

Windows 多媒体开发工具

ToolBook 的 语 言

Open Script 程序设计方法



秦笃烈 编写

北京希望电脑公司

963352

TP311
5081

Windows 多媒体开发工具

ToolBook 的语言

OpenScript 程序设计方法

秦笃烈 编写

北京希望电脑公司

内 容 简 介

ToolBook是Windows多媒体开发工具，它提供了一种面向对象的程序设计语言OpenScript。利用OpenScript和ToolBook，你可以创制各种应用：把相关的信息链接在一起，完成复杂的数学计算，使图形产生动画，完成希望的任务，即使你以前从未搞过程序设计，也可以用OpenScript写脚本。

本书包括八章三个附录。开头对OpenScript的程序设计方法作了详细介绍。一至六章详述了脚本、脚本基础、消息和性质、脚本的建立及录制和编辑、脚本的调试以及有关脚本的高级技巧；第七章为OpenScript语言成分词典。

本书不仅ToolBook的使用者必读，同时熟悉Windows和C语言等方面的软件人员也同样必读。因此本书既适于软件开发的科技人员，也适于高等院校从高软件工具教学和学习的教师、高年级学生，研究生等参考和阅读。

*

Windows多媒体开发工具

ToolBook的语言

OpenScript程序设计方法

秦笃烈 编写

*

北京希望电脑公司 北京昌平建华印刷厂印刷

开本：787×1092 1/16 印张：21.125字数：515千字

京准印字：3201-90201 内部成本价：15.00元

前 言

ToolBook是美国Asymetrix公司推出的Windows应用生成工具。它被认为是最佳多媒体软件开发工具之一。Windows作为新一代操作系统风靡世界,但在Windows下开发应用要读一系列C语言程序设计、Windows手册以及Windows下程序设计的书。因此,许多用户难以在Windows下开发出高水平软件。ToolBook就是为解决这一难题而设计的。掌握了ToolBook的使用方法,就可免去掌握以上所述技巧,而集中精力于开发自己的软件。Asymetrix公司总裁Allen是当年和Microsoft公司总裁Gates开发BASIC语言时的合作者。这次他倾公司全力开发Windows3.0的生成工具ToolBook是他们的又一次雄心勃勃的合作。本软件推出后已获得美国PC Magazine杂志技术最佳奖,Info world杂志购买者保险标志,Comnuter Language杂志高生产力奖,被Lotus杂志评为最佳软件。1992年10月在美国举行第一次ToolBook及多媒体开发者国际会议。有人认为ToolBook等于Windows文档+SDK+C语言程序设计+面向对象程序设计。因此读通本书的姊妹篇《TooBook使用方法》相当于读一大批书的效率。九十年代是软件工具盛行的时代。人们不掌握软件工具就不易开发高级的应用。本书不仅ToolBook的使用者必读,同时从事软件工设计的人也同样必读,因为ToolBook中包含的一系列方法具有创新之意。ToolBook本身还具提供了一种面向对象的程序设计语言OpenScript,它用于书写对象的脚本。它主要是为高级用户使用。本书对OpenScript的程序设计方法作详细介绍。

秦笃烈

1991.12于静思斋
首都医学院BME系
100054

21348102

目 录

欢迎进入OpenScript世界

- 0.1 你能用OpenScript建立什么? (1)
- 0.2 OpenScript和面向对象的程序设计 (1)
- 0.3 OpenScript的优越性 (3)
- 0.4 如果你是有经验的程序设计人员 (4)
- 0.5 阅读本书的建议 (4)
- 0.6 关于本书中的例子 (4)

第一章 概要

- 1.1 脚本：应用的威力所在 (6)
- 1.2 脚本是什么? (7)
- 1.3 什么是消息以及它们做的工作是什么? (7)
- 1.4 消息和ToolBook对象层次结构 (8)
- 1.5 通过脚本设置性质 (10)
- 1.6 建立脚本 (10)
- 1.7 脚本写作工具 (10)
- 1.8 建立应用的做法 (13)
- 1.9 在ToolBook的样本应用中的脚本构思 (13)

第二章 脚本基础

- 2.1 OpenScript基本概念 (14)
- 2.2 对对象的参考 (15)
- 2.3 值和值的表达式 (21)
- 2.4 变量 (26)
- 2.5 有运算符的表达式 (30)
- 2.6 有字符串说明符的表达式 (34)
- 2.7 函数 (36)
- 2.8 命令 (38)
- 2.9 语句 (41)
- 2.10 控制结构 (42)
- 2.11 处理子：脚本的构件 (45)
- 2.12 专用词小结 (47)
- 2.13 语法小结 (49)

