

OLE 2

从入门到精通

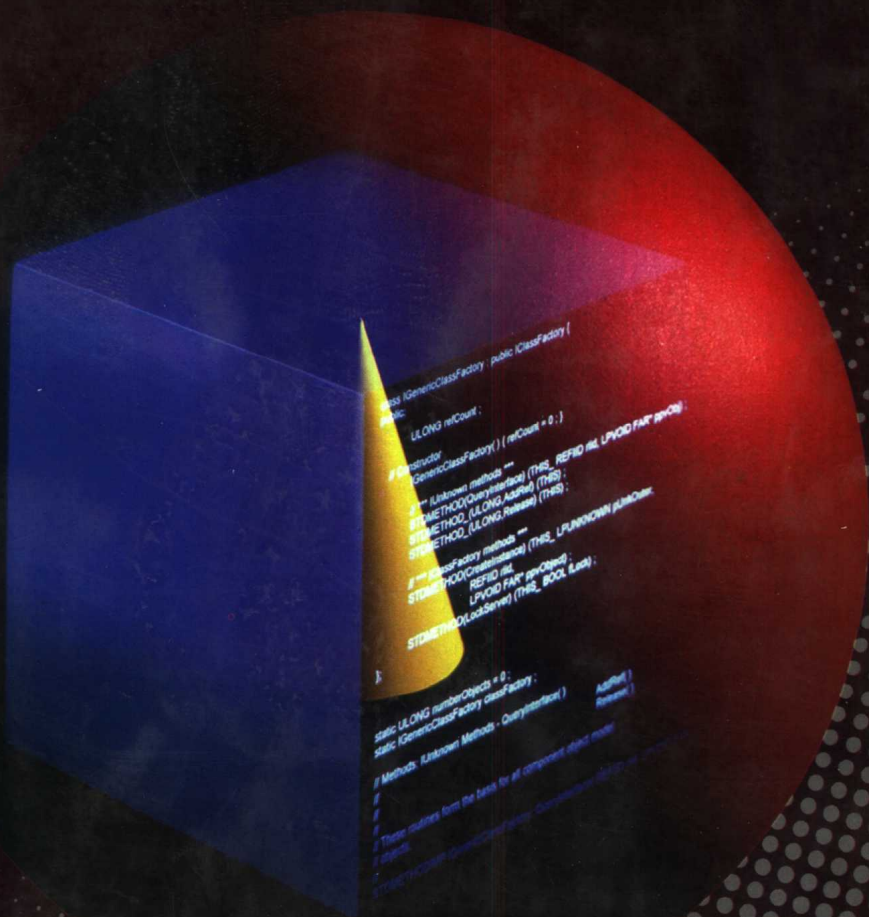
Mastering OLE 2

Bryan Waters

著

何玉洁 张建旭 译

麦中凡 何玉洁 校



电子工业出版社

Publishing House of Electronic Industry



OLE 2 从入门到精通

Mastering OLE 2

〔美〕 Bryan Waters 著

何玉洁 张建旭 毛新生 译

麦中凡 何玉洁 校

电子工业出版社

内 容 提 要

OLE（对象链接和嵌入）作为一组二进制协议，为应用程序间的集成和交互提供了迄今为止最为先进的手段。本书针对OLE 2的主要特点和功能，全面、详尽地阐述了OLE的基础——元件对象模型（COM）、复合文档、各种容器和各种服务器的原理及实现，作者用一个完整、实用的例子贯穿全书来阐述各种技术、概念和技巧，是学习OLE的一本不可多得的好书。

本书的读者对象为在Windows环境下编程的技术人员、研究人员和对OLE感兴趣的计算机爱好者。



Copyright©1995 SYBEX Inc., 2021 Challenger Drive, 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。

OLE 2从入门到精通

Bryan Waters 著

何玉洁 张建旭 毛新生 译

麦中凡 何玉洁 校

责任编辑 林 晶

*

电子工业出版社出版（北京市万寿路）

电子工业出版社发行 各地新华书店经销

北京美迪亚电子信息有限公司排版

北京顺义颖华印刷厂印刷

*

开本：787×1092毫米 1/16 印张：16 字数：410 千字

1996 年 1 月第 1 版 1996 年 1 月第 1 次印刷

印数：4000 册 定价：32.00 元

ISBN 7-5053-3489-1 / TP·1392

著作权合同登记章

图字：01-95-904号

出版说明

计算机科学技术日新月异。为了引进国外最新计算机技术，提高我国计算机应用与开发的水平，中国电子工业出版社与美国万国图文有限公司合资兴办的北京美迪亚电子信息有限公司取得了美国SYBEX公司的独家版权代理。SYBEX公司授权本公司通过电子工业出版社等出版机构全权负责在中国大陆出版该公司的中文版和英文版图书。现在与广大读者见面的是最近推出的第一批图书。今后我们还将陆续推出SYBEX公司的最新计算机图书和软件，为广大读者提供更好的服务，传递更多的信息。

