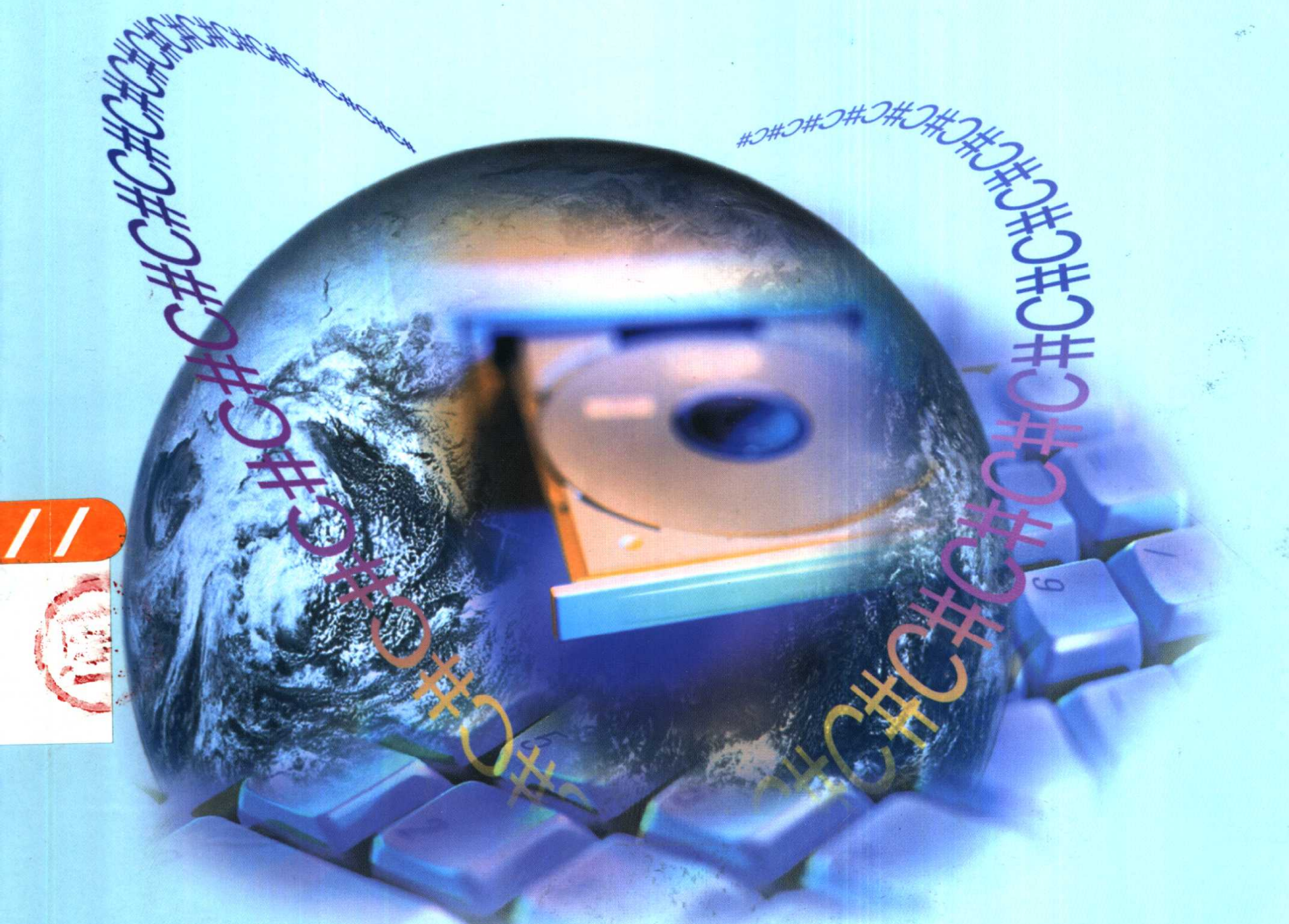




新一代专业软件开发人才培养 系列教材

C# 编程语言

麦中凡 陆永宁 编著



北京航空航天大学出版社

<http://www.buaapress.cn.net>

C# 编程语言

麦中凡 陆永宁 编著

北京航空航天大学出版社

内 容 简 介

C# 编程语言是微软公司推出的新一代软件开发环境 Microsoft .NET 的核心语言,旨在支持下一代 Web 服务的软件开发。

本书分为三篇:A 篇讲 C# 编程语言出现的背景,以较大的篇幅介绍 .NET 的基本思想和 .NET 框架,旨在说明 .NET 框架和 C# 语言是微软为解决 Web 服务的软件开发的核心理念,它们联系紧密不可分。B 篇是 C# 语言规范,基本上是 0.28 版的全译本,以便读者学习 C# 语言特征的每个细节。C 篇是 C# 编程的相关技术,体现全新思想的构件设计、分布式构件、组装运作、数据库、XML 连接、继承历史资源等,综合体现以 C# 实现 Web 服务的编程。

本书是 C# 编程语言的编程入门向导。读者对象是第一线软件开发人员,特别是那些想从 C/C++, VB,Java 编程转向 .NET 环境上开发软件者。对希望了解下一代网络计算技术的高校师生和高层软件工作者也是一本好教材。

