

Visual Basic Developer's Guide to COM and COM+

Visual Basic

开发指南

— COM 和 COM+ 篇

〔美〕 Wayne S. Freeze 著

金德洪 袁海洋 等译

金德洪 审校

- 用VB创建强大的分布式应用程序
- 用VB开发和访问COM对象
- 使COM+与SQL Server协同工作



電子工業出版社

Publishing House of Electronics Industry

URL: <http://www.phei.com.cn>

12
FRZ/1

**Visual Basic Developer's Guide
to COM and COM+**

Visual Basic开发指南 ——COM和COM+篇

〔美〕 Wayne S. Freeze 著

金德洪 袁海洋 等译

金德洪 审校



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

0057193

内 容 提 要

如今,随着网络技术的发展,计算机应用领域日渐扩大、应用需求日渐增多,专业人士对成功的企业级的分布式应用越来越瞩目,自然,焦点之一就是代表着基本编程技术的COM和定位于多层应用的COM+。

本书作为“Visual Basic开发指南”系列丛书的重要一篇,面向中、高级VB程序员介绍了COM的工作机理与实现、用VB进行COM编程、用VB进行COM+编程;并以三个不同的场景为背景,展示了COM+事务处理的开发过程。

本书注重理论,但更注重实践。相信读者会在专家的引领下,掌握COM和COM+的开发应用。



Copyright©2000 SYBEX Inc., 1151 Marina Village Parkway, Alameda, CA 94501. World rights reserved. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior agreement and written permission of the publisher.

本书英文版由美国SYBEX公司出版,SYBEX公司已将中文版独家版权授予中国电子工业出版社及北京美迪亚电子信息有限公司。未经许可,不得以任何形式和手段复制或抄袭本书内容。

图书在版编目(CIP)数据

Visual Basic开发指南——COM和COM+篇/(美)弗瑞兹(Freeze, W.S.)著;金德洪等译—北京:电子工业出版社,2000.9

书名原文: Visual Basic Developer's Guide to COM and COM+
ISBN 7-5053-6170-8

I. V… II. ①弗… ②金… III. BASIC语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2000)第69321号

书 名: **Visual Basic开发指南——COM和COM+篇**

著 者: [美] Wayne S. Freeze

译 者: 金德洪 袁海洋 等

审 校 者: 金德洪

责任编辑: 李莹

印 刷 者: 北京天竺颖华印刷厂

装 订 者: 三河金马印装有限公司

出版发行: 电子工业出版社 URL:<http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编: 100036 电话: 68279077

北京市海淀区翠微东里甲2号 邮编: 100036 电话: 68207419

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 17 字数: 430千字

版 次: 2000年9月第1版 2000年9月第1次印刷

书 号: ISBN 7-5053-6170-8

TP·3310

定 价: 28.00元

版权贸易合同登记号 图字: 01-2000-0346

凡购买电子工业出版社的图书,如有缺页、倒页、脱页,请向购买书店调换,若书店售缺,请与本社发行部联系。

J5345/32

致 谢

几乎每个人都认为当作家是一个美妙的职业，你可以在家工作，自由地安排时间，干自己真正愿意干的事情。而只有作家本人，他们的家庭成员以及把那些手稿魔术般地变成书的人却不以为然。他们知道写作是一件艰苦的工作，虽偶尔有短暂闲暇，也难以补偿连续数月、每天十六小时的辛劳。

我要感谢**Laura Belt**，谢谢她辛苦的工作。也许某一天我们能够以写作来创造一种真正的生活。我还要感谢**Ash Rofail**在百忙之中抽出时间为本书写了前言。

我还要对**Sybex**出版公司我的那些朋友们道谢。感谢开发编辑**Denise Santoro**，他让我在本系列书籍中选择了此书（什么时候会让我写那本**DirectX**书呢？）；项目编辑**Elizabeth Hurley-Clevenger**；生产队伍领导**Jennifer Durning**；电子出版物专家**Maureen Forsys**以及图形专家**Tony Jonick**。我还要感谢**Marilyn Smith**修改了我的作品，指出了我的所有错误——没有你的话我可干不了。没有你的努力和耐心，此书将不值一读。现在，什么时候我可以开始下一本书呢？

