



软件工程师丛书

XML Web Services 高级编程范例

胡海璐 彭接文 胡智宇 等编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 提 要

XML Web Services 是为其他应用程序提供数据和服务的应用程序逻辑单元。应用程序通过常见的 Web 协议和数据格式(如 HTTP、XML 和 SOAP)访问 XML Web Services，而无需考虑 XML Web Services 的实现细节。XML Web Services 将基于组件的开发的最佳方面和 Web 结合在一起，是 Microsoft .NET 编程模型的基础。

本书详细讨论了 XML 技术的起源、特点、语法规则、应用领域及其主要特性，并通过大量的实例演示了在 .NET 平台下应用 XML 进行程序设计的方法。介绍了使用 .NET 框架和新的 Visual Studio.NET 工具创建、测试、引用、部署 Web 服务的方法。并结合留言板、条形图、BBS、SQL 服务器远程监控等具体的 Web 应用全面深入地介绍了在 .NET 平台下进行 Web 服务编程的技巧。最后，研究了在 .NET 框架下创建安全的 XML Web Services 的方法。

本书适合所有在 .NET 下开发 XML Web Services 应用系统的软件工程师。通过阅读本书，读者能够理解并灵活运用这些知识点，迅速掌握这些技术，从而能够开发出功能强大的 Window/Web 常规应用程序和数据访问应用程序。

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。

图书在版编目(CIP)数据

XML Web Services 高级编程范例 / 胡海璐等编著 . —北京：电子工业出版社，2003.2

(软件工程师丛书)

ISBN 7-5053-8475-9

.X... 胡... .可扩充语言，XML - 程序设计 .TP312

中国版本图书馆 CIP 数据核字(2003)第 006119 号

责任编辑：寇国华

印 刷：北京市天竺颖华印刷厂

出版发行：电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销：各地新华书店

开 本：787×1092 1/16 印张：29.25 字数：687 千字 附光盘 1 张

版 次：2003 年 2 月第 1 版 2003 年 2 月第 1 次印刷

印 数：5000 册 定价：52.00 元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。联系电话：(010)68279077

出版说明

随着我国加入 WTO, 现代化建设也将以前所未有的步伐向前迈进。我们面临更大的挑战, 也面临更多的机遇。一个不争的事实是计算机的应用普及将更加深入, 将需要数量更多、水平更高的软件工程师。

我国的软件工程师队伍已有了长足的发展, 软件开发水平已有了长足的进步。作为中国人, 我们期望中国软件业走自主创新之路, 在世界上的地位越来越高。作为出版工作者, 为发展我国的软件事业尽最大努力, 是我们义不容辞的责任, 这正是我们于 1999 年底推出《软件工程师》丛书的初衷。

目前这套丛书已出版了 40 多种。从市场销售和读者反馈的情况看, 这套丛书已经得到了读者的首肯和厚爱, 这也是对我们下一步工作的激励。

可以说, 计算机应用系统的多样化、规模化和复杂化对软件工程师提出了更高的要求, 同时也为软件工程师提供了更多的施展个人才华的机会。

针对这种形势, 我们正在扩充《软件工程师》丛书的选题范围, 进一步界定这套丛书的特色, 并把丛书按如下类型整合。

一是开发类, 通过大量实例说明如何使用各种流行的高级语言、工具类软件开发不同的应用系统, 说明开发思想、开发过程、难点及其解决方案。为了适应我国软件工程师开发综合软件系统的需求, 我们把包含编程功能在内的高级应用软件的开发应用也纳入到丛书中。

二是技巧类, 通过大量实例说明在不同应用系统开发过程中, 有关缩短开发周期、提高开发质量、解决开发中的疑难问题的各种技巧。

三是技术类, 介绍软件开发的有关理论和技术, 以及在实际中的应用, 如系统分析与系统设计、软件测试和系统安全等。

四是手册类, 即每个软件工程师必备的案头书。

在新的一年里开始之际, 这套丛书从内容、开本、印刷及装帧等方面都将以全新的面貌与广大读者见面, 目的在于使其更受读者的欢迎, 每本书能容纳更多的信息。

我们以为软件工程师提供图书信息服务为宗旨, 坚持以图书质量为生命。我们希望《软件工程师》丛书能对读者有所帮助, 希望读者提出更多的宝贵建议和意见, 包括工作中遇到的技术难点、疑点和问题。希望更多的专家加入我们的作者行列, 推介自己的实践经验和累累硕果。我们的网址是 www.phei.com.cn, 请和我们联系。

