

万水创作效果百例丛书

Authorware 7.0 精彩设计百例

童 剑 等编著

中国水利水电出版社

内 容 提 要

Authorware 是多媒体工具软件供应商 Macromedia 公司的优秀产品, 该软件功能强大, 应用范围涉及教育、娱乐、科学等各个领域。最新推出的 Authorware 7.0 版本较以前的版本有了很大地改进, 其功能更加趋于全面化。

本书通过 100 个精彩实例, 由浅入深地介绍了 Authorware 7.0 设计多媒体的方法与实用技巧。为了方便读者学习与深入理解本书将这 100 个实例分为 3 篇, 即基础篇、中级篇和高级篇。在每个实例中, 除了介绍 Authorware 7.0 中所提供的基础知识点及编程技巧外, 还介绍了 Authorware 7.0 在其他方面的重要应用及开发技术。

本书附赠一张光盘, 其中包括全部 100 个实例的源代码及可执行文件。

本书体系结构清晰, 解释说明详尽, 开发实例典型, 适合于计算机多媒体制作用户、电视片头制作人员以及对 Authorware 7.0 感兴趣的用户阅读, 也可作为 Authorware 7.0 培训的参考书。

图书在版编目 (CIP) 数据

Authorware 7.0 精彩设计百例/ 童剑等编著. - 北京: 中国水利水电出版社, 2004

(万水创作效果百例丛书)

ISBN 7-5084-2074-8

A... 童 ... 多媒体—软件工具 Authorware 7.0 .TP311.56

中国版本图书馆 CIP 数据核字 (2004) 第 031958 号

书 名	Authorware 7.0 精彩设计百例
作 者	童 剑 等编著
出版 发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机) 68331835 (营销中心) 82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787mm × 1092mm 16 开本 18 印张 398 千字 2 彩插
版 次	2004 年 5 月第 1 版 2004 年 5 月第 1 次印刷
印 数	0001—5000 册
定 价	32.00 元 (含 1CD)

凡购买我社图书, 如有缺页、倒页、脱页的, 本社营销中心负责调换

版权所有 · 侵权必究

前 言

Authorware 是多媒体工具软件供应商 Macromedia 公司的优秀产品，该软件功能强大，应用范围涉及教育、娱乐、科学等各个领域。最新推出的 Authorware 7.0 版本较以前的版本有了很大的改进，其功能更加趋于全面化。

本书通过 100 个精彩实例，由浅入深地介绍了 Authorware 7.0 设计多媒体的方法与实用技巧。为了方便读者学习与深入理解 Authorware，本书将这 100 个实例分为 3 篇，即基础篇、中级篇与高级篇。

基础篇重点是让读者通过这些实例能对 Authorware 7.0 的基本知识进行全面的学习和了解，因此安排了 50 个实例来对 Authorware 的文本、图形、动画、交互、声音、视频、Flash 动画等方面的知识进行介绍，相信读者能够通过这些实例对 Authorware 7.0 的基本知识有深刻的理解。

中级篇安排了 25 个实例来介绍用户在进行多媒体创作过程中会经常使用到的一些技巧，这 25 个实例能够为用户将来的多媒体创作增添不少素材。

高级篇主要介绍利用 Authorware 7.0 中的库和知识对象等高级对象来制作实例，并且也安排了一些较大型的实例进行介绍。

在每个实例中，除了介绍 Authorware 7.0 中所提供的基础知识点及编程技巧外，还介绍了 Authorware 7.0 在其他方面的重要应用及开发技术。

本书最突出的特点，是通过大量的实例来体现的实战性，每一个实例都具有很强的应用背景。基本知识点部分，能够使读者明确实例操作的目的和知识要点，做到有的放矢；实例部分，讲述详细、语言生动、可操作性强，读者可以对照着进行练习，从而得到最佳的学习效果。

本书附赠一张光盘，其中包括全部 100 个实例的源代码及其他素材文件。

本书体系结构清晰，解释说明详尽，开发实例典型，适合于计算机多媒体制作用户、电视片头制作人员以及对 Authorware 7.0 感兴趣的用户阅读，也可作为 Authorware 7.0 培训的参考书。

本书由水利水电出版社策划，童剑执笔编写，张文松、陈杰、李海燕、赵威、王进、赵依、陈文平、黄卓、卢福子、戴威、杜艳萍等同志在整理材料方面给予了作者很大的帮助。

由于时间仓促，加之编者的水平有限，缺点和错误在所难免，恳请专家和广大读者不吝赐教，批评指正。

编者

2004 年 1 月

目 录

前言

第一篇 基础篇

实例 1	第一次使用 Authorware—你好	2
实例 2	图标操作 (1)	6
实例 3	图标操作 (2)	3
实例 4	插入图片	11
实例 5	插入外部文本	13
实例 6	奥运 2008	16
实例 7	我的小屋	19
实例 8	白雪公主	21
实例 9	精彩射门 (1)	25
实例 10	精彩射门 (2)	27
实例 11	可爱的机器猫	30
实例 12	Flash 动画	33
实例 13	按钮交互 (1)	35
实例 14	按钮交互 (2)	37
实例 15	按钮交互 (3)	39
实例 16	数学题	41
实例 17	提示向导	44
实例 18	地球 (1)	47
实例 19	地球 (2)	49
实例 20	动物乐园	51
实例 21	鸟叫声	54
实例 22	巧对唐诗	56
实例 23	小鸟图案的定向移动	60
实例 24	雨中登泰山	62
实例 25	家庭影院	64
实例 26	自定义 Beep 下拉菜单 (1)	67
实例 27	自定义 Beep 下拉菜单 (2)	69
实例 28	健康咨询	71
实例 29	智力问答	75
实例 30	用户登录	78

实例 31	空当接龙	80
实例 32	密码箱	82
实例 33	冒险岛	84
实例 34	拼图游戏 (1)	86
实例 35	拼图游戏 (2)	88
实例 36	纪念光盘片头曲	92
实例 37	滚动字幕	94
实例 38	动画播放	98
实例 39	图片演示	100
实例 40	纸牌游戏	102
实例 41	试题抽调 (1)	104
实例 42	试题抽调 (2)	107
实例 43	图片欣赏	109
实例 44	纪念光盘 (1)	112
实例 45	纪念光盘 (2)	114
实例 46	纪念光盘 (3)	117
实例 47	倒计时	120
实例 48	文件读取	123
实例 49	游标驱动	125
实例 50	时间和日期显示	130

第二篇 中级篇

实例 51	条形渐变	135
实例 52	径向渐变	138
实例 53	扇形渐变	140
实例 54	抢答题	142
实例 55	控制面板	145
实例 56	浏览器	148
实例 57	进度条	152
实例 58	插入幻灯片	154
实例 59	自定义标题栏	158
实例 60	文本闪烁	162
实例 61	控制音量	164
实例 62	相册	168
实例 63	“关于”对话框	171
实例 64	音乐播放器	173
实例 65	掷骰子游戏	176

实例 66	显示文件路径	178
实例 67	屏保程序 (1)	181
实例 68	屏保程序 (2)	184
实例 69	复选框	186
实例 70	超文本链接	189
实例 71	页面查询 (1)	194
实例 72	页面查询 (2)	197
实例 73	标尺程序	199
实例 74	判断网络连接	203
实例 75	星形线	206

第三篇 高级篇

实例 76	调色板	210
实例 77	文本输入框	214
实例 78	不规则窗口 (1)	216
实例 79	不规则窗口 (2)	220
实例 80	右键菜单 (1)	222
实例 81	右键菜单 (2)	225
实例 82	绘图程序	228
实例 83	寻找光驱	232
实例 84	转换音乐格式	234
实例 85	移动打靶 (1)	237
实例 86	移动打靶 (2)	239
实例 87	控制电影播放 (1)	242
实例 88	控制电影播放 (2)	246
实例 89	登录程序	249
实例 90	文本对接	252
实例 91	网址簿 (1)	255
实例 92	网址簿 (2)	257
实例 93	Flash 播放器 (1)	260
实例 94	Flash 播放器 (2)	263
实例 95	计算器 (1)	265
实例 96	计算器 (2)	267
实例 97	检测窗口大小 (1)	269
实例 98	检测窗口大小 (2)	272
实例 99	数据库操作 (1)	274
实例 100	数据库操作 (2)	277

第一篇 基础篇

Authorware 7.0 的出现使得许多非专业人员创作交互式多媒体软件成为可能, Authorware 应用相当广泛, 可以用于产品的演示过程, 展示各种工具的使用方法, 制作各种电子出版物和产品目录等等。目前使用 Authorware 7.0 制作的程序已经广泛应用于 CAI 教学和各个商业领域。

要掌握 Authorware 7.0 这样大型的多媒体开发工具, 就必须从最基础的内容入手, 在本篇中, 读者通过我们精心挑选的 50 个实例, 对 Authorware 7.0 的各个基本图标以及 Authorware 变量有初步的了解, 对 Authorware 开发环境有总体的印象, 为将来更进一步的学习打好基础。

Let's GO! 



实例

1

第一次使用 Authorware—你好



实例说明

本例主要讲解使用 Authorware 的基本操作，包括如何启动、退出 Authorware；如何保存文件、打开一个已有的文件；Authorware 界面的简单介绍；使用显示图标来显示文字；让程序运行等内容。本实例有些内容在后续章节还会详细介绍。

主要知识点：显示图标。



操作步骤

① 在 Windows 任务栏中单击“开始” “程序” Macromedia Macromedia Authorware 7.0，如图 1.1 所示。

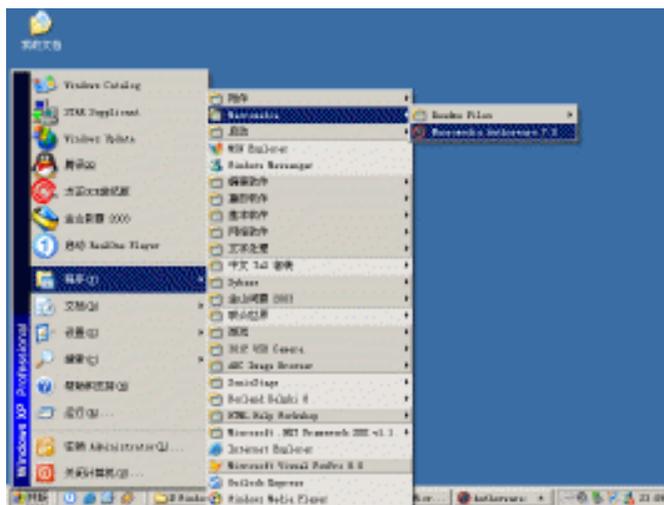


图 1.1 启动 Authorware 7.0

② 此时会出现 Authorware 的欢迎界面，然后出现如图 1.2 所示的画面。把 Show this dialog at startup（启动时显示此对话框）前的对勾划掉。然后单击 None 按钮，进入 Authorware。

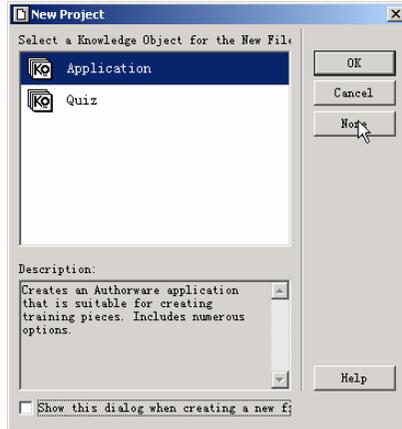


图 1.2 Authorware 启动对话框

③ 出现 Authorware 的工作界面，如图 1.3 所示。

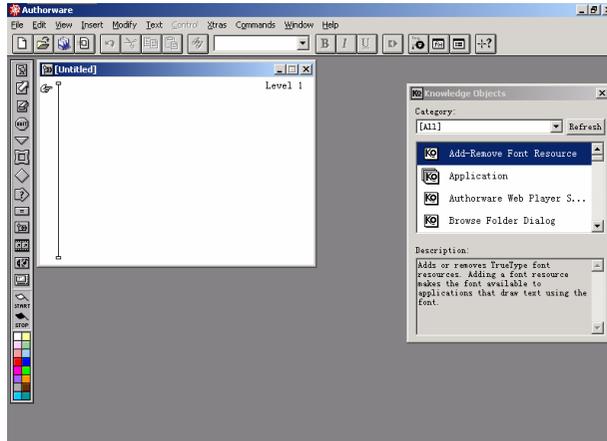


图 1.3 Authorware 的工作界面

④ 设计图标条上最上面的一个是 Display (显示) 图标 ，把它拖到中间的流程线上，如图 1.4 所示。现在来设计这个显示图标，双击鼠标打开它。

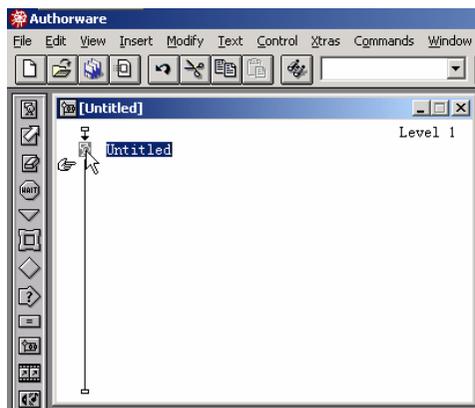


图 1.4 拖动一个显示图标到流程线上



⑤ 在弹出的 Presentation Window (演示) 窗口中单击 ，然后单击中间的空白设计窗口，输入“你好！”，如图 1.5 所示。

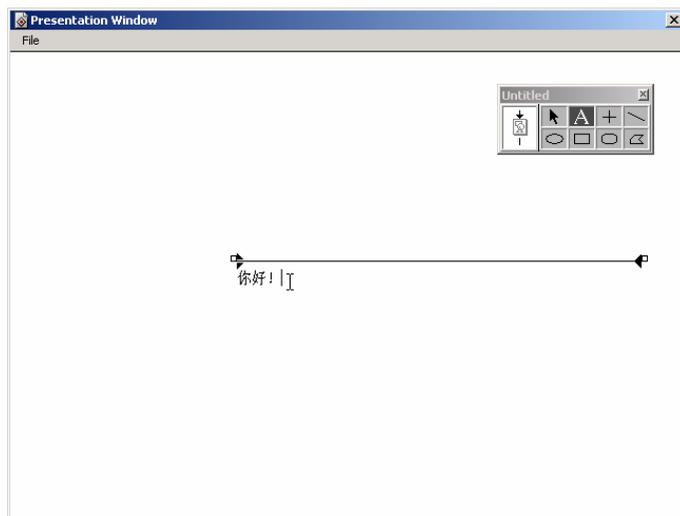


图 1.5 输入“你好！”文字

⑥ 单击面板上的关闭按钮退出设计模式，单击工具栏上的 Restart (启动)  按钮，可以预览本实例的第一个画面。单击右上角的关闭按钮结束程序运行。

⑦ 保存文件，选择 File (文件) Save (保存) 打开 Save File As (保存文件) 对话框，如图 1.6 所示。在文件名里面键入“01 你好”，单击“保存”按钮。

⑧ 下面来练习打开一个已有的文件，单击 File Exit (退出)，或者单击 Authorware 右上角的关闭按钮，退出 Authorware。然后再重新启动 Authorware。

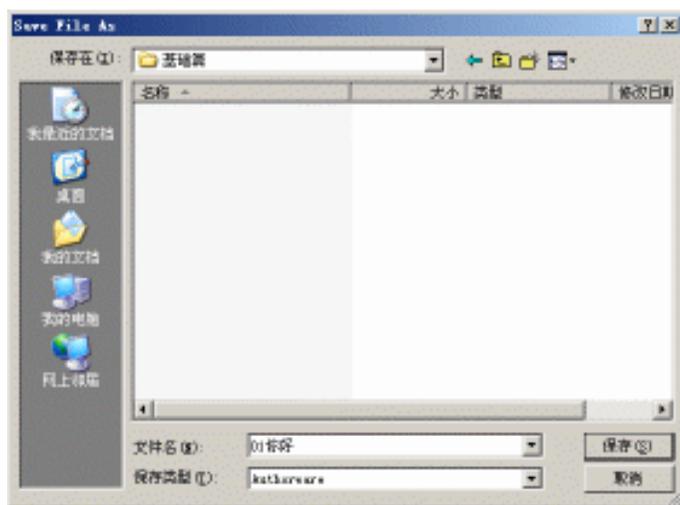


图 1.6 保存文件对话框

⑨ 选择 File Open (打开) File，弹出 Select a file (打开文件) 对话框，选择刚才保存的文件，单击“打开”按钮，如图 1.7 所示。这样就可以打开文件了。



图 1.7 打开文件对话框



技巧

启动 Authorware 还有一种方法，通过资源管理器或者我的电脑直接找到 Authorware 安装目录，如果安装时没有改变路径的话，一般在：C:\Program Files\Macromedia\Authorware 7.0\



心得体会

通过本实例，要掌握如何启动、退出 Authorware，如何保存、打开文件，另外，要对 Authorware 的工作界面有简单的了解。

记住一些快捷键能够让工作更有效率，如：Ctrl+O——可以调出打开文件对话框，Ctrl+N——新建一个文件，Ctrl+S——保存文件，注意要养成保存文件的好习惯。



实例

2

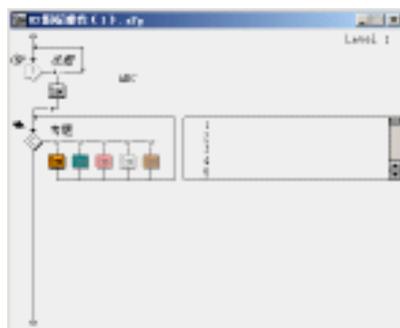
图标操作（1）



实例说明

该实例主要介绍图标的各种操作，在具体学习创建实例之前了解 Authorware 的基本操作是很必要的。

主要知识点：显示图标、图标操作。



操作步骤

- ① 新建文件，保存为“02 图标操作(1).a7p”。
- ② 通常，流程线上的图标是通过鼠标将其从图标栏上拖到流程线上的，拖动后的效果如图 2.1 所示，用鼠标按住图标栏上的 Display（显示）图标，然后将其拖至流程线上松开。这时，图标就被放置到流程线上了，同时，在图标的右面出现图标的默认名称 Untitled。用户可以给图标标记合适的名称，按下 BackSpace 键，将 Untitled 删除，然后就可以在光标停留处输入该图标的名称了。

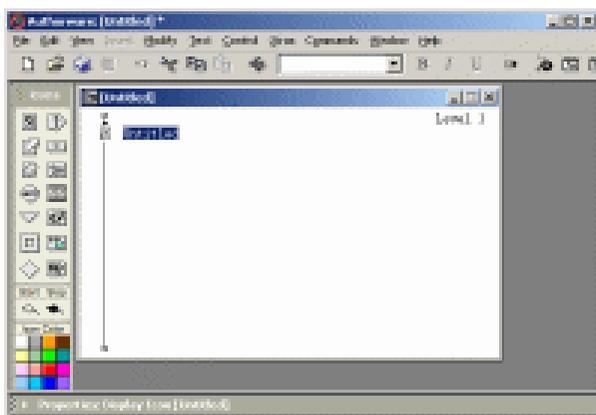


图 2.1 图标命名



说明

流程线上的图标可以是任意多个，并且图标的名称可以是重复的。不过，为了以后的检查方便，最好给图标以不同的名称。

- ③ 在流程线的编辑操作中，最常用的是移动、插入、删除、剪切、复制等操作，进

行以上操作时，首先必须要选择图标对象。单个图标的选择最简单，只要单击它即可。如果要选择多个相邻的图标，可以直接用鼠标画定图标范围进行选择。如图 2.2 所示，要选择 3 个群组图标，用户可以在第一个图标的左上方按住鼠标，然后向右下方拖动，此时，窗口上出现一个矩形虚线框，松开鼠标后，矩形虚线框内的图标颜色变暗，表示被选中。

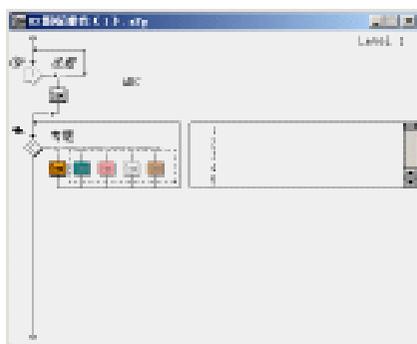


图 2.2 选择图标



技巧

如果要选择互不相邻的图标，可以先按住 Shift 键，然后在流程线上逐个单击需要选择的图标，选择完毕，松开 Shift 键。

④ 接下来，如果要将选择的图标转移到流程线的下方，首先单击工具栏上的剪切按钮 （如果要复制选择的图标，则单击工具栏上的复制按钮 ），此时图标消失（保存在剪贴板上），然后单击流程线的最下方，“手”型指针  随之下移，然后单击工具栏上的粘贴按钮 ，3 个图标就会被复制到当前位置上。

⑤ 图标的删除操作非常简单，选择好图标对象，然后直接按 Delete 键，所选的图标对象就被删除了。移动图标同样简单，先选择要被移动的图标，然后直接将它拖到流程线上的合适位置松开鼠标，图标就移动到当前位置了。Authorware 还提供了恢复功能，如果某一步操作失误，只要单击工具栏上的撤消按钮就能恢复到前一步的状态。



说明

移动图标一般只能对单个图标进行，多个图标只能采取剪切/粘贴操作或先进行图标归组再移动，然后再解组。



技巧

在流程线的编辑中，设计窗口的大小是可以调整的。当鼠标移至设计窗口的边框时，指针将变为双箭头。如图 2.3 所示，当鼠标移向窗口的 4 个角时，指针变为倾斜的双箭头，此时拖动鼠标，窗口会放大和缩小。当鼠标移向窗口的 4 个边框时，指针会变为垂直或水平的双箭头，此时拖动鼠标，窗口只会改变高度或长度。

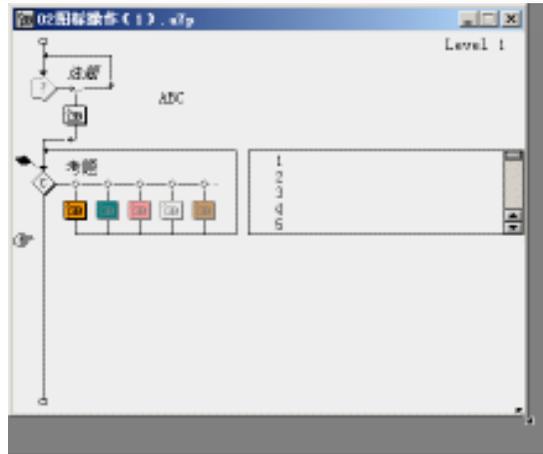


图 2.3 缩放窗口

实例

3

图标操作 (2)



实例说明

本例主要讲解图标的归组在流程线编辑中的作用,使用这一功能,Authorware 会将令人眼花缭乱的流程图变得十分清晰,这也在一定程度上简化了设计窗口的空间。

主要知识点:图标归组。



操作步骤

- ① 新建文件,保存为“03 图标操作(2).a7p”。
- ② 在流程线上,添加 3 个显示图标,分别命名为 A、B、C,如图 3.1 所示。如果想将这 3 个图标归为一组,首先按照上例介绍的方法选择它们,然后单击 Modify Group 命令将其归组。

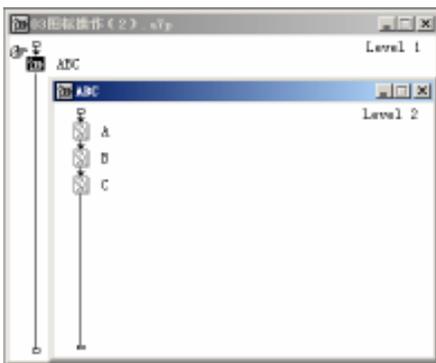


图 3.1 添加图标

③ 图标归组后,变为一个 Group (群组)图标,图标名称是 Untitled,如图 3.2 黑方块所示,不妨将其改名为 ABC。如果要将其中的图标释放出来,可单击 Modify 菜单中的 Ungroup 命令。

④ 每一个“群组”图标通常拥有自己的流程线,双击名称为 ABC 的图标,屏幕又将弹出一个窗口,标题为 Level 2 (第二级窗口),如图 3.3 所示。在第二级窗口中,同样也有流程线,当 Authorware 遇到一个 Group (群组)图标时,它将先执行图标内部的流程线,当执行完最后一个图标时,Authorware 将退出该 Group (群组)图标,执行 Level 1 中



的下一个图标。



图 3.2 图标归组

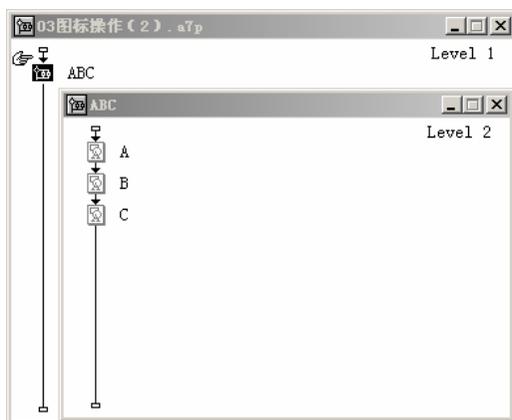


图 3.3 第二级窗口



技巧

在 Group(群组) 图标内可以放置各种图标 , 包括 Group(群组) 图标 , 这样使得 Authorware 的设计空间越来越大。

实例

4

插入图片



实例说明

本例主要讲解在 Authorware 的显示图标中插入图片。

主要知识点：插入图片。



操作步骤

- ① 新建文件,保存为“04 插入图片.a7p”。
- ② 单击 Insert 菜单下的 Image 命令,弹出 Properties : Image 对话框,如图 4.1 所示。
- ③ 单击对话框左下角的 Import 按钮,将会弹出 Import Which file 对话框,如图 4.2 所示。搜索驱动器和文件夹,然后在文件列表框中单击所要的文件,最后单击 Import 按钮将图片插入。

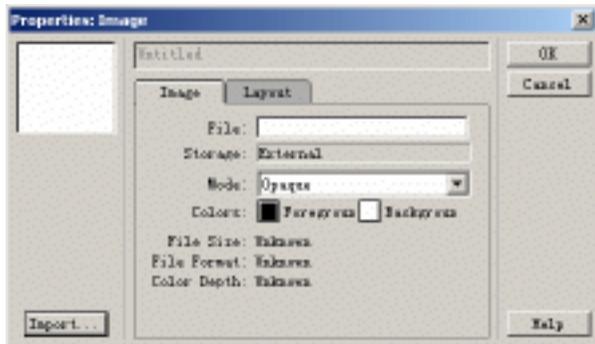
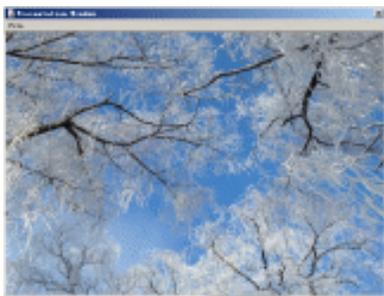


图 4.1 图像属性对话框

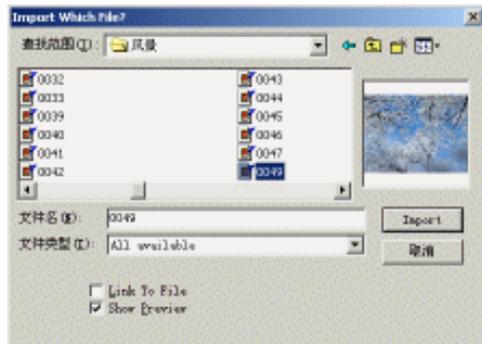


图 4.2 Import Which file 对话框

④ 插入图片后,如果觉得图片的大小或位置不太理想,还可以返回图像属性对话框进行设置。单击 Modify 菜单下的 Image Properties 命令或直接双击窗口中的图片对象都可调出该对话框。

⑤ 单击对话框上的 Layout 选项卡,将显示图片的布局属性,如图 4.3 所示。可以在 Position 的 X、Y 文本框中输入数字来设置图片的位置坐标;在 Size 的 X、Y 文本框中输入数字(像素)来控制图片的大小。

⑥ 保存文件,选择 File (文件) Save (保存) 打开保存文件对话框。

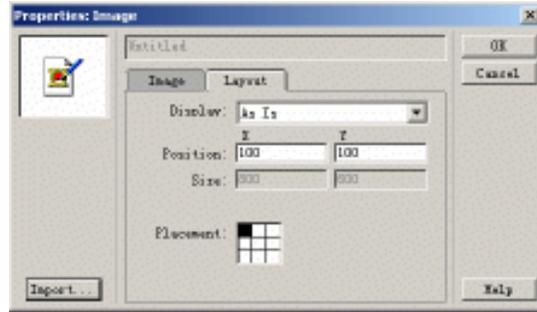


图 4.3 Layout 选项卡



技巧

(1) 如图 4.3 所示，单击 Display 列表框中的 Cropped 选项，在对话框下方出现 Placement 网格，单击其中的方块可以控制图片缩放时截取的位置，同时还可以通过观察窗口中的图片变化来决定如何设置。

(2) 若单击 As Is 选项，原来进行缩放设置的图片将变为原状。



心得体会

这里介绍用另外一种方式来插入图片，这种方式就是复制/粘贴操作。先将图片复制到剪贴板，然后再单击 Edit 菜单下的 Paste 命令，也可以将外部图片插入到 Authorware 内部。

实例

5

插入外部文本



实例说明

本例主要讲解如何在显示图标中插入文本，Authorware 也可以直接将外部的文本输入到 Authorware 的内部，但在文本的输入格式方面有一定的限制，Authorware 只能输入 TXT、RTF 格式的文件。

主要知识点：插入文章、文本格式。



操作步骤

- ① 新建文件，保存为“05 插入外部文本.a7p”。
- ② 单击 File 菜单下的 Import 命令，将弹出 Import file（输入文件）对话框。选择文件后单击 Import 按钮，Authorware 在遇到诸如 word 文件中的分页符时将弹出如图 5.1 所示的对话框。
- ③ 如果单击对话框左面的 Ignore 单选项，同时再选择右面的 Scrolling 单选项，Authorware 将创建如图 5.2 所示的滚动文本。

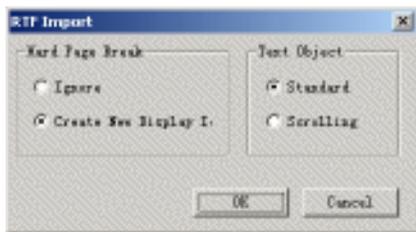


图 5.1 文本输入对话框

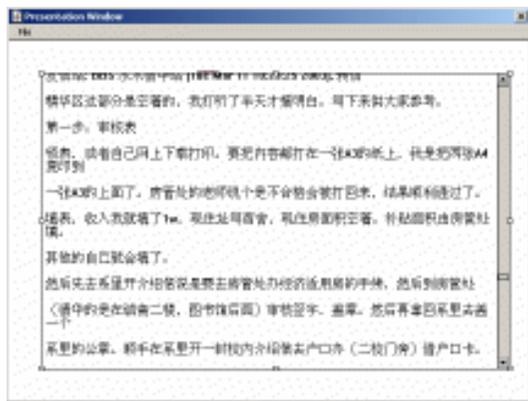


图 5.2 浏览文本

- ④ 如果要设置段落的第一行缩进两个字，可先将光标停留在该段落的前方，然后用鼠标拖动首行缩进标记，如图 5.3 所示，在该标记的下方出现一条标准线，当移动两个字的位置后松开鼠标，段首就自动缩进两个字符。



技巧

- (1) 单击Ignore单选项，Authorware将忽略分页符，把文本输入到一个显示图标中。
- (2) 单击 Create New Display Icon 单选项，Authorware 在遇到分页符时自动创建一个显示图标。
- (3) 单击 Standard 单选项，Authorware 将以标准格式输入文本。
- (4) 单击 Scrolling 单选项，Authorware 将输入的文本以滚动文本的格式显示，当文本过长时，可以拖动滚动条进行文本预览。



图 5.3 首行缩进

⑤ 设置文字的字体前，首先要拖动鼠标选择文字对象，文字背景变黑即被选定，然后打开 Text 菜单，再单击 Font 子菜单下的 Other 命令，就会弹出如图 5.4 所示的 Font 对话框。打开 Font 下拉列表框，在列表框中选择适当的字体，最后单击 OK 按钮确认，正文中选定的文字就以该字体显示了。

⑥ 如果要改变所选字体的大小，同样在 Text 菜单下单击 Size 子菜单中的 Other 命令，就会弹出如图 5.5 所示的 Font Size 对话框。只要在 Font Size 文本框中输入字体大小的磅值，然后单击 OK 按钮即可。如果不知道字体大小是否合适，输入字体大小数值后，在下面的预览框中就会显示该数值对应的字体大小。



图 5.4 设置字体



图 5.5 字型图

⑦ 另外，在 Text 菜单下还有一个 Style 子菜单，如图 5.6 所示，该菜单下的命令可以设置 Plain（平常字体）、Bold（粗体）、Italic（斜体）、Underline（下划线字体等）。在 Authorware 的工具栏中也有这些命令对应的按钮，如图 5.6 所示。

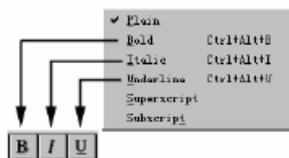


图 5.6 设置字型



图 5.7 对齐字

⑧ 最后介绍文字的对齐方式和滚动文本的转化。在 Text 菜单下打开 Alignment 子菜单，如图 5.7 所示，单击该菜单中的命令可以设置段落 Left（左对齐）、Right（右对齐）、Center（居中）和 Justify（自动调整段落左、右对齐）。

⑨ 在标准文本和滚动文本之间转换时，首先要选择文本对象，即鼠标在选择状态时单击文本，文本周围出现调节方块时即被选定，然后单击 Text 菜单下的 Scrolling 命令，标准文本就会转化为滚动文本，再次单击该命令将返回原状。



实例

6

奥运 2008



实例说明

本例主要讲解片头的制作，如右图所示，当程序开始运行时，出现北京 2008 奥运会的欢迎画面。

主要知识点：擦除图标。



操作步骤

① 新建文件，保存为“06 奥运 2008.a7p”，如图 6.1 所示。



图 6.1 建立一个新程序文件

② 为了让片头不带 Windows 标题栏和菜单栏，在文件属性对话框中关闭 Title Bar 和 Menu Bar 复选框，其他选项保持默认属性。

③ 给片头加一个背景。一般情况下，背景图像的选择要尽量保证程序主要内容的突出地位，不能对其他显示内容造成干扰。内容过于繁杂的背景图像很可能对看清屏幕上的文字造成障碍。

④ 要添加背景，首先在主流程线上拖放一个 Display (显示) 图标，然后将 Windows 资源管理器中选中的图标直接拖放到这个 Display (显示) 设计图标中，再把这个设计图标命名为“背景”。双击打开“背景”设计图标，如图 6.2 所示，可根据图像的位置和大小进行调节。

⑤ 在主流程线上拖放一个新的 Display (显示) 设计图标，命名为“新北京—新奥运”。按下 Shift 键，双击“新北京—新奥运”显示图标可以同时打开“背景”显示图标，创建一个文本对象，输入文字“新北京—新奥运”，此时就可以参照背景色彩为文本对象选择一种

醒目的文本色。这里为了突出绿色奥运，选择绿色为文本色。字体大小设为 60，覆盖模式设为透明模式，并将文本对象移动到窗口正中位置，如图 6.3 所示。



图 6.2 设置背景



图 6.3 设置文本对象属性

⑥ 下面来制作动态消失效果。制作动态消失效果最直接的方法就是为它创建一个 Erase (擦除) 设计图标 ，并在其中指定一种擦除过渡效果。

⑦ 向主流程线上拖放一个 Erase (擦除) 图标，将它命名为“画面滚动”，指定“新北京—新奥运”设计图标为它的作用目标，并在其中设置一个滚动擦除效果。为了使整个窗口内容同时发生滚动，选择 Entire Window 单选项。其他设置如图 6.4 所示。

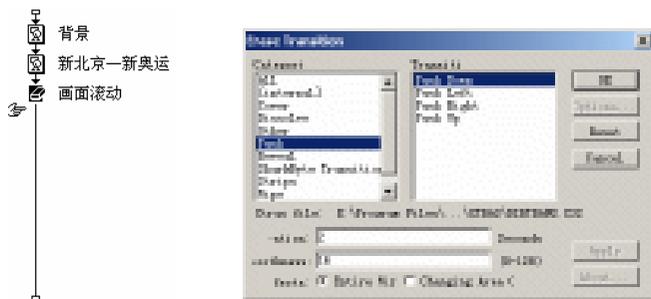


图 6.4 设置滚动擦除效果

⑧ 单击 Do 命令按钮，看一下运行结果，整个窗口发生滚动，最后出现文字部分。下面来添加一点延时，拖动一个 Wait (等待) 设计图标放在“新北京—新奥运”设计图标和“画面滚动”设计图标之间，将等待时间设置为 2 秒，具体设置如图 6.5 所示。再运行



一次，效果明显好多了。



图 6.5 添加延时效果

⑨ 最后再在主流程线上添加一个显示设计图标，并将它命名为“终止”，然后在其中创建一个文本对象，文本内容为 Welcome to Beijing，如图 6.6 所示。



图 6.6 显示另一个标题

⑩ 最后运行程序就可以依次看到如图 6.7 所示的效果。



图 6.7 程序运行结果（由左至右，由上至下）

实例

7

我的小屋



实例说明

本例制作一个小屋，目的是介绍绘图工具箱的使用，通过使用其中的工具图标，绘制出多种图形。实例运行后，演示窗口的最终效果如右图所示。

主要知识点：绘图工具箱。



操作步骤

- ① 新建文件，保存为“07 我的小屋.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择蓝色，如图 7.1 所示。

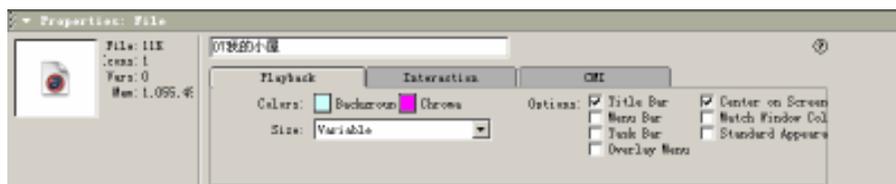


图 7.1 设置文件属性

- ③ 设置程序流程线。从面板中拖放一个 Display(显示)图标到流程线上，命名为“我的小屋”。
- ④ 双击“我的小屋”图标，打开演示窗口。用鼠标拖动演示窗口的边框将其大小调整至适当的尺寸。
- ⑤ 使用绘图工具箱中的多边形工具图标，绘制一个屋顶形状的多边形。方法如下：使用绘图工具箱中的多边形工具图标，在演示窗口中单击鼠标，然后拖动鼠标，拉出一条直线。在合适的位置单击鼠标，将多边形的第一条边画出来，重复以上操作，画出其他边，最后回到起点，双击鼠标，完成多边形的制作。通过调整控制点，使图形更加美观。
- ⑥ 双击绘图工具箱中的多边形工具，打开 Fills 对话框，确认演示窗口中的屋顶图形处于选中状态，双击绘图工具箱上的椭圆形工具图标，打开 Color 对话框，确认演示窗口中的屋顶图形处于选中状态，然后更改 Color 对话框中的图形颜色，更改背景色。



⑦ 使用绘图工具箱中的矩形工具图标绘制一个矩形，充当小屋的烟囱。重复以上操作绘制出一个矩形，充当小屋的主题部分。选中小屋的烟囱，单击绘图工具箱中的填充按钮，单击适合的颜色填充，小屋的其他部分可同样进行填充，如图 7.2 所示。

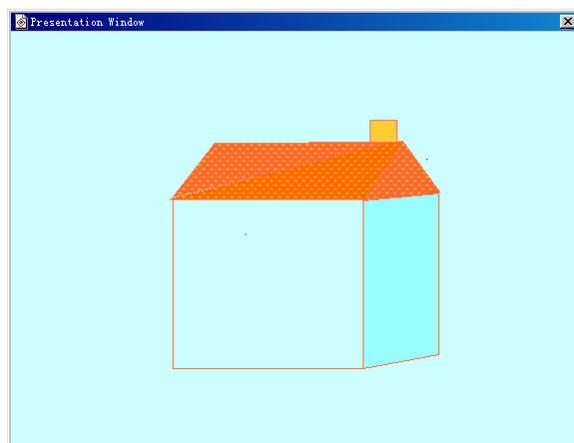


图 7.2 小屋的主题

⑧ 使用绘图工具箱中的多边形工具图标绘制一个心形状的多边形。绘画出烟囱向外冒出“心形”的效果。完成多边形的制作后，通过调整控制点，使得“心形”看上去更加平滑。这一点也可以通过在制作时多设置几个控制点的方法来实现。调节过程中要注意，可以先使用绘图工具箱中的选择工具图标，将“心形”的形状放大，以便更好地调整它的形状。单击工具栏中的 Copy 按钮，再在演示窗口内粘贴几个相同的“心形”，这样就避免重复制作复杂的图形。使用选择工具图标，调整新粘贴上的几个“心形”的大小。最后再将“心形”图形填充上红色即可。

⑨ 使用绘图工具箱中的圆角矩形工具图标绘制一个屋顶的大门。在适当的位置绘画后，调节矩形左上角的控制点，改变圆角的曲率，重复以上操作绘制出门的把手。

⑩ 使用绘图工具箱中的椭圆形工具图标绘制一个椭圆形，充当小屋的窗户。在适当的位置绘画完成后，通过使用绘图工具箱中的选择工具图标，调整椭圆形的大小。

⑪ 在演示窗口使用绘图工具箱中的文本工具图标，输入“我的小屋”，并保存好程序文件。

实例

8

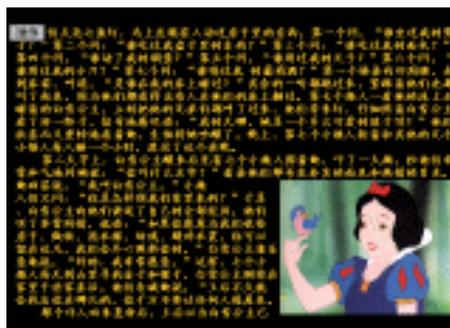
白雪公主



实例说明

本例是一个制作标题为“白雪公主”的文本实例。本例运行后在演示窗口的最终效果是白雪公主故事的显示并配以相关的插图，如右图所示。

主要知识点：显示图标、等待图标。



操作步骤

- ① 新建文件，保存为“08 白雪公主.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择黑色，如图 8.1 所示。

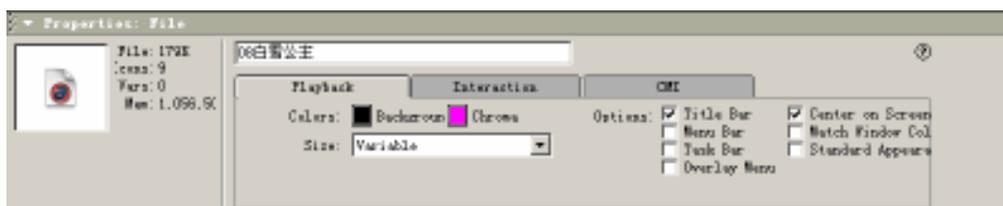


图 8.1 设置文件属性

- ③ 设计程序流程线，从左边的图标面板中拖放一个显示图标到流程线上，命名为“标题”，如图 8.2 所示。

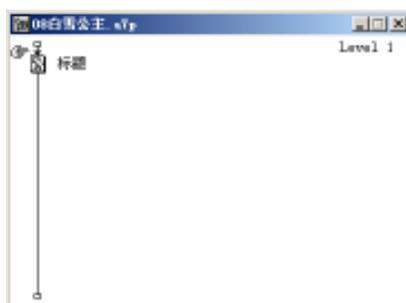


图 8.2 设置程序流程线



④ 双击标题设计图标，打开演示窗口，用鼠标拖动演示窗口的边框将其大小调整到适当的尺寸，然后使用文本工具创建一个文本对象。执行 Text 菜单中 Font 子菜单下的 Other...命令，打开 Font 对话框，在 Font 下拉列表框中选择“黑体”选项，然后执行 Text 菜单中 Size 子菜单下的 Other...命令，打开 Size 对话框，在 Size 文本框中输入“72”。执行 Text 菜单中 Style 子菜单下的 Bold 命令，将文字设置为粗体样式。完成后选用文本工具图标输入文本“白雪公主”。并且重复以上操作重新设置，输入“作者：格林”。最后双击绘图工具箱中的椭圆形工具图标，启动“调色”对话框，设置文本的颜色，最后标题演示窗口的效果如图 8.3 所示。



图 8.3 “标题”演示窗口

⑤ 打开 Properties :Display Icon 对话框，如图 8.4 所示。首先在显示区域中选中 Erase Previous Content 复选框，然后单击 Transition 选项后面的  按钮，打开 Transition 对话框。如图 8.5 所示，选择 Category 列表内的 Internal 选项，以及 Transition 列表内的 Barn Door Open 选项，并且可以根据自己的需要在下面的 Rotation 和 Smoothness 文本框内输入不同的值。这里在 Rotation 文本框内输入“0.5”，Smoothness 文本框内输入“48”，单击 OK 按钮。



图 8.4 设置图标的属性

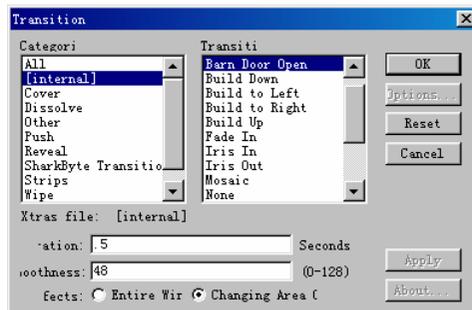


图 8.5 设置过渡方式

⑥ 从左边的 Icons 面板中拖放一个 Wait (等待) 图标到流程线上, 双击 Wait (等待) 图标, 打开 Properties : Wait Icon 对话框。为了使外形美观可以在 Options 区域中不选中 Display Icon 复选框。设置 Wait (等待) 图标对话框如图 8.6 所示。



图 8.6 设置 Wait (等待) 图标

⑦ 双击绘图工具箱中的多边形工具, 打开 Fills 对话框, 确认演示窗口中的屋顶图形处于选中状态, 双击绘图工具箱上的椭圆形工具图标, 打开 Color 对话框, 确认演示窗口中的屋顶图形处于选中状态, 然后更改 Color 对话框中的图形颜色, 更改背景色。

⑧ 新建一个文本对象并且输入故事内容, 并设置文字的字体、大小、风格、颜色以及图标的其他属性。特别是图标属性中过渡方式的选择多种多样, 可以根据自己的喜好作出不同的选择, 制作出不同样式的效果, 如图 8.7 所示。

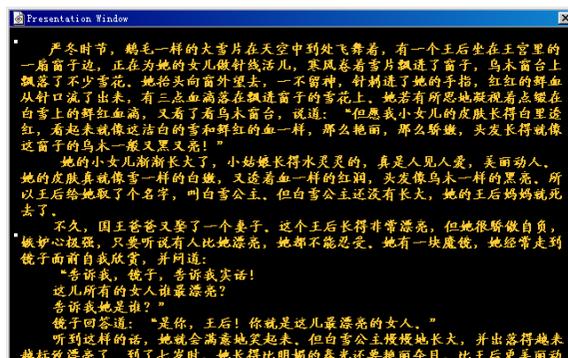


图 8.7 输入内容

⑨ 从左边的 Icons 面板中拖放一个 Wait (等待) 图标到流程线上, 双击流程线上的 Wait (等待) 图标, 打开 Properties : Wait Icon 对话框, 区别于步骤 5, 在时间设置上要求长一些 时间的选取要控制在用户读完这页文字的时限。选取 Options 区域中的 Show Button 复选框, 这是为了便于用户控制翻页, 如图 8.8 所示。



图 8.8 设置 Wait (等待) 图标



⑩ 设置程序流程线，从 Icon 面板中拖放一个 Display (显示) 到流程线上，命名为“文章 2”。

⑪ 为了使文章更加生动，可以向其中加入一些相关的图片。首先双击显示图标“文章 2”打开演示窗口，执行 Insert 菜单中的 Image 命令，打开 Properties: Image 对话框，选择自己所要使用的图片，单击 OK 按钮，如图 8.9 所示。在 Properties: Image 对话框中的 Layout 选项卡中的 Display 下拉列表框内选中 Scaled，单击 OK 按钮，如图 8.10 所示。

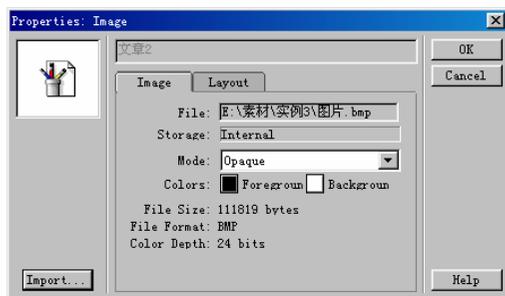


图 8.9 设置图像属性 1

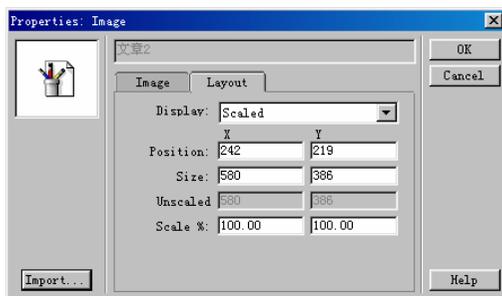


图 8.10 设置图像属性 2

⑫ 新建一个文本对象并且输入文章 1 后的故事内容，并根据需求设置文字的字体、大小、风格、颜色以及图标的属性。在输入文章覆盖到图片时，可以采用换行的方式，避免文章与图片的重叠，如图 8.11 所示。

⑬ 重复步骤 7，设置 Wait (等待) 按钮。

⑭ 依照以上的步骤，将故事内容输入完整，并配以生动的插图，编辑后的流程线如图 8.12 所示。



图 8.11 文章 2 的最终效果图



图 8.12 流程线

实例

9

精彩射门(1)



实例说明

在一些多媒体教学软件的运行过程中,经常会弹出按钮提示是否继续往下进行,在这个问题上 Authorware 是怎样处理的呢?图标栏中的 Wait(等待)图标可以解决这个问题,本例使用它在程序运行中显示为 Wait(等待)按钮,通过单击它、设置按键或等待一段时间来控制程序的向下执行。

主要知识点:等待图标。



操作步骤

- ① 新建文件,保存为“09 精彩射门.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,选择 Center On Screen 复选框,取消选择 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择绿色,如图 9.1 所示。

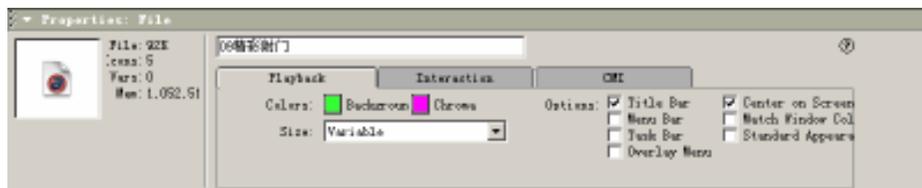


图 9.1 设置文件属性

- ③ 设置程序流程线。从面板中拖放 2 个 Display(显示)图标到流程线上,分别命名为“球场”和“足球”,然后再添加一个动画图标,命名为“射门”。
- ④ 最后在流程线的下面放置一个等待图标和一个显示图标分别命名为“等待”和“精彩回放”。最终的流程线,如图 9.2 所示,双击等待图标,弹出 Properties: Wait Icon(等待图标属性)对话框。
- ⑤ 设置完等待图标后,单击“运行”按钮,如图 9.3 所示,演示窗口中出现等待按钮和一个时钟,时钟在不断的显示剩余时间。可以拖动按钮和时钟来改变它们的位置,也



可以移动按钮旁的调节方块来改变它的大小。

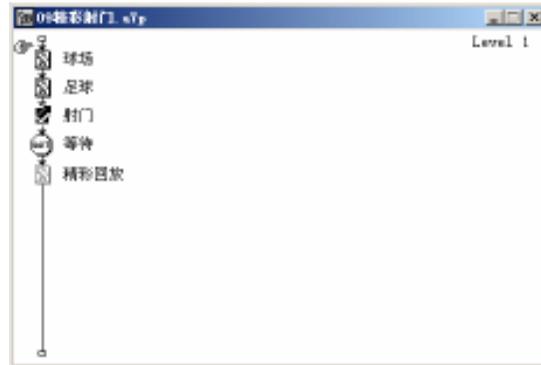


图 9.2 流程线



图 9.3 设置按钮位置

实例

10

精彩射门(2)



实例说明

本例接着上一节来完成制作精彩射门,本例主要来设置擦除画面和设置过渡效果。

主要知识点:等待图标、动画图标。



操作步骤

① 接上一例,流程图开始如图 10.1 所示,在流程线的最上方放置一个 Display (显示) 图标和一个 Erase (擦除) 图标,分别将它们命名为“开始”和“擦除”,然后在 Display (显示) 图标中插入效果图中的画面。



图 10.1 流程线

② 在制作序幕画面拉开的过程时,首选讲解 Display (显示) 图标过渡效果的设置过程。选择流程线上的“开始”图标,然后打开 Modify 菜单,在 Icon 子菜单中单击 Transition (过渡) 命令,此时,Authorware 将弹出如图 10.2 所示的 Transition 对话框。

③ 在对话框中的 Categories (种类) 列表框中选择过渡模式的种类,如 Reveal (显示)。然后在 Transition (过渡模式) 列表框中选择合适的过渡效果,如 Reveal Right (向右显示)。如果要观察此模式的效果,可单击 Apply (应用) 按钮进行擦除效果预览。

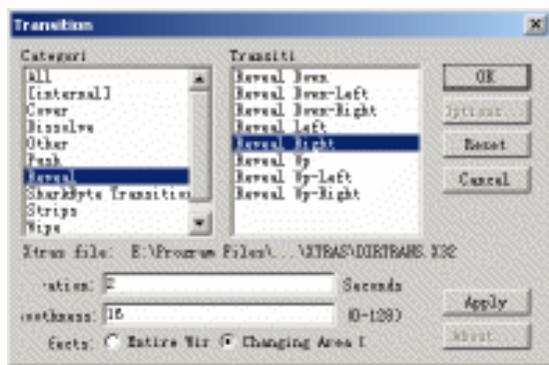


图 10.2 擦除方式



技巧

(1) 在 Duration (持续时间) 文本框中可输入擦除效果持续的时间, 单位是秒。

(2) 在 Smoothness (光滑度) 文本框中可输入 0~128 之间的整数, 这些整数代表过渡效果的光滑度, 值越大, 过渡过程越粗糙, 0 表示最光滑的过渡过程。

(3) Affects (影响范围) 中设有两个选项范围, 若单击 Entire Area, 过渡效果会影响全屏, 单击 Changing Area 只会影响擦除对象区域。

④ 设置完毕, 单击 OK 按钮返回主流程线。双击“擦除”图标或单击“运行”按钮, Authorware 都会将 Properties : Erase Icon (擦除图标属性) 对话框打开。如图 10.3 所示, 根据 Icon 选项卡中的标题 Click Object to 提示, 单击对话框后面的画面, 画面将被擦除, 同时, 在下面的列表框中将显示擦掉的图标名称。



图 10.3 设置擦除对象



技巧

选择 Icon to Erase (擦除图标) 单选按钮, 列表框中的图标将被擦除; 选择 Icon to Preserve (保留图标) 单选按钮, 列表框中的图标在擦除时将被保留。如果要取消擦除的图标, 可以先在列表框中选择该图标, 然后单击 Remove 按钮。

⑤ 在如图 10.3 所示的对话框中同样有 Transition 设置, 单击按钮, 将会打开擦除过



渡模式对话框，该对话框与前面讲的过渡模式对话框的使用基本相同，此处不再重复。若单击 Prevent Cross Fade（防止交叉干涉）复选框，Authorware 将会在显示下一个对象之前擦除当前对象。

⑥ 在擦除过渡模式对话框中可以设置擦除过渡模式为 Reveal Left（向右显示），当序幕拉开后，程序执行到“擦除”图标时，画面就会慢慢向右擦掉。



实例

11

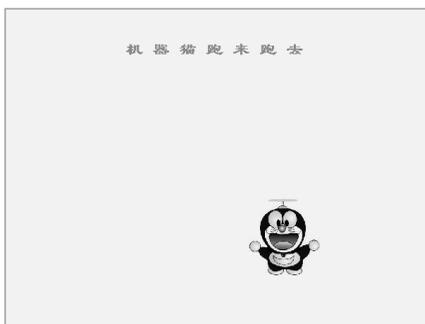
可爱的机器猫



实例说明

本例是一个制作机器猫的动画效果的实例,运行后可以通过设定循环次数看到机器猫来回运动的次数,如右图所示。

主要知识点:动画图标。



操作步骤

① 新建文件,保存为“11 机器猫.a7p”。

② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Center On Screen 复选框和 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择白色,如图 11.1 所示。



图 11.1 设置文件属性

③ 设计程序流程线。从面板中拖放一个 Calculation (计算) 图标到流程线上,并将图标命名为“初始化窗口”。然后再拖放一个 Display (显示) 图标到流程线上,命名为“背景”。接着拖放一个 Decision (显示) 图标到流程线上,命名为“循环”,然后放一个 Group (群组) 图标到 Decision (判断) 图标的右下方,命名为“跑来跑去”,程序流程线如图 11.2 所示。

④ 设置“初始化窗口”图标内容。双击流程线上的 Calculation (计算) 图标,打开代码编辑器,然后输入 Resizewindow(600,450),如图 11.3 所示。其目的是在程序执行时初始化展示窗口的大小。

⑤ 设置“背景”图标的内容。双击流程线上的“背景”显示图标,打开演示窗口,用文本工具在图中输入“机器猫跑来跑去”,将字体设为隶书,颜色为蓝色,调整其在窗口的位置。

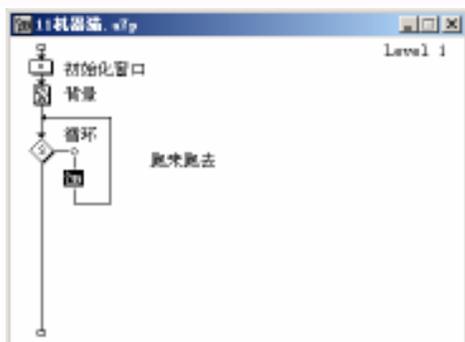


图 11.2 流程图

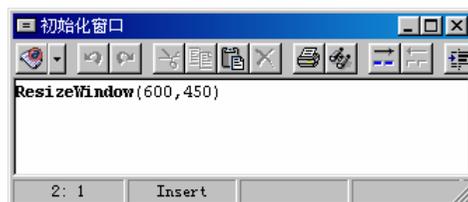


图 11.3 代码窗口

⑥ 设置 Decision 图标。双击流程线上的“循环”图标，弹出 Properties 对话框，在 Repeat 下拉列表框中选择 Fixed Number of Times，在下面的文本框中输入“8”，在 Branch 下拉列表框中选择 Sequentially 项，对话框设置如图 11.4 所示。



图 11.4 设置“循环”图标属性

⑦ 设置“跑来跑去”图标的内容。双击流程线上的 Group (群组) 图标，打开二级设计流程图窗口，设计流程图如图 11.5 所示。

⑧ 设置“机器猫”图标的内容。双击 Display 图标打开演示窗口，执行 File 菜单中的 Import 命令，选择机器猫的图片，并用鼠标拖动图片四周的控制点调整图片的大小以及在演示窗口中的位置。



图 11.5 设置 Group (群组) 图标

⑨ 设置“左运动”图标的属性。双击“机器猫”图标，然后双击“左运动”图标，单击“机器猫”为移动对象，单击 Direct to Point，将“机器猫”向左方移动一段距离。在 Timing 下拉列表中选择 Time(sec)，在下面的文本框中输入“2”，在 Concurrency 下拉列表中选择 Wait Until Done，如图 11.6 所示。

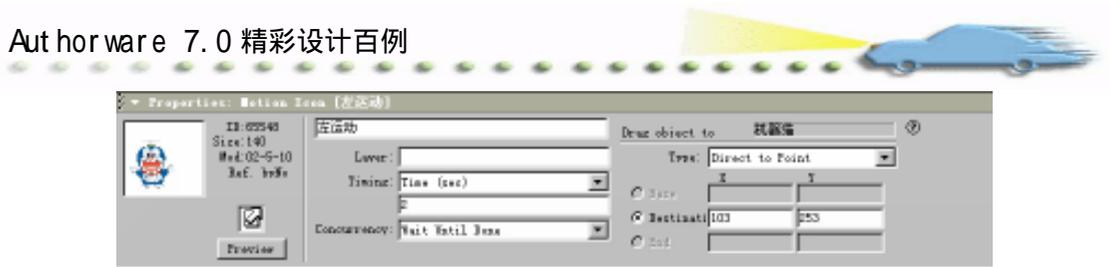


图 11.6 设置“左运动”图标属性

- ⑩ 重复以上的操作，设置“上运动”、“下运动”和“右运动”。
- ⑪ 预览效果。单击工具栏中的 Restart 按钮，便可以运行此实例。

实例

12

Flash 动画



实例说明

本例是一个制作“Flash 动画”效果的实例，运行后出现提示文字和开始按钮，单击“开始”按钮后演示窗口将播放 Flash 动画，如右图所示。

主要知识点：插入动画。



操作步骤

- ① 新建文件，保存为“12Flash 动画.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择蓝色，如图 12.1 所示。



图 12.1 设置文件属性

- ③ 设计程序流程图。从面板中拖放一个 Calculation（计算）图标到流程线上，并将其命名为“初始化窗口”，然后再拖放一个 Display（显示）图标到流程线上命名为“文字说明”。然后再拖放一个 Interaction（交互）图标到流程线上，命名为“响应”，最后拖放一个组图标到交互图标 \diamond 的右下方，命名为“开始”，如图 12.2 所示。

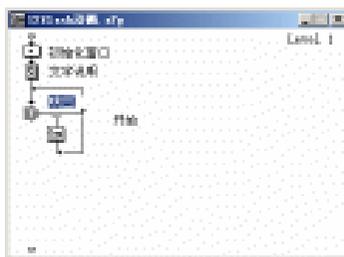


图 12.2 流程线

- ④ 设置“初始化窗口”图标内容。双击 Calculation（计算）图标，在打开的代码窗口中输入 Resizewindow(480,360)。



⑤ 设置“文字说明”图标内容。双击流程线上的“文字说明”图标，打开演示窗口，用文本工具在图中输入“这是一个在 Authorware 中插入 Flash 动画的实例”，将字体设置为隶书，颜色为红色，调整其在窗口中的位置，如图 12.3 所示。

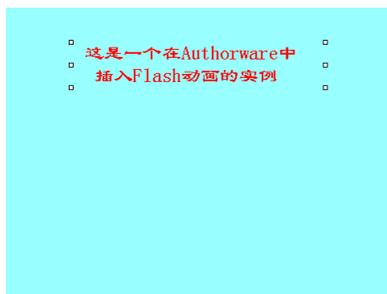


图 12.3 设置文字说明图标

⑥ 设置“开始”按钮。双击控制按钮打开属性对话框。单击对话框上方的 Button 按钮，打开按钮编辑器设置按钮字体大小为“10”磅，在 Type 下拉列表中选择 Button。属性设置如图 12.4 所示。

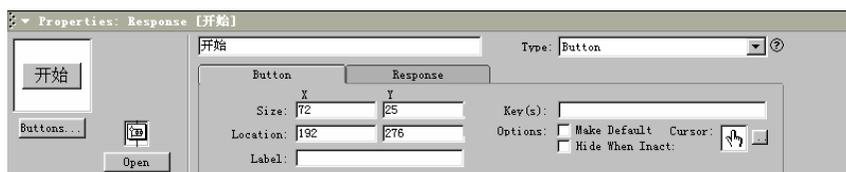


图 12.4 设置按钮属性

⑦ 设置“开始”图标响应属性。属性设置如图 12.5 所示。

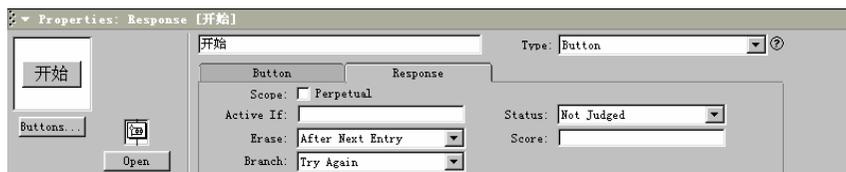


图 12.5 设置按钮响应属性

⑧ 设置“开始”按钮。双击流程线上的“开始”组图标，打开二级程序设计流程线，执行 Insert 菜单下 Media 子菜单中的 Flash 命令，打开对话框，单击 Browse 按钮，选择一个 Flash 动画文件，如图 12.6 所示。

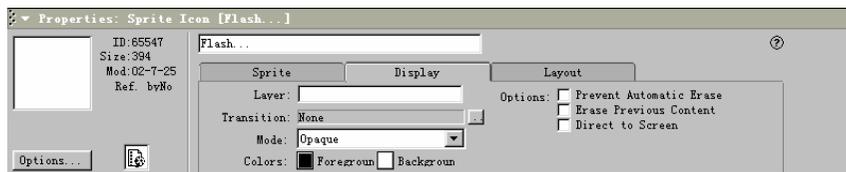


图 12.6 设置 Flash 动画属性

⑨ 预览效果。单击工具栏中的 Restart 按钮运行实例，并保存程序文件。

实例

13

按钮交互 (1)



实例说明

本例利用按钮交互来实现退出功能，设置的流程线如右图所示。

主要知识点：交互图标、计算图标。



操作步骤

① 新建文件，保存为“13 按钮交互 (1).a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框 取消选择 Menu Bar 复选框 单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 13.1 所示。