美国SYBEX公司是世界著名的计算机图书出版商，该公司自1976年创办开始，其宗旨就是通过出版有效的，高质量的图书向计算机用户介绍实用技巧。我们优选翻译出版的图书是SYBEX公司的最新计算机图书，并采用了该公司提供的电子排版文件，从而提高质量并大大缩短了图书的出版时间，从根本上改变了以往翻译图书要落后原版书较长的“时差”现象，这在电子技术日新月异的时代具有深远意义。

北京美迪亚电子信息有限公司

1996年1月

序 言

OLE，即“对象链接与嵌入”技术，有其独特的价值。它的出现首次解决了一个无数独立开发者梦寐以求试图解决的问题。然而，如何掌握这项技术同样困扰着他们。我无法确定造成这种局面的确切原因，只能在此冒昧地作些猜测。

首先，OLE技术不只是为了解决某些简单问题而设计的，它包含了更广泛的范畴。其次，它通常是在一些很简单的场合中被解释和表述的，所以当最终涉及到细节问题时，它变得难以驾驭，使开发者手足无措、陷入困境。最后一点，它来自于Microsoft，而任何Microsoft的东西都令人生畏……，对吗？下面我们着重谈头两点，至于第三点嘛，你只能自己慢慢领悟啦！

OLE最初的开发目标是让用户能更方便地在Microsoft的各种应用软件，如PowerPoint、Word及Excel之间共享图形程序及数据。不知有多少开发者曾试图实现一种软件工具，使得应用程序能直观地共享数据。我认为，虽然OLE 1存在一些严重的缺陷，不过还是有效的。但是在OLE 1中，问题范围定义得过窄，所以，尽管它能够满足最初的需求，我们却无法对它拓广来处理其它更多方面的需求。为了弥补OLE 1的不足，必须先建立某种基本的结构。OLE 2的设计者们认识到这些问题具有普遍性，他们设计了一种极其简单的结构来解决这些问题，这种结构称为“元件对象模型”，简称COM。

至此，我们可以明白为什么许多开发者对OLE技术困惑不解了。虽然OLE和COM是两种独立的技术（确实，其中一个建立在另一个之上），但是COM技术解决了比OLE技术更基本的问题。它为开发者提供了必要的工具来生成可被多个进程共享的对象。它涉及到远程过程调用（RPC）、内存分配、参数及变元操作规则、错误处理、以及类注册等一系列问题。所有这些方面都与OLE无关，OLE只是跨越进程边界来生成和管理对象。

另一方面，OLE只不过是一个规范。这个规范有两个要素：关于包含和连接OLE对象的应用程序的容器规范；以及生成对象本身的服务器规范。虽然规范本身并不是以这种方式划分的，但这却是它最终的使用方式。理解了这些概念以后，建立基本容器基本服务器几乎是轻而易举的。

现在，摆在读者面前的是一种挑战：忘记那些所谓OLE是难懂的、过份复杂的和难于实现的传言，设想一下如果要实现这项技术的话，应当怎么做，也许将发现：它是一种简单、易懂，而且具有多种卓越特性的优雅技术。好吧，希望在用COM和OLE技术开发新一代软件的过程中找到乐趣，祝好运！

Bryan Waters

附言：本书所附的磁盘中包含了两个应用程序OLE Publisher和OLE Draw，以及书中所有的源代码。

致 谢

感谢我的家人在写书期间对我的支持，尤其感谢Sybex公司的Rudolph Langer在本书出版过程中给予我的大力支持，还要感谢Gary Masters、Kris Vanberg-Wolff、Jane Reh、Dusty Bernard、Amrik Dhillon、Barbara Gordon及所有在本书的出版准备过程中做了大量工作的同仁，同时感谢Microsoft公司的Mary Kirtland和Tony Williams给予的帮助。