为了我国软件业的更加美好的明天, 让我们共同努力。

电子工业出版社

前 言

Internet 应用程序在信息和服务的共享方面正发生着翻天覆地的变化。使用 Web Services 是一种使用现有、可行的 Internet 技术使多个应用程序共享信息的新途径。

Web Services 提供一个抽象层，隐藏了软件功能的实现细节。它通过松散耦合的服务程序完成软件功能共享，而不必考虑隐藏在这些服务程序背后的实现细节，从而为建立新一代应用程序提供了一个灵活而反应迅速的结构。

Web Services 提供了互操作性，而 XML 是使之得以实现的基础部分之一。XML 是 Visual Studio.NET 和 .NET 框架(Framework)等功能的核心。 .NET 框架将 XML 用于多种传输功能，从配置文件和源代码文档到 SOAP 和 Web Services，同时 ADO.NET 也使用 XML 作为传输机制。由于 .NET 框架广泛使用 XML，所以使用 XML 开发应用程序可获得很好的支持。

本书详细讨论了 XML 技术的起源、特点、语法规则、应用领域及其主要特性，并通过大量的实例演示了在 .NET 平台下应用 XML 进行程序设计的方法。介绍了使用 .NET 框架和新的 Visual Studio.NET 工具创建、测试、引用及部署 Web Services 的方法，并结合留言簿、条形码、BBS 以及 SQL 服务器远程监控等具体的 Web 应用全面深入地介绍了在 .NET 平台下进行 Web Services 编程的技巧。最后，研究了在 .NET 框架下创建安全的 XML Web Services 的方法。

全书共 13 章，各章的主要内容说明如下。

第 1 章“ .NET 框架及 VS.NET 简介 ”介绍了 .NET 平台、Visual Studio.Net 功能及其 IDE 界面。

第 2 章“ XML 概述 ”说明了 XML 的起源和目标、特点与优点、应用领域及前景。

第 3 章“ XML 技术的主要特性 ”介绍了标准格式的 XML 文档、文档对象模型、链接和查询，以及 XML 和数据库。

第 4 章“ 在 .NET 框架中使用 XML ”介绍了 System.xml 命名空间、XmlNode 及其派生类编程实例、XmlWriter 及其派生类编程，以及 XmlReader。

第 5 章“ .NET 对 Web 应用的支持 ”介绍了 ASP.NET、Web Forms、ASP.NET 应用程序配置和 ASP.NET 程序编制。

第 6 章“ XML 在 .NET 框架中的应用 ”介绍了 XML 在配置应用程序、WSDL 和 SOAP 中的应用，以及关系数据和 ADO.NET 的 XML 集成，并提供了一个 XML 文档生成 Web 服务实例。

第 7 章“ 个性化 XML 实例编程 ”介绍了 CSS 文件功能及其用法和使用 CSS 样式表显示 XML，并提供了两个开发实例。

第 8 章“ XML Web Services ”介绍了 Web Services、在 VS.NET 中创建、测试和调试 XML Web Services、部署 XML Web Services 应用程序和 Web Services 的属性。

第 9 章~第 12 章分别提供了“留言簿”、“Web 方式商业图表绘制”、“完全基于 Web 的 BBS 编程”和“SQL Server 服务器远程监控”的编程实例，每个实例均说明了开发过程及技术要点分析，实用性很强。

第 13 章“VS.NET 与 XML Web Services 安全”全面说明了 XML Web Services 安全，包括代码访问安全、基于角色的安全和安全策略。

