



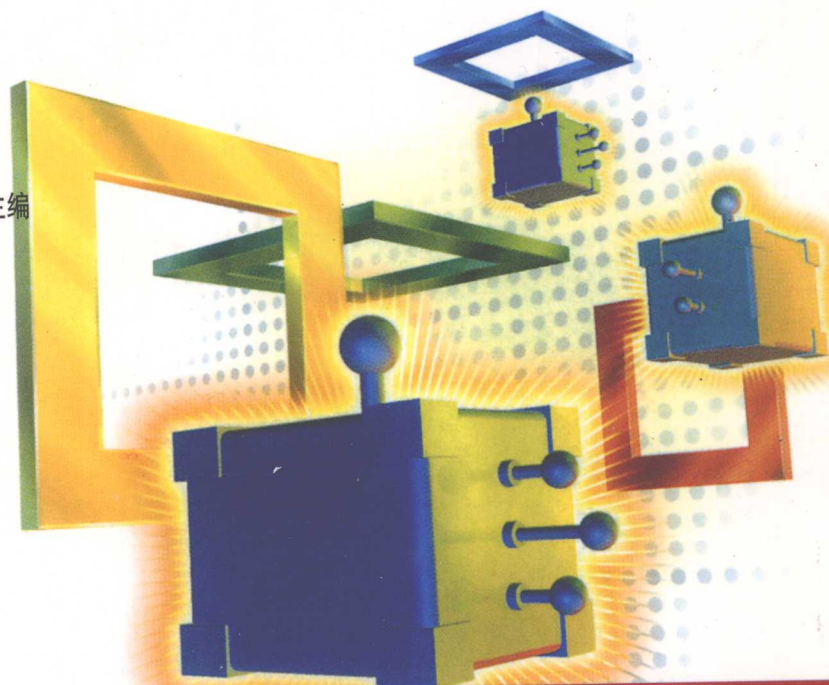
微软公司核心技术书库

Microsoft



COM+ Developer's Reference Library, Volume 4:  
Automation

(美) David Iseminger 主编  
李二勇 等译



COM+开发人员参考库之第4卷

# COM+ 自动化编程



机械工业出版社  
China Machine Press

PLC 应用案例解密



# COM+ 自动化编程

微软公司核心技术书库

# COM+自动化编程

COM+开发人员参考库之第4卷

(美) David Iseminger 主编

李二勇 等译

前导工作室 审校



机械工业出版社  
China Machine Press

本书介绍COM+中的自动化技术 (Automation), 主要内容包括: 自动化技术概述、提供ActiveX对象、访问ActiveX对象、标准对象及命名规则、分发接口及API函数、数据类型、结构和枚举、转换和操作函数、类型库及对象说明语言、类型说明接口、类型创建接口、错误处理接口、用户定义的数据类型以及API级格式化例程等。

本书内容翔实, 结构严谨, 并包含了大量的VB、VC范例代码, 读者可以结合代码前后的概念阐述, 更加深入地理解自动化技术的工作原理和编程思路。

David Iseminger: COM+ Developer's Reference Library, Volume 4:Automation.

Copyright © 2002 by Microsoft Corporation.

Original English language edition copyright © 2000 by Microsoft Corporation; portions copyright © 2000 by David Iseminger.

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A. All rights reserved.

本书中文简体字版由美国微软出版社授权机械工业出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

**本书版权登记号: 图字: 01-2000-4310**

### 图书在版编目 (CIP) 数据

COM+开发人员参考库 第4卷 COM+自动化编程/ (美) 艾塞明格 (Iseminger, D.) 主编; 李二勇等译. -北京: 机械工业出版社, 2002.1

(微软公司核心技术书库)

书名原文: COM+ Developer's Reference Library, Volume 4:Automation

ISBN 7-111-09362-3

I. C… II. ①艾… ②李… III. 软件接口, COM+-程序设计 IV. TP311.52

中国版本图书馆CIP数据核字 (2001) 第066120号

机械工业出版社 (北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 李云静

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2002年1月第1版第1次印刷

787mm × 1092mm 1/16 · 33印张

印数: 0 001-4 000册

定价: 59.00元 (全套300元)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

## 译者序

时下大概没有多少人不知道COM+技术。在热闹的BBS、纷飞的邮件和恬适的技术沙龙里充斥着COM这个字眼。有关COM/COM+的专业站点、专业刊物和探讨文章虽不能说多如牛毛，但形容为雨后春笋还是贴切的。