写作的另一方面是占据了如此多的时间，以致我总是没有时间去拜访我的朋友们，甚至找不到同他们交谈的时间。**Shaun, Elwyn, Rick, Bob W., Veronica, Scott, Bob K**和**Ian**，我许诺不久将与你们联系。谢谢你们的支持。

Bucky和**Goose**是可以找到的最好的两个姻亲（如果方便，下一个夏天可邀请孩子们外出三周）。

虽然我总是很愿意经常去看望我的父母亲，但我却没有做到。今后我将努力做好每件事，即使不得不占用我许多工作时间。

如果你仔细阅读本书，会发现我谈到了**Christopher, Samantha**和**Jill**。**Chris**现在六岁，但已懂得相当多的计算机知识，甚至可以跟过去曾和我一同工作的人相媲美，要知道这些人可都是拿高薪的行家呀。**Samantha**才五岁，但比起**Chris**来，她的能力也丝毫不逊色，即便常使**Windows**系统发生崩溃。我非常爱他们两个，真想开着我的宝时捷车带他们去兜风，不知他们谁会更愿意呢？

我可爱的妻子**Jill**在她自己的领域中是一个受人尊敬的作家，她已写过若干关于**Microsoft Office, Internet**和**Web TV**方面的书。我只想说明只要相信自己，什么事都有可能，还有我爱你。

前 言

自从COM面世以来时光已过了六载，它的历史比Visual Basic提供开发大规模应用能力的历史还要悠久。但是，在用Visual Basic来开发用户界面和一些小的商务逻辑时，某些Visual Basic的开发人员仍然工作于相同的其过去所熟悉的范围内。在COM和Visual Basic发展的这些年来，一些开发人员仍然没有理解存在于COM技术和Visual Basic工具之间的关系。

Visual Basic在近几年中已日臻成熟，而以COM为基础更便于开发分布式应用。COM已经存在并且将继续发展。Visual Basic作为应用最广泛的开发语言，将继续利用COM的优势。倘若你是一个Visual Basic开发人员，但还没有开始以COM的方式来编程，那你可得赶紧加快速度了。

Visual Basic的开发人员已不再孤立地进行编程，他们不满足于仅仅完成对于构建一个应用并使之能够运行所必须的基本任务。随着应用需求、大量的基于Web的应用、以及操作系统的嵌入式服务等方面的日趋复杂化，对于成功的企业级的分布式应用来说，精深的COM和COM+知识是必不可少的。

为了从Visual Basic方面来理解和掌握COM，就必须了解Visual Basic的对象是什么、它们如何工作，了解Visual Basic如何看待COM、如何利用COM的服务。此外，还需学习如何利用COM+的事务处理和新的COM+服务来构建一个基于COM的分布式应用。由于COM+是一个Windows 2000的集成部件，因此必须了解这种集成将如何影响应用的发展。

在本书中，Wayne Freeze在引导你进入COM和COM+方面做了大量的工作。本书从Visual Basic的角度解释了COM和COM+，加深了你现有的Visual Basic知识。你所需的基本的知识就在本书中。要了解COM只需从第1章开始，在其中Wayne Freeze讨论了COM的演变过程，作了一些后继章节中需用的约定，还讨论了组件、队列、事务和一些新的COM+特征。

每一个Visual Basic开发人员，尤其是那些在今后若干年仍然对做Visual Basic开发人员感兴趣的人都应在学习COM、COM+及其所提供的操作系统服务方面作些投资。如果你打算认真地扩充你的COM知识并掌握用COM来作应用开发，那么请完整地通读本书。

——Ash Rofail

简介

Visual Basic是我最喜爱的编程语言。而**BASIC**是我学会使用的第一门编程语言。过去，我要在纸带上编写程序，然后把它装到几英里外的主机上。从那以后尽管我还用过多种编程语言，但总是留恋于**BASIC**。