书中通过实例循序渐进地讲解 XML 和 Web Services 的语言知识点和编程技巧，每章都着重对其最重要和最新的技术进行了深入浅出的剖析，这些实例都能够为读者的实际编程起到很好的借鉴和指导意义，从而使读者牢牢地把握技术发展趋势，获得真正意义上的提高。本书适合对 .NET 下的 XML 和 Web Services 编程有应用需求或感兴趣的读者。通过阅读本书，读者能够理解并灵活运用这些知识点，迅速掌握这些技术，从而能够开发出功能强大的 Window/Web 常规应用程序和数据访问应用程序。

本书所附的光盘包括所涉及实例的所有源代码，本书的实例均使用 Visual C# 编写，所有实例均在 .NET 环境下调试实现。

本书由胡海璐、彭接文和胡智宇合作编写，其中第 1 章和第 8 章~第 13 章为胡海璐编写，第 2 章、第 4 章和第 7 章由彭接文编写，第 3 章、第 5 章~第 6 章由胡智宇编写。参与此书编写的人员还有：左震宇、季挺、木龙骨、郑竹子、阳国贵、温俊、陈文明、刘康珍、刘桂宾、张华栋、冷志成、林滢、麻尚春、刘建辉、胡水龙和吴玉云。

本书在写作中融合了作者多年的程序设计经验和对 XML 技术及 Web Services 的理解，力求做到言之有物，更好地体现实用性和操作性。由于作者的水平和经验有限，书中不足之处在所难免，恳请广大读者批评指正。

说明：

在 MSDN 中，Web 服务有时也写做 Web Services。在本书中，将 XML Web 服务写做 Web Services。

作者

2002 年 9 月

作者邮箱：

胡海璐：perilla@163.com 彭接文：pjiewen@21cn.com 胡智宇：fuyao0547@sina.com

目 录

第 1 章 .NET 框架及 VS.NET 简介.....	1
1.1 .NET 平台概述.....	2
1.1.1 .NET 平台产生背景.....	2
1.1.2 .NET 平台的组成.....	3
1.1.3 .NET 技术框架内核.....	3
1.1.4 .NET 主要特性.....	5
1.2 Visual Studio.Net 功能概述.....	5
1.2.1 VS .NET 的新特性.....	6
1.2.2 VS.NET 的版本及系统要求.....	7
1.3 VS.NET IDE 界面.....	10
1.3.1 主界面.....	11
1.3.2 起始页.....	12
1.3.3 服务器资源管理器.....	13
1.3.4 菜单与工具栏.....	14
1.3.5 属性窗口.....	14
1.3.6 代码编辑器.....	17
1.3.7 “动态帮助”窗口.....	21
1.3.8 文件区.....	21
1.3.9 工具箱.....	24
1.3.10 “Web 浏览器”窗口.....	24
1.3.11 命令窗口.....	25
1.3.12 “任务列表”窗口.....	26
1.3.13 安装工具.....	27
1.4 小结.....	27
第 2 章 XML 概述.....	29
2.1 XML 的起源和目标.....	30
2.1.1 标记语言.....	30
2.1.2 XML 起源.....	31
2.1.3 XML 目标.....	31
2.1.4 XML 定义.....	32

2.1.5	XML 与 HTML、SGML 之间的关系.....	33
2.1.6	XML 家族.....	33
2.2	XML 的特点与优点.....	34
2.2.1	特点.....	34
2.2.2	优点.....	35
2.3	XML 文档结构及语法.....	36
2.3.1	XML 文档结构.....	36
2.3.2	XML 文档的逻辑结构.....	36
2.3.3	XML 文档的实体结构.....	39
2.4	XML 与企业之间的关系.....	43
2.4.1	增强企业网站的职能.....	44
2.4.2	简化企业的资料管理.....	44
2.4.3	及时掌握消费者的反馈.....	44
2.4.4	企业伙伴信息共享.....	44
2.4.5	危机处理.....	45
2.5	XML 的应用领域.....	45
2.5.1	数据交换.....	45
2.5.2	内容管理.....	45
2.5.3	设计与特定领域有关的标记语言.....	46
2.5.4	文件保存.....	46
2.5.5	Web 应用.....	46
2.5.6	Web 服务.....	47
2.5.7	应用配置.....	47
2.6	XML 前景.....	48
第 3 章	XML 技术的主要特性.....	49
3.1	标准格式的 XML 文档.....	50
3.2	文档类型定义.....	55
3.2.1	定义 DTD.....	56
3.2.2	DTD 的基本结构.....	56
3.2.3	DTD 和 XML Schema 的比较.....	61
3.3	使用 XML Schema 验证 XML.....	62
3.4	文档对象模型.....	69
3.4.1	概述.....	69
3.4.2	实例编程.....	72
3.5	链接和查询.....	74
3.5.1	XPath 规范.....	74
3.5.2	XLink 规范.....	77
3.5.3	XPointer 规范.....	78