图书在版编目(CIP)数据

C# 编程语言/麦中凡等编著. —北京:北京航空航天大学出版社,2001.8

ISBN 7-81077-088-8

I. C… II. 麦… III. C 语言 程序设计
IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 049250 号

C# 编程语言

麦中凡 陆永宁 编著

责任编辑 许传安

责任校对 陈 坤

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:82317024 传真:82328026

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

E-mail: pressell@publica.bj.cninfo.net

河北省涿州市新华印刷厂印装 各地书店经销

*

开本:787×1092 1/16 印张:42.75 字数:1094 千字

2001 年 8 月第 1 版 2001 年 8 月第 1 次印刷 印数:5000 册

ISBN 7-81077-088-8/TP·045 定价:66.00 元

前言

微软公司近期推出 C# 程序设计语言,引起业界普遍关注。为什么有了 VB, VC++, 业界又公认了 Java, 还要推出一个新语言? C# 能代替 VB, VC++, 成为微软统一的语言吗? C# 和 Java 谁将取胜? 我们将以多大的关注去学习 C#? 这对于业界第一线软件开发者的,确是个问题。我们已被各种描述语言(Jscript, VBScript)、置标语言(XML, eXML, HTML, SGML)、模型语言(UML)、体系结构描述语言(ADL)弄得不堪重负,偏偏最直接表述软件的编程语言又出了一个 C#。殊不知,编程语言是专业程序员“找饭碗”的工具。

众所周知,经常换语言滋味是不好受的。这是因为无论什么语言既是表达工具,也是思维工具,习惯了某种语言的表达也就习惯这种语言的思维范型。一旦习惯,改起来就不舒服。前几年,从 C 到 C++, 不少人编程序依然是 DFD 图、结构图、过程模块,最后凑出对象,以 C++ 表示。直到相当一段时间之后,才理解到 OO 开发的好处。一遇到问题,首先想到的是做出什么对象,它和其它对象如何交互,直接用 C++ 表达对象。

一个编程语言不带来新的思维范型是好学的,但学不到软件开发新技术。正如 ALGOL, C 程序员学习 FORTRAN 和 Pascal 一样。仅仅学会一种新的表示法,从 C++ 到 C# 却不是这样, C# 带来新的编程范型,即从面向对象编程到 Web 服务编程。这中间有个业界的巨大转变:从分布式客户/服务器计算转向“软件就是服务”的网络计算。后者把资源共享、软件重用又提升到一个新阶段,租赁、下载在 Internet 上进行。在 ASP(应用服务提供者)提供的各种各样的服务基础上,业务系统的应用开发者几乎不用编制、维护(升级、纠错)软件就可以快速交付十分复杂的系统。而 ASP 用 C# 开发的 Web 服务也是大量地使用其它 ASP 提供的服务(当然也包括以 Java, 甚至 C++ 提供的服务构件)。

微软公司提供的 Microsoft .NET 就是为完善新一代网络计算的环境,而 C# 是其中的核心工具之一,简单说是表达 Web 服务软件最方便的工具。然而微软在开发推行 C# 时,又是非常稳健的。直到本书出版之时, C# 还没有正式发布的 1.0 版,网上能见到的是 0.17, 0.22, 0.28 三个版本(如果把发布 1.0 版算全程的话,那么 11% 的路程已出了三个版本!)。原因是网络计算还在纵深发展,目前成功的应用,如电子商务、企业门户、信息检索不过是网络计算早期的成果, C# 还在不断试探中完善。

本书的使命是全面介绍 C#。显然语言规范(B 篇)和编程技术(C 篇)是学会一种编程语言不可少的两部分。鉴于上述理由,本书增加了 A 篇: C# 语言产生

的背景,共3章,试图说明 Microsoft .NET 以 C# 和 .Net 框架全面支持 Web Service 的开发;C# 实现的核心是通用语言运行时。由此介绍 C# 源程序是如何运行的;最后说明 C# 是最新出现的编程语言,也必然吸收了程序员在使用 VB, VC++ 中的一些经验和编程语言的新技术(如即时编译等)。这些都反映为 C# 的特征。

B 篇共 17 章,完全按 0.28 版 C# Language Specification 译出。其中尚有约 10 小节只有标题没有内容(网上未发布),估计是尚在定稿之中。这也是我们全文照译的原因,章节号都没有变动。

C 篇共 10 章,是用 C# 编写程序的基本知识和一般技术。除一般基本支持以外,重点介绍两方面:构件及组装(汇编);.NET 框架的类库(C# 语言本身没有类库,它用 .NET 框架的通用类库)对编制各种程序的支持,是入门性的而非技巧性的。所有的程序就是基于 Visual Studio .NET Beta 1 版本的。

