

文章编号: 1005—8893 (2005) 01—0040—04

基于 Microsoft Agent 的 ActiveX 编程技术研究^{*}

薛 磊

(江苏工业学院 计算机科学与工程系, 江苏 常州 213016)

摘要: Microsoft Agent 以强大的交互功能、鲜明的人性特点、优美的操作界面和简便统一的编程方法, 对传统人机交互方式产生了很大影响。它既可以被加入到普通应用程序中供本地系统使用, 也可以嵌入到 HTML 文档中供 WEB 应用程序使用。Microsoft Agent 提供了 OLE 自动化服务器和 ActiveX 控件两种编程方法。文章从此入手, 研究了 Agent 控件的相关对象和方法, 并介绍了在 DELPHI 中调用 Agent 控件完成动画、语音合成和识别的详细方法。

关键词: 代理软件; ActiveX 技术; OLE 自动化服务器

中图分类号: TP 311.132

文献标识码: A

在人工智能领域, agent 被认作是一个实体, 它通过传感器感知环境, 通过效应器作用于环境, 具有自主性、协作性、交互性、主动性、反应性等特点^[1]。

Microsoft Agent 是微软公司于 1997 年 9 月发布的一项代理软件开发技术, 目前已有多种版本。它提供了生动的动画角色和丰富的交互方式, 既可以被加入到普通应用程序中供本地系统使用, 也可以嵌入到 HTML 文档中供 WEB 应用程序使用。通过编程, MS Agent 可具有根据用户的操作步骤“智能化”作出反应的能力, 在多媒体创作、教育软件制作、软件帮助系统等多方面具有广阔的应用前景。

1 基于 MS Agent 的编程方法

MS Agent 采用 OLE COM/ActiveX 结构开发体系, 为程序员提供了 OLE 自动化服务器和 ActiveX 控件两种编程方法。

1.1 OLE 自动化服务器

展现 OLE 自动化对象的应用程序, 称为 OLE 自动化服务器。OLE 基于 COM, COM 一方面可

以看作是一种编写基于组件系统的可重用软件的规范; 另一方面也是一种允许用户和对象跨越进程和机器的边界进行通信的复杂的基础结构。在体系结构上, COM 的接口和实现分离, 对象内部的实现可以使用任何语言, 接口则定义了一组公共的方法而不涉及任何实现^[2]。MS Agent 提供多种 COM 接口, 可以直接利用接口编程。

1.2 ActiveX 控件

ActiveX 控件是应用 OLE 技术的一种可重用的软件组件, 这种组件可以嵌入到 Web 页面中使用, 也可用于桌面应用程序和软件开发工具。ActiveX 控件屏蔽了 OLE 技术的一些细节, 使用支持 ActiveX 的编程工具可以方便地实现调用。本文以一个理财软件帮助系统的实现为例介绍 Delphi 中 MS Agent 的 ActiveX 控件编程方法。

2 MS Agent 的运行环境

MS Agent 可运行在 Windows95 以上的环境中, 但通常需要具备以下组件: ①MS Agent 的核心组件 Microsoft Agent Control; ②完成从文字合成语音工作的 Lemout & Hauspie TruVoice Text—

^{*} 收稿日期: 2004—10—14

基金项目: 江苏工业学院科技基金资助

作者简介: 薛磊 (1969—), 女, 山东青岛人, 讲师。

to—Speech Engine; ③提供语音识别功能的 Microsoft Command & Control Speech Engine; ④Agent 动画人物的数据文件。

3 MS Agent 的 ActiveX 编程技术

3.1 MS Agent 的动画人物

MS Agent 提供了生动的动画角色, 如吉尼 (Genie)、默林 (Merlin)、罗比 (Robby) 等, 每个动画人物都经过三维处理, 具有丰富的表情和数十种独特的动作。不仅如此, 用户还可以用 Agent Character Editor 来自行创建符合自己需要的个性化人物。

3.2 加载和卸载动画人物数据

在使用动画人物之前, 必须首先加载动画人物数据。微软定义了 Characters 对象作为角色的集合, 要访问某个角色必须先将该角色的数据载入到这个集合中, 这通过调用 IagentCtlCharacters 的 Load () 方法来完成。

以加载 Merlin 为例: Agent1. Characters. Load ('Merlin', 'd:\winnt\msagent\hars\Merlin.acs'); Load () 的第一个参数是给载入角色所起的名字, 第二个参数则是动画数据文件所在的路径或者 URL (如果数据文件在远程服务器上)。当某个动画人物使用结束后, 应当调用 Unload () 方法卸载:

```
Agent1. Characters. Unload ('Merlin');
```

3.3 MS Agent 的简单编程

在 MS Agent 编程过程中, 加载动画人物数据后, 可以使动画人物根据需要出现、隐藏、作出各种动作甚至会话。IagentCtlCharacters 的 Show () 和 Hide () 方法可分别用来显示和隐藏动画人物, Speak () 方法可激活 TTS 引擎朗读参数表中的文字。

```
Merlin; = Agent1. Characters. Character ('Merlin');
```

```
Merlin. Show (0); //Merlin 出现
```

```
Merlin. MoveTo (200, 200, 1); //Merlin 移动到指定位置
```

```
Merlin. Speak ('Hello, I am Merlin!', '');
```

```
Merlin. Hide (0); //Merlin 消失
```

```
Show () 与 Hide () 的参数表示动画人物显
```