第三章 消息和性质

- 3.1 消息和性质简介 (52)
- 3.2 消息 (53)
- 3.3 内部消息 (53)
- 3.4 从脚本和Command窗口发送消息 (57)

3.5	处理用户自定义消息	(61)
3.6	性质	(62)
第四章	脚本的建立、录制和编辑	
4.1	建立脚本以及对脚本进行工作概述	(72)
4.2	用脚本拷贝对象	(72)
4.3	在Script窗口中进行工作	(73)
4.4	录制脚本	(78)
4.5	检查脚本的语法	(81)
4.6	打印脚本	(82)
第五章	对脚本进行调试	
5.1	什么是调试?	(83)
5.2	在Debug窗口中进行工作	(84)
5.3	设置和撤除中断点	(88)
5.4	跟踪脚本执行	(89)
5.5	显示变量的值	(95)
5.6	理解调试工作	(98)
5.7	节省调试时间的诀窍	(100)
第六章	有关脚本的高级技巧	
6.1	你还能对脚本做些什么?	(102)
6.2	组织你的脚本	(102)
6.3	定义你自己的函数	(105)
6.4	定义你自己的性质	(106)
6.5	利用用户自定义性质做更多的事	(110)
6.6	控制键盘输入	(114)
6.7	使菜单条形区用户化	(119)
6.8	系统书的使用方法	(126)
6.9	利用Windows动态连接库(DLL)	(126)
6.10	利用Windows动态数据交换(DDE)	(127)
第七章	Openscript语汇参考	
附录		
附录A	OpenScript关键词.....	(314)
附录B	对ToolBook使用DLL	(319)
附录C	词汇表	(326)

欢迎进入OpenScript世界

欢迎你学习ToolBook的程序设计语言OpenScript。利用OpenScript为对象建立脚本时，对你的应用能做什么事情以及对你的应用的工作情况如何进行准确的控制都可以有很宽的选择范围。

0.1 你能用OpenScript建立什么？

所谓ToolBook的应用是为特定用途，如培训、信息管理或娱乐而设计的一本或几本书。

利用OpenScript可以建立各种各样的应用包括如下：

- 科学计算应用，能完成许多数学函数计算，计算抵押贷款支付并能包括一个可卷动字段，像打印的加法机纸带那样可供读者对计算过程和结果进行核对。
- 教学应用，如建立讲授元素周期表的书。当读者触发一个元素时，ToolBook就可以显示附有该元素文本原子结构图以及描述该元素文本的页面。
- 作图应用，包括根据读者输入的信息作出的条线图，柱形图，线图，区域图，馅饼图或其它种类图形。在一个页面上，读者输入用于作者的数值，然后触发一个按钮，这时ToolBook会接到一空白页面并画出该图形。

即使你以前从未进行过程序设计，也可以用OpenScript写脚本。脚本是定义对象对于像鼠标器触发这样的特殊事件如何作出响应的指令集。

如果你已经在Lotus 1-2-3、Microsoft Excel或者Word Perfect这样的程序中写过或录制过宏，则你对脚本如何进行工作的理解可以说已先行了一步。书写或录制脚本的过程和书写或录制宏的过程相类似。然而，脚本比宏更为复杂。通过书写宏，可以使已存在的应用更加用户化；而用OpenScript书写脚本则可以建立完全新的应用。

如果你是有经验的程序设计人员，会发现利用OpenScript比利用传统的程序设计语言的一大好处是你具有设计和修改图形用户界面的能力。利用OpenScript建立和修改按钮、字段、图形和菜单要比传统程序设计语言容易得多。这就使OpenScript对建立基于图形的应用原型是理想的，不管你是否是用OpenScript还是用传统语言建立应用的最后版本都可以这样做。在0.4节中还会讲到对程序设计人员特别感兴趣的OpenScript其它方面的细节。

0.2 OpenScript和面向对象的程序设计

OpenScript是一种面向对象的程序设计语言，这种语言由于它比传统的程序设计语言往往能较容易较快地使用而正在迅速流行。

面向对象的程序设计语言对建立应用的任务的处理方法和传统程序设计语言的做法是不同的。传统的处理方法是告诉计算机要执行哪些过程，而面向对象的语言则是告诉对象应做什么。在OpenScript中这些指令是采取对象的脚本的形式。通过对对象的理解以及对面向对象和传统语言之间的差别有更多的了解，你就会理解OpenScript提供的优点。

0.2.1 对象包含数据

如果你已经在ToolBook中作为作者进行过工作，你已经了解按钮、字段、图形和组群都是对象。热字、页面、背景和书也是对象。

在面向对象程序设计中一个重要概念是对象包含数据。这种数据可属于两个范畴：定义对象的默认行为的数据以及作为对象性质预先编程的数据。

当你建立某种类型对象时，它具有该类型所有其它对象所共同的特征。例如，所有按钮具有标识“Button”并以花式圆角矩形建立。当对象第一次建立时它的行为方式是对象的默认行为。默认行为是由对象的类型和它的性质的默认设置决定的。