12/5/84

译者序

OLE是Microsoft公司同计算机业界合作并发展起来的产品规范，目前已发展成为Microsoft面向对象策略的基石，OLE 2在Microsoft的未来产品（Windows 95及Windows NT的Cario版本）中将得到更为彻底的实现和更完全的功能扩展，从而结合网络，构造Windows产品家族的分布式计算环境（DCE），真正实现跨平台和跨网络的分布式对象计算。

借助OLE，用户能够构造声、文、图、像、影于一体的复合文档，可以更容易地集成应用程序。

目前，市场上具有OLE功能的产品越来越多，OLE构件，尤其是新的OLE控制（即OCX，类似VBX，但功能更强，更易移植）策略提供了强大的系统集成手段，为软件开发商提供了新的市场和发展机遇。

因而，OLE是在Windows产品家族中进行软件开发的一把通向成功的钥匙。

目前，关于OLE的好书并不多，OLE功能又十分复杂，学习和掌握它并非易事，而本书全面、详细地阐述了OLE技术，它将把读者带进OLE技术的精彩世界。

全书针对OLE 2的主要特点和功能，详尽介绍了OLE技术的基础、复合文档以及各种容器和各种服务器的原理和实现。

全书用一个完整、实用的例子逐步展开阐述了实现OLE所需的概念和技巧，对于希望尽快掌握OLE技术的人员来说，这是一本不可多得的好书。

我们谨将此书介绍给国内的业界同仁，希望对广大软件开发人员学习、理解和掌握OLE 2以及把握Windows产品家族的面向对象策略有所助益。限于时间和水平，我们对此书的翻译必有不当和错漏之处，敬请读者批评指正。

参与本书翻译工作的还有何玉书、赵海涛、毛健生等同志，在此一并表示感谢。

全书由麦中凡和何玉洁负责审校，在全书的翻译过程中始终得到麦中凡教授的精心指导，我们谨在此深表感谢。

同时感谢对本书的顺利出版作出努力的编辑和工作人员。

译者

1995年12月

目 录

第一篇 设计应用程序	1
第1章 OLE 2概论	2
OLE 2的特征	2
OLE的前景如何	5
第2章 元件对象模型	7
对象和对象服务器	7
消费者	15
注册数据库	18
内存管理	18
全局唯一的标识数	19
错误处理	22
第3章 创建一个元件对象	26
对象及消费者示例: IBeep	26
IBeep对象	26
客户程序示例	36
注册文件	40
第二篇 复合文件	43
第4章 在应用程序中使用复合文件	44
什么是复合文件	44
应用程序存储模型	44
OLE如何使用复合文件	46
设计文件格式	47
处理存储系统中的错误	49
第5章 IStorages和IStreams	51
IStorage类	51
IStream类	59
第6章 访问数据库和已有的文件格式	61
ILockBytes接口	61
支持已有的文件格式	63
使用ILockBytes访问数据库	69
复合文件的性能问题	70

第三篇 容器	73
第7章 建立一个简单的容器	74
什么是复合文档	74
容器的类型	74
容器如何工作	75
应用实例: OLE Publisher	83
OLE用户界面库	87
实现IOleClientSite和IAdviseSink接口	88
剪贴板操作	93
绘制对象	110
双击鼠标激活对象	112
基本容器的用户界面支持	114
我们已学到了什么	116
第8章 链接容器	118
链接如何工作	118
在OLE Publisher中增加对链接的支持	121
链接容器的用户接口支持	122
支持嵌入对象的链接	125
第9章 内置容器	136
内置激活如何运行	136
将内置激活加到OLE Publisher中	143
第10章 支持拖放	161
用户拖放操作的反馈	161
拖放接口	163
拖放操作与剪贴板	165
在OLE Publisher中加入拖放支持	166
建立一个放置对象源	166
建立一个放置对象目的	169
优化拖动操作	177
拖放接口用于普通数据的传输	177
第四篇 服务器	179
第11章 设计一个简单的对象服务器	180
三个基本的服务器接口	180
使用CoLockObjectExternal	184
通知容器	185
服务器的注册文件	186

OLE Draw对象服务器	187
注册文件	205
第12章 建立内置对象服务器	208
内置对象的工作原理	208
显示内置对象	211
建立内置式服务器	220
第13章 建立一个支持链接的服务器	228
理解链接	228
链接的机制	229
实现伪对象	232
附录	235
附录A 注册数据库REG.DAT与REGEDIT.EXE	236
附录B 复合文件的特征	241