本书是“COM+开发人员参考库”套书的第4卷。顾名思义，这套书解释的是COM/COM+技术。本套书的重点并不是COM或COM+规范说明，而旨在为广大程序员提供这些领先技术的最可靠、最详细的编程参考。COM+继COM（组件对象模型）之后推出，但并不只是COM的升级。COM+以COM、DCOM（分布式组件对象模型）、MTS（微软事务服务器）和MSMQ（微软消息队列）为基础，继承了四者的关键特性；但在面向分布式企业应用的基础上，添加了许多新的服务。

在进行项目开发时，你会发现项目组的成员各怀绝技，有人擅长Visual C++，有人专攻Visual Basic，还有的人钟情于Delphi。为他们指定一门共同的开发语言显然是不可能的，那意味着开发周期的无限延长和人力资源的巨大浪费。所以，就有必要用一种办法，把他们各自的工作成果在项目中进行整合。这不仅仅是开发模块的分配、重组和联调，而是在二进制级别的兼容并蓄。

市场和用户需求的改变催促着软件疯狂地升级，而程序员越来越沉陷于无休止的升级、修补、扩展，此时COM+无疑给他们带来了新的希望。当然，我们可以采用DLL解决软件局部更新的问题，而只需要更新软件相应的DLL文件中的实现函数。遗憾的是，DLL是面向应用程序的。而COM为Windows操作系统提供了统一的、面向对象的、可扩充的数据交换协议。COM的一个至关重要的特征是，它规定的接口与编程语言无关，也就是定义了二进制代码的标准。这使得组件不论用什么语言编写，只要它们符合COM规范，就能够进行相互通信。DCOM规范提供了分布式计算条件下的组件重用可能性。

本书介绍自动化技术（Automation）。自动化在COM+模型中占有至关重要的位置。使用自动化，应用程序能够提供自身的对象及其属性和方法。这样，使用与自动化兼容的应用程序和编程语言（如VC++甚至VBA这样的脚本语言），你也可以在自己的应用程序中调用MS Word、MS Excel中的那些常用对象。为其他应用程序和编程工具提供属性和方法的组件称为“自动化服务程序”，调用这些属性和方法的应用程序和编程工具称为“自动化客户程序”。通过组件提供的接口，自动化客户程序就可以利用自动化服务程序的成员。ActiveX其实就是自动化服务程序。

相信本书的出版，可以在很大程度上弥补国内有关COM/COM+技术资料的空白。本书编写体例独特，作者充分考虑了读者的实际需求，既可以作为COM+/自动化技术的学习指南，也可以作为编程时的参考。

本书由李二勇、刘莹、蔡颖、王景茹、李成武、蒋唯沧、牛会朋、于京生、钟鸣、康仑、白致铭、姜利、蒋华、叶全芝、龚伯川、刘小林、孙鹏、郭有为、郭军等进行翻译，并由李二

勇统稿。前导工作室全体工作人员为本书的翻译、审校、录排、校对付出了大量劳动。在此向他们表示诚挚的感谢！

由于时间仓促，且译者的水平有限，书中难免会出现一些错误或不当之处，恳请读者批评指正。如果你在阅读本书中遇到什么问题，或者对本书有什么意见和建议，请同前导工作室联系：qiandao@263.net。我们会尽力帮助你解决问题。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