虽然当对象第一次建立时它的性质设置为特殊值，仍可以对这些值进行修改以改变对象的行为。所有同样类型的对象具有相同的性质，这些性质的每一个都有特定的可能值范围。可以把性质考虑为可以按预定方式进行修改的预编程行为。例如，所有按钮都有性质borderStyle。这个性质对按钮是预先编程的从而它们知道它们可以是矩形、圆角矩形、有阴影、无线电还是核选框按钮。它们知道它们的边框类型只能是这些预定义花式之一。因为这个信息以性质border Style的形式预先通过程序设计赋给按钮，所以它们知道如何使自己从一种花式变为另一种花式。

图0-1展示对象及其脚本。它包括三个按钮。上面的按钮作为对象，它以基于它的类型的特殊方式起作用。中间的按钮的带阴影的外框是它的性质。它提供可预编程的行为并可按预定义方式对它进行修改。下面的一个按钮具有脚本，除了它的默认的以及预定义的行为以外，还起另外一些不同的作用。

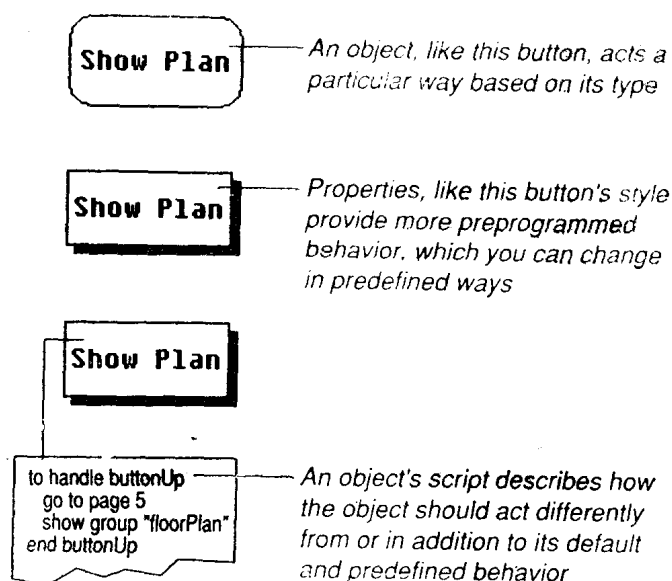


图 0.1 对象和它们的脚本

对象的默认的和预定义的行为是包含在对象中的数据，因此，也是不需要编程的信息，你可以把对象看作是由将它和其它所有对象分开的外壳包围的这种数据。数据的这种封闭有时称为封装。正是这种封装的数据，你可以将脚本附加于它以改变对象的默认的和预定义的行为。

0.2.2 面向对象程序设计和传统的程序设计

懂得ToolBook中的数据封装在对象之中这一点是理解面向对象程序设计语言和传统

程序设计语言之间差别的关键。

0.2.2.1 面向对象程序设计

在OpenScript以及其它面向对象的程序设计语言中,可以通过建立对象以及书写改变对象的默认的以及预定义的行为的指令的方法建造应用。在OpenScript中,这些指令和以对象的脚本的形式出现。脚本定义对象除了它默认的以及预定义的行为外不同的行为方式。换句话说,你只需对没有脚本时的对象工作方式不同的行为编制程序。

从技术上说,脚本是对象的性质之一。然而,因为多数性质可以设置为不同的特殊值,所以脚本在内容和长度上的变化范围相当宽,也就是说可以从许许多多方面改变对象的行为。

ToolBook和运行脚本的对象的通讯方式是发送给对象一个报文。当某个事件出现时,例如读者触发了两次鼠标器左按钮,ToolBook发送一个消息。对象的脚本的一部分内容是定义对象对事件如何响应,ToolBook读取并执行对象脚本的这一部分内容中的指令。如果对象的脚本没有定义如何对事件做出响应,则消息传送到另一个对象。有关发送和接收消息的细节可参阅1.3节和1.4节。

0.2.2.2 传统程序设计

在传统的程序设计语言中,数据和处理数据的指令并不总是很靠近地包装在一起。在传统的程序设计语言中,往往先要定义数据是什么,然后写一些过程解释哪些数据用于过程。对传统的语言来说,不能只对对象的非典型行为编制程序,所以传统的程序通常必须包括数量更多的过程以解释对特定数据做些什么事情。