最后,谈一谈 C# 的前景。

C#, Java, Xml 是网络计算时代的三种最重要的语言。Java+Xml 能做到的事 C#+Xml 也能做到。从对网络计算的支持上, C# 不及 Java,它是平台相关的。但 C#+.Net 框架似乎比 Java+Sun ONE 还要完善。Java 走的是全新的、平台无关的路,也正因为如此,它还要积累相当多的类库支持,开发使用才方便。而 C# 走的是继承、包容历史财富的路,缺点变优点,开发快速、执行效率高。如果 C#+.Net 走得好,今后是 Java 和 C# 并存的局面,即谁也挤不倒、代替不了谁。

然而,对编程语言研究者, C# 确是个值得研究的好语言,它不像 Java 那样“单纯”,比 C++ 增设了不少更安全、更方便的机制;它反映了当今编程语言的新技术,集编译、解释、即时编译于一身。

本书虽以介绍 C# 编程语言为主,但为了说明它的使用,对 Microsoft .NET 的思想、机制、Web Service 软件开发作了较多说明,故也是 C/C++ 程序员转向 .NET 新环境的教材。

原本想尽早编制 C# 概念入门的书,使国人早对 C# 有所接触。在收集资料的过程中已见关于 C# 概念性入门的《Presenting C#》的中译本。故本书立足于编程入门,无形之中对完整性、准确性、实用性要求高多了。

本书仅就网上公布及课题研究中收集材料编写,没有与 C# 设计者 Anders Hejlsberg 磋商。对 C# 和 .NET 的理解仅限作者水平,不确之处欢迎指正。

在成书过程中,樊平、张晓昉、戴彩霞在翻译、校对、编排 B 篇中作了大量的工作。储洪峰完成 A 篇第一稿,最后 A 篇、C 篇由陆永宁完成。程勇、李烨在分析比较、编排打印等工作中也付出极大的努力。

麦中凡教授亲自负责全书结构、撰写、翻译等工作安排,同时进行业务指导,并对全书各部分都进行了详细的修改和审定。

感谢微软中国公司吴雷女士,在我们完成第一稿向她征求意见时,她给我们最新资料,包括《C# Programming With the Public Beta》和《A Programmer's Introduction to C#》两书,使我们作了第三次修改。

本书有幸被教育部计算机课程教学指导委员会和微软中网公司指定为“新一代软件人才培训”教材。受他们的委托,我们将与北航出版社合作,翻译、出版“Microsoft, NET Best Practice”和“Programming C# With Microsoft® , NET Platform”等培训教材。

感谢北航出版社的有关人员,为了配合教育部计算机课程教指委与微软中国公司合办的培训班课程而夜以继日地编辑、出版。

关于批评、指正、帮助的联系地址是:oops@buaa.edu.cn

编著者

2001年7月于北京航空航天大学

目 录

A 篇 C#产生的背景

A1 章 网络计算与 Microsoft .NET	3	A2A2 指 令	34
A1.1 网络计算	3	A3 章 C#的主要特点与用途	44
A1.2 微软的 Internet 道路	5	A3.1 C#的主要特点	44
A1.3 .NET 平台	6	A3.1.1 简单性	44
A1.3.1 Windows .NET	7	A3.1.2 类型系统的统一性	45
A1.3.2 .Net FrameWork	7	A3.1.3 现代性	45
A1.3.3 .Net Enterprise Servers ...	8	A3.1.4 面向对象	46
A1.3.4 .NET 构建模块服务(Building Block Services)	9	A3.1.5 类型安全	46
A1.3.5 BizTalk Orchestration	9	A3.1.6 版本技术	47
A1.4 小 结	10	A3.1.7 兼容性	48
A2 章 .Net Framework 简介	11	A3.1.8 灵活性	48
A2.1 .Net Framework 的基本思想	11	A3.2 C#与其它语言对比	48
A2.2 .Net Framework 组成	13	A3.2.1 C#与 VB.NET	48
A2.3 公共语言运行时	16	A3.2.2 C#与 C++	49
A2.3.1 公共类型系统(CTS)简介	19	A3.2.3 C#与受管 C++	50
A2.3.2 元数据	22	A3.2.4 C#与 JScrip t	50
A2.3.3 语言互用性	23	A3.2.5 C#与 Java	50
A2.3.4 CLR 执行系统	24	A3.3 C#语言的应用	51
A2.4 公共语言规范 CLS	28	A3.3.1 .NET 构件	51
A2.5 .NET 对语言的要求	31	A3.3.2 控制台应用	52
A2.6 小 结	33	A3.3.3 Windows 应用	52
A2A 附录 微软中间语言	33	A3.3.4 Windows 控件	52
A2A1 数据类型	33	A3.3.5 ASP. Net 项目	52
		A3.3.6 Web 控件	53
		A3.3.7 网络服务	53
		A3.4 小 结	53

B 篇 C#语言规范

B1 章 C#语言概述	56	B1.2.2 转 换	61
B1.1 从引例开始	56	B1.2.3 数 组	61
B1.2 类 型	57	B1.2.4 类型系统的统一	63
B1.2.1 预定义类型	59	B1.3 变量和参数	64

