



深入解析ATL (第2版)

ATL Internals Second Edition

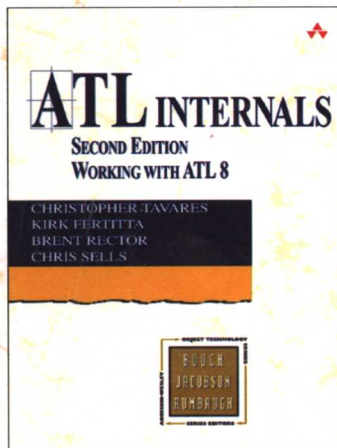
Working with ATL 8

[美] Christopher Tavares Kirk Fertitta 著
Brent Rector Chris Sells

赖仪灵 曹雨田 译



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



深入解析 ATL (第2版)

ATL Internals Second Edition

[美]	Christopher Tavares	Kirk Fertitta	著
	Brent Rector	Chris Sells	
	赖仪灵	曹雨田	译

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书主要介绍了 ATL 技术的原理、内部实现和应用技巧,由当今 4 位顶尖的 Windows 技术专家联合撰写。全书内容丰富,深入浅出,主要涵盖了 ATL 内部架构和实现方法、运用向导简化 ATL 开发、C++/COM/ATL 中字符串的使用技巧、理解并正确实现 IUnknown、充分利用 ATL 提供的 Smart Types、撰写能够暴露 COM 服务器上 COM 对象的粘合代码、利用 Canned Interface 实现来支持对象持久化/COM 群集/枚举、利用 ATL Windows 类建构独立的应用程序和 UI 组件等的技术内容。本书适合于广大 Windows 开发人员阅读参考,是 ATL 开发人员的必备权威参考书籍。

Authorized translation from the English language edition, entitled ATL INTERNALS: WORKING WITH ATL 8, SECOND EDITION, 978-0-321-15962-5 BY CHRISTOPHER, TAVARES; KIRK, FERTITTA; BRENT, RECTOR; CHRIS, SELLS, published by Pearson Education, Inc, publishing as Addison Wesley Professional, Copyright ©2006 Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY Copyright ©2007.

本书简体中文版由电子工业出版社和 Pearson Education 培生教育出版亚洲有限公司合作出版。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书简体中文版贴有 Pearson Education 培生教育出版集团激光防伪标签,无标签者不得销售。

版权贸易合同登记 图字: 01-2006-5604

图书在版编目(CIP)数据

深入解析 ATL: 第 2 版 / (美)塔瓦瑞斯(Tavares,C.)等著;赖仪灵,曹雨田译. —北京:电子工业出版社, 2007.11

书名原文: ATL Internals: Working with ATL 8

ISBN 978-7-121-04985-9

I. 深… II. ①塔…②赖…③曹… III. 软件开发 IV. TP311.52

中国版本图书馆 CIP 数据核字(2007)第 138340 号

责任编辑:王继花

印 刷:北京市铁成印刷厂

装 订:三河市皇庄路通装订厂

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×980 1/16 印张:55.5 字数:1212 千字

印 次:2007 年 11 月第 1 次印刷

定 价:89.90 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。服务热线:(010) 88258888。

Chris Tavares

感谢 Wendy 一直以来给我的信任。

Kirk Fertitta

感谢 Mom 和 Ed 一直以来给我的信任和鼓励。

Brent Rector

感谢 Lisa 在过去 30 年以来给我的所有鼓励。

Carly 和 Sean，他们并不为他们的父亲写书而感到惊奇，这就是他们所知道的一切。

Chris Sells

本书是专为 .NET 遗留下来的本地 Windows 程序员而写，虽然我希望 .NET 将来能让您受益，但现在，本书应该适合您！

译者序

从本书的第1版面世以来，它一直就是 ATL 开发人员不可或缺的书籍。经过近 10 的发展，ATL 本身也从 3.0 发展到了 8.0 版本。是的，我和大家一样期待了近 10 年的时间，终于盼来了《ATL Internals》第 2 版的出版。