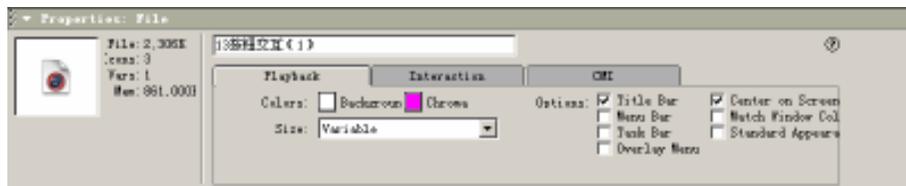
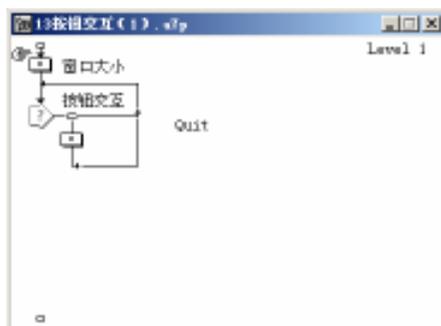


图 13.1 设置文件属性

③ 从图标栏上拖动一个计算图标到流程线上，更名为“窗口大小”。双击该计算图标 ，在打开的脚本编辑窗口中键入：`ResizeWindow (320,240)`。从而将演示窗口的大小设定为 320 × 240 大小，如图 13.2 所示。

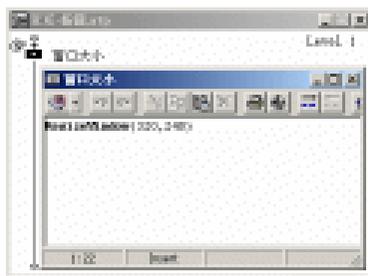


图 13.2 设置交互

④ 从图标栏上拖动一个交互图标放在流程线的下面，更改该交互图标名称为“按钮



交互”。从图标栏上拖动一个计算图标，将它拖动到交互图标的右侧，然后释放鼠标。

⑤ 当释放鼠标后，系统会弹出一个 Response Type 对话框。如图 13.3 所示，在该对话框中，确认选择了 Button 选项，然后单击 OK 命令按钮关闭该对话框。



图 13.3 Response Type 对话框

⑥ 将新添加的按钮交互分支更名为“退出”。双击该分支下的计算图标，打开“退出”脚本编辑窗口，在该窗口中键入：Quit(1)，从而完成整个程序流程的设计，如图 13.4 所示。

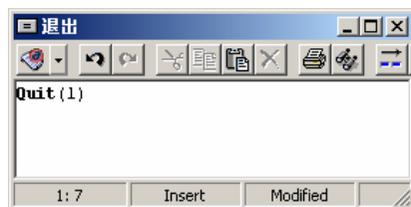


图 13.4 “退出”交互设计

⑦ 当用户完成了上面的操作步骤，同时按下 Ctrl + R 键运行项目文件，在展示窗口中可以看到出现了一个名为“退出”的按钮，单击该按钮，程序退出，返回到流程线的编辑交互窗口，从而证明该路按钮交互执行正确。



说明

对于交互，用户应该理解交互图标，该图标管理着下挂的所有交互分支。除了交互图标，交互的组成还包括交互标签，交互分支，以及交互后的流程走向。当用户向流程线上添加第一路交互分支时，系统都会弹出 Response Type 对话框。函数 Quit 用来实现程序的退出，参数“1”代表返回到桌面。如果当前处于编辑环境，参数“1”将使得程序运行退出，返回到编辑状态。

实例

14

按钮交互 (2)



实例说明

Buttons 对话框虽然也提供了一些不同样式的按钮,但样子都比较呆板,本例来创建一个有特色的“退出”按钮。程序流程图如右图所示。

主要知识点:设置按钮图形。



操作步骤



① 确认当前打开的是“14 按钮交互 (2) .a7p”项目文件的流程线窗口(这里的文件仅是将实例 13 中的文件重命名)。双击“退出”交互分支的交互标签,打开 Properties : Response 对话框。

② 单击 Properties :Response 对话框左上角的 Buttons...命令按钮,系统将打开 buttons 对话框。在打开的 Buttons 对话框中,单击左下角的 Add...命令按钮,系统将打开 Button Editor 对话框,如图 14.1 所示。

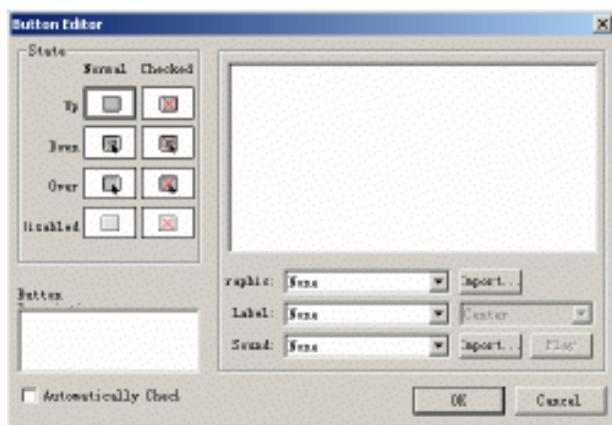


图 14.1 Button Editor 对话框

③ 单击 State 栏的 Normal 列的 Up 行所对应的按钮方框,确认它处于选中状态,然后单击 Graphic 栏后面的 Import...命令按钮,系统将打开 Import Which File? 对话框。

④ 在打开的 Import Which File? 对话框中,选择预先准备好的图形文件,然后选中该对话框下面的 Show Preview 检查框,从而可以预览选中的图形文件,如图 14.2 所示。

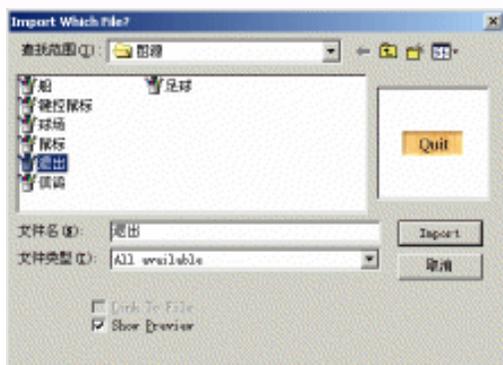


图 14.2 导入文件

⑤ 单击对话框中的 Import 命令按钮完成导入工作。当返回到如图 14.1 所示的 Button Editor 对话框之后，单击 State 栏的 Normal 列，Down 行所对应的方框，然后单击 Graphic 栏后面的 Import 命令按钮，导入按钮图形。

⑥ 返回到 Button Editor 对话框之后，在 Button Description 对话框中键入该按钮的描述文字：用户自定义的“退出”按钮。

⑦ 通过上述操作步骤，用户应该掌握如何生成自定义的命令按钮，从而丰富画面的显示。当创建了自定义的按钮之后，在 Buttons 对话框中为交互按钮分支指定该按钮即可完成整个工作。



说明

(1) 所谓创建用户自定义按钮，实际上是根据按钮的工作状态，为不同的状态设置不同的显示画面，例如按钮在鼠标未移动到它上面时的状态，鼠标移动到它上面的状态，以及鼠标在按钮上单击后的显示状态。对应不同的按钮状态，在 Authorware 中实际上是用不同的图形文件来表现的。在 Button Editor 对话框中，用户可以定义两种类型的按钮，一种是 Normal 普通按钮，另外一种是被选中类型的按钮。

(2) 对于用户自定义按钮，使用的图形文件只能采取复制的方式加载到项目文件中，在导入文件对话框中采用链接方式的检查框是无效选取的。对于通常的应用，只需要为按钮的 Up 状态 and Down 状态设置不同的图形显示，就可以达到很好的效果，但是，还可以为 Over 状态使用不同的图形显示状态，从而使得生成的按钮更加吸引人。对按钮的说明文字将在 Buttons 对话框中出现，它有助于用户管理按钮。

实例

15

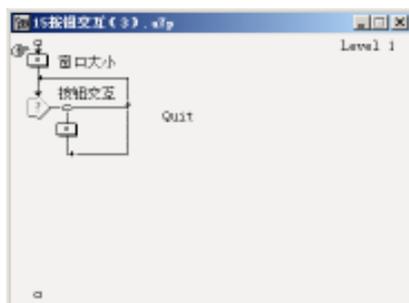
按钮交互 (3)



实例说明

随着多媒体技术的发展,在当前的软件开发中,开发者总是尽量做到不仅要满足视觉效果,而且要满足听觉效果。本例为前面设计的按钮交互添加音效,程序流程图如右图所示。

主要知识点: 为按钮添加音效。



操作步骤

① 确认当前打开的是“15 按钮交互 (3) .a7p”项目文件(这里的文件仅是将实例 14 中的文件重命名)。选择菜单 Window,在打开的下拉菜单选项中选择 Buttons...,系统将打开 Buttons 对话框。在打开的对话框中,用户可以看到自定义的 Quit 按钮,如图 15.1 所示。

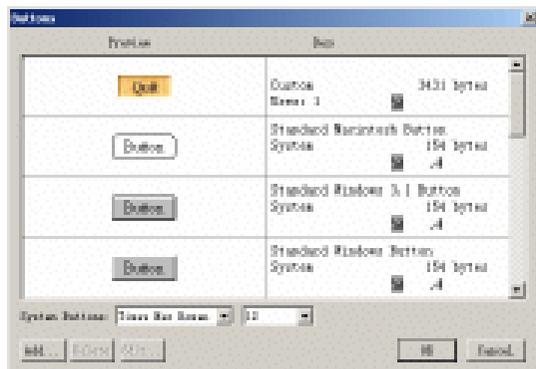


图 15.1 自定义按钮

② 在 Buttons 对话框中,单击自定义的 Quit 按钮,选中该按钮,然后单击对话框左下角的 Edit...命令按钮,系统将打开 Button Editor 对话框。

③ 在 State 栏中单击 Normal 列的 Down 行所对应的方框,然后单击 Sound 栏右侧的 Import...命令按钮,打开 Import Which File? 对话框,如图 15.2 所示。

④ 在 Import Which File? 对话框中,选择用于在按下按钮时发出音效的声音文件,然后单击该对话框中的 Import 命令按钮完成声音的导入过程。

⑤ 关闭 Import Which File? 对话框,返回到 Button Editor 对话框,用户可以看到 Sound



栏显示为 Use Imported ,如图 15.3 所示 ,单击该栏后面的 Play 命令按钮可以试听声音效果。

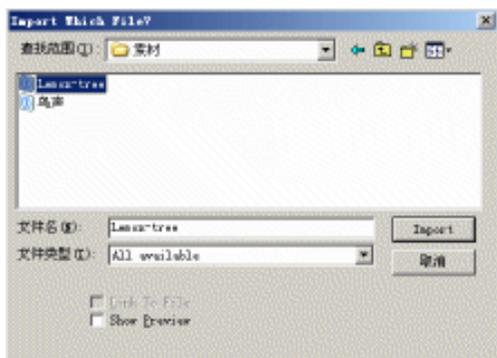


图 15.2 导入声音

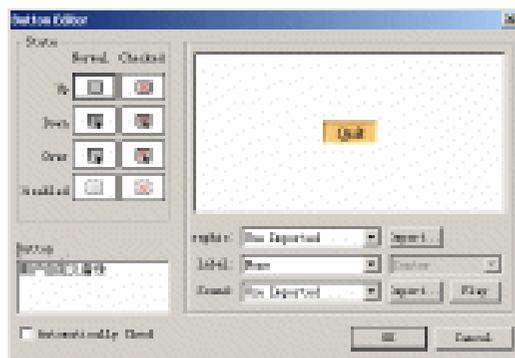


图 15.3 试听音乐效果

⑥ 关闭 Button Editor 对话框，返回到流程线窗口，同时按下 Ctrl + R 键运行项目文件，在演示窗口中可以看到 Quit 按钮，单击该按钮，退出程序的运行。

⑦ 通过上面的操作过程，用户应该掌握如何创建带有声音效果的按钮。在使用该项功能的时候，用户应该选择好所需的声文件。在后面介绍了声音图标的使用方法后，用户还可以学会如何使用声音图标来添加音效。



说明

(1) 在多媒体项目文件中，按钮的声音效果可以使得作品表现更加丰富。系统提供的按钮是不具备声音效果的，如果需要创建带有声音效果的按钮，必须使用自定义的按钮。按钮的声音效果需要在 Button Editor 对话框中完成。为按钮添加声音效果，用户可以使用最常见的 WAV 声音文件，还可以使用 AIF, PCM, SWA, VOX 声音文件。

(2) 如果用户不能确定导入何种声音文件，可以单击导入文件对话框中的“文件类型”右侧的向下箭头，就可以查看。在按钮编辑对话框中，用户可以充分利用 Play 命令按钮来试听声音效果。在为按钮添加声音效果时，需要注意的是该声音不能够太长，否则影响作品的表现效果。当程序运行时，在执行按钮的声音效果时，会停止其他的流程，当声音效果播放完之后，才继续下面的流程。

实例

16

数学题



实例说明

本例用按钮交互来制作一个实例“做数学题”。按钮的形状有多种多样，在平时遇到的考卷中，选择题的选项前就有单选按钮，用鼠标单击单选按钮就会选择该选项。在下面列举了一个做数学选择题的例子。

主要知识点：交互图标。



操作步骤

- ① 新建文件，保存为“16 数学题.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 16.1 所示。



图 16.1 设置文件属性

- ③ 先在设计窗口中创建如图 16.2 所示的流程线。在将 Display (显示) 图标放置在 Interaction (交互) 图标右侧时，Authorware 会弹出交互类型对话框，选择 Button 单选项后确认。
- ④ 再从图标栏中拖动 3 个 Display (显示) 图标，它们的交互类型将会默认为按钮交互。如图 16.2 所示更改标题名称为 A、B、C、D。在流程线最上方的 Display (显示) 图标中将存放试卷画面，在 Interaction (交互) 图标中将输入试题和选项。



技巧

可以不使用 Display (显示) 图标，将全部的内容都输入 Interaction (交互) 图标也可以达到同样的效果。

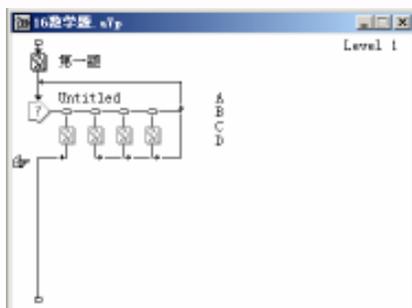


图 16.2 流程线

⑤ 打开“第一题”图标，在演示窗口中输入如图 16.3 所示的文字，然后再制作试卷背景，但要注意设置文字及背景的关系。打开 Properties :Display Icon(显示图标属性)对话框，设置画面的过渡模式为 Vertical Blind。这样，当程序一进入显示图标时就会显示出画面的过渡效果，如此设置会丰富画面的显示效果。



图 16.3 设置过渡效果

⑥ 双击流程线上的交互类型按钮，打开 Properties : Response (交互属性)对话框，如图 16.4 所示，打开 Branch 下拉列表框，在其中设置反馈分支为 Exit Interaction。其他选项取默认值。

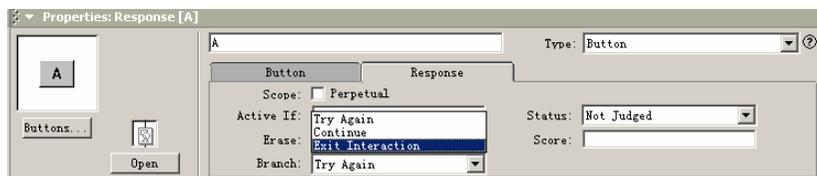


图 16.4 设置反馈分支



技巧

在 System Buttons 列表框中，可以选择按钮标题的文字字体。同时，也可以在旁边的列表框中选择标题文字的字号。

⑦ 如果觉得按钮设置不太理想，Authorware 也允许对其进行修改。选择按钮样式，然后单击对话框中的 Edit (编辑)按钮，接下来会弹出图 16.5 所示的 Button Editor (按钮

编辑)对话框。在 State (状态) 列表框中有两列按钮,单击它们可以用来预览按钮状态。在 Checked 列按钮上单击,在右侧的预览框中将显示按钮按下的状态。单击 Normal 列中的按钮显示正常状态的按钮形状。

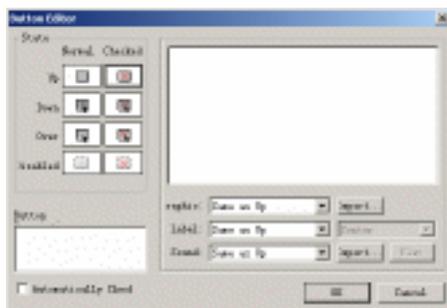


图 16.5 按钮编辑对话框



技巧

打开 Label (标签) 列表框,其中的选项主要用于设置按钮标签的显示状态,如选择 Show Label,在按钮的右侧会显示标签名称。同时,在其右侧的列表框中设有有关的选项,它们主要是用于设置标签在按钮上的显示位置,如 Align Left,设置标签显示在按钮左侧。如果单击 Sound Import (输入声音) 按钮,还会弹出一个对话框,要求加载该按钮的伴随声音文件。

⑧ 这样,第一个按钮设置完毕。用同样的方法可以设置另外 3 个 Display (显示) 图标,设置完毕,来看看按钮在演示窗口中的具体位置。打开 Interaction (交互) 图标,用鼠标将按钮一一拖至合适的位置。

⑨ 如果觉得按钮没有对齐,如图 16.6 所示,按住 Shift 键将按钮全部选中,然后调出 Align (排列) 选择框,单击合适的对齐方式。



图 16.6 鼠标列表框



技巧

也可以在程序运行过程中按 Ctrl+P 快捷键终止程序,然后再调节它们的位置。



实例

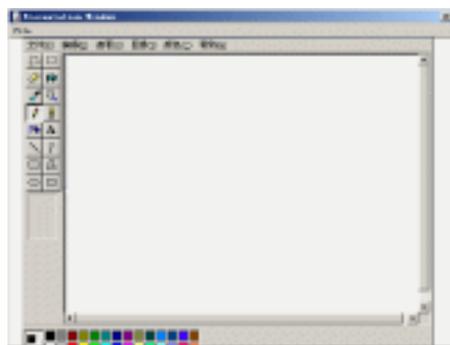
17

提示向导



实例说明

“画图”对我们来说是最熟悉不过了，打开 Windows 程序组中的附件，就能调用它。在“画图”的窗体中有一个工具栏，如果用鼠标移至某一工具按钮的上方，在鼠标下面就会出现它的名称。下面就这个名叫“提示向导”的例子介绍如何使用热区交互来实现这种功能，同时将它的功能加大，把有关它的帮助信息也显示出来。



主要知识点：热区交互。



操作步骤

- ① 新建文件，保存为“17 提示向导.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 17.1 所示。

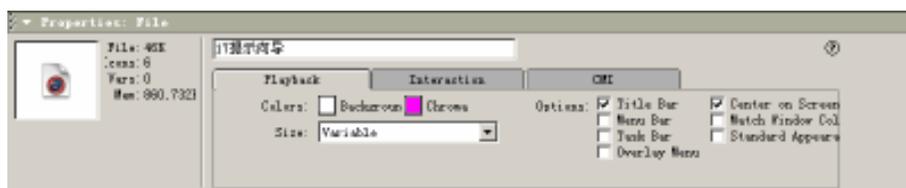


图 17.1 设置文件属性

③ 如图 17.2 所示，在设计窗口中创建流程线，当弹出 Response Type（交互类型）对话框时，选择 Hot Spot（热区交互）单选按钮，同样放置其余的 Display（显示）图标，然后将其一一更改为图中所示的标题名称。在 Display（显示）图标中将存储对应工具按钮的名称和基本功能。

④ 单击流程线上的交互类型按钮，弹出如图 17.3 所示的对话框。单击 Match 下拉列表框，其中的选项可设置与热区交互匹配的鼠标动作。

⑤ 单击 Highlight on Match 复选框，热区会产生如下变化：程序运行时，用鼠标单击或双击热区时，热区会以高亮显示，松开鼠标后热区状态复原。而单击 Mark on Match 复选框，在演示窗口中的热区内会显示一个匹配标志，一般是一个白色小方块，当产生交互

时，该匹配标志就被黑色填充，如果交互结束，黑色并不消失。

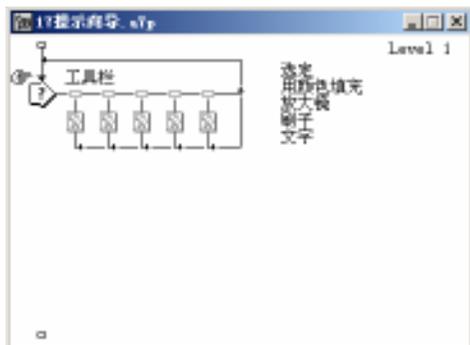


图 17.2 流程线

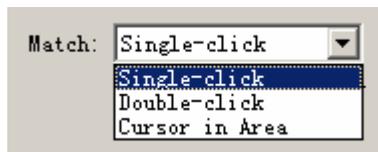


图 17.3 Match 选项

⑥ 如图 17.4 所示，用户可以在 Key(s) 文本框中输入与此热区交互对应的快捷键，如可以输入 1、2、3、4 等数字，也可以输入 a、b、c 等字母。如果需要使用组合键来代表快捷键，例如使用 Ctrl 和 Alt 组合键，可以在它们后面直接加上该字母，图 17.4 所示的 Ctrl+5 代表 Ctrl+5 快捷键。如果一个热区需要使用多个快捷键，必须在字符中间插入运算符“|”，例如“B|b”，表示按下 B 或 b 都能激活该热区交互。

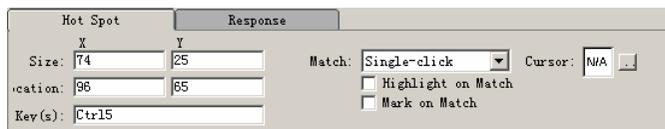


图 17.4 设置快捷键



说明

在键盘上的许多功能键可以直接使用，例如 Tab、Enter、Backspace、Esc、Home、End、方向键等，可以直接在文本框中输入该键的名字。

⑦ 在 Interaction (交互) 图标的右侧一共有 5 个热区交互，5 条分支路径。因此在演示窗口中必有 5 个热区方块。打开 Interaction (交互) 图标，然后单击工具栏上的“插入”按钮，在图标中插入“画图”窗体的界面。然后用鼠标将热区移至对应的工具按钮上方，如图 17.5 所示。如果热区大小与按钮大小不匹配，可以通过拖动热区的调节方块进行调整。热区的边框在程序运行时是不可见的，如果想将其变为可视，可以用鼠标在其周围画一个矩形的方框。



技巧

如果觉得用鼠标操作不太方便，也可以通过键盘来完成。先选定热区，然后按键盘上的上下左右方向键来移动，每按一下方向键对象移动一个像素点。

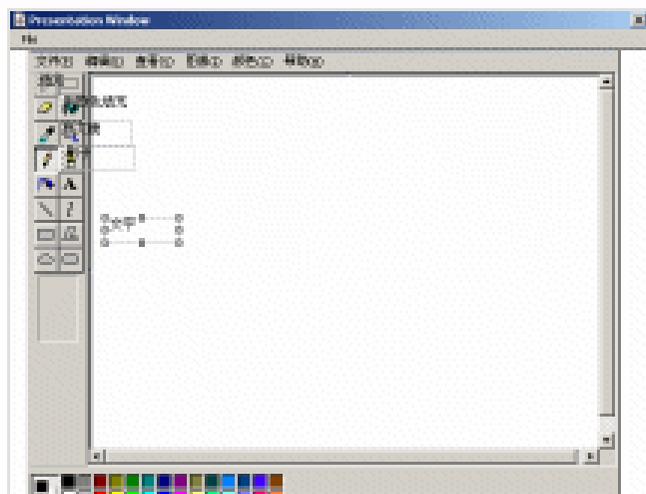


图 17.5 移动热区

⑧ 程序编辑完成后，移动鼠标到“刷子”按钮的上方时，在屏幕的右侧会显示按钮的名称及相应的帮助信息，然后再将鼠标移到其他按钮的上方，原来的信息消失而显示新的信息。因此，可以在分支的每个 Display（显示）图标中输入对应按钮的名称及相应的帮助信息。同时，在交互属性对话框的 Response 选项卡中设置分支为 Try Again，当然，设为 Continue 也可以。在这个程序中，只做了工具按钮功能的一部分，如果用户有兴趣的话，可以仿照此例继续做完其他按钮的功能。

实例

18 地球 (1)



实例说明

所谓的热区交互，就是允许用户将演示窗口中的某个区域定义为热区，在定义的热区区域中单击或双击，Authorware 就认为发生了交互，从而执行该路交互下面的流程。本例使用热区交互来制作一个地图导游，当在地球图形上单击时，就会显示相应的文本说明。

主要知识点：热区交互。



操作步骤

- ① 新建文件，保存为“18 地球.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 18.1 所示。

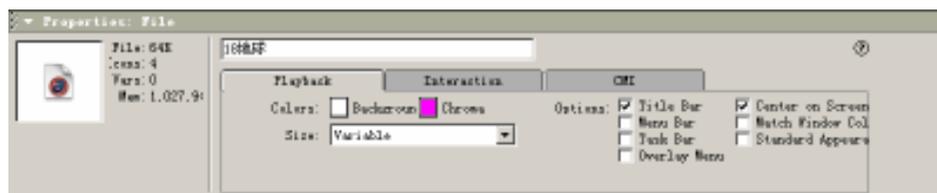


图 18.1 设置文件属性

- ③ 设置如图 18.2 所示的程序流程图。确认当前的粘贴小手位于流程图的最上方，选择菜单 File，在打开的下拉菜单中选择 Import...选项，打开 Import Which File? 对话框。

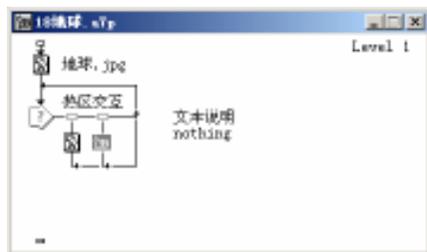


图 18.2 “地球”主流程线



④ 单击 Import Which File? 对话框中的 Import 命令按钮，系统自动在流程线上添加一个显示图标，该显示图标的名称以导入的图形文件名称命名，图中为“地球.jpg”。双击该显示图标，在打开的展示窗口中将该图形缩小为原来的 50%。

⑤ 从图标栏上拖动一个交互图标，放在“地球.jpg”显示图标的下面，把该交互图标的名称改为“热区交互”。从图标栏上拖动一个显示图标，在添加的交互图标右侧释放鼠标，系统自动打开 Response Type 对话框。

⑥ 在打开的 Response Type 对话框中选择 Hot Spot 选项，从而创建热区交互，将该路热区交互更名为“文本说明”。从图标栏上继续拖动一个组图标 ，在上一路热区交互右侧释放鼠标，从而添加第二路交互。

⑦ 在流程线上双击“文本说明”显示图标，在打开的展示窗口中键入所需要的说明文字（用户可以随意键入或者使用其他图形）。

⑧ 按下 Ctrl + R 键运行项目文件，在打开的演示窗口中，用户可以看到地球的图形对象。再按下 Ctrl + P 键暂停项目文件运行，在演示窗口中将出现两个虚线框。调整“文本说明”虚线框，使它包围地球图形，如图 18.3 所示。



图 18.3 运行画面

⑨ 通过上面的操作步骤，用户再次运行项目文件的时候，如果在地球图形对象上单击，程序将弹出“文本说明”显示图标中的内容。经过本例之后，用户应该对于热区交互有个初步的了解。



当用户已经为交互图标添加了交互分支之后，再次添加交互分支时，系统不会弹出 Response Type 对话框，并且后添加的交互分支将继承前一交互分支的类型属性。如果用户希望更改后面交互分支的交互类型，需要双击该路交互分支的交互标签，在打开的 Properties : Response 对话框中更改。程序运行过程中，按下 Ctrl + P 键可以暂停程序运行，同时显示当前演示窗口中存在的所有对象，包括非媒体对象，如热区交互定义的热区范围等。

实例

19

地球 (2)



实例说明

当设置交互分支时,可以使用空交互分支来实现返回程序,本例就在上一例的基础上设计交互分支来实现程序返回。实例运行后,演示窗口的最终效果如右图所示。

主要知识点:交互图标。



操作步骤

- ① 确定当前打开的是“19 地球.a7p”项目文件(将实例18中的文件重命名得到),在流程线上双击 nothing 分支的组图标,打开该组图标的流程线窗口。
- ② 从图标栏上拖动一个计算图标添加到 nothing 组图标流程线窗口中,更改该计算图标名称为“空定义”。
- ③ 在流程线上双击“空定义”计算图标,打开该计算图标的脚本编辑窗口。在打开的脚本编辑窗口中键入:“该脚本为空定义,不执行任何操作”,如图19.1所示。



图 19.1 设置 Nothing

④ 在流程线上选中 nothing 组图标,使其高亮显示,然后选择菜单 Modify,在打开的下拉菜单中选择 Ungroup,此时组图标将消失,代替它的是内部计算图标,如图19.2所示。

⑤ 同时按下 Ctrl + R 键运行项目文件,在地球图形对象上单击鼠标,程序将执行“文字说明”交互分支下的显示图标,然后在演示窗口非地球图形区域上单击鼠标,程序执行



nothing 交互分支，提示文本消失。

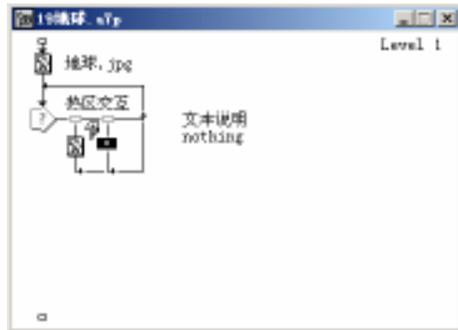


图 19.2 取消组

- ⑥ 通过上面的操作过程，用户应该掌握如何使用计算图标来创建空的程序流程。

实例

20

动物乐园



实例说明

本例的目的是在程序运行时,在画面上显示各种动物,如果操作者不知道它们的名称,用鼠标单击某个动物,在该动物的旁边就会出现该动物的名称。

主要知识点:交互图标。



操作步骤

① 新建文件,保存为“20 动物乐园.a7p”。

② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,选择 Center On Screen 复选框,取消选择 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择白色,如图 20.1 所示。



图 20.1 设置文件属性

③ 编辑如图 20.2 所示的流程线。在弹出 Response Type (交互类型)对话框时,选择 Hot Object 单选按钮,然后将其他 4 个显示图标同样放置在 Interaction (交互)图标的右侧。在狮子、狗、兔子和笼子 Display (显示)图标中分别插入它们的图片,最后还要将各个图标进行命名。



说明

在热区交互程序中,所有的画面可以放置在一幅图中,而在热对象交互的制作中却不可以,热对象中的每一个对象必须单独放置在一个 Display (显示)图标中。

④ 将所有的图片都插入图标后,运行程序时,所有的动物图片出现后立即消失,而



画面上最后只保留了一幅笼子的画面，实际上是要将所有的动物都显示在笼子的后面。选择“笼子”显示图标，然后按 Ctrl+I 快捷键打开属性对话框，如图 20.3 所示，在 Layer 文本框中输入图片的层为“-1”，这样笼子就不会将所有的动物图片挡在后面了。

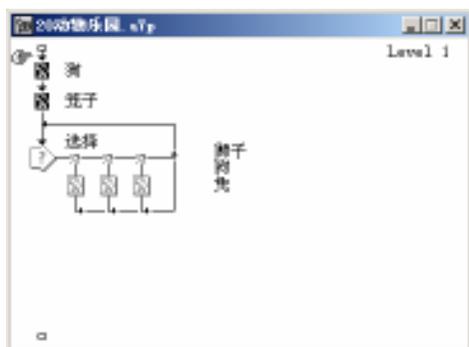


图 20.2 流程线

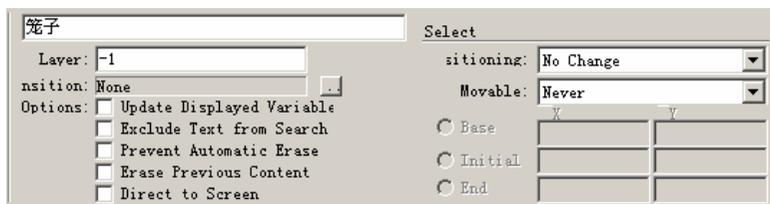


图 20.3 设置层



技巧

在 Authorware 的层设置中，如果 Layer 文本框内没有任何值，系统默认为“0”。

⑤ 在设置图标层之前，如果想将动物图片和笼子图片显示在同一窗口中，可以按住 Shift 键，然后用鼠标双击流程线上的“狗”显示图标，在窗口中就会同时出现“狗”的图片，如图 20.4 所示。“狗”图片的背景是白色的，它会将笼子的一部分遮住，因此，还要打开重叠模式选择框，选择其中的 Inverse（反显）模式。设置完毕，最好看一看效果如何。

⑥ 将所有的动物图片都设为反显模式后，单击 Restart 按钮后，屏幕上出现如图 20.5 所示的 Properties : Response（交互属性）对话框。同时，该对话框有标题提示 Click an object to make it the hot object，意思是让用户单击画面上的一个物体作为热对象。单击“狗”对象，在对话框的 Hot Object（热对象）文本框内就会显示该对象图标的名称。

⑦ 单击对话框的 OK 按钮确认后，Authorware 又会弹出下一个交互属性对话框，要求继续指定下一个热对象，用同样的方法设置其他交互属性对话框。

⑧ 在交互分支的各个显示图标中输入对应的动物名称，如图 20.6 所示。回到流程线后，如果用户觉得流程线太长，可以将它们进行归组。用鼠标将 Interaction（交互）图标上方的所有 Display（显示）图标都选中，然后，单击 Modify 菜单下的 Group 命令，所有

的 Display (显示) 图标就会转化为一个 Group (群组) 图标, 将它命名为“动物”。

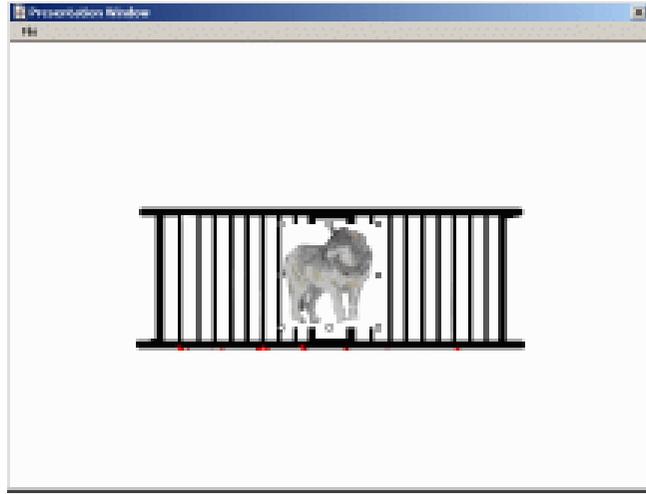


图 20.4 设置重叠模式



图 20.5 选择热体

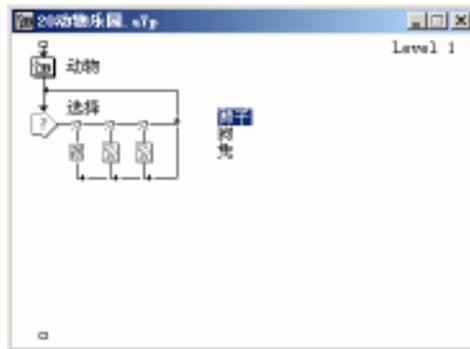


图 20.6 图标归组

⑨ 在交互属性对话框的 Match 下拉列表框中的默认选项为 Single-Click, 因此在程序运行后, 单击画面上的动物后, 就会在屏幕上出现该对象的名称。当然, 也可以设置双击等方式。



实例

21

鸟叫声



实例说明

所谓的热体交互，和热区交互类似，都是在演示窗口中制定某个范围或对象，作为交互发生的对象。这两者不同之处在于，热区交互定义的对象是演示窗口中的一个矩形区域，而热体交互制定的是某个对象，热体交互的范围是和指定的对象相关的，因此它不局限于规则的矩形区域。本例用热体交互来设计鸟叫声，程序流程图如右图所示。



主要知识点：交互图标、热体交互。



操作步骤

- ① 新建文件，保存为“21 鸟叫声.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框 取消选择 Menu Bar 复选框 单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 21.1 所示。

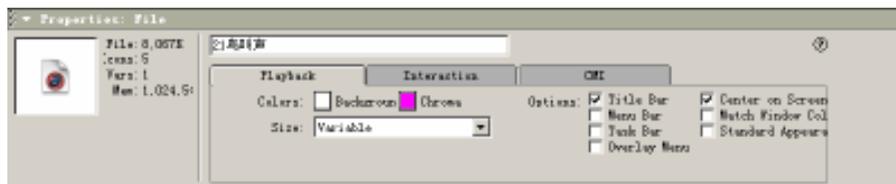


图 21.1 设置文件属性

- ③ 从图标栏上拖动一个计算图标到流程线上，更改该计算图标名称为“设置窗口大小”。在该计算图标的脚本窗口中设置大小，如图 21.2 所示。
- ④ 在流程线窗口中用鼠标单击计算图标下方的流程线，确认粘贴小手出现在计算图标的下方，选择菜单 File，在打开的下拉菜单选项中选择 Import...，导入带有小鸟图案的图形文件。
- ⑤ 从图标栏上拖动一个交互图标到流程线上，放在“鸟”显示图标的下面，更改该交互图标名称为“热体交互”。从图标栏上拖动一个声音图标为交互图标添加第一路热体交互。在弹出的如图 21.3 所示的交互类型对话框中选择 Hot Object。

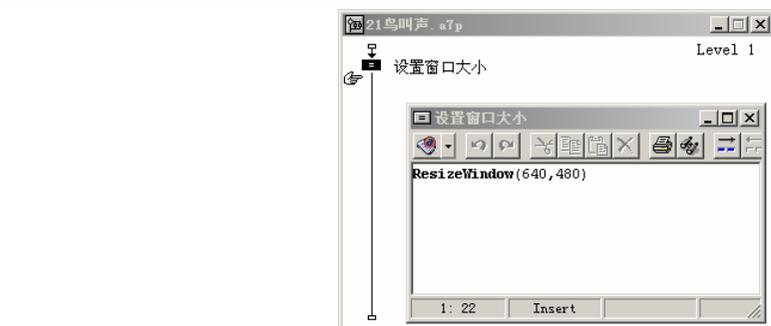


图 21.2 设计流程线

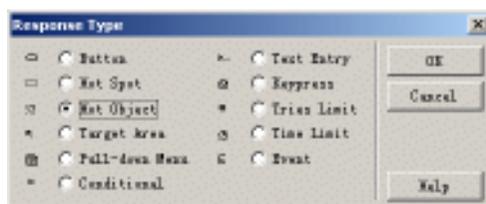


图 21.3 Response Type 对话框

⑥ 在流程线窗口中将热体交互分支更名为“声音交互”，双击添加的声音图标，系统将打开 Properties : Sound Icon 对话框，如图 21.4 所示。

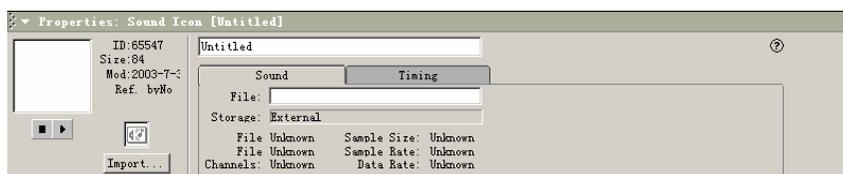


图 21.4 “声音交互”属性框

⑦ 在 Properties : Sound Icon 对话框中，单击该对话框左下角的 Import... 命令按钮，系统将打开 Import Which File? 对话框，在该对话框中选择带有鸟叫声的声音文件。

⑧ 在 Import Which File? 对话框中单击 Import 命令按钮导入声音文件，返回流程线窗口，同时按下 Ctrl + R 键运行项目文件，系统自动弹出 Properties : Response 对话框，在演示窗口中单击小鸟图形完成设置。



说明

热体交互的对象是在同一个图标中，例如显示图标中的所有对象，因此应该将需要作为热体的对象放在某个单独的显示图标中。对于声音文件，Authorware 专门提供了声音图标对其进行管理。当用户在声音图标中导入了声音文件之后，在声音图标的属性对话框中，用户还可以看到导入的声音文件的各种属性，包括声音文件的大小、文件格式、声音通道以及采样率等等。在声音图标属性对话框中，用户还可以在 Timing 选项卡中指定声音文件的播放效果，在本例中，系统默认的设置已经满足了程序的要求。



实例

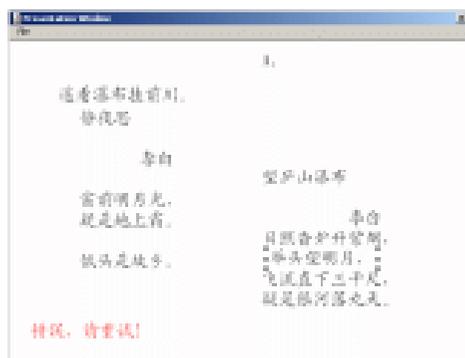
22

巧对唐诗



实例说明

本例制作一个对唐诗的游戏，如右图所示的画面上有两首古诗，每首诗都缺少一句，要求用户来选择正确答案，并将答案拖至诗中正确的位置。如果选择错误，选择的诗句会返回原处，并且画面上会显示错误信息；如果选择正确，诗句会停留在目标处，窗口中会显示鼓励性的文字。本例介绍的目标区域响应就能达到这种效果。



主要知识点：交互图标、目标区域交互。



操作步骤

- ① 新建文件，保存为“22 对诗.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 22.1 所示。

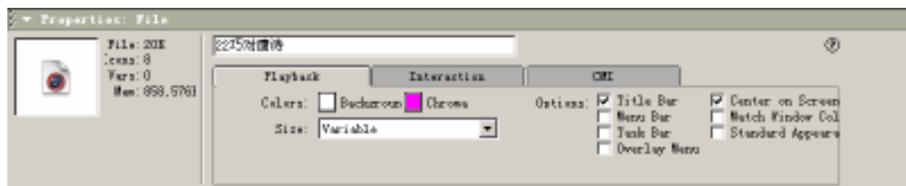


图 22.1 设置文件属性

- ③ 根据题目的要求，可以编辑如图 22.2 所示的流程线。在“诗”图标中放置程序运行的主画面，在图标 A、B 中分别输入要考的诗句 A 和 B。放置交互分支的 Display(显示)图标时，在交互类型对话框中选择 Target Area(目标区域交互)。



技巧

与热对象交互的设置方式一样，在目标区域交互中的每一个移动对象也必须单独放置在一个 Display(显示)图标中。

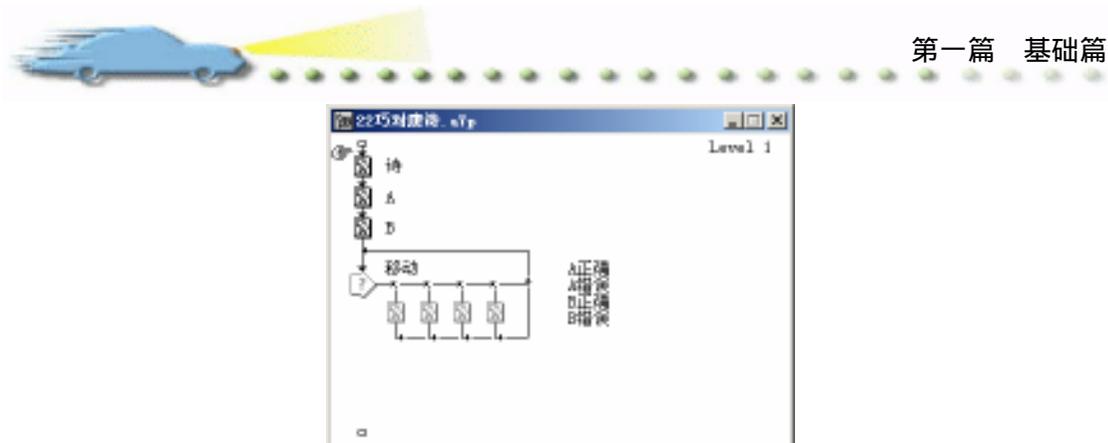


图 22.2 流程线

④ 将流程线组建完毕后，单击工具栏上的“运行”按钮，窗口中就会弹出交互属性对话框。如图 22.3 所示，在窗口中会出现一个矩形区域，这个矩形的大小和位置就是目标区域。对话框的标题提示为 Select a target object，提示用户选择一个目标对象，用鼠标单击画面上的“举头望明月”图片（该诗句是正确选择），在 Target Object（目标对象）文本框中就会出现该图片对应的标题。

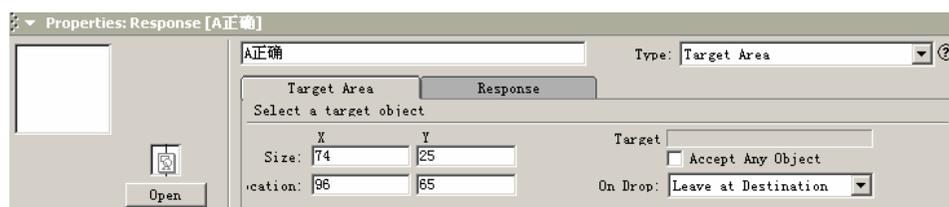


图 22.3 选择目标对象

⑤ 单击对象后，对话框的标题又改为 Drag object to the target position，要求将对象拖至画面上的目标位置。在将对象移至目标位置后，目标区域也同时自动到达该位置，如图 22.4 所示，将目标区域调节至与填充区域同样大小。这样，所设的区域就成为 A 诗句的正确目标区域。

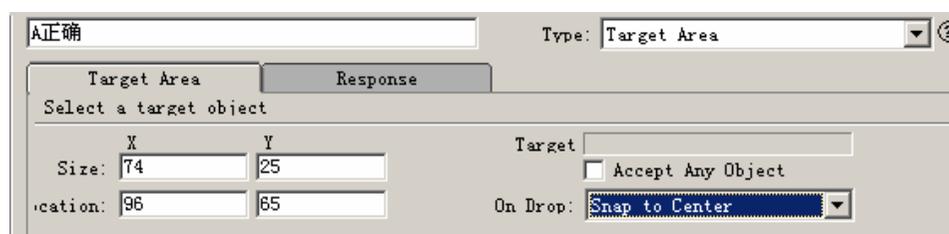


图 22.4 交互属性对话框

⑥ 在 On Drop 列表框中有 3 个选项，程序运行时，在将对象拖动释放后，对象在窗口中的位置就由它们来控制。单击 Snap to Center 选项，在程序运行时，当用户最终将对象拖到正确目标区域释放后，Authorware 将自动把该对象放置在目标区域的中央。此选项常常是用来设置对象移动正确的选项。



技巧

(1) 单击 Leave at Destination 选项, 在程序运行时, 当用户将对象拖到正确目标区域附近处释放, 对象将停留在当前位置, 但此位置并不一定是目标区域的中央。

(2) 单击 Put Back 选项, 在程序运行时, 当用户拖动对象的位置不正确, 对象将返回原处。

⑦ 单击 Accept Any Object 选项, 设定的目标区域将会接收任何对象, 而不管对象移动是否正确。

⑧ 设置完“ A 正确 ”对话框, 单击 OK 按钮。Authorware 将弹出下一个分支的交互对话框要求设置错误区域。如图 22.5 所示, 同样选择诗句 A 作为对象, 然后用鼠标将目标区域调至全屏大小。由于此分支设置为错误选项, 所以应设置 Put Back 选项, 这样, 当对象的拖放位置不正确, 该对象会返回原位置要求重试。

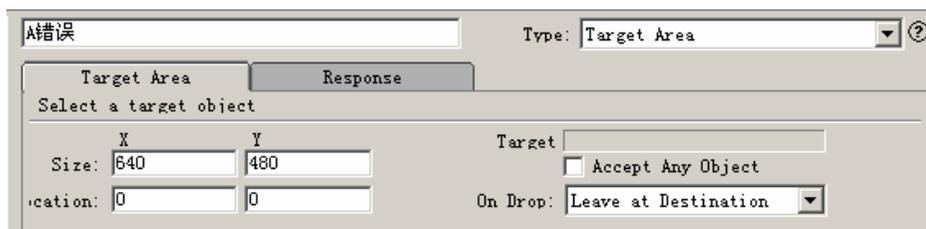


图 22.5 设置错误区域



技巧

由于设置程序的运行窗口大小是 640 × 480, 因此可以直接在对话框的 Size 和 Location 文本框中输入相应的坐标。

⑨ 单击工具栏上的运行按钮运行程序, 当画面出现时, 单击流程线上的 Interaction (交互) 图标, 在窗口中就会出现如图 22.6 所示的所有目标范围。可以根据对象来调整目标区域。



图 22.6 目标区域

⑩ 运行程序后，程序并没有达到所要求的效果，不管移动对象是否正确，对象总会在释放位置停留。问题出现在图片的移动属性上，前面提到过 Display(显示)图标的 Layer 选项卡，里面的选项就能实现图片各种类型的移动属性。打开显示图标属性对话框，如图 22.7 所示，选择 Movable 下拉列表框中的 Anywhere(任何位置)选项，设置的图片对象就可以被移动了。

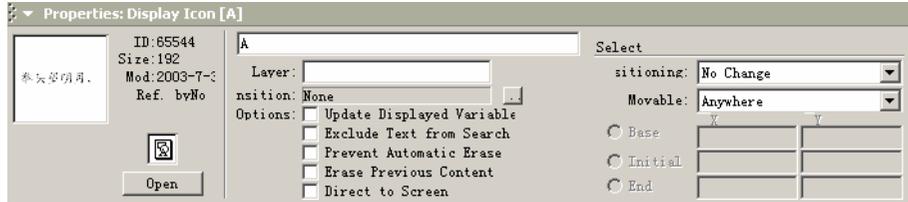


图 22.7 设置对象移动属性

⑪ 最后，再在“ A 正确 ”、“ A 错误 ”、“ B 正确 ”和“ B 错误 ”显示图标中分别输入“ 正确，请继续！ ”和“ 错误，请重试！ ”文字。运行程序时，画面上就会出现判断结果了。



实例

23

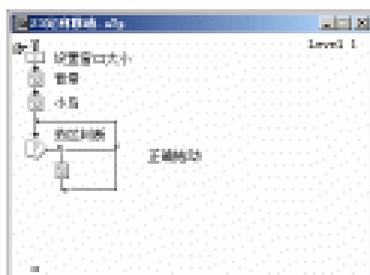
小鸟图案的定向移动



实例说明

使用目标区域交互制作一个实例，实现将小鸟移到圆内。程序流程图如右图所示。

主要知识点：目标区域交互。



操作步骤

- ① 新建文件，保存为“23 定向移动.a7p”。
- ② 执行命令 Modify File Properties，打开

属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 23.1 所示。



图 23.1 设置文件属性

③ 从图标栏上拖动一个显示图标，更名为“背景”，双击该显示图标，在打开的演示窗口中使用系统提供的工具绘制背景图，并且使用圆形工具创建一个圆形。

④ 在“背景”显示图标的下面添加一个显示图标，使用绘图工具箱的时候，同时按下 Ctrl + K 键可以快速打开颜色设置对话框。使用 Import... 命令功能导入小鸟的图形文件。

⑤ 从图标栏上拖动一个交互图标到流程线上，更改该图标名称为“热区判断”。从图标栏上拖动一个显示图标，为该交互图标添加第一路交互分支，在弹出的 Response Type 对话框中选择 Target Area，如图 23.2 所示。

⑥ 更改第一交互分支名称为“正确拖动”，在流程线上双击该显示图标，在打开的演示窗口中，使用文本工具添加正确拖动小鸟后的反馈信息文字。

⑦ 从图标栏上继续拖动一个显示图标，为“热区判断”交互图标添加第二路交互分支，由于交互属性的继承性，该路分支也为热区交互。更改该路交互分支名称为“错误判断”，在该路显示图标中添加错误拖动的反馈信息。

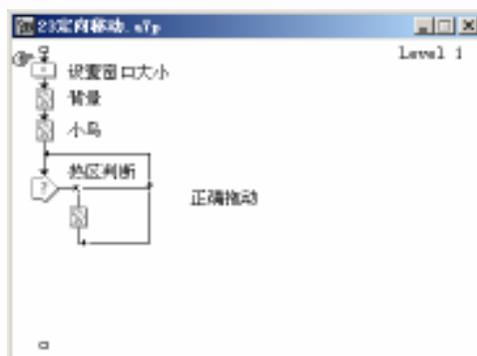


图 23.2 流程线

⑧ 通过上面的操作步骤，已经完成了流程线的程序设计，剩下的工作就是完成交互分支属性的设置，包括设定拖动的目标区域，正确拖动范围和错误拖动条件的设置。



实例

24

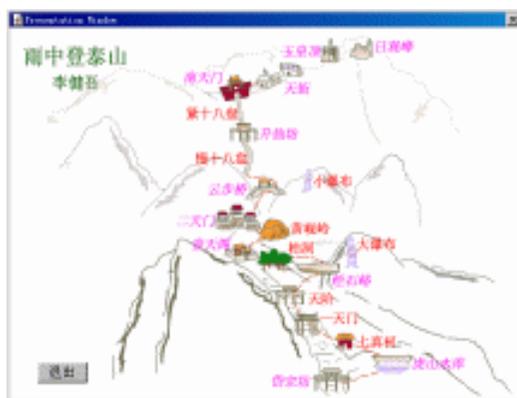
雨中登泰山



实例说明

本节制作实例——雨中登泰山，作为高中语文课本中《雨中登泰山》的课件，本例是利用“热区交互”制作的。程序运行效果如右图所示。

主要知识点：热区交互。



操作步骤

① 新建文件，保存为“24 雨中登泰山.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 24.1 所示。

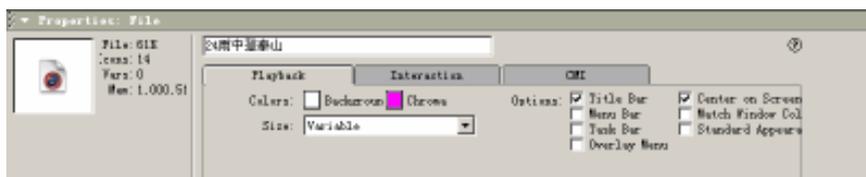


图 24.1 设置文件属性

③ 设计程序流程线。从面板中拖放一个显示图标到流程线上，命名为“背景”。

④ 双击显示图标，打开演示窗口，为“背景”导入一张图片。然后选取 Options 下拉列表中的 Scaled 选项。单击 OK 按钮完成设置。

⑤ 从面板中拖放一个交互图标到流程线上，命名为“简要介绍”。再拖放一个组图标到交互图标右边，命名为“岱宗坊”，选择交互类型为 Hot Spot，并将鼠标指针设置为手形鼠标指针，如图 24.2 所示。

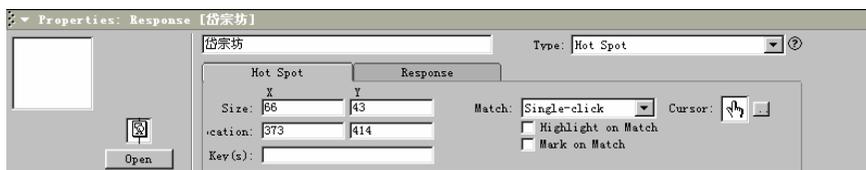


图 24.2 设置响应属性

⑥ 重复以上操作，创建 10 个新的显示图标，拖放到交互图标右边，分别命名为“虎山水库”、“经石峪”、“壶天阁”、“二天门”、“云步桥”、“升仙坊”、“南天门”、“天街”、“玉皇顶”和“日观峰”。完成以上步骤后，设计窗口如图 24.3 所示。

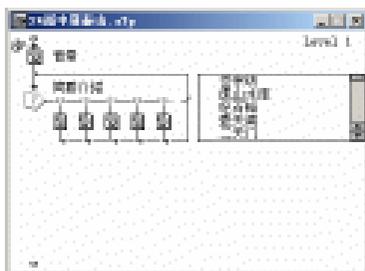


图 24.3 流程线

⑦ 双击“岱宗坊”图标，打开子程序设计窗口，创建新的显示图标，命名为“岱宗坊”，然后在演示窗口中创建新文本，输入“岱宗坊”的简要介绍。其他显示图标也同样进行设置。

⑧ 在流程线上添加一个 Calculation 图标，命名为“退出”。双击“退出”图标，打开 Calculation 编辑窗口，输入 Quit()。

⑨ 打开交互图标的属性对话框，设置 Button 选项卡，在 Type 下拉列表中选择 Button 选项，此时演示窗口出现“退出”按钮，调整其位置，如图 24.4 所示。

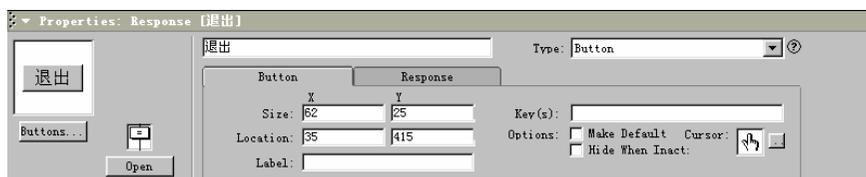


图 24.4 设置响应属性

⑩ 双击“背景”显示图标打开演示窗口，按住 Shift 键不放，双击“简要介绍”，此时演示窗口出现 11 个 Hot Spot 选择虚线框。使用鼠标拖动虚线框放到背景中的相应位置，如图 24.5 所示。

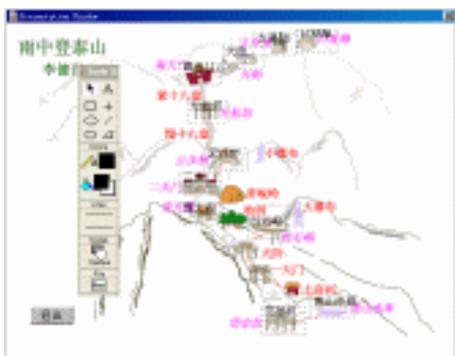


图 24.5 设置 Hot Spot 范围

⑪ 预览运行效果，并保存程序文件。



实例

25

家庭影院



实例说明

本例制作一个“家庭影院”，目的是介绍在 File 菜单的基础上扩展 Authorware 的菜单条，然后在各个菜单中再添加一些相关的命令。对于菜单中的各条命令不能一一进行详细地讲解，只是想通过这个例子来了解制作菜单交互的思路及一般菜单命令的制作方式。

主要知识点：交互图标、下拉菜单交互。



操作步骤

- ① 新建文件，保存为“25 家庭影院.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 25.1 所示。



图 25.1 设置文件属性

③ 如效果图所示，制作的这个作品一共有 3 个菜单，它们分别是 File 菜单、“控制”菜单和“帮助”菜单。

④ 对应图所示的 3 个菜单，在流程线上要使用 3 个 Interaction（交互）图标来分别创建。如图 25.2 所示，File 交互图标与图的 File 菜单对应，“控制”图标与图“控制”菜单对应，“帮助”图标与图的“帮助”菜单对应。Interaction（交互）图标的各个分支标题就是菜单中的各个命令。除此之外，Authorware 也可以制作子菜单，而子菜单的制作过程比较复杂，在一般的多媒体作品中使用菜单方式即可。



说明

Authorware 演示窗口的菜单 File 不能被取消，更不能更改为中文的“文件”菜单，但可以在 File 菜单中添加一些命令来丰富它。

⑤ 打开作品中的 File 菜单，如图 25.3 所示，在两个菜单命令之间有一条分隔线，在平时所接触的菜单中也会经常看到。这条分隔线的制作非常简单，如图 25.2 所示，只需在“打开”和“退出”分支之间加一条交互分支，然后将分支标题命名为“——”即可。

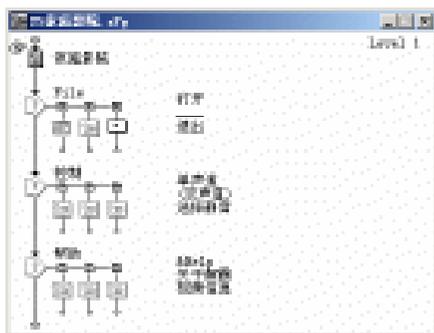


图 25.2 流程线



图 25.3 File 下拉菜单

⑥ 从图 25.3 中还可以看到，在菜单命令的右面都对应它的快捷键，例如，“打开”命令的快捷键是 Ctrl+O，单击“打开”命令和按 Ctrl+O 快捷键都能执行“打开”命令。单击交互类型按钮，打开交互属性对话框，如图 25.4 所示，可以在 Menu (菜单) 选项卡的 Key(s) 文本框中输入快捷键。快捷键的格式是功能键再加上字母键，如图中的 Ctrl+O 就代表快捷键 Ctrl+O。

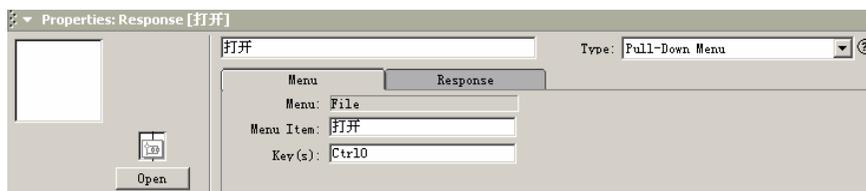


图 25.4 设置快捷键

⑦ 如图 25.5 所示，在“控制”菜单中有一个灰色命令，该命令在程序执行时不可用。要制作此命令效果，用户可以在图 25.2 的“控制”交互图标的“双声道”标题上加“()”。实际上，这样制作的灰色命令没有多大的实用价值，而真正有用的灰色命令在使用时会被激活，通过设置 Active If 文本框中的变量可以完成这一效果。单击 Response 标签，打开 Response 选项卡，在相应文本框中设置即可。例如，可以在 Active If 文本框中输入一个变量表达式 Use=1，这样，当 Use=1 时，命令变灰色；当 Use=0 时，命令就变为黑色。

⑧ 打开“帮助”菜单，如图 25.6 所示，菜单中的 Help 命令的 H 下面有一条下划线，这也代表一种快捷键的方式。即打开 Help (帮助) 菜单，然后按下 H 键也能执行该命令。



要实现上述的命令方式，可以在快捷键字母前加上“&”字符。若要使“&”也作为快捷键，可用“&&”来表示。



图 25.5 灰色命令



图 25.6 下划线快捷键



技巧

如果快捷键的字母是单词的第二个字母，可以将&加在第二个字母前面，依次类推。

⑨ 图 25.2 中流程线的形状在前面曾经提到过，用户可以打开 Response 选项卡，如图 25.7 所示，选中 Perpetual (持续性交互) 选项，流程线就会直接穿过 Interaction (交互) 图标。当一个交互被设置为持续性交互响应时，在整个程序的执行过程中可以随时匹配这个交互。

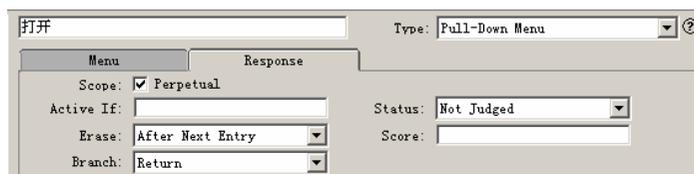


图 25.7 设置永久属性



技巧

选中 Perpetual 复选框后，Branch 下拉列表框中增加了 Return 选项。由于本程序的设置为持续性菜单交互，因此可以随时使用菜单中的命令。

⑩ 在 File 菜单下制作了一个“退出”命令，该命令的功能是退出程序，即在程序运行时单击“退出”命令，程序就会终止。要实现该功能，可以双击图 25.2 中的“退出”计算图标，在打开的文本框中输入 Quit(0)命令，如图 25.8 所示。

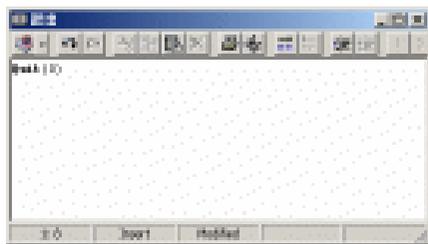


图 25.8 退出函数

⑪ 用同样的方法，用户也可以设置帮助信息，这涉及到使用 Authorware 的 JumpFile 函数，此处不再讲解。

实例

26

自定义 Beep 下拉菜单 (1)



实例说明

本例使用下拉菜单交互制作一个选择菜单，即让计算机发出指定的声音。

主要知识点：交互图标、下拉菜单交互。



操作步骤

① 新建文件，保存为“26 自定义 BEEP 下拉菜单.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 26.1 所示。

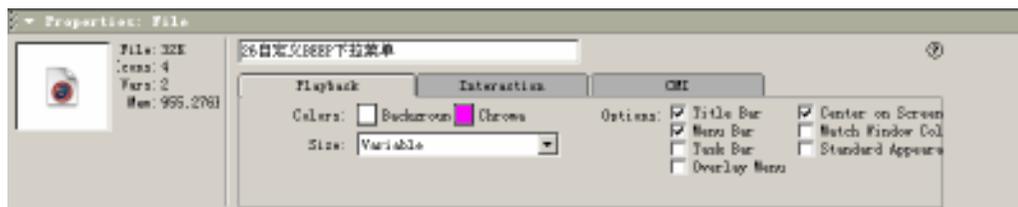


图 26.1 设置文件属性

③ 从图标栏上拖动一个显示图标，更名为“背景”。双击该显示图标，在打开的演示窗口中导入一幅图形文件作为背景，然后添加提示文字：选择 Beep 菜单，让计算机发声，如图 26.2 所示。

④ 从图标栏上继续拖动一个计算图标，为 Beep 交互图标添加第二路交互分支，更名为 Beep 2。双击该计算图标，在脚本窗口中重复使用 Beep 函数，使计算机发出两下 Beep 声，如图 26.3 所示。