3.5.4	VS.NET 中的 XPath 实现	79
3.6	通过 XSLT 转换 XML 文档	79
3.7	XML 和数据库	82
3.7.1	XML 与数据库的比较	82
3.7.2	XML 与数据集的相互转换	83
第 4 章	在 .NET 框架中使用 XML	85
4.1	System.xml 命名空间	86
4.1.1	标准	86
4.1.2	命名空间层次结构	86
4.2	XmlNode 及其派生类编程实例	89
4.2.1	XmlNode 节点管理实例	89
4.2.2	使用 XmlDocument 类实现商场商品管理实例	99
4.3	XmlWriter 及其派生类编程	112
4.3.1	常用方法	113
4.3.2	编程基本步骤	118
4.3.3	程序说明及运行结果	119
4.3.4	实际编程	119
4.3.5	技术要点分析	123
4.4	XmlReader	123
4.4.1	概述	123
4.4.2	常用方法	124
4.4.3	编程基本步骤	128
4.4.4	程序说明及运行结果	128
4.4.5	实际编程	128
4.4.6	技术要点分析	133
4.5	小结	133
第 5 章	.NET 对 Web 应用的支持	135
5.1	ASP.NET 概述	136
5.1.1	ASP.NET 优点	136
5.1.2	ASP.NET 应用程序组成	138
5.2	Web Forms 概述	139
5.3	ASP.NET 应用程序的命名空间和常用类	141
5.3.1	System.Web.UI 命名空间	141
5.3.2	System.Web.UI.HtmlControls 命名空间	142
5.3.3	System.Web.UI.WebControls 命名空间	143
5.4	ASP.NET 应用程序的配置	143
5.5	ASP.NET 程序编制	144

5.5.1	编写程序	144
5.5.2	ASP.NET Web 控件编程	150
5.6	小结	163
第 6 章	XML 在 .NET 框架中的应用	165
6.1	XML 技术在配置应用程序中的应用	166
6.1.1	web.config 配置文件	166
6.1.2	web.config 文件格式	167
6.1.3	web.config 文件结构	168
6.2	使用 XML 序列化 ASP.NET 编写的类	170
6.2.1	XML 序列化	170
6.2.2	XML 序列化实例	171
6.3	XML 在 WSDL 和 SOAP 中的应用	176
6.4	关系数据和 ADO.NET 的 XML 集成	176
6.4.1	ADO.NET 与 XML 的关系	176
6.4.2	DataSet 与 XmlDataDocument 同步	177
6.5	XML 文档生成 Web 服务实例	178
6.5.1	基本步骤	178
6.5.2	程序运行结果	178
6.5.3	实际编程	178
6.5.4	技术要点	197
6.6	小结	199
第 7 章	个性化 XML 实例编程	201
7.1	CSS 文件功能及其用法	202
7.1.1	CSS 样式规则	202
7.1.2	CSS 样式规则用法	203
7.1.3	CSS 样式规则的优先级	205
7.1.4	CSS 样式规则属性	205
7.2	使用 CSS 样式表显示 XML	206
7.2.1	新建 CSS 样式表	207
7.2.2	在 CSS 样式表文件中添加样式规则	207
7.2.3	链接 CSS 样式表到 XML 文件	211
7.2.4	编程实例	212
7.3	XSL 文件功能及用法	216
7.3.1	XSL 样式表结构	217
7.3.2	XSL 样式表与 XML 文件的关系	218
7.3.3	XSL 样式表用法	219
7.4	使用 XSL 显示 XML 文件	220