示、隐藏的速度, 为 0 表示 Agent 飞入、飞出, 为 1 表示 Agent 直接出现、消失, 没有动画过程。

3.4 MS Agent 的进一步编程

3.4.1 MS Agent 的动画编程

MS Agent 提供的动画人物通常都具有一系列特定的动作形态, 比如声明、建议、悲伤、祝贺、寻找等等, 所有这些都可以通过 IagentCtlCharacters 的 Play () 方法进行调用, 只需要将 Play () 的参数设置为相应的字符串, 如 Announce, Suggest, Sad 等, 就能使其完成相应的动画动作。

如: Merlin. Play ('LookRight'); //Merlin 作出向右看的动作。

但是在这些动作形态中, 有一类比较特殊的动作, 它们的名字都以 -ing 结尾, 如 Processing、Reading 等等, 一旦执行便循环播放, 只有调用 Interrupt () 方法方可终止循环。不过所调用的 Interrupt () 方法只能是其它对象的, 否则会发生异常。

3.4.2 MS Agent 的多角色编程

由于 MS Agent 采用异步方式处理各个角色系列, 这使得系统在处理某个角色的同时能处理其它代码, 因此当应用中出现两个或两个以上角色时, 其动作并不会按照代码的顺序进行, 此时不同角色之间的动作的协调可以通过 Wait () 方法。下面的例子是 Merlin 先出来, 询问谁是下一个, 并作出查询的样子, 这时 Robby 出现, 终止了 Merlin 的查找行为, 并移到 Merlin 的身边。

```
Merlin. Show (0);
```

```
MerlinRequest1; =Merlin. Speak ('Hello, I am Mr. Merlin! Who will be the next one?', '');
```

```
MerlinRequest2; = Merlin. Play ('searching');
```

```
Robby. Wait (MerlinRequest1);
```

```
Robby. Show (0);
```

```
Robby. Speak ('It. is me.', '');
```

```
Robby. Interrupt (MerlinRequest2);
```

```
Robby. MoveTo (300, 200, 1);
```

这里调用 Robby 的 Wait () 方法, 使 Robby 在 Merlin 说完话后出现; 调用 Robby 的 Interrupt () 方法, 终止了 Merlin 的 Searching 动作。

程序中的 MerlinRequest1 和 MerlinRequest2 是 MS Agent 的 Request 对象的实例, 当一个客户端

应用调用了 Load、Get、Play 和 Speak 方法中的一个, 将会针对这个请求产生并返回一个 Request 对象。根据这个对象的 State 属性可以判断请求的完成情况, 以便协调 Agent 调用之间和其它程序代码间的关系。

4 MS Agent 语音合成及识别的实现

MS Agent 扩展了原有的鼠标、键盘等传统交互方式, 除有气球状消息框和输入提示条作为人机接口外, 还可以利用 TTS 语音合成引擎和 C & C 语音识别引擎进行语音合成、语音识别和播放文本文件, 使得用户能通过麦克风、声卡、音箱和系统进行交互, 这也是 MS Agent 构建人性化交互环境的重要方面。

4.1 语音合成

MS Agent 的语音功能基于微软的语音程序设计接口 (Speech API), 要实现语音合成, 必须首先安装 TTS 语音合成引擎, 它能把文字生成语音信号并由声卡和音箱播放出来。Agent 对象的语音合成功能通过调用 IagentCtlCharacters 的 Speak () 方法实现, 在调用 Speak () 方法时, Agent 对象自动加载与角色的 LanguageID 相符的引擎用作语音输出。下面是 1 个示例代码:

```
Merlin. Speak ( ' 您好, 我是您的理财助手, 有何吩咐?', ' d: \hello. wav' ); //在消息框中输出中文的同时, 对应输出 hello. wav 中存放的中文语音 (目前 MS Agent 还不支持中文)。
```

Speak () 有 2 个参数, 第 1 个参数是文本 (text), 即气球状消息框中的内容, 第 2 个参数是 1 个 WAV 文件的路径。如果装有相匹配的 TTS 引擎, 则只需要提供第 1 个参数, Agent 角色在气球状消息框中显示文本的同时会自动合成播放语音; 否则必须提供 2 个参数, 运行时 Agent 服务器直接播放 WAV 文件的内容, 并在气球状消息框中显示 text 参数的内容。如果给第 2 个参数传递的是带音节信息的 LWV 文件的路径, 可不在第 1 个参数中提供文字, 因为 LWV 文件中本身包括文字信息, 而且当使用 LWV 文件时, 动画人物的嘴形能与发音保持一致。这种格式的文件可由 Microsoft Agent Linguistic Information Sound Editing Tool 编辑 WAV 文件来生成。目前 MS Agent 还不支持多个人物混声的方式, 所以当出现多个角色都要说话时, 同样应该使用 Wait () 方法, 让他们一个一