本书在编写过程中得到了许多人的帮助，特别是前导工作室的同事们，他们为本书提供了大量的资料，并参与了本书的编写工作。在此向他们表示衷心的感谢。

## 目 录

译者序	1	4.2.1 创建可编程接口	24
<b>第一部分 概 述</b>		4.2.2 创建IUnknown接口	25
第1章 欢迎使用“COM+开发人员参考库”	1	4.2.3 创建IDispatch接口	25
1.1 “COM+开发人员参考库”的组织方式	1	4.2.4 实现双重接口	26
1.2 本书的组织方式	2	4.2.5 注册接口	26
第2章 “COM+开发人员参考库”使用指南	3	4.2.6 创建类标识符	27
2.1 本卷内容	3	4.2.7 传递格式化数据	27
2.2 其他各卷内容	6	4.2.8 实现IEnumVARIANT接口	27
2.3 获得更多信息	10	4.2.9 实现_NewEnum属性	27
<b>第二部分 自动化指南</b>		4.3 类型库	28
第3章 自动化概述	11	4.3.1 创建类型库	28
3.1 为什么提供对象	12	4.3.2 构建类型库	29
3.2 什么是ActiveX对象	12	4.3.3 注册类型库	30
3.3 什么是ActiveX客户程序	14	4.3.4 返回错误	31
3.4 客户程序与对象如何交互	15	4.4 Hello示例	31
3.5 什么是类型库	16	4.5 Lines示例	43
第4章 提供ActiveX对象	18	4.6 支持多种本地语言	61
4.1 提供对象	18	4.6.1 为支持多种语言的应用程序实现IDispatch	62
4.1.1 初始化被提供的对象	18	4.6.2 实现IDispatch成员函数	62
4.1.2 实现被提供的对象	19	4.6.3 创建独立的类型库	64
4.1.3 实现类工厂	19	4.6.4 定义场合ID	65
4.1.4 提供Application对象	20	4.6.5 加载类型信息	65
4.1.5 创建注册文件	20	4.6.6 定义基于场合ID的自变量和字符串	66
4.1.6 释放OLE和对对象	23	第5章 访问ActiveX对象	68
4.1.7 检索对象	23	5.1 使用Visual Basic创建脚本	68
4.1.8 返回对象	23	5.1.1 访问远程对象	69
4.1.9 关闭对象	23	5.1.2 操作对象	73
4.2 应用程序设计注意事项	24	5.2 创建访问对象的应用程序和工具	74
		5.2.1 通过VTBL访问成员	74
		5.2.2 通过IDispatch访问成员	75
		5.3 创建类型信息浏览器	81



第6章 标准对象和命名规则 .....	82	第9章 转换和操作函数 .....	150
6.1 使用标准对象 .....	82	9.1 数组操作API函数 .....	150
6.1.1 对象属性 .....	83	9.2 变量操作API函数 .....	182
6.1.2 集合对象属性 .....	83	9.3 数据类型转换函数 .....	189
6.1.3 在类型库中使用Application对象 .....	85	9.4 BSTR和Vector转换函数 .....	306
6.1.4 Document对象属性 .....	86	9.5 日期和时间转换函数 .....	311
6.1.5 Documents集合对象 .....	87	9.6 变量计算 .....	317
6.1.6 Font对象 .....	87	9.7 小数计算 .....	330
6.1.7 Picture对象 .....	88	9.8 货币计算 .....	336
6.1.8 属性框架对象 .....	88	9.9 异类支持函数 .....	341
6.2 命名习惯 .....	89	第10章 类型库和对象说明语言 .....	345
6.2.1 使用完整的词和音节 .....	89	10.1 类型库的内容 .....	345
6.2.2 使用混合大小写 .....	89	10.2 使用MIDL和MkTypLib .....	346
6.2.3 使用界面中与通常用法相同的单词 .....	89	10.2.1 向IDL定义添加ODL .....	346
6.2.4 使用正确的类名称单复数 .....	89	10.2.2 MIDL和MkTypLib之间的区别 .....	346
6.3 可编程接口 .....	90	10.3 MkTypLib: 类型库创建工具 .....	348
6.3.1 访问包含文档 .....	90	10.4 ODL文件语法 .....	350
6.3.2 访问包含应用程序 .....	90	10.4.1 ODL文件举例 .....	351
		10.4.2 源文件内容 .....	354
		10.5 ODL参考 .....	356
		10.5.1 属性说明 .....	356
		10.5.2 ODL语句和指令 .....	378
		第11章 类型说明接口 .....	389
		11.1 概述 .....	389
		11.2 ITypeLib 接口 .....	391
		11.3 ITypeLib2接口 .....	398
		11.4 ITypeInfo 接口 .....	401
		11.5 ITypeInfo2接口 .....	416
		11.6 ITypeComp接口 .....	425
		11.7 类型编译及库函数概述 .....	428
		11.8 LHashValOfName .....	428
		11.9 LHashValOfNamesSys .....	428
		11.10 LoadTypeLib .....	429
		11.11 LoadTypeLibEx .....	430
		11.12 LoadRegTypeLib .....	431
		11.13 RegisterTypeLib .....	432
		11.14 UnRegisterTypeLib .....	433
<b>第三部分 自动化参考</b>			
第7章 分发接口及API函数 .....	91		
7.1 IDispatch接口 .....	91		
7.2 创建分发的API函数 .....	105		
7.3 CreateDispTypeInfo .....	106		
7.4 CreateStdDispatch .....	108		
7.5 DispGetIDsOfNames .....	110		
7.6 DispGetParam .....	111		
7.7 DispInvoke .....	114		
7.8 使用API函数注册活动对象 .....	116		
7.9 GetActiveObject .....	117		
7.10 RegisterActiveObject .....	117		
7.11 RevokeActiveObject .....	119		
7.12 IEnumVARIANT接口 .....	119		
第8章 数据类型、结构和枚举 .....	125		
8.1 IDispatch数据类型和结构 .....	125		
8.2 ITypeLib结构和枚举 .....	146		
8.3 ITypeComp结构和枚举 .....	148		