⑤ 返回流程线窗口，选择菜单 Ctrl + R 运行项目文件，在运行的程序演示窗口中，用户可以看到自定义的 Beep 菜单。单击该菜单，可以打开下拉菜单，选择其中的任何一个菜单选项，计算机将按程序设定发声。



图 26.2 设置背景窗口



图 26.3 设置 Beep 函数



说明

(1) 在该项目文件中, 用户必须保证文件属性设置对话框中选择了 Menu Bar 选项, 也就是说, 选择了显示菜单条。关闭计算图标的脚本窗口时, 可以按下小键盘上的 Enter 键快速关闭, 如果脚本没有错误, 系统不会出现确认对话框, 从而节省编程时间。当使用交互图标创建下拉菜单交互时, 交互图标的名称就是菜单的名称, 它下挂的交互分支的名称就是打开的下拉菜单选项的名称。

(2) 在本例中使用了系统的 Beep() 函数, 该函数的功能是计算机的扬声器发出 Beep 声音。对于用户创建的下拉菜单选项交互, 通常应该将该交互放在程序流程的最上面, 每路交互都设置成永久交互, 返回方式设置为 Return。

实例

27

自定义 Beep 下拉菜单 (2)



实例说明

所谓的菜单选项快捷键,就是当用户直接按下该快捷键之后,实现的程序功能和用鼠标选择该菜单选项相同。本例对上例设计的发声程序设置快捷键。

主要知识点: Beep()函数。



操作步骤

- ① 新建文件,保存为“27 自定义 BEEP 下拉菜单.a7p”。
- ② 执行命令 Modify File Properties, 打开属性对话框, 设置 Size 为 Variable, 选择 Center On Screen 复选框, 取消选择 Menu Bar 复选框, 单击 Color 右边的 Background Color 按钮, 打开 Color 对话框, 从中选择白色, 如图 27.1 所示。

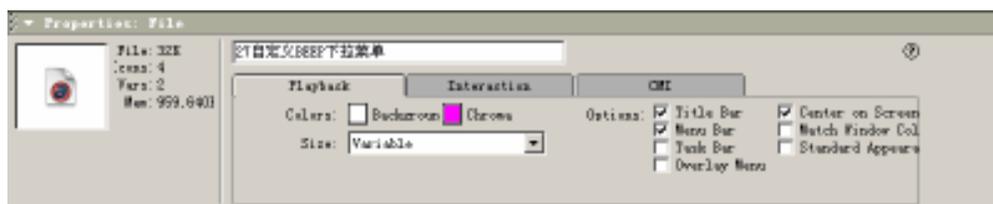


图 27.1 设置文件属性

- ③ 设置程序流程线如图 27.2 所示, 在流程线上按下 Ctrl 键的同时双击 Beep1 交互分支的交互标签, 快速打开 Properties : Response 对话框。

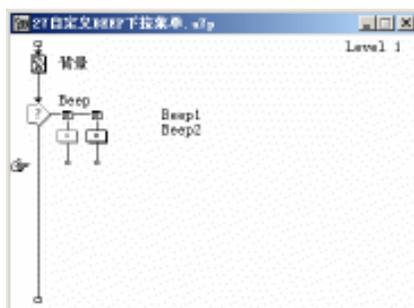


图 27.2 流程线



④ 为了给 Beep1 菜单选项添加快捷键,在 Key(s)文本框中键入“1”,从而设定 Ctrl_1 为该菜单选项的快捷键,如图 27.3 所示。

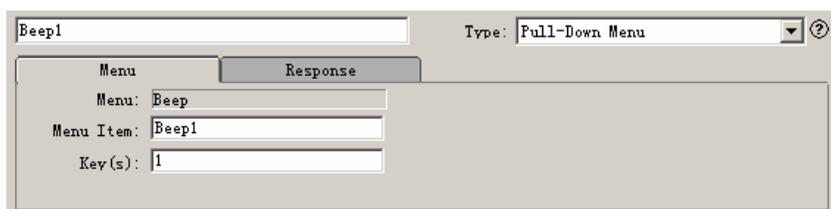


图 27.3 设置快捷键

⑤ 单击对话框中的 Response 标签,打开 Response 选项卡。选中 Scope 前面的检查框,然后在 Branch 下拉菜单中选择 Return。

⑥ 参照 Beep 1 交互分支的设置,完成 Beep 2 交互分支的 Properties : Response 对话框。定义该路交互分支菜单选项快捷方式为 Ctrl + 2。

⑦ 当设置了这两路交互分支的 Response 属性之后,返回到流程线上,用户会发现程序流程的显示和先前不同。注意对于 Return 返回类型的交互程序流程的显示。

⑧ 在程序流程线窗口,再次按下 Ctrl + R 键运行项目文件,在打开的程序窗口,选择 Beep 菜单,打开 Beep 下拉菜单,可以看到每项菜单后面的快捷键显示,如图 27.4 所示。



图 27.4 Beep 菜单



说明

在 Authorware 中,默认的快捷键的组合为 Ctrl 键,也就是说,在 Key(s)中键入 1,那么快捷键就定义为 Ctrl+1。如果用户希望定义 Alt 键的组合,那么使用 Alt_1 即可。在 Response 选项卡中选择 Perpetual 前面的检查框,即可将该交互定义为永久交互。对于交互分支的返回选项 Branch 设置,只要设置了永久交互属性的交互分支才可以设置 Return 选项。对于那些 Return 类型的交互分支,当程序执行到该交互图标时,系统会记录存在的交互,然后继续下面的程序流程,无论用户在何处进行该类型的交互,系统都将执行。对于不同的 Branch 选项设置的交互分支,在程序流程线上显示不同。

实例

28

健康咨询



实例说明

一个健康咨询的例子。在这个例子中先做如下假设：若体重（斤）与身高（厘米）的比值大于 0.8 时，略显肥胖；比值小于 0.6 时略显消瘦；当比值在 0.6 与 0.8 之间时，则肥瘦适中。在这个例子中，要定义一个变量来代表体重与身高的比值，如下文所用的 Result。如右图所示，这是程序运行后的画面，画面上病人在向医生询问病情，如果在画面上的光标处输入你的体重和身高值，按回车键后，医生就会给出正确的答复。



主要知识点：交互图标、条件交互。



操作步骤

- ① 新建文件，保存为“28 健康咨询.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 28.1 所示。

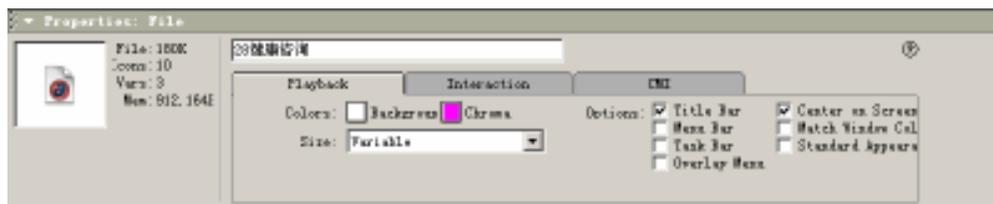


图 28.1 设置文件属性

- ③ 设置程序的流程线如图 28.2 所示，在黑圈内的流程用来实现文本输入，即病人往程序中输入的体重值和身高值。在 Calculation（计算）图标中，通过设置一对变量来获取这组值。由于屏幕输入的值要赋予 Calculation（计算）图标中的变量，所以该分支标题要



虚设，即在交互属性对话框的 Pattern 文本框内不输入任何字符。同时，此处的交互分支类型应设为 Exit Interaction，这样，在输入数值后，程序将继续往下运行。

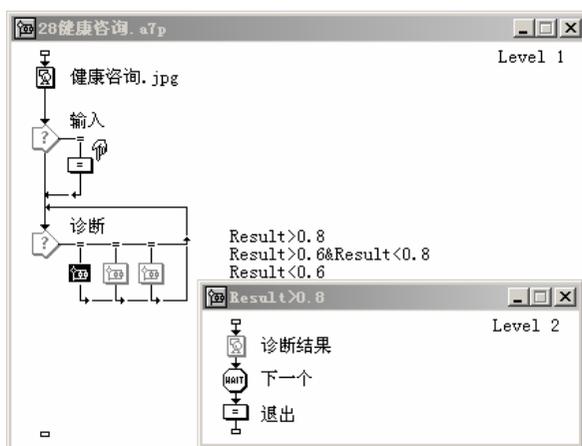


图 28.2 流程线

④ 双击 Calculation (计算) 图标，打开计算图标文本框。如图 28.3 所示，在文本框内输入 `a:=NumEntry` 和 `b:=NumEntry2`，其中 NumEntry 是一个系统变量，它会获取键盘的输入值。因此，这两条语句的功能是分别将键盘输入的的第一个值和第二个值赋予变量 a 和 b。而 `Result:=a/b` 语句的功能是将 a 和 b 的比值 (也就是体重与身高的比值) 赋予变量 Result。

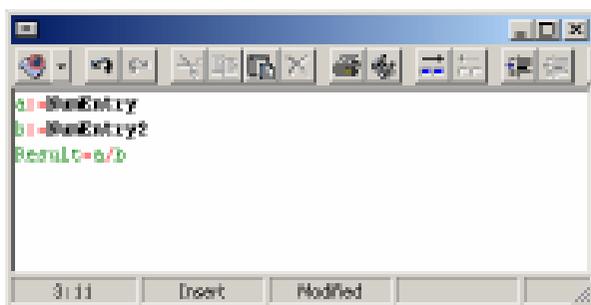


图 28.3 设置计算图标

⑤ 单击文本框上角的“关闭”按钮，Authorware 会弹出一个 New Variable (新变量) 对话框，该对话框要求给新变量赋初值及进行变量描述，如图 28.4 所示。在 Initial Value 文本框中输入变量 a 的初值“0”，然后关闭对话框。接下来，Authorware 还会弹出 b 和 Result 的新变量对话框，同样可以按上述方法设置它们。

⑥ 返回流程线，双击交互类型按钮，打开 Properties : Response (交互属性) 对话框。如图 28.5 所示，在 Condition 文本框中输入 `Result > 0.8`，这样，当变量 Result 的值为大于 0.8 的数值时满足该条件，Authorware 将执行该分支。

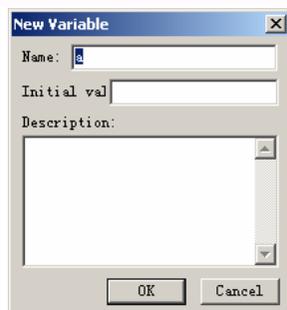


图 28.4 变量赋初值



图 28.5 设置条件

⑦ 打开 Automatic 下拉列表框,如果选择 When True 选项,在程序运行时,Authorware 就会根据条件变量的值来判断交互是 True 还是 False,一旦符合条件,Authorware 将执行此交互分支;若选择 On False to True 选项,则只有在条件由假变为真时,Authorware 才会执行交互响应;如果选择 Off 选项,Authorware 将关闭判断功能。因此,只有在指明条件是正确时才能产生交互。

⑧ 在 Automatic 列表框中选择 When True 选项后,单击 OK 按钮。打开交互分支中的 Result > 0.8 群组图标,在流程线上放置 Display (显示) 图标用于存储诊断结果的画面。打开“返回”计算图标,在文本框中输入 GoTo (IconID@ "输入") 语句,这样当程序执行到 Calculation (计算) 图标时,流程将跳转到输入主流程线的“输入”显示图标处,如图 28.6 所示。还要在流程线设置一个 Wait (等待) 图标,该图标的作用主要是使“诊断结果”画面暂停,同时,在屏幕上显示一个标签为“下一个”的等待按钮。

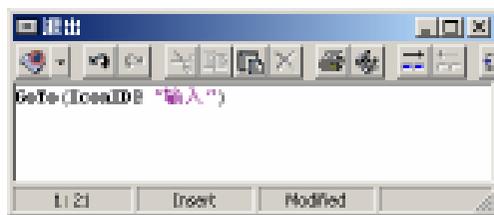


图 28.6 设置程序跳转

⑨ 下面再来看一下程序流程是怎样执行的。如图 28.7 所示,假如在光标处输入 100/180 这一组数值,当它进入 Calculation (计算) 图标,变量进行赋值。即 $a=100$, $b=180$ 程序, $Result=a/b=0.55$ 。

⑩ 当变量 Result 进入 Interaction (交互) 图标后,由于 $Result=0.55 < 0.6$,所以它将执行第三个分支 $Result < 0.6$,如图 28.7 所示,当执行分支中的群组图标时,首先显示“诊



断结果”显示图标。然后，执行 Wait (等待) 图标，即显示画面左上角的“下一个”按钮。单击此按钮后，程序执行 Calculation (计算) 图标中的 GoTo 语句，程序将再次返回到流程线的最开始端，等待再次输入下一组值。

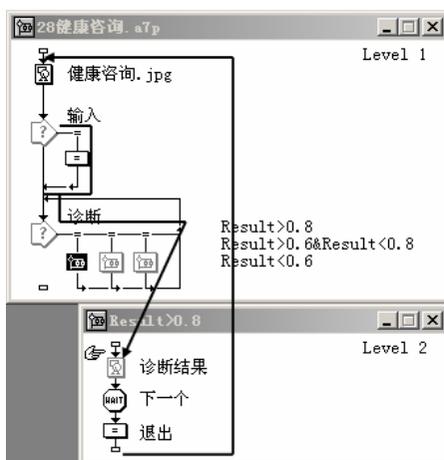


图 28.7 程序流向



技巧

使用 GoTo 语句还有另外一个作用：如果把“返回”计算图标去掉，当输入一组数值后，程序将永久不断地执行一条分支。使用该图标将避免这一现象。

实例

29

智力问答



实例说明

本例使用文本输入交互来制作一个智力问答游戏。

主要知识点：交互图标、交本输入交互。



操作步骤

- ① 新建文件,保存为“29 智力问答.a7p”。
- ② 执行命令 Modify File Properties,

打开属性对话框,设置 Size 为 Variable,选择 Center On Screen 复选框,取消选择 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择白色,如图 29.1 所示。

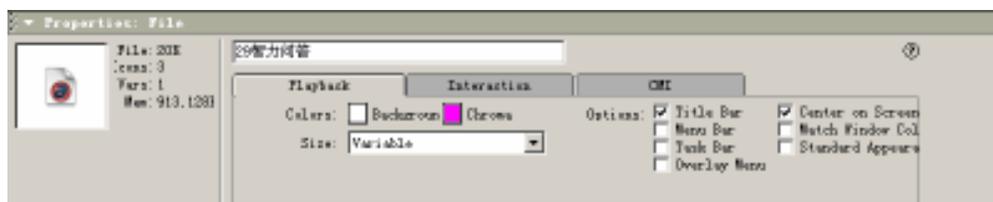


图 29.1 设置文件属性

- ③ 建立如图 29.2 所示的程序流程线。

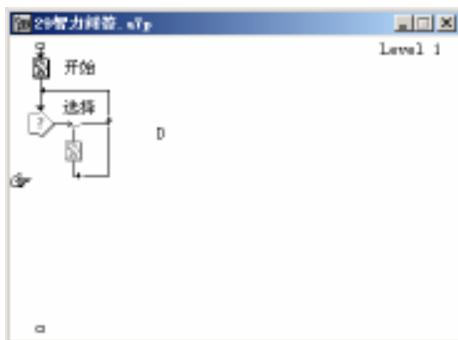


图 29.2 流程线

④ 在流程线上选择 Interaction (交互) 图标,然后打开其交互属性对话框,最后单击对话框中的 Text field(文本区域)按钮,如图 29.3 所示,Authorware 将会弹出 Properties :



Interaction Text Field 对话框。



图 29.3 改变文本输入区域

⑤ 同时，图 29.3 中的演示窗口中出现一个黑色三角，当程序运行时，该三角符号后面将出现闪烁的光标，就可以在后面输入各种文字了。如图 29.3 中所示，用鼠标拖动文本输入区域的调整方块可以改变它的大小，它的设置与热区的设置基本相同。在 Character Limit (字数限制) 文本框中，可以输入某个数值，该数值用来限制用户输入交互字符的个数，比如，在图 29.3 的文本框中输入数字 1，那么当程序运行时，在光标处 1 个字符后，将不能继续输入。

⑥ 选中 Auto-Entry at Limit (自动限制输入) 复选框，在程序运行时，如果输入的字符个数达到限制的字符数，Authorware 将自动结束输入。用户无须再按确认键 (如回车键) 程序就自动往下执行程序。

⑦ 打开 Interaction 选项卡，如图 29.4 所示，用户可以在 Action 文本框中输入文本结束的确认键，回车键 Enter。也可以定义其他的功能键，如 End，当用户在文本区域输入框内输完字符后，按键盘上的 End 键就可以确认。

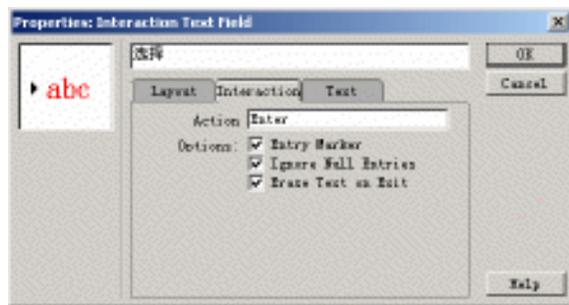


图 29.4 文本区域交互对话框

⑧ 打开 Text 选项卡，其中的选项可用来设置输入文字的字体、字号、字型、颜色及字体模式。在 Font 下拉列表框内可设置输入字符的字体；在 Size 列表框中可选择输入字

符的字号；单击 Style 各选项的复选框可以设置字符的字型；Text 颜色按钮或 Background 颜色按钮将弹出 Color 调色板，单击颜色方块来设置字符颜色或字符背景颜色；在 Mode 下拉列表框中可以选择输入文本对象的重叠模式。设置完毕，还可以根据预览框中显示的示例来决定各个选项，如图 29.5 所示。

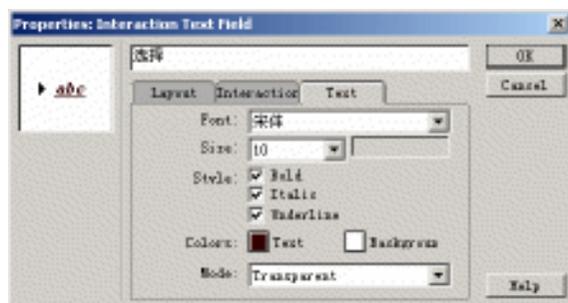


图 29.5 文字设置



技巧

双击文本输入区域也能打开 Properties :Interaction Text Field(文本区域交互属性) 对话框。

⑨ 返回流程线，用鼠标双击交互类型按钮，打开 Properties : Response (交互属性) 对话框，如图 29.6 所示。应该引起注意的是，在 Pattern 文本框内的字符与该交互的分支标题相同，这样 Interaction (交互) 图标就会根据用户输入的字符来判断是否产生交互。

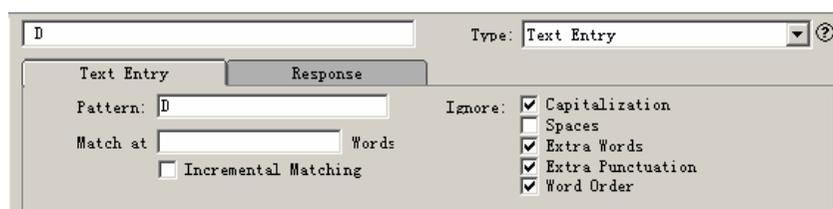


图 29.6 文本输入选项卡

⑩ 在 Pattern 文本框内输入 The word，当程序运行时，在文本输入区内输入此词组程序就会得到交互。如果想使用多组字符来匹配此交互，那么在这些字符之间要加上“|”。例如，在 Pattern 文本框内输入 Boy|Girl，则程序运行时，输入 Boy 或 Girl 都会产生交互。

⑪ 可以在 Match at least (至少匹配几个字符) 文本框内输入数字，该数字决定输入字符与分支标题至少应匹配的字符数目。例如在此处输入数字 2，如果分支标题是 Boy，那么只要输入 bo 或 oy 即可产生交互。

⑫ 单击 Incremental Matching (增加匹配) 复选框，该选项可以达到如下效果：如果分支标题为 A Boy，可以先输入 A，然后按确认键，随后再接着输入 Boy，按下确认键后也能产生交互。



实例

30

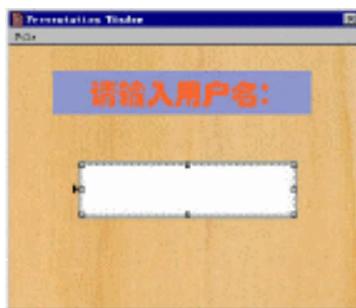
用户登录



实例说明

在很多应用软件中，往往为了安全起见，都要设置一个用户登录对话框，只有拥有正确用户名和密码的用户才能使用该软件，但大众软件编制的用户登录界面形式比较呆板，利用 Authorware 能够设计出具有相同功能，但界面比较吸引人的登录界面。

主要知识点：交互图标、文本输入交互。



操作步骤

- ① 新建文件，保存为“30 用户登录.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框和 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择黄色，如图 30.1 所示。



图 30.1 设置文件属性

③ 从图标栏上拖动一个显示图标，更名为“背景”。双击该显示图标，在打开的展示窗口中使用绘图工具箱上的工具创建登录的背景图案。

④ 从图标栏上拖动一个交互图标到流程线上，更名为“文本输入”。从图标栏上拖动一个计算图标到流程线上，为该交互图标添加第一路交互分支。在弹出的 Response Type 对话框中选择 Text Entry。

⑤ 在流程线上将新添加的交互分支更名为“*”，并且用鼠标单击该交互分支下方的流程走向箭头，从而将该交互分支的 Branch 设置为 Exit Interaction。流程线如图 30.2 所示。

⑥ 在流程线上双击该路交互分支下面的计算图标，在打开的脚本编辑窗口中键入：`username:= EntryText`，如图 30.3 所示。



图 30.2 流程线

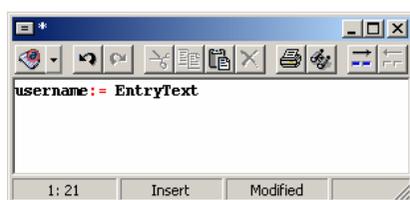


图 30.3 编辑脚本窗口

⑦ 在流程线上按下 Ctrl + R 键运行项目文件，在打开的演示窗口中，用户可以看到文本输入区域和背景上的文本输入区域框没有重合。按下 Ctrl + P 键暂停程序运行，在演示窗口中调整它们的位置，如图 30.4 所示。

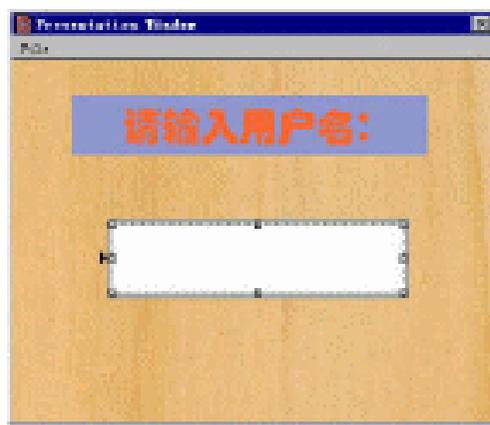


图 30.4 运行图

⑧ 调整完成后，继续运行程序，可以在文本输入提示符后面键入任何文本，当用户按下回车键之后，输入的文本消失。通过上面的操作，用户应该掌握创建文本输入的基本方法。



实例

31

空当接龙



实例说明

本例使用键盘交互来编辑一个纸牌的小游戏。游戏开始后，当用户按下 Continue 按钮，画面上就会提示出牌，按下 1、2、3 等数字键或 J、Q、K 等字母键，对应的纸牌就会弹出来。

主要知识点：交互图标、按键交互。



操作步骤

- ① 新建文件，保存为“31 空当接龙.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框和 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择绿色，如图 31.1 所示。



图 31.1 设置文件属性

- ③ 根据题目要求编辑如图 31.2 所示的流程线。在 Interaction（交互）图标的右侧放置 13 个 Group（群组）图标，分别将它们设置为键盘交互，并命名。

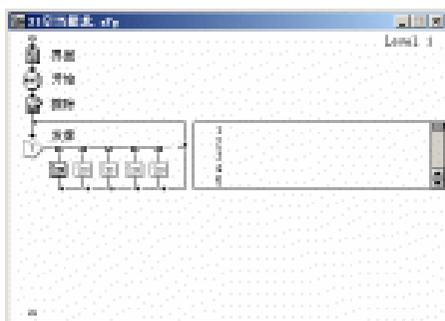


图 31.2 流程线

④ 打开 Group (群组) 图标 , 然后在第二级窗口中放置一个 Display (显示) 图标和一个 Motion (动画) 图标 , Display (显示) 图标用于存放纸牌图片 , Motion (动画) 图标用来控制出牌。

⑤ 双击交互类型按钮 , 打开如图 31.3 所示的交互属性对话框。可以在 Key 文本框中输入控制交互的字母键 (a、b、c、d...)、数字键 (1、2、3、4...)、标点符号和功能键 (Enter、Esc、Insert、Home、End、PageUp、PageDown、F1 ~ F12 等)。



图 31.3 交互属性对话框

⑥ 返回流程线 , 先选择 Display (显示) 图标 , 然后按住 Shift 键双击 Motion (动画) 图标 , 打开 Properties : Motion Icon (动画图标属性) 对话框 , 如图 31.4 所示 , 设置动画方式为点到直线的动画。同样方法设置交互分支中的其他 Group (群组) 图标。



图 31.4 设置动画

⑦ 程序编辑完毕 , 下面再给小游戏添加一个启动画面。返回到前面 , 看一下流程图。在主流程线上还设置了一个 Wait (等待) 图标和一个 Erase (擦除) 图标 , 它们的作用是暂停游戏界面和设置擦除效果。如图 31.5 所示 , 在画面左上角有一个 Continue 按钮 , 此按钮就是等待图标所做的设置 , 通过选择等待图标属性对话框中的 Show Button 复选框可以显示这个按钮。

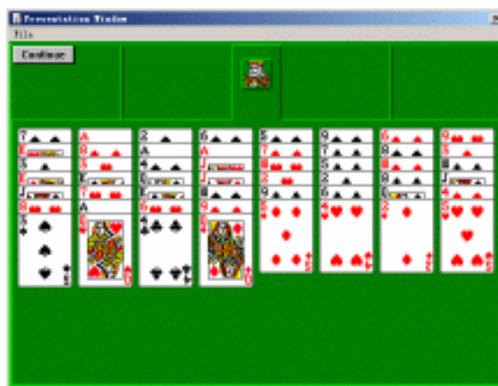


图 31.5 运行开始界面



实例

32

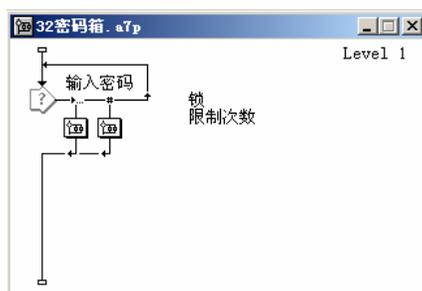
密码箱



实例说明

本例举一个开密码箱的例子,并设置密码的重试次数为3次,如果输入密码的次数超过3次,程序就会弹出一个报警的画面。如果在规定的次数内输入正确的密码,就可以打开密码箱了。流程线如右图所示。

主要知识点:交互图标、文本输入交互、重试限制交互。



操作步骤

- ① 新建文件,保存为“32 密码箱.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,选择 Center On Screen 复选框,取消选择 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择白色,如图 32.1 所示。

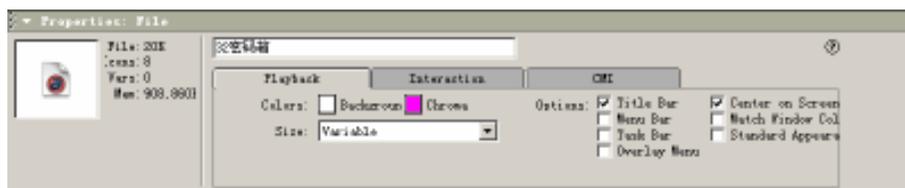


图 32.1 设置文件属性

③ 如图 32.2 所示,在 Interaction (交互) 图标的右面放置一个文本输入交互“锁”,第二条分支设为重试限制交互“限制次数”,更改分支标题为“限制次数”。在“重试限制”群组图标内设置二级流程线,在流程线上使用一个 Display (显示) 图标来设置报警画面。

④ 双击重试限制交互类型按钮,打开 Properties: Response (交互属性) 对话框。如图 32.3 所示,该对话框的设置同样简单。只需要在 Maximum Tries 文本框中输入重试交互的最大次数即可。如果在此文本框中不输入任何值,Authorware 将默认为最大次数是 0 或 1。在程序运行时,如果输入的次数超过此数值,Interaction (交互) 图标就会执行该分支。

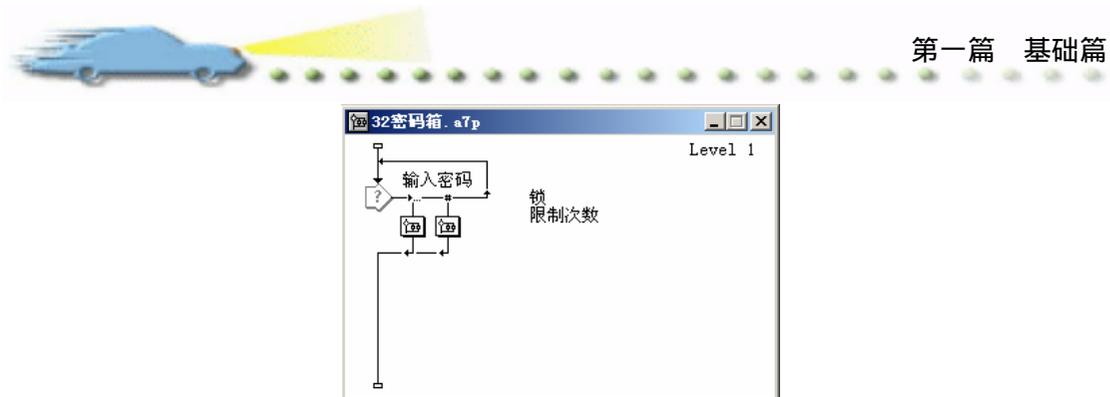


图 32.2 流程线

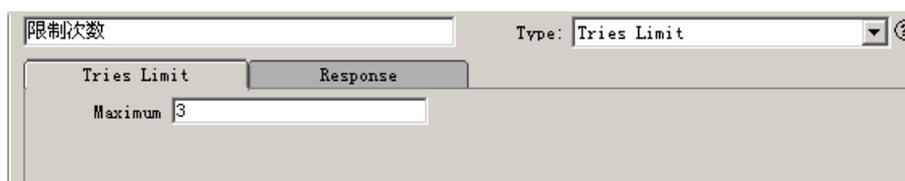


图 32.3 交互属性对话框

⑤ 单击 OK 按钮，返回流程线。双击第一分支中的 Group（群组）图标，在第二级窗口中设置如图 32.4 所示的流程线。在 Display（显示）图标中存储一个打开密码箱后的界面图片。在流程线最后的“秘密”群组图标中就可以放置需要保密的东西了。



图 32.4 流程线

⑥ 返回主流程线，双击第一分支的交互类型按钮，打开文本交互属性对话框。如图 32.5 所示，将所有选择的复选框都取消，也就是取消了 Authorware 默认的忽略选项。这样，取消忽略选项自然就增加了密码的安全性。

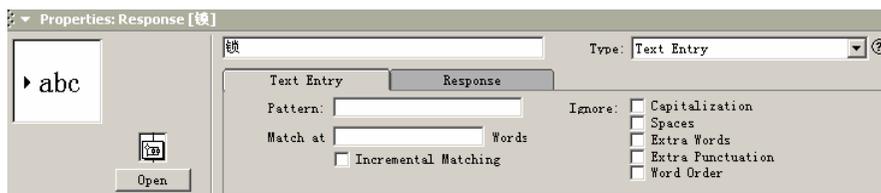


图 32.5 设置交互属性

⑦ 程序设置结束，现在返回程序试运行。如果想把自己制作的多媒体作品加一层保护措施，可以在程序的最前面加上一个与此类似的重试交互。有了密码保护，就再不会担心被别人偷用了。



实例

33

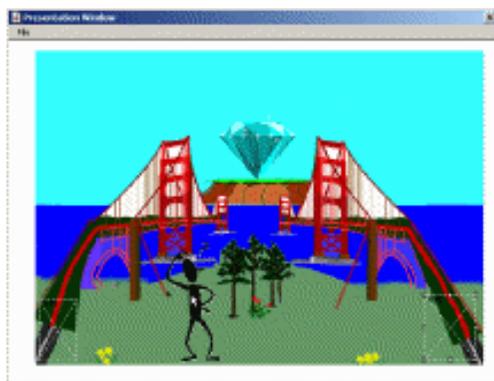
冒险岛



实例说明

本例制作一个“冒险岛”例子，在画面上有一个四面环海的小岛，岛上有一颗硕大的钻石，现在就来决定是走左面的桥还是右面的桥。如果走错，岛上的炸弹就会把画中人炸死；如果走对了，就可以发一笔横财。另外，设定的时间限制是3秒，超过此时间，岛上的炸弹就会自动引爆，将钻石炸飞。

主要知识点：交互图标、目标区域交互、限制时间交互。



操作步骤

- ① 新建文件，保存为“33 密码箱.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 33.1 所示。

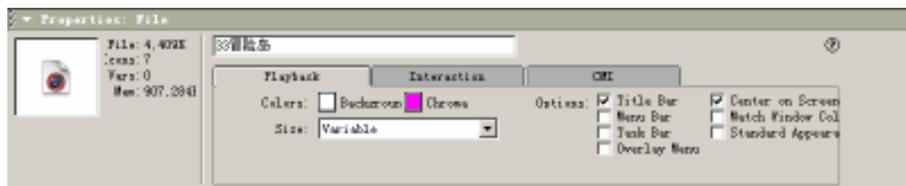


图 33.1 设置文件属性

③ 先来创建程序所要求的流程图，如图 33.2 所示，在“人”显示图标中放置移动的对象。在 Interaction（交互）图标的右侧设置三条分支，前两条分支用于设置目标区域响应，最后的分支设置时间限制交互。当在规定的时间内没有进行移动操作时，Interaction（交互）图标就执行时间限制交互分支。

④ 双击第一分支的交互类型按钮，弹出的对话框如图 33.3 所示，根据对话框的提示，用鼠标将“人”图片移至窗口的左下角，即画面上左面的大桥路口。然后将 On Drop 下拉列表框中的选项设为 Snap to center。用同样的方式设置流程线上的第二条分支。

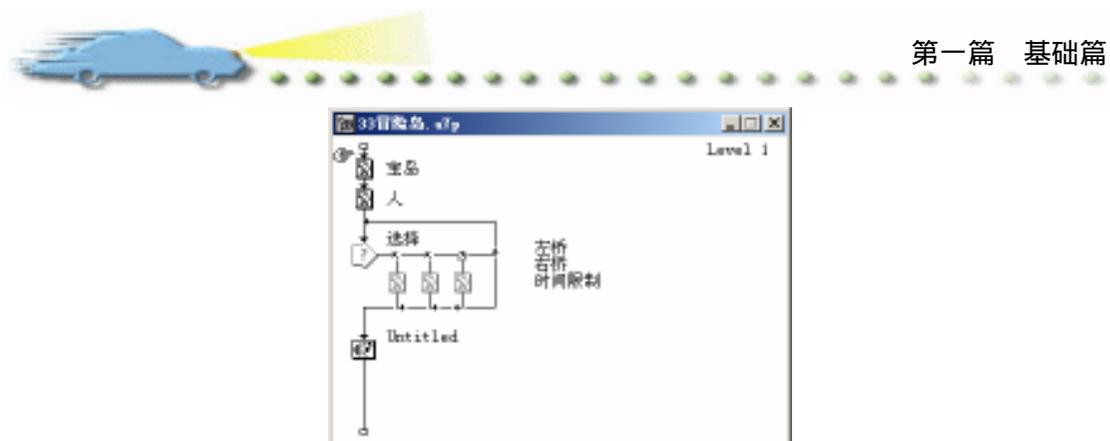


图 33.2 流程图

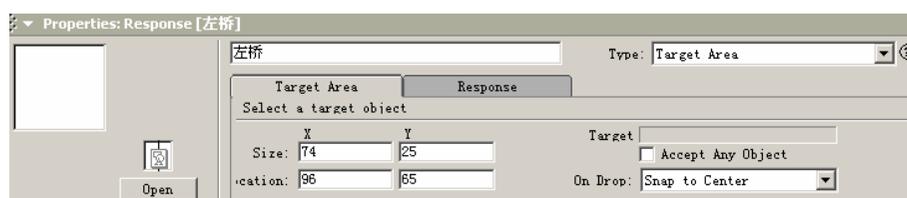


图 33.3 设置目标位置

⑤ 设计完目标区域交互后，返回主流程线。双击时间限制交互类型按钮，弹出的对话框如图 33.4 所示，在 Time Limit（限制时间）文本框中输入交互响应的限制时间为 3。

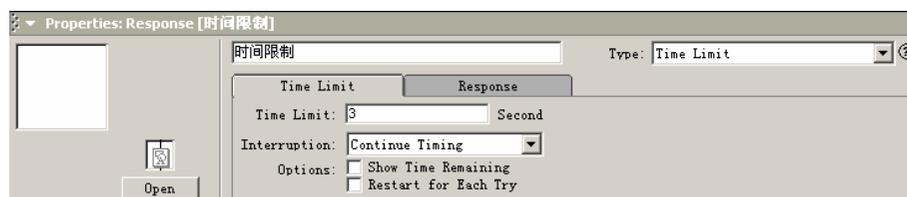


图 33.4 设置时间限制

⑥ 打开 Interruption（中断）下拉列表框，其中设有 4 个选项，它们是用来设置交互执行过程中同时又执行其他内容时的计时方式。

⑦ 最后，在分支中还有 3 个 Display（显示）图标没有设置，在“左桥”图标中可以绘制一幅庆祝发财的画面，在“右桥”图标中绘制一幅炸弹爆炸的画面，然后在“时间限制”显示图标中拷贝同样一幅炸弹爆炸的画面即可。



实例

34

拼图游戏 (1)



实例说明

本例是一个制作“拼图游戏”效果的实例，程序运行后出现 QQ 图片，接着图片开始渐变。接着在演示窗口的左上角出现一个完整的 QQ 图，用户将按照此图进行拼图，完成后会出现祝贺文字。

主要知识点：交互图标、擦除图标。



操作步骤

- ① 新建文件，保存为“34 拼图游戏.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Center On Screen 复选框和 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择粉红色，如图 34.1 所示。



图 34.1 设置文件属性

- ③ 设计程序流程线。从面板中拖放一个 Calculation (计算) 图标到流程线上，并将 Calculation (计算) 图标命名为“初始化窗口”，然后拖放一个显示图标到流程线上，命名为“背景”，接着拖放一个 Group(群组)图标，命名为“组图标”。然后再拖放一个 Interaction (交互) 图标到流程线上，命名为 Interaction (交互)，再拖放 5 个组图片到交互图标的右下方，分别命名为“右上”、“左下”、“右下”、“错误位置”和 AllCorrectMatched。主流程线如图 34.2 所示。

- ④ 设置“初始化窗口”图标内容。双击 Calculation (计算) 图标，在打开的代码窗口中输入 ResizeWindow(480,360)，如图 34.3 所示。

- ⑤ 设置 QQ 图标的内容，将 QQ 图标导入到显示图标。调整其在演示窗口中的位置并调整其大小和颜色，并设置为透明模式，如图 34.4 所示。

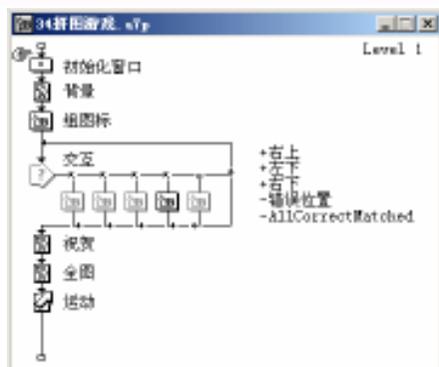


图 34.2 流程线

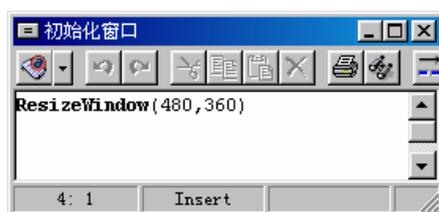


图 34.3 代码窗口



图 34.4 设置背景图标

⑥ 设置二级程序流程线，双击流程线上的“组图标”打开二级设计流程线。从面板中拖放一个显示图标到流程线上，命名为“全图”，然后拖放一个 Wait（等待）图标到流程线上，命名为“暂停 2 秒”，接着拖放一个 Erase（擦除）图标到流程线上，命名为“擦除全图”，最后拖放 4 个显示图标到流程线上，命名为“QQ1”、“QQ2”、“QQ3”、“QQ4”，设计完成的流程线如图 34.5 所示。



图 34.5 二级流程线



实例

35

拼图游戏 (2)



实例说明

本例接着制作“拼图游戏”效果的实例，程序运行后出现 QQ 图片，接着图片开始渐变。接着在演示窗口的左上角出现一个完整的 QQ 图，用户将按照此图进行拼图，完成后会出现祝贺文字。

主要知识点：动画图标。



操作步骤

- ① 确认打开的文件为“35 拼图游戏”(由实例 34 的源文件重命名而得)。
- ② 设置“祝贺”图标内容。双击流程线上的“祝贺”图标，打开演示窗口，用文本工具在图中输入“祝贺你！”，将字体设为隶书，颜色设为黄色，调整其在窗口的位置。
- ③ 设置“全图”图标的内容。双击显示图标打开演示窗口，执行 File 文件中的 Import 命令，选择一副 QQ 图片，并用鼠标拖动图片四周的控制点调整图片大小以及在演示窗口中的位置。
- ④ 设置 Motion (动画) 图标的属性。双击“全图”图标，然后双击 Motion (动画) 图标，打开 Properties 对话框，单击 Motion 选项卡，在 Type 下拉列表中选择 Point to Point，单击 QQ 图片为移动对象，然后安装鼠标移动 QQ 创建一个黑色三角形控制点，不断移动松开鼠标创建一系列的路径控制点，用鼠标双击控制点使路径变得平滑。在 Timing 下拉列表中选择 Time(Sec)，在文本框中输入 3，在 Concurrency 下拉列表中选择 Concurrent，在 Move When: 文本框中输入 1，对话框设置如图 35.1 所示。



图 35.1 设置运动图标属性

- ⑤ 设置“暂停 2 秒”图标。双击二级流程线上的“等待 2 秒”图标，打开属性对话框，其设置如图 35.2 所示。



图 35.2 设置等待图标

⑥ 设计“擦除全图”图标。双击二级流程线上的“全图”图标，使 QQ 图标显示在演示窗口，双击“擦除全图”图标，打开属性对话框，用鼠标单击演示窗口中的 QQ 图片，单击对话框下面的 List 选项卡，选中 Icons to Erase，这时在下面方框中出现了“全图”图标，表示擦除图标，擦除的是“全图”图标里面的内容，对话框设置如图 35.3 所示。

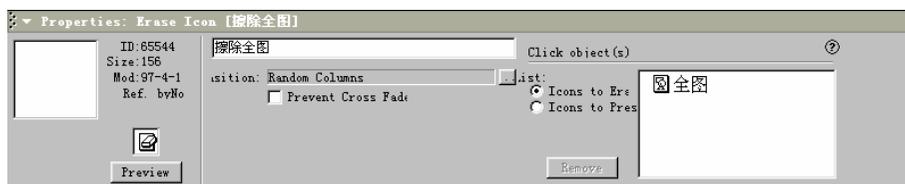


图 35.3 设置擦除图标

⑦ 设置 QQ 图片的各个部分。双击流程线上的“背景”图标，使背景图显示在演示窗口中，然后按住 Shift 键的同时双击二级流程线上的 QQ1 图标，打开演示窗口，执行“文件”菜单上的 Import 命令，选择一副 QQ 左上部分的图片，并用鼠标拖动图片四周的控制点，调整图片的大小，以及其在演示窗口中的位置。用同样的方法设置 QQ 其他部分的图片，如图 35.4 所示。



图 35.4 设置 QQ 各部分

⑧ 设置“右上”组图标的属性。按住 Shift 键的同时双击流程线上的“右上”组图标上方的控制按钮，打开属性对话框。在 Type 下拉列表中选择 Target Area，用户可以看到演示窗口中出现一个标有“右上”的矩形虚线框，用鼠标单击窗口中 QQ 的右上部分，可以看到虚线框自动附在图片上，然后用鼠标将 QQ 右上部分图片拖动到 QQ 左上部分的右边。在 On Drop 下拉列表选取 Snap to Center，如图 35.5 所示。

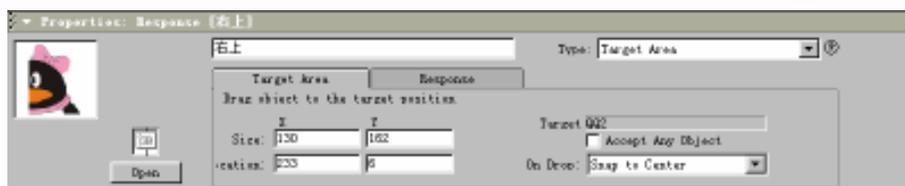


图 35.5 设置“右上”组图标属性

⑨ 设置目标区域响应属性。在刚才的对话框中单击 Response 选项卡，如图 35.6 所示。

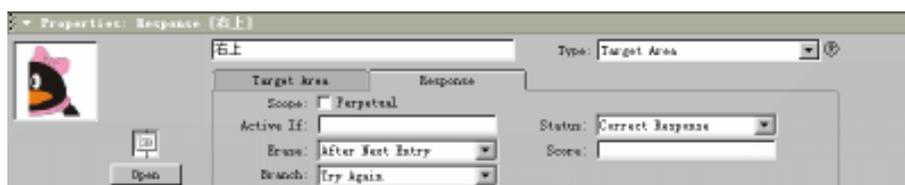


图 35.6 设置响应属性

⑩ 依照 7~9 设置“左下”、“右下”组图标，并设置目标区域响应属性。

⑪ 设置“错误位置”组图片的属性。按住 Shift 键的同时双击流程线上的“错误位置”组图标上方的控制按钮，打开属性对话框。在 Type 下拉列表中选择 Target Area，用户可以看到演示窗口中出现一个标有“错误位置”的矩形虚线框，调整虚线框的大小，在 On Drop 下拉列表中选择 Put Back，选取允许任何对象，如图 35.7 所示。

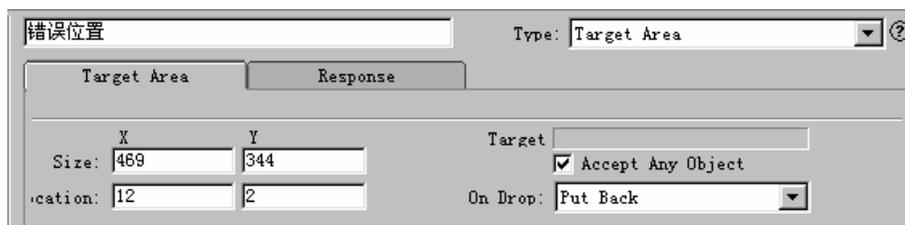


图 35.7 设置“错误位置”组图标属性

⑫ 接着设置目标区域响应属性。单击 Response 选项卡，设置如图 35.8 所示的选项。



图 35.8 设置响应属性

⑬ 设置 AllCorrectMatched 组图标的属性。双击流程线上的 AllCorrectMatched 组图标上方的控制按钮，打开属性对话框。在 Type 下拉列表中选择 Conditional，在 Conditional 选项卡中的 Conditional: 文本框中输入 AllCorrectMatched，在“自动”下拉列表中选择 True，如图 35.9 所示。单击 Response 选项卡进行设置，结果如图 35.10 所示。

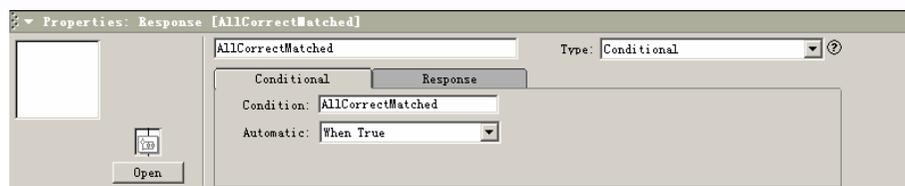


图 35.9 设置 Conditional 选项卡

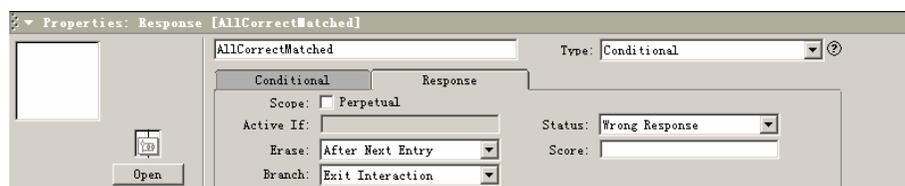


图 35.10 设置 Response 选项卡

- 14 预览效果。单击工具栏中的 Restart  按钮，便可运行此实例。



实例

36

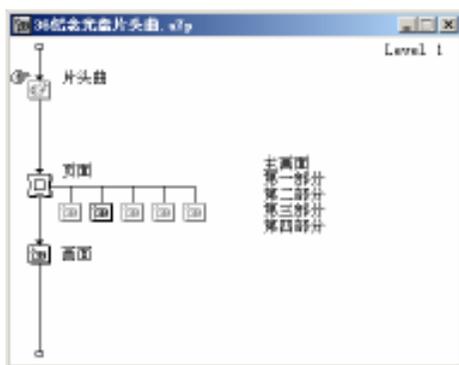
纪念光盘片头曲



实例说明

本例设计纪念光盘片头曲,当播放片头曲时,显示窗口连续放映图片。在流程线上添加 Sound (声音) 图标来增加作品的演示效果。流程线如右图所示。

主要知识点: 框架图标。



操作步骤

- ① 新建文件, 保存为“36 纪念光盘片头曲.a7p”。
- ② 执行命令 Modify File Properties, 打开属性对话框, 设置 Size 为 Variable, 选择 Center On Screen 复选框, 取消选择 Menu Bar 复选框, 单击 Color 右边的 Background Color 按钮, 打开 Color 对话框, 从中选择白色, 如图 36.1 所示。

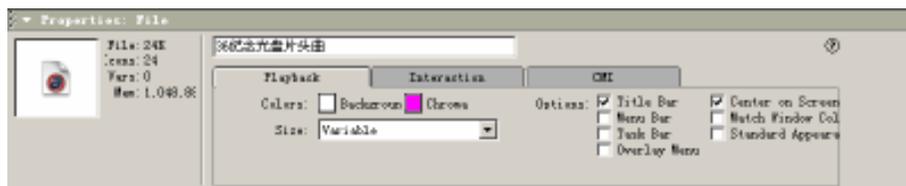


图 36.1 设置文件属性

③ 在流程线的最前面放置一个“片头曲”声音图标, 在这个图标中用户可以插入自己喜爱的音乐, 程序一开始就播放此音乐。另外, 还要在程序内部的每一个部分添加一个 Sound (声音) 图标, 这样可以根据不同的作品内容而设置不同的音乐效果。

④ 插入片头曲音乐后, 双击该 Sound (声音) 图标, 打开 Properties : Sound Icon (声音图标属性) 对话框, 如图 36.2 所示。想这样设置片头音乐, 当程序运行时, 音乐就开始播放, 当单击画面上的超文本进入某个部分时, 音乐便嘎然而止, 否则音乐将一直播放。因此, 先要设置 Concurrency 下拉列表框中的 Perpetual (永久) 选项, 然后选择 Play 下拉列表框中的 Until True 选项, 最后在下面的文本框中输入音乐停止条件 MouseDown。MouseDown 是一个系统变量, 在程序运行时单击鼠标, MouseDown 的值即为真, 音乐也就随之停止。



图 36.2 设置播放条件

⑤ 下面再来设置一下作品里面部分的伴随音乐。首先要在流程线上的各个“伴随音乐”声音图标中插入相应的音乐，然后双击该 Sound（声音）图标，打开如图 36.3 所示的 Properties :Sound Icon(声音图标属性)对话框。在 Concurrency 下拉列表框中选择 Perpetual（伴随）选项，然后在下面的条件文本框中输入 MouseDown，即单击画面上的定向按钮（前一页按钮、下一页按钮等）时，当前音乐停止播放。

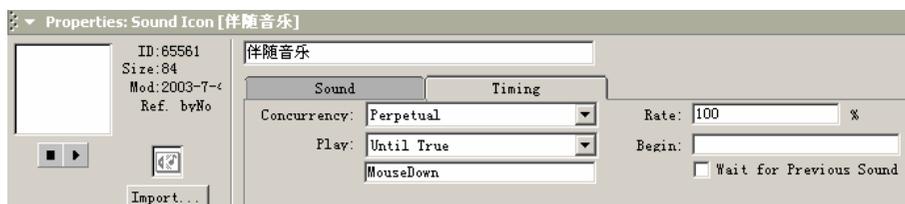


图 36.3 设置伴随音乐



实例

37

滚动字幕



实例说明

本例是一个制作“滚动字幕”效果的实例，运行后随着音乐的响起屏幕上出现滚动字幕。

主要知识点：显示图标、动画图标、等待图标。



操作步骤

- ① 新建文件，保存为“37 滚动字幕.a7p”。
- ② 执行命令 **Modify** **File** **Properties**，打开属性设置窗口，设置 **Size** 为 **Variable**，选择 **Center On Screen** 复选框，取消选择 **Menu Bar** 复选框，如图 37.1 所示。

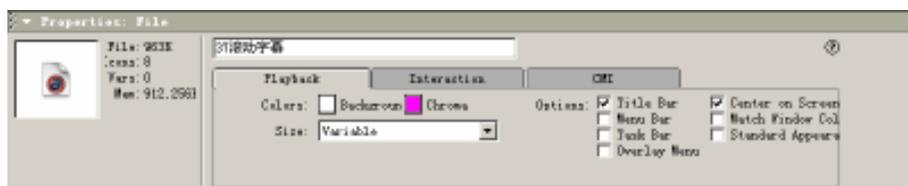


图 37.1 设置文件属性

- ③ 在流程线上添加一个 **Calculation**（计算）图标，命名为“初始化窗口”，然后再添加一个 **Display**（显示）图标和一个 **Sound**（声音）图标，分别命名为“背景”和“音乐”，再拖放一个 **Decision**（判断）图标到流程线上，命名为“循环”，最后再拖放一个 **Group**（群组）图标到 **Decision**（判断）图标的右下方，命名为“滚动文字”，如图 37.2 所示。

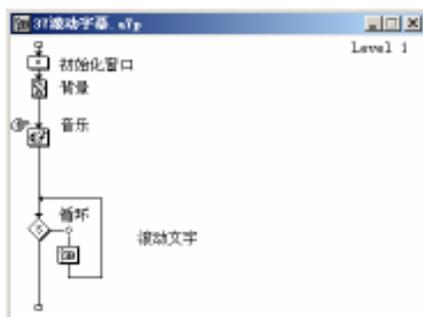


图 37.2 流程线

④ 设置“初始化窗口”图标内容。双击 Calculation (计算) 图标, 在打开的代码窗口中输入 Resizewindow(480, 360), 如图 37.3 所示, 其作用是初始化演示窗口的大小。

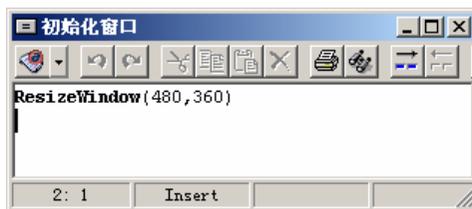


图 37.3 代码窗口

⑤ 设置“背景”图标的内容。用鼠标双击流程线上的“背景”显示图标, 打开演示窗口, 执行 File Import and Export Import Media 命令, 导入一幅图片, 调整图片的大小, 让其充满整个窗口, 如图 37.4 所示。

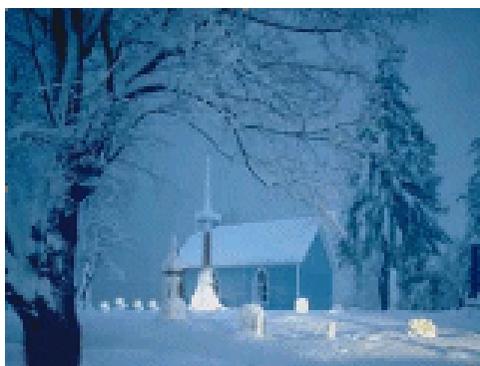


图 37.4 设置背景图标

⑥ 设置 Sound (声音) 图标。单击流程线上的“音乐”图标, 打开属性对话框, 单击对话框左下角的 Import (导入) 按钮, 输入一个声音文件, 如图 37.5 所示。单击“确定”按钮返回对话框。单击 Timing 选项卡, 对话框设置如图 37.6 所示。

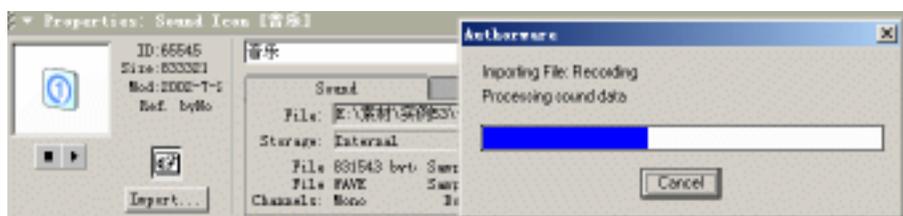


图 37.5 导入音乐

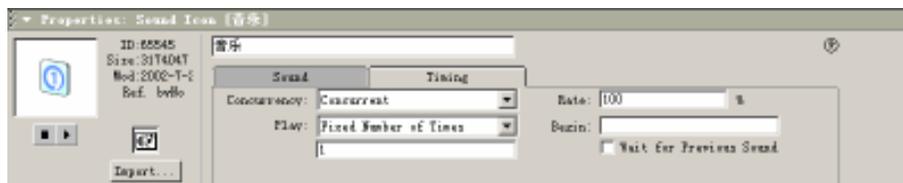


图 37.6 设置“音乐”图标属性



⑦ 设置“循环”图标的内容。单击流程线上的“循环”判断图标，打开属性对话框，在 Repeat:下拉列表框中选择 Until Click/Keypress，在 Branch 下拉列表框中选择 Sequentially，属性设置如图 37.7 所示。



图 37.7 设置“循环”图标属性

⑧ 双击流程线上的“滚动文字”群组图标，打开二级设计流程线窗口，向流程线添加一个 Display (显示) 图标，命名为“文本”，然后再添加一个 Wait (等待) 图标和一个 Motion (动画) 图标，分别命名为“等待”和“移动文本”。如图 37.8 所示。

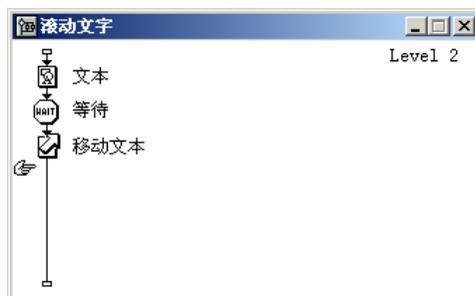


图 37.8 二级流程线

⑨ 设置“文本”图标的内容。双击图 37.8 中“文本”显示图标打开演示窗口，用绘图工具箱中的文本工具输入要显示的文字。设置文字的大小及颜色，设置文字显示模式为透明模式。

⑩ 设置“等待”图标的内容。单击“等待”图标，打开属性对话框，属性设置如图 37.9 所示。

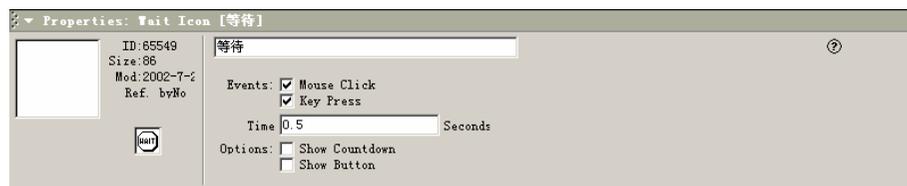


图 37.9 设置“等待”图标属性

⑪ 设置“移动文本”图标的属性。单击“移动文本”图标打开属性对话框，在 Type 下拉列表框中选择 Direct to Point，用鼠标将“文本”由上往下移动到窗口上方。在 Timing 下拉列表框中选择 Time(sec)，在下面的文本框中输入“15”，在 Concurrency 下拉列表框中选择 Wait Until Done，属性设置如图 37.10 所示。



图 37.10 设置“移动文本”图标属性

⑫ 存盘，运行此实例。



实例

38

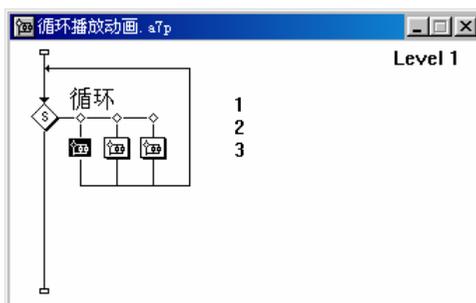
动画播放



实例说明

举一个制作“动画播放”效果的实例，介绍 Digital Movie（电影）和 Decision（判断）图标的使用方法，程序运行后，实现动画的循环播放，流程线如右图所示。

主要知识点：判断图标。



操作步骤

- ① 新建文件，保存为“38 播放动画.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框。取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框。从中选择白色，如图 38.1 所示。



图 38.1 设置文件属性

- ③ 设置流程线。从面板中拖放一个 Decision（判断）图标到流程线上，命名为“循环”。
- ④ 双击“循环”设计图标。打开属性对话框，在 Repeat 下拉列表中选择 Until True，在下面的文本框中输入 FALSE，在 Branch 下拉列表中选择 Sequentially。属性设置如图 38.2 所示。

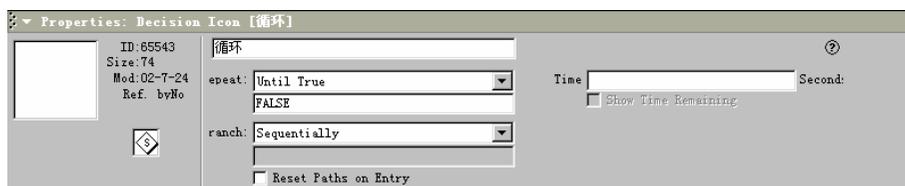


图 38.2 设置 Decision 图标属性

⑤ 从面板中拖放一个 Group (群组) 图标到流程线上，为“循环”判断图标设置 3 个分支，分别命名为“1”、“2”、“3”。

⑥ 双击 1 组图标，打开子程序设置窗口，从面板中拖放一个 Digital Movie (电影) 图标到流程线上，命名为“电影 1”，如图 38.3 所示。



图 38.3 设置子程序窗口

⑦ 双击“电影 1”数字电影图标，打开属性对话框，单击 Import 按钮，选择需要导入的数字电影。单击 Timing 按钮，进入 Timing 选项卡，在 Concurrency 下拉列表中选择 Wait Until Done，如图 38.4 所示。

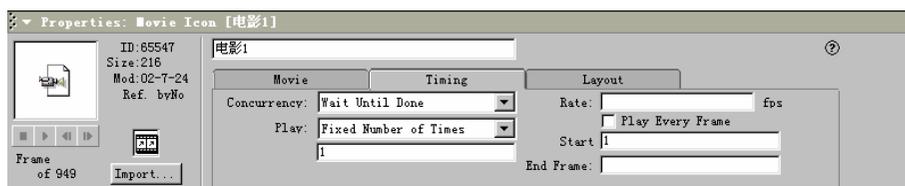


图 38.4 设置“电影 1”数字电影图标属性

⑧ 重复步骤 6~7 分别创建并设置“电影 2”，“电影 3”数字电影图标的属性。



实例

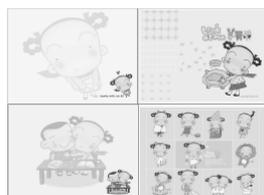
39

图片演示



实例说明

本例是一个制作“图片演示”的实例，如右图所示，程序运行时在演示窗口依次以不同的方式出现下面 4 幅图片。



主要知识点：显示图标。



操作步骤

- ① 新建文件，保存为“39 图片演示.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择黄色，如图 39.1 所示。

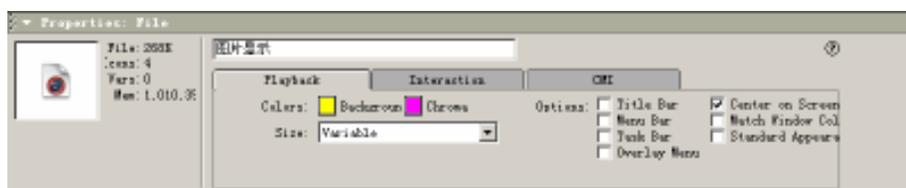


图 39.1 设置文件属性

- ③ 设计程序流程线。从面板中拖放一个显示图标到流程线上，将图标命名为“图片 1”。用同样的方法拖放其他 3 个显示图标到流程线上，设计流程线如图 39.2 所示。



图 39.2 流程线

- ④ 设置“图片 1”图标的内容。双击“图片 1”图标打开演示窗口，执行 File 菜单

中的 Import 命令，输入一幅图片，按住 Shift 键的同时用鼠标拖动图片，调整其大小和位置，这时会弹出一个“提示”对话框，询问用户是否调整改变图片的大小，单击 OK 按钮。用鼠标拖动控制点调节图片大小，使图片充满整个演示窗口，如图 39.3 所示。



