件构成: .NET 产品和服务、第三方.NET 服务和.NET 平台本身。其中, .NET 平台是核心, 具体如图 1.3 所示。

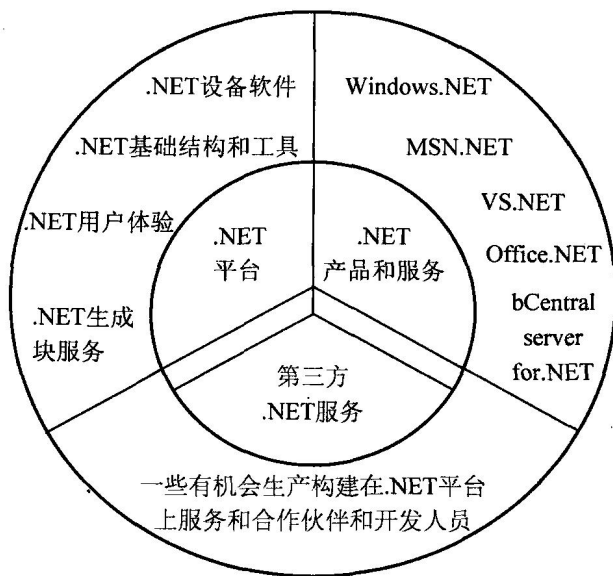


图 1.3 .NET 组件

1.2.1 .NET 平台概述

.NET 平台是用于开发新一代软件的一套工具、服务集和模型，旨在提供下列功能：集成 Internet 上的任何资源、请求各种服务提供的功能以及将这些功能集成到应用程序中。由于 .NET 是利用早期的组件方法生成的，因此它可以使应用程序的开发、实现和部署更快、更容易。 .NET 平台构建在 XML 和 Internet 协议之上，这两个协议为 .NET 新平台提供技术基础。

可扩展的标记语言(extensible markup language, XML)的特征是将实际数据与表现形式分开。这使得 XML 完全不同于 HTML。它被认为是构建新一代 Internet 的关键,使得用户能够对网站信息进行整理、编程和编辑。它还提供了一种将数据发布到多种设备的方法。它与 Internet 协议相结合,使得站点能够协同工作并彼此交互,从而生成一种“Web 服务组合体”。

TCP/IP 是 Internet 基础协议。它的最高层是应用层,包括所有的高层协议:HTTP(超文本传输协议)、FTP(文件传输协议)、DNS(域名系统)、SMTP(电子邮件传输协议)、NNTP(网络新闻传输协议)和 SOAP(简单对象访问协议)。SOAP 以 XML 形式提供了一个简单、轻量的用于在分散或分布环境中交换结构化和类型信息的机制,是基于 XML/HTTP 的协议。通过该协议,可以以独立于平台的方式访问服务、对象和服务器。SOAP 有助于在分散的分布式环境中进行信息交换,这是它完全适用于 .NET 平台的一个主要原因。.NET 平台的组件如图 1.4 所示。

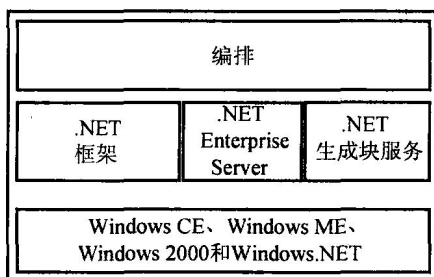


图 1.4 .NET 平台的组件

.NET 应用程序可以构建在所有最近的 Windows 操作系统(Windows CE、Windows ME、Windows 2000)和 Windows.NET 上。它们都属于 Windows 2000 服务器系列。因此可以说,.NET 平台是构建在 Windows 2000 服务器系列的可伸缩性、可靠性、安全性和可管理性的基础之上的。

.NET 最重要的组件是 .NET 框架(.NET Framework)。.NET 框架由公共语言运行库(CLR)、基类、数据访问类、XML、ASP.NET、WinForms 和语言类库构成。

.NET Enterprise Server 由服务器系列组成,通过该服务器系列,可以快速地生成和管理一个集成的、启用了 Web 的企业系统。设计的这些服务器具有可伸缩性,并且可以与最新的 Internet 和数据标准实现互操作。7 个核心 .NET Enterprise Server 分别是 SQL Server 2000、ISA (Internet Security and Acceleration) Server 2000、Host Integration Server、Exchange 2000 Server、Exchange 2000 Conferencing Server、Commerce Server 2000、BizTalk Server 2000 和 Application Server 2000。