11.15 QueryPathOfRegTypeLib .....	434	14.1.1 在IDL文件中说明UDT.....	486
第12章 类型创建接口 .....	435	14.1.2 获取IRecordInfo接口 .....	486
12.1 ICreateTypeInfo接口 .....	436	14.1.3 传递单个UDT .....	487
12.2 ICreateTypeInfo2接口 .....	451	14.1.4 传递UDT的安全数组 .....	488
12.3 CreateTypeLib API .....	459	14.2 用户定义数据类型的支持方法 .....	490
12.4 ICreateTypeLib接口 .....	460	14.3 GetRecordInfoFromTypeInfo .....	490
12.5 CreateTypeLib2 API .....	467	14.4 GetRecordInfoFromGuids .....	491
12.6 ICreateTypeLib2接口 .....	467	14.5 IRecordInfo接口 .....	491
第13章 错误处理接口 .....	471	第15章 API级格式化例程 .....	503
13.1 返回错误信息 .....	471	15.1 VarTokenizeFormatString .....	503
13.2 检索错误信息 .....	471	15.2 VarFormat .....	504
13.3 IErrorInfo接口 .....	472	15.3 VarFormatDateTime.....	505
13.4 ICreateErrorInfo接口 .....	475	15.4 VarFormatNumber .....	506
13.5 ISupportErrorInfo接口 .....	479	15.5 VarFormatPercent .....	507
13.6 错误处理API函数 .....	481	15.6 VarFormatCurrency .....	508
13.7 CreateErrorInfo .....	481	15.7 VarWeekdayName .....	509
13.8 GetErrorInfo.....	482	15.8 VarMonthName .....	510
13.9 SetErrorInfo .....	483	15.9 VarFormatFromTokens .....	510
第14章 用户定义的数据类型 .....	485	COM+词汇表 .....	512
14.1 传递UDT .....	485	自动化词汇表 .....	515

# 第一部分 概述

本部分旨在介绍本卷其他各部分的内容并提供其他各卷的概述信息，以便于你查阅本卷及套书“COM+开发人员参考库”。

## 第1章 欢迎使用“COM+开发人员参考库”

如今，我们中的大部分人都已在“互联网时代”工作了至少几年。白天的时间从来都不够用，要做的工作从未完成，加快工作节奏所带来的利益从未如此显著。在互联网时代，努力更多、效率更高意味着进展更大、完成更快，而这些代表了更多——金钱、问世时间、知名度、市场份额或者悠长的舒适生活。有些事情不能一蹴而就，例如矫牙，通常不能仓促完成；然而在应用程序或者Web开发领域，由于前面提到的所有原因，互联网时代意味着可以而且必须迅速进行开发。在这个领域，COM+理解你所处的困境和所受的压力，创建COM+正是为了使你的开发工作更轻松、更迅速。

为帮助你了解有关COM+的必要信息及其COM基础，我编写了“COM+开发人员参考库”套书，它是有关COM和COM+编程指南、参考材料以及密切相关的或者新兴技术（如MIDL、ActiveX、自动化和结构化存储）的一套必备资料。

“COM+开发人员参考库”套书是关于COM+参考材料的完整参考集，包括COM+编程指南和全部参考材料。因此无论对使用COM+创建组件而言还是对创建工具而言，本套书都是“一站式”参考资料。由于COM+是对COM基础的补充，因此本套书也包括完整的传统COM编程指南和参考资料。

本套书的脉络很清晰。它完全围绕COM+主题编写。编写这套书旨在传达Windows编程中关于该主题的最完整、最权威以及最易读的可用参考信息，同时又不失侧重点。本套书中的每卷都论及一组逻辑相关的技术或者开发事项。这种编写方式经过特别安排，使你能够快速、高效而直观地查到所需信息。