图 39.3 输入图片 1

⑤ 用与上步同样的方式设置其他 3 张图片的图标内容。

⑥ 设置“图片 1”的过渡效果。用鼠标单击选取流程线上的“图片 1”图标，执行 Modify|Icons|Properties 命令，打开属性对话框，在 Categories 列表中，单击 Internal，在右边的 Transition 列表中选择 Spiral。在 Duration (周期) 文本框中输入“2”，设置完成过渡效果所需的时间是 2 秒。在 Smoothness 文本框中输入“73”，设置过渡效果的平滑度，数字越大过渡效果越粗糙。在 Effects 内选取 Changing Area(s)选项，表示过渡效果作用的范围是显示对象所占据的区域。用户若想预览所选的过渡效果，可以单击对话框中的 OK 按钮，如图 39.4 左图所示。

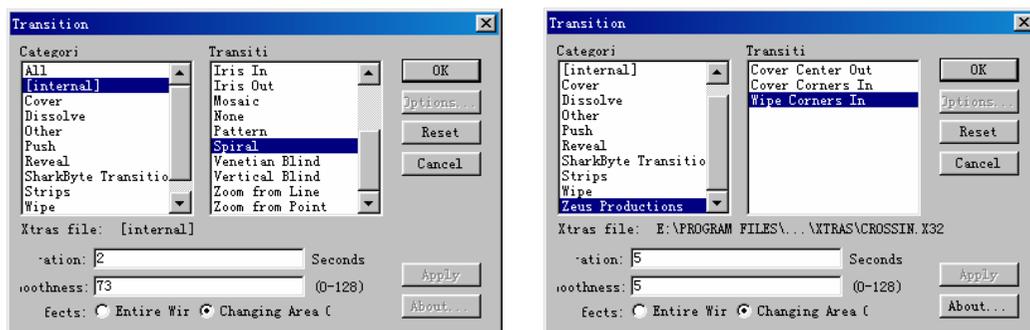


图 39.4 设置过渡效果

⑦ 设置“图片 2”的过渡效果。依照和步骤 6 同样的方法，在过渡效果类别列表中选择 Zeus Productions，在右边的过渡效果列表中选择 Wipe Corners In。Duration (周期) 设为 5，平滑度设为 5，如图 39.4 的右图所示。

⑧ 同样设置“图片 3”和“图片 4”。“图片 3”的过渡效果是 Wipe 中的 Edge in, Horizontal，周期和平滑度分别是“2”和“4”；“图片 4”的过渡效果是 Reveal 中的 Reveal Down，周期和平滑度分别是“8”和“4”。



实例

40

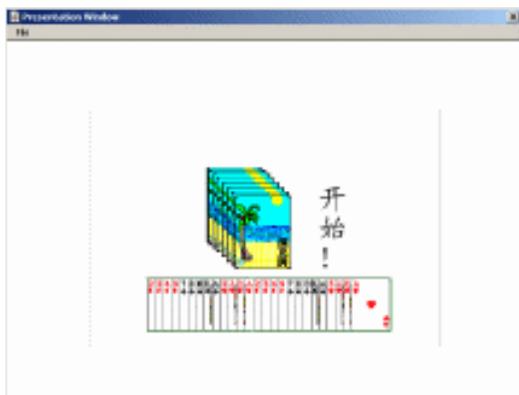
纸牌游戏



实例说明

前面已经举过一个空当接龙的例子，是使用按钮响应的方式制作的。下面为此程序设置一个启动画面，即当游戏刚进入时，纸牌自动从左到右排列显示。如果你玩过空当接龙的游戏，对此画面可能再熟悉不过了。

主要知识点：判断图标、擦除图标。



操作步骤

- ① 新建文件，保存为“40 纸牌游戏.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框和 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 40.1 所示。

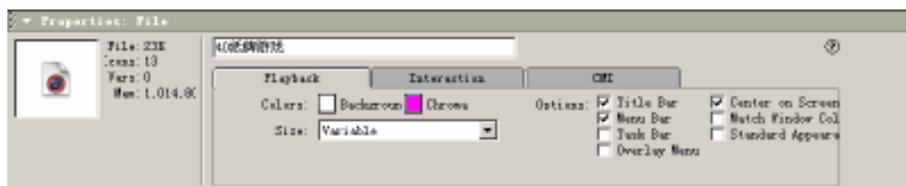


图 40.1 设置文件属性

- ③ 设计如图 40.2 所示的流程线。

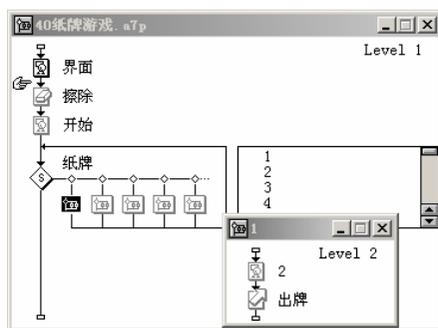


图 40.2 流程线

④ 双击 Decision (判断) 图标，打开 Properties : DecisionIcon (判断图标属性) 对话框。如图 40.3 所示，选择 Repeat 下拉列表框中的 Until Click/Keypress 选项则只有在单击鼠标或按下键盘后，分支路径的执行才会停止。选择 Branch 下拉列表框中的 Sequentially 选项 Decision (判断) 图标将从左至右执行分支路径。

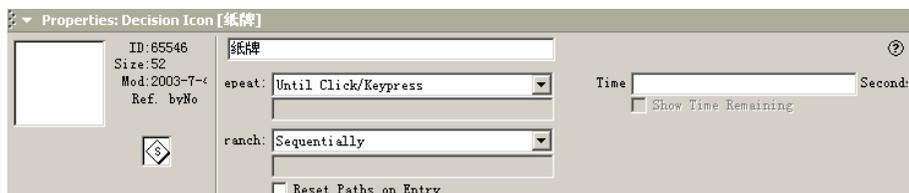


图 40.3 判断图标属性对话框

⑤ 设置完 Properties : DecisionIcon(判断图标属性)对话框后，单击 OK 按钮返回流程线。双击分支上的响应按钮，打开 Properties : DecisionPath(判断路径属性)对话框，如图 40.4 所示，在 Erase 下拉列表框中选择 Don't Erase 选项，这样，当纸牌弹出后且在执行下一分支前，纸牌将不会消失。

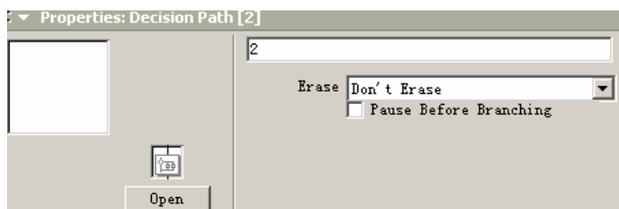


图 40.4 设置动画过程

⑥ 进入“2”交互图标，然后双击 Motion(动画)图标，打开 Properties : MotionIcon(动画图标属性)对话框，用鼠标拖动纸牌移到适当位置，然后单击 OK 按钮。另外，还要在 Motion 选项卡中设置纸牌运动的时间，比如 1 秒。

⑦ 用同样的方法分别设置另外几个群组图标，应注意的是要把纸牌移动的最后位置对齐。



实例

41

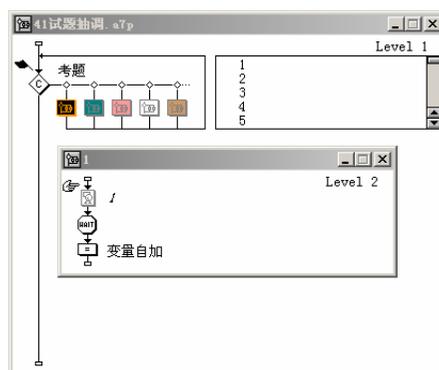
试题抽调 (1)



实例说明

制作一个试题抽调的例子,假设题库里有 10 道题,如果考卷要抽调 5 道选择题,并且在一份考卷内不能有重复的试题,因此就必须使用 Decision (判断) 图标的第 2 种分支执行方式。流程图如右图所示。

主要知识点: 判断图标。



操作步骤

- ① 新建文件,保存为“41 试题抽调.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,选择 Center On Screen 复选框,取消选择 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择白色,如图 41.1 所示。

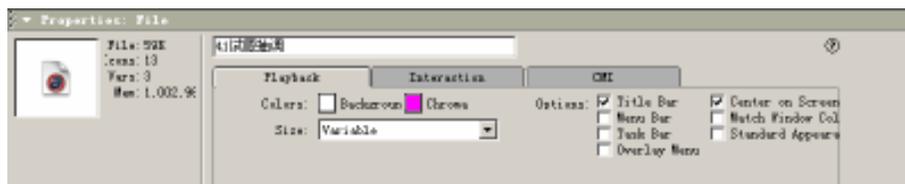


图 41.1 设置文件属性

③ 如图 41.2 所示,可以根据题目的要求编写流程线,由于题库共有 10 道题,因此使用 10 个 Group (群组) 图标构成 10 个分支来提供选择,在每个 Group (群组) 图标内,可以使用相同的流程结构,如 Group (群组) 图标 1 中所示,我们可以在“Display”(显示)图标中存放试题。使用 Wait (等待) 图标可以使画面暂停,以便于观察试题。在下面还要放置一个 Calculation (计算) 图标,用它来控制选择试题的数目,后面再对其具体功能进行介绍。

④ 双击 Decision (判断) 图标,打开 Properties: DecisionIcon (判断图标属性) 对话框,如图 41.3 所示,在 Repeat 下拉列表框中选择 Until True 选项,然后在下面的文本框中输入判断结束的条件 $Ti=5$ 。然后在 Branch 下拉列表框中选择 Sequentially 选项。如果返回到流程线,用户会看到 Decision (判断) 图标的标记变为。在此例中,使用变量 Ti 来控制

选择考题的数目，也就是说，当变量 Ti 的值是 5 的时候，Decision（判断）将退出，不再执行分支路径中的内容。

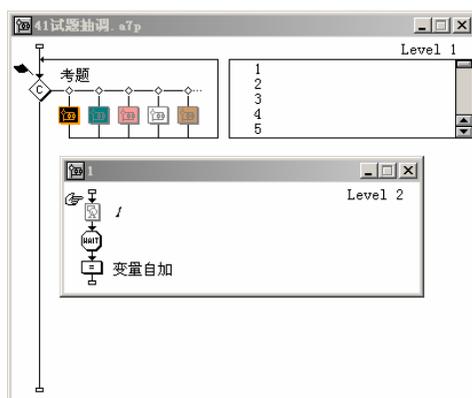


图 41.2 流程线

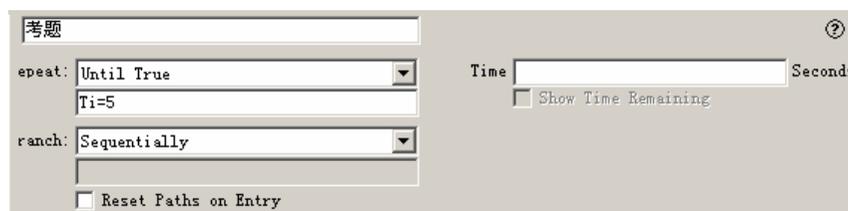


图 41.3 设置重复次数

⑤ 当关闭该对话框时，Authorware 会弹出一个 New Variable（新变量）窗口，要求用户对该变量进行初始化，如图 41.4 所示，在 Initial Value 文本框中输入数值 0，然后单击 OK 按钮。

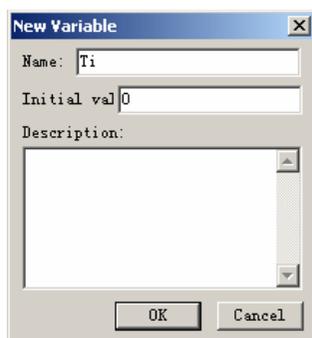


图 41.4 变量初始化

⑥ 打开 Calculation（计算）图标文本框，如图 41.5 所示，在文本框中输入 $Ti:=Ti+1$ ，如果用户学过某种编程语言，对此表达式可能不会陌生，实际上该表达式就是一个计数器，每当执行该表达式时， Ti 的数值就会增加 1，当它的值增加到 5 时，Decision（判断）图标就会退出。



图 41.5 设置计数器

⑦ 在 Group (群组) 图标中我们还使用了一个 Wait (等待) 图标，这个 Wait (等待) 图标不是用来设置程序的，主要用它来调试程序。在程序运行时，当 Authorware 每次进入一条分支路径时，都会执行 Wait (等待) 图标，因此可以打开 Properties : WaitIcon (等待按钮属性) 对话框，单击 Events 的 Mouse Click 复选框和 Options 下的 ShowButton 复选框。在程序运行时，每当抽调一份试题，屏幕上就会出现一个等待按钮，单击该按钮，Decision (判断) 图标就开始执行下一条分支，当执行第 5 条分支后，程序将跳出 Decision (判断) 图标，试题也抽调结束。

实例

42

试题抽调 (2)



实例说明

前面讲述过如何制作“试题抽调”的例子。本例使用判断分支来设计。程序流程线如右图所示。

主要知识点：判断图标。



操作步骤

① 确认当前打开的是“42 试题抽调”(由实例 41 的文件重命名而得)。

② 如图 42.1 所示,如果变量或表达式的值等于 1,例如图中的 $Number=1$,则 Authorware 就会执行第 1 条分支路径;如果变量式表达式的值等于 5,即 $Number=5$,则 Authorware 执行第 2 条分支路径;其余情况依次类推。

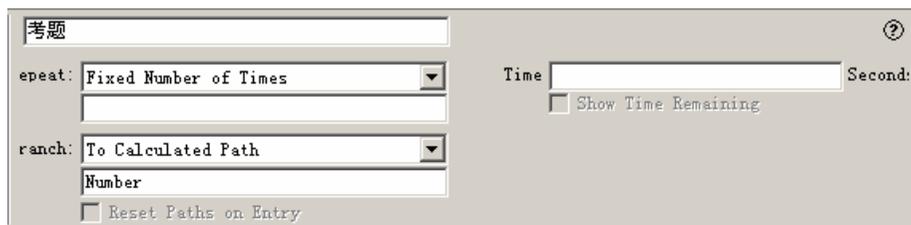
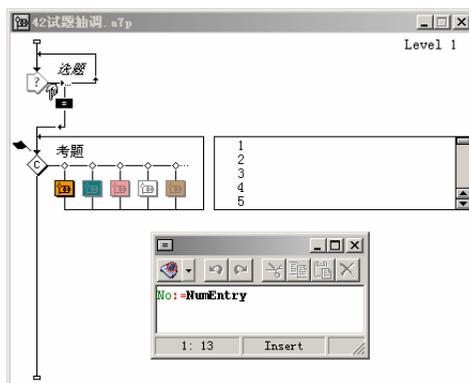


图 42.1 设置变量

③ 如图 42.2 所示,在流程线上,添加一个 Interaction (交互) 图标,同时使用文本输入响应来控制题目的调用,比如,在程序运行时的文本输入框内输入 1,程序就会调用第一道试题;输入 2,程序就会调用第二道试题。

④ 如图 42.2 中的 Calculation (计算) 图标所示,使用变量 No 来接受从键盘上输入的数值,然后将该变量送到下面的 Decision (判断) 图标中去。

⑤ 打开 Decision (判断) 图标,在 Repeat 下拉列表框中选择 Until True 选项,然后在下面的文本框中输入重复条件 $Ti \geq 5$,即当试题抽调 5 套时跳出 Decision (判断) 图标。在 Branch 下拉列表框中选择 To Calculated Path 选项,然后在下面的文本框输入控制变量 No,如图 42.3 所示。

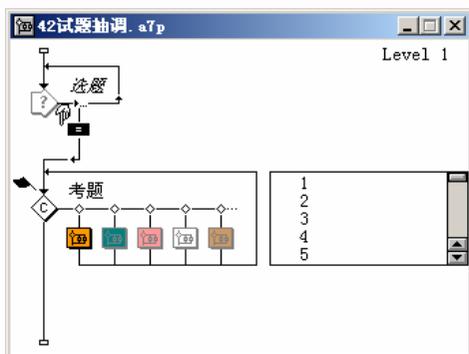


图 42.2 流程线

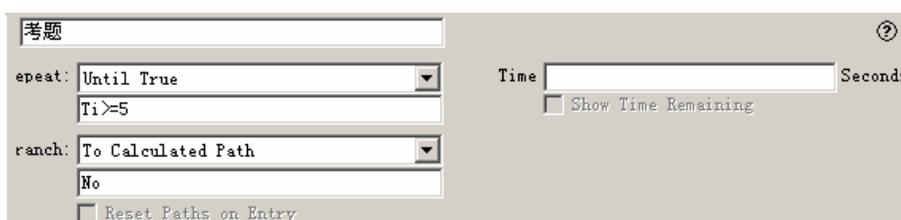


图 42.3 设置控制变量

⑥ 在输入一次数值后，程序往往不能继续要求进行输入，因此，还要使用前面曾经提到过的跳转返回函数 GoTo。如图 42.4 所示，打开 1 群组图标中的 Calculation(计算)图标，在文本框中再加上一条语句 GoTo(IconID@"选题")，这样，在每次抽调完一次试题后，Authorware 将由 GoTo 函数跳转到程序的最前面“试题”图标处等待继续输入。同样道理，在 Decision(判断)图标每一个分支中的 Calculation(计算)图标中都要输入此语句。



图 42.4 设置程序跳转



如果设置的变量或表达式的值为一个小数，则 Authorware 只取整数部分的值。例如，当设置的变量或表达式的值为 4.35 或 4.65 时，Authorware 会执行第 4 条分支路径。

实例

43

图片欣赏



实例说明

本例是一个制作图片欣赏的实例,运行后演示窗口中出现一幅图片,单击下面的控制按钮,可以进行前进和后退的翻页等操作,如右图所示。

主要知识点: 框架图标。



操作步骤

- ① 新建文件,保存为“43 图片欣赏.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,选择 Center On Screen 复选框,取消选择 Menu Bar 复选框,单击 Color 右边的 Background Color 按钮,打开 Color 对话框,从中选择浅蓝色,如图 43.1 所示。

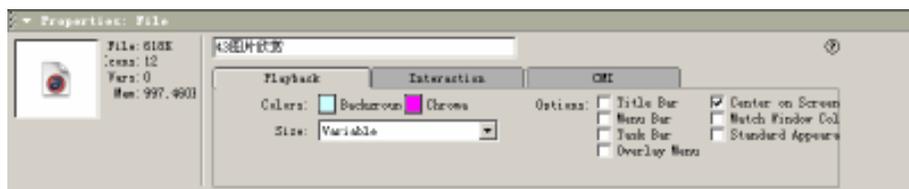


图 43.1 设置文件属性

- ③ 设计程序流程线。从设计图标面板中拖放一个 Calculation (计算) 图标到流程线上,并将 Calculation (计算) 图标命名为初始化窗口,然后拖放一个 Framework (框架) 图标到流程线上,命名为“图片框架”,然后拖放 4 个 Display (显示) 图标到 Framework (框架) 图标的右方,分别命名为“图片 1”、“图片 2”、“图片 3”、“图片 4”,如图 43.2 所示。

④ 设置“初始化窗口”图标内容。双击 Calculation (计算) 图标,在打开的代码窗口中输入 Resizewindow(480,360),如图 43.3 所示。其作用是初始化演示窗口的大小。

⑤ 设置“图片框架”图标的内容。双击流程线上的“图片框架”,弹出框架流程线设计图,将显示图标“灰色导航面板”删除。将交互图标命名为“按钮”,然后分别将其中的“返回”、“最近页”和“查找”按钮删除,将“退出框架”改名为“退出”,框架流程线如图 43.4 所示。

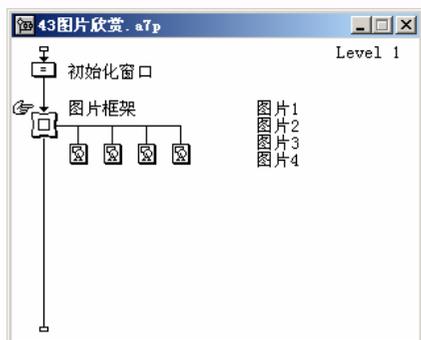


图 43.2 流程线



图 43.3 代码窗口

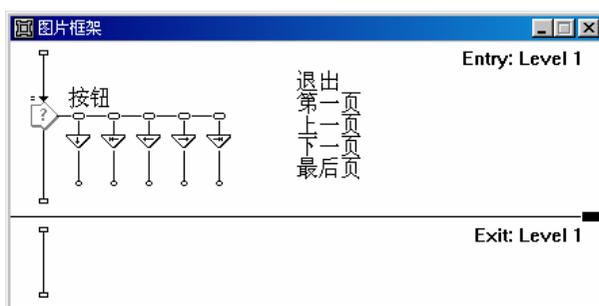


图 43.4 设置图片框架流程线

⑥ 设置“按钮”的外观形状。在打开的“图片框架”窗口中，单击“退出”图标上的控制标志，打开 Properties 对话框。单击对话框左上角的 Buttons 控制标志，打开按钮编辑器，选择 Standard Windows 3.1 Button System，在 System Buttons 下拉列表中选择 System 和 10，单击 OK 按钮，如图 43.5 所示。

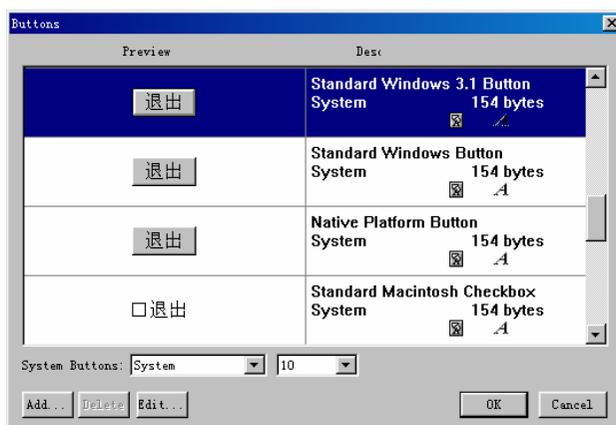


图 43.5 设置按钮外观形状

⑦ 回到 Properties 对话框，单击鼠标指针右边的小按钮。弹出 Cursor 对话框，选择 Standard Cursor，单击 OK 按钮。回到 Properties 对话框，在 Type 下拉列表框中选择 Buttons，将 Options 区域内的 Hide When Inaction 选项前面的复选框选上，如图 43.6 所示。

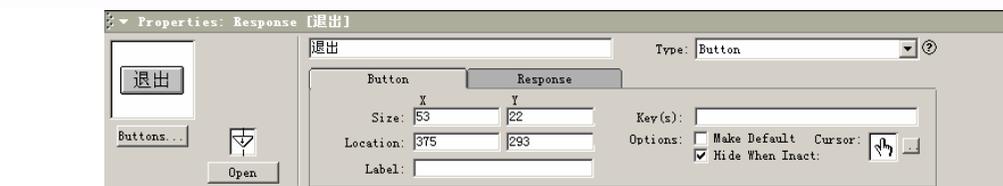


图 43.6 设置退出图标属性

⑧ 单击对话框中的 Response 选项卡，勾选 Scope 选项的 Perpetual 选项，其他设置如图 43.7 所示。用同样的方法设置其他功能按钮的响应属性。

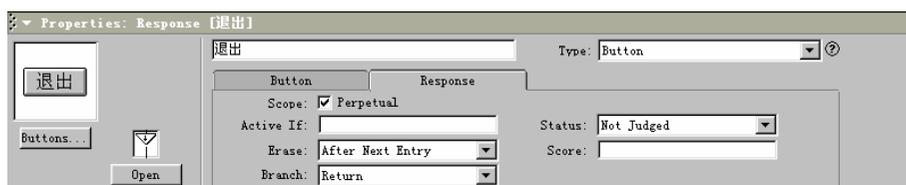


图 43.7 设置退出图标响应属性

⑨ 设置“图片 1”图标的内容。双击显示图标“图片 1”打开演示窗口，执行 File 菜单中的 Import 命令。导入一幅图片，并用鼠标拖动图片四周的控制点，调整图片的大小，以及在演示窗口的位置，如图 43.8 所示。用同样方法设置其他 3 幅图片。

⑩ 对齐排列按钮。按 CTRL+R 键运行程序，然后按 CTRL+P 键暂停程序。按住 Shift 键的同时用鼠标逐个选择窗口中的功能按钮，执行 Modify 菜单下的 Align 命令，弹出对齐设计图标面板，单击面板中右排的第 2 个居中对齐排列按钮和左排的第 4 个水平方向等间距排列按钮，这样就能使按钮对齐排列，如图 43.9 所示。

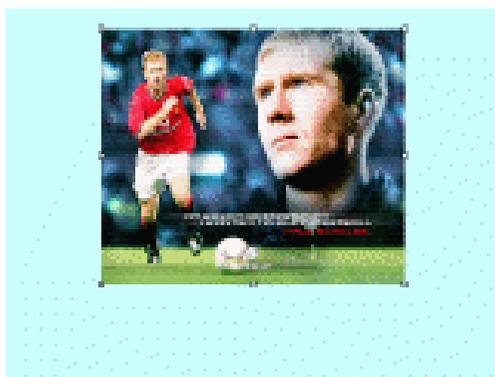


图 43.8 设置图片 1

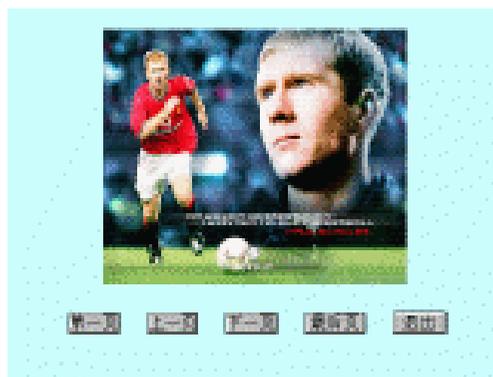


图 43.9 对齐排列图标

⑪ 预览效果。单击工具栏中的 Restart 按钮运行程序，可以单击其中任意按钮进行选择欣赏图片。



实例

44

纪念光盘 (1)



实例说明

本例制作一个“纪念光盘”，目的是介绍使用超文本跳转方式。

主要知识点：框架图标。



操作步骤

① 新建文件，保存为“44 纪念光盘.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择白色，如图 44.1 所示。

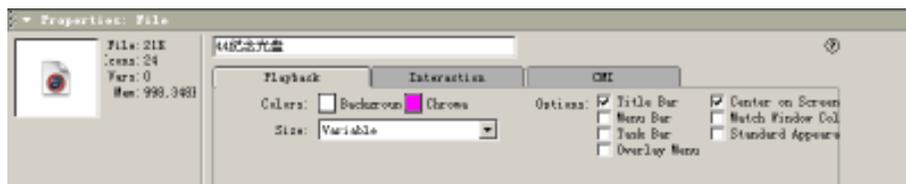


图 44.1 设置文件属性

③ 制作超文本首先要定制一个文本样式，然后在该样式中设置导航属性，将对象具体到指定的页。

④ 要建立文本对象，可以在 Interaction (交互) 图标或 Display (显示) 图标中直接创建。

⑤ 还要将该文本样式应用到文本对象上，这样，超文本就建好了。只要单击一下该文本，程序就会跳转到指定的页面上了。

⑥ 如图 44.2 所示，当程序运行时，将进入主画面，当将鼠标移向画面上的文本对象(超文本)时，光标将变为手型，如果单击超文本，将进入该文本所链接的页面。若用户单击画面下方的定向按钮，那么就可以进行前后翻页、到第一页、到最后一页等操作。

⑦ 程序编辑如图 44.3 所示的流程线，在 Framework (框架) 图标的右侧放置 5 个

Group (群组) 图标，分别命名为“主页面”、“第一部分~第四部分”。



图 44.2 程序主画面



图 44.3 程序流程线

⑧ 当程序进入“框架”图标，程序将首先执行该图标中的内容，因此将主画面部分放入其中。打开“主画面”图标，在其中放置程序的主画面，然后在“选择”图标中设置超文本对象，如图 44.3 所示。



实例

45

纪念光盘 (2)



实例说明

本例制作一个“纪念光盘”，目的是介绍使用超文本跳转方式。

主要知识点：导航图标。



操作步骤

- ① 确认当前打开的文件为“45 纪念光盘”（由实例 44 文件重命名而得）。
- ② 打开 Define Styles（文本样式）对话框，如图 45.1 所示。

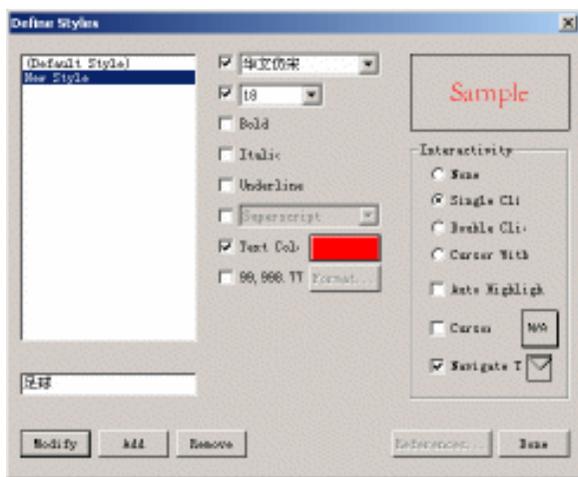


图 45.1 定制文本格式

- ③ 单击 Add 按钮，输入文本风格标题为“足球”，然后选定字体的各项设置。
- ④ 单击 Single Click 单选按钮，设置单击响应；单击 Auto Highlight 复选框，设置单击后文本以高亮显示；单击 Navigate To 复选框，设置文本导航属性。
- ⑤ 单击对话框中的按钮，打开 Properties：Navigation Style（导航属性）对话框，如图 45.2 所示，拖动 Page 列表框右侧的滚动条，在列表中选择“第一部分”图标作为超文本的跳转页面，单击 OK 按钮确认。

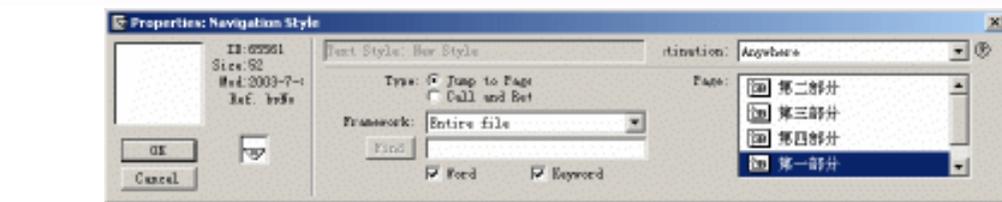


图 45.2 指定页链接



技巧

当该超文本产生响应时，程序就会自动跳至此处选择的“第一部分”的页面。

⑥ 返回图 45.1 所示的对话框后，单击 Done 按钮保存文本样式。然后打开“选择”图标，在演示窗口中输入“我们的冠军足球队”，如图 45.3 所示。再单击 Text 菜单下的 Apply Styles 命令，可以打开如图 45.3 左上角所示的 Apply Styles (应用文本样式) 对话框，单击“足球”复选框，窗口中的正文对象就变为超文本了。当程序运行时，如果用鼠标单击画面上的超文本时，程序就会跳转到指定的“第一部分”页面中去。



图 45.3 应用文本样式

⑦ 用同样的方式创建另外的几种文本样式，如图 45.3 中的“学习”，分别将它们应用于另外的两个超文本。

⑧ 返回到流程线后，打开“页面”框架图标，如图 45.4 所示，保留 4 个“导航”图标，将其他的“导航”图标删除。这样，在程序运行的画面上就只有四个按钮，First Page (第一页)、Previous Page (前一页)、Next Page (下一页) 和 Last Page (最后一页)。

⑨ 双击 First page (第一页) 导航图标，打开 Properties: Navigate Icon (导航图标属性) 对话框，如图 45.5 所示，选择 First page 单选按钮。在程序运行时，单击此按钮，程序就会直接跳转到 Framework (框架) 图标的第一页，即“主画面”图标。

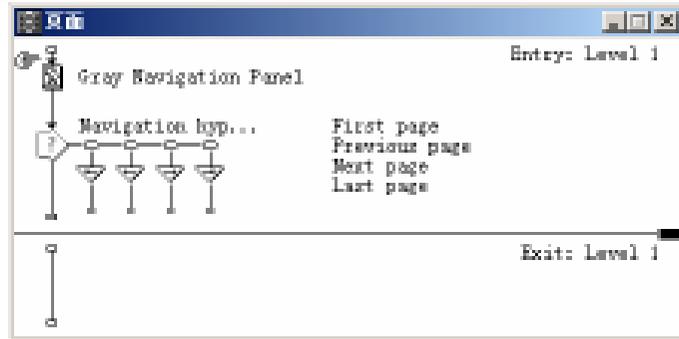


图 45.4 设置定向按钮

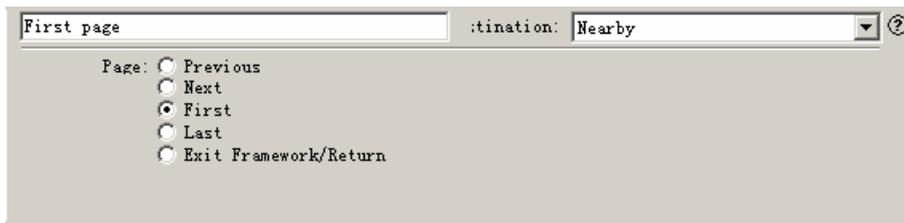


图 45.5 设置第一页

⑩ 用同样的方法设置其他 Navigation (导航) 图标。最后, 还要设置定向按钮的位置, 打开 Framework (框架) 图标, 进入 Interaction (交互) 图标的演示窗口, 将所有的按钮移至屏幕下方。



说明

在 Authorware 的默认定向结构中, 所有的 Navigation (导航) 图标都已被设置完毕, 可以直接使用。如果导航图标是后来添加的, 则必须重新进行设置。



技巧

为了增加画面的欣赏效果, 可以打开 Framework (框架) 图标的属性对话框, 在 Page Transition 下拉列表框中设置比较漂亮的页面过渡效果。

实例

46

纪念光盘(3)



实例说明

本例将讲述指定页面的设置方法，该方法在前面的超文本对象设置中也曾使用过，即在超文本的页面设置中使用了“指定的页面设置”方法。在此，就不再列举有关这方面的例子。为了能够掌握指定页面的设置方法，这里把其中的关键做一下提示。

主要知识点：导航图标。



操作步骤

- ① 确认前打开的是“46 纪念光盘”(由实例 45 文件重命名而得)。
- ② 在设置指定页面时，一般将 Properties : NavigateIcon (导航图标属性) 对话框的 Destination 下拉列表框设置为 Anywhere，如图 46.1 所示。

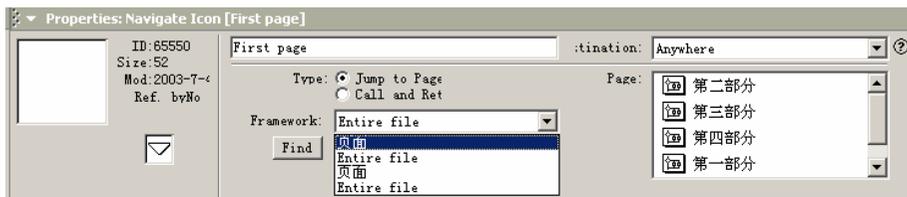


图 46.1 指定页面设置

- ③ 打开 Page 下拉列表框，选择页面跳转的范围，或在某个页面中跳转，或在整个文件中所有的页面中进行跳转。
- ④ 设置好页面范围后，在下面的列表框中就显示所有的页面，此时就可以查找所需要的页面，然后选择它，最后单击 OK 按钮确认。
- ⑤ 与超文本链接相比，使用“导航”图标来访问指定页的链接比较简单，跳转设置显得更为直接化，在某些框架结构复杂的程序中，使用指定页链接往往会使程序变得更加结构化。
- ⑥ 在程序执行时，单击“查找”按钮，Authorware 会弹出一个 Find 对话框，如图 46.2 所示，用户可以在 Find：文本框中输入一个查找的文本对象，然后单击 Find 按钮，Authorware 就会将所有的含有该查找对象的页面以列表的形式显示在 Page 列表框中。

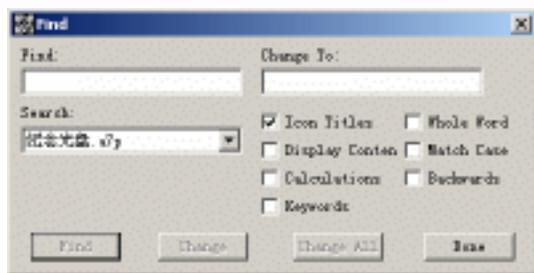


图 46.2 查找对话框

⑦ 如果要具体查阅某一页，可以单击列表框中的该页面，最后再单击 Go to Page 按钮即可显示相关的页面。在前面例子的基础上，再添加一个“查找”按钮，单击“查找”按钮，Authorware 就会弹出如图 46.2 所示的对话框。

⑧ 下面介绍“查找”按钮设置的全过程。打开 Framework(框架)图标 ,在 Interaction (交互) 图标的右侧再添加一个 Navigation (导航) 图标，该分支会被自动默认为按钮响应。

⑨ 双击响应类型按钮，打开 Properties : Response (响应属性) 对话框，如图 46.3 所示，在 Label 文本框中输入按钮的标签为“搜索”。同时，在对话框的预览框中还显示该按钮的样式，如果不想使用该按钮，可以单击对话框上的 Buttons 按钮来进行设置。

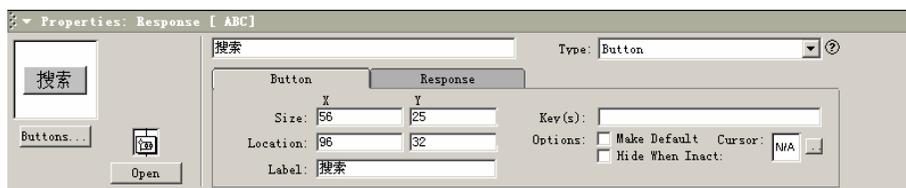


图 46.3 设置按钮标签

⑩ 单选 Buttons 按钮，弹出 Buttons 对话框，如图 46.4 所示，可以在按钮列表框中选择任何一种按钮样式，如图中黑圈内所示，设置合适的标签的字体和字号。如果还想将按钮设置得更加完美，可以单击 Edit 按钮进行设置。



图 46.4 设置标签字体

⑪ 设置完按钮样式后，还可以打开 Interaction（交互）图标，在演示窗口中将按钮拖放到合适的位置。

⑫ 双击“导航”图标，打开 Properties: NavigateIcon（导航图标属性）对话框，如图 46.5 所示，打开 Destination 下拉列表框，选择 Search 选项。然后，再把对话框中的各个选项进行具体设置，如果你还想在查找对话框弹出时显示某一特定的查找内容，可以在 Preset 文本框中直接输入，这样将避免每次重复输入。

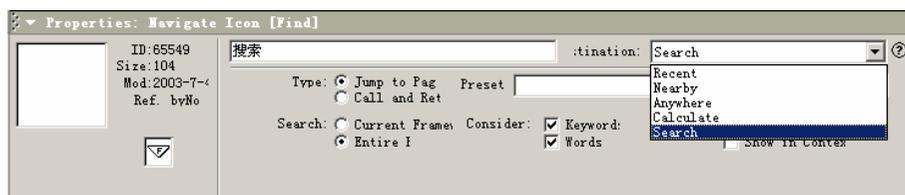


图 46.5 设置查找

⑬ 设置好查找属性后，运行程序，来看一下查找对话框的具体功能。如图 46.6 所示，在 Find：文本框中输入查找对象“足球队”。



图 46.6 查找内容

⑭ 使用查找功能可以很方便地查阅自己感兴趣的内容，如果觉得该对话框的界面设置不太友好，Authorware 还提供了一个 Navigation Setup 对话框对该界面进行修改。



实例

47

倒计时



实例说明

本例是一个制作“倒计时”效果的实例，运行后出现时钟提示，屏幕中出现倒计时提示，当剩余时间为 0 秒时，自动退出程序。

主要知识点：交互图标。



操作步骤

- ① 新建文件，保存为“47 倒计时.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，单击 Color 右边的 Background Color 按钮，打开 Color 对话框，从中选择蓝色，如图 47.1 所示。



图 47.1 设置文件属性

③ 设计程序流程线。从面板中拖放一个 Calculation（计算）图标到流程线上，并将 Calculation（计算）图标命名为“初始化窗口”，然后再拖放一个显示图标到流程线上，命名为“背景”。接着拖放一个交互图标到流程线上，命名为 Interaction（交互），最后再拖放一个 Calculation 图标和一个组图标到交互图标的右下方，分别命名为 Times=0 和 TRUE。流程线如图 47.2 所示。

④ 设置“初始化窗口”图标内容。双击计算图标，在打开的代码窗口中输入 Resizewindow(480,360) Times:=0，如图 47.3 所示。

⑤ 设置“背景”图标的内容。利用导入命令导入一幅图片，调整图片大小使其充满整个窗口。

⑥ 设置交互图标的内容。按住 Shift 键，双击流程线上的交互图标，打开演示窗口，用绘图工具箱中的文本工具输入“恭喜你们马上就要成为夫妻了！让我们一起倒数吧：



{Times} ”, 设置文字的大小及颜色, 设置文字显示模式为透明模式, 如图 47.4 所示。

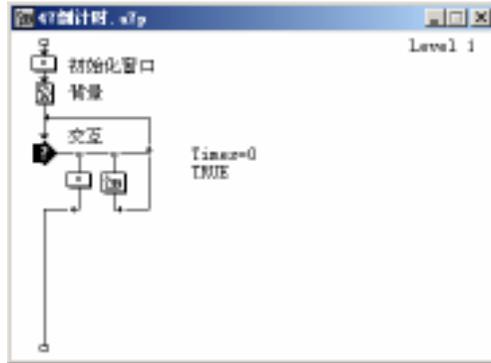


图 47.2 流程线



图 47.3 初始化窗口代码



图 47.4 设置交互图标

⑦ 设置 Times=0 图标属性, 双击流程线上 Times=0 图标上方的控制按钮, 打开属性对话框, 设置如图 47.5 所示。

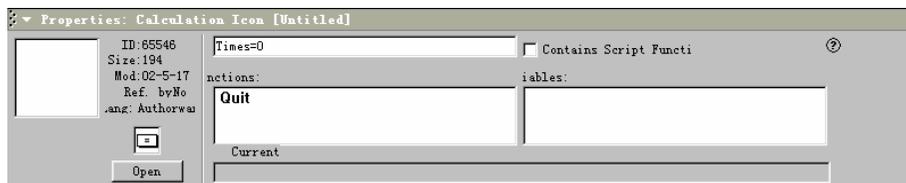


图 47.5 设置条件响应



⑧ 设置二级程序流程线。双击流程线上的 TRUE 图标，打开二级流程线窗口，从设计窗口拖放一个 Wait 图标到流程线上，命名为 Wait（等待），然后拖放一个计算图标到流程线上，并将计算图标命名为“倒计时”，如图 47.6 所示。

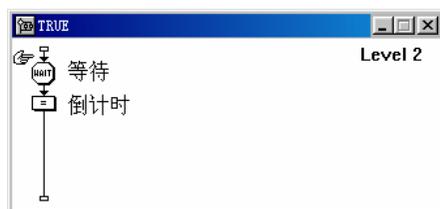


图 47.6 二级程序流程线

⑨ 设置等待图标属性。双击流程线上的等待图标，打开等待属性窗口进行设置，如图 47.7 所示。



图 47.7 设置等待图标

⑩ 设置“倒计时”图标的代码。双击“倒计时”计算图标，在打开的代码窗口中输入 `Times:=Times-1`，如图 47.8 所示。

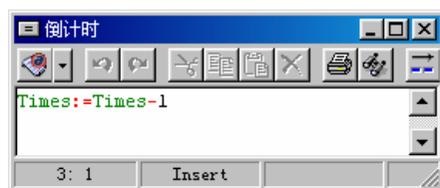


图 47.8 代码窗口

⑪ 单击工具栏上的 Restart 按钮预览效果，并保存程序文件。

实例

48

文件读取



实例说明

本例能够实现单击演示窗口中的“读入”按钮,打开“选择读入文件”对话框,单击“退出”按钮,退出程序。

主要知识点: Authorware 函数。



操作步骤

- ① 新建文件,保存为“48 文件读取.a7p”。
- ② 执行命令 Modify File Properties,打开属性设置窗口,设置 Size 为 Variable,选择 Center On Screen 复选框和 Menu Bar 复选框,如图 48.1 所示。

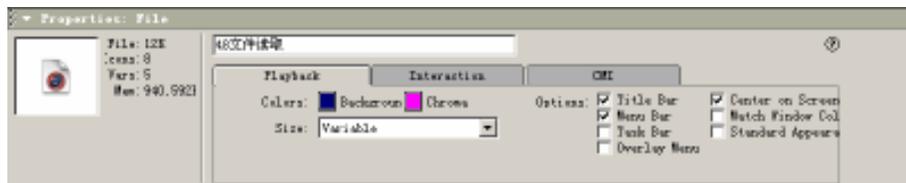


图 48.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击该图标,打开演示窗口,使用绘图工具箱中的文本工具,在演示窗口内创建两个新的文本,输入“读取文件”。

④ 在流程线上添加一个 Calculation (计算) 图标,命名为“初始化”,双击该图标打开编辑窗口,如图 48.2 所示输入条件语句。

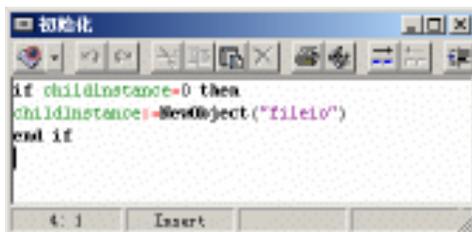


图 48.2 设置“初始化”窗口



⑤ 在流程线上添加一个 Interaction (交互) 图标到流程线上, 命名为“控制”, 再添加一个 Group (群组) 图标到交互图标的右边, 打开交互类型对话框, 选择 Button 交互类型, 命名为“读取”。

⑥ 双击“读取”图标, 打开二级流程线设计窗口, 添加两个 Calculation (计算) 图标, 分别命名为“打开对话框”、“读取文本”。双击“打开对话框”图标打开编辑窗口, 输入如图 48.3 所示的语句, 然后再添加一个 Display (显示) 图标命名为“显示文本”, 如图 48.4 所示。使用绘图工具箱中的文本工具在演示窗口中创建一个新的文本, 输入“{text}”。



图 48.3 设置“读取文本”窗口



图 48.4 二级流程线

⑦ 向一级流程线中添加一个 Calculation (计算) 图标, 命名为“退出”。双击“退出”图标打开编辑窗口, 输入 Quit()。

⑧ 存盘, 运行此实例, 最终的主程序流程线如图 48.5 所示。最后运行实例, 单击其中的“读取”按钮, 弹出如图 48.6 所示的对话框。

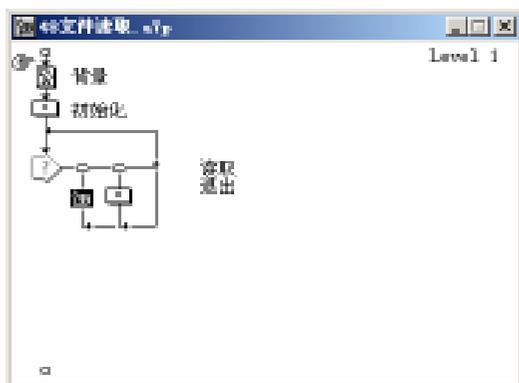


图 48.5 演示效果图



图 48.6 “打开”对话框

实例

49

游标驱动



实例说明

本例是一个制作“游标驱动”效果的实例，运行后屏幕上出现蓝色的椭圆形驱动游标和提示信息，用鼠标拖动游标上的控制滑杆，同时在游标的下方显示游标的刻度。

主要知识点：知识对象、Authorware 函数。



操作步骤

- ① 新建文件，保存为“49 游标驱动.a7p”。
- ② 执行命令 Modify File Properties，打开属性设置窗口，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，如图 49.1 所示。



图 49.1 设置文件属性

- ③ 设计程序流程线。首先添加一个 Calculation (计算) 图标，命名为“初始化窗口”，然后添加一个 Display (显示) 图标，命名为“背景”。

- ④ 设置“初始化窗口”图标内容。双击该图标，在打开的代码窗口中输入 Resizewindow(480, 360)，如图 49.2 所示，其作用就是初始化演示窗口的大小。

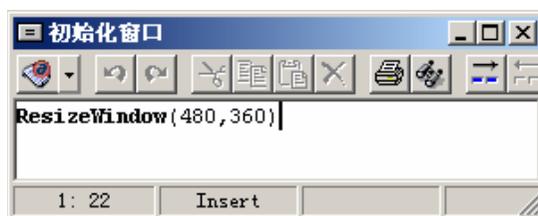


图 49.2 设置“初始化窗口”



⑤ 设置“背景”图标的内容，双击“背景”图标打开演示窗口。用文本工具在窗口中输入提示用户输入信息的文字“请拖动游标上的滑杆：”和“游标的刻度为：{PathPosition@“slider”}”，调整文字的大小、颜色以及在演示窗口中的位置，并将文字的显示模式设置为透明模式，如图 49.3 所示。



图 49.3 设置“背景”图标

⑥ 执行 Window Panels Knowledge Objects 菜单命令，打开 Knowledge Objects 窗口，在 Category 下的列表中选择 All 知识对象，在列表框中选择 Slider，如图 49.4 所示，将其拖动到流程线中，命名为“Slider 对象”。

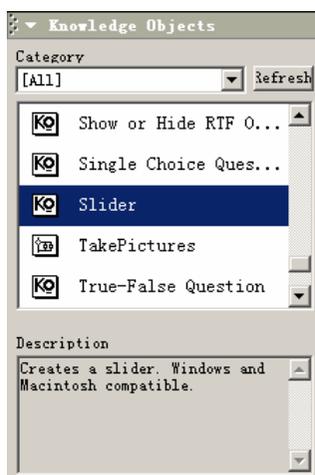


图 49.4 选择知识对象

⑦ 当将知识对象拖动到流程线时会弹出如图 49.5 所示的知识对象向导，该对话框左边是知识对象的 7 个选项设置。

⑧ 设置游标样式。单击 Next 按钮，进入下一步的 Slider Layout 对话框，在此对话框中，系统提供了 27 种游标样式和 5 种游标形状供用户选择。单击 Orientation 下的 Horizontal 选项，在 Select a layout 列表框中选择 Red, rounded，这样就将游标设置成红色的椭圆形

游标，对话框设置如图 49.6 所示。

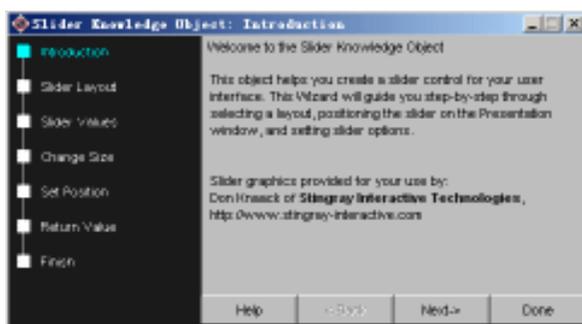


图 49.5 “知识对象设置向导”对话框

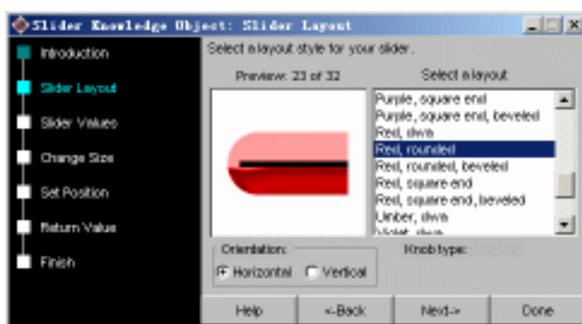


图 49.6 设置游标格式

⑨ 设置游标区间。单击 Next 按钮，进入下一步的 Slider Values 对话框，在此对话框中，可以设置游标刻度的起止区间。在 Minimum Value 文本框中输入“1”，在 Maximum Value 文本框中输入“100”，这样就将游标刻度的起止区间定义为 1~100，对话框设置如图 49.7 所示。

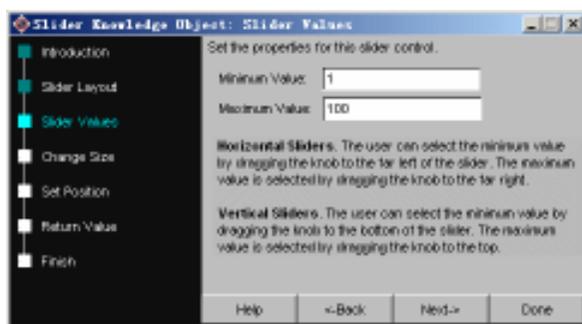


图 49.7 设置游标区间

⑩ 设置游标尺寸。单击 Next 按钮，进入下一步的 Change Size 对话框，如图 49.8 所示。

⑪ 设置游标位置。单击 Next 按钮，进入下一步的 Set Position 对话框，如图 49.9 所示。



图 49.8 设置游标尺寸

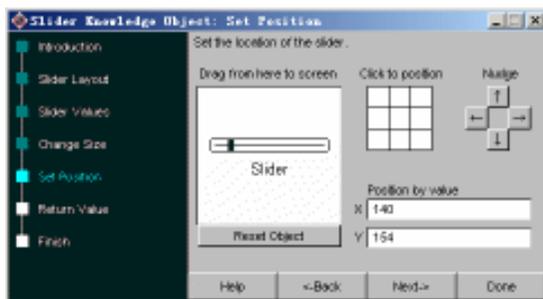


图 49.9 设置游标位置

12 游标变量值介绍。单击 Next 按钮,进入下一步的 Return Value 对话框,如图 49.10 所示。

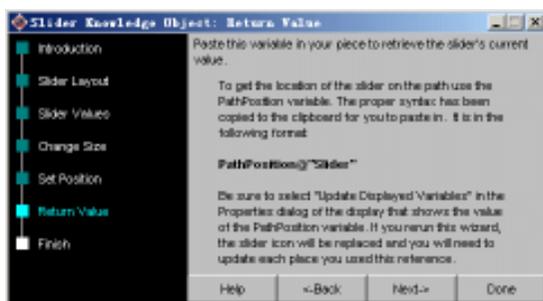


图 49.10 游标变量值介绍

13 完成向导的设置。单击 Next 按钮,进入下一步的 Finish 对话框,如图 49.11 所示。

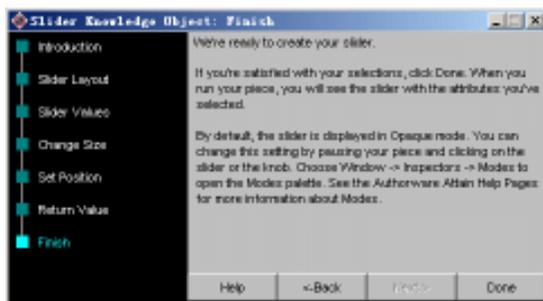


图 49.11 Finish 对话框

14 存盘,运行此实例,程序流程线如图 49.12 所示。



图 49.12 流程线



实例

50

时间和日期显示



实例说明

本例是一个制作“时间和日期显示”效果的实例，运行后在屏幕的左边缓慢地显示出当前系统的日期，然后再显示当前系统的时间。

主要知识点：Authorware 函数、过渡效果。



操作步骤

- ① 新建文件，保存为“50 时间和日期显示.a7p”。
- ② 执行命令 Modify File Properties，打开属性设置窗口，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，如图 49.1 所示。

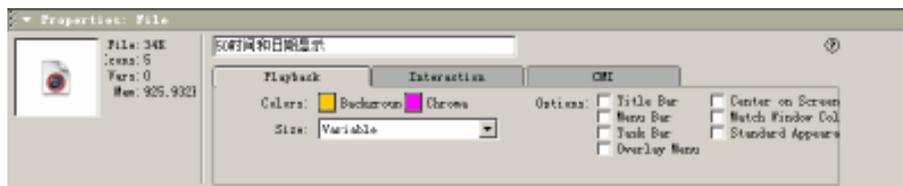


图 50.1 设置文件属性

- ③ 设计程序流程线。首先添加一个 Calculation (计算) 图标，命名为“初始化窗口”，然后添加一个 Display (显示) 图标，命名为“显示日期”。再添加一个 Wait (等待) 图标，命名为“等待”。最后再添加一个 Display (显示) 图标，命名为“显示时间”，程序流程线如图 50.2 所示。



图 50.2 流程线

④ 设置“初始化窗口”图标内容。双击“初始化窗口”图标，在打开的代码窗口中输入 `ResizeWindow(650, 430)`，如图 50.3 所示，其作用就是初始化演示窗口的大小。

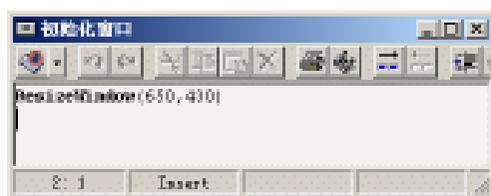


图 50.3 设置“初始化窗口”

⑤ 设置“背景”图标的内容。双击“背景”图标打开演示窗口，执行 `File Import and Export Import Media` 命令，导入一幅图片，并调整大小，如图 50.4 所示。

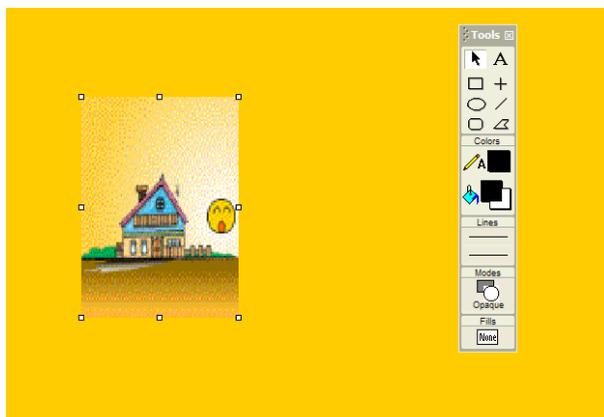


图 50.4 设置“背景”图标

⑥ 设置“显示日期”图标的内容。双击该图标，选择绘图工具箱中的文本工具，在图中输入如图 50.5 所示的文字，并对字体进行设置。

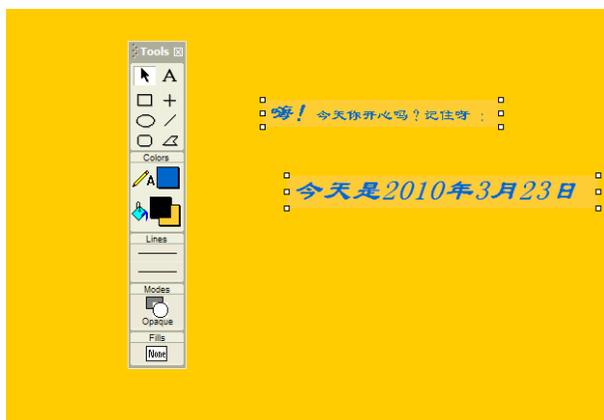


图 50.5 设置显示日期

⑦ 设置 Wait (等待) 图标的内容。单击该图标打开属性对话框，选择 `Events` 区域中的 `Mouse Click` 和 `Key Press` 复选框，在 `Time` 文本框中输入“0.5”，如图 50.6 所示。



图 50.6 设置属性对话框

⑧ 设置“显示时间”图标的内容。双击流程线上的“显示时间”图标，选择绘图工具箱中的文本工具，在图中输入“现在的时是{FullTime}”，并对字体进行设置，如图 50.7 所示。



图 50.7 设置显示时间

⑨ 设置文本的过渡效果。双击流程线上的“显示日期”图标，选中窗口中的文字，按住 Ctrl+I 打开属性对话框，单击 Transition 选项右边的按钮，弹出 Transition 对话框，在 Catogor 分类列表选取 Push，在转换列表选取 Push Left，在 Duration 文本框中输入“2”，在 Smoothness 文本框中输入“16”，在 Effects 游标中选择 Changing Area(s)，单击 OK 按钮即可，如图 50.8 所示。

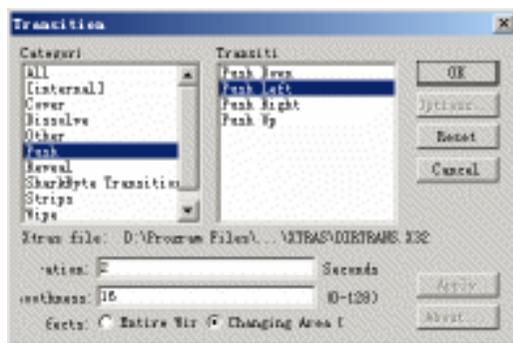


图 50.8 设置过渡效果

⑩ 在第 9 步中单击 OK 按钮后会出现 Properties:Display Icon 对话框，选中 Options

区域内的 Update Displayed Variable 选项前的复选框，如图 50.9 所示。

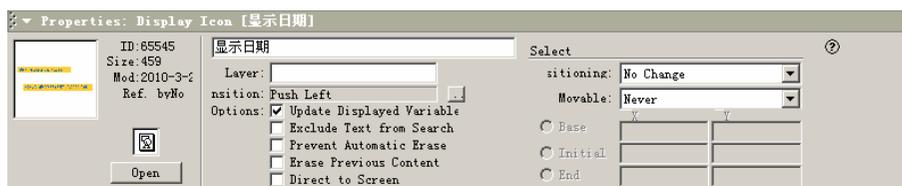
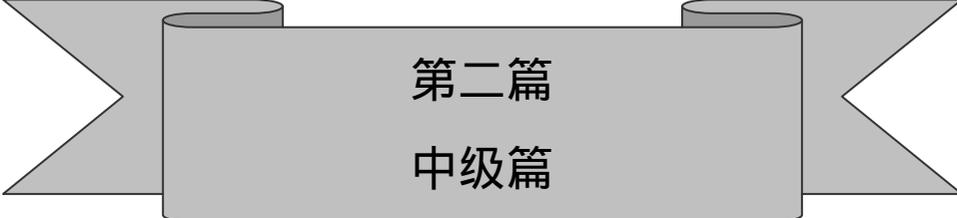


图 50.9 更新变量的值

② 存盘，运行此实例。



第二篇 中级篇

本篇对第一篇所学的内容做了综合性的融合,通过第一篇的学习,读者对 Authorware 7.0 有了初步的了解,对 Authorware 的各种图标,如显示图标、动画图标、交互图标、声音图标、导航和框架图标等的使用有了一定的认识,但是把这些基本的功能、命令等有机地结合在一起还有一些困难。读者通过本篇 25 个实例,可以学习在多媒体制作过程中的常用技巧。

Let's GO! 