0.3 OpenScript的优越性

OpenScript语言以及ToolBook的用于写脚本和纠错的工具使得开发应用较为容易。下面列举由OpenScript提供的某些优越性。

0.3.1 程序设计工作量减少

因为对象具有系统赋予的默认的和预编程行为,所以你只要对系统行为或和它们不同的行为编制程序。而且对象的多数行为可以简单地通过修改它的性质值而加以改变。

0.3.2 在小的单元中进行程序设计

从本质上说,脚本是模块化的程序。脚本之间互相独立,它们的目的很清楚:定义对象的行为。此外,脚本可分解为一些称为处理子的模块,每个处理子具有非常特殊的用途。这种模块化方式使程序设计比较容易。

0.3.3 在一个脚本中写的内容可在其它地方重用

你要为某个新对象建立的脚本可能和某个已存在的脚本相似。你可以把已有的脚本拷贝到新对象,然后对脚本作必要的修改。重用已有脚本既可节省时间也能避免出现错误。

当对象接收一个特殊的消息时,对象脚本中的某个特殊的处理子执行。因为可以从各种对象并以多种方式发送消息,你可以从应用中的不同位置触发处理子的执行。因此,不必在几个地方重复设置处理子。

另外,ToolBook的对象是以分层结构排列的,较低层次中的对象可以访问较高层次中对象的脚本。所以通过在顶层附近对象的脚本中放置对若干对象有影响的指令,可以减少需要写的脚本的数量和规模。

0.3.4 容易建立有吸引力的用户界面

利用OpenScript,你可以容易建立有用的以及有吸引力的用户图形界面。因为脚本附属于对象,所以不难在脚本中包括影响对象屏幕外观和屏幕作用方式的指令。也可以设计和修改菜单。这类工作在传统程序设计语言工作量是不小的。

0.3.5 OpenScript的行文很象英语

OpenScript的命令和其它术语经常都是英语单词,而且OpenScript中指令也经常和英语句子相类似。所以利用OpenScript可以按比较自然的方式写作脚本。想一下你用英语怎样向某人写一项指示,那么可以说在OpenScript中写同样指令的方式往往与之相同或类似。例如,如果你要求某人翻动20页书以建立动画效果,你可以用英语说:“Flip 20 pages.”为了告诉ToolBook完成这一同样动作,你可以在书的脚本中使用完全一样的词。flip 20 pages。

0.3.6 OpenScript是可扩充的

虽然OpenScript语言是强有力的,你仍然可以建立和使用你自己的性质、函数和消息扩充它的功能。

0.3.7 ToolBook帮助你纠正脚本错误

当你写脚本时,ToolBook保证你遵守语句书写规则,所以ToolBook总是能理解它们。如果你犯了错误,会得到出错报文并在往前走以前加以纠正。除了检查这些规则以外,ToolBook有一种特殊工具即Debug(调试)窗口,它帮助你把其它脚本方面问题孤立起来并加以解决。

0.4 如果你是有经验的程序设计人员

如果你是有经验的程序设计人员,将会发现对OpenScript以下功能特别有兴趣的:

- OpenScript没有数据类型。数据是根据数据使用时所用的命令、函数以及运算行加以解释的。你不必指定值的类型,也不必指明特定变量能够包含的数据类型。
- 通过写自己的函数和性质或通过使脚本的部件对你自己建立的而不是OpenScript固有的消息作出响应就能扩充OpenScript的功能。通过用C,Pascal或汇编程序设计语言编写过程并作为OpenScript的Windows的动态连接库(DLL)访问它们。
- OpenScript并没有长的编译周期,它在每逢你书写或修改脚本时,以增量方式进行编译。因为脚本的执行和脚本的编译是独立的,所以使性能更为改善。
- OpenScript包含集成化调试工具,这些工具可对执行中的脚本进行访问。

0.5 阅读本书的建议

不论你的经验水平如何也不论你的学习风格如何,ToolBook的系列手册都考虑到对OpenScript学习的需要。表0-1对查找有关OpenScript不同类型信息提供一些建议。

0.6 关于本书中的例子

在本书的脚本例子中使用一些约定。

0.6.1 脚本例子中的注解

许多例子都包含注解,你可以将它们放在脚本中使得你和其它作者容易理解。脚本中的注解总是以两个连字号开始。例如:

表 0.1 关于学习OpenScript的建议

要学习...	可参阅...
OpenScript的基本内容	第 1, 2, 3 章
编辑和录制脚本	第4章
发现和纠正脚本错误	第 5 章
每个命令、函数、消息、性质、运算符、常数、特殊用语以及使用格式	第 7 章
写脚本（包括定义自己的函数和性质）的技巧	第 6 章

```

to handle buttonUp
  go to page "waves"    --Goes to the page named waves
  flip 20 pages         --This creates an animation effect
end buttonUp