如今，占统治地位的**BASIC**方言是微软（**Microsoft**）公司的**Visual Basic**。它是一种通用性强、易于使用的编程环境，是为**Windows**环境编程的最佳手段。尽管可能**Visual C++**的效率略高、开发能力略强，但它使用起来需花费的时间更多。在目前，当计算机以**650MHz**或更快的速度来运行时，应用程序的运行速度与你如何快速构建它、维护它来比，就显得没有那么重要了。

目前，市场人士经常变换一个产品或一种技术的名字以使它听起来更新鲜，更引人注目。例如，**Windows 2000**服务器比**Windows 5.0**服务器听起来更令人感兴趣。在**2000**年前更新到**Office 2000**比起更新到**Office 9**听起来似乎更为重要。

即使应用程序开发人员也会受到市场风气的影响。起先被称做对象链接和嵌入的技术（**Object Linking and Embedding**，简称**OLE**）几乎每更换一次产品的版本，则有一个新的名字。不管名字如何改变，其产品核心仍是原先的那个。**OLE**已经进化成组件对象模型（**Component Object Model**简称**COM**），而这就是微软的供**Windows**编程人员构建可重用对象的方法。

微软不断引入一些新的**COM**工具，如**Microsoft Translation Server**（简称**MTS**）和**Microsoft Message Queue**（简称**MSMQ**）。就好像为了证明市场风气还是如此，微软选择了把**MTS**和**MSMQ**作为一种称作**COM+**的新服务的部件。因而，就使开发人员有理由认为它们的功能已经发生变化，相信它们比起旧一代版本运行得更好。

Visual Basic和COM

Visual Basic是微软战略上用来进行快速应用开发（**Rapid Application Development**，简称**RAD**）的语言。它定位于那些需要快速开发企业应用而不用关心大量细节的编程者。为此，**Visual Basic**接管了在**Windows**中编程的大量烦琐细节，使开发者可集中注意力于所需的应用上。

使用**Visual Basic**编程，你甚至可以不用写一条代码，即可构建一个有意义的程序。如若不信，请按此做一下：启动一个新的程序，放一个数据控件于其上，指定数据库名，指定需进入数据控件属性中的表名，然后放一些文本框控件或标签控件在窗体（**Form**）上。设置数据源特性为数据控件，并把数据字段特性设置为你数据库中的某一列名。上述工作做完后，点击“运行”。你就建成了一个不需一行代码的、可以在数据库表内进退的程序。

在**Visual C++**中要完成这样的一个程序需要编写大量代码，花费大量的时间把各部分代码拼在一起，尽管用**C++**编写的程序可能运行得更快，但一般用户不会注意到二者之间速

度上的差异。因此，为什么要用Visual C++呢？

像C++这样的编程语言也还总是有用的。Visual Basic与硬件结合较差，不适于构建高效的操作系统和编译器。但是你上一次写操作系统或者编译器是在什么时候呢？

遗憾的是，Visual Basic不是选作用以构建COM和COM+组件的语言。COM和COM+是被设计于C++领域的，但这并不意味着在Visual Basic中不能构建这些组件，只是说明在某些时候为了充分地利用组件的能力，你还必需等待下一版本。

迄今为止，在Visual Basic中只要你明白如何去干，没什么用COM不能做的。糟糕的是对COM+来说则不然，一些COM+的特征在作者编写本书的时候仍然在Visual Basic中不能使用。幸运的是COM+还是非常壮健的，你可以拐弯抹角地处理这些限制。我完全可以预见这些问题将在Visual Basic的下一版中被解决掉，但目前，这些问题还存在。