B1.4 自动存储管理	67	B2.5.1 条件编译符号	110
B1.5 表达式	69	B2.5.2 预处理表达式	111
B1.6 语句	70	B2.5.3 声明指令	111
B1.7 类	74	B2.5.4 条件编译指令	112
B1.7.1 常量	76	B2.5.5 行指令	114
B1.7.2 域	76	B2.5.6 诊断指令	115
B1.7.3 方法	77	B2.5.7 区段指令	115
B1.7.4 性质	79	B3 章 基本概念	117
B1.7.5 事件	80	B3.1 程序开始	117
B1.7.6 操作符	81	B3.2 程序终止	118
B1.7.7 索引器	82	B3.3 声明	118
B1.7.8 实例构造子	83	B3.4 成员	120
B1.7.9 析构子	84	B3.4.1 名字空间成员	120
B1.7.10 静态构造子	85	B3.4.2 结构成员	120
B1.7.11 继承	85	B3.4.3 枚举成员	121
B1.8 结构	87	B3.4.4 类成员	121
B1.9 接口	88	B3.4.5 接口成员	121
B1.10 委派	89	B3.4.6 数组成员	121
B1.11 枚举	90	B3.4.7 委派成员	121
B1.12 名字空间和组装	91	B3.5 成员访问	122
B1.13 版本	92	B3.5.1 声明的可访问性	122
B1.14 属性	95	B3.5.2 可访问域	123
B2 章 词法结构	97	B3.5.3 保护性访问	125
B2.1 程序	97	B3.5.4 访问限制	125
B2.2 文法	97	B3.6 型构和重载	126
B2.2.1 文法表示法	97	B3.7 作用域	127
B2.2.2 词法文法	98	B3.7.1 名字隐藏	129
B2.2.3 语法规则	98	B3.8 名字空间和类型名	131
B2.3 词法分析	98	B3.8.1 完全限定名	132
B2.3.1 行终结符	99	B3.9 自动存储管理	133
B2.3.2 空白	99	B4 章 类型	135
B2.3.3 注释	100	B4.1 值类型	135
B2.4 标记	101	B4.1.1 缺省构造子	136
B2.4.1 泛代码字符转义序列	101	B4.1.2 结构类型	137
B2.4.2 标识符	102	B4.1.3 简单类型	137
B2.4.3 关键字	104	B4.1.4 整型	138
B2.4.4 字面量	104	B4.1.5 浮点类型	139
B2.4.5 操作符和标点符号	109	B4.1.6 十进制数类型	140
B2.5 预处理指令	109	B4.1.7 bool 类型	140

B4.1.8 枚举类型	141	B6.2.5 用户定义的显式转换	155
B4.2 引用类型	141	B6.3 标准转换	155
B4.2.1 类类型	142	B6.3.1 标准隐式转换	156
B4.2.2 object 类型	142	B6.3.2 标准显式转换	156
B4.2.3 string 类型	142	B6.4 用户定义转换	156
B4.2.4 接口类型	142	B6.4.1 许可的用户定义转换	156
B4.2.5 数组类型	142	B6.4.2 用户定义转换的判定	156
B4.2.6 委派类型	142	B6.4.3 用户定义的隐式转换	157
B4.3 装匣和脱匣	142	B6.4.4 用户定义的显式转换	158
B4.3.1 装匣转换	143	B7 章 表达式	159
B4.3.2 脱匣转换	144	B7.1 表达式分类	159
B5 章 变 量	145	B7.1.1 表达式的值	160
B5.1 变量种类	145	B7.2 操作符	160
B5.1.1 静态变量	145	B7.2.1 操作符优先级和结合性	160
B5.1.2 实例变量	145	B7.2.2 操作符重载	161
B5.1.3 数组元素	146	B7.2.3 单目操作符重载分辨	162
B5.1.4 值参数	146	B7.2.4 双目操作符重载分辨	163
B5.1.5 引用参数	146	B7.2.5 候选的用户定义操作符	163
B5.1.6 输出参数	146	B7.2.6 数值提升	163
B5.1.7 局部变量	147	B7.3 成员查找	164
B5.2 缺省值	147	B7.3.1 基类型	165
B5.3 明确赋值	147	B7.4 函数成员	165
B5.3.1 赋值初始化变量	150	B7.4.1 变元表	167
B5.3.2 无赋值初始化变量	150	B7.4.2 重载分辨	169
B5.4 变量引用	150	B7.4.3 函数成员调用	170
B6 章 转 换	151	B7.4.4 虚函数成员查询	171
B6.1 隐式转换	151	B7.4.5 接口函数成员查询	172
B6.1.1 标识转换	151	B7.5 字面量	172
B6.1.2 隐式数转换	151	B7.5.1 简单名字	172
B6.1.3 隐式枚举转换	152	B7.5.2 基原表达式	173
B6.1.4 隐式引用转换	152	B7.5.3 带括号的表达式	174
B6.1.5 装匣转换	152	B7.5.4 成员访问	174
B6.1.6 隐式常量表达式转换	153	B7.5.5 调用表达式	176
B6.1.7 用户定义的隐式转换	153	B7.5.6 元素访问	178
B6.2 显式转换	153	B7.5.7 this 访问	179
B6.2.1 显式数转换	153	B7.5.8 基访问	180
B6.2.2 显式枚举转换	154	B7.5.9 后缀增减量操作符	180
B6.2.3 显式引用转换	155		
B6.2.4 脱匣转换	155		