实例

51

条形渐变



实例说明

该实例建立一个条形渐变效果。

主要知识点：Authorware 变量和函数。



操作步骤

① 新建文件，保存为“51 条形渐变.a7p”。

② 执行命令 Modify File

Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 51.1 所示。

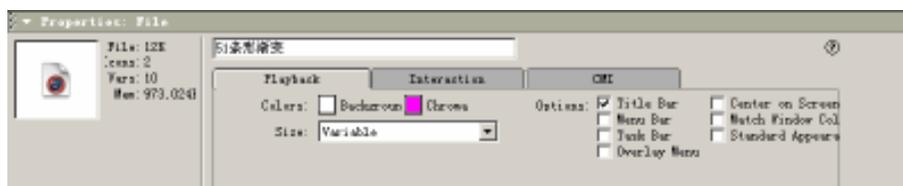


图 51.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“条形渐变”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 51.2 所示。

④ 在流程线上添加一个 Calculation (计算) 图标，命名为“条形渐变”，双击打开代码窗口，输入代码，如图 51.3 所示。

输入的代码如下：

```
h:=50
k:=160 --中心位置坐标，根据窗口大小设定
x:=h
b:=0
```



```
repeat while x<=400
x1:=x+1
if b>255 then
b:=255
end if
SetFrame(1,RGB(255,b,0))
Line(20,x,k,x1,k)
x:=x+1
b:=b+1
end repeat
```



图 51.2 设置背景

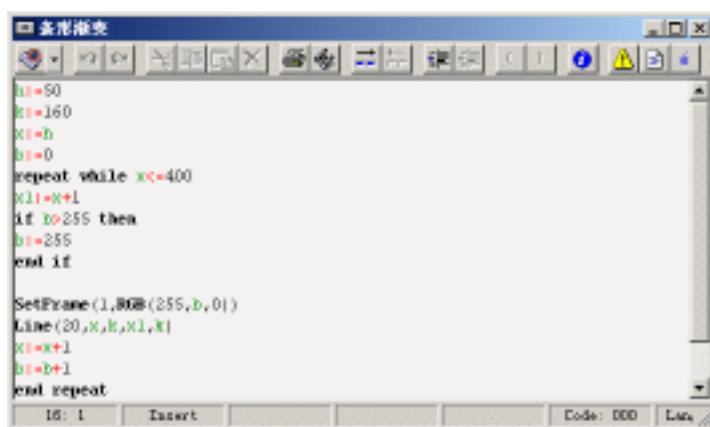


图 51.3 输入代码



如果要改变色带的宽度,更改 Line(20,x,k,x1,k)中的数值 20 即可;如果要改变渐变条的长度,需要调整 repeat while x<=400 语句中的数值 400。



技巧

获取颜色的 RGB 值其实很容易：打开 Windows 系统自带的绘图程序，执行命令“颜色”“编辑颜色”，打开“编辑颜色”对话框，单击 **规定自定义颜色 (U) >>** 按钮，在对话框右侧可以查看颜色的 RGB 值，如图 51.4 所示。

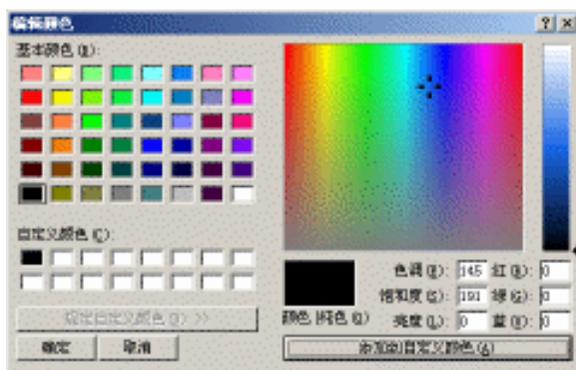


图 51.4 获取颜色 RGB 值

⑤ 存盘，运行，流程线如图 51.5 所示。



图 51.5 流程线



实例

52

径向渐变



实例说明

该实例建立一个径向渐变效果。

主要知识点：Authorware 变量和函数。



操作步骤

- ① 新建文件,保存为“52 径向渐变.a7p”。
- ② 执行命令 Modify File Properties, 打开属性对话框, 设置 Size 为 Variable, 取消选择 Menu Bar 复选框, 如图 52.1 所示。

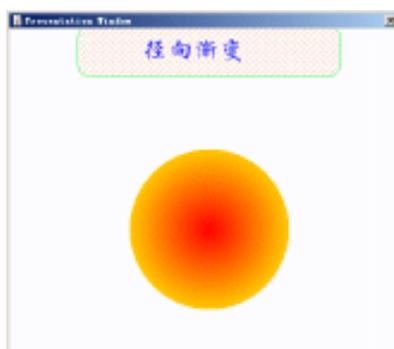


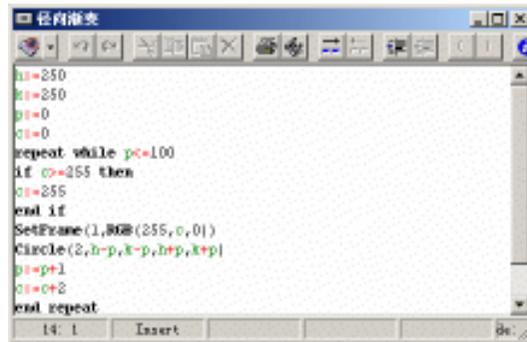
图 52.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标, 命名为“背景”, 双击打开演示窗口, 使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域, 双击圆角矩形工具 , 打开填充面板, 设置填充方案, 并双击椭圆形工具 , 打开调色板, 为填充图案着色。然后, 使用文本工具  输入标题“径向渐变”, 执行命令 Text Font 设置字体为“华文行楷”, 执行命令 Text Size, 设置字体大小为“24”磅, 同时设置字体颜色, 双击箭头工具 , 设置文本的显示模式为 Transparent, 如图 52.2 所示。



图 52.2 设置背景

④ 在流程线上添加一个 Calculation (计算) 图标, 命名为“条形渐变”, 双击打开代码窗口, 输入代码, 如图 52.3 所示。



```

径向渐变
h:=250
k:=250
p:=0
c:=0
repeat while p<=100
if c>=255 then
c:=255
end if
SetFrame(1,RGB(255,c,0))
Circle(2,h-p,k-p,h+p,k+p)
p:=p+1
c:=c+2
end repeat

```

图 52.3 输入代码

输入的代码如下：

```

h:=250
k:=250  --中心位置坐标, 根据窗口大小设定
p:=0
c:=0
repeat while p<=100
if c>=255 then
c:=255
end if
SetFrame(1,RGB(255,c,0))
Circle(2,h-p,k-p,h+p,k+p)
p:=p+1
c:=c+2
end repeat

```

⑤ 存盘, 运行, 流程线如图 52.4 所示。

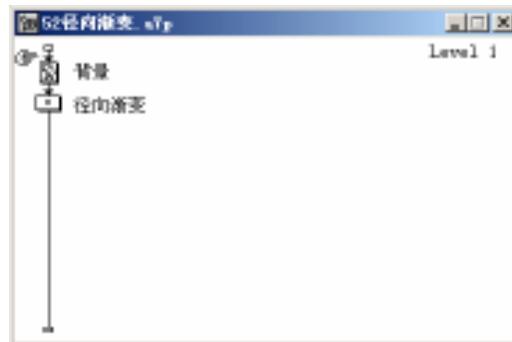


图 52.4 流程线



实例

53

扇形渐变



实例说明

该实例建立一个扇形渐变效果。

主要知识点：Authorware 变量和函数。



操作步骤

- ① 新建文件，保存为“53 扇形渐变.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 53.1 所示。

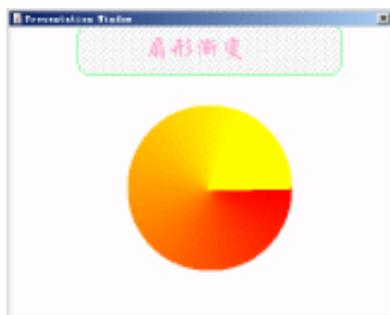


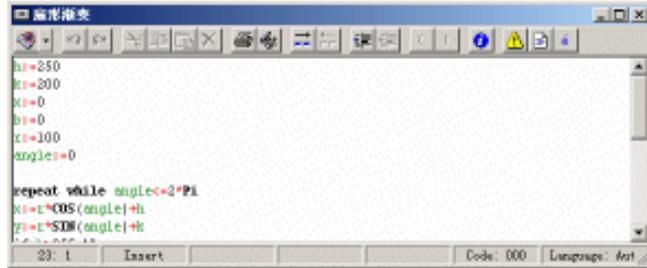
图 53.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“扇形渐变”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 53.2 所示。



图 53.2 设置背景

④ 在流程线上添加一个 Calculation (计算) 图标, 命名为“扇形渐变”, 双击打开代码窗口, 输入代码, 如图 53.3 所示。



```

h:=250
k:=200
x:=0
b:=0
r:=100
angle:=0

repeat while angle<=2*Pi
x:=r*COS(angle)+h
y:=r*SIN(angle)+k
if b>255 then
b:=255-(b-255)
end if
SetFrame(1,RGB(255,b,0))
Line(5,h,k,x,y)
b:=b+1
x:=x+1
angle:=angle+0.02
end repeat

```

图 53.3 输入代码

输入的代码如下：

```

h:=250
k:=200 --中心位置坐标,根据窗口大小设定
x:=0
b:=0
r:=100
angle:=0
repeat while angle<=2*Pi
x:=r*COS(angle)+h
y:=r*SIN(angle)+k
if b>255 then
b:=255-(b-255)
end if
SetFrame(1,RGB(255,b,0))
Line(5,h,k,x,y)
b:=b+1
x:=x+1
angle:=angle+0.02
end repeat

```

⑤ 存盘, 运行, 程序流程线如图 53.4 所示。

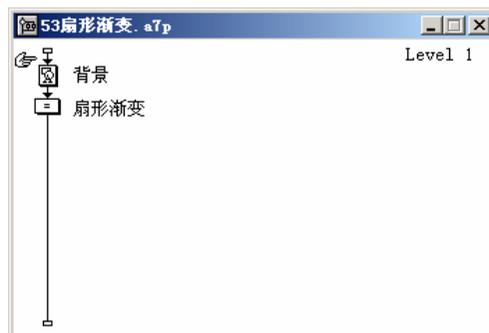


图 53.4 流程线



实例

54

抢答题



实例说明

该实例建立一个抢答题效果,用户单击“按键答题”按钮,将在屏幕上循环显示题号,直到用户按下 Ctrl 键才停止显示题号,显示题目供用户作答。

主要知识点:判断图标。



操作步骤

- ① 新建文件,保存为“54 抢答题.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 54.1 所示。

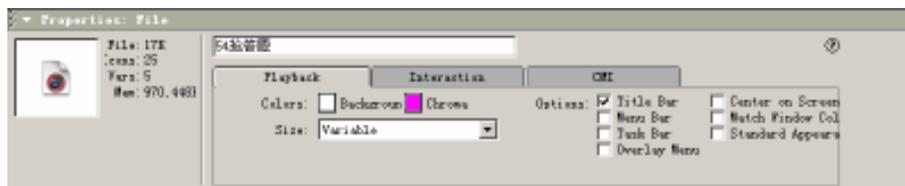


图 54.1 设置文件属性

- ③ 在流程线上添加一个 Interaction (交互) 图标,命名为“抢答题”,在其右侧添加一个 Group (群组) 图标,交互类型设置为 Button,并命名为“按键答题”。双击交互类型标记,打开其属性对话框,在 Response 选项卡内选择 Perpetual 复选框,如图 54.2 所示。

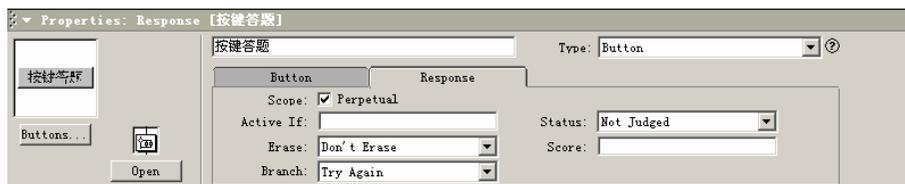


图 54.2 设置按钮交互属性

- ④ 双击群组图标,打开二级窗口,添加两个 Decision (判断) 图标,分别命名为“显示编号”和“显示题目”,并分别在其右侧添加 3 个 Group (群组) 图标。双击“显

示编号”图标，打开其属性对话框，在 Repeat 下拉菜单中选择 Until True，选项，在下面的文本框中输入 ControlDown，选择 Branch 下拉菜单中的 Sequentially 选项，如图 54.3 所示。



图 54.3 设置分支图标属性

⑤ 双击“显示题目”图标，打开其属性对话框，选择 Branch 下拉菜单中的 To Calculated Path，在下面的文本框中输入 xd，如图 54.4 所示。



图 54.4 设置分支图标属性

⑥ “按键答题”二级窗口的流程图如图 54.5 所示。

⑦ 双击“分支 1”图标，在其窗口内分别添加 Calculation（计算）图标、Display（显示）图标和 Wait（等待）图标，如图 54.6 所示。双击“判断 1”图标，在代码窗口中输入下列代码：

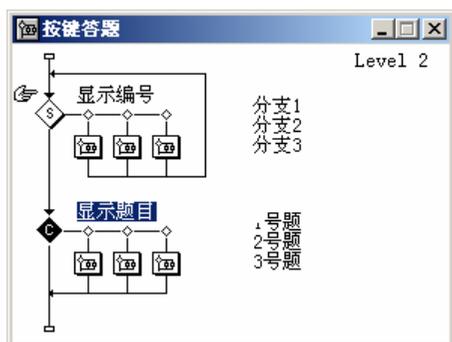


图 54.5 程序流程图

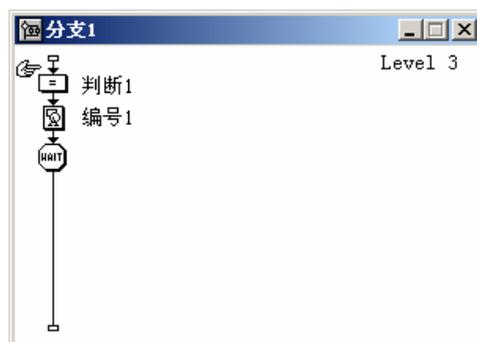


图 54.6 添加图标

```

xd:=1
if ce=1 then
    GoTo(IconID@"分支 2")
end if
    
```

⑧ 双击“编号 1”图标，在窗口中输入题目编号“1”。双击 Wait（等待）图标，打开其属性对话框，在 Time Limit 下拉菜单中输入“0.1”，取消选择其他复选框，如图 54.7 所示。



图 54.7 设置等待图标属性

⑨ 重复步骤 8 和 9，设置其他 2 个分支的内容，所不同的是，在“分支 2”和“分支 3”的显示窗口中分别输入“2”和“3”。

⑩ 回到“按键答题”二级窗口，双击“1 号题”群组图标，在三级窗口中分别添加 Calculation（计算）和 Display（显示）图标，形成图 54.8 所示的流程线。

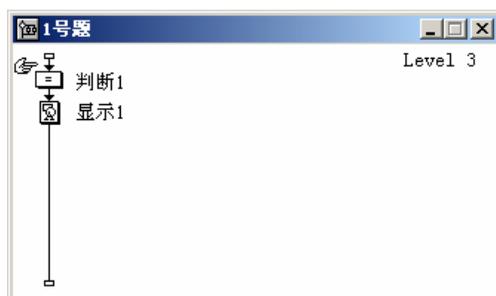


图 54.8 在三级窗口中添加图标

⑪ 双击“判断 1”图标，在代码窗口中输入下列代码：

```
ce1:=1
```

双击“显示 1”图标输入题目 1 的内容。

⑫ 使用和上述步骤同样的方法设置其他 2 个群组图标的内容。

⑬ 存盘，运行，流程线如图 54.9 所示。

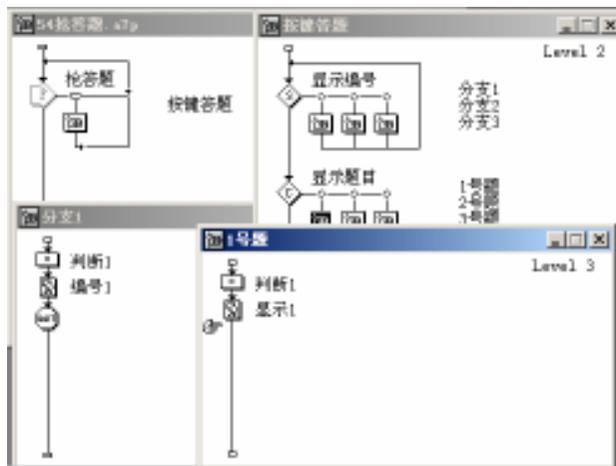


图 54.9 流程线

实例

55

控制面板



实例说明

该实例建立一个应用程序,单击“打开控制面板”按钮,将打开 Windows 系统的控制面板。

主要知识点:交互图标、按钮交互、函数。



操作步骤

- ① 新建文件,保存为“55 控制面板.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 55.1 所示。



图 55.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“打开控制面板”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 55.2 所示。



图 55.2 设置背景



④ 在流程线上添加一个 Interaction (交互) 图标, 命名为“打开控制面板”, 在其右侧添加 2 个 Calculation (计算) 图标, 交互类型均设置为 Button, 分别命名为“打开控制面板”和“退出”, 如图 55.3 所示。

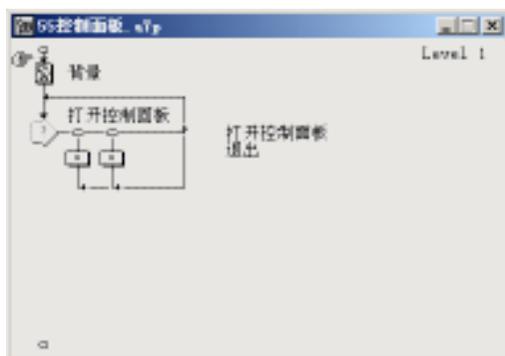


图 55.3 流程线

⑤ 双击“打开控制面板”图标上面的交互类型标记, 打开其属性对话框, 在 Button 选项卡内选择 Hide When Inactive 复选框, 如图 55.4 所示。单击 Buttons 按钮, 打开 Buttons 对话框, 设置按钮的样式, 如图 55.5 所示。

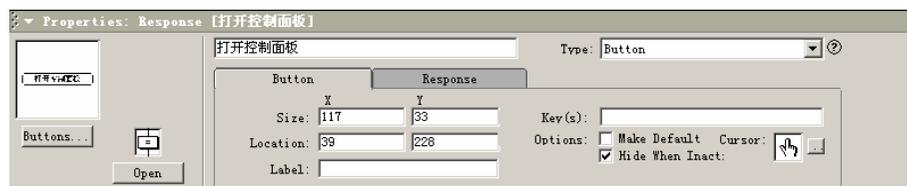


图 55.4 设置按钮交互属性

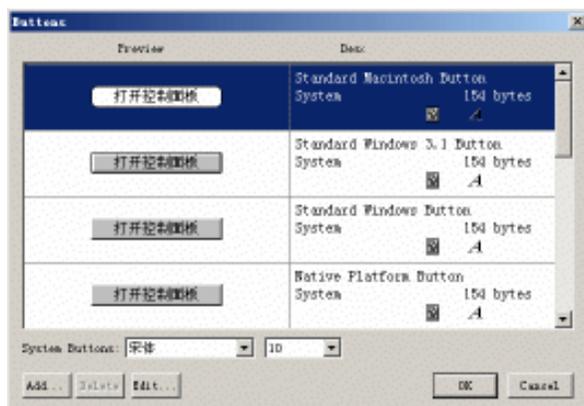


图 55.5 设置按钮样式

⑥ 设置光标的样式, 如图 55.6 所示。重复步骤 5 和 6 设置“退出”按钮的属性。

⑦ 双击“打开控制面板”图标, 打开代码窗口, 输入代码 WinExec("rundll32.exe shell32.dll,Control_RunDLL",1), 如图 55.7 所示。关闭代码窗口时, 需要调用 Windows 系统的 Rundll32.exe 函数。

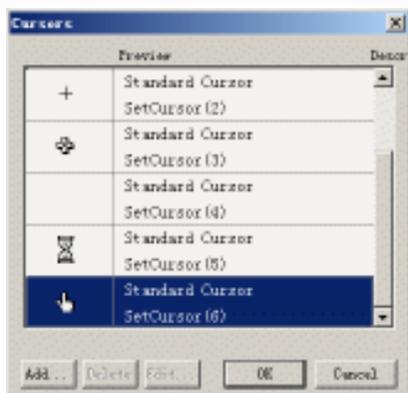


图 55.6 设置光标属性

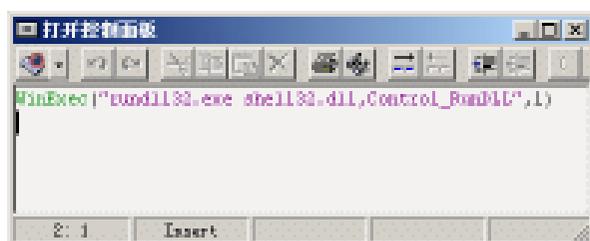


图 55.7 输入代码

⑧ 双击“退出”图标，输入代码 Quit(0)，如图 55.8 所示。

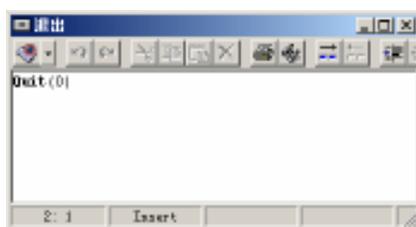


图 55.8 输入代码

⑨ 存盘，运行，单击程序画面上的按钮，打开控制面板，如图 55.9 所示。

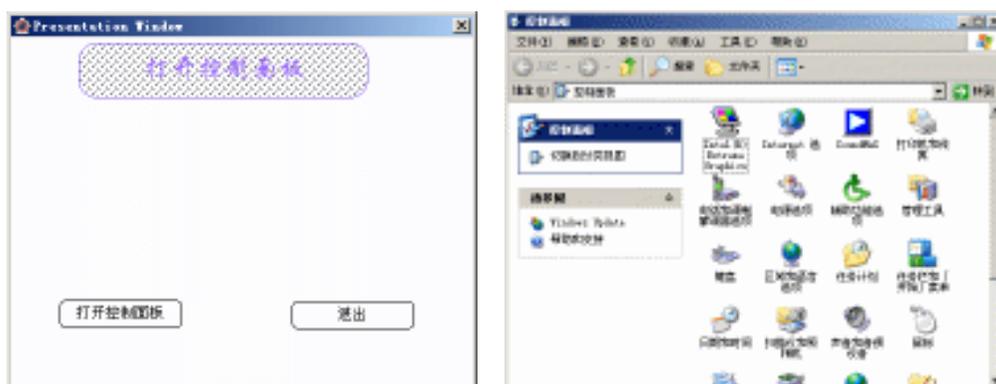


图 55.9 程序运行效果



实例

56

浏览器



实例说明

该实例建立一个简单的 Web 浏览器，可以实现基本的网页浏览操作，例如：倒退、前进、停止、主页、刷新等。

主要知识点：Authorware ActiveX 控件。



操作步骤

- ① 新建文件，保存为“56 浏览器.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 640 × 480 VGA, Mac13，取消选择 Menu Bar 复选框，如图 56.1 所示。

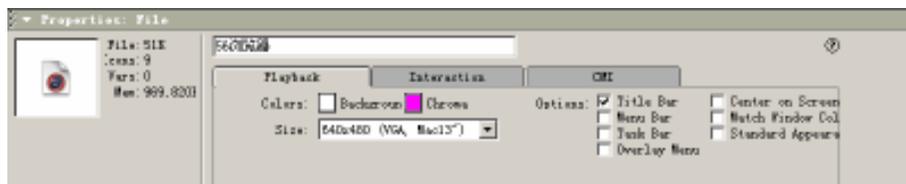


图 56.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，打开演示窗口，使用文本工具在适当的位置输入“网页地址”，并设置文本的属性。

④ 回到流程线，执行命令 Insert Control ActiveX，打开 Select ActiveX Control 对话框，在列表中选择“Microsoft Web 浏览器”，单击 OK 按钮插入控件，如图 56.2 所示。插入控件后，流程线上自动添加图标，将其重新命名为 Browser，方便以后调用，如图 56.3 所示。

⑤ 接下来，打开图 56.4 所示的属性配置窗口，单击 OK 按钮，选择默认配置即可。

⑥ 在流程线上添加一个 Interaction (交互) 图标，命名为“菜单”，在其右侧添加一个 Calculation (计算) 图标，交互类型设置为 Text Entry，命名为“*”，双击该图标，打开代码窗口，输入下列代码 CallSprite(@"Browser",#navigate,EntryText)，如图 56.5 所示。此时在演示窗口中出现文本那框，将它移动到“网页地址”文本后面。

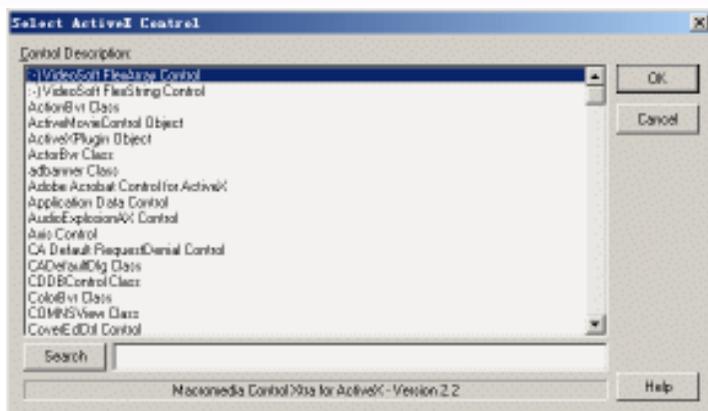


图 56.2 选择插入控件



图 56.3 插入控件后的流程线

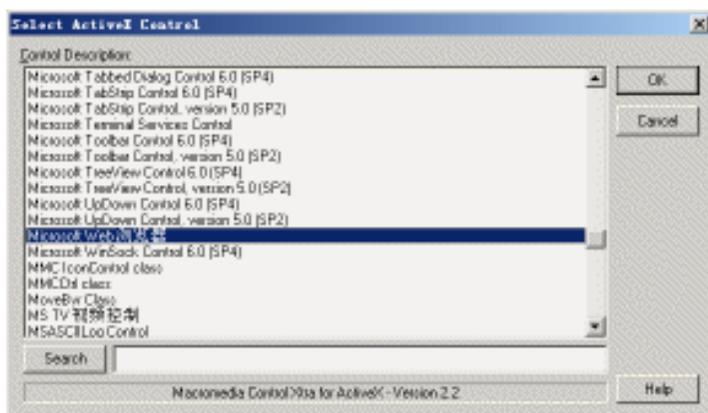


图 56.4 控件属性配置

⑦ 继续添加 5 个 Calculation (计算) 图标, 分别命名为“后退”、“前进”、“停止”、“刷新”、“主页”, 双击图标上面的交互类型标记, 打开交互属性对话框, 在 Type 下拉菜单中选择 Button 选项, 如图 56.6 所示。

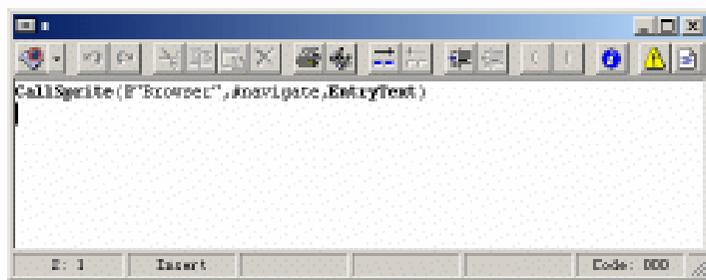


图 56.5 输入代码

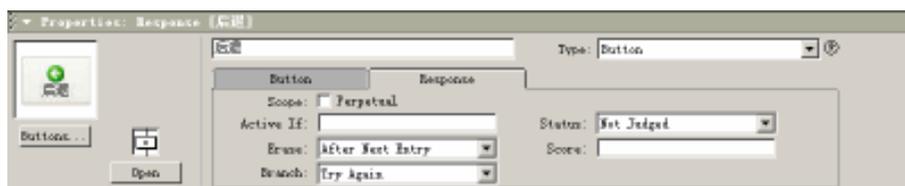


图 56.6 设置交互类型

⑧ 设置这 5 个按钮的样式，以“后退”按钮为例。单击 Properties : Response 对话框中的 Buttons 按钮，打开 Buttons 对话框，单击 Add...按钮，打开 Button Editor 对话框，单击 Graphic 后面的 Import..按钮导入按钮图像，其他选项不用设置，如图 56.7 所示。

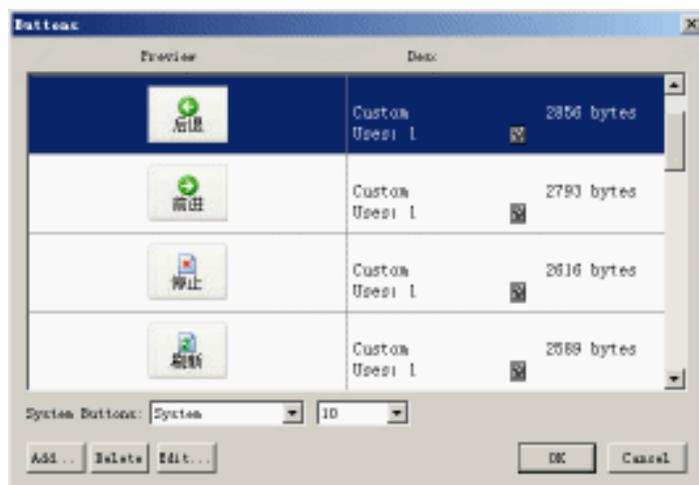


图 56.7 自定义按钮样式

⑨ 打开 5 个 Calculation (计算) 图标的代码窗口，分别在各自的窗口中输入下列代码：

- 后退：callsprite(@?browser?, # goback)
- 前进：callsprite(@?browser?, # goforward)
- 停止：callsprite(@?browser?, # stop)
- 刷新：callsprite(@?browser?, # refresh)
- 主页：callsprite(@?browser?, # gohome)

⑩ 存盘，运行，程序流程图如图 56.8 所示。

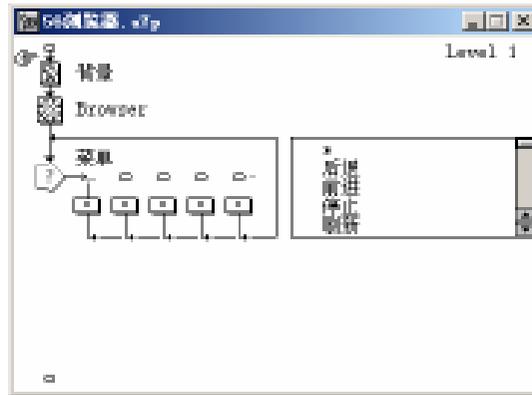


图 56.8 流程线



实例

57

进度条



实例说明

该实例建立进度条效果。

主要知识点：Authorware 变量和函数。



操作步骤

- ① 新建文件，保存为“57 进度条.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 57.1 所示。



图 57.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，在演示窗口中输入提示文本“正在拷贝文件，请稍候...”，并设置文本的属性。接着，使用绘图工具箱中的矩形工具绘出图 57.2 所示的矩形框，作为进度条的容器。

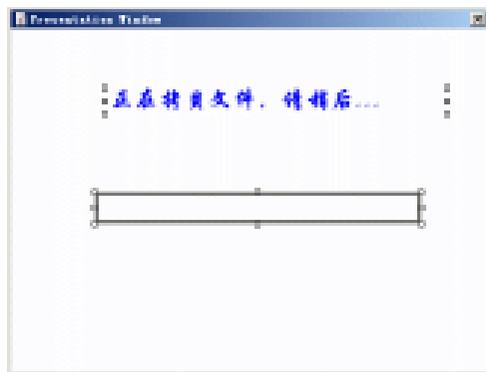


图 57.2 绘制背景

- ④ 在流程线上添加两个 Calculation (计算) 图标，分别命名为“画线”和“动态画

线”。双击“画线”图标，在代码窗口中输入下列代码：

```
SetFrame(1,RGB(255,0,0))
Line(25,100,180,100+i,180)
```



注意

如果要改变进度条的宽度，只需改变线条的宽度即可，也就是说，改变上述语句 Line 中的数值 25。

⑤ 双击“动态画线”图标，在代码窗口中输入代码，如图 57.3 所示。

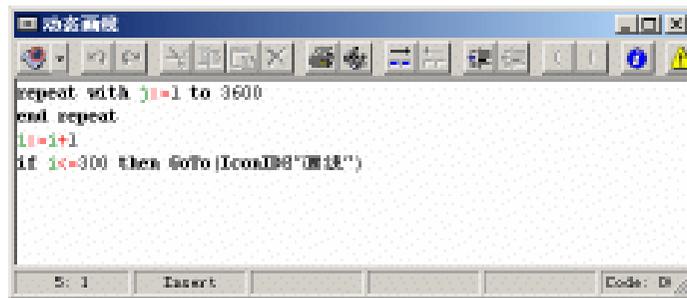


图 57.3 输入代码

输入的代码如下：

```
repeat with j:=1 to 3600
end repeat --延时
i:=i+1
if i<=300 then GoTo(IconID@'画线') --动态画线
```

⑥ 存盘，运行，流程线如图 57.4 所示。

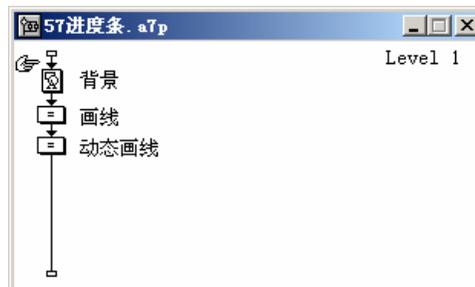


图 57.4 流程线



实例

58

插入幻灯片



实例说明

该实例建立一个插入幻灯片文件 (.ppt) 的程序，运行后，可以对插入的幻灯片进行显示、编辑和打开等操作。

主要知识点：OLE 技术。



操作步骤

- ① 新建文件，保存为“58 插入幻灯片.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 58.1 所示。

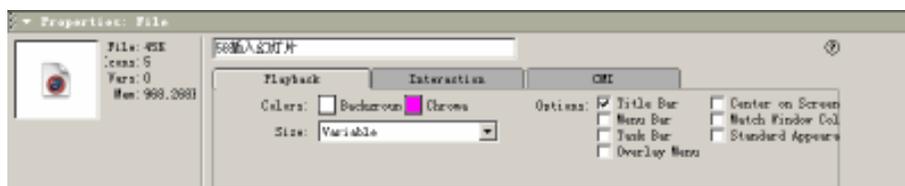


图 58.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标，命名为“幻灯片”，双击该图标，打开演示窗口，执行命令 Insert OLE Object，打开 Insert Object...对话框，选择由文件创建，如图 58.2 所示。



图 58.2 Insert Object 对话框

- ④ 单击“浏览”按钮，选择要插入的 PPT 文件，然后单击对话框中的“打开”按钮，

如图 58.3 所示。此时 Insert Object...对话框中“由文件创建”后面的文本框中出现该文件的路径，如图 58.4 所示。

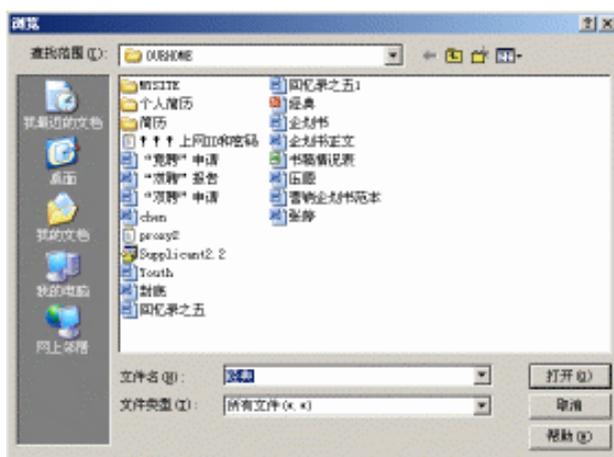


图 58.3 选择要插入的文件



图 58.4 插入幻灯片

⑤ 单击“确定”按钮后，演示窗口中出现插入的幻灯片文件，此时可以拖动控制句柄改变幻灯片的大小和位置，如图 58.5 所示。



图 58.5 演示窗口中的幻灯片



⑥ 选择幻灯片文件，执行命令“编辑” “演示文稿 OLE Object” Attributes，打开 Object Attributes 对话框，在 Activation Trigger 下拉菜单中选择 OLE 对象的激活事件 Single-click，在 Trigger Verb 下拉菜单中选择交互动作 Display(显示)，同时选择 Package as OLE Object 复选框，否则无法控制该 OLE 对象，如图 58.6 所示。

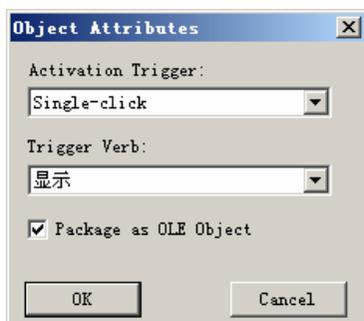


图 58.6 设置 OLE 对象属性

⑦ 在流程线上添加一个 Interaction (交互) 图标，命名为“控制 OLE 对象”，在其右侧添加三个 Calculation (计算) 图标，交互类型设置为 Button，分别命名为 Display (显示)、编辑、打开，形成图 58.7 所示的流程线。

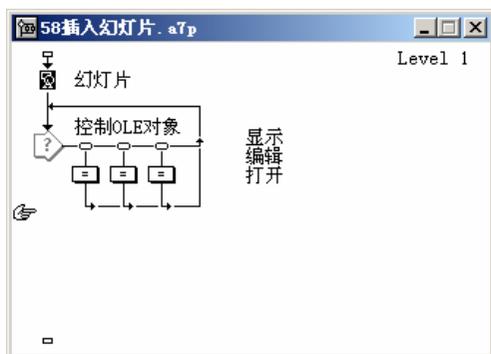


图 58.7 流程线

⑧ 双击 Display (显示) 图标，在代码窗口中输入下面的代码：

OLEDoVerb(IconID@"幻灯片", "显示")

同理，在其他两个“Calculation”(计算)图标的代码窗口中输入下列代码：

编辑：OLEDoVerb(IconID@"幻灯片", "编辑")

打开：OLEDoVerb(IconID@"幻灯片", "打开")

⑨ 重新定义按钮样式，并调整按钮和幻灯片的位置，程序画面如图 58.8 所示。

⑩ 下面运行程序，看看效果。单击 Display (显示) 按钮，将全屏播放幻灯片，因为前面的步骤 6 中设置了动作和事件，所以单击该幻灯片也可以全屏播放。

⑪ 单击“编辑”按钮，在窗口中编辑该对象。

⑫ 单击“打开”按钮，将在 Microsoft PowerPoint 中打开该幻灯片。



图 58.8 程序画面



注意

如果你的机器中没有安装 Microsoft PowerPoint, 则上述编辑和打开功能将无法实现。同时, 程序打包发行时, 要把 Authorware 安装目录下的 a6wole.dll (16 位) 和 a6wole32.dll (32 位) 的 OLE 支持库拷贝到打包文件夹下, 否则, 程序运行时会出错。



心得体会

使用 OLE (Object Linking and Embedding, 对象链接与嵌入, 简称 OLE) 技术, 在一个文件或一个程序中能够包含多种不同数据格式的数据内容。

任何一个软件的功能都不是十全十美, Authorware 也不例外, 所以 OLE 技术和 Authorware 的扩展功能接口成为扩展 Authorware 功能的得力助手。

通过引入其他应用程序设计的对象 (如 Word 文本、图形、数字电影等) 到 Authorware 文件里, 使 Authorware 中可以显示更多类型的文件。



实例

59

自定义标题栏



实例说明

该实例建立一个自定义的标题栏，包括最小化和关闭两个按钮。

主要知识点：Authorware 系统函数、自定义按钮。



操作步骤

① 新建文件，保存为“59 自定义标题栏.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Title Bar 和 Menu Bar 复选框，如图 59.1 所示。

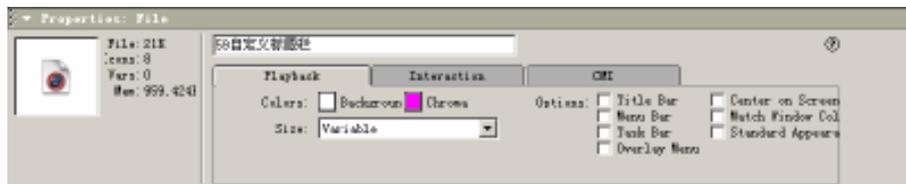


图 59.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，执行命令 File Import，导入背景图片，如图 59.2 所示。

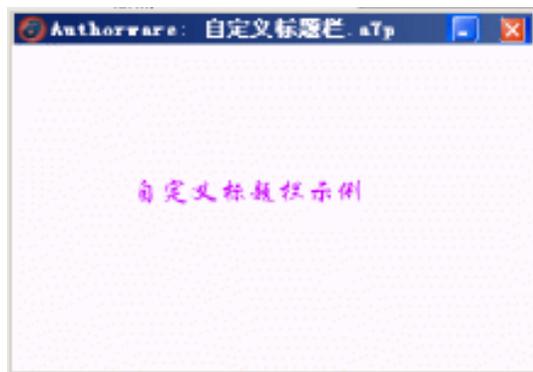


图 59.2 导入背景图片

该图片可以由图像编辑软件设计，也可由抓图软件抓下来。并把背景图像移动到窗口

顶部，作为标题栏的背景。为了防止该图片移动，在流程线上添加一个 Calculation (计算) 图标，输入下面的代码控制：

```
Movable@("背景"):=FALSE
```

④ 在流程线上添加一个 Interaction (交互) 图标，命名为“窗口控制”，在其右侧添加两个 Calculation (计算) 图标，交互类型均设置为 Button，分别命名为“最小化”和“关闭”。双击“最小化”图标上面的交互类型标记，打开其属性设置框，单击 Buttons 按钮，打开 Buttons 对话框，单击该对话框中的 Add... 按钮，添加自定义的按钮，如图 59.3 所示。同理自定义“关闭”按钮，并将两个按钮移动到适当的位置，如图 59.4 所示。

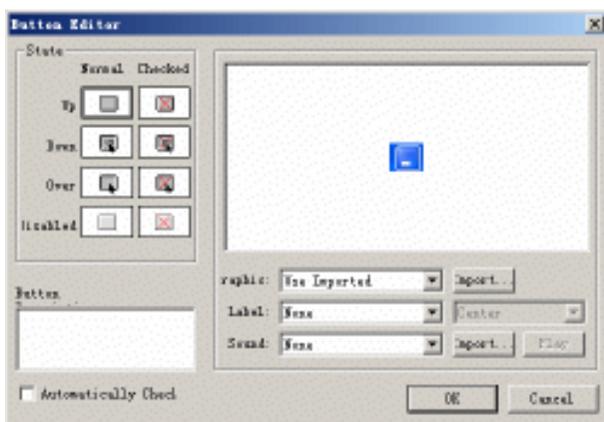


图 59.3 自定义“最小化”按钮

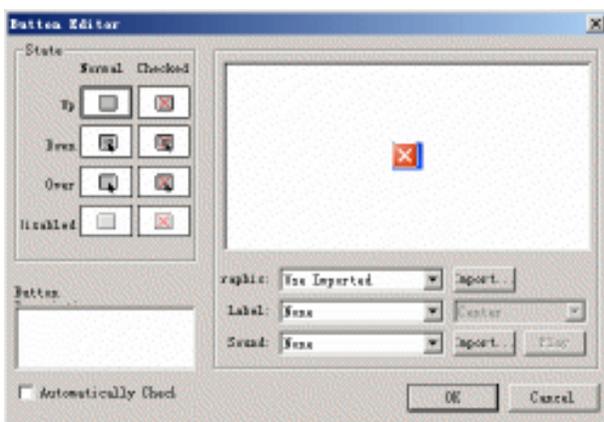


图 59.4 自定义“关闭”按钮

⑤ 将 Properties :Response 对话框切换到 Response 选项卡内，选择 Perpetual 复选框，并选择 Branch 下拉菜单中的 Return 选项，如图 59.5 所示。同理，设置“关闭”按钮的交互属性。

⑥ 双击“最小化”图标，输入下面的代码：

```
ShowWindowAsync(WindowHandle, 6)
```



图 59.5 设置按钮交互属性

关闭代码窗口时系统提示选择函数的来源，选择 Authorware 安装目录下的 winapi.u32 即可。

双击“关闭”图标，输入下面的代码：

Quit(0)

⑦ 在流程线上添加一个 Group (群组) 图标，命名为“移动窗口”，将实现窗口的移动控制。它默认的交互类型和前面的一样，即 Button，双击交互标记，打开属性对话框，在 Type 下拉菜单中选择 Hot Spot 选项，将整个标题栏设置为热区。该图标其他属性设置和前面一致，如图 59.6 所示。



图 59.6 设置群组图标交互属性

⑧ 双击 Group (群组) 图标，打开二级窗口，添加一个 Iteration (交互) 图标，命名为“移动控制”，在其右侧添加一个 Calculation (计算) 图标，交互类型为 Conditional，命名为 MouseDown，打开其属性设置框，选择 Automatic 下拉菜单中的 When True 选项，如图 59.7 所示。在 Response 选项卡内，选择 Branch 下拉菜单中的 Try Again，如图 59.8 所示。

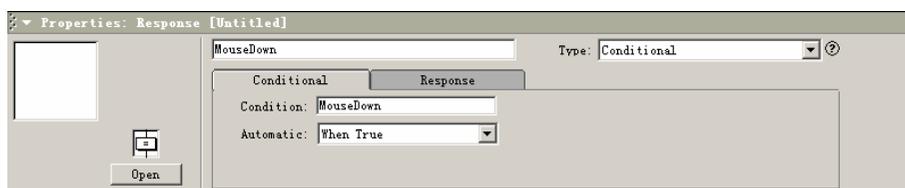


图 59.7 设置 MouseDown 交互属性 1



图 59.8 设置 MouseDown 交互属性 2

⑨ 双击 Calculation (计算) 图标, 输入下列代码:

```
ReleaseCapture()  
SendMessage( WindowHandle, 161, 2, 0)
```

关闭代码窗口时提示选择函数来源, 同样选择 winapi.u32 函数。

⑩ 存盘, 运行, 流程线如图 59.9 所示。

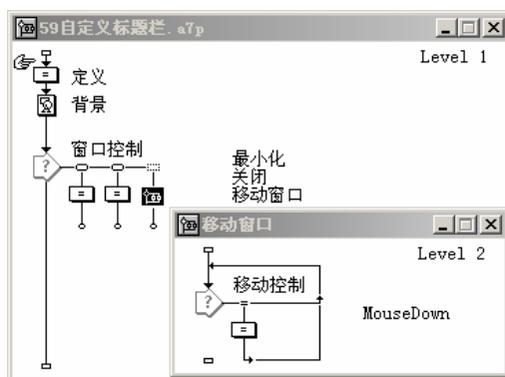


图 59.9 流程线



实例

60

文本闪烁



实例说明

该实例建立一个文本闪烁效果,文本以不同的颜色交替显示,就像霓虹灯一样。

主要知识点:判断图标、等待图标。



操作步骤

- ① 新建文件,保存为“60 文本闪烁.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 60.1 所示。



图 60.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“文本闪烁”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 60.2 所示。



图 60.2 设置背景

④ 在流程线上添加一个 Decision (判断) 图标, 命名为“文本闪烁”, 选择 Repeat 下拉菜单中的 Until Click/Keypress, 选择 Branch 下拉菜单中的 Sequentially 选项, 如图 60.3 所示。



图 60.3 设置分支图标属性

⑤ 在分支图标右侧添加 3 个群组图标, 分别命名为“显示 1”、“显示 2”和“显示 3”, 双击“显示 1”群组图标, 在二级窗口中添加一个 Display (显示) 图标和 Wait (等待) 图标, 如图 60.4 所示。



图 60.4 添加图标

⑥ 双击“显示文本”图标, 在演示窗口中输入文本“强大的多媒体设计软件 Authorware!”, 并设置文本的属性。双击 Wait (等待) 图标, 在 Timing Limit 文本框中输入“0.3”, 取消选择其他复选框, 如图 60.5 所示。



图 60.5 设置等待图标属性

⑦ 重复步骤 5 和 6, 设置“显示 2”和“显示 3”图标的内容, 将显示的文本设置为其他的颜色, 这些文本的位置要一致。

⑧ 存盘, 运行, 流程线如图 60.6 所示。

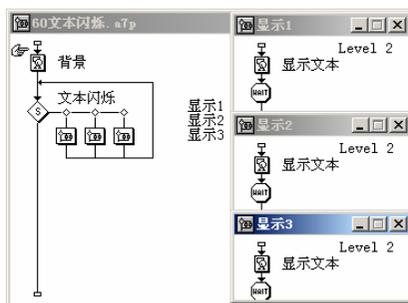


图 60.6 流程线



实例

61

控制音量



实例说明

该实例建立一个控制音量的应用程序，在播放音乐的过程中，单击界面上的“控制音量”按钮，将打开系统的声音控制面板，调解音量的大小。

主要知识点：交互图标、系统函数。



操作步骤

- ① 新建文件，保存为“61 控制音量.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 61.1 所示。

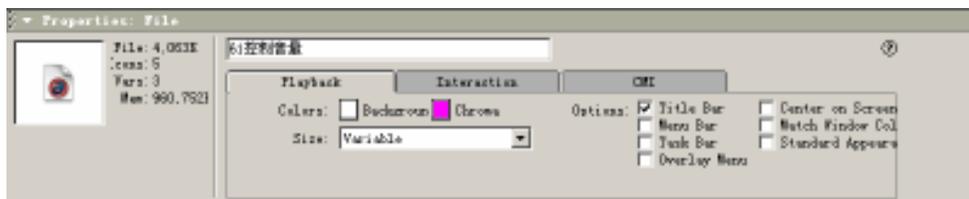


图 61.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“控制音量”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 61.2 所示。

- ④ 在流程线上添加一个 Sound (声音) 图标，命名为“二泉映月”，双击该图标，打开其属性设置框，单击 Import 按钮导入音乐，如图 61.3 所示。导入音乐后，在属性对话框中进行设置，在 Timing 选项卡内选择 Concurrency 下拉菜单中的 Perpetual，其他选项默认即可，如图 61.4 所示。



图 61.2 设置背景

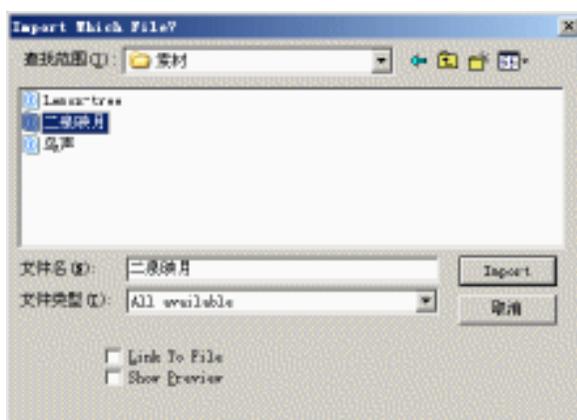


图 61.3 导入音乐

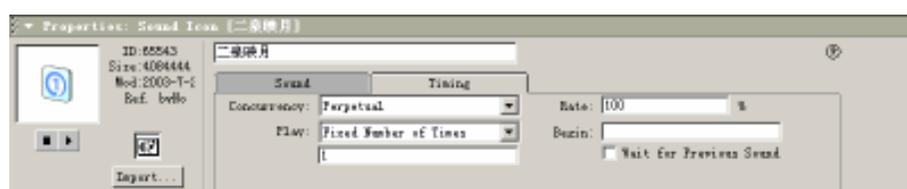


图 61.4 设置音乐文件属性

⑤ 在流程线上添加一个 Interaction (交互) 图标, 命名为“控制”, 在其右侧添加 2 个 Calculation (计算) 图标, 交互类型均设置为 Button, 分别命名为“控制音量”和“退出”。双击“控制音量”图标上面的交互类型标记, 打开其属性设置框, 单击 Buttons 按钮, 打开 Buttons 对话框, 设置按钮的样式, 如图 61.5 所示。

⑥ 单击属性对话框中的 Cursor 按钮, 打开 Cursors 对话框, 设置光标的样式, 如图 61.6 所示。重复步骤 5 和 6 设置“退出”按钮的属性。

⑦ 双击“控制音量”图标, 打开代码窗口, 输入代码, 在 Windows XP 中, sndvol32.exe 位于 C:\WINDOWS\system32 目录下, 如图 61.7 所示。



图 61.5 设置按钮样式

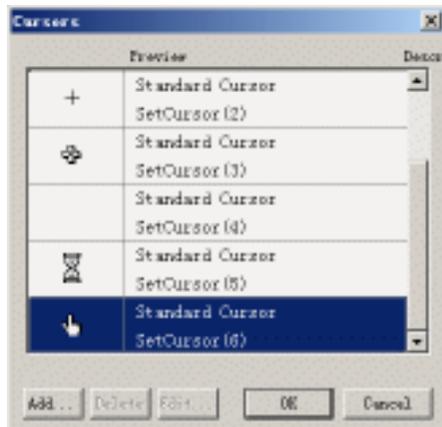


图 61.6 设置光标属性

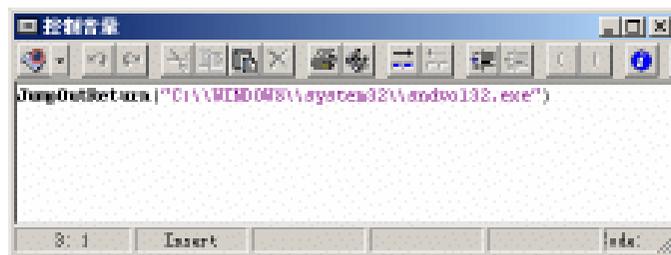


图 61.7 输入代码

⑧ 双击“退出”图标，输入代码 Quit(0)，如图 61.8 所示。

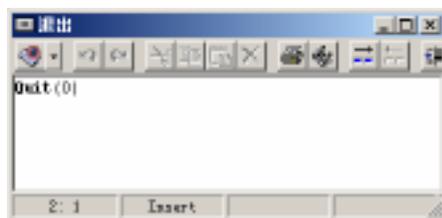


图 61.8 输入代码

9 存盘，流程线如图 61.9 所示。

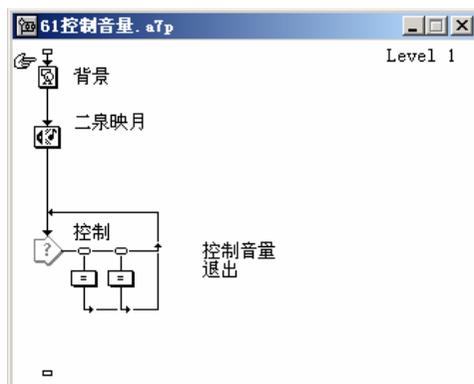


图 61.9 流程线

10 运行，单击程序画面上的“控制音量”按钮，打开声音控制面板，如图 61.10 所示。

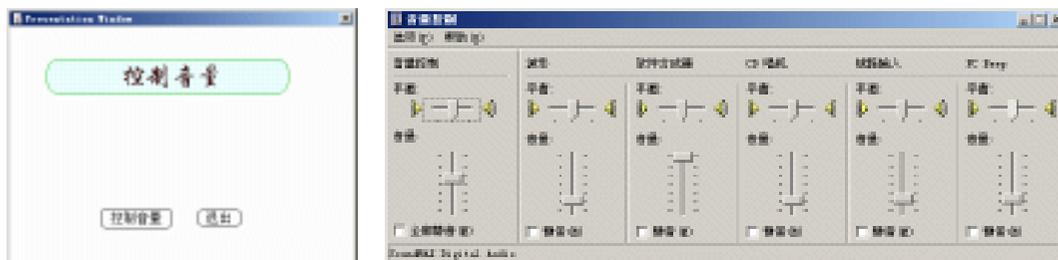


图 61.10 程序运行效果



实例

62

相册



实例说明

该实例建立一个相册应用程序,在相册中有多个相片可以上下翻页,同时显示当前相片的说明,同时还有音乐的伴奏

主要知识点:框架图标和导航。



操作步骤

- ① 新建文件,保存为“62 相册.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 62.1 所示。

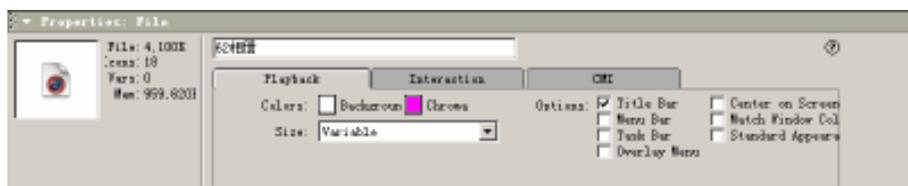


图 62.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“相册”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 62.2 所示。

④ 在流程线上添加一个 Sound (声音) 图标,命名为“背景音乐”,双击该图标,打开其属性对话框,单击 Import 按钮导入音乐,如图 62.3 所示。导入音乐后,在属性对话框中进行设置,在 Timing 选项卡内选择 Concurrency 下拉菜单中的 Prepetual,其他选项默认即可,如图 62.4 所示。

⑤ 在流程线上添加一个 Framework (框架) 图标,命名为“相册”,双击打开,删除其中不需要的内容,并将英文名称改为中文,如图 62.5 所示。



图 62.2 设置背景

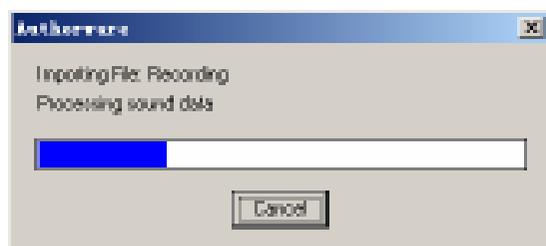


图 62.3 导入音乐

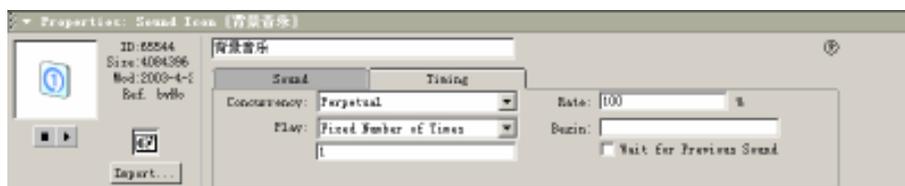


图 62.4 设置音乐文件属性

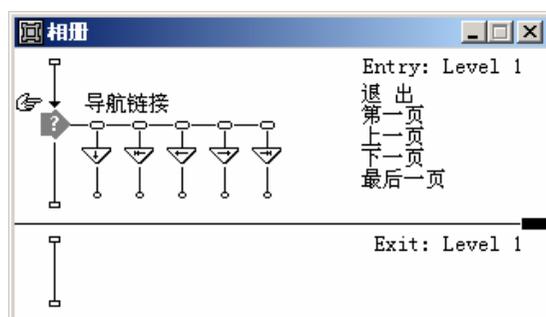


图 62.5 设置框架图标内容

⑥ 在 Framework (框架) 图标的下面添加 3 个 Group (群组) 图标, 分别命名为“相片 1”、“相片 2”、“相片 3”, 在 Group (群组) 图标中分别添加 2 个 Display (显示) 图标, 命名为“1”、“介绍 1”, 如图 62.6 所示。



图 62.6 在群组图标内添加图标



图 62.7 设置“1”图标内容

⑦ 双击“1” Display (显示) 图标, 导入相片 1, 并在相片下面输入导航信息“第 {CurrentPageNum} 张, 共 {PageCount} 张”, 并设置字体属性, 如图 62.7 所示。

⑧ 双击“介绍 1”图标, 输入相片 1 的介绍性文字, 并调整好它和相片的位置关系。如果输入的文本较多, 选择 Text Scrolling 命令, 如图 62.8 所示。

⑨ 重复步骤 7 和 8 设置其他 2 个 Group (群组) 图标的内容。

⑩ 存盘, 运行, 程序流程图如图 62.9 所示。



图 62.8 输入相片介绍文本



图 62.9 流程线

实例

63

“关于”对话框



实例说明

该实例建立一个调用“关于”对话框的应用程序，类似于 Windows 和其他软件的“关于”对话框。

主要知识点：调用 Shell32.dll 函数。



操作步骤

- ① 新建文件，保存为“63 关于.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 63.1 所示。



图 63.1 设置文件属性

- ③ 执行命令 Windows Functions，打开 Functions 对话框，在 Category 下拉菜单中选择该程序的文件名。
- ④ 单击 Load 按钮，打开 Load Functions 对话框，在 Windows 系统目录下的 System 文件夹中找到 Shell32.dll，单击“打开”按钮，如图 63.2 所示。



图 63.2 载入函数



⑤ 弹出对话框，在 Function Name 下面的文本框中输入 ShellAboutA，在 Arguments 下方的文本框中输入 “ long,string,string,long ”，在 Return 下方的输入框中输入 long，然后单击 Load 按钮，如果函数被顺利导入，在对话框的下方给出相应的提示 Successfully loaded "ShellAboutA()"，如图 63.3 所示。单击 Done 按钮，如果再次出现导入函数的对话框，单击“取消”按钮即可。

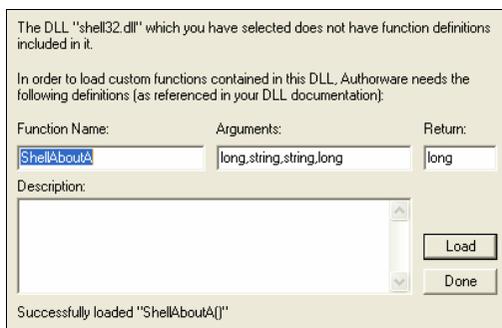


图 63.3 成功导入函数

⑥ 在 Functions 对话框左侧的列表中出现刚才导入的函数，在流程线上添加一个 Calculation（计算）图标，双击该图标，输入代码，如图 63.4 所示。

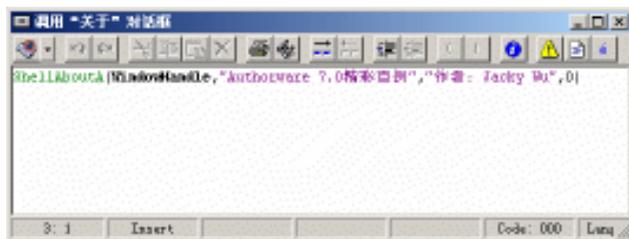


图 63.4 输入代码

⑦ 存盘，运行，程序及其流程线如图 63.5 所示。



图 63.5 流程线

实例

64

音乐播放器



实例说明

该实例建立一个音乐播放器程序,通过按钮控制乐曲的选择、播放等。

主要知识点:交互图标、函数调用。



操作步骤

- ① 新建文件,保存为“64 音乐播放器.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 64.1 所示。



图 64.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“音乐播放器”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 64.2 所示。

④ 在流程线上添加一个 Interaction (交互) 图标,命名为“按钮”,在其右侧添加 3 个 Calculation (计算) 图标,交互类型设置为 Button,分别命名为“重放”、“选曲”和“退出”,分别设置 3 个按钮的交互属性,在 Response 选项卡内选择 Perpetual 复选框,如图 64.3 所示。在这 3 个 Calculation (计算) 窗口中分别输入下列代码:

```
重放: GoTo(IconID@"play");
选曲: GoTo(IconID@"openfile");
退出: Quit()。
```



图 64.2 设置背景



图 64.3 设置按钮属性

⑤ 打开知识对象窗口,在流程线上添加一个 Open File Dialogue 知识对象,命名为“打开文件”,打开的向导保持默认设置,选择打开文件类型时,选择 Audio 复选框即可,如图 64.4 所示。

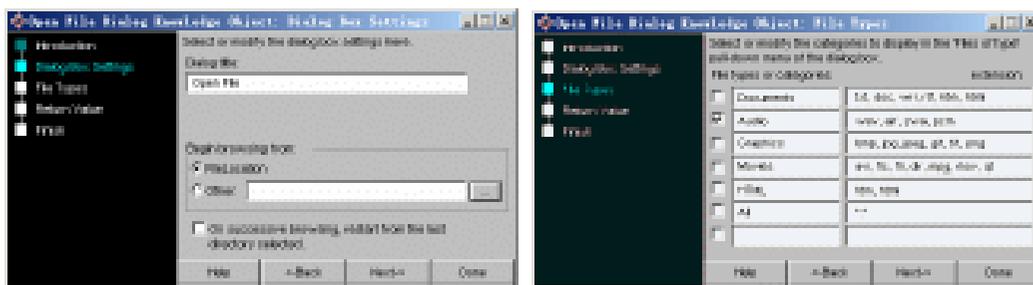


图 64.4 知识对象设置向导

⑥ 导入 A5wmme.u32,引入外部函数 WavePlay、WavePause、WaveResume 三个函数。

⑦ 拖入一个计算图标,命名为 Play,写入代码 WavePlay(wzFileChosen),如图 64.5 所示。

⑧ 再拖入一个计算图标,命名为 Resume。双击写入:WaveResume()。

⑨ 在下面拖入一个等待图标,双击它,选择 Mouse click 或 Key press,其他选项默认即可,如图 64.6 所示。

⑩ 在流程线上再添加一个计算图标,起名为 Pause,双击写入:WavePause()。

⑪ 再添加一个等待图标,同上面第一个等待图标一样设置。

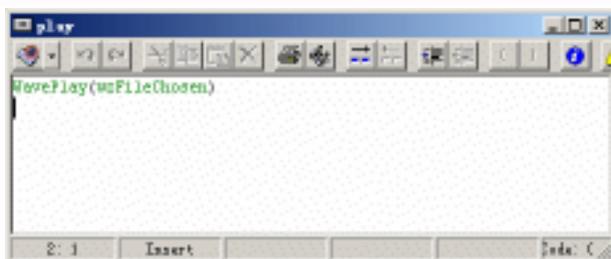


图 64.5 输入代码



图 64.6 设置等待图标属性

- ⑫ 最后，添加一个计算图标，起名为 Goto，双击它，写入：GoTo(IconID@"resume")。
- ⑬ 存盘，运行，流程线如图 64.7 所示。



图 64.7 流程线



实例

65

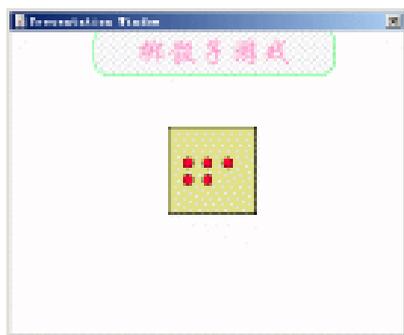
掷骰子游戏



实例说明

该实例建立一个掷骰子游戏,程序运行后,模拟骰子效果,随机显示骰子的点数,单击鼠标或按键后停止。

主要知识点:判断图标。



操作步骤

- ① 新建文件,保存为“65 掷骰子游戏.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 65.1 所示。

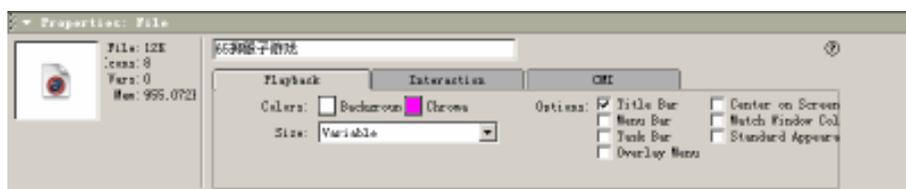


图 65.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“掷骰子游戏”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 65.2 所示。

④ 在流程线上添加一个 Decisor (判断) 图标,命名为“掷骰子”,在其右侧添加 6 个 Display (显示) 图标,分别命名为“6、5、4、...”。

⑤ 双击 Decision (判断) 图标,打开其属性对话框,选择 Repeat 下拉菜单中的 Until Click/Keypress,选择 Branch 下拉菜单中的 Randomly to Any Path,如图 65.3 所示。

⑥ 双击“6”显示图标,在演示窗口中导入实现准备好的图片,也可以使用 Authorware 的绘图工具绘制,如图 65.4 所示。



图 65.2 设置背景



图 65.3 设置分支图标属性

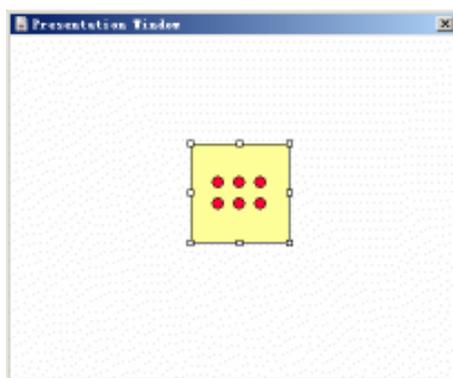


图 65.4 导入骰子图片

- ⑦ 同理在其他演示窗口中导入对应点数的骰子图片。
- ⑧ 存盘，运行，流程线如图 65.5 所示。

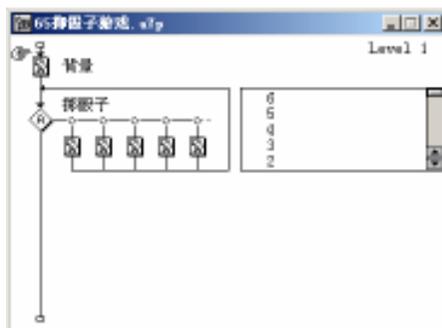


图 65.5 流程线



实例

66

显示文件路径



实例说明

该实例建立一个显示文件路径的应用程序，程序运行后，打开对话框浏览目录，打开目录后显示其路径。

主要知识点：知识对象。



操作步骤

- ① 新建文件，保存为“66 显示文件路径.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 66.1 所示。