```

0.6.2 脚本例子中的大小写

在本手册的例子中，某些字母大写使OpenScript的专用语更为可读。例如：

```

buttonup
openFile
SYSHotwordsShown

```

几乎在所有场合下，OpenScript允许对脚本中的专用语均用大写字母，均用小写字母或二者的任意组合。

有一种情况你必须对大小写要十分仔细，即你要ToolBook完全按脚本中写出的样子去读和使用特殊字符。在脚本中，你总是把这样的字符放在引号内，这时ToolBook就知道按字母去阅读它们。例如：

```
put "Mr.Jonathan MacIntyre" into command Window
```

这个语句把*Mr.Jonathan MacIntyre*按所示大小写放入Command（命令）窗口。

第一章 概 要

利用OpenScript和ToolBook, 你可以创制各种应用: 把相关的信息链接在一起, 完成复杂的数学计算, 使图形产生动画以及完成任何希望完成的任务。

本章讲解可以用脚本做的事情, 脚本如何工作以及可用于编制脚本的工具。

1.1 脚本: 应用的威力所在

脚本是ToolBook应用的能源所在。通过建立脚本, 可以开发TooBook的应用对一群大学生进行基础化学考试, 可以解释你的公司向雇员提供的实惠, 也可以充当另一项应用(如数据库)的图形界面。

下面是ToolBook可以通过脚本完成的各种动作的部分清单:

- 利用读者在字段中输入的数据计算各种数量并根据计算结果完成相应的动作。
- 使图形产生动画。
- 响应读者的动作使对象显示和隐藏。
- 选择菜单命令和模拟击键。
- 增加或删除菜单和菜单项以及建立客户对话框和报文框。
- 向读者提一个问题并根据所作响应完成一项动作。
- 显示字段中的文本并对它格式化。
- 修改对象的性质。
- 检索诸如当前日期的系统信息。
- 运行其它ToolBook示例和其它Windows应用。

可用几种方法建立脚本, 包括在样本应用中拷贝对象以其脚本, 修改对象脚本来满足新的需要, 录制动作以及通过打入语句编写脚本。

表1.1列出本章会遇到的一些术语的解释。

表 1.1 和OpenScript有关的术语

术语	定义
对象 (Object)	按钮、字段、热字、图形、组群、页面、背景和书; 任何对象可以有脚本。所有对象具有性质。
性质 (Property)	对象的属性。性质是可以按预定义方式修改的预编程行为。
脚本 (Script)	一系列OpenScript语句, 它分为处理子的各个部分, 处理子定义对象对特定事件作出的响应。
消息 (Message)	当事件出现时发送给对象的讯息。事件可以是由读者或作者给出的鼠标器或键盘动作, 也可以是ToolBook的内部动作。
消息处理子 (Message handler)	对象脚本中的一组指令或语句, 告诉ToolBook收到特殊消息时ToolBook要完成的动作。
对象层次结构 (Object hierarchy)	为了查找相应的消息处理子而规定的消息在对象之间传送的顺序。

1.2 脚本是什么？

脚本是一组指令，它定义针对特定事件对象应具有的行为。脚本中的每条指令称**语句**。

语句可概括用途而进行分组。这些语句组称为**处理子**。例如，可以写一个处理子，当读者触发某个按钮时它计算纯收入。共有三种处理子：消息处理子，to set处理子以及to-get处理子。

最常用的处理子是消息处理子。**消息处理子**是一组语句，它告诉ToolBook当接到某项特定消息时所完成的动作。

图1-1展示脚本的基本结构。它包含三个消息处理子，分别针对消息buttonUP, buttonDown, buttonDoubleClick。处理子分别由一组语句组成，每一行代表一个语句。