除了COM+开发信息，“COM+开发人员参考库”还包括提示，它们是为使开发工作更轻松而编写的。例如，其中包括微软开发者网络在线的完整解释和详细叙述，这很大程度上有助于你摆脱微软开发者网络预订。如果你没有预订微软开发者网络，或者不知道为什么应该预订，书中也有相关信息，包括三种级别微软开发者网络预订方式之间的不同、每一种订阅方式所提供的内容，以及可通过互联网访问微软开发者网络在线时为何需要预订。

### 1.1 “COM+开发人员参考库”的组织方式

“COM+开发人员参考库”套书由五卷组成，每一卷讲述的编程重点或者直接与COM+相关，

或者与COM及其功能相关。这五卷指南和编程参考分述如下：

第1卷	COM+程序员指南
第2卷	COM+编程参考
第3卷	COM技术
第4卷	COM+自动化编程
第5卷	COM+结构化存储与ActiveX

将“COM+开发人员参考库”分为这五卷，使读者能够根据任务快速确定所需内容，并且有利于保持对该任务的侧重。

在第1卷中有一张微软开发者网络快照DVD。

## 1.2 本书的组织方式

在“COM+开发人员参考库”中，每卷都遵循通用规则划分章节，如下列标识：

第一部分：概述

第二部分：指南、示例及编程参考

本卷的格式与此稍微不同，这是为了以最清晰、最精确的方式表达其内容。

本卷最终分为如下几部分：

第一部分：概述

第二部分：自动化指南

第三部分：自动化参考

这里并没有很大的出入：第一部分提供概述，其余部分提供选定的参考或者概述，并对其巧妙分组。

### 本套书的编写方式

“COM+开发人员参考库”套书的编写方式是为了以最易读的方式传达最适当的信息。“COM+开发人员参考库”提供的外观和特性与以电子方式发布Microsoft参考信息的外观和特性一致，从而达到和微软开发者网络在线无缝集成的编写目的。换句话说，本书中的给定函数参考的表示方式经过特别编写，以模仿MSDN和微软开发者网络在线中函数参考页的表示方式。

保持这种集成性的理由很简单：使你（Windows应用程序开发人员）更容易地使用这些工具，更方便地获取要创建高质量程序所需的进行时信息。通过在参考资料中提供“公用界面”，如果你熟悉“COM+开发人员参考库”参考材料，则可以立即将经验应用到微软开发者网络或者微软开发者网络在线，反之亦然。简言之，这意味着一致性。

## 第2章 “COM+开发人员参考库”使用指南

如果你即刻拥有了本卷，则在本章中就能随意浏览“COM+开发人员参考库”中其他各卷的内容。

本章按照先后顺序，以各章提要的方式详细叙述了本卷内容。其他各卷内容比较简略，仅以各部分概述的形式提供了其他各卷的信息。

这种编写方式使你能够获得手头这卷书的详细信息，并获得其他各卷的概要信息，而不必被细节所淹没。如果对另一卷的概括信息感兴趣，你可以拿起该卷，翻到它的第一部分中题为“‘COM+开发人员参考库’使用指南”的那一章，你将发现，在那一章中依次详细叙述了各章信息。

本章叙述尽量简洁，以缩小篇幅（本卷如此，整个“COM+开发人员参考库”也是如此），并使各卷之间的重复内容最少。有些信息应归入每一卷中，如COM+词汇表等；所以，在每一卷中你都会找到这些内容。

### 2.1 本卷内容

在“COM+开发人员参考库”中，本卷的题目是自动化。它是提供有关自动化的所有指南和参考信息的“一站式”参考资料。自动化是一种基于COM的技术，它主要是为Visual Basic程序员而设计的，以使开发者能够创建自动执行经常性任务的自定义应用程序。

自动化允许利用脚本语言和其他解释型语言控制应用程序的功能。在使用其他语言编写组件以及使用解释型语言访问组件时，利用自动化可使之更方便。

#### 第一部分：概述

这几章介绍了本卷其他部分的内容并提供概述信息，以便你能更方便地浏览其他各卷以及本卷内容。即本卷的第一部分内容经过精简，以便为自动化指南和参考提供更多空间。第4卷第一部分只有两章，因此可以快速跳到第二部分。

#### 第二部分：自动化指南

本部分讲解自动化知识基础。这部分内容以自动化概述开始，接着几章是有关提供ActiveX对象（包括许多代码示例）、访问ActiveX对象，还有一章关于标准对象和命名规则。当然，关于这些主题有更多详细的信息，因此，接下来的几部分逐章提供了有关各个主题的信息。