第一篇 设计应用程序

本篇介绍了建立OLE应用程序所需的基本信息，包括对元件对象模型COM的详细剖析，以及建立对象必须要理解的重要设计问题。

第1章 介绍OLE 2的概念及特征

第2章 详细讨论了COM

第3章 逐步地展示建立一个元件对象的过程

第1章 OLE 2概论

对象链接与嵌入（OLE）这项技术最初在1990年被提出，它是Microsoft及其它几家软件开发商共同努力的结果。当初的目的是为了对来自多种应用程序的数据能够方便地集成，以生成更丰富的文档。OLE 1是对剪贴板的合理扩展，它允许应用程序从剪贴板上粘贴数据对象，而不必考虑数据的格式。当用户需要编辑或操纵对象时，应用程序将请求该对象的应用程序来提供所需的服务。OLE 1通过一种数据对象的标准格式来实现这一功能。这种格式封装了对象的缓存表示、对象服务器的信息、以及构成对象本身的数据。由于每个数据对象保存了创建者的信息，OLE 1只是一种简单的技术，用户通过用鼠标双击（double-click）一个嵌入对象即可使服务器程序运行起来。OLE更令人关注的是用户界面问题，即当客户程序与服务器程序同时运行时用户界面如何协调。

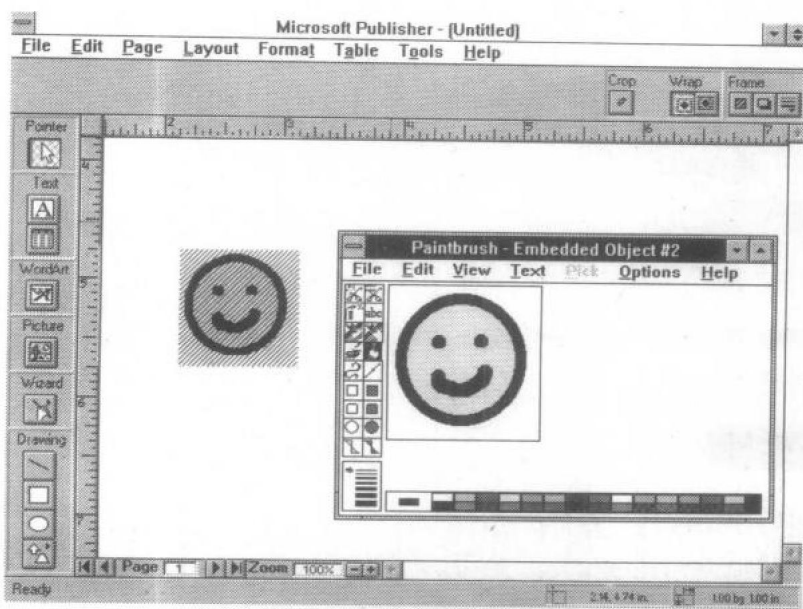


图1.1 OLE 1风格的编辑示例

OLE 2的特征

在与第1个版本向上兼容的同时，OLE 2引入了一系列丰富的新特征，如内置编辑、拖放操作，以及应用程序的自动生成。

内置编辑

内置编辑（In-Place Editing）允许用户在客户程序的环境中编辑嵌入的数据对象。使

用户感觉始终在一个文档内工作，而不再象以前那样要在多个不相容的应用程序之间来回切换。当激活一个内置对象时，服务器将与客户程序针对其菜单和工具控制板的范围进行协调（见图1.2和1.3）。