7.4.1	编程基本步骤	220
7.4.2	程序说明及运行结果	220
7.4.3	实际编程	220
7.4.4	技术要点分析	224
7.5	小结	225
第 8 章	XML Web Services	227
8.1	Web 服务概述	228
8.1.1	Web 服务的过去与未来	228
8.1.2	Web 服务的特点	229
8.1.3	Web 服务的体系结构	230
8.1.4	Web 服务的适用范围	231
8.2	Web 服务协议	233
8.2.1	SOAP 协议	233
8.2.2	WSDL	242
8.2.3	UDDI	244
8.2.4	DISCO 协议	245
8.3	在 VS.NET 中创建 XML Web Services	246
8.3.1	编程基本步骤	246
8.3.2	程序说明及运行结果	246
8.3.3	实际编程	246
8.3.4	技术要点分析	249
8.4	测试和调试 Web 服务	251
8.4.1	测试	251
8.4.2	调试	252
8.5	使用 XML Web Services	252
8.5.1	编程基本步骤	253
8.5.2	程序说明及运行结果	253
8.5.3	实际编程	253
8.5.4	技术要点分析	257
8.6	XML Web Services 的配置及部署	258
8.6.1	编译 XML Web Services	258
8.6.2	配置 XML Web Services	258
8.6.3	将 XML Web Services 部署到非开发服务器的服务器	259
8.7	部署 XML Web Services 应用程序	265
8.8	Web 服务的属性	266
8.8.1	WebService 属性	267
8.8.2	WebMethod 属性	268
8.9	小结	271

第 9 章 XML.NET 留言簿实例编程	273
9.1 留言簿功能分析	274
9.2 编程步骤	274
9.3 运行结果	274
9.3.1 我要留言	274
9.3.2 全部留言	275
9.3.3 留言管理	277
9.4 实际编程	278
9.4.1 新建工程	278
9.4.2 新建 XML 文件	278
9.4.3 在留言簿中添加记录	281
9.4.4 浏览留言簿	289
9.4.5 删除留言簿中的记录	297
9.4.6 留言簿 Web 应用部署	308
9.5 技术要点分析	311
9.5.1 XML 文件与 DataSet 之间的转换	311
9.5.2 使用 DataGrid 控件	312
9.6 小结	313
第 10 章 Web 方式商业图表绘制实例	315
10.1 技术难点	316
10.2 编程步骤	316
10.3 运行结果	317
10.4 新建工程	318
10.4.1 添加 ASP.NET 页面	319
10.4.2 Web 页面设计	319
10.5 读取并显示 Excel 表格中数据	321
10.5.1 访问本地 Excel 表格数据	322
10.5.2 在 Web 页面中显示数据表格	326
10.5.3 实现代码	327
10.6 会话状态变量传递	330
10.6.1 会话状态概述	330
10.6.2 会话状态变量使用	332
10.7 在 Web 页面中绘制条形图	332
10.7.1 图像处理基本步骤	333
10.7.2 实现代码	335
10.8 技术要点分析	338
10.8.1 使用 ADO.NET 访问 Excel 表格数据	338

10.8.2	使用会话状态传递变量	339
10.8.3	在 Web 页面中绘制图像	339
10.9	小结	340
第 11 章	完全基于 Web 的 BBS 编程实例	341
11.1	概述	342
11.2	主要步骤	343
11.3	运行结果	343
11.4	数据库结构设计	351
11.5	创建 SQL Server 访问 Web 服务	352
11.5.1	新建工程	352
11.5.2	添加自定义类 DataControl	352
11.5.3	添加 Web 服务方法	355
11.6	创建 BBS Web 应用	355
11.6.1	新建工程	355
11.6.2	添加 Web 引用	356
11.6.3	登录 BBS	356
11.6.4	新成员注册	360
11.6.5	个人资料修改	364
11.6.6	版面选择	367
11.6.7	浏览讨论主题	369
11.6.8	查看回帖	374
11.6.9	添加回帖	379
11.6.10	版务管理	381
11.6.11	成员管理	385
11.7	小结	388
第 12 章	SQL Server 服务器远程监控实例	389
12.1	概述	390
12.2	主要步骤	390
12.3	运行结果	391
12.4	创建 Web 服务	394
12.4.1	新建工程	394
12.4.2	使用 SQLDMO 从 SQL Server 中获取信息	395
12.5	创建 Web 应用	399
12.5.1	新建工程	399
12.5.2	添加 Web 引用	399
12.5.3	获取服务器名称列表及身份认证	399
12.5.4	获取数据库列表及其数据表、数据视图和存储过程	404