#### 第3章：自动化概述

本章介绍自动化（正式名称为“OLE自动化”），这种技术允许使用脚本语言和其他应用程序控制软件包的独有特性。自动化使用组件对象模型（COM），但也可由其他OLE特性独立实现，

例如区域激活。使用自动化可以：

- 创建提供对象的应用程序和编程工具。
- 在一个应用程序中创建和操纵另一个应用程序提供的对象。
- 创建访问和操纵对象的工具。这些工具可以包括嵌入式宏语言、外部编程工具、对象浏览器和编译器。

应用程序或编程工具提供的对象称为ActiveX对象。访问这些对象的应用程序和编程工具称为ActiveX客户程序。

支持ActiveX技术的应用程序和其他软件包定义和提供可供ActiveX组件使用的对象。ActiveX组件是包含类的物理文件（例如.exe和.dll文件），这些类就是对象的定义。类型信息说明被提供的对象。这些信息可供ActiveX组件在编译时或运行时使用。

#### 第4章：提供ActiveX对象

提供对象使之可供其他应用程序和编程工具编程使用。本章讨论如何设计提供对象的应用程序，然后使用Platform Software Development Kit（平台软件开发工具包，SDK）中“COM Programmer's Reference”中的不同示例演示设计方法。

#### 第5章：访问ActiveX对象

要访问被提供的对象，可以使用Microsoft Visual Basic、Microsoft Visual C++、Microsoft Excel以及其他支持自动化技术的应用程序和编程语言创建ActiveX客户程序。本章讨论了几种访问被提供的对象的方法。

#### 第6章：标准对象和命名规则

本章说明了标准ActiveX对象，并讨论了创建对象时的命名规则，这些对象在应用程序尤其是支持多文档界面（MDI）的用户交互式应用程序中是惟一的。如果ActiveX对象不是用户交互式的或者仅支持单文档界面（SDI），则应对标准和规则做适当修改。

**标准对象** 它由一套自动化所定义的对象构成。在应用程序中你可以适当地使用这些对象。本章说明的对象面向基于文档的用户交互式应用程序。其他应用程序（例如非交互式数据库服务器程序）可能有不同的要求。

**命名规则** 它是一些建议，以提高应用程序之间的一致性。

本章还按照派生自Visual Basic的语法提供了示例。

### 第三部分：自动化参考

本卷的第三部分完整地叙述了自动化参考信息。各章参考信息类型各异，包括类型说明接口和错误处理接口。但这部分中以第9章为核心，而其他各章都是它的补充！下面逐章提供了第三部分各章的信息。

#### 第7章：分发接口及API函数

分发接口提供了一种在应用程序内部提供和访问对象的方法。本章提供有关自动化如何定义下列分发接口和函数的编程参考：

- **IDispatch接口** 为自动化编程工具和其他应用程序提供对象、方法和属性。
- **分发创建API函数** 简化IDispatch接口的实现。使用这些函数可以自动生成IDispatch接口。



- **注册活动对象的API函数** 提供帮助标识对象的运行中实例的函数。
- **IEnumVARIANT接口** 为ActiveX客户程序提供一种枚举集合对象的方法。这是一个分发接口。

### 第8章：数据类型、结构和枚举

每一个自动化接口都与数据信息相关。本章包括下列信息：

- 数据类型
- 数据结构
- 数据枚举

### 第9章：转换和操作函数

由于本章篇幅较长，所以请把握好阅读期限。数据操纵和转换函数访问并操纵数组、字符串和自动化使用的各种数据类型。本章包括有关下列数据操纵和转换函数的信息：

- 数组操纵
- 字符串操纵
- 数据类型转换函数
- BSTR和矢量转换
- 数字解析函数
- 日期和时间转换
- 变量计算
- 货币计算
- 混合支持函数

### 第10章：类型库和对象说明语言

提供ActiveX对象时，能够使其他厂商的程序相互操作。为了使用这些对象，厂商必须能够访问对象的特性（属性和方法）。要使这些信息可用，开发者必须：

- 发布对象和类型定义（例如，以印刷文档的形式发布）。
- 将对象编译为二进制应用程序，以便能够使用IDispatch::GetTypeInfo或者ITypeInfo和ITypeLib接口的实现访问这些对象。
- 使用“微软接口定义语言（MIDL）”编译器或MkTypLib实用工具创建包含对象的类型库，然后使类型库可用。