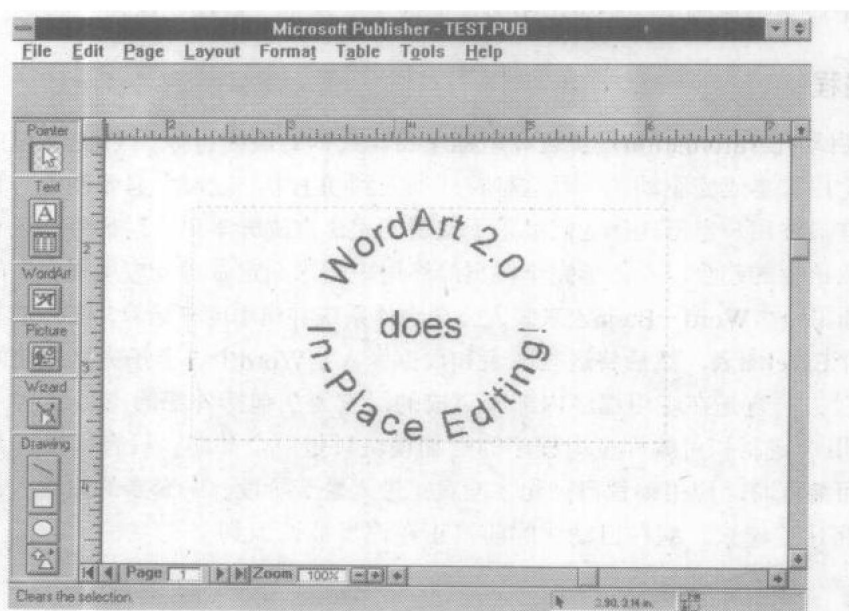


图1.2 带嵌入对象的OLE 2服务器

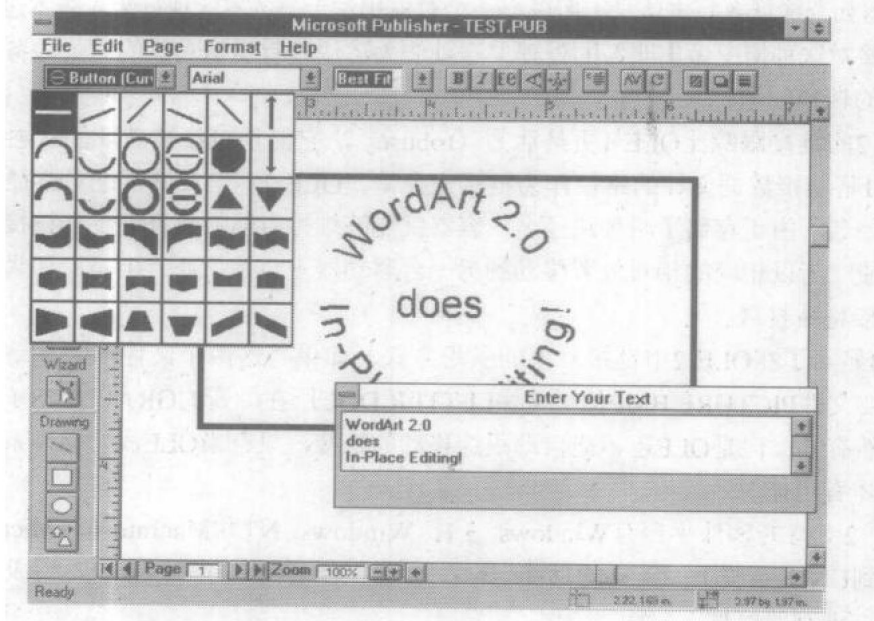


图1.3 内置编辑嵌入对象

拖放

在OLE 1中，虽然用户能够在多个应用程序之间自由地嵌入和链接对象，但要做到这一

点的唯一途径是通过剪贴板 (Clipboard)。这给人一种错觉: OLE只不过是剪贴板的一种扩充。当然, 剪贴板是这个过程的一个重要成分, 但OLE的主要目的在于共享对象。通过引入拖放 (Drag-and-Drop) 操作, OLE 2提供了一条共享对象的新途径。这项技术允许用户将对象从一个应用拖放到另一个应用中, 从而嵌入、拷贝、链接、移动它们。

自动化编程

软件自动化Automation是封装应用程序及其数据的最终目标。OLE编程自动化允许应用程序完全由脚本或宏驱动。一旦这种特征加入到OLE中, 就没有必要制定一种标准的宏语言。它允许最终用户自己选择他们最乐于使用的宏语言或脚本语言。另外, 这种在大的应用程序中嵌入语言的趋势, 开创了完全按照最终用户需求而定做“元应用程序”的可能性。例如, 你可能用一个Word - Basic宏来装入公司会计系统并作季度末结算, 抽取有关的数据后, 再建立一个Excel图表, 然后将这些图表和数据嵌入到Word中一个预先格式化好的报表中。这种过程过去一直是在应用程序内部来完成的, 它无法包括外部的应用程序。