B7.5.10 new 操作符	181	B7.11.2 用户定义条件逻辑操作符	207
B7.5.11 typeof 操作符	185	B7.12 条件操作符	207
B7.5.12 checked 和 unchecked 操作符	186	B7.13 赋值操作符	208
B7.6 单目表达式	188	B7.13.1 简单赋值	208
B7.6.1 单目加操作符	188	B7.13.2 复合赋值	210
B7.6.2 单目减操作符	188	B7.13.3 事件赋值	211
B7.6.3 逻辑反操作符	189	B7.14 表达式	211
B7.6.4 按位补操作符	189	B7.15 常量表达式	211
B7.6.5 间接寻址操作符	190	B7.16 布尔表达式	212
B7.6.6 地址操作符	190	B8 章 语 句	214
B7.6.7 前缀增量和减量操作符	190	B8.1 结束点和可达性	214
B7.6.8 强制表达式	191	B8.2 块	216
B7.7 算术操作符	191	B8.2.1 语句表	216
B7.7.1 乘法操作符	191	B8.3 空语句	217
B7.7.2 除法操作符	192	B8.4 标号语句	217
B7.7.3 求余操作符	193	B8.5 声明语句	217
B7.7.4 加法操作符	194	B8.5.1 局部变量声明	218
B7.7.5 减法操作符	196	B8.5.2 局部常量声明	218
B7.8 移位操作符	197	B8.6 表达式语句	219
B7.9 关系操作符	199	B8.7 选择语句	219
B7.9.1 整数比较操作符	199	B8.7.1 if 语句	220
B7.9.2 浮点比较操作符	200	B8.7.2 切换语句	220
B7.9.3 十进制比较操作符	201	B8.8 迭代语句	224
B7.9.4 布尔等价操作符	201	B8.8.1 while 语句	224
B7.9.5 枚举比较操作符	201	B8.8.2 do 语句	225
B7.9.6 引用类型等价操作符	202	B8.8.3 for 语句	225
B7.9.7 串等价操作符	203	B8.8.4 foreach 语句	226
B7.9.8 委派等价操作符	203	B8.9 跳转语句	228
B7.9.9 is 操作符	203	B8.9.1 break 语句	229
B7.9.10 as 操作符	204	B8.9.2 continue 语句	230
B7.10 逻辑操作符	204	B8.9.3 goto 语句	230
B7.10.1 整数逻辑操作符	205	B8.9.4 return 语句	231
B7.10.2 枚举逻辑操作符	205	B8.9.5 throw 语句	232
B7.10.3 布尔逻辑操作符	206	B8.10 try 语句	232
B7.11 条件逻辑操作符	206	B8.11 checked 和 unchecked 语句	235
B7.11.1 布尔条件逻辑操作符	206	B8.12 lock 语句	236

B8.13 using 语句	236	B10.6 性 质	278
B9 章 名字空间	239	B10.6.1 静态性质和实例性质 ...	279
B9.1 编译单元	239	B10.6.2 访问器	280
B9.2 名字空间声明	239	B10.6.3 虚、封固、覆盖和抽象访问器	284
B9.3 使用指令	241	B10.6.4 外部性质	286
B9.3.1 使用别名指令	241	B10.7 事 件	286
B9.3.2 使用名字空间指令	243	B10.7.1 事件访问器	288
B9.4 名字空间成员	245	B10.7.2 静态事件和实例事件 ...	290
B9.5 类型声明	246	B10.7.3 虚、封固、覆盖和抽象访问器	290
B10 章 类	247	B10.7.4 外部事件	290
B10.1 类声明	247	B10.8 索引器	291
B10.1.1 类修饰符	247	B10.8.1 索引器重载	294
B10.1.2 类基规范	248	B10.9 操作符	294
B10.1.3 类 体	250	B10.9.1 单目操作符	295
B10.2 类成员	250	B10.9.2 双目操作符	296
B10.2.1 继 承	251	B10.9.3 转换操作符	296
B10.2.2 new 修饰符	252	B10.10 实例构造子	298
B10.2.3 访问修饰符	252	B10.10.1 构造子初始化式	299
B10.2.4 成分类型	252	B10.10.2 实例变量初始化式	299
B10.2.5 静态成员和实例成员 ...	252	B10.10.3 构造子执行	300
B10.2.6 内嵌类型	253	B10.10.4 缺省构造子	302
B10.2.7 保留的成员名字	257	B10.10.5 私有构造子	302
B10.3 void Finalize();常量	257	B10.10.6 可选的实例构造子参数	303
B10.4 域	259	B10.11 静态构造子	303
B10.4.1 静态域和实例域	260	B10.12 析构子	304
B10.4.2 只读域	261	B11 章 结 构	307
B10.4.3 可变域	262	B11.1 结构声明	307
B10.4.4 域初始化	262	B11.1.1 结构修饰符	307
B10.4.5 变量初始化式	263	B11.1.2 结构接口	308
B10.5 方 法	264	B11.1.3 结构体	308
B10.5.1 方法的参数	265	B11.2 结构成员	308
B10.5.2 静态和实例方法	271	B11.3 类和结构的区别	308
B10.5.3 虚方法	271	B11.3.1 值语义	308
B10.5.4 覆盖方法	273	B11.3.2 继 承	309
B10.5.5 封固方法	275	B11.3.3 赋 值	309
B10.5.6 抽象方法	275	B11.3.4 缺省值	310
B10.5.7 外部方法	277		
B10.5.8 方法体	277		
B10.5.9 方法重载	278		