近年来，随着 .NET 的快速发展，越来越多的人开始关心 C++ 语言的发展等问题。事实上，我们应该相信 C++ 从其发展至今，已经得到了明确的认可。C++ 凭借其功能强大、可控性强等特性占据着一定的、非常重要的应用领域。另一个方面，COM 技术的地位似乎也受到了冲击。但是，COM 仍然是 Windows 中最重要的跨语言、二进制级别的源代码共享技术之一。作为 Windows 平台的一名好的开发人员，熟悉这项技术、了解该技术发展的流脉也是非常重要的。

活动模板库（ATL）是微软提供的致力于改善进行 COM 相关开发过程的 C++ 类库。ATL 一开始的目标便是成为一套短小、灵活、功能强大的 C++ 类库，从 1998 年发表 ATL 3.0 开始，经过多年的发展它已经变得非常成熟。强大的 ATL 库经过近 10 年的改进已经发展到 8.0 版，现在它的功能更强大，结构更优化，使我们可以更方便地构建 COM 对象和组件。ATL 8.0 更是添加了 ATL Server 类集帮助构建网页应用程序扩展。

本书的组织结构与第 1 版类似，首先第 1 章是快速浏览 IDE 的向导功能，接着重点分析了 ATL 实现 COM 的内幕细节，这也是本书的核心，从第 7 章开始介绍了 ATL 的各种综合应用，第 13、14 章重点介绍了用于建立网页应用程序扩展的 ATL Server 类集。

幸运的是，本书的作者比第 1 版时还多了两位，他们都是 Windows 编程领域的大师，有着多年的实践经验。本书就是他们四人通力合作的结晶，书中不但非常详尽地解释了

深入解析 ATL（第 2 版）

ATL 8.0 实现 COM 功能的各种细节，同时也展示了 COM 应用中的各种漂亮技巧。

在翻译一本如此优秀的书籍时，我们尽力保持原文的含义和风格。在翻译的过程中，我们也深刻体会到该书的确是又一经典之作，它将带领我们深入理解 ATL 的设计和实现过程。

本书的翻译工作由曹雨田老师和我合作完成。该书的第 8 章至第 12 章由曹雨田老师翻译，其余内容均由我翻译，最终，所有内容均由我统稿整理。和其他的翻译工作一样，虽然我们努力尝试消除中文版的每一个错误，但因时间和水平有限，终究难免存在不足。所有的错误都应归咎于我的把关不严，责任不在其他。欢迎广大读者朋友在阅读该书的过程中指出我们的不足之处。我的邮件联系地址为 laiyiling@gmail.com。

赖仪灵

2007 年 8 月于上海

第 2 版序

Foreword to the Second Edition

哇噢！从我为《ATL Internals》第 1 版作序到现在，已经过了很长时间。通读旧的序言，又把我带回到了从前的记忆里。让人难以置信，时间已经过去了 8 年。为第 1 版作序后不久，我就转到了 Microsoft 的 Windows 组，一年之后离开了 Microsoft。几年前又回到了 Microsoft（Visual C++组），现在我管理着 Visual C++的几个开发组。其中之一是程序库(Library)组，ATL 也包含其中，我非常高兴再次从事与 ATL 相关的工作。Jan 和 Christian 也在一起继续开发，尽管 Nenad 把曾在第 1 版序言提及的 ATL 中的窗口类扩展成了单独的 WTL（Windows Template Library¹）。现在，WTL 是一个由 Nenad 管理的 Microsoft 开源项目。

ATL 已经以我无法预言的方式发生了改变，虽然对它的评价一直都是功过参半，但是它的持续发展并没有受到这些评价的影响。多年来，已经有多位大师级人物致力于 ATL 的开发。其中一些我非常熟悉，但对其他的却一无所知。