图 66.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“显示文件路径”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 66.2 所示。

④ 执行命令 Windows Knowledge Objects，打开 Knowledge Objects (知识对象) 对话框，在 Category 下拉菜单中选择 All，并在下面的列表中选择 Browser Folder Dialog 对象，如图 66.3 所示。

⑤ 将选择的知識对象拖放到流程线上。打开知识对象设计向导，第一个对话框是介绍性的，不予理睬，单击 Next 按钮进入下一个对话框，在两个文本框中分别输入标题和提示信息，并选择 FileLocation 单选按钮，如图 66.4 所示。



图 66.2 设置背景



图 66.3 选择知识对象

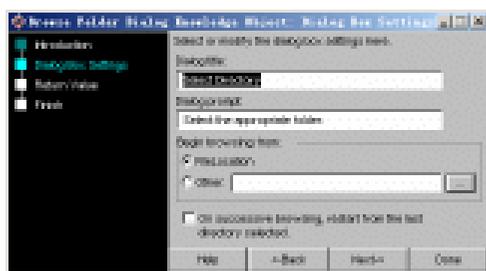


图 66.4 对话框设置

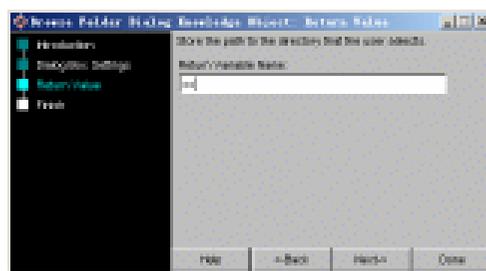


图 66.5 设置返回值

⑥ 单击 Next 按钮进入下一个对话框，在文本框中输入“=x”，将用户选择的结果赋予变量 x，如图 66.5 所示。

⑦ 单击 Done 按钮退出向导，在流程线上出现知识对象图标，重新命名为“浏览文件夹对话框”。

⑧ 在流程线上添加一个 Display（显示）图标，命名为“显示路径”，双击打开演示窗口，输入文本“文件的路径是{x}”，并设置文本的字体、大小、颜色等属性，如图 66.6 所示。



图 66.6 设置显示文本

⑨ 存盘，运行，流程线如图 66.7 所示。



图 66.7 流程线

实例

67

屏保程序(1)



实例说明

该实例建立一个屏幕保护程序,在设定的时间之内用户如果不操作计算机则运行该程序,当用户激活计算机时该程序自行退出。

主要知识点:判断图标。



操作步骤

- ① 新建文件,保存为“67 屏幕保护程序.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 800×600 SVGA,取消选择 Title Bar 和 Menu Bar 复选框,如图 67.1 所示。

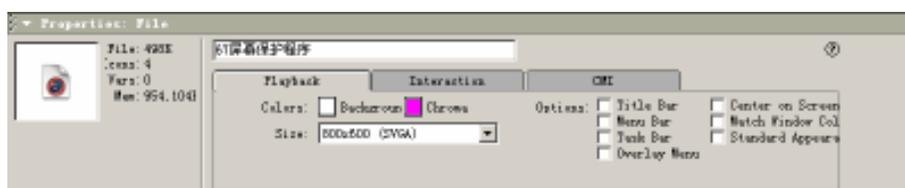


图 67.1 设置文件属性

- ③ 在流程线上添加一个 Decision (判断) 图标,命名为“屏幕保护”,在其右侧添加 2 个 Display (显示) 图标,分别命名为“图 1”和“图 2”,双击 Interaction (交互) 图标,打开其属性对话框,Repeat 设置为 Until Click/keypress, Branch 设置为 Sequentially,如图 67.2 所示。



图 67.2 设置分支图标属性

- ④ 双击“图 1”显示图标上面的分支标记,打开其属性对话框,选择 Erase 下拉菜单中的 Before Next Selection 选项,如图 67.3 所示。

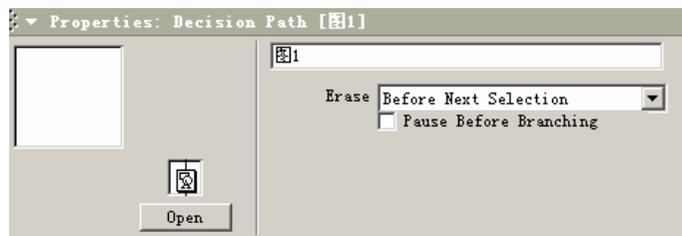


图 67.3 设置分支属性

⑤ 双击打开“图 1”显示图标，执行命令 File Import，导入图片，如图 67.4 所示。双击导入的图片，打开图片属性对话框，切换到 Layout, Display 设置为 Scaled, Position 和 Size 分别设置为“0, 0”和“800, 600”，使得图片充满整个屏幕，如图 67.5 所示。注意此时的分辨率是“800 × 600”。

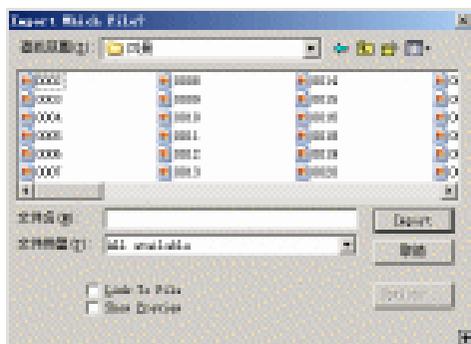


图 67.4 导入图片



图 67.5 设置图片属性

⑥ 右键单击“图 1”图标，选择命令 Properties，打开属性对话框，单击 Transition 后面的按钮，设置图片的过渡方式，如图 67.6 所示。

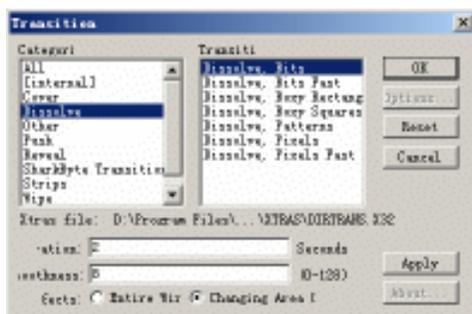


图 67.6 设置图片的过渡方式

- ⑦ 重复步骤 4~6，设置“图 2”的属性 and 内容。
- ⑧ 在流程图中添加一个 Calculation（计算）图标，命名为“退出”，双击输入代码 Quit(0)，如图 67.7 所示。

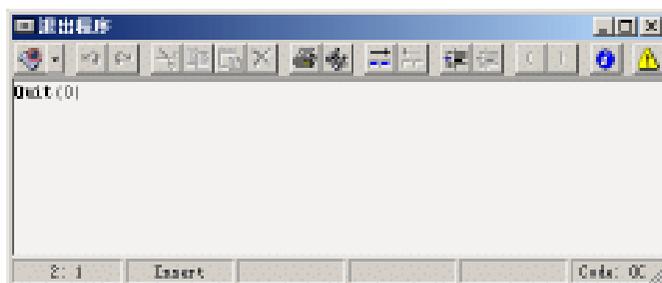


图 67.7 输入代码

- ⑨ 存盘，流程线如图 67.8 所示。

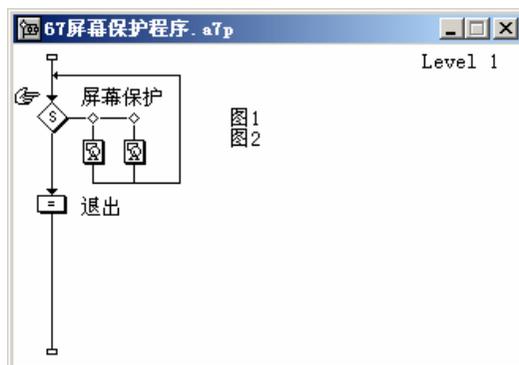


图 67.8 流程线



实例

68

屏保程序 (2)



实例说明

该实例将上例创建的屏保程序进行打包，并设置为系统的屏幕保护程序。

主要知识点：文件打包。



操作步骤

① 接上例，程序设计完毕后，执行命令 File Publish Package，打开 Package File 对话框，在下拉菜单中选择 For Windows 9x and NT variant，并选择 Package All Libraries internal 复选框，然后单击 Save File(s) & Package 按钮，如图 68.1 所示。

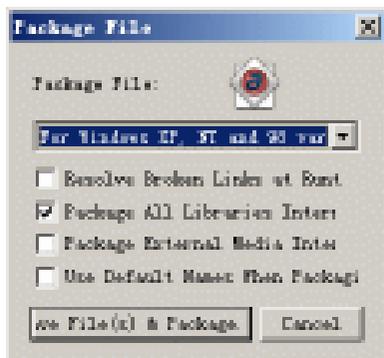


图 68.1 打包文件

② 接着，打开 Package File As 对话框，选择保存文件的目录，在“保存类型”下拉菜单中选择 All Files，将其保存为“屏幕保护程序.scr”（.scr 是 Windows 的屏幕保护文件）单击“保存”按钮，如图 68.2 所示。

③ 将“屏幕保护程序.scr”和 extras 文件夹一同拷贝到 C:\WINDOWS 目录下。

④ 返回系统桌面，右键单击，选择“属性”命令，打开“显示属性”对话框，切换到“屏幕保护程序”栏，在下拉菜单中找到刚才保存的屏保程序，在下面的 Wait（等待）栏中设定时间，

⑤ 单击“设定”按钮即可，如图 68.3 所示。



图 68.2 保存文件



图 68.3 设置屏幕保护程序



实例

69

复选框



实例说明

该实例建立一个复选框效果,界面上有多个选项供用户选择,用户可以一次选择多个选项。

主要知识点:交互图标、按钮属性。



操作步骤

- ① 新建文件,保存为“69 复选框.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 69.1 所示。



图 69.1 设置文件属性

- ③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“复选框示例”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 69.2 所示。



图 69.2 设置背景

④ 在流程线上添加一个 Display (显示) 图标, 命名为“题目”, 双击打开演示窗口, 输入题目“下列哪些是欧洲城市?”并设置文本的属性, 如图 69.3 所示。



图 69.3 添加题目

⑤ 在流程线上添加一个 Interaction (交互) 图标, 命名为“选项”, 在其右侧添加 4 个 Group (群组) 图标, 交互类型设置为 Button, 分别命名为“伦敦”、“东京”、“华盛顿”和“巴黎”,

⑥ 双击 Group (群组) 图标上面的交互类型标记, 打开交互属性对话框, Branch 设置为 Try Again, Status 设置为 Not Judged, 如图 69.4 所示, 其他几个按钮的交互属性相同。

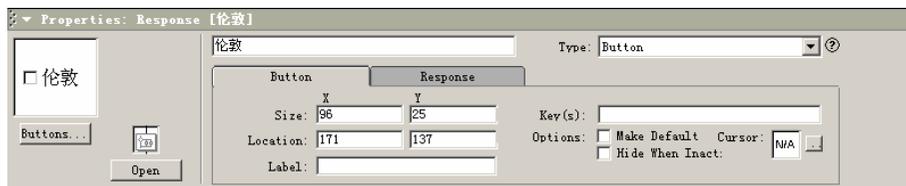


图 69.4 设置交互属性

⑦ 单击按钮交互属性对话框中的 Buttons 按钮, 打开 Buttons 对话框, 选择复选框形式, 如图 69.5 所示。



图 69.5 设置按钮属性



- 8 同理设置其他几个按钮属性。
- 9 双击 Interaction (交互) 图标, 按住 Shift 键同时选择 4 个按钮, 然后选择命令 Text Align, 打开 Align 面板, 对齐 4 个按钮, 如图 69.6 所示。

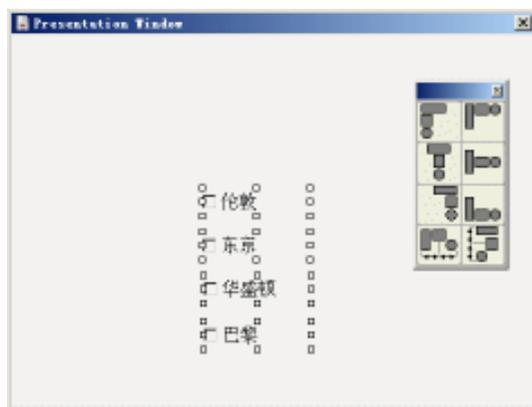


图 69.6 对齐按钮

- 10 存盘, 运行, 程序流程线如图 69.7 所示。

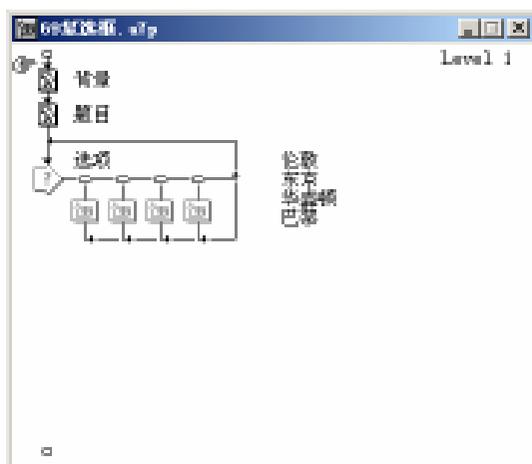


图 69.7 设置文件属性

实例

70

超文本链接



实例说明

该实例建立一个超文本链接，单击界面上的文本，将打开与之对应的图片。

主要知识点：框架图标、链接导航。



操作步骤

- ① 新建文件，保存为“70 超文本链接.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 70.1 所示。

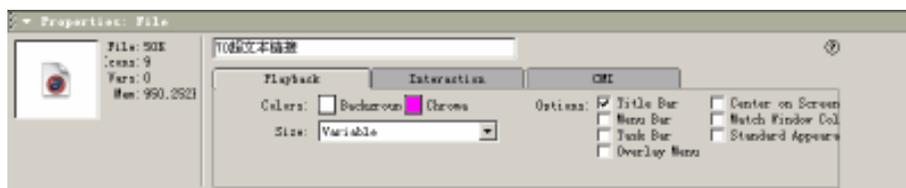


图 70.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“链接示例”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 70.2 所示。

④ 在流程线上添加一个 Group (群组) 图标，命名为“返回”，双击打开二级窗口，添加一个 Interaction (交互) 图标，并在其右侧添加一个 Calculation (计算) 图标，交互类型设置为 Button，双击交互类型标记，打开属性对话框，在 Button 选项卡内填入 Size 和 Location 的数值，选择 Hide When Inactive 复选框，设置光标形状为小手状；在 Response 选项卡内选择 Perpetual 复选框，如图 70.3 和图 70.4 所示。

⑤ 双击 Calculation (计算) 图标，输入代码 GoTo(IconID@"主页面")，此二级窗口的流程图如图 70.4 所示。

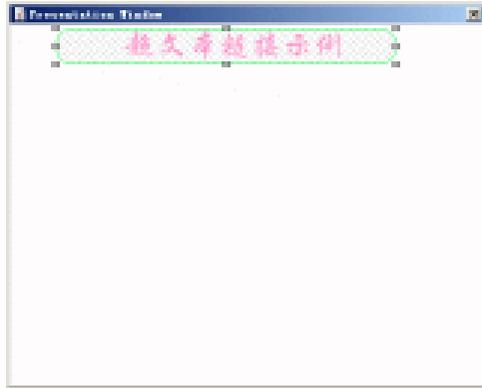


图 70.2 设置背景

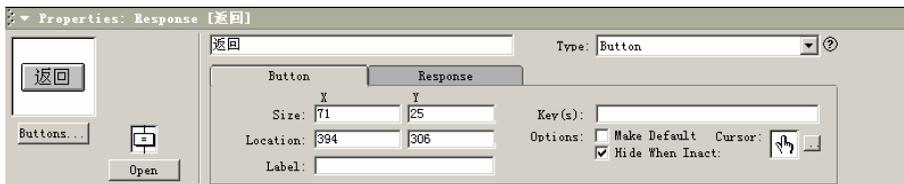


图 70.3 设置交互属性的 Button 选项卡

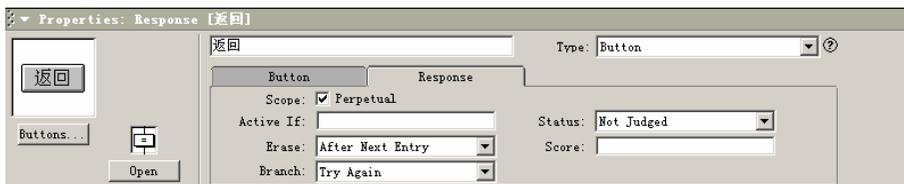


图 70.4 设置交互属性的 Response 选项卡

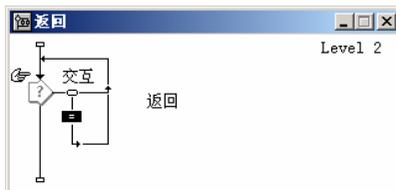


图 70.5 “返回”流程图

⑥ 回到主程序，添加一个 Framework（框架）图标，命名为“主页面”，双击打开该图标，删除其中的所有图标来自定义链接，如图 70.6 所示。



图 70.6 框架图标内容为空

⑦ 在主程序的 Framework (框架) 图标下面添加 4 个 Display (显示) 图标, 分别命名为“目录”、“苹果”、“蘑菇”和“红叶”, 如图 70.7 所示。双击“目录”图标, 打开演示窗口, 分别输入文本 (注意, 不是一个文本的 3 行, 而是 3 行文本), 如图 70.8 所示。

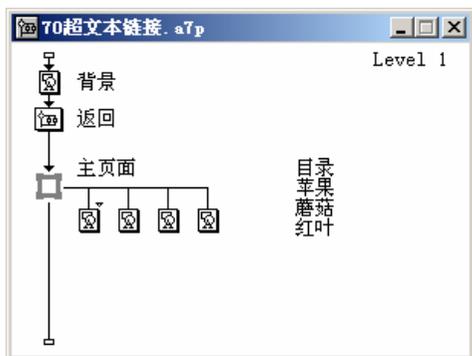


图 70.7 添加显示图标



图 70.8 输入文本

⑧ 执行命令 Text Define Styles, 打开 Define Styles 对话框, 新建一个“超文本链接”的样式, 设置好字体相关属性, 选择 Single Click 单选按钮, 并选择 Cursor 复选框, 如图 70.9 所示。

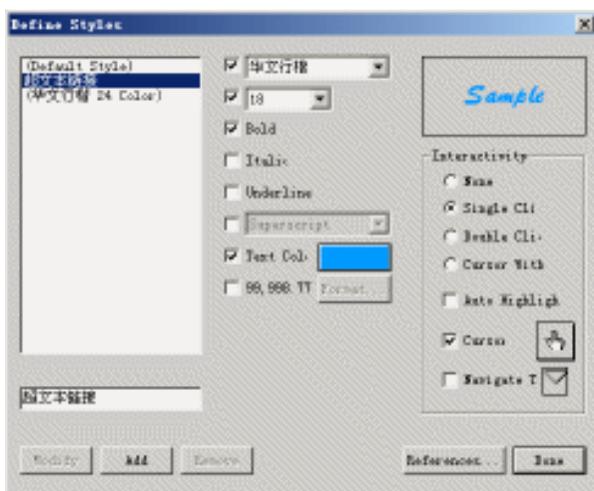


图 70.9 自定义文本样式

⑨ 在“目录”演示窗口中, 选中文本“苹果”, 执行命令 Text Apply Style, 给文本套用样式, 如图 70.10 所示。

⑩ 双击“苹果”显示图标, 打开演示窗口, 导入苹果图片, 如图 70.11 所示, 调整图片的大小。在“苹果”显示图标的属性对话框, 单击 Transition 后面的按钮, 在 Transition 对话框中设置图片的过渡方式, 如图 70.12 所示。

⑪ 此时, 运行程序, 就会出现演示窗口, 如图 70.13 所示。单击“苹果”文本, 在如图 70.14 所示的导航属性对话框中, 选择 Jump to Page 单选按钮, 并选择链接对象。



图 70.10 套用文本样式

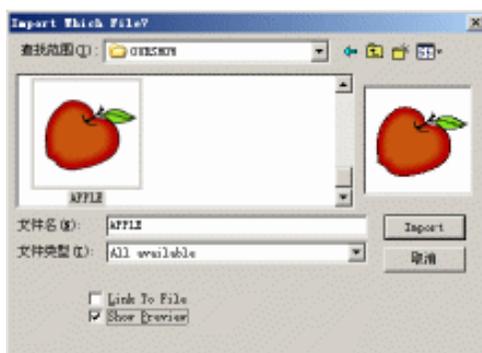


图 70.11 导入图片



图 70.12 设置图片过渡方式

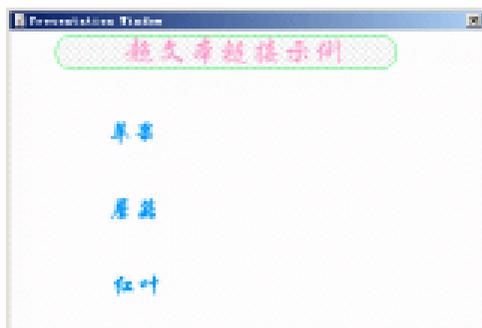


图 70.13 程序演示窗口

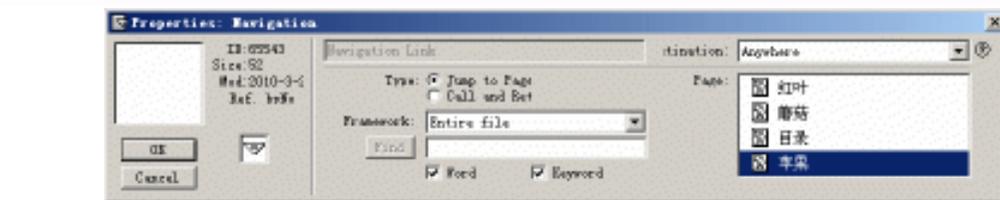


图 70.14 导航属性对话框

- ⑫ 重复步骤 9~11，设置其他两个文本的属性和超链接。
- ⑬ 存盘，运行，流程线如图 70.7 所示。



实例

71

页面查询 (1)



实例说明

该实例建立一个页面查询的应用程序，类似于其他软件的查找功能，输入关键字，可以查到与之相关的内容。

主要知识点：框架图标、交互结构、导航结构。



操作步骤

- ① 新建文件，保存为“71 页面查询 (1) .a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 71.1 所示。

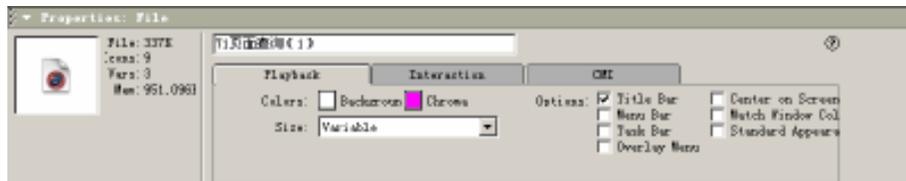


图 71.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“页面查询”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 71.2 所示。

④ 在流程线上添加一个 Interaction (交互) 图标，命名为“交互”，在其右侧添加一个 Navigation (导航) 图标和 Calculation (计算) 图标，交互类型均设置为 Button，分别命名为“查找”和“退出”。双击“查找”图标上面的交互类型标记，打开属性对话框，将属性框切换到 Button 选项卡，在 Cursor 栏设置光标形状为小手，如图 71.3 所示。在 Response 选项卡内选择 Perpetual 复选框，其他选项默认，如图 71.4 所示。



图 71.2 设置背景

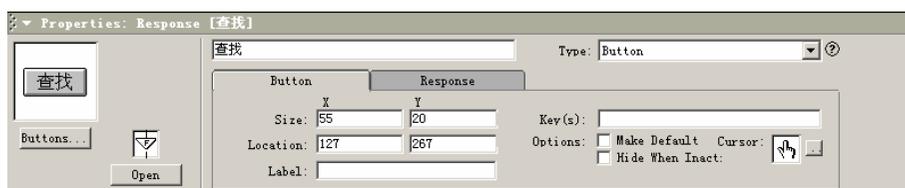


图 71.3 设置光标属性



图 71.4 设置按钮属性

⑤ 双击“查找”导航图标，打开属性对话框，选择 Destination 下拉菜单中的 Search 选项，其他选项默认，如图 71.5 所示。同理，设置“退出”按钮的属性。

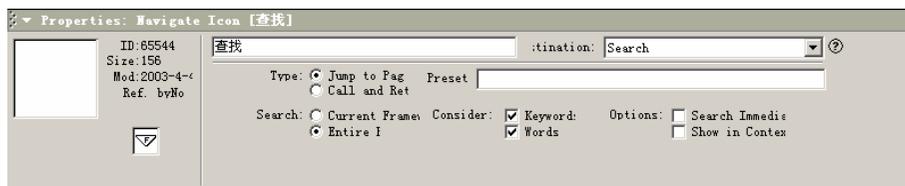


图 71.5 设置“查找”导航图标属性

⑥ 双击“退出”计算图标，输入代码 Quit(0)，单击此按钮将退出程序，如图 71.6 所示。

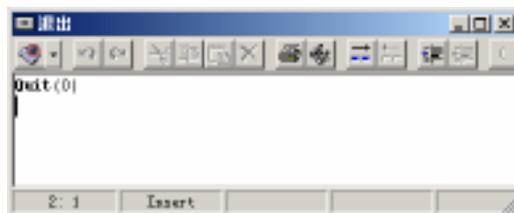


图 71.6 属性代码



⑦ 此时的流程线如图 71.7 所示。

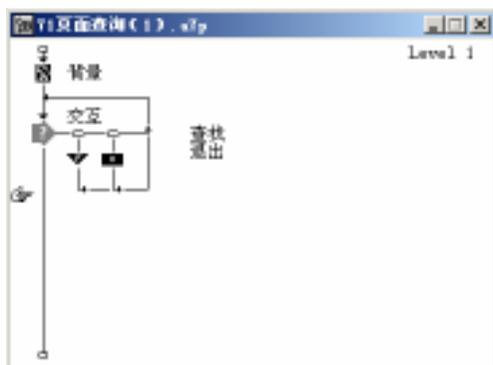


图 71.7 流程线

⑧ 在流程线上添加一个 Framework (框架) 图标, 命名为“框架”, 双击打开, 删除其中所有图标, 并在主页面的 Framework (框架) 图标右侧添加 4 个 Display (显示) 图标, 分别命名为“第一篇”、“第二篇”、“第三篇”和“第四篇”, 在各自的窗口中输入内容, 并设置字体的属性, 如图 71.8 所示。

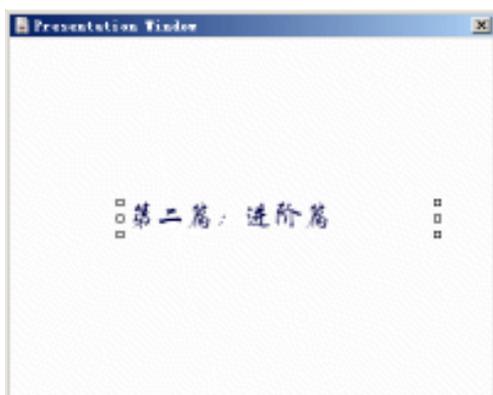


图 71.8 在显示图标中输入内容

⑨ 存盘, 运行, 程序最终流程线如图 71.9 所示。

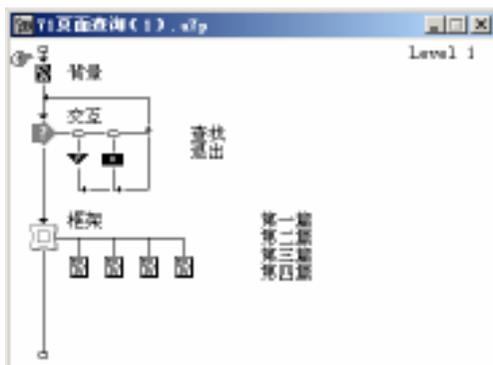


图 71.9 程序最终流程线

实例

72

页面查询 (2)



实例说明

该实例对上面例子中建立的程序进行修改,修改程序界面,并将查询到的内容用红色字体显示。

主要知识点: Navigation Setup。



操作步骤

① 执行命令 File Open, 打开“71 页面查询 (1) .a7p”, 执行命令 File Save as, 将其保存为“72 页面查询 (2) .a7p”, 如图 72.1 所示。



图 72.1 另存文件

② 执行命令 Modify File Navigation Setup 打开 Navigation Setup 对话框 如图 72.2 所示, 该对话框中选项的含义如下:

- Window Title: 设置窗口标题;
- Word Prompt: 设置关键字标题;
- List Heading: 设置列表标题;
- Find Button: 设置查找按钮;
- Go to Button: 设置跳转按钮;
- Cancel Button: 设置取消按钮;
- Pause/Resume Button: 设置暂停/继续按钮;



- Highlight Found Word：设置文字高亮显示颜色；
- Maximum Pages to：设置可查询的最大页面数。



图 72.2 Navigation Setup 对话框

③ 了解对话框的用法之后，对它进行设置，设置高亮显示颜色为红色，并选择 Close When Page is Selected 复选框，如图 72.3 所示。



图 72.3 设置 Navigation Setup 对话框

④ 存盘，运行，可以看到，程序的界面完全是中文的，而且查找到的文字用红色高亮显示，如图 72.4 所示。



图 72.4 程序运行效果

实例

73

标尺程序



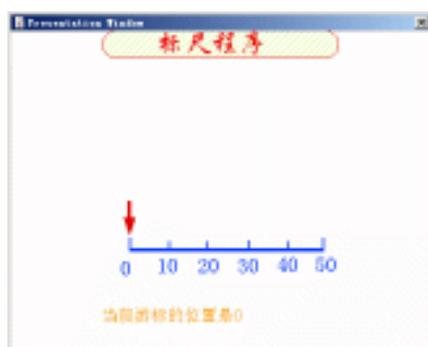
实例说明

该实例建立一个标尺程序,在标尺上拖动游标,显示游标在标尺中的位置,单击鼠标或按键,程序退出。

主要知识点:显示图标、等待图标、群组图标、计算图标。



操作步骤



① 新建文件,保存为“73 标尺.a7p”。

② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 73.1 所示。

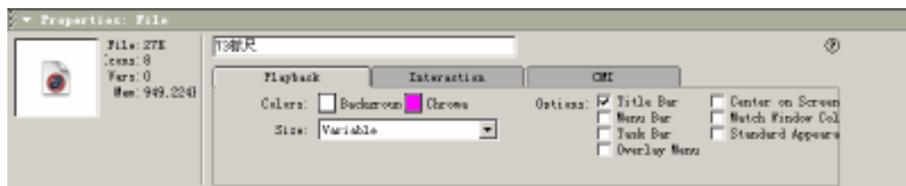


图 73.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域,双击圆角矩形工具 ,打开填充面板,设置填充方案,并双击椭圆形工具 ,打开调色板,为填充图案着色。然后,使用文本工具  输入标题“标尺程序”,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,同时设置字体颜色,双击箭头工具 ,设置文本的显示模式为 Transparent,如图 73.2 所示。

④ 在流程线上添加一个 Group (群组) 图标,命名为“游标刻度”,双击打开二级窗口,添加如图 73.3 所示的图标。

⑤ 双击“定义”图标,打开代码窗口,输入代码 `Movable@"标尺":=0`,确保标尺不移动,如图 73.4 所示。



图 73.2 设置背景



图 73.3 添加图标

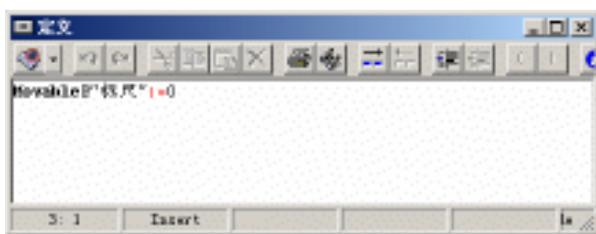


图 73.4 添加代码

⑥ 双击“标尺”图标，打开演示窗口，使用绘图工具箱中的工具绘制图 73.5 所示的标尺，并输入标尺刻度。

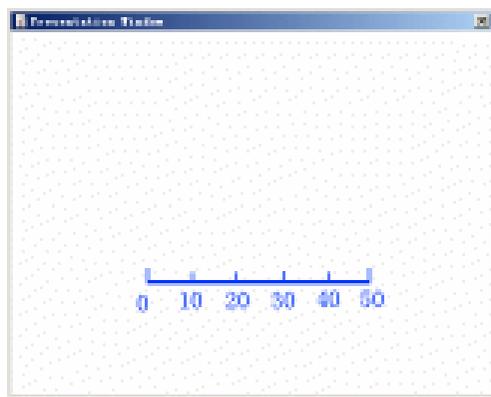


图 73.5 绘制标尺

⑦ 双击“游标”图标，使用绘图工具箱中的直线工具绘制游标，并设置游标为红色，如图 73.6 所示。

⑧ 按住 Ctrl 键双击“游标”图标，打开其属性对话框，在 Positioning 和 Movable 下拉菜单中均选择 On Path 选项，在 Base 和 Initial 中输入“0”，在 End 文本框中输入“50”，如图 73.7 所示。此时，在窗口中拖动游标到适当的位置，出现控制线，控制线的首尾和标尺首尾相对应，如图 73.8 所示。

⑨ 双击“位置显示”图标，输入文本“当前游标的位置是 PathPosition@“游标””，

并设置文本的属性，如图 73.9 所示。

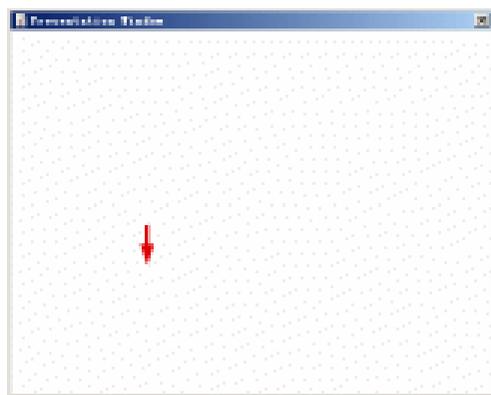


图 73.6 绘制游标

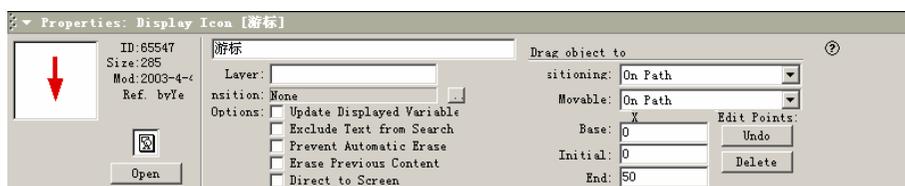


图 73.7 设置游标属性

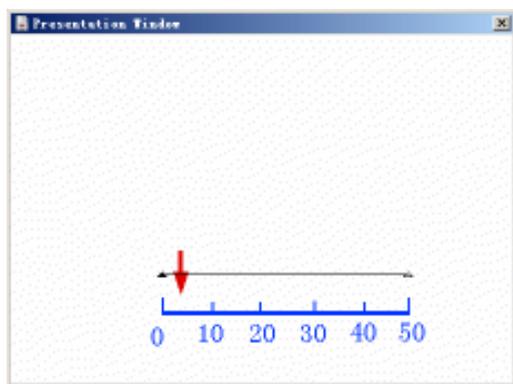


图 73.8 设置游标位置

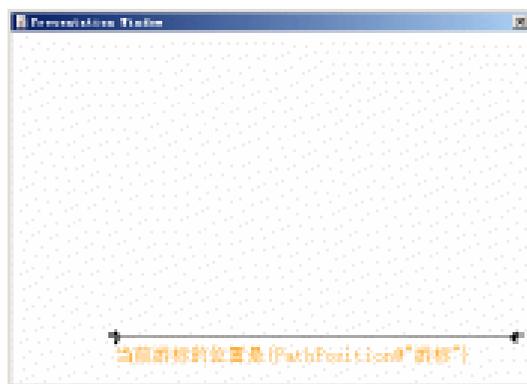


图 73.9 设置“位置显示”文本

⑩ 双击 Wait (等待) 图标，打开其属性对话框，选择 Mouse Click 和 Key Press 复选框，取消选择其他复选框，如图 73.10 所示。

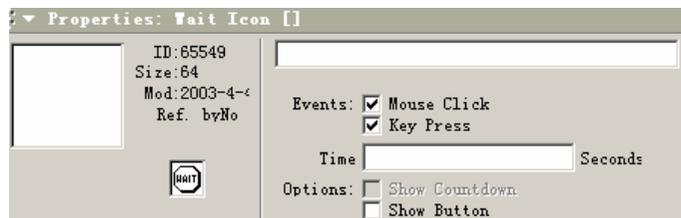


图 73.10 设置等待图标属性



- ⑪ 双击“退出”计算图标，输入代码 Quit(0)，如图 73.11 所示。

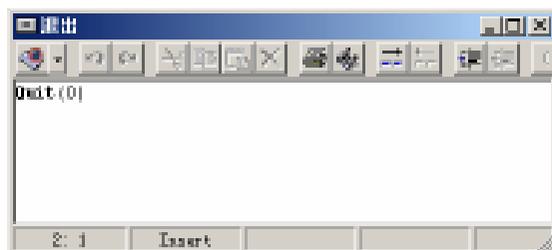


图 73.11 输入代码

- ⑫ 存盘，运行，程序流程线如图 73.12 所示。

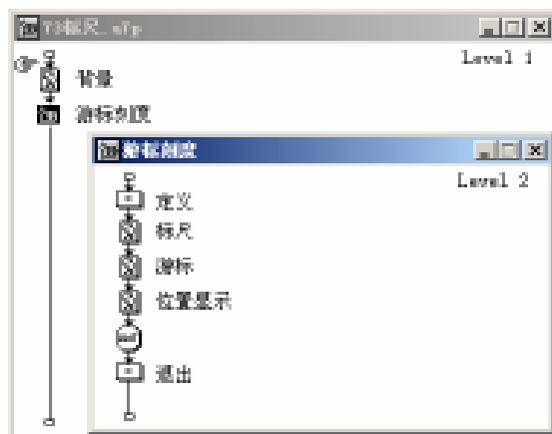


图 73.12 流程线

实例

74

判断网络连接



实例说明

该实例建立一个判断网络连接的应用程序,程序运行后,自动判断是否连接网络,并显示相应的提示,单击按钮,重新测试网络连接状态。

主要知识点:调用函数、判断图标、交互图标。



操作步骤

- ① 新建文件,保存为“74 判断网络连接.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 74.1 所示。

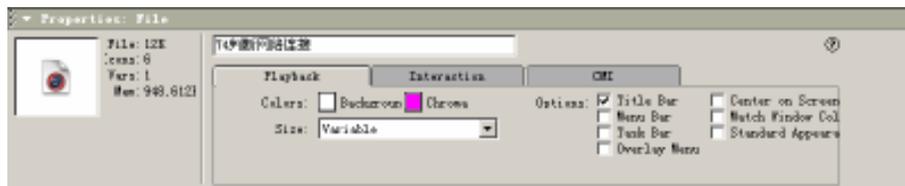


图 74.1 设置文件属性

- ③ 在流程线上添加一个 Calculation (计算) 图标,命名为“网络”,打开代码窗口,输入下列代码:

```
net:=InternetGetConnectedState(0,0)
```



说明

函数 InternetGetConnectedState 在 C:\windows\system\wininet.dll 中。
若返回值 net=0,则未连接;net=1,则已连接。

关闭代码窗口时,选择函数来源,根据上述说明选择,如图 74.2 所示。

- ④ 在流程线上添加一个 Decision (判断) 图标,双击打开其属性对话框,Branch 设置为 To Calculated Path,并在下面的文本框中输入“net+1”,如图 74.3 所示。

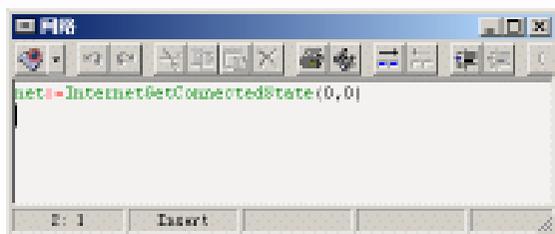


图 74.2 输入代码



图 74.3 设置分支图标属性

⑤ 在 Decision (判断) 图标右侧添加 2 个 Display (显示) 图标, 分别命名为“未连接”和“已连接”, 分别双击 2 个图标上面的 Decision (判断) 标记, 打开属性对话框, 在 Erase Contents 下拉菜单中选择 Before Next Selection 选项, 如图 74.4 所示。



图 74.4 设置分支属性

⑥ 双击“未连接”显示图标, 在演示窗口中输入文本“未连接网络! ”。同理在“已连接”显示图标的演示窗口中输入文本“已连接网络!”。

⑦ 按住 Ctrl 键双击“未连接”显示图标, 打开属性对话框, 选择 Prevent Automatic Erase 复选框, 如图 74.5 所示。按照同样的方法设置“已连接”图标的属性。

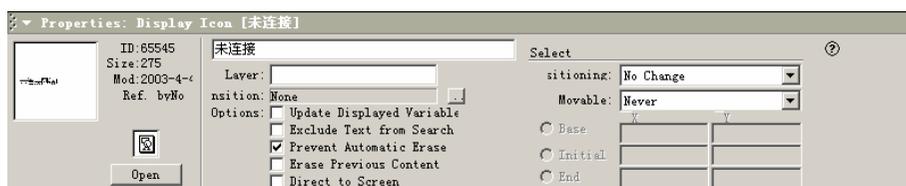


图 74.5 设置显示图标属性

⑧ 在 Decision (判断) 图标下面添加 1 个 Interaction (交互) 图标, 命名为“按钮”, 在其右侧添加一个 Calculation (计算) 图标, 命名为“重新测试”, 双击打开, 输入代码 Restart(), 如图 74.6 所示。

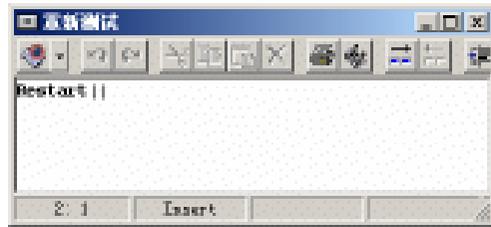


图 74.6 输入代码

⑨ 存盘，运行，程序流程线如图 74.7 所示。

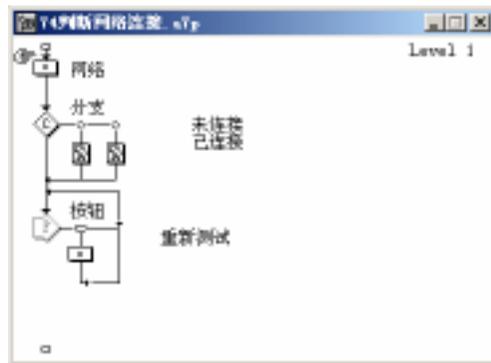


图 74.7 流程线



实例

75

星形线



实例说明

该实例建立一个绘制星形线的应用程序，在界面上移动鼠标，绘制星形的任意效果，颜色随即产生，五彩斑斓。

主要知识点：交互图标、分支图标、计算图标、等待图标、热区交互。



操作步骤

- ① 新建文件，保存为“75 星形线.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 75.1 所示。

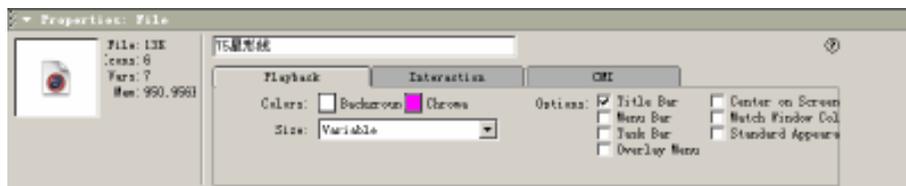


图 75.1 设置文件属性

- ③ 在流程线上添加一个 Interaction（交互）图标，命名为“绘图区域”，在其右侧添加一个 Group（群组）图标，交互类型设置为 Hot Spot，命名为“开始绘图”，双击交互类型标记，打开交互属性对话框，Match 设置为 Single-click，如图 75.2 所示。

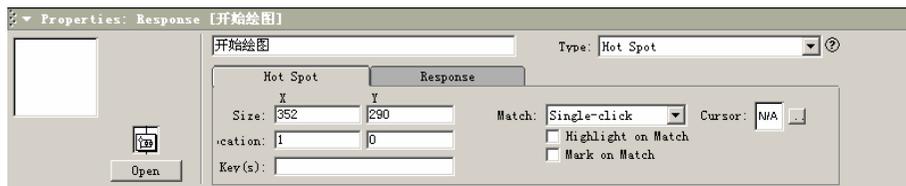


图 75.2 设置交互属性

- ④ 双击 Group（群组）图标，打开二级窗口，在流程线中添加图标，形成图 75.3 所示的流程线。
- ⑤ 双击“鼠标”图标，打开代码窗口，输入代码：



SetCursor(2)
Line(1,CursorX,CursorY,CursorX,CursorY)

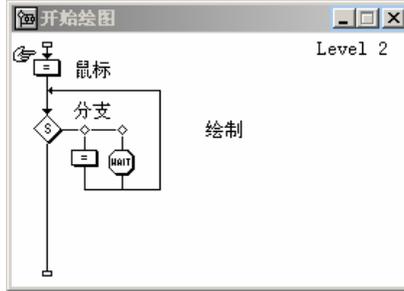


图 75.3 二级窗口流程图

⑥ 双击 Interaction (交互) 图标，打开其属性对话框，Repeat 设置为 Until True，在下面的文本框中输入~MouseDown，Branch 设置为 Sequentially，如图 75.4 所示。

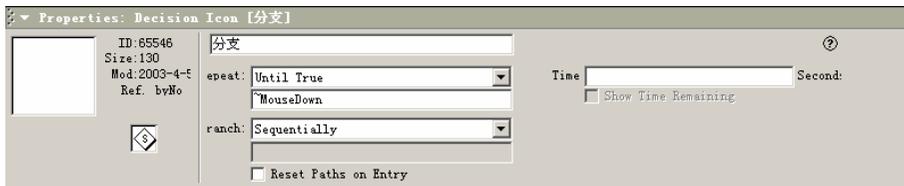


图 75.4 设置分支图标属性

⑦ 双击 Decision (判断) 图标右侧的 Calculation (计算) 图标，输入代码，如图 75.5 所示。

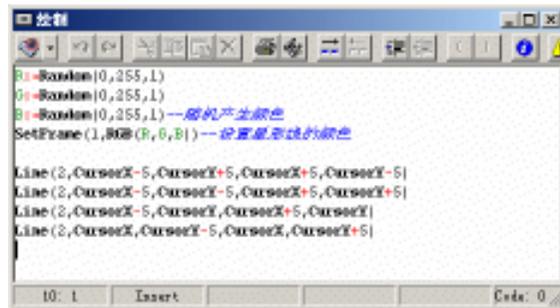


图 75.5 设置判断图标属性

输入的代码如下：

```
R:=Random(0,255,1)
G:=Random(0,255,1)
B:=Random(0,255,1)--随机产生颜色
SetFrame(1,RGB(R,G,B))--设置星形线的颜色
Line(2,CursorX-5,CursorY+5,CursorX+5,CursorY-5)
Line(2,CursorX-5,CursorY-5,CursorX+5,CursorY+5)
Line(2,CursorX-5,CursorY,CursorX+5,CursorY)
Line(2,CursorX,CursorY-5,CursorX,CursorY+5)
```



⑧ 双击 Decision(判断)图标右侧的 Wait(等待)图标,打开其属性对话框,在 Time Limit 文本框中输入“0.05”,取消选择复选框,如图 75.6 所示。

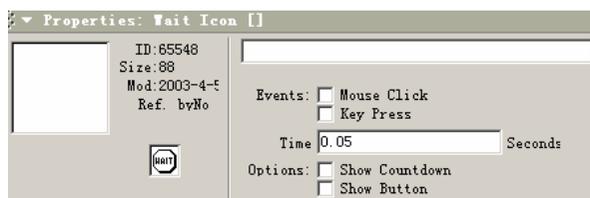


图 75.6 设置等待图标属性

⑨ 存盘,运行,程序流程线如图 75.7 所示。

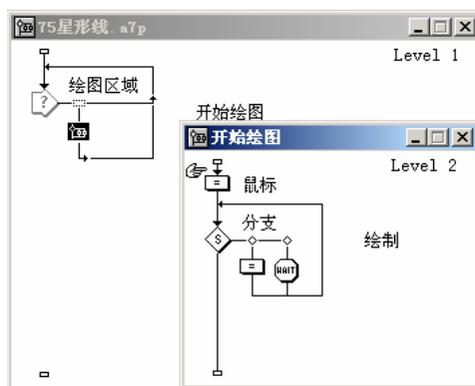
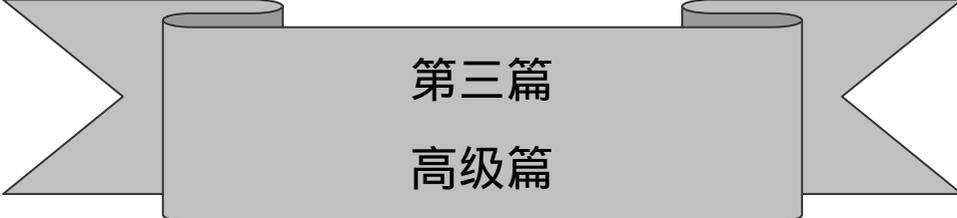


图 75.7 流程线



第三篇 高级篇

本篇运用前两篇所讲述的内容和常用技巧进行综合实例的开发。一个完整的实例，要能够在公共场合演示，或者能够真正成为一件上品，就必须在设计阶段考虑到各种可能出现的情况，所以对 Authorware 7.0 的掌握程度必须达到很高的水平。

本篇讲述如何进行综合实例的开发。读者通过实例的学习，可以很方便地进行多媒体产品的开发。

Let's GO! 



实例

76

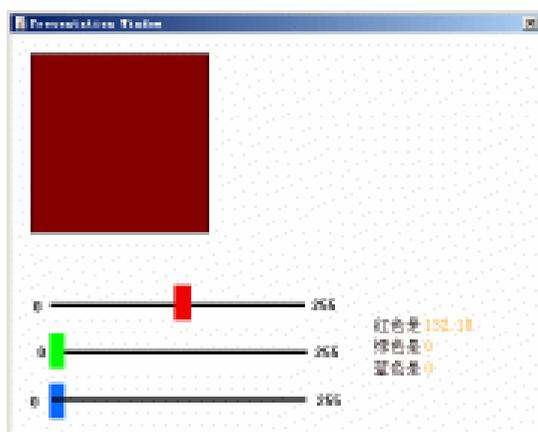
调色板



实例说明

该实例建立一个调色板程序，拖动红绿蓝 3 个滑块改变色值，矩形的颜色也随之而变，并显示当前色值。

主要知识点：交互图标、判断图标、Authorware 变量和函数。



操作步骤

- ① 新建文件，保存为“76 调色板.a7p”。
- ② 执行命令 Modify File Properties，打开属性设置窗口，设置 Size 为 Variable，选择 Center On Screen 复选框，取消选择 Menu Bar 复选框，如图 76.1 所示。

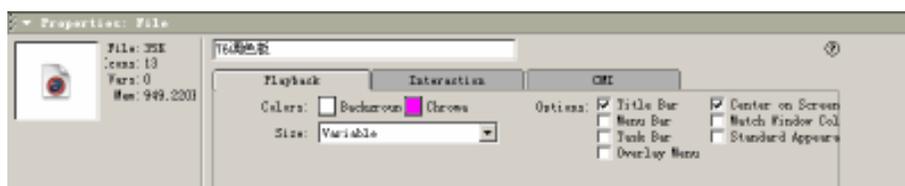


图 76.1 设置文件属性

- ③ 在流程线上添加一个 Group (群组) 图标，命名为“滑块”，用于绘制滑块和滑杆。双击 Group (群组) 图标，打开二级窗口，在其中添加 1 个 Calculation (计算) 图标和 6 个 Display (显示) 图标，并命名，如图 76.2 所示。

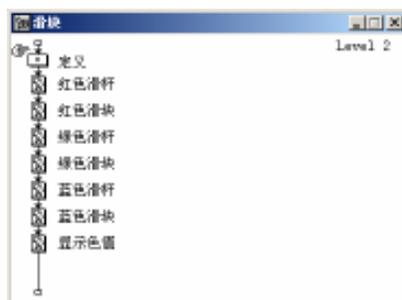


图 76.2 添加图标

④ 双击“定义”计算图标，在代码窗口中输入下列代码：

```
Movable@"红色滑杆":=FALSE
```

```
Movable@"绿色滑杆":=FALSE
```

```
Movable@"蓝色滑杆":=FALSE
```

⑤ 双击“红色滑杆”，在窗口中绘制滑杆，如图 76.3 所示。

⑥ 双击“红色滑块”图标，在其中绘制红色滑块，调整滑块和滑杆的位置关系，如图 76.4 所示。

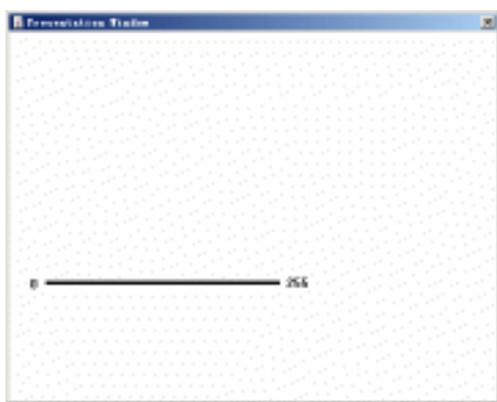


图 76.3 “红色滑杆”内容

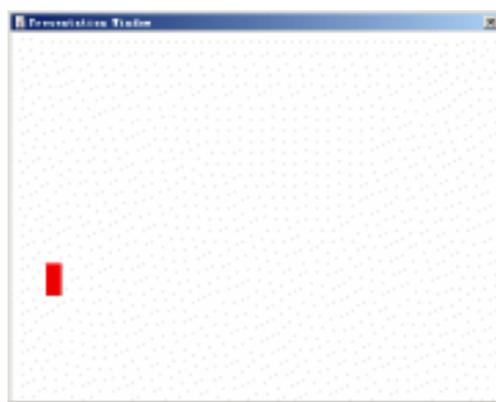


图 76.4 “红色滑块”内容

⑦ 单击“红色滑块”图标，打开其属性对话框，在 Positioning 和 Movable 下拉菜单中均选择 On Path，并设置滑块的运动轨迹为图 76.3 中所绘制的滑杆，如图 76.5 所示。

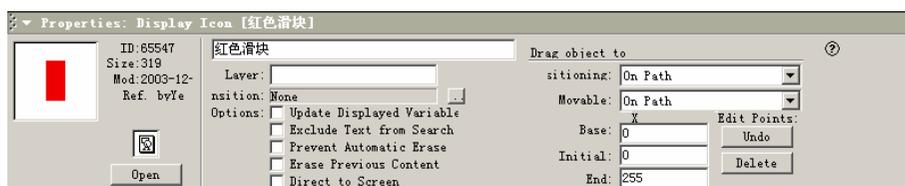


图 76.5 设置红色滑块属性

⑧ 重复步骤 5~7 设置其他几个显示图标的内容。

⑨ 双击“显示色值”图标，在演示窗口中输入下列文本，并设置文本的属性。

红色是{PathPosition@"红色滑块"}

绿色是{PathPosition@"绿色滑块"}

蓝色是{PathPosition@"蓝色滑块"}

⑩ 按住 Ctrl 键双击“显示色值”图标，打开属性对话框，选择 Update Displayed Variables 复选框，如图 76.6 所示。

⑪ 回到主程序线，添加一个 Interaction (交互) 图标，在其右侧添加一个 Group (群组) 图标，交互类型设置为 Conditional，单击交互类型标记，打开属性对话框，在名称栏内输入“Dragging@"红色滑块"|Dragging@"绿色滑块"|Dragging@"蓝色滑块"”，也就是执行分支的条件，即拖动红色滑块、绿色滑块或蓝色滑块时执行该分支。同时，在 Response 选



项卡内选择 Perpetual 复选框，Erase 设置为 Don't Erase，Branch 设置为 Exit Interaction，如图 76.7 和图 76.8 所示。

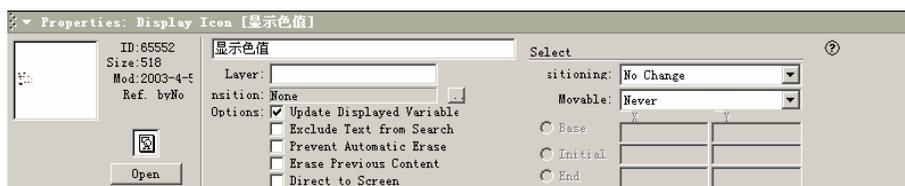


图 76.6 设置“显示色值”图标属性

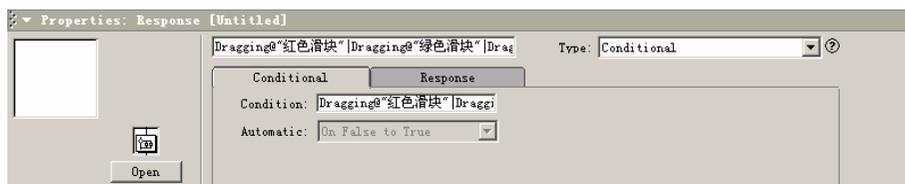


图 76.7 交互图标属性的 Conditional 选项卡

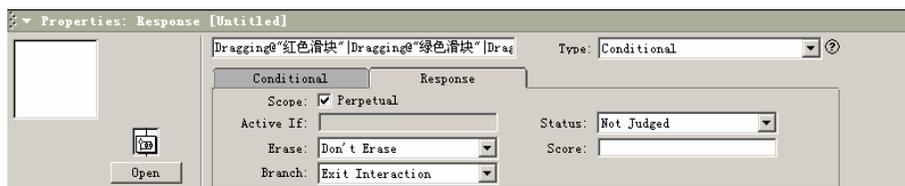


图 76.8 交互图标属性的 Response 选项卡

12 双击 Group (群组) 图标，打开二级窗口，添加一个 Decision (判断) 图标，命名为“循环”，在其右侧添加一个 Calculation (计算) 图标，命名为“绘制矩形”。单击 Decision (判断) 图标，进行属性设置，Repeat 设置为 Until True，在下面的文本框中输入“~Dragging@"红色滑块" & ~Dragging@"蓝色滑块" & ~Dragging@"绿色滑块"”，Branch 设置为 Sequentially，如图 76.9 所示。

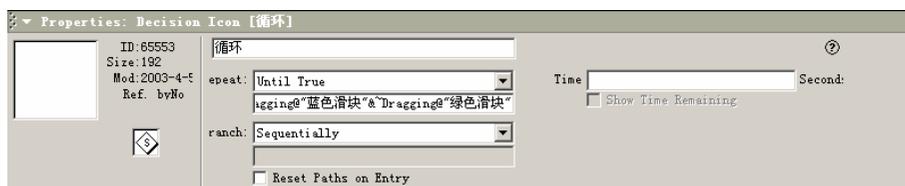


图 76.9 设置 Decision (判断) 图标属性

13 单击 Calculation (计算) 图标上面的 Decision (判断) 标记，打开 Decision (判断) 属性对话框，选择 Don't Erase 选项，如图 76.10 所示。

14 双击 Decision (判断) 图标右侧的 Calculation (计算) 图标，输入代码，如图 76.11 所示。

输入的代码如下：

```
SetFill(1,RGB(PathPosition@"红色滑块",PathPosition@"蓝色滑块",PathPosition@"绿色滑块"))
```



Box(1,20,20,200,200)



图 76.10 设置 Decision (判断) 属性

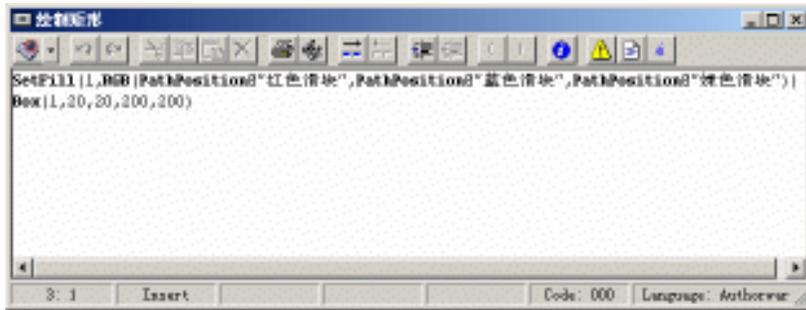


图 76.11 输入代码



Box()函数用于绘制矩形，其基本形式是：
Box(pensize, x1, y1, x2, y2)，(x1, y1)和 (x2, y2) 是矩形对角线上的两点，pensize 表示矩形边线的宽度。

15 存盘，运行，程序流程线如图 76.12 所示。



图 76.12 流程线



实例

77

文本输入框



实例说明

该实例建立一个文本输入框效果，程序运行后，弹出输入框，提示用户输入姓名，然后将用户输入的内容显示在演示窗口中。

主要知识点：函数调用、计算图标。



操作步骤

- ① 新建文件，保存为“77 文本输入框.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 77.1 所示。

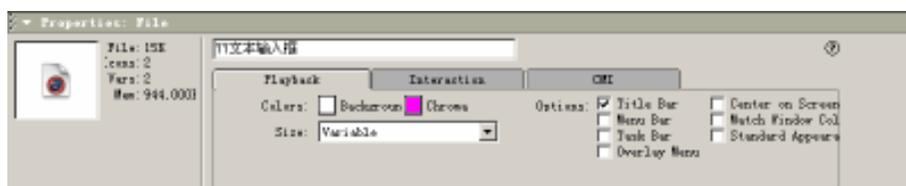


图 77.1 设置文件属性

- ③ 在流程线中添加一个 Calculation (计算) 图标，命名为“定义”，双击打开代码窗口，输入代码，如图 77.2 所示。

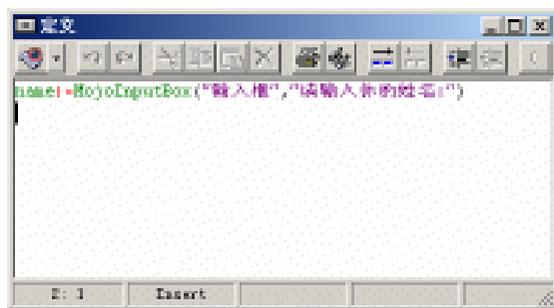


图 77.2 输入代码

- ④ 关闭代码窗口，系统提示选择函数来源，选择 MInputDialog 函数，如图 77.3 所示。



图 77.3 选择函数来源

⑤ 在流程线上添加一个 Display (显示) 图标, 命名为 Display (显示), 双击该图标, 在演示窗口中输入: “你的姓名是:{name}”, 如图 77.4 所示。

⑥ 存盘, 程序流程线如图 77.5 所示。



图 77.4 输入显示内容

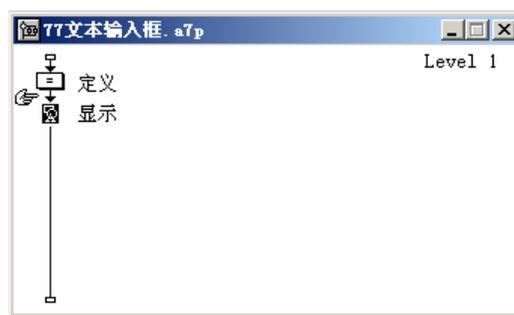


图 77.5 流程线

⑦ 运行, 程序效果如图 77.6 所示。



图 77.6 程序效果



实例

78

不规则窗口 (1)



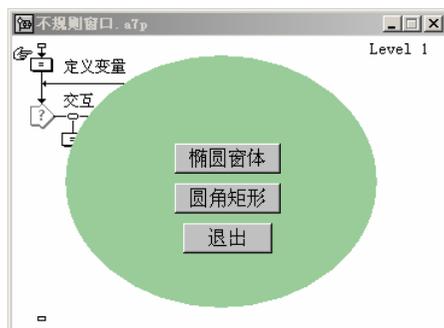
实例说明

该实例建立一个不规则窗口效果,单击程序界面上的“椭圆窗体”按钮,窗体将变为椭圆形。

主要知识点:调用函数、交互图标、按钮交互。



操作步骤



① 新建文件,保存为“78 不规则窗口.a7p”。

② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,设置背景色为浅绿色,取消选择 Menu Bar 复选框,如图 78.1 所示。



图 78.1 设置文件属性

③ 执行命令 Windows Panels Functions,打开 Functions 对话框,如图 78.2 所示,在 Category 下拉菜单中选择本例子的名称。



图 78.2 Functions 对话框

④ 单击 Functions 对话框中的 Load 按钮，在 C:/Windows/Syetem32 目录下选择 gdi32.dll，单击“打开”按钮，如图 78.3 所示。