B11.3.5 装匣和脱匣	310	B15 章 委派	341
B11.3.6 this 的含义	310	B15.1 委派声明	341
B11.3.7 域初始化式	311	B15.2 委派实例化	343
B11.3.8 构造子	311	B15.3 委派调用	343
B11.3.9 析构子	311	B16 章 异常	346
B11.4 结构的例子	312	B16.1 异常的产生	346
B11.4.1 数据库整类型	312	B16.2 类 System.Exception	346
B11.4.2 数据库布尔类型	313	B16.3 异常怎样处理	346
B12 章 数组	316	B16.4 通用异常类	347
B12.1 数组类型	316	B17 章 属性	348
B12.1.1 System.Array 类型	317	B17.1 属性类	348
B12.2 数组创建	317	B17.1.1 属性用法	348
B12.3 数组元素访问	317	B17.1.2 位置和命名参数	349
B12.4 数组成员	317	B17.1.3 属性参数类型	350
B12.5 数组协变	317	B17.2 属性规范	350
B12.6 数组初始化式	318	B17.3 属性实例	353
B13 章 接口	320	B17.3.1 属性的编译	354
B13.1 接口声明	320	B17.3.2 运行时属性实例的再查	354
B13.1.1 接口修饰符	320	B17.4 保留的属性	354
B13.1.2 基接口	320	B17.4.1 属性 AttributeUsage	354
B13.1.3 接口体	321	B17.4.2 Conditional 属性	355
B13.2 接口成员	321	B17.4.3 Obsolete 属性	357
B13.2.1 接口方法	322	附录 BA 不安全代码	359
B13.2.2 接口性质	322	BA.1 不安全上下文	359
B13.2.3 接口事件	323	BA.2 指针类型	362
B13.2.4 接口索引器	323	BA.3 固定的和可移动的变量	364
B13.2.5 接口成员访问	323	BA.4 指针转换	364
B13.3 接口成员的完全限定名	325	BA.5 表达式中的指针	366
B13.4 接口实现	326	BA.5.1 指针间接寻址	366
B13.4.1 显式接口成员实现	327	BA.5.2 指针成员访问	366
B13.4.2 接口映射	329	BA.5.3 指针元素访问	367
B13.4.3 接口实现继承	331	BA.5.4 取地址操作符	368
B13.4.4 接口再实现	333	BA.5.5 指针增量和减量	369
B13.4.5 抽象类和接口	334	BA.5.6 指针算术	369
B14 章 枚举	336	BA.5.7 指针比较	370
B14.1 枚举声明	336	BA.5.8 sizeof 操作符	370
B14.2 枚举修饰符	337	BA.6 固定语句	371
B14.3 枚举成员	337	BA.7 堆栈分配	374
B14.4 枚举值和操作	339		

BA. 8 动态存储分配	375	BC 文 法	390
BB 可交互操作性	378	BC. 1 词法文法	390
BB. 1 ComAliasName 属性	378	BC. 1.1 行终结符	390
BB. 2 ComImport 属性	378	BC. 1.2 空 白	390
BB. 3 ComRegisterFunction 属性	379	BC. 1.3 注 释	391
BB. 4 ComSourceInterface 属性	379	BC. 1.4 标 记	391
BB. 5 ComUnregisterFunction 属性	379	BC. 1.5 泛代码转义序列	391
BB. 6 ComVisible 属性	380	BC. 1.6 标识符	392
BB. 7 DispId 属性	380	BC. 1.7 关键字	392
BB. 8 DllImport 属性	380	BC. 1.8 字面量	393
BB. 9 FieldOffset 属性	381	BC. 1.9 操作符和标点符号	395
BB. 10 Guid 属性	382	BC. 1.10 预处理指令	395
BB. 11 HasDefaultInterface 属性	382	BC. 2 语法规则	397
BB. 12 ImportedFromTypeLib 属性	382	BC. 2.1 基本概念	397
BB. 13 In 和 Out 属性	383	BC. 2.2 类 型	397
BB. 14 IndexerName 属性	383	BC. 2.3 变 量	399
BB. 15 InterfaceType 属性	384	BC. 2.4 表达式	399
BB. 16 MarshalAs 属性	384	BC. 2.5 语 句	402
BB. 17 NoIDispatch 属性	385	BC. 2.6 类	406
BB. 18 PreserveSig 属性	385	BC. 2.7 结 构	412
BB. 19 StructLayout 属性	385	BC. 2.8 数 组	412
BB. 20 TypeLibFunc 属性	386	BC. 2.9 接 口	413
BB. 21 TypeLibType 属性	386	BC. 2.10 枚 举	414
BB. 22 TypeLibVar 属性	386	BC. 2.11 委 派	414
BB. 23 枚举支持	387	BC. 2.12 属 性	415
		BC. 3 不安全代码的语法扩展	416
		BC. 3.1 不安全上下文	416