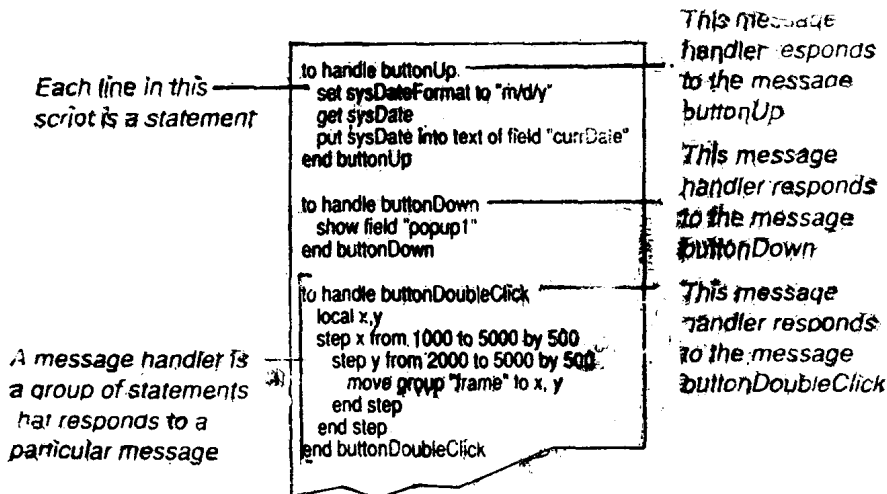


图 1.1 脚本的基本结构

1.3 什么是消息以及它们做的工作是什么？

消息是事件出现时发送给对象的讯息。**事件**是由作者或读者给出的鼠标器或键盘动作或者是ToolBook的内部动作。

例如，当你进入书中某个页面时，ToolBook发送消息enterpage给该页面。当你触发某个图形对象时，按下鼠标器按钮时ToolBook发送消息buttonDown，而松开鼠标器按钮时发送消息buttonUp。

图1.2展示事件出现时向对象发送消息的过程。当读者触发按钮Calculate Tax时，ToolBook为该按钮送一个消息buttonDown，然后再送消息buttonUp。当指示器进入字段Tax时，ToolBook向这个字段送一个消息mouseEnter。

当对象接收到消息时，ToolBook在对象的脚本中查找与该消息对应的消息处理子。如果ToolBook找到相应的消息处理子，ToolBook就执行，或者运行消息处理子中的语句。通过在脚本中写处理子，可以使对象按照对特定事件发生时[你希望]的方式具有它的行为特征。

由执行消息处理子而引起的动作也可以进而使ToolBook向本书或另一本书中的其它对象发送其它消息。这样，一个事件可以导致另一事件，一次鼠标器触发可为一本书甚至几本

书产生活力。

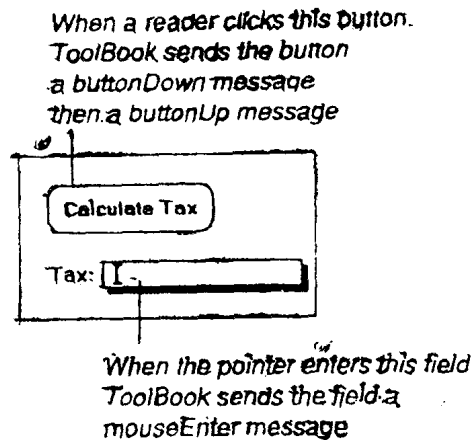


图 1.2 当事件出现时向对象发送消息

图1.3展示当对象接收到消息时发生的事。当按钮Calculate Tax收到消息button Up以后，ToolBook在该按钮的脚本中查找描述对这一消息如何作出反应的处理子。

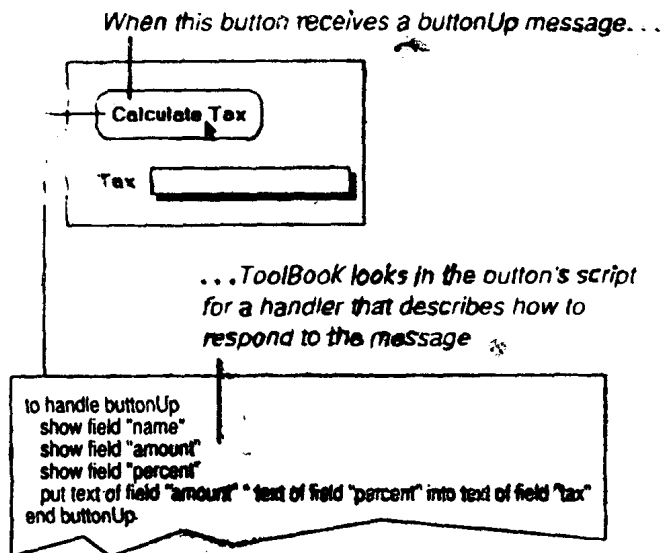


图 1.3 当对象收到消息时发生的事情

1.4 消息和ToolBook对象层次结构

如果对象接收到一个消息而且它的脚本中并不包含针对该消息处理子，则ToolBook把消息传送给对象层次结构，直到它发现相应的处理子为止。**对象层次结构**决定在查找相应处理子过程中消息在对象之间的传递顺序。例如，如果某字段的脚本并不包含针对发送给该字段的消息的消息处理子，则该消息进入该字段所在组群，如不在组群中则进入字段所在页面，依此类推在对象层次结构中向上搜索。

如果在层次结构中对象的脚本中ToolBook未发现针对某消息的相应处理子，该消息进入ToolBook本身。ToolBook有时作为系统参考，它位于层次结构的顶部。