本书适合的对象

本书定位于那些中级到高级的、愿意在他们的程序中更多地与COM和COM+打交道的Visual Basic程序员。也就是说，你应了解如何构造Visual Basic程序，熟悉开发环境，知道如何使用大部分通用控件，如文本框，标签和命令按钮等。我还假设读者知道怎样使用Windows，并且了解诸如文件、目录和窗口重迭等概念。假如你还不了解上述内容，那么我建议你去读由Evangelous Petrouous写的“Mastering Visual Basic 6”（Sybex, 1998，已由美迪亚公司译作《Visual Basic 6从入门到精通》）。读完该书后，再浏览一下我的“Expert Guide to Visual Basic 6”（Sybex, 1998，已由美迪亚公司译作《Visual Basic 6高级开发指南》）。该书论及了大量的其他任何地方所未涉及的话题，将使你洞悉Visual Basic如何工作的机理。

本书涉及的内容

本书包括了使用COM和COM+。大多数应用可以受益于COM组件，COM代表着一种基本的编程技术，有助于使你构建面向对象的、可重用的组件。

另一方面，COM+主要定位于多层应用，它使你可建立独立于客户程序和数据库服务器的应用逻辑层，有助于构建易扩充的应用。

本书的组织

本书由四部分构成。第一部分“了解COM”，介绍在Windows下组件对象模型（COM）是如何工作的基本知识。首先，你将学会COM是什么，它是如何演化的，为什么要使用它。在对COM对象如何工作回顾之后，接下来讲述COM在Windows中是如何实现的，该部分中的最后一章解释了在Visual Basic中怎样构建类模块。

本书的第二部分“用Visual Basic进行COM编程”，讨论了如何建立三种主要类型的COM组件：ActiveX DLL，ActiveX控件和ActiveX EXE。这些章节讨论了形成所有COM组件的基本编码细节，并且将用于本书的其余部分。你还将学会利用DCOM来转换已有的Active EXE程序。在该部分的最后一章，谈到了如何创建自己的数据绑定对象的控件。这

意味着你可以建造用于自动显示和更新数据库字段的控件，而对应用程序员来说，使用你的组件将不需要任何编码。

第三部分“用Visual Basic进行COM+编程”，论述了Visual Basic范围内COM+的一些主要特性。你将学会如何使用COM+事务处理服务器、消息队列和驻留内存的数据库系统。该部分最后一章涉及到在你的COM+应用中其他安全性方面的基本话题。

本书的最后部分“开发COM+事务处理”，显示了在三个不同场景中如何使用COM+事务处理：一个传统的基于客户机/服务器的程序，一个基于Web浏览器的应用，以及一个计算机并不能保证总是连接到的客户机/服务器应用。

示例程序

与书几乎一样重要的是程序，在本书中讨论到的每一个程序都可从Internet下载。你可以从Sybex站点www.sybex.com或从本人的站点www.JustPC.com得到这些程序，只需连接到本书，然后按照那些指令，即可下载那些文件。

你会发现一些用于示例中的代码贯穿于本书。但是，我极力建议你花费一些时间去读一下这些完整的源代码。如有可能，还应试着运行这些程序。如若愿意，可以复制部分这些程序到你的自己写的程序中。

虽然我为本书写的这些程序都能够运行，但对一个非编程人员来说，还需要做一些其他的工作。这是有意而为之的，因为我的目的是交流如何利用Visual Basic的专门特性，而不是构建成品似的应用。如果遗漏了诸如Exit按钮这样美妙的特性，而我还能达到此目的，那就让它遗漏吧。

本书没有涉及的内容

本书没有涉及到Visual Basic的基本内容。一些章节将认为你已经知道如何使用SQL Server和常用的数据库编程。同样，本书也没有谈到如何编写IIS应用和Internet Information Server如何工作。

如果你不知道一个事件和一个文本框之间的区别，那么先花几个月时间阅读“Mastering Visual Basic 6”并使用一下Visual Basic。如果你已经知道怎样进行Visual Basic编程，也许还应看看该系列中的其他书籍。由Mike Gunderloy写的“VB Developer's Guide to ADO”（Sybex, 1999, 已由美迪亚公司译作《Visual Basic开发指南——ADO篇》），谈到了所有有关开发数据库应用方面的内容。如果你要学习关于IIS和相关的應用，请参阅A. Russell Jones写的“VB Developer's Guide to ASP and IIS”（Sybex, 1999, 已由美迪亚公司译作《Visual Basic开发指南——ASP和IIS篇》）。Steve Bronn写的“VB Developer's Guide to the Win32 API”（Sybex, 2000, 已由美迪亚公司译作《Visual Basic开发指南——Win32 API篇》），书中有大量有用的关于使用Win32 API的材料。