C 篇 C# 编程入门

C1 章 C# 编程的基本知识	420	C1. 3.3 框架类的方法和接口	440
C1. 1 .NET 框架基本类	420	C1. 4 格式化	441
C1. 2 对象的公共方法	421	C1. 4.1 数字格式化	441
C1. 2.1 ToString()	421	C1. 4.2 时间日期格式	444
C1. 2.2 Equals()	423	C1. 5 流对象	445
C1. 2.3 哈希表和 GetHashCode()	425	C1. 5.1 基本流操作	445
C1. 3 System. Array 和数组类	427	C1. 5.2 字节类型	446
C1. 3.1 排序和查找	427	C1. 5.3 读写器	446
C1. 3.2 ICloneable	438	C1. 5.4 格式器	448
		C1. 5.5 流	450

C1.6 调试类	451	C2.6.4 Autos 窗口	478
C1.6.1 条件方法	451	C2.6.5 This 窗口	478
C1.6.2 Debug 和 Trace 类	452	C2.6.6 调用栈窗口(Call Stack)	478
C1.6.3 使用切换控制 Debug 和 Trace	455	C2.6.7 断点窗口(Breakpoints)	479
C1.7 小 结	460	C2.6.8 反汇编窗口(Disassembly)	479
C2 章 C# 的编程和编译环境	461	C2.6.9 内存窗口(Memory)	479
C2.1 命令行编译	461	C2.7 定制 VS.NET	480
C2.2 VS.NET IDE 概述	463	C2.7.1 自定义风格	480
C2.2.1 Start 页	463	C2.7.2 过滤帮助	481
C2.2.2 标签式和自动隐藏窗口	464	C2.7.3 选项窗口	481
C2.2.3 解决方案浏览器(Solution Explorer)	464	C2.7.4 自定义工具条	483
C2.2.4 类视图(Class View)	465	C2.8 其它 VS.Net 特性	483
C2.2.5 性质窗口(Properties Window)	465	C2.8.1 宏浏览器(Macro Explorer)	484
C2.2.6 工具箱(Toolbox)	465	C2.8.2 插件管理器(Add - In Manager)	484
C2.2.7 服务器探查器(Server Explorer)	466	C2.8.3 HTML,Xml,CSS 编辑	485
C2.3 创建项目	467	C2.9 WinForm 编程	485
C2.3.1 项目和方案(Projects and Solutions)	467	C2.9.1 什么是 Win Forms	486
C2.3.2 向导和项目类型	468	C2.9.2 在 VS.NET 中创建 Win- Forms 应用	487
C2.4 编写代码	469	C2.9.3 其他特性	491
C2.4.1 代码窗口	469	C2.10 小 结	492
C2.4.2 代码折叠	471	C3 章 构件编程的基本概念	494
C2.4.3 导航语法(Intellisense) ..	471	C3.1 构件与接口	494
C2.4.4 对象浏览器(Object Browser)	473	C3.2 构件编程模式	497
C2.4.5 引用构件和控件	473	C3.3 C# 为设计构件提供的特征 ..	499
C2.4.6 引用外部控件	473	C3.3.1 性 质	499
C2.5 编译代码	474	C3.3.2 索引器	502
C2.5.1 输出窗口(Output)	474	C3.3.3 委派/事件	503
C2.5.2 任务列表(Task List)	474	C3.4 设计构件	507
C2.6 调试代码	476	C3.4.1 .NET Framework 构件	507
C2.6.1 命令窗口(Command)	477	C3.4.2 定制元数据	508
C2.6.2 查看窗口(Watch)	477	C3.4.3 性质浏览器接口	509
C2.6.3 本地窗口(Locals)	478		