图1-4展示ToolBook的对象层次结构。该结构的左下枝为前景对象，右下枝为背景对象。

系统包含一些处理子，它们的指令代表对内部消息的默认响应。**内部消息**是指作为OpenScript语言一部分提供的消息，如buttonUp，enterBook，make以及destroy。当系统接收到一项内部消息时，ToolBook按对该消息的默认响应作出反应。

你也可以发送自己的消息，它们通常由你写的处理子进行处理。对用户定义的消息，系统并不具有默认的响应处理子；因此，如果某个用户定义的消息遍历系统的所有通路时，会出现错误。有关用户定义消息的细节，可参阅3.5节。

1.4.1 系统书和Windows动态连接库

在图1.4中展示的对象层次结构包含一些你可能不熟悉的术语。**系统书**(syetem book)

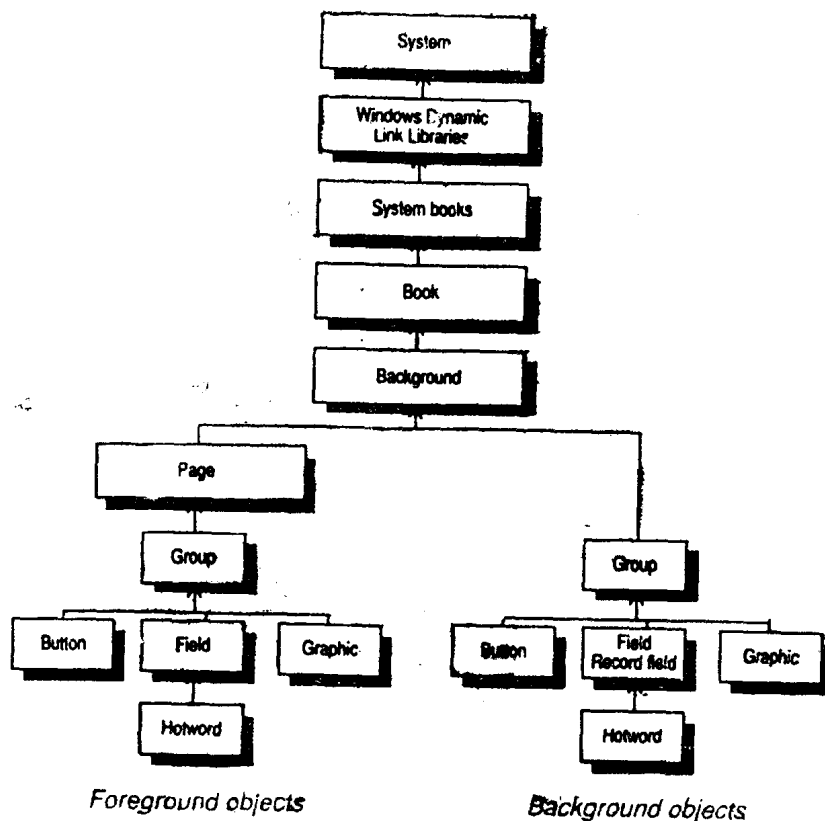


图 1.4 ToolBook的对象层次结构

是这样的书，它的脚本在其它ToolBook应用运行时可以利用。你可以指定任何书为针对所有其它书或针对特定书的系统书。其细节可参阅6.8节。

Windows动态连接库 (DLL) 是一些用C, Pascal或汇编语言写的并可以由ToolBook访问的过程。其细节可参阅节6.9。

1.4.2 有效的使用对象层次结构

消息总是沿同样方向传送：从它们的入口进入层次结构然后沿对象层次结构向上。通过

按你的要求对书进行仔细规划，可以把消息处理子放在层次结构中最有利的位置，避免在若干对象的脚本中重复出现相同的处理子。其细节可参阅6.2节。

1.5 通过脚本设置性质

通过参考对象的性质，脚本可以利用对象预定义的行为。**性质是对象**可以按预定义方式进行修改的属性。例如，矩形的性质pattern决定在矩形内部显示哪种图案。改变对象行为的一种方法是改变它的性质值，我们可以在Properties（性质）对话框或者在脚本中写入特殊语句做这件事。

例如，你可以写一些语句指明如果读者触发标识为Red的按钮，当前页面背景的性质fill Color的值应予以改变，使得背景出现红色。如果读者触发标识为Yellow的按钮，该背景fillColor的值应该改变使背景呈现黄色。