关于作者

如果你想知道，我告诉你我是一个全日制的计算机书籍撰写人，同时也是一位计算机技术顾问，本书是我的第七本书。我以前的那些书讨论到了用**Visual Basic**编程、**SQL**和**Internet**方面的内容，我还是一个称作“汽车收藏家”流行软件的作者，该软件用**Visual Basic**编写。

我使用个人电脑的经验始于1977年，那时我用一套东西建造了一个原始的个人电脑系统，**Altair 8800**，只有基本的4KB的ROM和1KB的RAM，这是不太实用的，但当时它是太有意思了。从那以来我用过几乎每一种主要类型的个人电脑，以及许多中型和大型计算机。在我的家里安放有6至7台计算机，但目前我常用的是**Gateway 9100**便携机。

打我开始写作生涯以来，几乎再也没有时间顾及我的爱好了。如我确有闲暇的时候，你可能会在某个室外展览场上看到我正在拍二次世界大战中服役的战斗机的照片（我盼望某一天我能学会驾驶**P-51**翱翔）。我还喜欢收集各种汽车，从一辆1:144的道奇**Viper**到一辆1:1的保时捷**Turbo**（只能在无雨的天气下短距离驾驶）。我还有一所住着各种宠物的房子，目前，我有一条狗（**Kokomo**小姐），三只猫（**Pixel**，**Terry**和**Cali**），和一只我叫做**Raymond**（以我岳父的名命名）的蜥（**stingray**），它喜爱吃蠕虫，一种叫**guppy**的水面小虫和小虾。

如今我同我的家人住在**Maryland**州的**Beltsville**。我可爱的妻子**Jill**，总是帮助我在我的写作中删去那些不必要的逗号，使之没有废话。**Jill**也是个全日制的作家，已写过半打以上关于**微软Office 2000**到**Web TV**方面的书。我的六岁的儿子**Christopher**曾是**微软Windows 98**的非官方**Beta**版测试者，在**Window 98**未正式发布之前他已使用了四个多月。五岁的女儿**Samantha**美丽而逗人喜欢，闲暇时总爱推着她的手推车。

我维护着一个<http://www.JustPC.com>站点，提供关于我妻子和我所写书的其他另外的信息。我还经常回答关于这些书的问题，指出一些读者可能感兴趣的资源的出处。

你可随时通过**Wfreeze@JustPC.com**给我发电子邮件，我将尽可能回答你的问题或评论。但是请理解，对我来说，写作与个人生活是两码事，因而回复可能不能如你所愿那样的快捷和完整。若你发现在示例项目中有错误，或有改进示例项目的想法，敬请告知本人。经过你的同意，我会在我的**Web**上增加任何好的信息，以使其他人也能受益。

目 录

第一部分 了解COM	1
第1章 COM介绍	1
COM是什么?	1
从DDE到COM+	4
本章结束语	10
第2章 了解Windows机理	11
程序文件的类型	11
程序执行	12
虚拟内存	14
程序执行控件	17
本章结束语	19
第3章 运行COM对象	20
一个对象的生命周期	20
一个COM对象的内部是些什么?	22
接口	22
进程内COM对象	25
进程外COM对象	27
DCOM对象	30
COM+对象	30
本章结束语	31
第4章 在Visual Basic中构建类模块	32
用类进行编程	32
类模块的要素	33
模块变量	34
属性例行程序	35
方法例程	37
事件声明	38
一个关于类的小应用	39
本章结束语	45