当我在第 1 版序言提及“访问 ATL 功能的一些新方法”时，指的就是 Attribute。这种技术被移植到 Visual Studio .NET 2002，但它绝没有如我们想象般真正地得到改进。ATL Attribute 在现在的发布版仍然可以工作，仍然非常强大，但没有扩展它们用法的计划。新版《ATL Internals》提供了很多更新，包括 Attribute，但也不假设我们会依靠这个特征。在本版中，对 ATL Server 也有一个很好的介绍，它提供了一种灵活、高效的方法来创建 Web 应用程序。如果性能需求很重要，您就应该选择使用 ATL Server。ATL 8 的其他改进包括更好的安全性、完全的 64 位支持、更好的伸缩性、调试改进、支持 C++/CLI 和托管

¹ 注: <http://wtl.sourceforge.net>

ATL 组件。

虽然 .NET 系统从 1998 年才开始其发展进程。但它已经使很多开发人员的编程发生了革命性的变化，并在未来几年将持续得到改进。但是，COM 编程（和 ATL）也依然非常有活力，对 Microsoft 内外的很多开发人员都非常重要。本书的第 2 版，与第 1 版一样，提供了您所需要的信息，使我们在这些技术上的投资也将获得最大的回报。

Jim Springfield

2006 年 4 月

第 1 版序

Foreword to the First Edition

第一次看到本书的标题时，我就告诉 Chris Sells，这听起来很像我一直想写的书。自从发布了 ATL 后，我们当中的一些人就一直在说：“应该写一本 ATL 如何工作的书。”在读了本书之后，我认为留给我写的内容已经很少了。实际上，这也是一种解脱。此时此刻我认为本书已经涵盖了 ATL 的大部分内容，为 ATL 内部工作原理提供了一个绝好的信息源。因此，Chris 叫我提供一些无法通过查看 ATL 源代码得到的信息。

ATL 简史

第一次接触模板是在 1995 年后期，当时我还是 MFC 组的一名开发人员。我的一位朋友正在评估 Visual C++ 产品对各种 STL 的实现。他向我谈起了很多关于模板的信息。我开始有些了解模板，但并没有深入研究。很快，VC 组分离出一个企业组以全力集中在 Visual C++ 4.2 企业版（我们的第一个 VC 企业产品）上。我被调离负责 VCEE 程序库的开发。那时，我们研究了几个不同的想法。当时 Microsoft 事务服务器（Microsoft Transaction Server）也刚刚启动，我们与他们谈论了很多关于 COM、事务、数据库、中间件业务对象的信息。很快，我们意识到需要一种更好的机制来从 C++ 中创建 COM 对象。当时，Jan Falkin 和 Christian Beaumont 在为我工作。Jan 致力于 ODBC 数据源的自动化接口，Christian 致力于基于模板访问 ODBC 数据源（现在的 OLEDB 消费者模板的前身）的方法。我致力于 COM 基础结构，至此，一切都在开始准备转向 COM。

开始，我只是研究 COM 和模板，但是，逐渐地对一些重要的概念产生了想法。最开始，我希望支持所有的线程模型，并希望只在必要时付出代价。对于聚合也是如此。我不

希望每个人都有借口不使用 ATL。（我不想听到：“ATL 是很酷，但是如果我自己实现，还可以节省更多的字节。”）因此，我们在做决定时，效率和灵活性比易用性更重要。其中产生的最重要的概念之一就是：用户编写的类不是真正实例化的类。这允许存在一些其他环境下根本不可能发生的优化。其他一些概念还包括接口的多继承、创建者函数和数据驱动的 COM 映射表。我们开始宣传这些概念并获得了很多好的反馈。一些人认为应该尽快让用户知道这些概念，因此，在 1996 年夏初，我们决定将它们发布到 Web 上，那就是 ATL 1.0。开始，这套程序库的工作名称是 Microsoft 企业类（Microsoft Enterprise Classes, MEC），但是，市场人员认为应该起一个更能反应我们工作的名字。由于我们的焦点集中在 COM，并且在当时的市场环境下，几乎所有的产品或技术名称都有“ActiveX”，我们选择了名称 ActiveX Template Library。我们获得了很多的反馈。在 1996 年夏末，我们发布了 ATL 1.1。到现在，Jan 和 Christian 已经直接在此 ATL 基础上开始工作。ATL 1.1 已经修复了很多问题，并支持很多新的特征，比如连接点、NT 服务、RGS 注册表支持和安全性。