C3.5 构件设计器	515	C5.5 线程同步	557
C3.5.1 访问设计器服务和基础结构	517	C5.6 线程的优先权	560
C3.5.2 为构件授予许可证	517	C5.7 小 结	560
C3.6 构件与型	519	C6章 网络编程	561
C3.6.1 抽象服务器 abstract server	520	C6.1 System.NET 类概貌	561
C3.6.2 适配器 adapter	520	C6.1.1 请求/回应模型	562
C3.6.3 观察器 observer	522	C6.1.2 HTTP/UDP/TCP/SOCKET	562
C3.6.4 桥 bridge	525	C6.1.3 TCP 和 UDP	563
C3.6.5 抽象工厂 abstract factory	527	C6.1.4 Sockets	564
C3.7 小 结	530	C6.2 网络资源交互	565
C4章 组装和清单	531	C6.3 特性深入观察	566
C4.1 .NET 框架下构件组装	531	C6.3.1 请求/回应模型的特性	566
C4.1.1 组装的组成	531	C6.3.2 HTTP 相关类的特性	567
C4.1.2 两类组装	532	C6.3.3 TCP 和 UDP 类的特性	569
C4.1.3 组装的好处	533	C6.4 小 结	572
C4.2 创建一个 DLL 组装	534	C7章 分布式编程	573
C4.2.1 库项目	534	C7.1 远程及代理对象	573
C4.2.2 客户端	536	C7.2 通 道	575
C4.3 使用 ILDasm 查看组装	538	C7.3 对象激活	576
C4.4 共享名	541	C7.4 租约生命周期	577
C4.5 全局共享库	541	C7.5 两个例子	579
C4.6 专用组装到共享组装的转换	541	C7.5.1 使用 TCP 通道的远程例子	579
C4.6.1 生成密钥对	541	C7.5.2 使用 HTTP 通道的远程例子	581
C4.6.2 对组装签名	542	C7.6 小 结	583
C4.6.3 放入全局库	543	C8章 ADO.Net	584
C4.7 替换密钥文件	543	C8.1 ADO 到 ADO.Net	584
C4.8 跨语言特性	544	C8.2 ADO 对象模型	585
C4.8.1 继 承	544	C8.3 DataSet	587
C4.8.2 调 试	547	C8.3.1 受管提供器	588
C4.9 小 结	549	C8.3.2 用 SQL 受管提供器读数据库	589
C5章 多线程编程	550	C8.3.3 读入 Access 数据库	591
C5.1 线程的基本概念	550	C8.3.4 得到列的信息	591
C5.2 创建线程	551	C8.3.5 利用 DataGrid 自动显示	593
C5.3 线程状态	553		
C5.4 线程结束	555		

C8.3.6 修改数据	594	C9.4.5 属性描述	638
C8.3.7 RowStates	597	C9.4.6 编译文档	640
C8.3.8 使用 DataReader	598	C9.5 小 结	640
C8.3.9 用 .Net 连接数据库	599	C10 章 与 COM 和本地资源交互	641
C8.4 与 XML 交互	602	C10.1 COM 与 .NET	641
C8.4.1 写 Xml 文件	603	C10.1.1 类型库中生成元信息 ...	641
C8.4.2 读 XML 文件	604	C10.1.2 束定和调用 COM 构件	644
C8.4.3 XmlDataDocument	604		644
C8.5 小 结	606	C10.1.3 其它接口访问和发现动	644
C9 章 与 XML 交互	607	态类型	644
C9.1 .NET 的 System.Xml	607	C10.1.4 COM 对象的晚束定	645
C9.2 XML 读写器	608	C10.1.5 ActiveX 控件	647
C9.2.1 抽象基类	609	C10.1.6 理解 COM 线程模型和套间	648
C9.2.2 XmlReader	610		648
C9.2.3 XmlTextReader 和其它	616	C10.1.7 COM 中调用 .NET 构件	649
Reader	616		649
C9.2.4 XmlWriter 和 XmlTextWriter	623	C10.2 CLR 对 COM 的改进	651
.....	623	C10.2.1 COM 中的类型信息限制	651
C9.3 Xml DOM 实现	624		651
C9.3.1 遍历和 XmlNavigator ...	626	C10.2.2 CLR 新体验	653
C9.3.2 DocumentNavigator	628	C10.3 本地资源交互	662
C9.3.3 转化 XslTransform	629	C10.3.1 不安全和非受管代码 ...	662
C9.4 XML 注释	630	C10.3.2 平台请求服务	663
C9.4.1 元素简单描述	630	C10.3.3 其它细节	663
C9.4.2 备注和表单	632	C10.4 小 结	664
C9.4.3 参数描述	635	参 考 文 献	666
C9.4.4 实例提供	636		

A 篇 C# 产生的背景

程序设计语言(简称编程语言)是表示软件的工具。一般说来,编程语言总力图做成独立的表示系统。人们学习了它的语法规则,理解了各种机制就可以用它来编程上机了。只有在编程用到比较深层、底层的机制,涉及到执行语义出现微妙差异才去弄清楚与其相关的环境特征,Pascal,C++,Ada,Java莫不如此。

C#语言有点特殊,类似于C。C虽是完全独立的编程语言,它和UNIX关系密切。搞清楚了UNIX对活用C语言有极大的好处。C#和Microsoft.Net的关系就如同C和UNIX一样,甚至更加依赖于.NET。

本篇以网络计算为背景介绍了微软公司为支持Web Services开发的框架,说明C#与.Net Framework的关系,以及它们如何联手解决网络计算的问题。

接着讨论了Windows下不同项目、不同语言软件互操作实现的核心技术:公共语言运行时CLR、公共语言运行规范CLS,以及MS的中间语言IL和C#语言源程序如何编译、连接、解释、即时编译以及最后执行的全过程。为了完整性,我们把IL指令的格式及数据表示法当作A2章的附录置于该章的最后,感兴趣的读者可以学习。

最后介绍作为新一代语言的C#除了满足.NET网络设计需求之外,如何纳入编程语言新技术和软件开发人员的经验,以C#特点逐一说明。

有了以上导引,读者在阅读C#编程语言规范(即参考手册,本书作为B篇)时就清晰得多。