第二部分	用Visual Basic编程COM	47
第5章	在Visual Basic中创建ActiveX DLL对象	47
	什么是ActiveX DLL?	47
	为何使用ActiveX DLL.....	48
	构建ActiveX DLL.....	48
	测试ActiveX DLL.....	61
	使用ActiveX DLL.....	64
	本章结束语.....	66
第6章	创建ActiveX控件	67
	何以致ActiveX控件有所不同?	67
	UserControl对象.....	68
	构建简单的ActiveX控件.....	72
	增加关于属性、方法和事件的信息.....	79
	创建属性页.....	80
	增加一个About框.....	84
	本章结束语.....	86
第7章	创建ActiveX EXE对象	87
	何以致ActiveX EXE有所不同?	87
	更多的类编程.....	88
	构建一个ActiveX EXE.....	91
	异步处理.....	94
	设计多目的程序.....	98
	完成ActiveX EXE服务器.....	99
	访问远程对象.....	100
	本章结束语.....	101
第8章	从COM到DCOM	102
	DCOM有何不同?	102
	运行DCOM配置工具.....	103
	EchoServer和重访问的EchoClient.....	117
	排除常见的DCOM错误.....	119
	本章结束语.....	120
第9章	用COM对象进行数据库编程	121
	使用数据源和数据用户.....	121
	构造捆绑型控件.....	123
	本章结束语.....	131

第三部分 用Visual Basic编程COM+	133
第10章 介绍COM+	133
事务处理	133
消息队列	135
In-Memory Database系统	135
其它COM+特性	136
COM+和Visual Basic	136
一个简单的COM+应用	140
本章结束语	154
第11章 了解消息队列	155
了解消息队列的工作原理	155
从Visual Basic来使用消息队列	158
建立COM+消息队列	166
介绍排队的组件	170
本章结束语	171
第12章 了解In-Memory Database系统	173
了解IMDB是怎样工作的	173
确定IMDB是否适合你	174
准备使用IMDB	175
在应用程序中使用IMDB	180
本章结束语	186
第13章 了解COM+的安全性	188
COM+下面的安全类型	188
配置安全措施	189
为安全措施编程	195
本章结束语	197
第四部分 开发COM+事务处理	199
第14章 为SQL服务器开发COM+事务处理	199
设计Contact Manager应用程序	199
用对象通信	201
为Contact Manager应用程序编程	205
本章结束语	215
第15章 为IIS应用程序开发COM+事务处理	216
将ContactManager移植到Web上	216
了解IIS Application	217

编程主web页	218
编程细目web页	224
使用Contact Manager事务处理	231
本章结束语	236
第16章 为MSMQ开发COM+事务处理	237
应用程序的设计	237
编程Contact Client	238
编程ContactQueue管理器	244
编程ContactTypes	247
本章结束语	250
附录A 朮语汇编	252

第一部分 了解COM

第1章 COM介绍

- COM元素
- COM的重要性
- COM发展的简史
- 事务处理和消息队列
- COM+特征

多年来，恨不得每一个人都一直在说：用可重用对象来构建应用程序。自Visual Basic 5以来，人们就具备了构造自己的可重用对象的能力。但是有多少编程人员真正地利用了这种能力呢？确实，用对象来构建应用比不用对象来说需要更周到的考虑，但是，结果还是值得一试的。

COM是微软的用于构建可重用对象的标准。这些对象是独立于任何某种编程语言的，也就是说，Visual Basic程序能够方便地访问用VC++或VJ++写的COM对象。同样，可用Visual Basic来写你自己的对象，使其也可用VC++程序来调用。如果你像我一样，你会忘掉其他的语言而只是用Visual Basic来生成自己的COM对象，并用Visual Basic来调用它们。

在本章中，我将泛泛回顾关于面向对象编程的一些背景材料和它与COM之间的关系。我还将论及COM是怎样从早期发展到微软市场队伍所称的COM和COM+的。

COM是什么？

COM代表组件对象模型（Component Object Model）。它是微软的一种标准规范，描述了程序员如何在Win32编程环境中生成可重用对象。COM是一种二进制标准，也就是说任何编程语言（当然应具有某种合适的能力）均可生成COM对象。