在 ATL 1.1 之后，我们开始了 ATL 2.0 的开发。它的主要重点是创建 ActiveX 控件。Jan 和 Christian 在这点上做了很多工作，而我的重点仍然在核心代码（比如重写连接点，使它们变得更小）。Nenad Stefanovic 那时也加入了我们并开始着手 ATL 的窗口支持工作，以及在 VC 6.0 上实现复合控件支持。我们最初计划在 Web 上发布基于 VC4.2 的 ATL 2.0。但是，计划有变，我们决定与 VC 5.0（1996 年 12 月）一起发布 ATL 2.0，在 Visual C++ 5.0 的 Alpha 版本发布 ATL 2.1。ATL 2.0 和 2.1 之间的唯一区别就是修复了 Alpha、MIPS 和 PowerPC 的问题。我们同时还在 Web 上发布了 ATL 2.1，在 VC 4.2 上支持 AppWizard 和 ObjectWizard。在 ATL 3.0（那时称为 ATL 2.5）上工作数月后，Christian 和我离开了 ATL 一段时间，由 Jan 作为负责人接管 ATL。几个月之后，我们回来了，Christian 成为了 ATL 的负责人，而我开始为 Visual C++ 研究一些其他的東西，但我仍然不时地参与源代码的工作。

1998 年 7 月我们发布了 VC 6.0，现在开始了下一个发布版本的开发。期待着在 ATL 上看到更多更酷的新素材，以及一些访问 ATL 功能的新方法。我非常高兴地看到 ATL 在持续改进，同时还保持着最初生成短小、高效代码的目标。因此，建议您看看本书，学习一些新的技巧，深入理解 ATL 是如何工作的。

Jim Springfield

1998 年 10 月

前言

Preface

.NET 如同一枚重磅炸弹冲击了 Windows 程序员社区,它推翻了我们过去的编程基础。它非常贴切地满足了大多数 Web 应用程序和服务应用程序,以及与以前我们用 Visual Basic 和 MFC 开发的大多数业务相关的应用程序的需要。

但是,一些顽固的程序员在他们的“基地”声称除非最后用武力强迫,否则不会放弃本地代码。很多人在 C++代码上投入了很多年,不相信一些新奇的编译器开关,会使他们的本地代码被“托管”。不管是否会得到好处,这些人都会移植他们的代码。只要能说服他们把 ATL 3/Visual C++ 6 项目移植到 ATL 8 和 Visual Studio 2005 之上,在某种程度上,本书就适合他们。

另一类开发人员居住在“Windows 城市”的“市区”,不会受到暴风雨的袭击,他们仅仅知道正在发生暴风雨。他们发布的应用程序必须在 Windows 95 及更高版本之上快速、良好地运行,这样的环境中没有足够的内存或者 CPU 来运行 .NET 应用程序,或者即使满足这两点,也没有足够的带宽来下载 .NET 应用程序。他们还必须在服务器范围内最大可能地减少资源,充分利用可用的每个资源。他们没有豪华的 CPU、内存或者存储系统来满足垃圾回收、即时编译或者包含有不必要内容的庞大类库的需求,因为它们需要只有现代机器才能明确满足的环境。这些开发人员计算加载时间,执行速度,以在各种不利环境下直接访问平台。对他们而言,所使用的任何框架如果有不被使用的零开销特征时,都必须有严格的策略:自定义时需要最大的灵活性以及绝对的效率。对于这些开发人员,ATL 8 成为了 Windows 平台最后最好的本地框架。

对于客户,ATL 提供了窗口、COM 客户智能类型、广泛的 COM 控件和控件宿主、与 MFC 的集成(包含几个不再需要 MFC 的其他部分就可以使用的 MFC 类)和生成 Web 服务代理。对于服务器,ATL 提供全面的 COM 服务器和对象服务,广泛支持高吞吐量、

深入解析 ATL (第2版)

