

# Visual C++多媒体开发方法研究

张晓军 周家纪 孙淑霞

(成都理工学院计算机工程系 成都 610059)

**摘要:** 针对使用 Visual C++ 4.0 以上的开发平台设计多媒体应用程序这一课题, 本文试图在方法上进行一些研究探讨, 阐述了使用 OLE、OCX、DLL、多媒体类、MFC 类、Windows 类、API 等多种方法进行多媒体应用程序设计的思想。

**关键词:** Visual C++ 多媒体 Windows MCI 类

随着 Microsoft 公司 Visual C++ (以下简称 VC) 软件开发平台各种版本产品的不断推出, 其“可视化”用户界面驱动模型、面向对象编程技术、结构化事件驱动编程技术及其他技术的应用, 使得 VC 的功能变得越来越强大。本文全面地论述了用 VC 开发多媒体系统的各种方法。

## 1 Windows 的多媒体特性和 VC 中的有关内容

VC 下多媒体的编程是基于 Windows 的, Windows 的多媒体特性以及 VC 中相关的内容主要有以下几个方面:

(1) 图形设备接口(GDI) GDI 的一个主要目标是在输出设备上支持与设备无关的图形, 由 GDI. EXE 模块中的图形设备接口函数来处理。VC 在 MFC 中实现 GDI, 设备上下文提供了用户程序与显示或打印硬件的接口, 并由 CDC 类及其派生类来处理, 绘画、字体、显示和位图操作是以 CGdiObject 派生出的类来处理。

(2) 图形用户界面(GUI) GUI 采用了面向对象的设计概念, 把用户界面的诸元素定义为窗口、菜单、按钮、对话框等类, 一个程序的 GUI 部分实质上就是由这些类的对象构成的。VC 中“可视化”用户界面驱动模型为 GUI 提供了强大的支持, 在短时间内即可编制出复杂而美观的界面来。

(3) 媒体控制接口(MCI) MCI 是微软 Windows 定义的多媒体接口标准。MCI 包括了音频、视频、激光视盘机和录像机等多媒体的主要产品, 能满足大多数应用软件的需要。MCI 的最大优点是应用系统与设备无关, 这使得程序员在多媒体应用系统的开发中, 无需了解每种产品细节就能开发出通用的多媒体应用系统。

(4) 对象的链接和嵌入(OLE) OLE 技术发展到现在已不再是单纯的“对象的链接和嵌入”了, 其服务系统囊括软件部分的各个方面, 包括复合文档、OLE 控件(OCX)、交互应

本文于 1997-09-30 收到, 1998-01-16 收到修改稿。

用程序脚本、数据传送等等。VC 和 MFC 类库中的 OLE 类,封装了 API 中的 OLE 操作函数,派生出各种 OLE 类,对 OLE 提供了有力的支持。

(5) 动态连接库(DLL) DLL 是 Windows 环境中实现软件资源共享的主要方式,在多媒体系统开发中也具有不可估量的作用。VC 具有专门生成 DLL 的功能,且方便快捷。

## 2 VC 下多媒体开发方法

### 2.1 使用 OLE 技术

OLE 是一种动态信息交换协议,它通过一个由 OLE 包容器(Container)和 OLE 服务器(Server)组成的系统来运作。OLE 允许一个应用程序调用另一个独立的应用程序来提供服务。在多媒体开发中,一个多媒体应用程序要求的服务可能是播放音频,也可能是更复杂的音频、视频的同时播放;使用 OLE,就可以调用一个图形包来画或显示一幅图像,调用声音包来为此图像配音。实现 OLE 的方法,一种是直接地连接或嵌入;另一种是以图标的方式完成即使用包(Packages)。一个例子是复合对象中嵌入用传声器图标表示的声音片断,选定时,嵌入对象连同它相连的应用程序一起被激活,即激活 Windows 和 Sound Blaster 应用程序来播放 WAV 声音文件。

VC 下,在 App Wizard step 3 of 6 中设置可选项可以非常容易地创建 OLE 包容器和 OLE 服务器。MFC 中 OLE 类是对象链接和嵌入处理的基础。开发者可以把实现多媒体某一功能的模块如播放动画设计为一个最小服务器,然后作为对象嵌入到主应用程序(设计为包容器)中,也可以使用现成的各种封装了多媒体功能的服务器应用程序。