最终用户定制和可编程应用程序的附加模块只是一个开端。只有当操作系统及其实用程序完全可编程时, 应用编程自动化才能真正进入繁荣阶段。一套新的设置、调试、及故障排除工具将建立起来。编程自动化的前景正在逐步被认识到。

其它特征

OLE 2其它的改进及特征包括: 版本管理, 更加灵活的链接跟踪, 及对多平台的支持。对象服务器如今能嵌入版本信息, 所以当对象相对于服务器的现行装入版本过低的时候, 服务器能够对这种情况做出明智的处理。这种能力还使具有类似功能的对象服务器能够处理外部对象的转换问题。

OLE 2的链接跟踪比OLE 1更具活力 (robust), 它能从链接被破坏的状态下更好地恢复。OLE 1将链接数据文件的路径作为链接来存储。OLE 2不但存储了绝对路径名, 还存储了相对路径名。由于存储了两种路径名, 容器就能够维持容器的数据文件与链接文档之间的链接, 即使二者以相同的相对位置移动到另一台驱动器上也是如此。但是, 如果改变文件名的话, 链接将被打破。

图1.4显示了在OLE 2中链接对象的引用方式。如果将这两个文件同时移动到其它某个位置, 只要文件PICTURE.BMP位于文件LETTER.DOC所在位置的GRAPHICS子目录下, 则该链接依然有效。但是OLE还不能自动跟踪移动的链接, 只有当OLE成为操作系统的更完备的部分时才有可能。

OLE 2支持的软件平台有Windows 3.1、Windows NT和Macintosh。Microsoft正积极将它移植到UNIX系统上。支持的功能包括: 所有平台上相似特征的集合, 以及复合文件存储格式的二进制兼容性。

容器和对象服务器

本章把在OLE中参与对话的双方称为客户应用程序和服务器应用程序。用OLE 2的术语来说, 客户应用程序, 即包含着链接或嵌入数据的应用程序, 称为“容器” (container), 而