MIDL编译器和MkTypLib实用工具都能编译用“对象说明语言（Object Description Language, ODL）”编写的脚本。微软已扩展了“接口定义语言（IDL）”使之包含完整的ODL语法。

你应优先采用MIDL编译器而不是MkTypLib，这是因为MkTypLib正被逐步淘汰，并将不再受到支持。

有关MIDL编译器的详细信息，请查阅第1卷第四部分。

### 第11章：类型说明接口

本章定义并提供类型说明接口的编程参考。类型说明接口提供了一种读取并绑定类型库中的对象说明的方法。ActiveX客户程序在浏览、创建和操作ActiveX（自动化）对象时使用这些说明。

本章说明的类型说明接口包括：

- **ITypeLib** 检索有关类型库的信息。
- **ITypeLib2** 包括附加的类型库检索操作。
- **ITypeInfo** 读取类型库中的类型信息。

- **ITypeInfo2** 包括附加的类型信息检索操作。
- **ITypeComp** 包括类型查找和绑定方法。

本章还说明了加载、注册和查询类型库的函数。

### 第12章：类型创建接口

本章提供创建类型创建接口的编程参考。类型创建接口用以创建自动进行类型说明的生成和类型库的生成的工具。例如，**MkTypLib**实用工具和微软接口定义语言（MIDL）编译器使用这些接口创建类型库。

### 第13章：错误处理接口

绑定在虚函数表（VTBL）中产生的对象需要使用自动化错误处理接口和API函数，以定义和返回错误信息。本章提供这些错误处理接口的编程参考。

### 第14章：用户定义的数据类型

本章提供在自动化中使用用户定义数据类型的编程参考和信息。用户定义数据类型（在Microsoft Visual Basic中称为用户定义类型（UDT），在C和C++等语言中称为结构，一般情况下指记录）就是声明为一种信息类型的几组相关数据项。要以结构化的方式将几条相关信息分组时，用户定义数据类型非常有用。

### 第15章：API级格式化例程

要将变量格式化为字符串，首先必须标记格式字符串，然后将格式应用到字符串。下面说明了本章中的API级格式化例程：

- **VarTokenizeFormatString** 将实际格式字符串解析为一系列标记。
- **VarFormat** 在调用**VarTokenizeFormatString**和**VarFormatFromTokens**时，通过使用用户默认场合将变量格式化为字符串形式。
- **VarFormatDateTime** 使用指定的日期格式将变量格式化为字符串。
- **VarFormatNumber** 将包含数字的变量格式化为字符串形式。
- **VarFormatPercent** 将包含百分数的变量格式化为字符串形式。
- **VarFormatCurrency** 将包含货币值的变量格式化为字符串形式。
- **VarWeekdayName** 返回包含本地工作日名称的字符串。
- **VarMonthName** 返回包含本地月份名称的字符串。
- **VarFormatFromTokens** 获取**VarTokenizeFormatString**的输出，并将其应用到变量以产生格式化的输出字符串。

## COM+词汇表

在该词汇表中按字母顺序列出了COM+词汇。每一卷（包括第5卷）的书后均附有该词汇表。

## 自动化词汇表

确切地说，该词汇表是本卷特有的术语词汇表。

## 2.2 其他各卷内容

本节提供了yu其他各卷的概述。



## 第1卷：COM+程序员指南

“COM+开发人员参考库”第1卷解释了COM+提供的强大特性。除此之外，该卷还解释了在COM+应用程序中实际应用那些重要特性的方法。

### 第一部分：概述

和其他各卷的第一部分一样，这几章介绍了本卷其他部分的内容，并提供其他各卷的概述信息，使你更加方便地浏览这套“COM+开发人员参考库”。

在该卷第一部分的开始处提供了如何充分利用微软开发人员资源（如微软开发者网络和微软开发者网络在线），接着几章解释Windows DNA 2000 框架以及COM+与此框架相符之处。

### 第二部分：COM+编程指南

本部分提供了一套详尽的编程指南，用以解释如何充分利用COM+。第1卷的这一部分以COM+应用程序概述开始，接着是设计COM+应用程序，以及其他重要章节。

本卷中最吸引人之处或许是第8章。该章篇幅较长但易于理解。第8章提供了COM+所有重要特性的详细解释和编程指南。下面按照它们在第1卷第8章中的出现顺序列出这些重要特性：