高并发的 Web 应用程序和服务。对于客户和服务端，ATL 都疯狂地使用了宏和模板来给予我们最大的灵活性和低开销，确保我们只为所使用的功能付出代价，通过源代码完全透明地向我们提供了这些类是如何把他们的功能映射到平台的。对于生产力，ATL 提供了一套向导用于开始和建立客户及服务器项目。

Attributes

在 ATL 7 和 Visual Studio 2003 上，为了提高生产力，ATL 组引入了属性化 ATL，它允许 ATL 程序员使用向 IDL 接口和 coclass（比如 `uuid Attribute`）添加元数据的相同技术来注解他们的代码。事实上，在 VS03，向导非常乐意向我们展示这种风格的代码，属性化选项被默认是打开的。但是，Attribute 并非一切都是美好的。在 .NET 和 IDL 中，Attribute 确实是属于编程模型的一部分，所有向下的方法都支持它们。在 ATL 中，Attribute 与等级宏、生成的基类、宏映射、注册脚本和 IDL 文件一样，远比编译器技巧复杂。

但是，Attribute 与宏不同，它不是透明的，不能很好地看到其工作过程。它包含了一个编译器开关来展示生成的代码，比如添加了什么基类，但这在 VS05 中有所退步。这已经导致了理解和调试方面的问题，对 Attribute 生成的代码问题也没有任何帮助。但这不表示 ATL 的其他部分的问题是开放的（也不意味着任何软件的问题都是开放的），但是当它引入了基类或宏中的问题时，ATL 总是有几种方式使我们可以替换有问题的功能。事实上，本书第 1 版的很大一部分代码都是解决问题，因为我们可以很容易地回避这些问题。

在 VS05 中的提示说明 Attribute 不再是 ATL 组的工作重点。比如，编译器开关提示的关于 Attribute 生成的信息更少，而不是更多。其实更明显的是在 VS05 中属性化选项不再被默认选中¹。基于这个原因，我在附录 D “属性化 ATL” 中涉及了 ATL Attribute 的基本原理。本书不会有更多的内容来介绍它。我们相信三心二意的 Attribute 将不会再使 ATL 8 程序员兴奋，因为他们可以选择本地框架。

读者

本书适用于转向 ATL 8 的 C++/COM 程序员，ATL 8 与 Visual Studio 2005 一起发布。ATL 的构建基于很多的假设，因此，要想成为一个 ATL 专家，不仅需要理解 ATL 是如何构建的，还需要知道原因。当然，为了理解 ATL 的原因，我们必须理解 ATL 的开发环境：COM。本书没有把所需要的 COM 知识压缩到一到两章的内容，本书假定读者已经理解了

¹ 注：生成 ATL Server Web 服务项目时是一种例外，属性化代码选项被选择并禁用，因此我们必须选择它。

COM, 本书所有的内容都是展示 ATL 的设计、使用以及内幕。Don Box 的《COM 本质论》(*Essential COM*, Addison-Wesley Professional, 1997) 很好地讲述了 COM 的知识。在深入 ATL 之前可以用它复习 COM 知识。

概要

除了第 1 章外, 本书的内容包含了 ATL 的最低层到最高层次的技术, 每章都建立在前一章的基础之上。第 1 章是 ATL 普通用法的简单概述, 以及相关用法的辅助向导。但是, 在第 1 章当需要对细节进行说明时, 我常常引用后续的章节来提供更深层次的讨论。

第 2 到第 5 章展示了 ATL 的核心。第 2 章“字符串和文本”涉及了在 C++、COM 和 ATL 中处理杂乱的字符串。第 3 章“ATL 智能类型”讨论了 ATL 的智能类型, 比如 CComPtr、CComQIPtr、CComBSTR 和 CComVariant。第 4 章“ATL 中的对象”讨论了 ATL 如何实现对象, 并集中讨论了实现 IUnknown 的大量可选方法。第 5 章“COM 服务器”讨论了 COM 服务器暴露 COM 对象所必须的代码。第 6 章“接口映射表”再次深入分析了 IUnknown 的实现, 这次集中在如何实现 QueryInterface, 本章展示了 Tear-off 接口和聚合等技术。第 7 章“ATL 中的永久性”、第 8 章“集合和枚举器”、第 9 章“连接点”分别讨论了 ATL 提供的封装接口实现以支持对象的永久性、COM 集合和枚举器、连接点。这些服务可用于组件, 可以提供用户界面, 也可以不提供。第 10 章“窗口”、第 11 章“ActiveX 控件”和第 12 章“控件包容”集中在如何建立独立应用程序和用户接口组件。这些章节分别涉及了 ATL 窗口类、控件和控件容器。最后, 第 13 章“你好, ATL Server”和第 14 章“ATL Server 内幕”涉及了 ATL Server, 它帮助我们在 IIS 上建立应用程序。第 13 章介绍了 ISAPI 和 ATL Server, 第 14 章深入研究了 ATL Server 的内幕。