12.5.5	查看数据表详细内容	409
12.6	技术要点分析	414
12.6.1	使用 SQLDMO 对象	415
12.6.2	使用 ADO.NET	416
12.7	小结	418
第 13 章	VS.NET 与 XML Web Services 安全	419
13.1	XML Web Services 安全概述	420
13.1.1	攻击类型	420
13.1.2	保护 XML Web Services 安全的方法	421
13.1.3	Global XML Web Services 体系结构	424
13.1.4	实现 XML Web Services 安全的需求	425
13.2	代码访问安全	426
13.2.1	目的	426
13.2.2	功能	427
13.2.3	基本概念	427
13.2.4	工作原理	428
13.2.5	常用方法	428
13.3	基于角色的安全	429
13.3.1	应用领域	430
13.3.2	标识对象与用户对象及其编程实例	430
13.3.3	PrincipalPermission 对象及其编程实例	438
13.3.4	安全检查	442
13.4	安全策略	443
13.4.1	安全策略模型	443
13.4.2	安全策略级别	443
13.4.3	证据、代码组和权限集	444
13.4.4	权限	444
13.4.5	将代码组映射到权限集	446
13.4.6	管理和配置安全策略	446
13.4.7	设置基于角色的安全策略和原则	446
13.5	设计和开发安全的 XML Web Services	447
13.5.1	设计	447
13.5.2	开发	450
13.6	小结	452



第 1 章

.NET 框架及 VS.NET 简介



1.1 .NET 平台概述

.NET 为构建安全、可靠、高扩展性和高可用性的 Web 服务提供了从创建到配置、维护等一系列的应用程序模型和开发支持。

.NET 是微软成为 Web 服务软件供应商的战略方向，包括两个方面：公司重新编写其应用程序和服务以通过 Internet 进行通信和共享数据，并为开发商和集成商提供用于网络化解决方案的工具。

在开发等级上，.NET 框架在一个环境中集成了多种流行的程序语言，在其中可以非常流畅地共享类型，如同这些语言都具有同一个编译器一样。同时，.NET 在 Web 方面有很多创新，使得 Web 服务的开发和使用变得非常容易，包括使用可扩展标记语言(XML)格式化数据、使用 SOAP (Simple Object Access Protocol, 简易对象访问协议)在应用程序间传递数据，以及使用 UDDI (统一描述发现和集成协议)为 Web 服务提供目录等。

1.1.1 .NET 平台产生背景

虽然人们潮水般地涌向 Internet，然而理解和应用软件的方式却一直停留在使用 WWW 浏览、E-mail 信函和 FTP 下载等基本服务上，距离“5 个 Any” (任何人、任何地方、任何时间、任何设备和任何手段)的 Internet 梦想还十分遥远。如果沿着传统客户机/服务器模式走下去，即使能够创造出足够强大且安全的中间件，在现有网络架构下的互连集成也将是举步维艰，代价高昂。

2000 年 10 月，微软提出“.NET”战略构想，旨在建立一个无缝连接各个系统并充分注重用户体验的 Internet 系统，为用户创造一种安全无障碍地在 Internet 上连接商务的运作方式。.NET 的策略是将 Internet 本身作为构建新一代操作系统的基础，对 Internet 和操作系统的设计思想进行合理延伸。这样，开发人员将能够创建出摆脱设备硬件束缚的应用程序，轻松实现互连。.NET 平台包括一个相当广泛的产品家族，其构建于 XML 和 Internet 产业标准之上，为用户提供 XML Web Services 的开发、管理、应用和体验。