- **上下文** 说明上下文（context）。COM+服务以上下文为基础，提供给在COM+应用程序中运行的已配置组件。
- **事务处理** 介绍一般的事务处理原理，提供有关COM+中事务的较详细信息，并给出实际编写事务组件的提示。
- **并发** 说明在COM+的上下文中运行的并发。
- **COM+中的安全** 提供使用COM+安全特性的指导以保护COM+应用程序，适用的应用程序范围从在管理配置的服务到在代码中调用的API。
- **对象缓冲池** 提供有关对象缓冲的详细信息。对象缓冲是COM+提供的一种自动化服务，使用这种服务可以配置组件，使其实例在缓冲池中保持活动状态，以供任何请求组件的客户程序使用。
- **即时激活（JIT, Just-in-Time）** 提供COM+提供的一种自动化服务的详细信息。该服务可帮助你更有效地利用服务器资源，尤其在处理大量事务而相应增加应用程序的规模时更是如此。
- **排队组件** 说明排队组件。它是一种基于消息队列服务（MSMQ）的COM+特性，用以提供一种异步调用和执行组件的方法。
- **COM+事件** 说明COM+事件。COM+事件扩展了COM+编程模型，以支持在发行者或预定义者与事件系统之间的延迟绑定事件或方法调用。
- **共享属性管理器** 说明用以管理对象的共享瞬间状态。
- **补偿资源管理器** 提供有关“补偿资源管理器（CRM）”详细信息的链接。这是一种COM+特性，它提供了一种快速又方便的方法，将应用程序资源与“微软分布式事务协调器（MS DTC）”事务进行集成。

在第1卷的这一部分中，还有其他重要章节，但此处只介绍关键特性。

### 第三部分：COM+工具编程指南

第1卷第三部分提供“微软分布式事务协调器（MS DTC）”、“补偿资源管理器（CRM）”、

“资源分配器”和“COM+设备”的概括性说明和编程指南。

**注意** 在Windows 2000的某些beta版本中，COM+设备指“事件接收器”。我尽量确保在文字中剔除“事件接收器”这一术语。如果你发现引用了“事件接收器”，请将其当做“COM+设备”。

#### 第四部分：Microsoft接口定义语言

第1卷第四部分完整地论述了微软接口定义语言（MIDL）。MIDL定义客户程序和服务器程序之间的接口。微软随“platformSDK”提供了MIDL编译器，以使开发者能够创建远程过程调用和COM接口所需的接口定义语言（IDL）文件和应用程序配置文件（ACF）。MIDL为C/C++程序员而设计。还要求熟悉远程过程调用。

所有基于Microsoft Windows 98和Windows 2000操作系统的客户/服务器应用程序都可使用MIDL。它还能用以为不同的网络环境（包括Unix和Apple之类的操作系统）创建客户和服务器程序。

#### 词汇表

在这一部分你将找到COM+词汇表。

### 第2卷：COM+编程参考

“COM+开发人员参考库”第2卷集中论述了完整的COM+编程参考。我在编写时试图将所有参考都放在这一卷中，以便你可以方便地使用COM+编程参考。

#### 第一部分：概述

和其他各卷的第一部分一样，这几章介绍了本卷其他部分的内容，并提供其他各卷的概述信息，使你更加方便地浏览本套“COM+开发人员参考库”。

这一部分还包括“充分利用微软开发人员资源之二”（接第1卷）。这一部分说明如何扩充微软开发人员资源，例如学习材料、技术支持和其他有用的信息。

第2卷第一部分的最后一章包括：简单对象访问协议（SOAP）1.1版规范。在那些不得不慎重地使用COM、CORBA或其他架构内项目的开发人员心中，SOAP应该占有特殊的地位。如对其感兴趣，请查阅第2卷第4章获取有关SOAP的详细信息。

#### 第二部分：COM+参考

该卷第二部分完整地论述了COM+参考，包括所有那些不可或缺的COM+编程元素，比如COM+对象、COM+接口、COM函数和COM+枚举等。

这一部分还包括管理对象、管理集合和管理接口。

#### 第三部分：COM+工具编程参考

此部分包括COM+工具开发人员所需的所有参考材料，例如事务参考对象、DTC和CRM接口参考、资源分配器以及COM+设备信息，还有“MTS支持”接口参考。

#### 词汇表

在这一部分中你将找到COM+词汇表。