.NET 生成块服务组件包括由 Microsoft 以及其他应用程序服务提供商提供的商业 Web 服务。这些服务可以用来创建其他应用程序以及其他 Web 服务。这些服务包括以下内容。

(1) 标示:构建在 Microsoft 密码和 Windows 身份验证技术基础之上,提供从密码和 Wallet 到智能卡和生物测量学设备的各种级别的身份验证。通过它,开发人员可以生成为客户提供个性化和私密性的服务,而客户则可以对他们的服务享受新的安全访问级别,无论他们位于什么位置,或者使用何种设备。Windows.NET 的第一个主要版本(代码为“Whistler”)支持此项服务。

(2) 通知和消息处理：将即时消息、电子邮件、传真、语音邮件以及其他通知和消息处理形式集成为一种统一的体验，并且可以传递到任何 PC 或智能设备，构建在 Web 的 Hotmail 电子邮件服务、Exchange 和 Instant Messenger 的基础之上。

(3) 个性化：使用这种服务，可以通过创建规则和首选项进行控制，这些规则和首选项隐式和显式地定义如何处理通知和消息，如何处理要求共享数据的请求，以及多个设备应如何协同工作(例如，使便携式计算机始终与 .NET 存储服务的所有内容保持同步)。此外，还可以通过此服务将用户的数据快速地移动到另一台 PC。

(4) XML 存储：使用通用语言(XML)和协议(SOAP)描述数据的含义，能够在由多个 Web 站点和用户传送和处理数据时维护数据的完整性。这使得 Web 站点成为了非常灵活的服务，可以进行交互、交换和利用彼此的数据。.NET 还为在 Web 上存储数据提供了一个安全、可寻址的位置。用户的每台设备都可以访问该位置，从而可以以最佳方式复制数据，以提高效率和脱机使用这些数据。其他服务可以在用户的同意下访问其存储数据，将 NTFS、SQL Server、Exchange 以及 MSN Communities 的元素结合到了一起。

(5) 日历：用户控制的一个极为重要的方面是时间，如什么时候可以来找我，以及什么时候不要打扰我。随着人们使用的设备的增多、使用时间的增长，以及用户和服务交互的内容的增多，这一点变得非常重要。.NET 提供了一种将工作、社会和家庭日历安全、保密地集成到一起的基础，集成后，可以通过所有设备访问这些日历，其他服务和个人在经过用户同意的情况下也可以访问这些日历。它构建在 Microsoft Outlook 消息、协作客户端以及 Hotmail Calendar 的基础之上。

(6) 目录和搜索：通过 .NET，可以搜索要与之交互的服务和用户。.NET 目录不仅仅是搜索引擎或“黄页”。它们可以以编程方式与服务进行交互，回答有关这些服务的功能和基于特定架构的问题。它们还可以由其他服务聚合和自定义，并可与其他服务组合在一起。

(7) 动态传递：使得 Microsoft 和开发人员可以在需要时动态地提供更高级别的功能以及可靠的自动升级，而无需用户进行安装或配置。.NET 可以自动适应用户所要完成的任务或用户所使用的任何设备。它与依赖于安装的传统应用程序模型相反，在用户希望在多台设备上享受服务带来的好处的今天，这已经成为一种必然。

编排提供进程和 Web 服务是一种既在组织内部运行，又在组织之间运行的组织服务。组织中可能存在多种软件系统和多种硬件平台。此外，组织中可能还存在多个这样的客户、合作伙伴和服务提供商场所。在这种情况下，就需要使用某种工具来处理组织内部以及组织之间的编排任务。BizTalk Server 2000 提供了能够实现编排功能的工具。

1.2.2 .NET 框架

.NET 框架是 .NET 平台最主要的部分。它的作用是提供一个一致的面向对象的编程环境；提供一个将软件部署和版本控制冲突最小化的代码执行环境；提供一个保证代码安全执行的代码执行环境；提供一个可消除脚本环境或解释环境的性能问题的代码执行环境；使开发人员的经验在面对类型大不相同的应用程序时保持一致；按照工业标准生成所有通信，以确保基于 .NET 框架的代码可与其他代码集成。