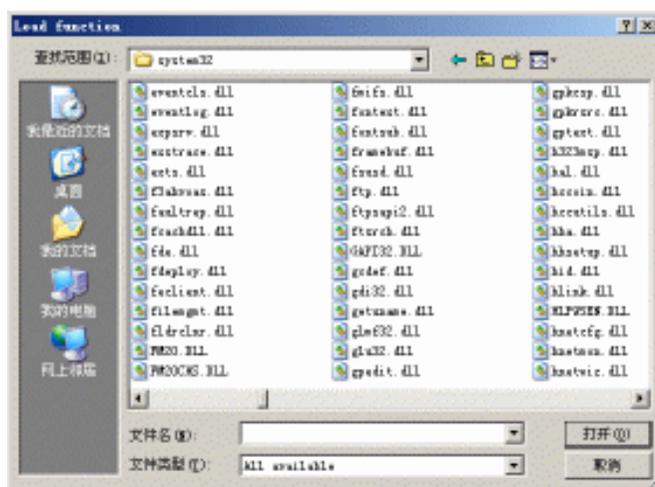


图 78.3 选择函数

⑤ 在图 78.3 中选择 gdi32.dll 后，单击“打开”按钮打开如图 78.4 所示的对话框，在该对话框中设置如下：

- Function Name：输入函数名称 CreateEllipticRgn；
- Arguments：输入参数 long,long,long,long；
- Return：输入返回值 long。

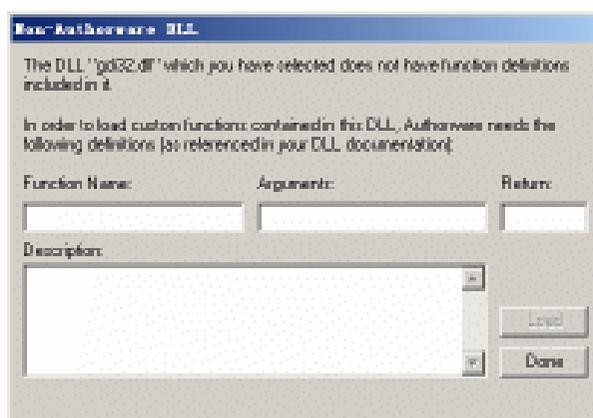


图 78.4 函数设置

⑥ 参数设置完毕，单击 Load 按钮载入函数。

⑦ 单击 Done 按钮，系统打开 Load Functions 对话框，提示选择函数，选择 C:/Windows/Syetem32 目录下的 user32.dll，如图 78.5 所示。

⑧ 打开图 78.3 所示的函数设置对话框，在次对话框中设置如下：

- Function Name：输入函数名称 SetWindowRgn；



- Arguments：输入参数 long,long,long；
- Return：输入返回值 long。

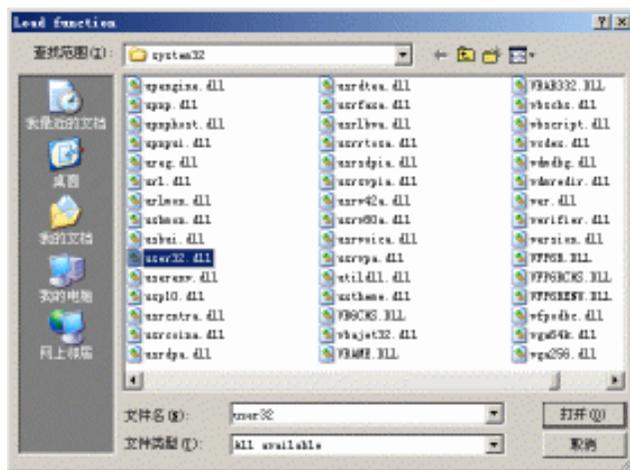


图 78.5 选择函数

⑨ 单击 Load 按钮载入函数，然后单击 Done 按钮结束加载函数，系统仍然打开对话框，单击“取消”按钮即可。

⑩ 此时，在 Functions 对话框的列表中列出了刚才加载的 2 个函数，单击 Done 按钮退出该对话框。

⑪ 在流程线中添加一个 Calculation (计算) 图标，命名为“定义变量”，打开代码窗口，输入代码，如图 78.6 所示。

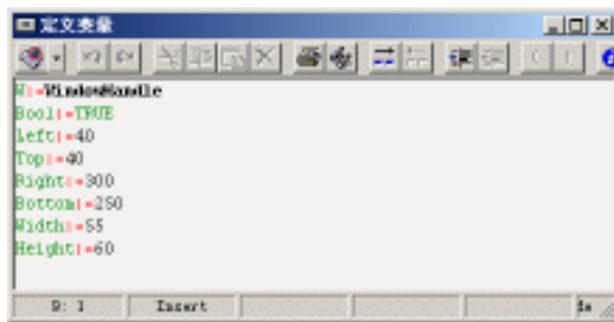


图 78.6 定义变量



说明

Left,Top,Right,Bottom,Width,Height 分别代表不规则窗口左上角和右下角的坐标，以及生成圆角的椭圆的宽度和高度。

⑫ 在流程线上添加一个 Interaction (交互) 图标，并在其右侧添加一个 Calculation (计算) 图标，交互类型设置为 Button，命名为“椭圆窗体”，双击该图标上面的交互类型

标记,打开属性对话框, Erase 设置为 After Next Entry, Branch 设置为 Try Again,如图 78.7 所示。



图 78.7 设置交互属性

⑬ 双击“椭圆窗体”计算图标,打开代码窗口,输入代码,如图 78.8 所示。

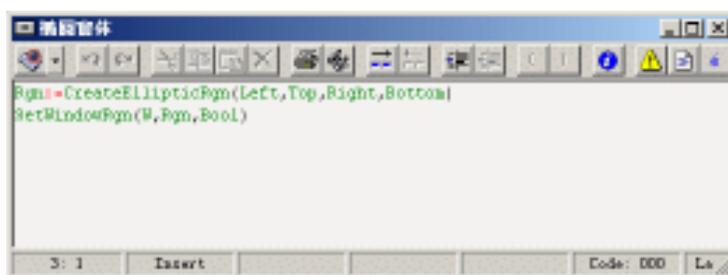


图 78.8 定义椭圆窗体

输入的代码如下：

```
Rgn:=CreateEllipticRgn(Left,Top,Right,Bottom)
SetWindowRgn(W,Rgn,Bool)
```



技巧

如果要生成圆形窗口,则需令 Left=Top, Right=Bottom。

⑭ 存盘,运行,程序流程线如图 78.9 所示。

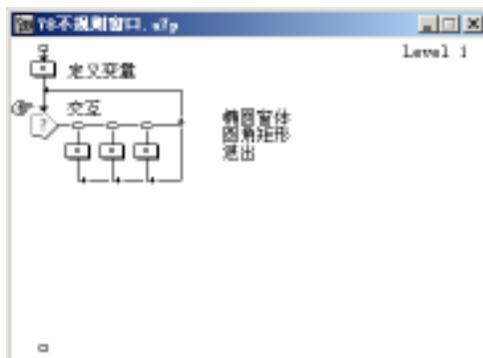


图 78.9 流程线



实例

79

不规则窗口 (2)



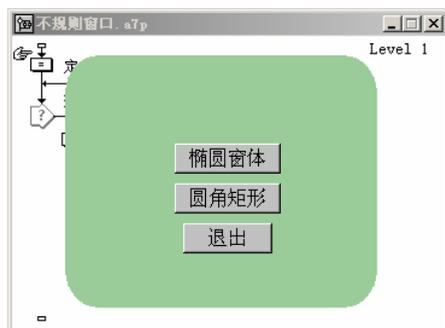
实例说明

该实例建立一个不规则窗口效果,单击程序界面上的“圆角矩形”按钮,窗体将变为圆角矩形。

主要知识点:调用函数、交互图标、按钮交互。



操作步骤



① 同上例,单击 Functions 对话框中的 Load 按钮,载入 C:/Windows/Sytem32 目录下的 gdi32.dll,单击“打开”按钮,打开函数设置对话框,在该对话框中设置如下:

- Function Name: 输入函数名称 CreateRoundRectRgn;
- Arguments: 输入参数 long,long,long,long,long;
- Return: 输入返回值 long。

如图 79.1 所示。

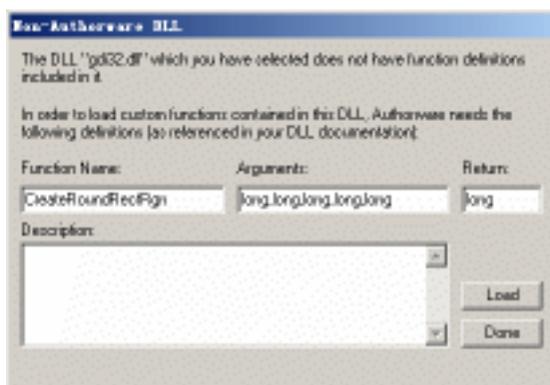


图 79.1 设置函数参数

② 参数设置完毕,单击 Load 按钮载入函数,如果加载成功,则在窗口下面给出成功载入函数的提示。单击 Done 按钮,系统打开 Load Functions 对话框,因为前面已经导入了 user32.dll,所以这里单击“取消”按钮即可。

③ 此时,在 Functions 对话框的列表中列出了加载的个函数,如图 79.2 所示,单击 Done 按钮退出该对话框。

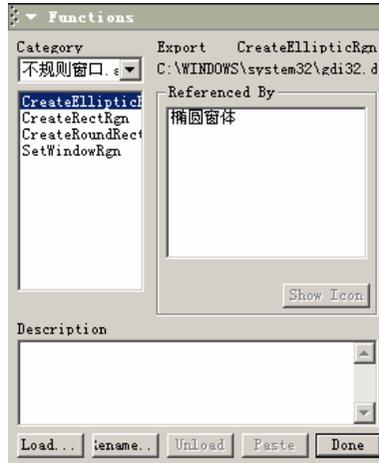


图 79.2 已加载的函数

④ 接上例，在流程线中的 Interaction(交互)图标右侧再添加一个 Calculation(计算)图标，命名为“圆角矩形”，打开代码窗口，输入代码，如图 79.3 所示。

输入的代码如下：

```
Rgn:=CreateRoundRectRgn(Left,Top,Right,Bottom,Width,Height)
SetWindowRgn(W,Rgn,Bool)
```

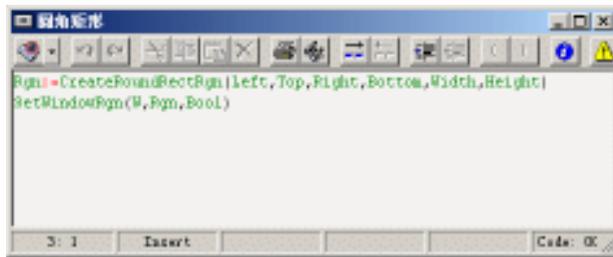


图 79.3 输入代码

⑤ 在 Interaction(交互)图标右侧再添加一个 Calculation(计算)图标，命名为“退出”，打开代码窗口，输入代码 Quit(0)。

⑥ 存盘，运行，程序流程线如图 79.4 所示。

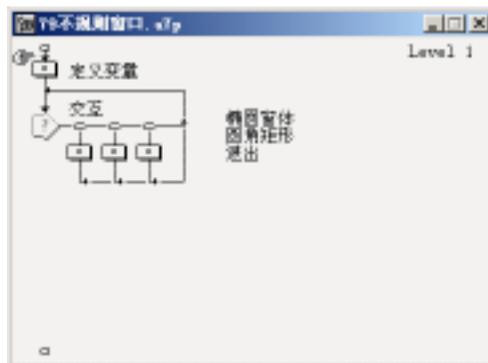


图 79.4 流程线



实例

80

右键菜单 (1)



实例说明

该实例建立一个右键菜单效果，程序运行后，不仅可以从菜单栏执行菜单，单击右键，也可以执行菜单，菜单可以显示多级。

主要知识点：交互图标、函数调用。



操作步骤

- ① 新建文件，保存为“80 右键菜单.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，如图 80.1 所示。

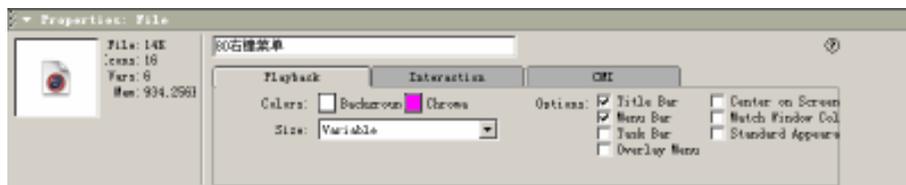


图 80.1 设置文件属性

- ③ 这里可以采用和 78、79 例中相同的方法导入 AppendMenu、DrawmenuBar、GelMenu、Gclsubmenu、InsertMenu、RemoveMenu 和 TrackPopupMenu 函数。
- ④ 在流程线中添加一个 Interaction (交互) 图标，命名为“编辑”，在其右侧添加 2 个 Group (群组) 图标，交互类型设置为 Pull-Down Menu，分别命名为“复制”和“粘贴”，单击“复制”群组图标上面的交互类型标记 , 打开属性对话框，在 Key(s) 栏中输入快捷键 Ctrl+C，切换到 Response 选项卡内，选择 Perpetual 复选框，Branch 设置为 Return，其他选项默认，如图 80.2 和图 80.3 所示。

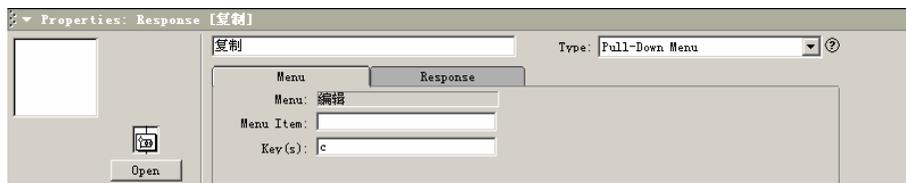


图 80.2 设置“复制”交互属性的 Menu 选项卡

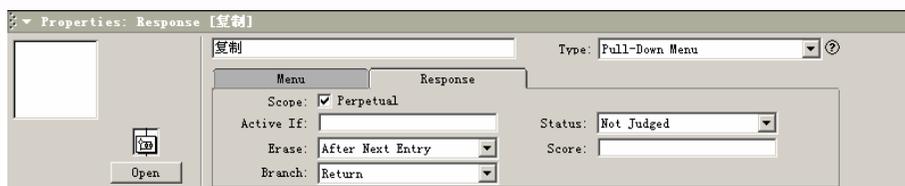


图 80.3 设置“复制”交互属性的 Response 选项卡

⑤ 同理，设置“粘贴”的交互属性，如图 80.4 和图 80.5 所示。

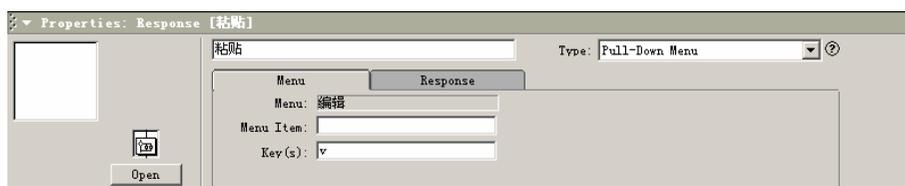


图 80.4 设置“粘贴”交互属性的 Menu 选项卡

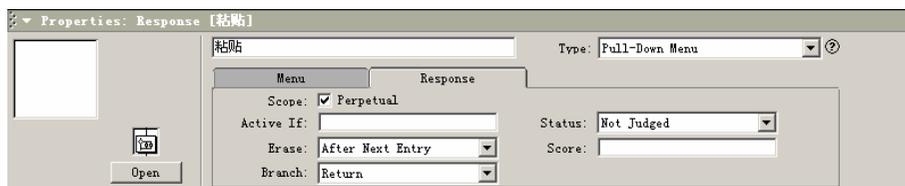


图 80.5 设置“粘贴”交互属性的 Response 选项卡

⑥ 此时程序流程线如图 80.6 所示。

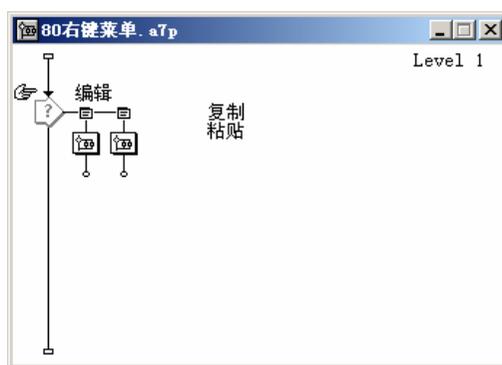


图 80.6 流程线

⑦ 双击“复制”图标，打开二级窗口，添加一个 Display（显示）图标，在其中输入文本：你选择了“复制”命令，如图 80.7 所示。

⑧ 同理，设置“粘贴”图标的内容，输入的文本为：你选择了“粘贴”命令。

⑨ 在主流程线中添加一个 Interaction（交互）图标，并在其右侧添加一个 Group（群组）图标，命名为“子菜单一”，其属性设置方法和内容与前面相似，其中，群组图标中显示图标的内容如图 80.8 所示，完成后流程图如图 80.9 所示。



图 80.7 输入交互文本

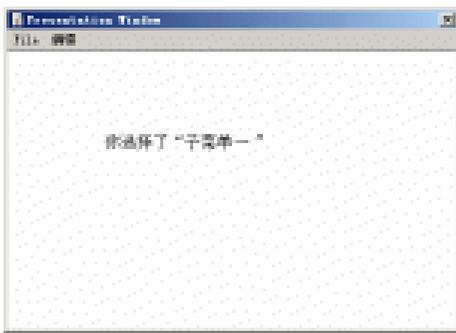


图 80.8 输入内容

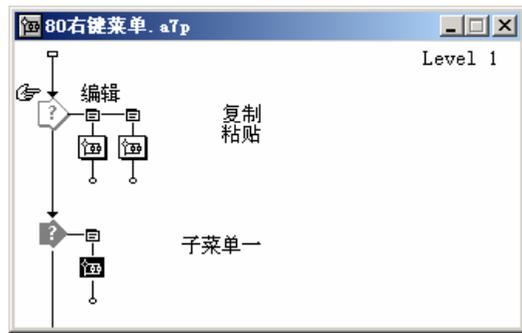


图 80.9 流程线

⑩ 依照同样的方法建立“子菜单二”和“子菜单三”，如图 80.10 所示。

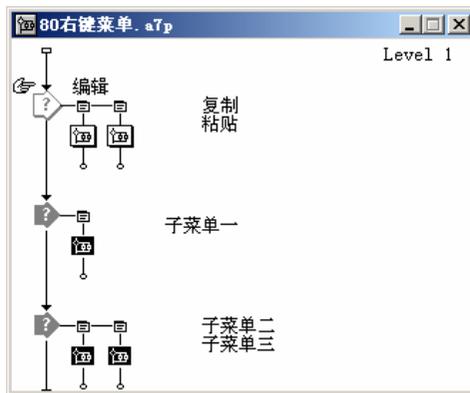


图 80.10 建立“子菜单二”和“子菜单三”

实例

81

右键菜单(2)



实例说明

该实例建立一个右键菜单效果，程序运行后，不仅可以从菜单栏执行菜单，单击右键，也可以执行菜单，菜单可以显示多级。



主要知识点：交互图标、函数调用。



操作步骤

① 接上例，在主流程图中添加一个 Calculation(计算)图标，命名为“创建子菜单”，双击打开代码窗口，输入下列代码：

```
--获取菜单的句柄
subhandle1:=GetSubMenu(menuhandle,1)
subhandle2:=GetSubMenu(menuhandle,2)
subhandle3:=GetSubMenu(menuhandle,3)
RemoveMenu(menuhandle,3, 1024)
RemoveMenu(menuhandle,2, 1024) --从菜单栏上删除子菜单
DrawMenuBar(WindowHandle) --重画菜单栏
--追加或插入子菜单
AppendMenu(subhandle1, 16, subhandle2, "一级子菜单")
InsertMenu(subhandle2,0,1040,subhandle3,"二级子菜单")
```



说明

函数 GetSubMenu(HMENU hMenu, int nPos)的作用是获取各菜单的句柄，其中：参数 HMENU hMenu 是上级菜单的句柄；参数 int nPos 是条目在菜单中的位置，第一个条目的编号为 0。

AppendMenu 是追加，InsertMenu 是插入，可以根据需要将子菜单设定在任意位置。

InsertMenu 中的第二个参数是插入的位置。



注意

使用 RemoveMenu 函数时，删除顺序不能颠倒。与 DeleteMenu 函数不同的是：RemoveMenu 函数并不能将菜单清除，只是不显示。



② 在流程线中添加一个 Interaction (交互) 图标, 在其右侧添加一个 Calculation (计算) 图标, 交互类型设置为 Conditional, 双击交互类型标记, 打开属性对话框, 在 Condition 栏中输入 RightMouseDown, 并选择 Automatic 下拉菜单中的 When True 选项, 在 Response 选项卡内设置 Branch 为 Try Again, 如图 81.1 和图 81.2 所示。

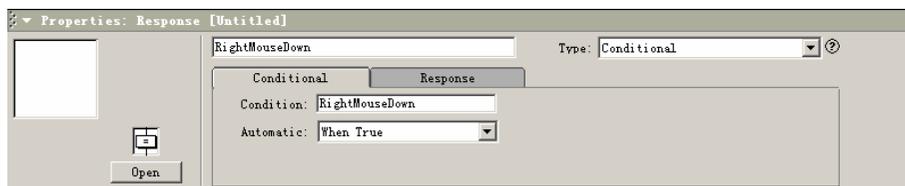


图 81.1 设置交互属性的 Conditional 选项卡



图 81.2 设置交互属性的 Response 选项卡

③ 双击 Calculation (计算) 图标输入下列代码:

```
X:=CursorX + WindowLeft
Y:=CursorY + WindowTop
TrackPopupMenu(subhandle1, 0, X, Y, 0, WindowHandle,0)
```

关闭代码窗口时, 系统弹出窗口, 要求选择函数 TrackPopupMenu() 的来源, 选择 C:/Windows/System32 目录下的 user32.dll, 如图 81.3 所示。

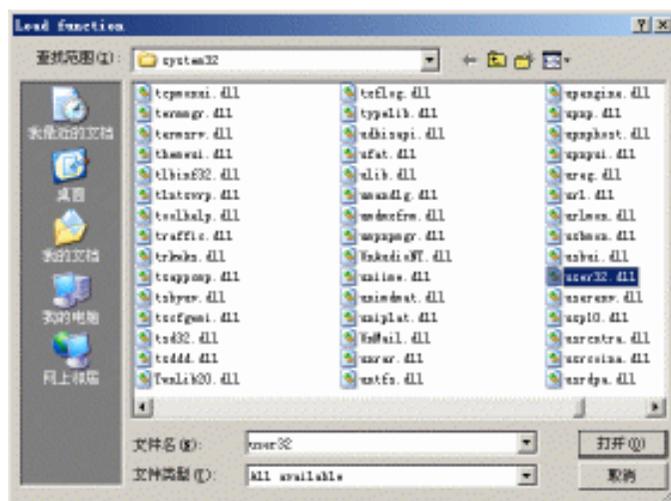


图 81.3 选择函数来源

④ 单击“打开”按钮, 弹出图 81.4 所示的函数设置对话框, 在该对话框中设置如下:

- Function Name: 输入 TrackPopupMenu;

- Arguments：输入 Long,Long,Long,Long,Long,Long,Long；
- Return：输入 Long。

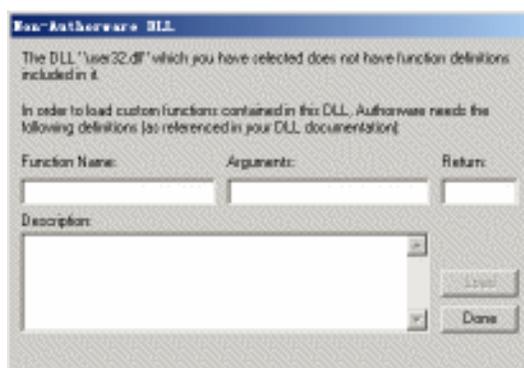


图 81.4 函数设置对话框

输入完成后，单击 Load 按钮和 Done 按钮完成函数加载。

⑤ 该程序中的右键菜单只有打包后才能看到效果，下面我们将它打包。执行命令 File Publish Package，打开 Package File 对话框，按照图 81.5 所示设置，单击 Save File(s)&Package 按钮打包。

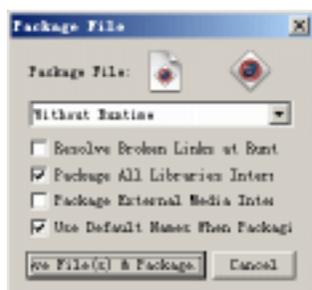


图 81.5 打包文件

⑥ 存盘，运行，程序流程线如图 81.6 所示。

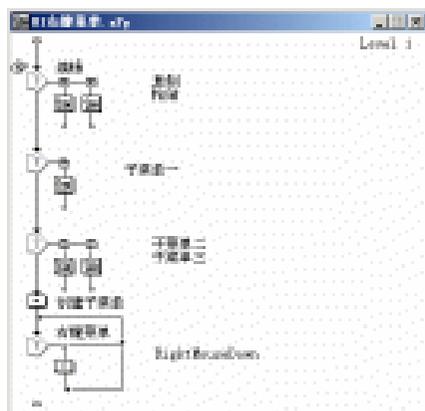


图 81.6 流程线



实例

82

绘图程序



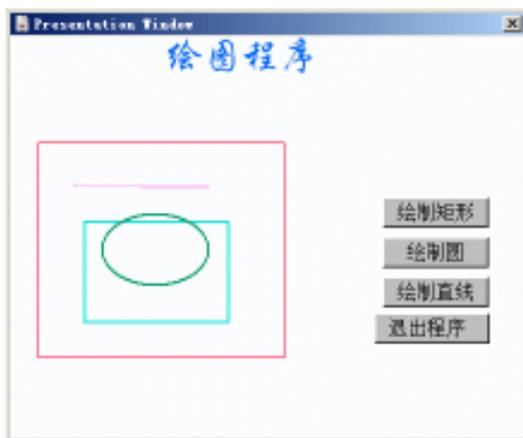
实例说明

该实例建立一个简单的绘图程序,单击界面上的按钮,在矩形区域中移动鼠标,可以绘制对应的图形,图形的颜色随机产生。

主要知识点:交互图标、热区交互、调用函数。



操作步骤



① 新建文件,保存为“82 绘图程序.a7p”。

② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 82.1 所示。

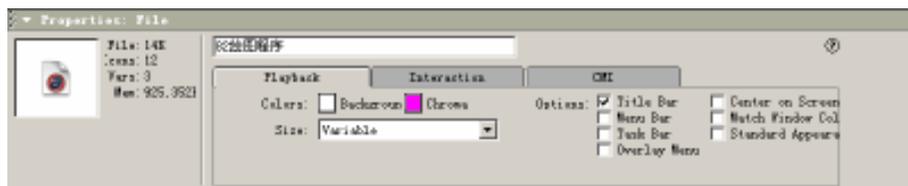


图 82.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,打开演示窗口,使用文本工具  输入标题“绘图程序”,并双击椭圆形工具 ,打开调色板,为文本着色。然后,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,双击箭头工具 ,设置文本的显示模式为 Transparent。使用绘图工具箱中的矩形工具  在窗口中绘出一个矩形区域,作为绘图区域,如图 82.2 所示。

④ 为了防止背景移动,需要添加代码予以控制,右键单击“背景”图标,选择 Calculation 命令,在代码窗口中输入代码: Movable:=False,如图 82.3 所示。

⑤ 在流程线中添加一个 Interaction (交互) 图标,命名为 Interaction (交互),在其右侧添加 3 个 Group (群组) 图标和 1 个 Calculation (计算) 图标,分别命名为“绘制矩形”、“绘制圆”、“绘制直线”和“退出程序”,交互类型均设置为 Button。

⑥ 双击 Group (群组) 图标上面的交互类型标记,打开属性对话框, Erase 设置为

Don't Erase , Branch 设置为 Continue , 在 Button 选项卡内设置光标形状, 如图 82.4 和图 82.5 所示。



图 82.2 设置背景

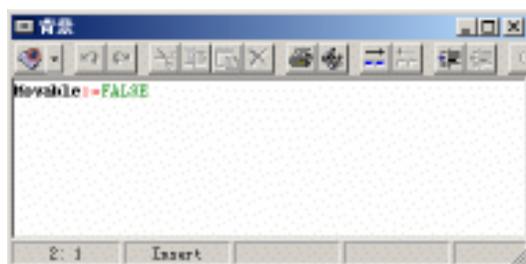


图 82.3 输入代码控制背景

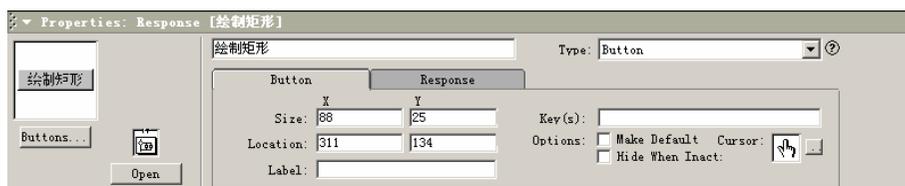


图 82.4 设置按钮交互属性的 Button 选项卡

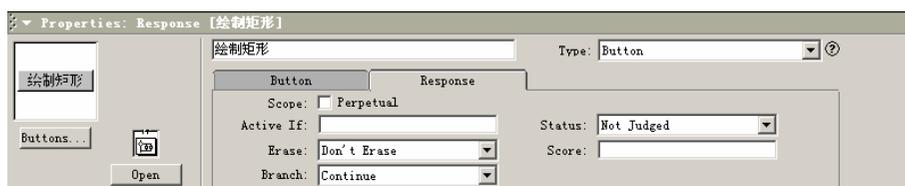


图 82.5 设置按钮交互属性的 Response 选项卡

⑦ 按照同样的方法设置其他 2 个 Group (群组) 图标的交互属性。双击 Calculation (计算) 图标上面的交互类型标记, 打开属性对话框, Erase 设置为 Don't Erase , Branch 设置为 Exit Interaction , 在 Button 选项卡内设置光标形状, 此时的流程图如图 82.6 所示。

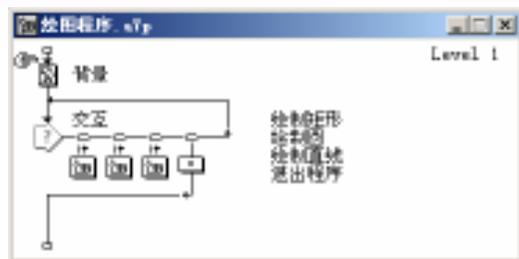


图 82.6 流程图

8 双击“退出程序”图标，在代码窗口中输入代码，如图 82.7 所示。

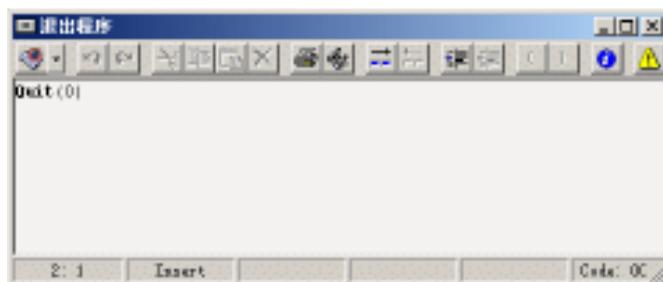


图 82.7 输入代码

9 双击“绘制矩形”图标，打开二级窗口，添加图标，并命名，如图 82.8 所示。

10 双击图 82.8 中 Calculation (计算) 图标上面的交互类型标记，打开属性对话框，在 Type 下拉菜单中选择 Hot Spot，将“背景”中的矩形区域作为热区，如图 82.9 和图 82.10 所示。在 Response 选项卡内将 Erase 设置为 Don't Erase，Branch 设置为 Exit Interaction，如图 82.11 所示。

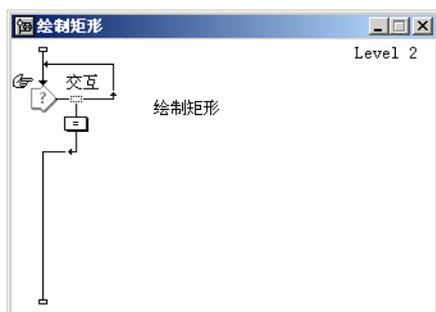


图 82.8 添加图标



图 82.9 定义热区

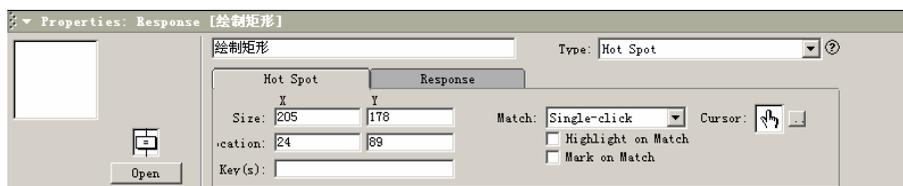


图 82.10 设置交互属性的 Hot Spot 选项卡

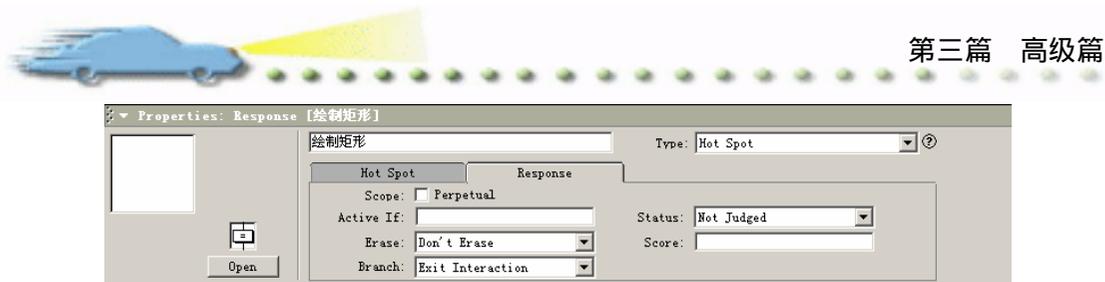


图 82.11 设置交互属性的 Response 选项卡

⑪ 双击图 82.8 中 Calculation (计算) 图标，打开代码窗口，输入下列代码：

```
R:=Random(0,255,1)
G:=Random(0,255,1)
B:=Random(0,255,1)
SetFrame(1,RGB(R,G,B))
DrawBox(2)
```

⑫ 重复步骤 9~11，设置“绘制圆”和“绘制直线”图标的内容。其中，绘制圆的函数是 DrawCricle(2)，绘制直线的函数是 DrawLine(2)，括号中的数字表示线的宽度。

⑬ 存盘，运行，程序流程线如图 82.12 所示。

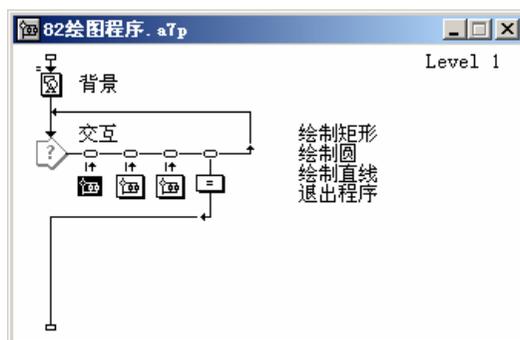


图 82.12 流程线



实例

83

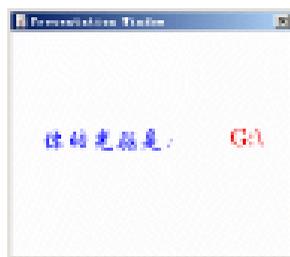
寻找光驱



实例说明

该实例建立一个寻找光驱的应用程序，将光驱的盘符显示在窗口中。

主要知识点：知识对象。



操作步骤

- ① 新建文件，保存为“83 寻找光驱.a7p”。
- ② 执行命令 Modify File Properties，打开属性设置窗口，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 83.1 所示。



图 83.1 设置文件属性

- ③ 如果没有打开知识对象，执行命令 Windows Knowledge Objects 将其打开，在 Category 下拉菜单中选择 File，在下面的列表中可以发现 Find CD Drive 知识对象，双击该对象，打开向导，第一个界面是该对象的说明介绍，如图 83.2 所示。

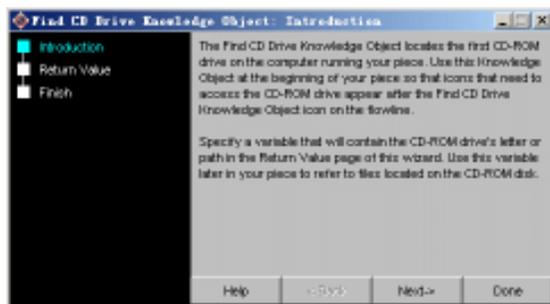


图 83.2 知识对象向导 1

- ④ 点击 Next 按钮进入第二个界面，要求输入一个变量作为返回值，输入 cd，同时选择 Drive Path 单选按钮，如图 83.3 所示。
- ⑤ 点击 next 按钮进入本向导的结束界面，提示将 winapi.ucd 和 winapi.u32 拷贝到指

定目录，单击 Done 按钮结束向导，如图 83.4 所示。



图 83.3 知识对象向导 2

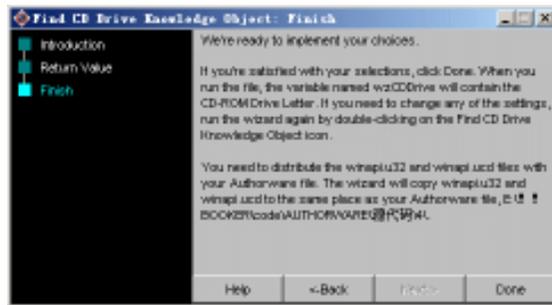


图 83.4 知识对象向导 3

- ⑥ 此时，流程线中出现知识对象图标，将其重命名为“寻找光驱”。
- ⑦ 在流程线中添加一个 Display (显示) 图标，命名为“显示光驱”，打开演示窗口，使用文本工具输入：“你的光驱是：{cd}”，并设置文本属性，如图 83.5 所示。
- ⑧ 存盘，运行，程序流程线如图 83.6 所示。

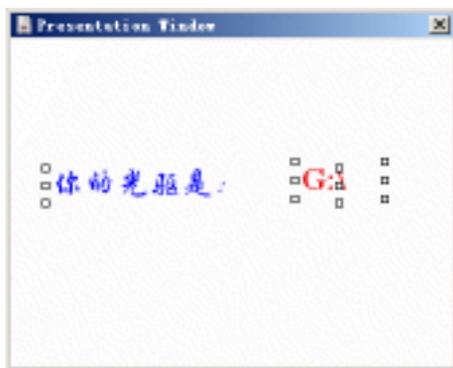


图 83.5 显示光驱

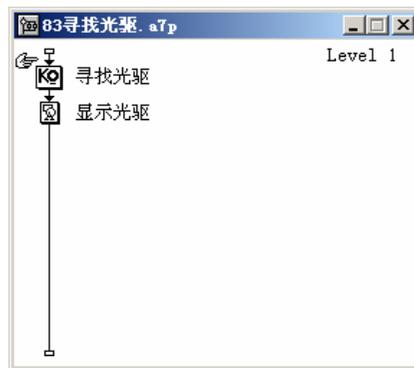


图 83.6 流程线



实例

84

转换音乐格式



实例说明

该实例演示如何将 WAV 格式的音乐转换为 SWA 格式，并播放转换后的音乐。

主要知识点：转换音乐格式、声音图标。



操作步骤

① 新建文件，保存为“84 转换音乐格式.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 84.1 所示。

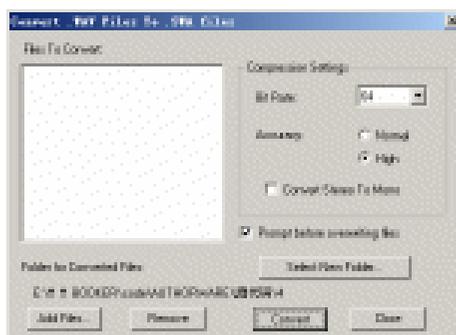


图 84.1 设置文件属性

③ 执行命令 Xtras Other Convert WAV to SWA，打开文件转换对话框，在 Bit Rate 中选择“64”，Accuracy 设置为 High，选择 Convert Stereo To Mono 和 Prompt before overwriting files 复选框，如图 84.2 所示。

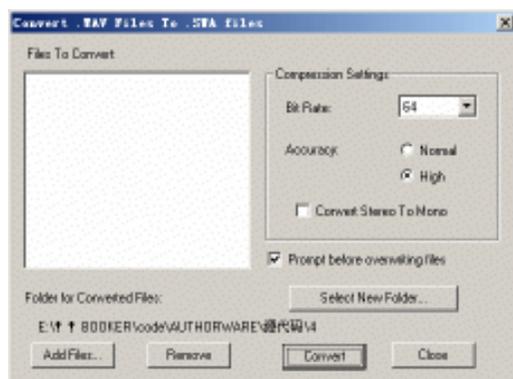


图 84.2 转换音乐文件对话框

④ 单击该对话框中的 **Select New Folder...** 按钮，选择输出文件的路径，默认的路径是存放 Authorware 源程序的文件夹，如图 84.3 所示。

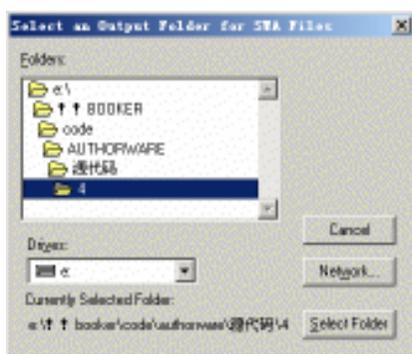


图 84.3 选择输出文件的路径

⑤ 再单击对话框中的 **Add Files...** 按钮，打开对话框，选择要转换的 WAV 文件，选择后单击“打开”按钮，如图 84.4 所示。



图 84.4 选择要转换的音乐文件

⑥ 在如图 84.5 所示对话框的 Files to Convert 列表中显示要转换的文件，确认后，单击 **Convert** 按钮进行转换，转换后，单击 **Close** 按钮，退出对话框。

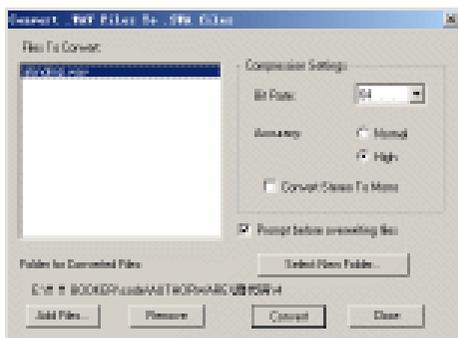


图 84.5 转换音乐文件



⑦ 在流程线中添加一个 Sound (声音) 图标, 双击该图标, 打开其属性设置框, 单击 Import 按钮导入刚才转换生成的 SWA 文件, 如图 84.6 所示。

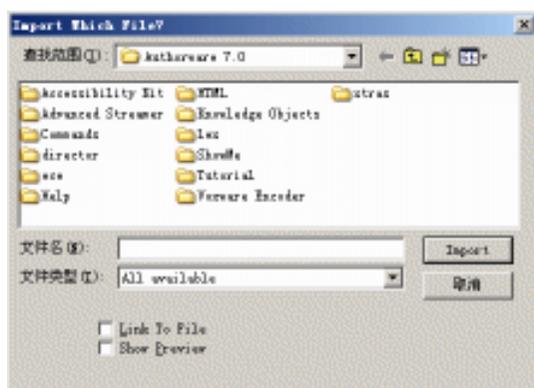


图 84.6 导入 SWA 声音文件

⑧ 将声音属性对话框切换到 Timing 选项卡内, Concurrency 设置为 Wait Until Done, Play 设置为 Until True, 在下面的文本框中输入 False 或 “0”, 表示循环播放该声音文件, 其他选项默认, 如图 84.7 所示。



图 84.7 设置声音图标属性

⑨ 存盘, 运行, 播放 SWA 声音文件, 程序流程线如图 84.8 所示。

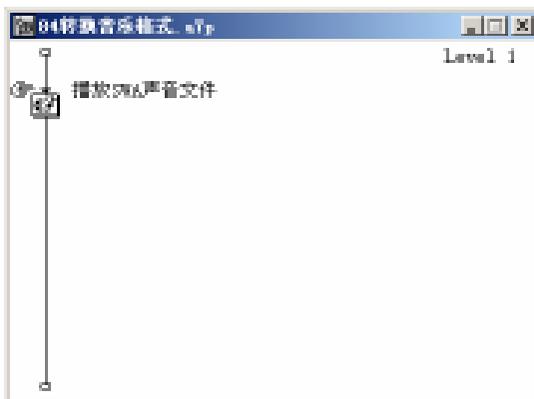


图 84.8 流程线

实例

85

移动打靶 (1)



实例说明

该实例建立一个移动打靶的效果，输入靶位号，靶移动到靶位处，子弹击中靶位。

主要知识点：交互图标、热区交互、动画图标。



操作步骤

① 新建文件，保存为“85 移动打靶.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 85.1 所示。

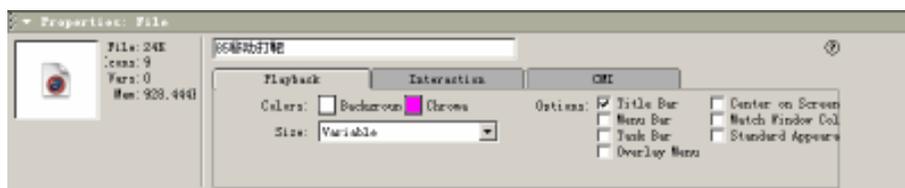
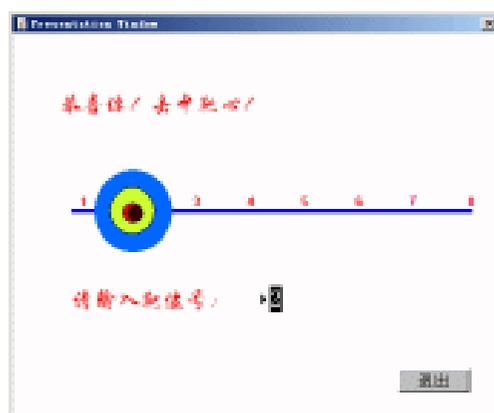


图 85.1 设置文件属性

③ 在流程线中添加 2 个 Display (显示) 图标，分别命名为“靶”和“靶位”，双击“靶”图标，使用绘图工具箱在演示窗口中绘制靶，如图 85.2 所示。



图 85.2 绘制靶



④ 双击“靶位”图标，打开演示窗口，首先绘制一条直线，并在直线上标明靶位号“1~8”，如图 85.3 所示。

⑤ 在流程线中添加一个 Interaction（交互）图标，命名为 Interaction（交互），在其右侧添加一个 Group（群组）图标，交互类型设置为 TextEntry，命名为“*”，表示可以接受任何字符。在 Interaction（交互）图标右侧添加一个 Calculation（计算）图标，交互类型设置为 Button，并命名为“退出”，流程线如图 85.4 所示。

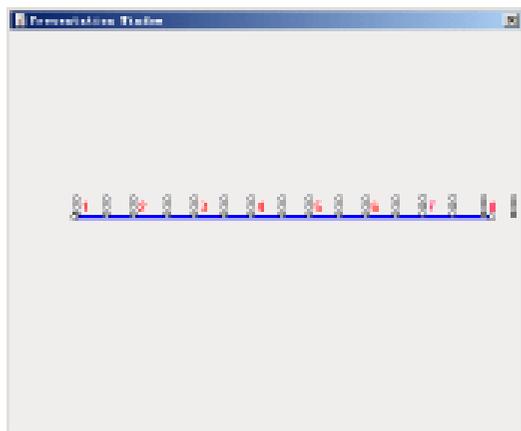


图 85.3 绘制靶位

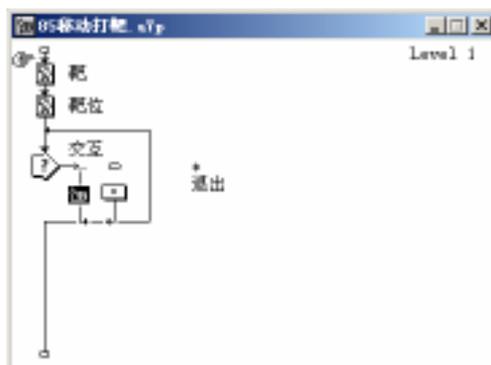


图 85.4 流程线

⑥ 单击“*”图标上面的交互类型标记，打开其属性对话框，选择 Space 复选框，这样就可以忽略输入字符的空格、大小写、顺序等，如图 85.5 所示。

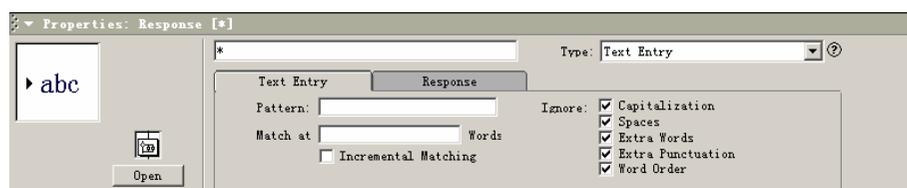


图 85.5 设置“*”交互属性

⑦ 单击“退出”按钮，打开其属性对话框，Branch 设置为 Exit Interaction，其他选项默认，如图 85.6 所示。

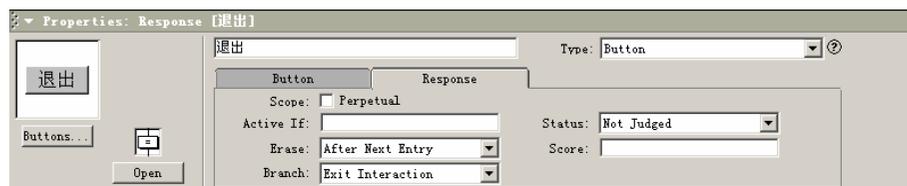


图 85.6 设置“退出”交互属性

实例

86

移动打靶 (2)



实例说明

该实例建立一个移动打靶的效果，输入靶位号，靶移动到靶位处，子弹击中靶位。

主要知识点：交互图标、热区交互、动画图标。



操作步骤

① 接上例，双击“*”群组图标，打开二级窗口，添加图标，并命名，如图 86.1 所示。

② 双击“子弹”图标，打开演示窗口，使用绘图工具箱中的椭圆工具绘制子弹，并设置填充色，如图 86.2 所示。

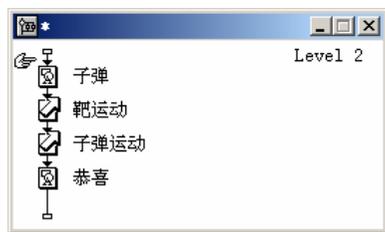
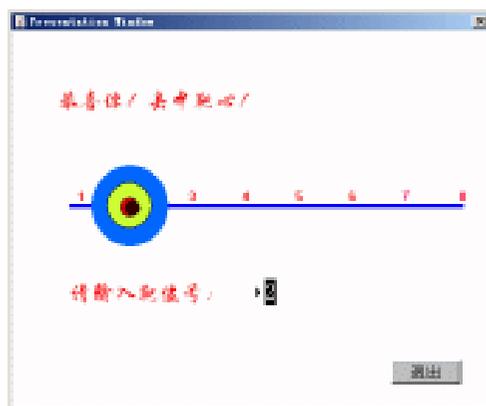


图 86.1 添加图标

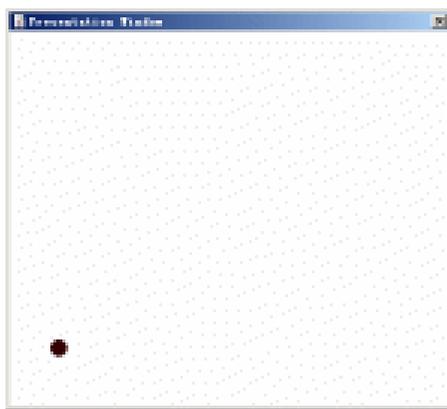


图 86.2 绘制子弹

③ 单击“靶移动”图标，打开动画属性对话框，在演示窗口中单击靶，将靶作为移动对象，在 Type 下拉菜单中选择 Direct to Line，在 Timing 下拉菜单中选择 Time (sec)，并在下面的文本框中输入“1”，表示在 1 秒内完成该运动。Concurrency 设置为 Wait Until Done，Beyond Range 设置为 Stop at Ends，如图 86.3 所示。

④ 选择 Base 单选按钮，在 X 栏中输入“1”，将靶移动到 1 号靶位处；选择 End 单



选按钮，在 X 栏中输入“8”，将靶移动到 8 号靶位处；选择 Destination 单选按钮，在 X 栏中输入 NumEntry，如图 86.3 所示。

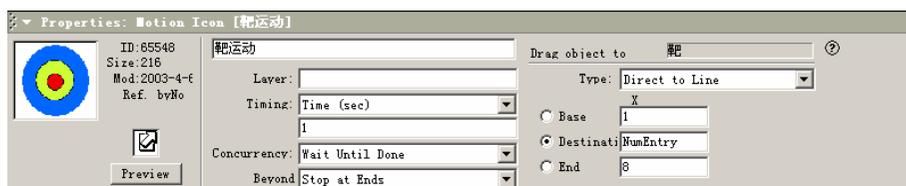


图 86.3 设置“靶”运动属性

⑤ 双击“子弹移动”图标，打开运动属性对话框，在演示窗口中单击子弹，将子弹作为移动对象，在 Type 下拉菜单中选择 Direct to Line，在 Timing 下拉菜单中选择 Time (sec)，并在下面的文本框中输入“1”，表示在 1 秒内完成该运动。Concurrency 设置为 Wait Until Done，Beyond Range 设置为 Stop at Ends，如图 86.4 所示。

⑥ 选择 Base 单选按钮，在 X 栏中输入“1”，将子弹移动到 1 号靶位处；选择 End 单选按钮，在 X 栏中输入“8”，将子弹移动到 8 号靶位处；选择 Destination 单选按钮，在 X 栏中输入 NumEntry，如图 86.4 所示。

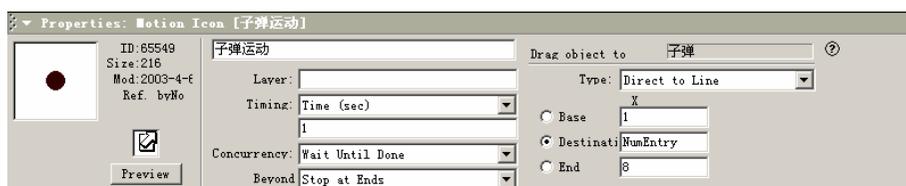


图 86.4 设置“子弹”运动属性

⑦ 双击“恭喜”图标，打开演示窗口，使用文本工具输入“恭喜你，击中靶心”，并设置文本的属性，如图 86.5 所示。右键单击该图标，选择 Properties 目录，打开其属性对话框，单击 Transition 后面的按钮，选择过渡方式，增加文字的动态效果，如图 86.6 所示。用户击中靶心后，将显示该文本。



图 86.5 输入恭喜文本

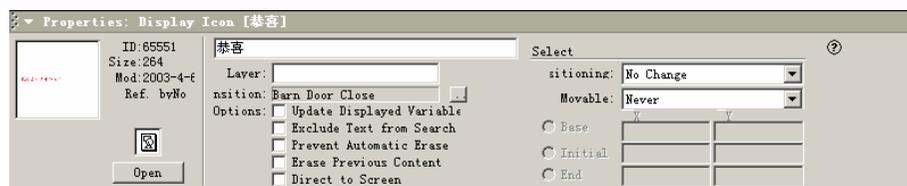


图 86.6 设置过渡显示属性

⑧ 回到主程序窗口中，双击“退出”计算图标，输入代码 `Quit(0)`，如图 86.7 所示。

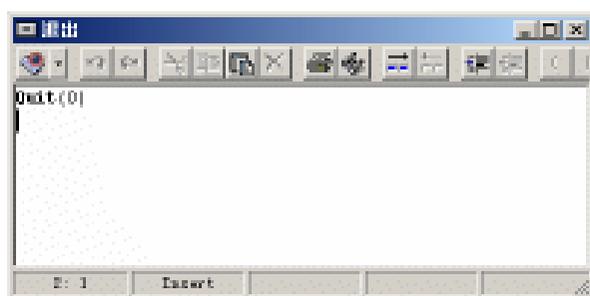


图 86.7 输入代码

⑨ 存盘，运行，程序流程线如图 86.8 所示。

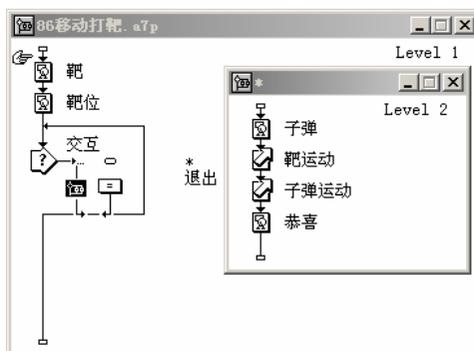


图 86.8 流程线



实例

87

控制电影播放 (1)



实例说明

该实例建立一个控制电影播放的应用程序,单击界面上的控制按钮,可以实现电影的播放、暂停、前进、后退等操作。

主要知识点:电影图标、交互图标、按钮交互、Authorware 变量和函数。

控制电影播放



操作步骤

① 新建文件,保存为“87 控制电影播放.a7p”。

② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 87.1 所示。

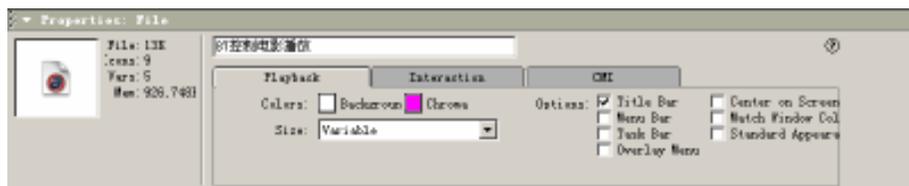


图 87.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标,命名为“背景”,打开演示窗口,使用文本工具 **A** 输入标题“控制电影播放”,并双击椭圆形工具 **O**,打开调色板,为文本着色。然后,执行命令 Text Font 设置字体为“华文行楷”,执行命令 Text Size,设置字体大小为“24”磅,双击箭头工具 **H**,设置文本的显示模式为 Transparent,如图 87.2 所示。

④ 在流程线上添加一个 Calculation (计算) 图标,命名为“初始化”,双击打开代码窗口,输入代码,如图 87.3 所示。

⑤ 在流程线上添加一个 Digital Movie (电影) 图标,命名为“电影”,双击打开属性对话框,单击 Import 按钮导入电影,将对话框切换到 Timing 选项卡内,Concurrency 设置为 Concurrent,Play 设置为 Fixed Number of Times,并在下面的文本框中输入“1”,StartFrame

设置为 startframe , EndFrame 设置为 endframe , 如图 87.4 所示。

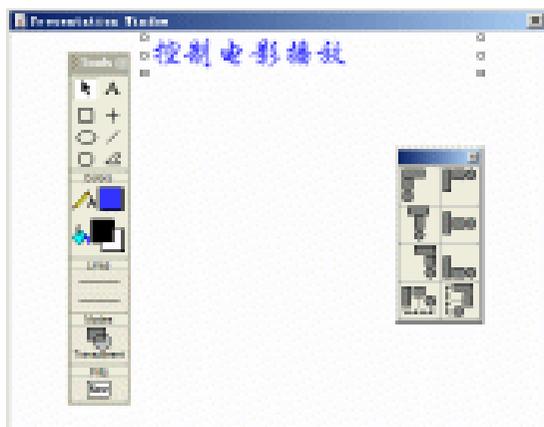


图 87.2 设置背景

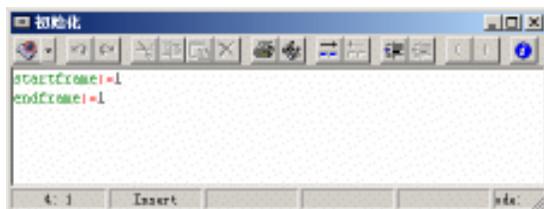


图 87.3 初始化变量

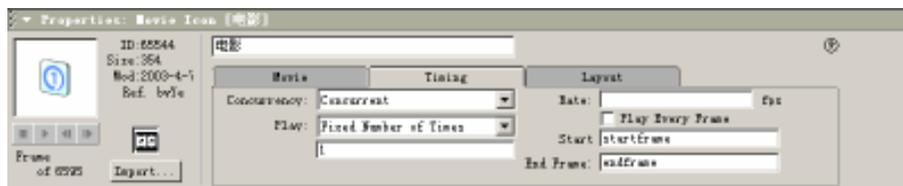


图 87.4 设置电影图标属性

⑥ 在流程线中添加一个 Interaction (交互) 图标, 在其右侧添加 5 个 Calculation (计算) 图标, 交互属性均设置为 Button, 并命名, 如图 87.5 所示。

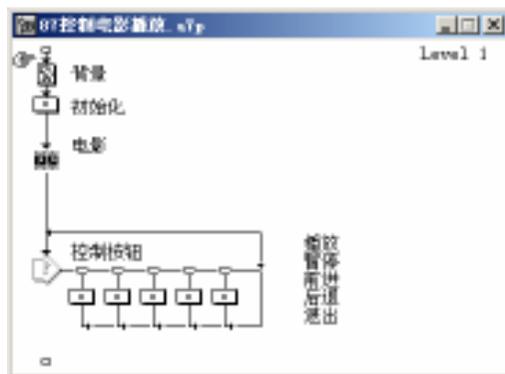


图 87.5 流程线



⑦ 双击“播放”按钮上面的交互类型标记，打开属性对话框，在 Response 选项卡内 Erase 设置为 After Next Entry，Branch 设置为 Try Again。在 Button 选项卡内设置光标形状为小手状，如图 87.6 和图 87.7 所示。同理设置其他 4 个图标的交互属性。



图 87.6 设置按钮交互属性的 Response 属性

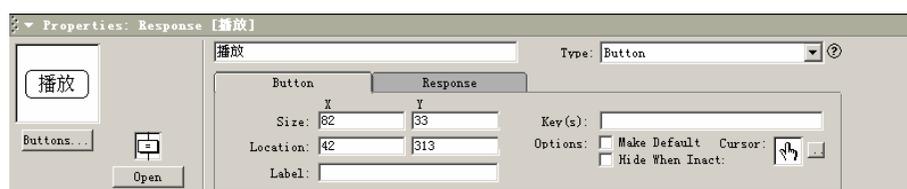


图 87.7 设置按钮交互属性的 Button 属性

⑧ 在按钮交互属性对话框中，单击 Buttons 按钮，打开 Buttons 对话框，设置按钮的样式，如图 87.8 所示，同理设置其他 4 个按钮的样式。



图 87.8 设置按钮样式

⑨ 双击“播放”图标，输入下列代码：

```
startframe:=MediaPosition@"电影"
endframe:=MediaLength@"电影"
MediaPlay(IconID@"电影")
```

⑩ 双击“暂停”图标，输入下列代码：

```
MediaPause(IconID@"电影",TRUE)
```

⑪ 双击“前进”图标，输入下列代码：

```
startframe:=MediaPosition@"电影"
endframe:= MediaPosition@"电影"+1
MediaPlay(IconID@"电影")
```

⑫ 双击“后退”图标，输入下列代码：

```
startframe:=MediaPosition@"电影"
endframe:= MediaPosition@"电影"-1
MediaPlay(IconID@"电影")
```

⑬ 双击“退出”图标，输入代码 Quit(0)。

⑭ 双击 Interaction (交互) 图标，打开其演示窗口，按住 Shift 键选择 5 个按钮，执行命令 Modify Align，打开 Align 面板，排列这 5 个按钮，如图 87.9 所示。

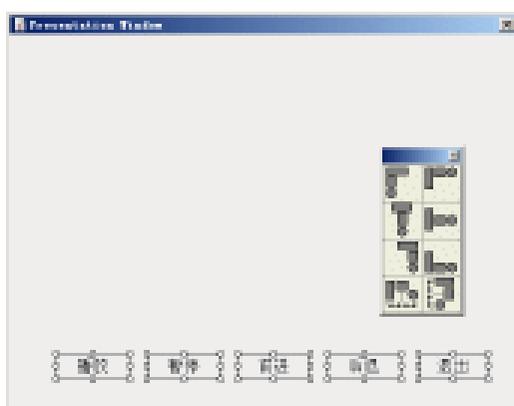


图 87.9 排列按钮

⑮ 存盘，运行，程序流程线如图 87.10 所示。

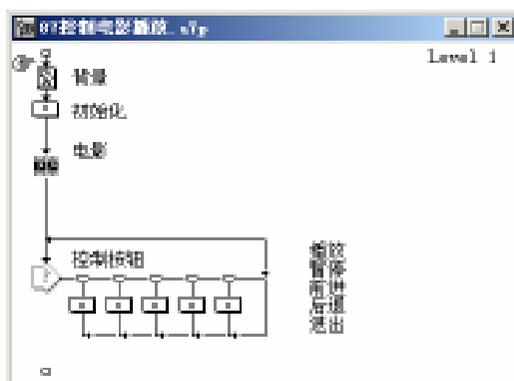


图 87.10 流程线



实例

88

控制电影播放 (2)



实例说明

该实例使用 Authorware 知识对象建立一个应用程序，用于控制电影的播放，如有图所示。

主要知识点：Movie Controller 知识对象。

控制电影播放



操作步骤

- ① 新建文件，保存为“88 控制电影播放.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 88.1 所示。

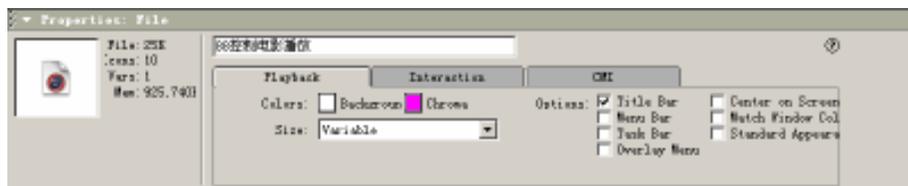


图 88.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在左侧画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“控制电影播放”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 88.2 所示。

④ 执行命令 Windows Knowledge Objects，打开知识对象窗口，在 Category 下拉菜单中选择 All，双击下面列表中的 Movie Controller 知识对象，如图 88.3 所示。