正因为COM对象是二进制的，它们可以包含在它们自己的可执行文件中。这就易于开发对象，使对象可用独立于某个应用程序文件的方式来发布。为了解COM究竟是什么，有必要了解词汇组件和对象的含义。

什么是组件？

一个组件（Component）就是一段独立的代码，它可以由若干不同的程序来共享。当设计一个应用程序时，可以选择开发一个极复杂的大型程序，或者把它分解为若干较小的模块。如果你觉得这听起来像结构化编程，那就对了。

通过将一个不易解决的难题分解为多个较小的问题，在相当程度上就减少了该问题的复杂性。同时，也使该问题更实际地可由不同的人员共同参与解决它，因为每个人可以负责一个或多个组成部分。

作为一个例子图1.1显示了一个典型的文字处理器的各个组成部分。文档的存储组件与拼写检查、文档显示和文档打印组件是分离的。

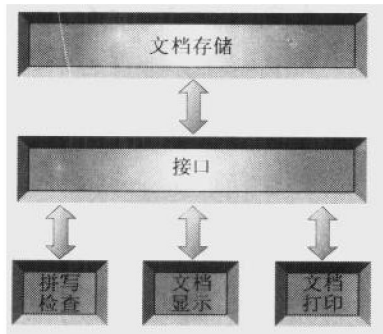


图1.1 一个典型的文字处理器的各组件

由于每个组件是独立于其他组件的，因此明确地定义各组件的边界是非常重要的。至少，你该向其他编程人员描述清楚如何从他们的组件中来访问你的组件。这称为组件的接口。在图1.1的例子中，每个欲与文档存储组件交互的组件都通过一个有效定义接口来进行。在构造应用程序的各个组成部分时，该接口使程序员可以相互独立地开展工作和完成相应的功能。

什么是对象？

一个对象（Object）就是一组代码，这些代码设计来可重复使用，并具有合理定义的接口。除非通过该接口，没有其他手段可取得该对象的信息。其优点是将该对象如何存储各种信息的细节同使用该对象的程序分离开了。你可以改变该对象怎样去存储它的数据，而不会影响到调用程序的工作。

注意：一个相当聪明的程序员可以绕过这些限制，甚至进入由某个对象占用的内存中。但是这不是一种好的办法，因为绕过对象的接口可能会导致该对象的数据管理崩溃。

一个对象事实上不仅相当于一段代码，它还表示其他的一些东西。它可以表示一段信息，一个文件，来自数据库的一组数据，或者文字文档中的某一段落；也就是说，它可以表示你所选择的任一事物。对象接口则定义了一系列的函数或子程序，它们将执行针对该对象所含数据的各种操作。

图1.2显示了一个用一个对象来访问字典的简单例子。此组件具有一个拼写检查器接口，包含如下四种功能：

- 当产生对象时调用Initialize。
- 当删除对象时调用Terminate。
- 如果在字典中找到该单词，WordIsSpelledOK返回真。
- 利用AddWordToDictionary在用户词典中增加一个词。

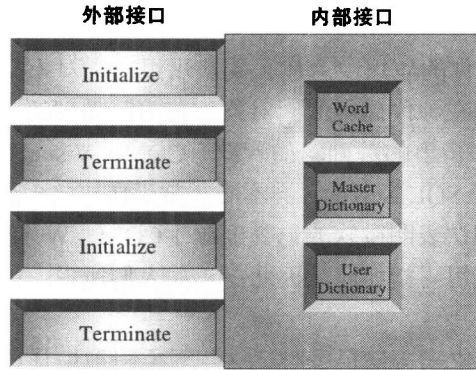


图1.2 一个简单的例子

在内部，该组件维护三种内部资源，它们只能在对象内部使用：

- Word Cache追踪常用的单词。
- Master Dictionary包含所有由字处理器所提供的单词。
- User Dictionary用以保留由用户加入到字典中的单词。