真正使得 ATL 源代码难读的是大量的高级模板用法。附录 A “实例展示 C++ 模板”提供了一套例子说明模板是如何被使用和编译的。如果你在查看 ATL 源代码时对将派生类名称传递给基类感到奇怪, 你就会发现这个附录很有用。附录 B “ATL 头文件”提供了 ATL 头文件列表, 并提供了一些描述信息帮助我们快速找到感兴趣的 ATL 实现。如果你熟悉 ATL 3, 只对 ATL 8 所提供的新功能感兴趣, 那么附录 C “移植到 ATL 8”就是为你量身定做的一章。最后, 如果你需要了解 Attribute 的相关内容(以及为什么我们只在附录中用少量的篇幅介绍 Attribute), 可以阅读附录 D “属性化 ATL”。

约定

在编写本书时, 不仅需要展示图表和示例代码, 展示一些 ATL 内部的实现代码也是

非常必要的。通过这些实现代码，本书可以帮助你阅读 ATL 的源代码。为了帮助你区别作者编写的代码和 ATL 内部的实现代码，我们采用了如下的约定：

```
// 这样的代码由作者编写，是我们的输入示例。  
// 粗体显示的代码要求我们特别注意。  
CComBSTR bstr = OLESTR("Hello, World.");
```

```
// 具有灰色背景的代码属于 ATL 或者 Windows 内部的代码  
CComBSTR(LPCOLESTR pSrc) { m_str = ::SysAllocString(pSrc); }
```

由于 ATL 组并没有以书籍出版的形式编写他们的代码，通常我们必须对它们进行格式化或者简写。每一步都是为了保持原始代码的本质，但也必须认识到 ATL 源代码才是真正的判断依据。

源代码和进一步阅读

关于本书的其他信息，包括示例源代码，可以在 www.sellsbrothers.com/writing/atlbook 访问到。在站点上，你还可以找到联系信息，以报告错误或者提供反馈信息。

致谢

Acknowledgments

在此，作者要感谢很多人，感谢他们为本书作出的贡献。Chris Sells 想要感谢他的妻子 Melissa，以及他的儿子 John 和 Tom，感谢他们在无数个夜晚和周末陪伴他写作。Chris 还想要感谢 Brent Rector 让他研究本书的第 1 版，感谢 Kirk Fertitta 为本书更新到 ATL 7 做了大量的工作，感谢 Chris Tavares 经常回家后加班写作。

由于本书交稿时间的严重推迟，Brent 想要感谢他的妻子 Lisa 以及他的孩子 Carly 和 Sean。如果没有他们，他可能会连续几天不离开计算机。Brent 还想要感谢 Chris Sells 的智慧、耐心和英俊¹。

Kirk Fertitta 想要感谢读者，是他们使得写作成为一件有价值和有回报的工作。感谢 Chris Sells 让他参与本书的写作，感谢他对 ATL 的深刻理解和在写作过程中表现出来的洞察力。感谢 Brad Handa 和 Hugues Valois 为揭开 COM 和 ATL 的神秘面纱所付出的无数小时的努力工作。感谢 DevelopMentor 的 ATL 讨论列表的所有订阅者，他们在 ATL 的新特性上分享了很多自己的见解和经验。感谢 MusicMatch（现在属于 Yahoo!）的雇员和订阅者反馈了在巨大的商业应用程序中使用 ATL 的相关信息（使用某些属性化 ATL 特征的大量警告信息都是这些聪明、有耐心的开发人员所提供的），感谢 Addison-Wesley 的 Stephane Thomas，他的耐心使得本书的写作项目得以顺利开始。