服务器则处理容器对一些特定数据的所有请求，称为“对象服务器”(object server)。

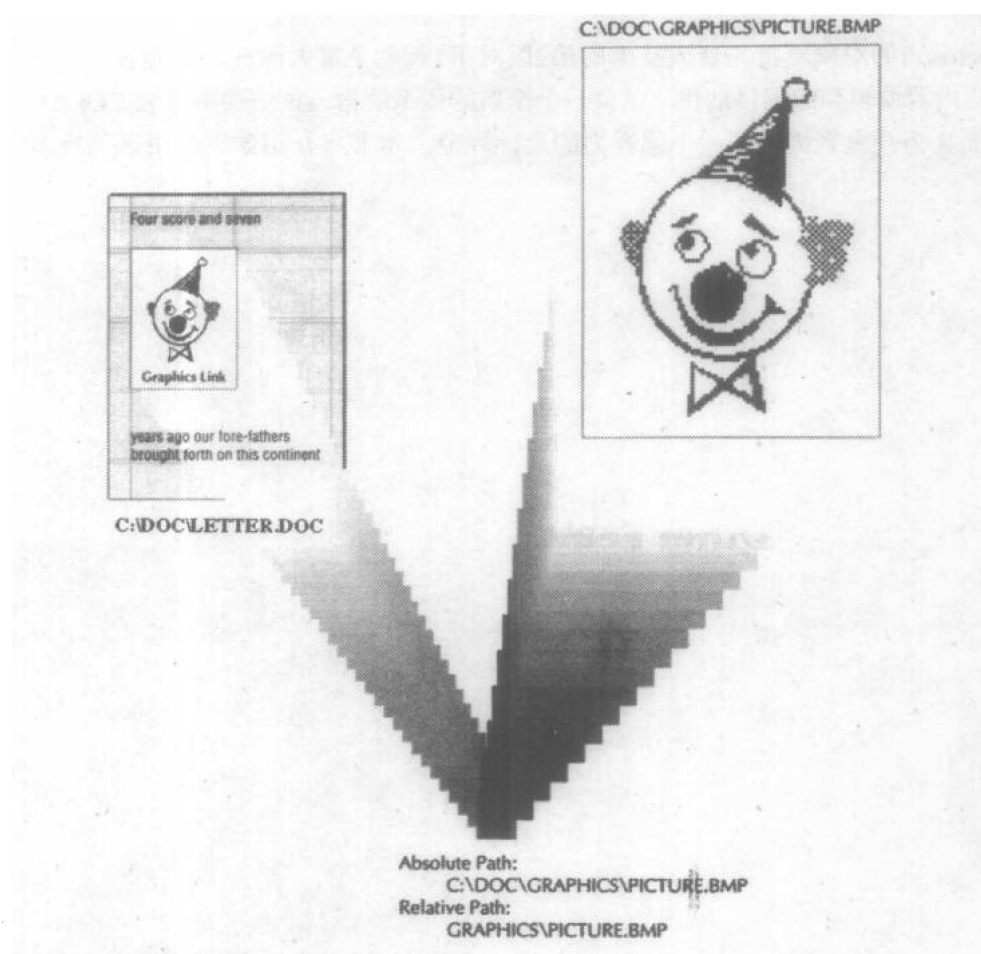


图1.4 带有图形的链接字处理文件

OLE的前景如何

将来，不支持OLE的应用程序会愈来愈少。最终用户已经在使用OLE 1，虽然他们可能还没有意识到这一点。即将面市的Windows 95和以及Windows NT Cario版本都把OLE作为操作系统及系统实用工具的一部分。Microsoft正在加速修改已有的应用程序以增加对OLE的支持。那些包含对OLE某种程度支持的应用程序将会处于十分有利的地位——当它们被装入最终用户的系统时，就会有OLE所支持的最新功能。

Microsoft最近宣称：他们将把一个重要的特征——OLE控制（OCX）加入到OLE中。OLE控制将使开发者能很容易地把诸如电子表控制等复杂的特征，以类似VBX控制的方式加到他们的程序中，OCX将取代VBX控制。

小结

Microsoft的对象链接与嵌入技术的第2版对第1版做了重大改进。它包含了一系列新特征，包括内置编辑和编程自动化，以及一个全新的体系结构。该结构不仅允许OLE可无限地扩展，而且为开发者提供了一个强有力的设计规范。本书将介绍如何使用这项技术。

第2章 元件对象模型

OLE 1的设计不具备OLE 2所要求的各种特性。为了开发OLE 2, Microsoft不得不重新设计对象链接与嵌入的整体结构。这种新结构就是元件对象模型 (Component Object Model), 简称COM。它不但提供了一个实现OLE的框架, 还为一整类新软件提出了一种全新的设计思想。COM是一种面向对象的结构, 它规定了如何使一些小的、定义良好的软件模块 (称为元件 (component)), 在系统中共存并且充分地相互作用。依照这种结构, 可以将若干元件组合起来, 以建立更大的和更复杂的系统。OLE就是完全按照COM的结构来构造的。COM所提供的灵活方法允许软件开发者方便地定义和建造新的即插即用 (plug-n-play) 的元件系统。

COM由一组构造规范和一个动态链接库COMPOBJ.DLL构成, 二者共同工作以协调对象及其消费者 (使用该对象的任何软件) 之间的相互作用。如果希望更多的软件元件, 以一种复杂的方式相互作用, 其体系结构需要解决各种元件之间相互通讯的大量细节问题。

对象和对象服务器

COM将对象定义为“智能数据”, 这是一类知道如何操作和管理自己的数据。对象能被一个或多个客户任务创建和使用。在任何时刻, 对象知道有多少个客户引用它, 所以当所有的客户都完成对它的访问后, 它将自动删除自己。对象定义了一组接口, 通过这组接口与客户任务通信。每个接口由一组方法构成, 这些方法帮助客户完成特定的操作, 或从对象那里请求数据。

对象服务器是对象的软件部分, 它为其所支持的类建立对象。对象服务器还负责管理所有实例化的对象。

服务器的类型

COM有两种不同类型的对象服务器。它们的定义是由实现元件对象的方式决定的。这种实现方式有两种: 动态连接库 (Dynamic-Link Library) 和可执行程序 (EXE)。动态连接库服务器称为进程服务器 (In-Process Server), 可执行程序服务器称为本地服务器 (Local Sever)。

本地服务器

本地服务器是作为可执行程序实现的。所以从客户程序到服务器的连接必须是一个RPC (远程过程调用) 连接。对象的处理进程运行于客户程序的进程空间内, 它把对对象的访问请求, 通过进程的边界传递到实际对象所处的位置——服务器的进程空间中去。对象处理进程充当实际对象的一个代表。在大多数情况下, 客户程序并不知道进程内对象和远程对象 (本