什么是组件对象模型？

组件对象模型即COM，可把产生单个组件的能力和产生可重用对象的能力结合起来。在微软的Windows环境下，它提供了一个强有力的工具，用于创建独立于任何一种编程语言的二进制对象模块，该模块具有明确定义的、面向对象的各种接口。

COM是基于客户/服务器模型的，每个对象的工作方式类似于服务器，可接收和处理各种来自客户程序的请求，并产生回复信息。图1.3显示了这种概念。请注意在此我并没有描述这些请求和应答是如何被处理的，因为这种处理完全取决于COM对象的设计者。

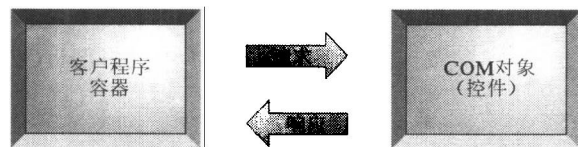


图1.3 一个COM对象和它的客户

正如你通过Visual Basic所知的那样，一个控件可以与你的程序通过三种方式来交流：即分别利用属性、方法和事件。

- 一个属性在对象中表示一个变量，然而它通常是由一对例程来实现的：一个例程用以提取属性值，另一例程用以设置此属性值。

- 一个方法只是一个函数或子程序，可由Visual Basic程序来调用。
 - 一个事件是一个子程序，它的参数在控件中被定义，存在于程序中。子程序由控件来调用，以便通知各种状态给程序。它的响应将指挥该控件所采取的动作。
- 此三种基本元素构成了程序将如何与一个COM对象进行交互的基础。

注意：一个COM客户有时被称用一个容器。一个COM对象有时被称作控件。

为什么COM很重要？

现在，你已经知道在你的应用中为什么要使用组件和对象。然而，许多语言，其中包括Visual Basic，都具有在源代码中嵌入对象的能力，为何还要使用COM呢？

COM是微软所做的许多事的基础。如若没有COM，Visual Basic会无法运行，其他的任何微软产品如Windows，SQL Server等均不能正常工作。由于微软的应用程序大都建立于COM技术之上，故实证测试表明微软的信誉度源于COM。Word 2000和Excel 2000严重地依赖COM，其他的微软应用同样如此。因此，正如人们所说，如果这玩艺对微软有益，对我也有益。

使用基于源代码的对象来构建程序时，这些对象将被编译成目标码。这有两个缺点：

- 如果有多个程序，每个可执行程序便拥有它自己的这些对象的复制品。因而只要改变对象，就必须重新编译所有使用这些对象的程序。
- 为了产生自己的应用程序，必须使用同编写这些对象时所用的相同的语言，这限制了人们在语言方面的选择。尽管你可能愿意用Visual Basic来编程，但有时会发现使用其他的编程语言对自己更有帮助。

由于COM对象是以其独立的目标码文件方式储存的，可以改变该对象而不必重新编译调用该COM对象的应用。为了提供新的特征和能力，可以给COM对象增加新的接口。（当然，如果修改了某个对象的现存的接口，那么就需要更新调用它的应用）。

从DDE到COM+

在今天的COM技术中，许多特征可以追溯到许多年以前。COM从若干不同的技术上发展起来，这些技术包括动态数据交换（Dynamic Data Exchang，简称DDE），Visual Basic扩展（Visual Basic Extensions，简称VBX），和对象链接和嵌入（Object Linking and Embedding，简称OLE）。了解COM历史演变过程将有助于更深入了解COM是如何运行的以及为何这样运行，这是为第3章做一些准备，在第3章中我将剖析COM的运行机制。

DDE

在Windows早期，如同今天一样，人们频繁使用剪贴板。用户可以把一个程序中的部分文档复制到剪贴板中，然后将它粘贴到另一程序的文档。可是，当人们处理复杂的图形文档时，发现简单的剪贴板不能很好地工作。

在80年代后期，微软开发了一种称作DDE的技术，它用程序简化了用剪贴板共享数据的处理。这种技术当时只适合于微软的应用，而并不通用。