### 2.2 运用 OCX(OLE Control eXtensions)

由于 Visual Basic 和 VBX 扩充有移植限制,使得 OLE2 控件成为真正 32 位编程扩充的有力工具。OLE 控件由三个接口类型来定义:属性、事件和方法。属性(Property)是 OLE 控件内有名字的数值,如名字、颜色、宽度等;事件(Event)是 OLE 控件响应一些诸如鼠标单击或键盘事件的行动时激发的事件,事件可伴有参数,如鼠标坐标或键状态;方法(Method)是 OLE 控件内的函数,可由包容器程序在外部调用。

OCX 可分为两类,一类是 OLE 公用控件;另一类是 OLE 自定义控件。在 VC 4.0 以上版本中增加了用于提高代码重用的重要工具——组件平台(Component Gallery),平台中提供了大量的公用 OCX,其中便含有 MCI32.OCX,即多媒体控制(Microsoft Multimedia Control)。利用 App Wizard 创建项目,在 Step 3 of 6 中设置 OLE Controls 选项以支持 OCX;使用组件平台将多媒体控制插入到应用程序中,VC 自动生成相应的控制类和实现文件。

MFC4.0 提供类 COleDispatchDriver 用于实现 OLE 自动化的客户一方,OLE 调度界面提供对一个对象的方法和属性的访问,该类的成员函数 SetProperty、GetProperty 分别用于设置和返回一个对象的属性;同时 MFC4.0 增强的 CWnd 类提供成员函数 SetProperty 和 GetProperty 用于设置和返回控制的属性。自动生成的 CMciCtrl 类封装所有的多媒体控制属性的操作,如成员函数 GetCommand 和 SetCommand 可获得和设置 Command 属性,向控制发送各种命令;GetDeviceType 和 SetDeviceType 可捡取和设置 MCI 设备的类型。用 App Studio 新建对话框 IDD—MCIIOCXDLG,通过鼠标右键加入多媒体控制,设置多媒体控制的按钮,利用这些按钮即可播放 WAV|MIDI|AVI 文件。组件平台的 OLE Controls 中还有许多 OCX,如 Videoplay Control 非常出色地对视频功能实现了封装,用起来极为方便。

另外,可以利用 VC 下 New Project Workspace 中的 OLE Control Wizard 项生成新工程,设计自定义控件。设工程名为 MCI,则典型的工程中包含以下类:CMciApp、CMciCtrl、CMciPropage 等,然后就可以在这个工程中开始编程者创造性的工作了,通过增加新的属性、方法、事件来完成所需的多媒体功能,自定义 OCX 在编译注册后会被自动加到组件平台中去。

### 2.3 创建和使用 DLL

DLL 是一个包含了若干函数的可执行模块,可以实现应用程序共享代码和资源。使用 DLL 时还要用到一个输入库,库中包含运行时刻定位 DLL 中代码的信息,从而创建应用程序和 DLL 之间的“动态连接”,VC 下多媒体程序设计中可以考虑利用 DLL 来扩充多媒体系统的功能。例如,在 DLL 中定义实现诸如播放动画、音频文件的函数 PlayAnimate、PlayWave,然后在主应用程序中通过 LoadLibrary 函数装载 DLL,即可调用 DLL 中的函数实现所需功能。VC 下通过 New Project Workspace 中的 MFC App Wizard(dll)或 Dynamic Link Library 来生成 DLL(VC 4.0 下的入口点函数是 DllMain)。

另外,也可以开发 Windows 下“纯资源”DLL(最好使用较简单的 Dynamic Link Library 选项),在 DLL 中根据多媒体系统的要求可包含位图、图像、声音、视频等各种资源,利用 DLL 作为载体,需要时才把资源调入系统。其优点是资源实现共享,节约系统内存空间,减小 EXE 文件,提高性能等。

近几年,第三方开发商开发出了具有多媒体特性的市场化 DLL,如 AAPLAY.DLL 可将 3DS 动画文件(.FLIC 格式)无缝地加到应用程序中,由程序控制动画的播放、状态并实现与其他事件的同步,可参阅文献 [3]。

### 2.4 使用多媒体类产品

VC 下,可以使用某些公司开发的多媒体类,如 CTegMM 类提供了对多媒体设备的控制,可以播放 WAV 文件、CD 盘、影视文件、MIDI 文件等。编程时,利用 VC 的 App Wizard 产生一程序框架,把 CTegMM 类包含进去,定义 CTegMM 的对象,在应用程序源文件中即可使用它来实现特定的多媒体功能。