⑤ 打开知识对象向导，首先是该对象的介绍画面，如图 88.4 所示，直接单击 Next 按钮进入下一个画面。

⑥ 进入第二个画面，单击 FileName 文本框后面的  按钮选择电影文件，电影加载后，在窗口右下角可以预览，如图 88.5 所示。

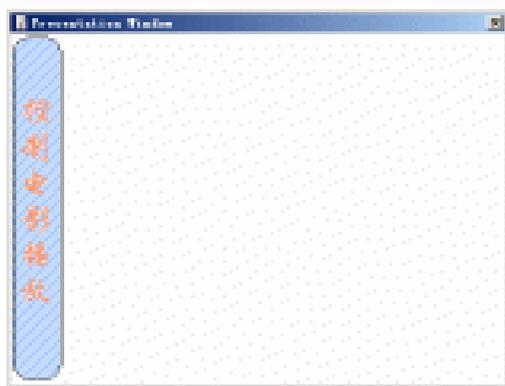


图 88.2 设置背景

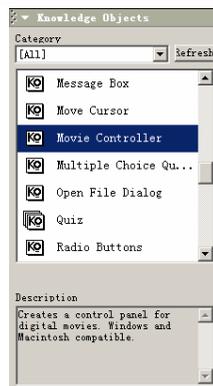


图 88.3 选择知识对象

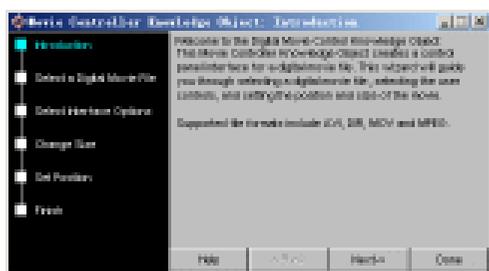


图 88.4 向导 1

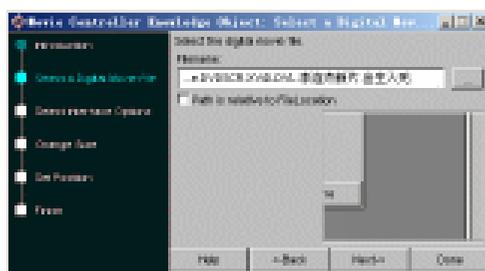


图 88.5 向导 2

⑦ 单击 Next 按钮进入第 3 个画面，这里设置控制电影的按钮，选择这些复选框，如图 88.6 所示。

⑧ 单击 Next 按钮进入第 4 个画面，这里设置电影的尺寸，在 Set Size to 下面的文本框中输入电影屏幕的大小，这些选项可以不用设置，等会运行程序后，根据实际情况，可以更方便地进行调整。在 Resize By 下面的文本框中输入“100”；“1000”，并选择 Proportional 复选框，电影等比例进行缩放，右侧窗口中的 Adjust 可以对屏幕的尺寸大小进行微调，如图 88.7 所示。

⑨ 单击 Next 按钮进入第 5 个画面，这里设置电影的位置，这些选项可以不用设置，等会运行程序后，根据实际情况，可以更方便地进行调整，如图 88.8 所示。

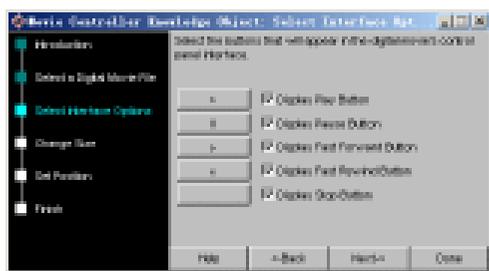


图 88.6 向导 3

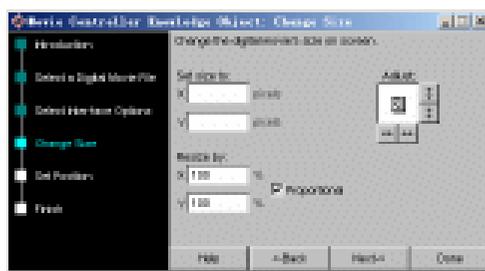


图 88.7 向导 4

⑩ 单击 Next 按钮进入最后一个画面，单击 Done 按钮结束向导，如图 88.9 所示。

⑪ 此时，在流程线中自动添加 Movie Controller Knowledge Object 知识对象图标，重



命名为“控制电影”，如图 88.10 所示。

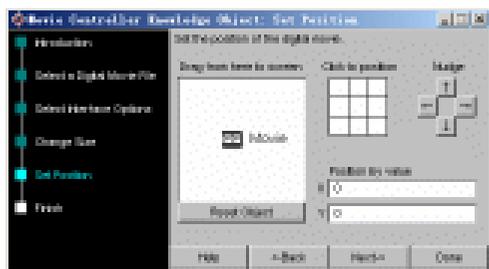


图 88.8 向导 5

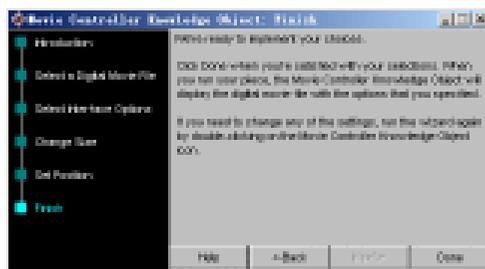


图 88.9 向导结束画面

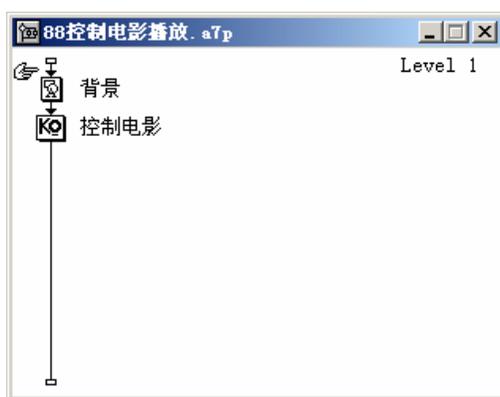


图 88.10 流程线

实例

89

登录程序



实例说明

该实例使用知识对象建立一个登录界面程序，当用户输入正确的登录信息，允许用户进入下一步操作，并给出信息提示。

主要知识点：Login 知识对象。



操作步骤

- ① 新建文件，保存为“89 登录程序.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 89.1 所示。

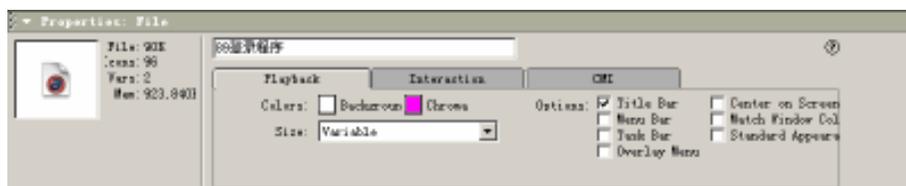


图 89.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“登录程序”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 89.2 所示。

④ 执行命令 Windows Knowledge Objects，打开知识对象窗口，在 Category 下拉菜单中选择 All，双击下面列表中的 Login 知识对象，如图 89.3 所示。

⑤ 打开知识对象向导，首先是该对象的介绍画面，如图 89.4 所示，直接单击 Next 按钮进入下一个画面。

⑥ 进入第二个画面，设置信息显示方式，选择默认的 Text File (All Platforms)，如图 89.5 所示。

⑦ 单击 Next 按钮进入第 3 个画面，这里设置记录属性，单击  按钮选择存放记录信息的路径。选择下面的几个复选框，设置用户输入 ID、密码、接受新用户注册，并在



Maxium Logins 文本框重输入最大登录次数，如图 89.6 所示。

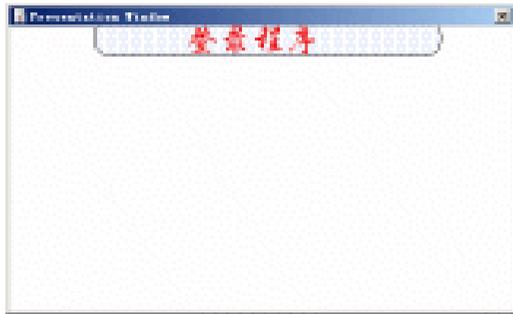


图 89.2 设置背景

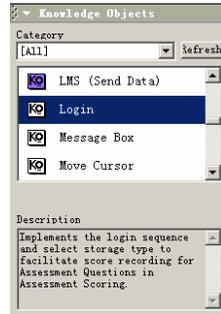


图 89.3 选择知识对象



图 89.4 向导 1

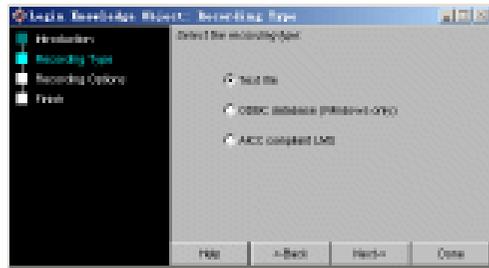


图 89.5 向导 2

8 单击 Next 按钮进入最后一个画面，单击 Done 按钮结束向导，如图 89.7 所示。

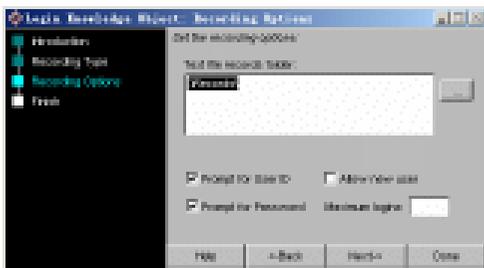


图 89.6 向导 3

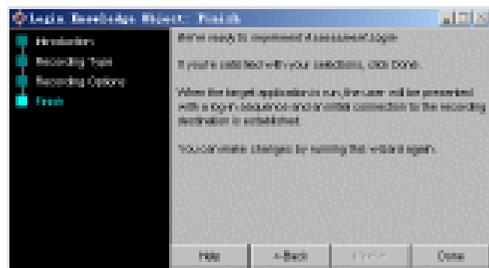


图 89.7 向导结束画面

9 此时，在流程线中自动添加 Login 知识对象图标，如图 89.8 所示。



图 89.8 流程线

⑩ 在流程线中添加一个 Display(显示)图标,命名为 welcome,双击打开演示窗口,输入欢迎信息,如图 89.9 所示。当用户登录成功后显示该图标内容。为了增添动态气氛,最好给该显示图标添加过渡显示效果。



图 89.9 输入欢迎信息

⑪ 存盘,程序流程图如图 89.10 所示。



图 89.10 流程线

⑫ 运行,如果是第一次登录,系统提示注册为新用户,并将注册信息存储在指定路径下,如图 89.11 所示。如果输入密码错误,显示错误提示,如图 89.12 所示。

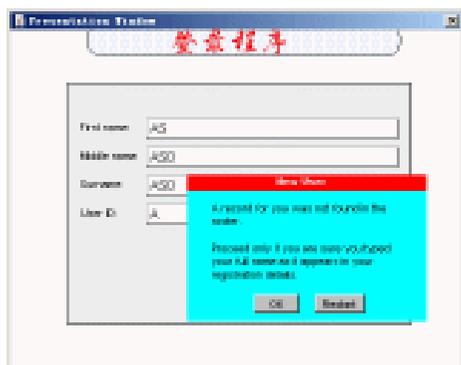


图 89.11 新用户注册

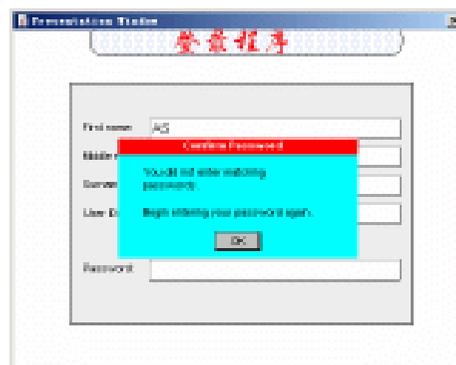


图 89.12 登录失败



实例

90

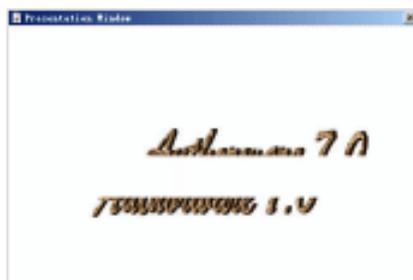
文本对接



实例说明

该实例建立一个文本对接的程序,程序运行后,文本的两部分同时运动,最后拼接成完整的艺术字效果。

主要知识点:创建艺术字、动画图标。



操作步骤

- 1 新建文件,保存为“90 文本对接.a7p”。
- 2 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 90.1 所示。



图 90.1 设置文件属性

- 3 在 Microsoft Word 2002 中完成艺术字,并粘贴在图像编辑软件中,如图 90.2 所示。

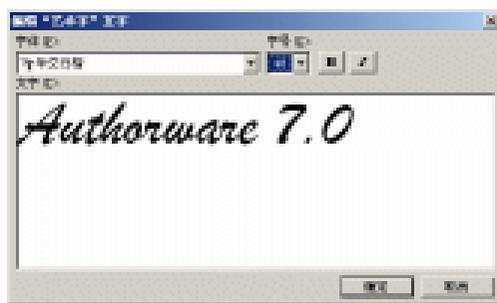


图 90.2 在 Word 中编辑艺术字

- 4 在图像编辑软件中,将艺术字切割成上下两部分,分别保存。
- 5 在 Authorware 流程线中添加 2 个 Display(显示)图标,分别命名为“图 1”和“图 2”。

2”。双击“图 1”图标，打开演示窗口，执行命令 File Import，导入艺术字图像的上半部分，并将图像移动到窗口的右上角，如图 90.3 所示；同理，在“图 2”图标中导入上艺术字图像的下半部分，并将图像移动到窗口的左下角，如图 90.4 所示。



图 90.3 在“图 1”中导入图像

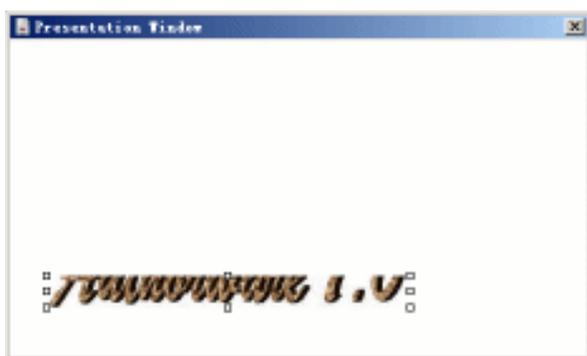


图 90.4 在“图 2”中导入图像

⑥ 在 Authorware 流程线中添加 2 个 Motion(动画)图标 ，分别命名为“移动图 1”和“移动图 2”，单击“移动图 1”图标，打开运动属性对话框，单击图 1 作为运动对象，在 Type 下拉菜单中选择 Direct to Point，Timing 设置为 Time (sec)，并在下面的文本框中输入“0.5”，Concurrency 设置为 Concurrent，如图 90.5 所示，并在演示窗口中将图移动到适当的位置。同理，设置“移动图 2”的运动属性，如图 90.6 所示，移动图 2 的位置，将图 2 和图 1 拼接成完整的艺术字。

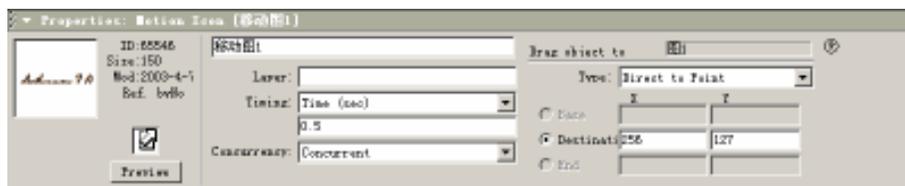


图 90.5 设置“移动图 1”运动属性

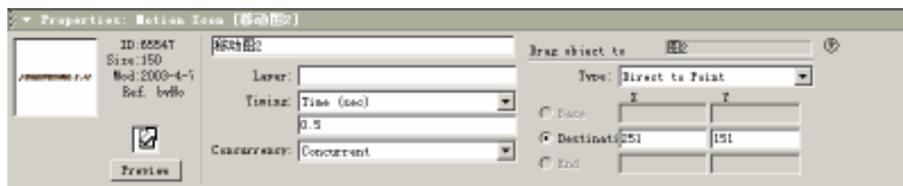


图 90.6 设置“移动图 2”运动属性

⑦ 存盘，运行，程序流程线如图 90.7 所示，运动结束后，拼接成完整的艺术字。



图 90.7 流程线

实例

91

网址簿(1)



实例说明

该实例建立一个简单的网址簿应用程序，单击界面上网站的名称，将链接到相应的网站。

主要知识点：交互图标、热区交互、知识对象。



操作步骤



① 新建文件，保存为“91 网址簿.a7p”。

② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 91.1 所示。

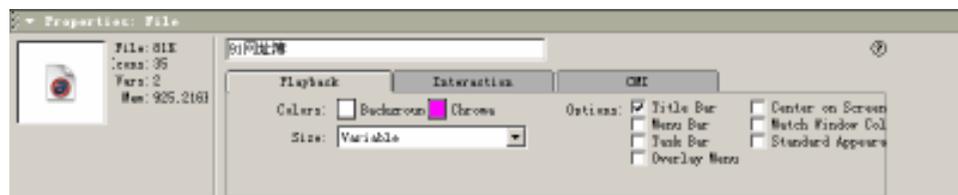


图 91.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的圆角矩形工具  在顶部画出圆角矩形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“网址簿”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 91.2 所示。

④ 在 Microsoft Word 中创建表格，并在表格中输入网站名称，如图 91.3 所示，将其复制到剪贴板中。

⑤ 在流程线上添加一个 Display (显示) 图标，命名为“网址”，双击打开演示窗口，执行命令 Edit Paste Special，打开 Paste Special 对话框，选择 Microsoft Word 文档，如图 91.4 所示，将上面创建的表格粘贴在演示窗口中。



图 91.2 设置背景

搜 狐	网 易
新 浪	ChinaRen
太平洋电脑	中关村在线
清华大学	北京邮电大学

图 91.3 在 Microsoft Word 中创建表格



图 91.4 粘贴表格

⑥ 为了防止“网址”图标中的表格发生移动，输入代码予以控制。右键单击图标，选择 Calculation 命令，输入代码 `Movable:=False`，如图 91.5 所示。

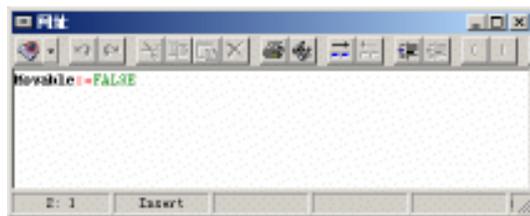


图 91.5 输入控制代码

实例

92

网址簿 (2)



实例说明

该实例建立一个简单的网址簿应用程序,单击界面上网站名称,将链接到相应的网站。

主要知识点:交互图标、热区交互、知识对象。



操作步骤

- ① 接上例,在流程线上添加一个 Interaction (交互) 图标,命名为“打开网址”。
- ② 执行命令 Window Knowledge Objects,打开知识对象窗口,在 Category 下拉菜单中选择 All,然后双击下面列表中的 Launch Default Browser 知识对象,如图 92.1 所示。
- ③ 打开知识对象向导,首先是介绍画面,如图 92.2 所示,直接单击 Next 按钮进入下一个画面。



图 92.1 双击选择的知识对象

④ 打开第 2 个画面,选择 Absolute 单选按钮,在上面的 URL 文本框中输入网址 <http://www.sohu.cn>,如图 92.3 所示。

⑤ 单击 Next 按钮进入第 3 个画面,设置当打开链接时是否退出 Authorware,不需要选择 Quit Authorware on Jump 复选框,如图 92.4 所示。单击 Next 按钮进入最后一个画面,



单击 Done 按钮结束向导，如图 92.5 所示。

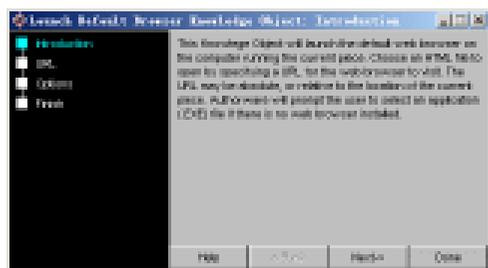


图 92.2 知识对象向导

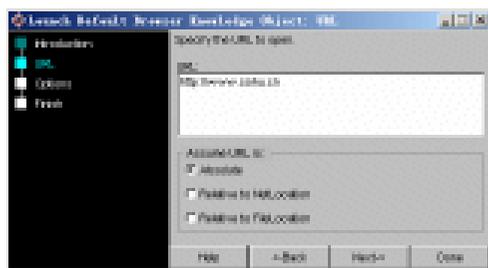


图 92.3 输入网址

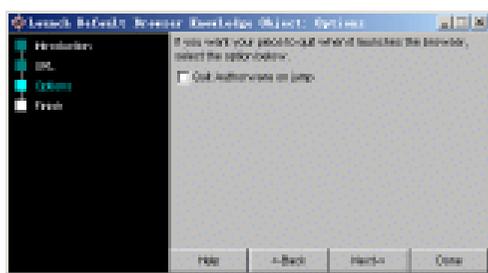


图 92.4 知识对象向导



图 92.5 结束向导

⑥ 此时，在流程线上自动添加了知识对象图标，重新命名为 sohu，并将此图标拖到 Interaction（交互）图标的右侧，交互类型设置为 Hot Spot，如图 92.6 所示。

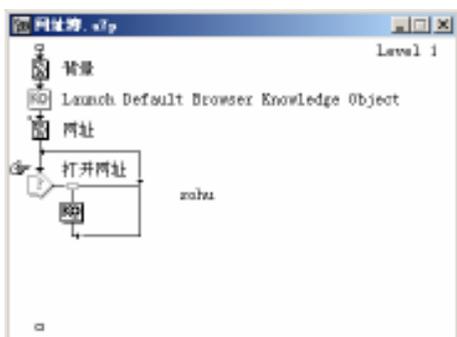


图 92.6 流程图

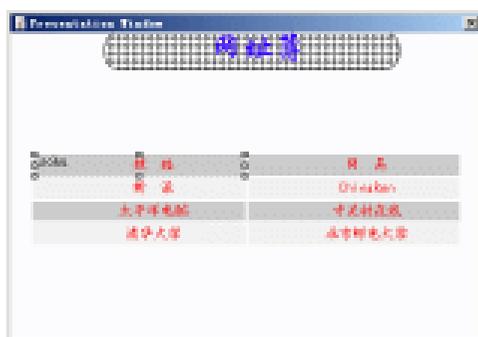


图 92.7 定义热区

⑦ 单击知识对象图标上面的交互类型标记，打开属性对话框，在演示窗口中拖动虚线框定义交互区域，如图 92.7 所示。在属性对话框中，Erase 设置为 After Next Selection，Branch 设置为 Try Again，在 Button 选项卡内设置光标形状，如图 92.8 和图 92.9 所示。

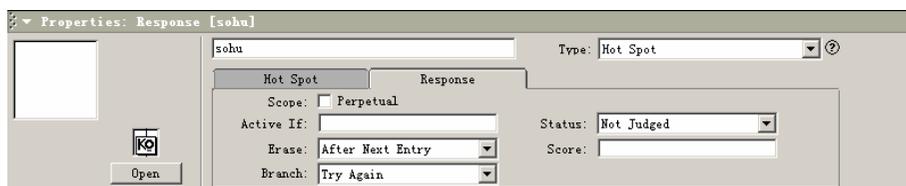


图 92.8 设置交互属性的 Response 选项卡

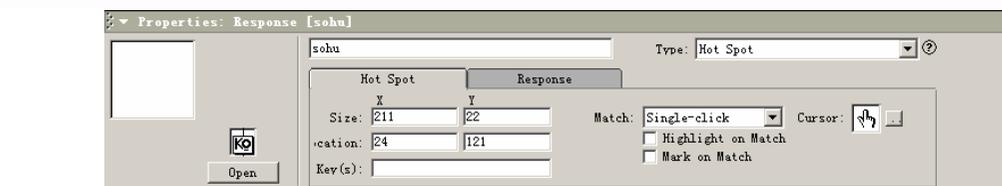


图 92.9 设置交互属性 Button 选项卡

⑧ 使用同样的方法，再添加 7 个知识对象，并设置其交互属性。



技巧

可以通过复制的方法添加其他的知识对象，再修改其名称、URL、交互区域等属性。

⑨ 存盘，运行，程序流程线如图 92.10 所示。

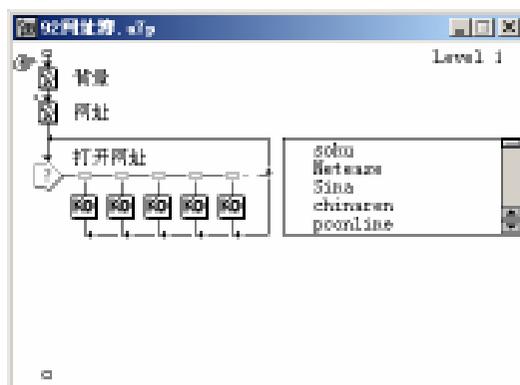


图 92.10 流程线



实例

93

Flash 播放器 (1)



实例说明

该实例建立一个播放 Flash 影片的应用程序，单击界面中的“打开”按钮，打开文件目录选择 Flash 影片，单击其他按钮对影片的播放进行控制。

主要知识点：交互图标、知识对象、ActiveX 控件。



操作步骤

- ① 新建文件，保存为“93Flash 播放器.a7p”。
- ② 执行命令 Modify File Properties，打开属性对话框，设置 Size 为 Variable，取消选择 Menu Bar 复选框，如图 93.1 所示。



图 93.1 设置文件属性

③ 在流程线上添加一个 Display (显示) 图标，命名为“背景”，双击打开演示窗口，使用绘图工具箱中的椭圆工具  在顶部画出椭圆形区域，双击圆角矩形工具 ，打开填充面板，设置填充方案，并双击椭圆形工具 ，打开调色板，为填充图案着色。然后，使用文本工具  输入标题“Flash 播放器”，执行命令 Text Font 设置字体为“华文行楷”，执行命令 Text Size，设置字体大小为“24”磅，同时设置字体颜色，双击箭头工具 ，设置文本的显示模式为 Transparent，如图 93.2 所示。

④ 在流程线上添加一个 Interaction (交互) 图标，命名为“按钮控制”，并在其右侧添加一个 Group (群组) 图标，交互类型设置为 Button，并命名为“打开”，双击该图标上面的交互类型标记，打开属性对话框，选择 Perpetual 复选框，Erase 设置为 Don't Erase，Branch 设置为 Return，如图 93.3 所示。

⑤ 双击“打开”图标，打开二级窗口。执行命令 Window Knowledge Objects，打开知识对象窗口，选择 Category 下拉菜单中的 All 选项，在下面的列表中查找并双击 Open

File Dialog 对象，如图 93.4 所示。

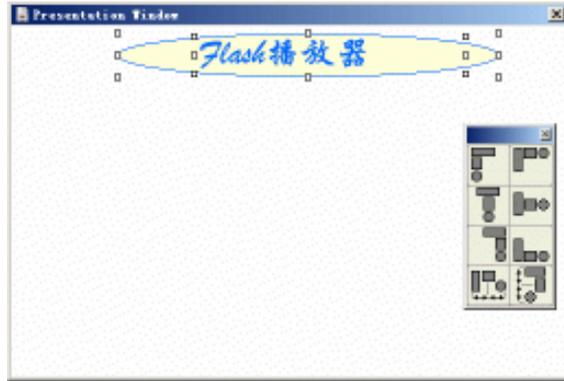


图 93.2 设置背景

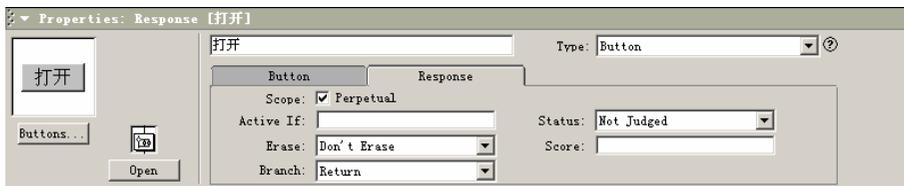


图 93.3 设置按钮交互属性



图 93.4 选择知识对象

⑥ 打开知识对象向导，首先是该对象的介绍画面，如图 93.5 所示，直接单击 Next 按钮进入下一个画面，在 Dialog Title 文本框中输入对话框的标题，如图 93.6 所示。

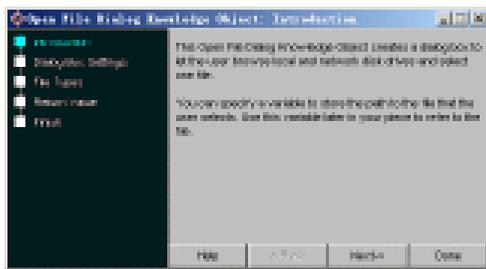


图 93.5 知识对象向导

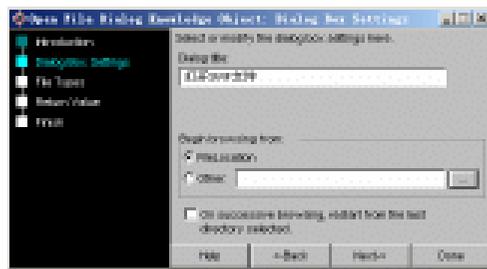


图 93.6 输入对话框标题



⑦ 单击 Next 按钮进入第 3 个画面，这里设置打开的文件类型，选择 Flash 前面的复选框，如图 93.7 所示。

⑧ 单击 Next 按钮进入第 4 个画面，在等号的后面输入 swffile，将打开的 swf 文件的路径信息储存在此变量中，如图 93.8 所示。

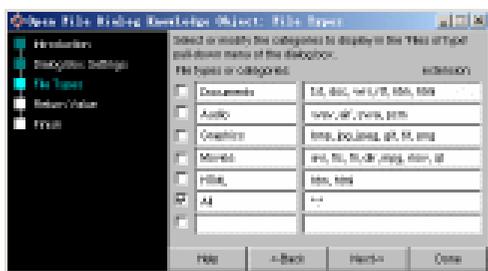


图 93.7 选择文件类型

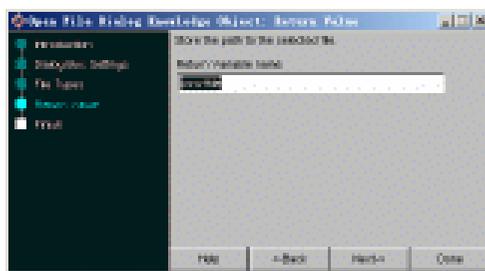


图 93.8 输入变量名称

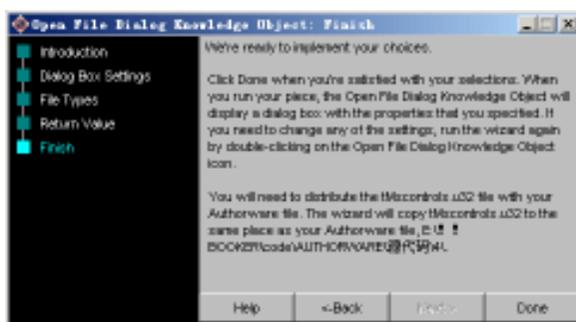


图 93.9 向导结束画面

⑨ 单击 Next 按钮进入最后一个画面，提示将 tMscontrols.u32 拷贝到程序所在目录，单击 Done 按钮结束向导，如图 93.9 所示。

⑩ 此时，流程图中自动添加一个知识对象图标，重命名为“打开 SWF 文件”，如图 93.10 所示。



图 93.10 重命名图标

实例

94

Flash 播放器(2)



实例说明

该实例建立一个播放 Flash 影片的应用程序，单击界面上的“打开”按钮，打开文件目录选择 Flash 影片，单击其他按钮对影片的播放进行控制。

主要知识点：交互图标、知识对象、ActiveX 控件。



操作步骤

① 接上例，在“打开”二级窗口中，执行命令 Insert Control ActiveX，打开 Select ActiveX Control 对话框，在列表中选择 Shockwave Flash Object，单击 Ok 按钮，插入控件，如图 94.1 所示。在流程线中自动添加控件图标，重命名为 flash。

② 打开图 94.2 所示的控件属性对话框，选择默认设置即可，单击 OK 按钮退出对话框。

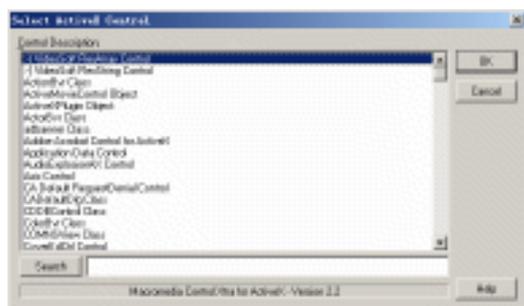


图 94.1 选择插入控件

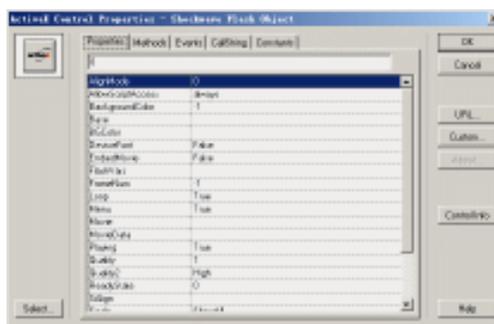


图 94.2 控件属性对话框

③ 在“打开”二级窗口中添加一个 Calculation (计算) 图标，命名为 SetFile，双击打开代码窗口，输入代码 SetSpriteProperty(@"flash", #movie, swfile)，如图 94.3 所示。

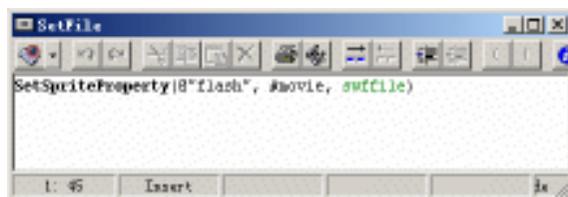


图 94.3 输入代码



④ 回到主流程图，在 Group（群组）图标右侧添加 3 个 Calculation（计算）图标，交互类型和属性均与 Group（群组）图标相同，如图 94.4 所示。

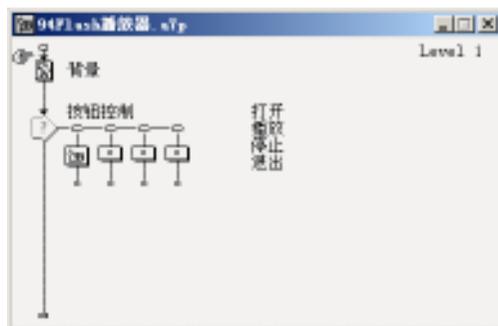


图 94.4 流程线

⑤ 双击“播放”计算图标，打开代码窗口，输入下列代码：

```
CallSprite(@"flash", #play)
```

⑥ 双击“停止”计算图标，打开代码窗口，输入下列代码：

```
CallSprite(@"flash", #stop)
```

⑦ 双击“退出”计算图标，打开代码窗口，输入代码 `Quit(0)`。

⑧ 双击 Interaction（交互）图标，打开演示窗口，按住 Shift 键选择 4 个按钮，执行命令 `Modify Align`，打开 Align 面板，排列这 4 个按钮，如图 94.5 所示。

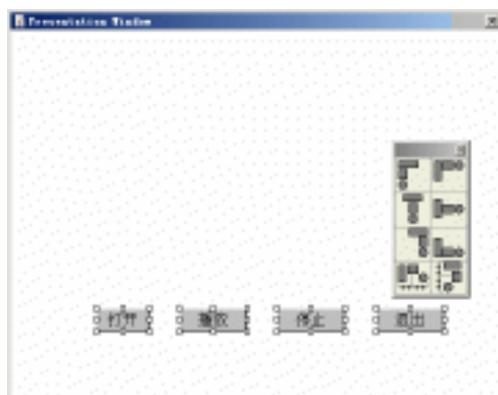


图 94.5 排列按钮

实例

95

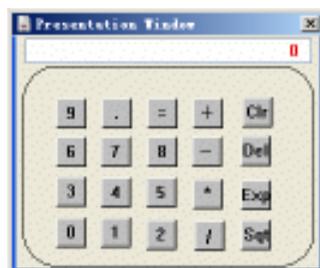
计算器 (1)



实例说明

该实例建立一个计算器应用程序,界面上有 20 个功能按钮,可以进行简单的加减乘除、乘方、开方运算。

主要知识点:交互图标、按钮交互、变量和函数。



操作步骤

- ① 新建文件,保存为“95 计算器.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 95.1 所示。



图 95.1 设置文件属性

- ③ 在流程线中添加一个 Display (显示) 图标,命名为“背景”,双击打开演示窗口,执行命令 File Import 导入实现准备好的计算器背景图案,并调整窗口的大小,如图 95.2 所示。

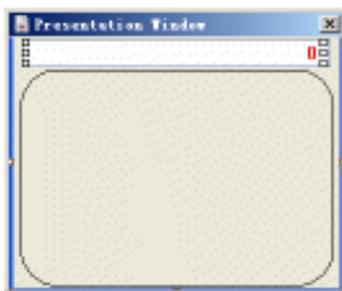


图 95.2 导入背景图案

- ④ 右键单击“背景”图标,选择命令 Calculation,打开代码窗口,输入代码 Movable:



=False，防止背景发生移动。

⑤ 双击“背景”图标，打开演示窗口，在背景中显示数字结果的地方输入变量{X}，并设置文本的属性，执行命令 Text Alignment Right，设置文本为右对齐，如图 95.3 所示。



图 95.3 输入变量

⑥ 按住 Ctrl 键双击“背景”图标，打开属性设置框，选择 Update Displayed Variables 复选框，及时更新显示的变量，如图 95.4 所示。

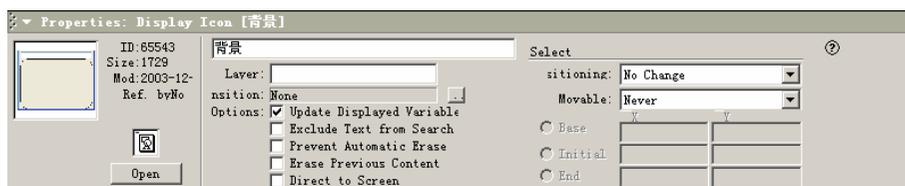


图 95.4 设置显示图标属性

⑦ 在流程线中添加一个 Interaction (交互) 图标，命名为“按钮”，在其右侧添加 20 个 Calculation (计算) 图标，交互类型均设置为 Button，分别命名为“0、1、2、3、4、5、6、7、8、9、.、+、-、*、/、=、Clr、Del、Exp、Sqrt”，如图 95.5 所示。

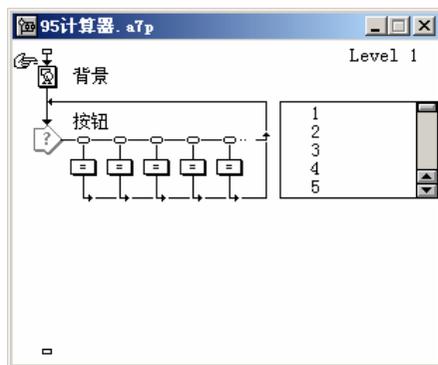


图 95.5 添加 20 个计算图标

实例

96

计算器 (2)



实例说明

该实例建立一个计算器应用程序,界面上有 20 个功能按钮,可以进行简单的加减乘除、乘方、开方运算。

主要知识点:交互图标、按钮交互、变量和函数。



操作步骤

① 接上例,双击“1”计算图标,打开代码窗口,输入代码“ $X:=X^1$ ”,如图 96.1 所示。

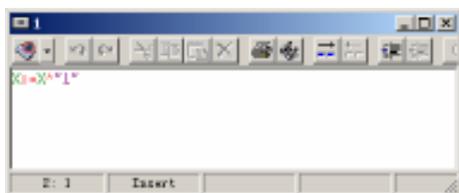


图 96.1 输入代码



说明

“^”是 Authorware 中比较常见的运算符,可以将它两边的字符串连接起来。上述代码“ $X:=X^1$ ”表示将“1”连接到变量 X 存储的字符串后面。

② 使用同样的方法在“0、1、2、3、4、5、6、7、8、9、.、+、-、*、/”等图标对应的窗口中输入代码,代码和前面相似。

例如在“*”图标中输入代码“ $X:=X^*$ ”。

③ 双击“=”计算图标,在代码窗口中输入下列代码:

```
X1:=Replace("^","*",X)
X1:=Eval(X1)
if ABS(X1>9999999999) then X1:="Error"
X1:=Replace(","," ",X1)
```



X:=X1

- ④ 双击 Clr 计算图标，在代码窗口中输入代码 “ X:="" ”。
- ⑤ 双击 Del 计算图标，在代码窗口中输入代码 “ X:=SubStr(X,1,CharCount(X)-1) ”。
- ⑥ 双击 Exp 计算图标，在代码窗口中输入代码 “ X:=X^^ ”。
- ⑦ 双击 Sqt 计算图标，在代码窗口中输入代码 “ X:=SQRT(“^X^”) ”。
- ⑧ 双击 1 图标上面的交互类型标记，打开交互属性对话框，在 Button 选项卡内，设置 Size 为 “ 25, 25 ”，在 Keys 栏中设置键值 “ 1 ”，就是说，按键盘上的 “ 1 ” 键也可以达到同样的效果，如图 96.2 所示。同理，设置其他 19 个按钮的属性。

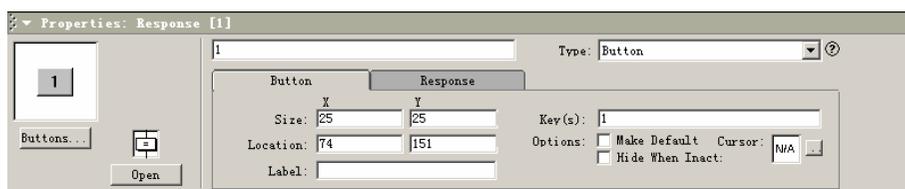


图 96.2 设置按钮属性

- ⑨ 双击 “ 背景 ” 图标，再按住 Shift 键双击 Interaction (交互) 图标，打开演示窗口，按住 Shift 键选择 20 个按钮，选择命令 Modify Align，打开 Align 面板，排列 20 个按钮，如图 96.3 所示。



图 96.3 排列按钮

- ⑩ 存盘，运行，程序流程线如图 96.4 所示。

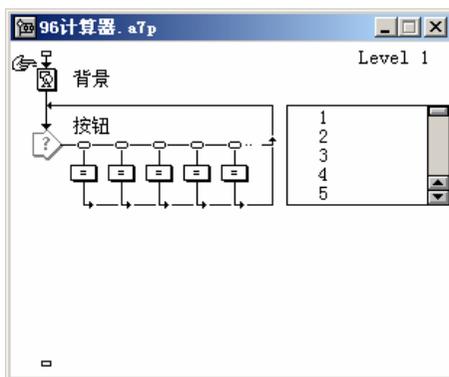


图 96.4 流程线

实例

97

检测窗口大小(1)



实例说明

该实例建立一个检测窗口大小的应用程序,拖动窗口边缘,改变窗口大小,程序显示的当前窗口的宽度和高度。

主要知识点:交互图标、热区交互、系统变量。



操作步骤

- ① 新建文件,保存为“97 检测窗口大小.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 97.1 所示。

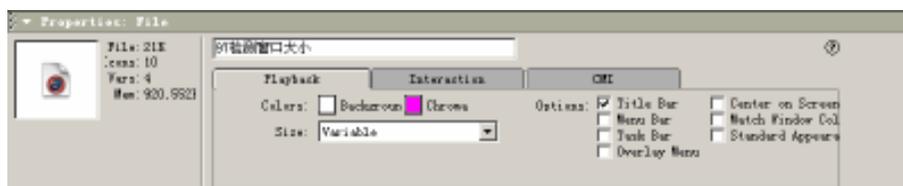


图 97.1 设置文件属性

- ③ 在流程线中添加一个 Interaction (交互) 图标,命名为“调整窗口大小”,在其右侧添加 3 个 Group(群组)图标,交互类型均设置为 Hot Spot,分别命名为 Y resize、X resize 和 XY resize,如图 97.2 所示。

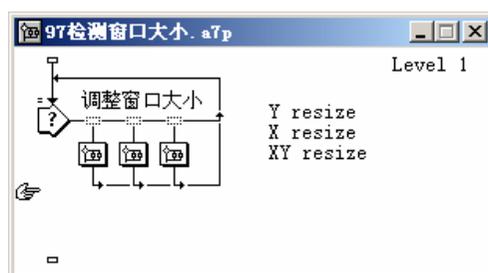


图 97.2 添加图标

- ④ 右键单击 Interaction (交互) 图标,从快捷菜单中选择命令 Calculations,打开代



码窗口，输入代码：SetCursor(0)，如图 97.3 所示。

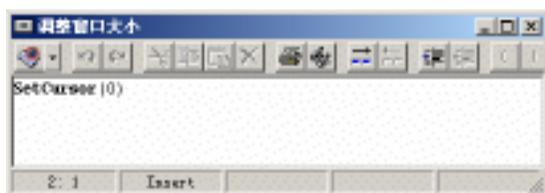


图 97.3 输入代码

⑤ 双击 Y resize 图标上面的交互类型标记，打开属性对话框，选择 Perpetual 复选框，如图 97.4 所示。将对话框切换到 Hot Spot 选项卡内，在 Size 栏设置为 WindowWidth-20 和 20，Location 设置为 0 和 WindowHeight-20，如图 97.5 所示。



图 97.4 定义交互区域

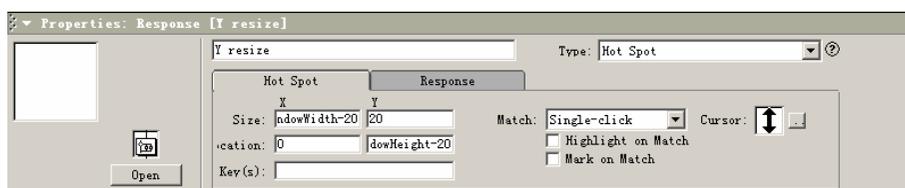


图 97.5 设置热区属性

⑥ 在 Hot Spot 选项卡内，Match 设置为 Single-click，单击 Cursor 后面的按钮，打开 Cursors 对话框，选择光标形状，如图 97.6 所示。如果想要的光标不在列表中，单击 Add 按钮，从 Windows 系统目录下的 Cursors 文件夹中加载光标，如图 97.7 所示。

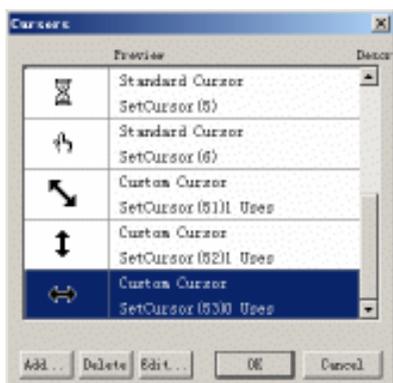


图 97.6 设置光标



图 97.7 加载光标

⑦ 双击 X resize 图标上面的交互类型标记，打开属性对话框，在 Response 选项卡内

选择 Perpetual 复选框，如图 97.8 所示。在 Hot Spot 选项卡内，在 Size 栏设置为 20 和 WindowWidth-20，Location 设置为 WindowWidth-20 和 0，Match 设置为 Single-click，并设置光标形状，如图 97.9 所示。

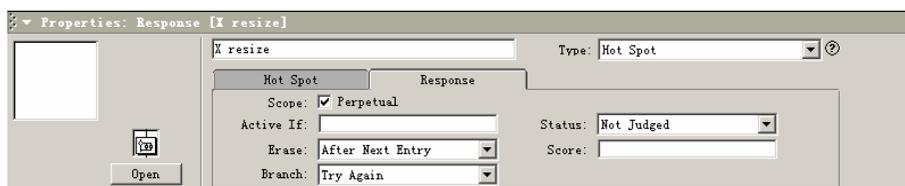


图 97.8 定义交互区域

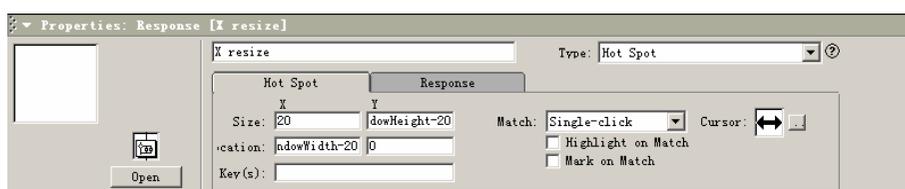


图 97.9 设置热区属性

⑧ 双击 XY resize 图标上面的交互类型标记，打开属性对话框，在 Response 选项卡内选择 Perpetual 复选框，如图 97.10 所示。在 Hot Spot 选项卡内，在 Size 栏设置为 20 和 20，Location 设置为 WindowWidth-20 和 WindowWidth-20，Match 设置为 Single-click，并设置光标形状，如图 97.11 所示。

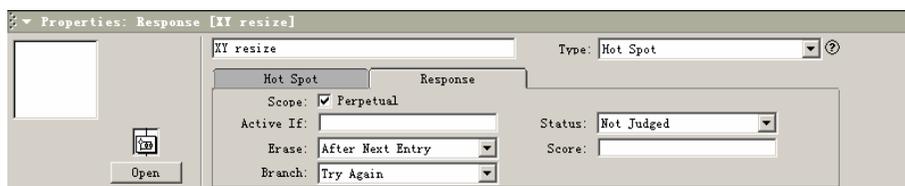


图 97.10 定义交互区域

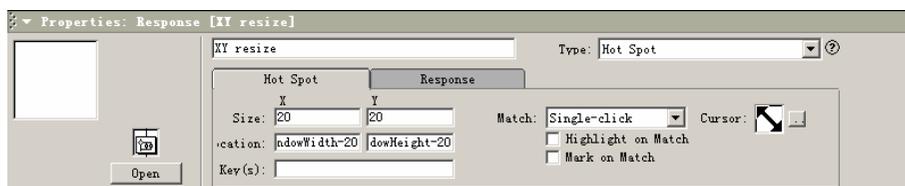


图 97.11 设置热区属性



实例

98

检测窗口大小 (2)



实例说明

该实例建立一个检测窗口大小的应用程序，拖动窗口边缘，改变窗口大小，程序显示当前窗口的宽度和高度。

主要知识点：交互图标、热区交互、系统变量。



操作步骤

① 接上例，双击 Interaction (交互) 图标，打开演示窗口，输入文本“窗口宽度={WindowWidth} 窗口高度={WindowHeight}”，并设置文本的属性，如图 98.1 所示。

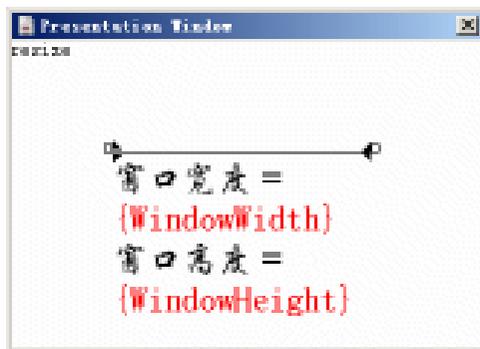


图 98.1 输入显示文本

② 双击 Y resize 群组图标，打开二级窗口，在二级窗口中添加一个 Decision (判断) 图标，并在在其右侧添加一个 Calculation (计算) 图标，命名为 resize，如图 98.2 所示。

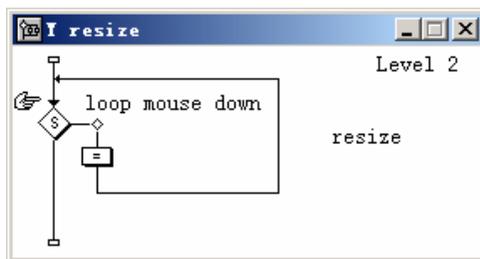


图 98.2 添加图标

③ 双击 Decision (判断) 图标, 打开属性对话框, Repeat 设置为 Until True, 在下面的文本框中输入 “~MouseDown”, Branch 设置为 Sequentially, 如图 98.3 所示。



图 98.3 设置分支图标属性

④ 双击 Decision (判断) 图标下面的 Calculation (计算) 图标, 打开代码窗口, 输入下列代码:

```
ResizeWindow(WindowWidth, CursorY)
SetCursor(52)
```

⑤ 重复步骤 3~4, 在 X resize 和 XY resize 群组图标中添加图标, 并设置其属性。

在 X resize 图标中的 Calculation (计算) 图标中输入下列代码:

```
ResizeWindow(CursorX, WindowHeight)
SetCursor(53)
```

在 XY resize, Calculation (计算) 图标中输入下列代码:

```
ResizeWindow(CursorX, CursorY)
SetCursor(51)
```

⑥ 存盘, 运行, 程序流程图如图 98.4 所示。

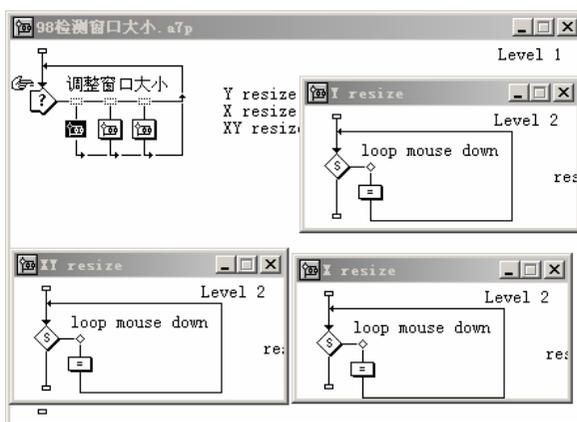


图 98.4 流程线



实例

99

数据库操作 (1)



实例说明

该实例演示数据库操作,不仅可以显示数据库中的记录,而且可以从数据库中查询到需要的记录。

主要知识点:函数和变量、交互图标、交互类型。



操作步骤

- ① 新建文件,保存为“99 数据库操作.a7p”。
- ② 执行命令 Modify File Properties,打开属性对话框,设置 Size 为 Variable,取消选择 Menu Bar 复选框,如图 99.1 所示。



图 99.1 设置文件属性

- ③ 打开 Microsoft Access,新建一个空白数据库,保存为 Data,打开数据表设计向导,定义 id、name、sex、year 等字段 将 id 设置为主键,如图 99.2 所示 将数据表保存为 user.mdb。



图 99.2 创建数据库

④ 回到 Authorware 设计环境中, 执行命令 Window Functions, 打开 Functions 对话框, 在 Category 下拉菜单中选择“99 数据库操作.a7p”, 单击 Load 按钮, 打开 Load Functions 对话框, 选择 Authorware 安装目录下的 ODBC.u32, 如图 99.3 所示。单击“打开”按钮, 打开图 99.4 所示的对话框, 按住 Ctrl 键选择 3 个函数, 单击 Load 按钮。

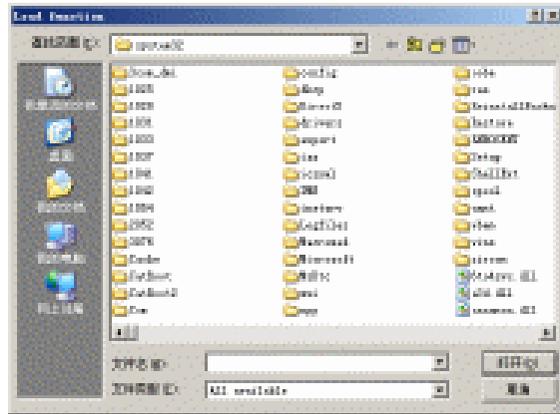


图 99.3 导入函数

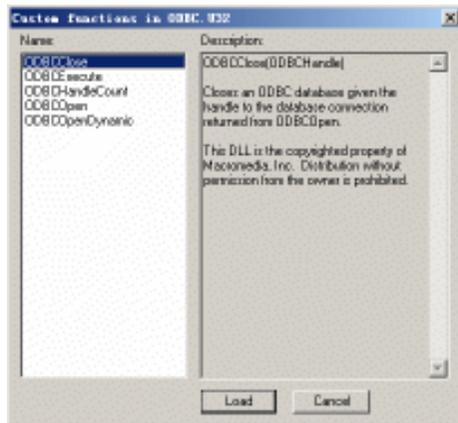


图 99.4 函数对话框

⑤ 在流程线中添加一个 Calculation (计算) 图标, 命名为“配置数据源”。双击打开代码窗口, 输入代码配置数据源, 如图 99.5 所示。

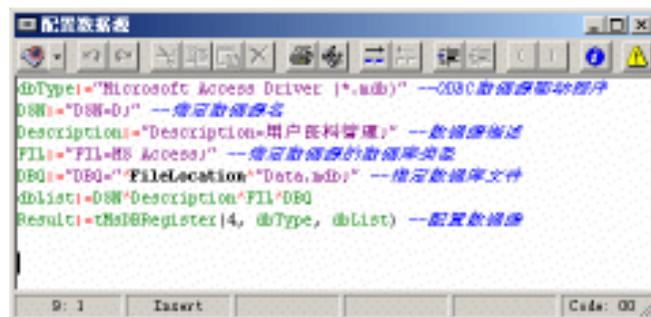


图 99.5 输入代码



输入的代码如下：

```
dbType:="Microsoft Access Driver (*.mdb)" --ODBC 数据源驱动程序
DSN:=" DSN= D;" --指定数据源名
Description:="Description=用户信息;" --数据源描述
FIL:="FIL=MS Access;" --指定数据源的数据库类型
DBQ:="DBQ="^FileLocation^"Data.mdb;" --指定数据库文件
dbList:=DSN^Description^FIL^DBQ
Result:=tMsDBRegister(4, dbType, dbList) --配置数据源
```

关闭代码窗口时，系统提示选择函数 tMsDBRegister () 的来源，选择 Authorware 安装目录下的 tMsDSN.u32。

⑥ 在流程线中添加一个 Calculation (计算) 图标，命名为“打开数据库”。双击打开代码窗口，输入下列代码：

```
ODBCHandle:=ODBCOpen(WindowHandle, "", "D", "", "")
getRecord:="Select count(id) from user;"
record:=ODBCExecute(ODBCHandle, getRecord)
```

⑦ 在流程线中添加一个 Interaction (交互) 图标，命名为“数据库操作”，在其右侧添加 2 个 Group (群组) 图标和 1 个 Calculation (计算) 图标，并命名，如图 99.6 所示。

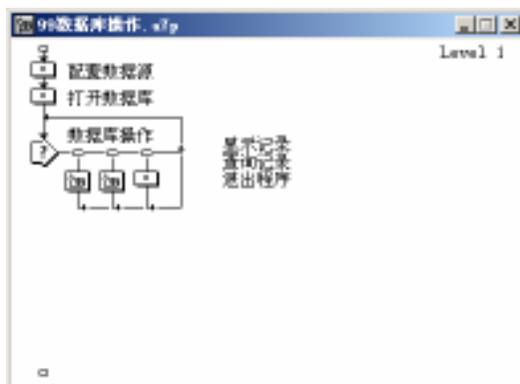


图 99.6 添加图标

实例

10

数据库操作(2)



实例说明

该实例演示数据库操作,不仅可以显示数据库中的记录,而且可以从数据库中查询到需要的记录。

主要知识点:函数和变量、交互图标、交互类型。



操作步骤

① 接上例,此时的流程图如图 100.1 所示,双击流程图中的 Interaction(交互)按钮,打开演示窗口,输入文本标题“数据库操作”,并设置文本的属性,调整按钮的位置。在窗口的右上角,输入文本“共有{record}条记录”,如图 100.2 所示。

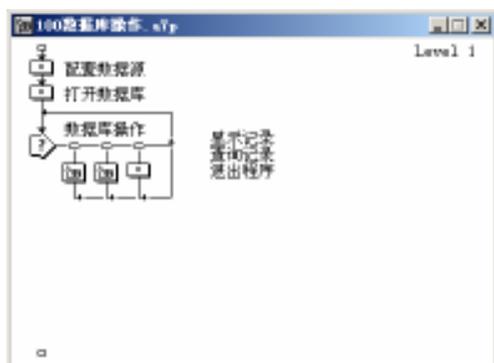


图 100.1 流程线



图 100.2 设置交互图标内容

② 双击“显示记录”图标,打开二级窗口,在窗口中添加图标,并命名,如图 100.3 所示。

③ 在“显示记录”二级窗口中,双击“擦除”图标,将界面上的按钮设置为擦除对象,如图 100.4 所示。

④ 在“显示记录”二级窗口中,双击“读取记录”图标,输入下列代码:

```
SQLstring1:="Select id from user order by id"
SQLstring2:="Select name from user order by id"
SQLstring3:="Select sex from user order by id"
SQLstring4:="Select year from user order by id"
```



```

ODBCData1:=ODBCExecute(ODBCHandle, SQLstring1)
ODBCData2:=ODBCExecute(ODBCHandle, SQLstring2)
ODBCData3:=ODBCExecute(ODBCHandle, SQLstring3)
ODBCData4:=ODBCExecute(ODBCHandle, SQLstring4)
ODBCClose(ODBCHandle)
    
```

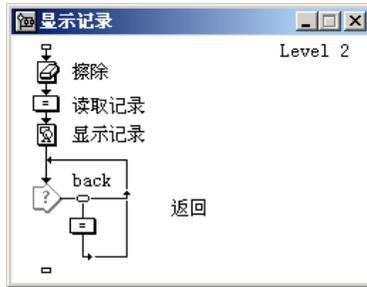


图 100.3 “显示记录”二级窗口



图 100.4 设置擦除属性

⑤ 在“显示记录”二级窗口中，双击“显示记录”图标，打开演示窗口，制作图 100.5 所示的显示窗口，并在相应的位置输入记录名称。

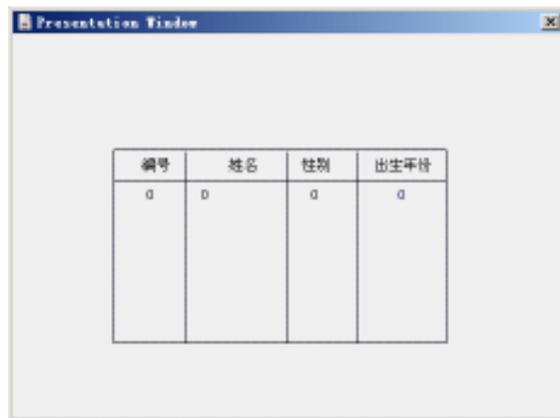


图 100.5 显示记录

⑥ 在“显示记录”二级窗口中，双击 Interaction (交互) 图标下面的 Calculation (计算) 图标，输入代码 GoTo(IconID@"打开数据库")。

⑦ 回到主流程图中，双击“查询记录”图标，打开二级窗口，添加图标，并命名，如图 100.6 所示。

⑧ 在“查询记录”二级窗口中，“擦除”图标的设置方法和前面一样，不再赘述。双

击“背景”图标，打开演示窗口，输入文本，如图 100.7 所示。



图 100.6 “查询记录”二级窗口



图 100.7 输入文本

⑨ 在“查询记录”二级窗口中，设置 Interaction (交互) 图标下的 Group (群组) 图标的交互类型为 Text Entry，命名为“*”，双击该图标，打开三级窗口，添加图标，并命名，如图 100.8 所示。

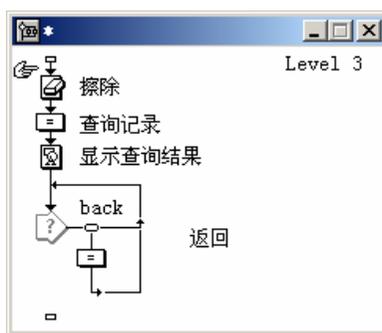


图 100.8 “*”三级窗口

⑩ 在“*”三级窗口中，双击 Erase (擦除) 图标，将背景和输入框中的内容设置为擦除对象，如图 100.9 所示。

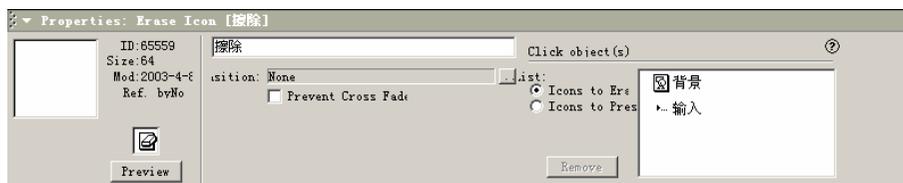


图 100.9 设置擦除对象

⑪ 在“*”三级窗口中，双击“查询记录”图标，输入下列代码：

```
X:=EntryText
SQL1:="select id from user where name="'^X'^"
SQL2:="Select name from user where name="'^X'^"
SQL3:="Select sex from user where name="'^X'^"
SQL4:="Select year from user where name="'^X'^"
```



```

record1:=ODBCExecute(ODBCHandle, SQL1)
record2:=ODBCExecute(ODBCHandle, SQL2)
record3:=ODBCExecute(ODBCHandle, SQL3)
record4:=ODBCExecute(ODBCHandle, SQL4)
ODBCClose(ODBCHandle)
if record2="" then
    hint:="没有与"^X"^相关的记录！"
    MessageBox(WindowHandle, hint, title, 16)
    GoTo(IconID@"打开数据库")
end if
    
```

⑫ 在“*”三级窗口中，双击“显示查询结果”图标，使用绘图工具绘制表格，并输入文本，将数据库中查询的字段添加到表格中对应的地方。例如，在“姓名”下面的单元格中输入{record2}，如图 100.10 所示。



图 100.10 显示查询结果

⑬ “*”三级窗口中的交互结构和前面的一样，不再赘述。

⑭ 存盘，运行，程序流程线如图 100.11 所示。感兴趣的读者可以扩展该程序，实现添加记录和删除记录的功能板块，这里不再赘述。



图 100.11 流程线