.NET 将重点从连接到 Internet 的单一网站或设备转移到计算机、设备和服务群组上，使其通力合作，提供更广泛且更丰富的解决方案。用户能够控制信息的传送方式、时间和内容。计算机、设备和服务相辅相成，从而提供丰富的服务。.NET 允许企业使用开放的 Internet 协议将不同的应用程序和系统，甚至是运行在不同平台上的应用程序和系统联系起来，无缝地集成为一个完整的系统。数据和软件功能能够无缝地从计算机转移到任何相关设备上，极大地增加了获得和生成数据的能力。

.NET 的出现是软件业的一次革命，其新特性解决了当今软件开发中的如下问题。

(1) 应用程序的互操作性(Interoperability)、集成性(Integration)和可扩展性(Extensibility)。利用原有的开发工具很难实现上述特性，即使实现也要耗费高昂的代价。.NET 依靠 XML 技术标准消除了数据共享和应用集成的障碍。

(2) 在现有的环境中，用户不能在程序之间方便地共享数据，无法对要访问的数据直接操作。而.NET 对 XML 的支持，使得程序间的数据交换变得简单易行。

(3) .COM 公司和 Web 站点很难为用户提供有价值且可操作的数据，一个个信息孤岛之

间程序和服务很难很好地合作，而 .NET 技术可以整合所有 Web 上的程序和服务。

(4) 用户的个人隐私和安全成为 Web 使用的一大障碍。 .NET My Service 提供了一整套服务，增加了用户信息的可管理性和安全性。

1.1.2 .NET 平台的组成

.NET 平台由以下 5 个部分组成。

(1) .NET 服务框架和开发工具。一个坚实、可靠和强大的编程框架是必须的，Microsoft .NET 框架便充当了这一角色。方便易用的 RAD 也是非常有必要的，2002 年 3 月发布的 VS.NET 就是一个优秀的开发工具。

(2) .NET 服务器。能够提供广泛聚合和继承 XML Web Services 的服务器是搭建 Microsoft .NET 平台的后端基础，广泛支持 XML 的数据库、操作系统及商务交易系统将为 Microsoft .NET 平台提供一个坚实的后端基础。

(3) .NET 基础服务。密码认证、日历、文件存储和用户信息等基础服务是必不可少的。微软正在着力建设的 Microsoft .NET My Services 等基础性服务平台将为用户提供安全可靠的基础服务。

(4) .NET 终端设备。广泛连接 Internet 并体验 XML Web Services 的终端设备是实现的前端基础，PC、PDA 以及各种嵌入式设备将在这个广阔的天地里发挥作用。

(5) .NET 用户体验。快捷、高效、方便、易用且能满足人们各种需求的用户体验是 Microsoft .NET 的终极目的，也是 Microsoft .NET 的价值实现。将通信和计算两大 IT 主题通过 XML Web Services 完美地演化为人们日益增长的体验需求是 Microsoft .NET 平台中最具创造性，也是最具价值的部分。

在这 5 个组成部分中，“.NET 服务框架和开发工具”是整个 .NET 平台赖以生存和发展的核心，其他 4 个部分则是紧紧围绕着 .NET 框架来进行组织整合。一个健壮、高效和安全的框架，一个经过检验的坚实的结构，对于现代软件开发是非常关键的。用户无需再从底层开始完全靠自己来实现规则和数据处理，所要做的只是按照自己的商业逻辑组织服务应用。Microsoft .NET 框架为基于下一代 Internet 网络的 XML Web Services 及应用程序提供了一个托管的、安全和高效的执行环境。

1.1.3 .NET 技术框架内核

.NET 技术框架实现了从开发语言、代码编译、组件配置、程序运行和对象交互等各个层面的创新，既是微软对几十年来软件开发的经验整合，也是面向未来基于 Web 服务应用的创新开拓。 .NET 技术框架从底层的系统服务开始，提供了一个托管的代码执行环境——通用语言运行时。在此之上，是为组件开发和运行提供的从基础编程、数据访问和程序装配等各种服务的一个坚实、强大和全面可重用类的集合。 .NET 框架以组件的形式包装这些可重用类，为程序开发提供 3 种基本编程模板：基于 ASP.NET 的 Web 表单应用、基于 ASP.NET 的 Web 服务和基于传统 GUI 交互的 Windows 应用。图 1-1 所示是 .NET 技术框架的总体描述。