1.6 建立脚本

你可以用五种方式建立脚本。可以分别地使用这些方法，也可以组合地利用它们。

- 拷贝已存在的对象。新的对象将具有和原来对象相同的脚本。
- 拷贝或裁剪已存在对象的部分脚本并将它们贴附到新对象的脚本。
- 利用Button或Hotword Properties对话框用按钮和热字把页面链接在一起。ToolBook会建立在链接的页面之间进行航行的脚本。如果希望把更复杂的信息加到某一个简单的航行脚本，可以将其它语句写入或拷贝到该脚本。
- 用脚本记录器录制动作。脚本记录器把鼠标器和键盘动作的结果翻译成OpenScript语句，你可以把它们贴附到对象的脚本。对一个简单的脚本，你需要做的全部工作是录制动作。对较复杂的脚本，例如脚本是在某种条件下ToolBook只执行一部分消息处理子而在另一种条件下执行其它部分，你就需求使用录制和写作的组合。
- 利用OpenScript写作整个脚本。这一方法提供最强的力量和灵活性。

1.7 脚本写作工具

ToolBook提供若干可帮助你建立脚本的工具：

- Script窗口
- Command窗口
- 脚本记录器
- Debug窗口

1.7.1 Script窗口

在Script窗口中你可以观察、写作、编辑对象的脚本以及把录制内容贴附到对象的脚本。ToolBook也从Script窗口进行语法检查，也就是检查脚本语句放在一起的方式。对脚本所做的多数工作是在Script窗口中做的。图1.5展示的是脚本窗口。在标题条形区内显示对象名称或ID，它的下面是Script窗口自己的菜单条形区。两次触发标题条形区左端的菜单控制框可使Script窗口消失。在脚本窗口内可以将插入点移动到需要打入或贴附语句的地方，而拖动它的任何边框可以改变它的大小。

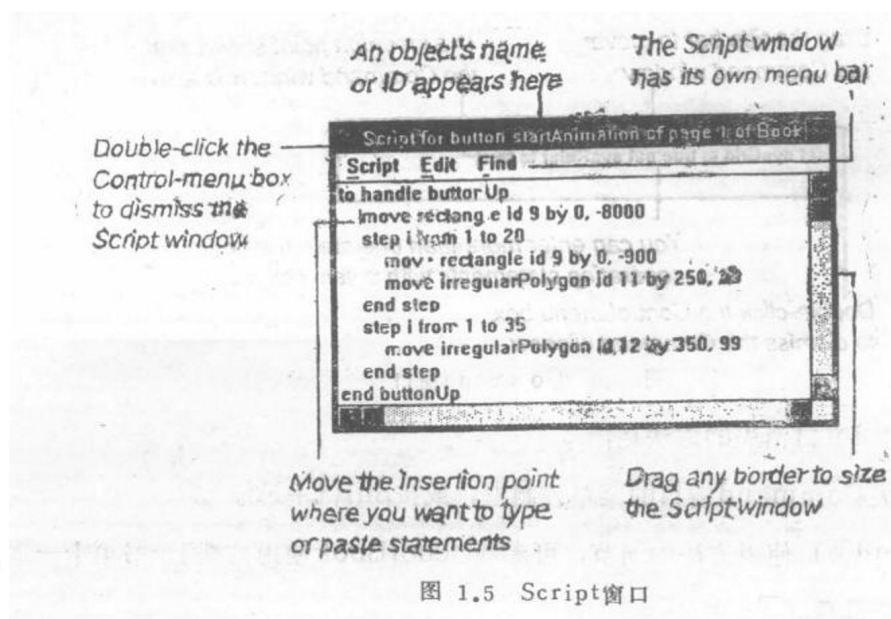


图 1.5 Script窗口

表1.2说明如何显示针对各种对象的Script窗口。有关Script窗口用法的细节可参阅4.3节。

表 1.2 显示Script窗口

为...显示Script窗口	要执行的是...
按钮、字段、 图形、热字或组群 页面	选择该对象，从Object菜单选取适当的Properties命令，然后选择Script。也可以按Ctrl键并两次触发该对象。 切换到前景，从Object菜单选择Page Properties，然后选择Script，或者切换到前景并按Shift+F7。
背景	切换到背景，从Object菜单选择Background Properties，然后选择Script，或者切换到背景并按Shift+F7。
书	从Object菜单选择Book Properties，然后选择Script，或者按shift+F8。

1.7.2 Command窗口

Command窗口是一种窗口，可以在其中打入并执行在脚本行文外部的OpenScript语句。当按Enter时，ToolBook就执行这些语句。因为语句执行时可以看到发生的事，在Command窗口中打入命令是熟悉OpenScript语言并对脚本的部件进行测试的好方法。

图1.6展示的是Command窗口，两次触发菜单控制框可使Command窗口消失。窗口内出现插入点表明Command窗口处于激活状态。拖动标题条形区可使Command窗口移动。利用分号将语句分隔开可以输入不止一个语句。要显示Command窗口的方法是：

- (1) 从Window菜单选择Command，或按Shift+F3。