Chris Tavares 想要感谢他长期受苦的妻子 Wendy，感谢她的理解、关爱和支持。深夜的一杯水和一碗冰淇淋让 Chris 保持了清晰的头脑，为本书的完成提供了很大的帮助。Chris 还要感谢他的儿子 Matthew，在父亲一连几天躲在办公室时，他没有任何的抱怨。Chris 还想要感谢 Chris Sells，使其有机会帮助第 2 版在 ATL 社区的出版。

¹ 注：你也许在猜想这部分内容是谁写的，他没有得到我的认可或者没有我英俊。

Chris、Kirk、Brent 和 Chris 要一起感谢其他的一些人，首先是评论者：Bill Craun、Johan Ericsson、Igor Tandetnik、Kim Gräsman、Jeff Galinovsky、Igor Tandetnik 和 Nenad Stefanovic。特别感谢 ATL 组的成员，包括 Christian Beaumont、Jim Springfield、Walter Sullivan 和 Mark Kramer，感谢他们一直忍受着不停的提问，而且还抽出时间回答。更特别的感谢送给 Don Box，因为他在 MSJ 上介绍 ATL 特征，这些内容影响到了 ATL 的有关课程，进而影响本书。感谢下面的评论者：Don Box、Keith Brown、Jon Flanders、Mike Francis、Kevin Jones、Stanley Lippman、Dharma Shukla、Jim Springfield、Jeff Stalls、Jaganathan Thangavelu 和 Jason Whittington。特别感谢 Dharma 给予的非常全面和具有教育性的评论。感谢 Fritz Onion 深入地研究了 ATL 的控件包容。感谢以前的一位学生，Valdan Vidakovic，是他鼓励 Chris 更深入的研究 HTML 控件。感谢 Tim Ewald、Jim Springfield 和 Don Box，感谢他们帮助开发了传递夹层（Forwarding Shim）技巧。感谢 ATL 和 DCOM 邮件列表的成员，特别是 Don Box、Tim Ewald、Charlie Kindel、Valery Pryamikov、Mark Ryland 和 Zane Thomas。同样，我们还要感谢 George Shepherd 对 ATL 服务器的初步研究，他甚至编写了 ATL 服务器章节的部分内容。最后（但并非最不重要），感谢 Addison-Wesley，特别是 Karen Gettman、Lori Lyons 和 Kim Boedigheimer，感谢他们提供给了我们实际想要的写作环境（尽管没有原来期望的快速或者简明……）。

关于作者

About the Author

Chris Tavares: 现在是 Microsoft 模式和实践组的一名软件开发工程师，他致力于帮助开发人员学习 Microsoft 平台上最好的开发方式。第一次接触电脑是在三年级，在一台 8080 机器上完成机器码的手动汇编。机器配置是内存 512 字节（是的，字节），一个十六进制的键区和 7 段的 LCD 显示器。从那时起，他就开始深入研究计算机和软件。

Kirk Fertitta: Pacific MindWorks 公司的 CTO，该公司专门提供电子测试和测量的工具和服务。Kirk 与他在 Pacific MindWorks 的团队一起广泛地致力于代码生成技术和 Visual Studio 的扩展性研究。他还是 Pluralsight 的 .NET/C# 讲师。

Brent Rector: Wise Owl 咨询公司的总裁和创始人，著名的演说家、顾问家和作家。擅长 .NET、ASP.NET、XML、COM+ 和 ATL。

Chris Sells: Connected Systems Division 的一名程序经理。他已经编写了多本著作，包括《Programming Windows Presentation Foundation》、《Windows Forms Programming in C#》和《ATL Internals》。在业余时间，Chris 举办各种会议，使他成为了 Microsoft 内部产品组讨论列表的“害虫”。关于 Chris 的更多信息以及他的各种项目，可以在 <http://www.sellsbrothers.com> 访问到。