### 2.5 编制 MFC 类

开发者可以从 MFC 的基本窗口类 CWnd 出发派生出多媒体基类 CMMWnd,此类抽取出来要处理的多媒体的共同特征。从 CMMWnd 类可派生出各种媒体类:如 CVoice 类管理声音文件的播放停止和编辑;CVideo 类管理对视频文件的操作;CPicture 类管理图像文件的显示等。最后,再对这些类对象进行必要的组织、集成或链接,即可开发出多媒体系统来。

在 VC4.0 版本中完全实现的公共控件类(在 COMCTL32.DLL 中)中有一个动画控件类 CAnimateCtrl,利用它也可播放一个 AVI 文件的视频部分,但 AVI 文件的音频部分未被支持。

### 2.6 Windows 类 MCIWnd 的应用

MCIWnd(Media Control Interface Windows)是一个预包装的 MCI 播放器的 Windows 类(而不是一个 VC 类),在 Video for Windows(VFW)中实现,开发者几乎不用编程就可以使用它提供的各种播放控制如暂停、速度、大小、音量等。VC 下使用 MCIWnd,须包括文件 VFW.H,连接设置中指出 VFW32.LIB,然后可用 MCIWndCreat 函数来建立一个 MCIWnd。MCIWnd API 包含许多控制 MCIWnd 操作的宏(类似于函数),使用这些宏,开发者除了直

接控制 MCIWnd 外,也可以从代码级来更灵活地控制。

## 2.7 使用 Windows 多媒体 API 函数

Windows 多媒体系统函数在 DLL 中,Windows 提供了低级和高级两组多媒体 API 函数,它们提供了 WAVE 音频播放和录音、合成音频和 MIDI、动画播放、视频播放等多媒体接口。为了加入 MCI 接口,要在 VC 相应的源文件中包含“mmsystem.h”文件,并在连接设置中指出 WINMM.LIB。表 1 给出主要的多媒体 API 函数接口及有关内容。开发者可以灵活地选用高级或低级函数进行多媒体编程。VC 联机手册中有许多封装了这些函数的例子,如 playWAVEFile、playMIDIFile、playCDTrack 等,供开发者参考使用。

表 1 主要的多媒体 API 函数接口

MCI 接口	解 释	级 别
PlaySound SndPlaySound	是两个发声函数,只执行单一的播放 WAVE 文件功能的高级函数	处于多媒体接口的顶点
(命令-消息接口) micSendCommand micGetDeviceID	两个高级命令接口——命令-消息和命令-字符串执行同样的功能,区别是单词和数字的区别——最后的结果是一样的	高级 MCI 接口
(命令-字符串接口) micSendstring		
(命令-字符串和命令-消息接口) micGetErrorString micSetYieldProc		
(常用的多媒体文件 I/O 函数) mmioOpen mmioClose mmioRead mmioWrite mmioSendMessage mmioDescend mmioAscend	高级 MCI 函数即封装调用了这些 I/O 函数,用于处理 RIFF 文件,可执行各种复杂的操作	低级多媒体函数
timeGetDevCaps timeBeginPeriod timeSetEvent timeKillEvent timeEndPeriod	实现媒体播放等功能的高精度计时	更低层的计时器函数

上述方法彼此之间并非相互独立,例如 OCX 本身就以 DLL 形式存在。所有的方法归根结底是使用低层的 Windows 多媒体 API 函数来实现,只是高层的实现方法封装的程度较高,编程较简单,不过灵活性也随之降低。开发者可以根据给定的多媒体项目找到实现各项功能的最佳结合方法。笔者相信,以 VC 作为开发平台,开发者定能开发出优秀的多媒体软件来。

## 参 考 文 献

- 1 Peter Aitken 等著,李鹤文,张文新译. Visual C++ 多媒体开发指南. 北京: 科学出版社, 1996
- 2 Ben Ezzell 著,廖俊,段爱民译. Windows 32 位编程指南. 北京: 清华大学出版社, 1996, 11
- 3 杨俊明. 在 Windows 环境下使用 3DS 的 FLIC 动画. 电子与电脑, 1995, 12
- 4 赵忠伟. Windows 下动态连接库的使用. 中国计算机用户, 1995, 12
- 5 陈坚. Visual C++ 4.0 组件平台与 OLE 控制编程技术. 中国计算机用户, 1997, 3