个说。

4.2 语音识别

MS Agent 不仅能根据文字发出声音, 而且可以根据用户的语音指令完成某种操作, 不过必须安装 Command & Control 语音识别引擎, 它包含 SAPI 所需要的各种支持。MS Agent 控件为每个角色提供声音命令窗口, 用来列出该角色当前能够接受声音驱动的命令的名称, 这些命令不能用鼠标和键盘来选取, 只能通过语音。该窗口在使用 Open 命令或者将它的 Visible 属性设置为 True 时出现, 而使用 Close 命令或它的 Visible 属性设置为 False 时关闭。IagentCommands 的 Add () 方法可用来为命令窗口添加自定义命令。该方法有 5 个参数, 依次为 ConstName —— 语音命令的标识, Caption —— 命令窗口中的显示标题, Voice —— 命令对应的语句, 还有 Visible 和 Enable 用来设置命令的可见与可用。如:

```
Merlin. Commands. Add ( ' income', ' 本月收入', ' income', True, True);
```

当然要使自定义命令起作用, 还必须在 Agent 对象的 Command () 事件中编写代码来完成相应的功能^[3]。因为发布命令或者使用 Agent 的快捷菜单都激发 Agent 对象的 Command 事件。

5 调用实例

下面以个人理财系统中的一个应用说明在 Delphi 中的具体使用方法。

首先必须在 Delphi 环境中中添加 MS Agent (见图 1), 然后在窗体中添加 Agent 控件 (见图 2), 最后编写代码完成相应的功能 (见图 3)。



图 1 添加 MS Agent 后的工具栏

Fig 1 Toolbar with MS Agent

当用户有疑问时, 可单击帮助按钮, 启动帮助。代码为:

```
Procedure TForm1. BitBtn1Click (Sender: TOb-  
ject);  
begin  
Agent1. Characters. Load ( ' merlin', ' d: \  
winnt \msagent \chars \Merlin. acs' ); //加载动
```

画人物

```
Merlin := Agent1.Characters.Character ('Merlin');
Merlin.Show (1);
Merlin.MoveTo (500, 200, 1);
Merlin.Speak ('您可以建立一个现金帐户管理所有现金, 也可以分类或分成员建立帐户, 以便管理, 'd:\new2.wav'); //弹出文字提示框, 同时播放 new2 文件中存放的中文语音.
end;

当不需要帮助时, 可以单击“取消帮助”按钮, 隐藏并卸载动画人物。代码为:
Procedure TForm1.BitBtn4Click (Sender: TObject);
begin
    Merlin.Hide (0);
    Agent1.Characters.Unload ('merlin');
end;
```

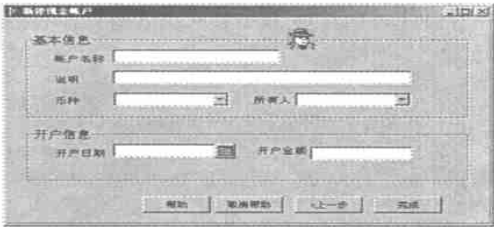


图 2 实例设计界面

Fig. 2 A design form of the sample

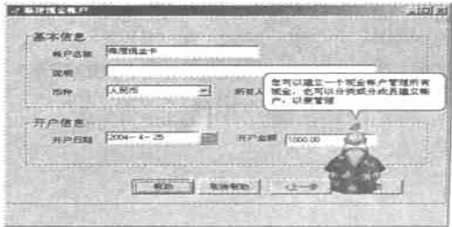


图 3 实例运行界面

Fig. 3 A running form of the sample

6 结束语

MS Agent 是一项较新的、也是很有前途的技术, 其内部机制相当复杂, 本文仅从 ActiveX 控件的角度出发, 介绍了 Delphi 中 MS Agent 的使用。但 ActiveX 控件隐藏了许多 OLE 技术的细节, 如果要对 MS Agent 更灵活的使用, 采用 OLE 自动化服务器编程方法, 直接对 COM 接口进行编程会更好。

参考文献:

[1] 何言祥, 陈莘萌. Agent 与多 Agent 系统的设计与应用 [M] . 武汉: 武汉大学出版社, 2001.
[2] Ted Pattison (美). COM+ 与 VB6.0 分布式应用程序设计 [M] . 王新昌, 陈越, 译. 北京: 机械工业出版社, 2001.
[3] 赵建华, 朱信忠. 基于 Agent 的 Visual C++ COM 编程实现技术 [J] . 微电子学与计算机, 2002, (5): 59—62

Study of ActiveX Programming Technology of Microsoft Agent

XUE Lei

(Department of Computer Science and Technology, Jiangsu Polytechnic University, Changzhou, 213016, China)

Abstract: The technology of Microsoft Agent influences strongly the way people communicate with their computers because of its powerful interactive capability, humanization, versatility, and ease of development. It is a set of software services that enable developers to incorporate interactive animated characters into their applications and Web pages and it can be implemented as an OLE Automation server or ActiveX control. Some relative objects and ways of MS agent control are discussed in this paper, and the way of Delphi ActiveX programming technology based on MS Agent is also presented in detail.

Key words: Microsoft Agent; technology of ActiveX; OLE Automation Server