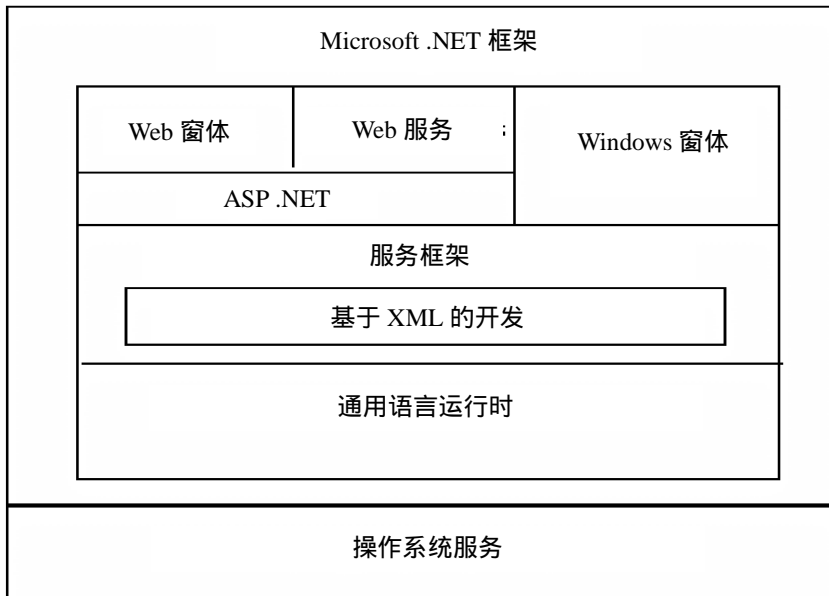


图 1-1 .NET 技术框架的总体描述

1. 通用语言运行时(Common Language Runtime , CLR)

通用语言运行时为 .NET 应用程序提供了一个托管的代码执行环境,托管是指将原来由程序员或操作系统做的工作剥离出来交由通用语言运行时处理,这样的工作通常包括内存管理、即时编译、组件自描述、安全管理和代码验证,以及其他一些系统服务。通用语言运行时是一个技术规范,无论程序使用什么语言编写,只要能编译成中间语言(IL),即可在其支持下运行,使得 .NET 应用程序得以独立于语言。通用语言运行时的设计目的是直接在应用程序运行环境中为现代基于组件的编程提供第 1 步的支持。它直接支持属性、事件、对象、继承性、多态性和接口。

自动垃圾收集器负责 .NET 应用程序运行时的内存分配、释放和对对象布局等内存管理问题,彻底解决了多年来困扰程序员的内存泄漏问题,使程序员能够专注于应用的商业逻辑,并且大大增强了应用程序的稳定性。

即时编译器在运行时将中间语言以调用的对象方法为单位动态编译成本地二进制代码。中间语言是在 .NET 平台下编译器输出的 PE 文件的语言,为 .NET 平台提供多语言支持,允许开发人员任意选用 20 多种不同编程语言中的任一种。

元数据是一个内嵌于 PE 文件的表的集合,描述了代码中的数据类型等一些通用语言运行时在代码执行中需要的信息。它使得 .NET 应用程序代码具备自描述特性且提供了类型安全保障,从而改变了需要额外类型库或接口定义语言(IDL)的方式。

通用语言运行时根据托管组件的来源(如 Internet、局域网或本地机)等因素判定其信任度,然后根据其信任度来限定其执行某些敏感操作的权限,如读取文件或修改注册表等。基于角色的认证为 Internet 上分布式组件的执行提供了安全保证,借助通用类型系统对代码类型进行严格的安全检查避免了不同组件之间可能存在的类型不匹配的问题。通用语言运行时不仅规定代码访问的安全,还规定基于角色的安全。通过代码访问安全机制,开发