

为 Windows 的时钟应用程序增加定时和报时功能

张俊锋 (郑州中船总七一三研究所 450052)

摘要:本文采用 Windows 的窗口子类化方法,通过在时钟应用程序的系统菜单上挂接新的菜单项,为其增加了“整点报时”和“定时设置”两项新功能,具有相当的实用性。本文所介绍的窗口子类化方法和编程思想对解决其他问题也有一定的参考价值。

关键词:窗口子类化 消息 Windows 应用程序

一、问题的提出

熟悉 Windows 的计算机用户对随系统提供的时钟(CLOCK)应用程序也一定不会陌生。它能够以窗口或图标(其实是一个极小化的窗口)两种形态运行,用指针或数字形式随时向用户显示系统时间。

然而,时钟应用程序恰恰缺少我们日常生活中使用的时钟上最普遍最实用的“整点报时”和“定时设置”这两项基本功能。如果能够为时钟应用程序增加上这两项功能,不仅可以保证工作按部就班有条不紊地进行,还可以用来提醒用户每隔一段时间暂停工作休息一下而不致影响身体健康。

二、窗口子类化方法

应用程序为了登记一个窗口类,首先要填写好一个 WNDCLASS 结构,其中的结构元素 lpfnWndProc 就是该类窗口函数的地址。然后调用 RegisterClass() 函数向 Windows 系统申请登记这个窗口类。Windows 会为其分

配一块内存来存放该类的全部信息,这个内存块称为窗口类内存块。

当应用程序要创建一个属于某一已注册窗口类的窗口时,Windows 便为这个窗口分配一块内存即窗口内存块,用来存放与该窗口有关的专用信息。这些信息一部分来自传递给窗口创建函数如 CreateWindow() 或 CreateWindowEX() 等的参数信息,另一部分则来自所属的窗口类内存块。其中参数 lpfnWndProc 便被 Windows 从窗口类内存块拷贝到为新创建窗口分配的窗口内存块中。当有消息被发送到这个窗口实例时,Windows 检查该窗口内存块中的窗口函数地址(lpfnWndProc)值,并调用该地址上的函数来处理这些发送来的消息。

所谓窗口子类化,实际上就是改变窗口内存块中的窗口函数地址,使其指向一个新的窗口函数地址,从而改变原窗口函数的处理方法或增加其功能。由于这些修改只涉及到一个窗口内存块,所以它不会影响到属于同一窗口类的其他 © 中国科学院软件研究所 <http://www.c-s-a.org.cn>

对一个窗口进行子类化的基本步骤如下：

1. 编制子类化的窗口函数。该函数必须为标准的窗口函数格式即：

LONG FAR PASCAL -export WndSubClassProc (HWND, UINT, WPARAM, LPARAM);

在这个函数里对我们感兴趣的消息进行处理，而未处理或需要原窗口函数进一步处理的其他消息传送给原窗口函数。

2. 利用待子类化窗口的句柄，调用 GetWindowLong (hWnd, GWL-WNDPROC) 函数获得其原窗口函数的地址并保存起来。

3. 调用 SetWindowLong (hWnd, GWL-WNDPROC, fpWndSubClassProc) 把窗口函数设置成子类化窗口函数，完成窗口子类化。

经过上述子类化后的窗口实例的消息处理过程如图 1 所示，其中的虚线表示子类化之前的消息派送情形。

名“CLOCK”按如下调用形式：

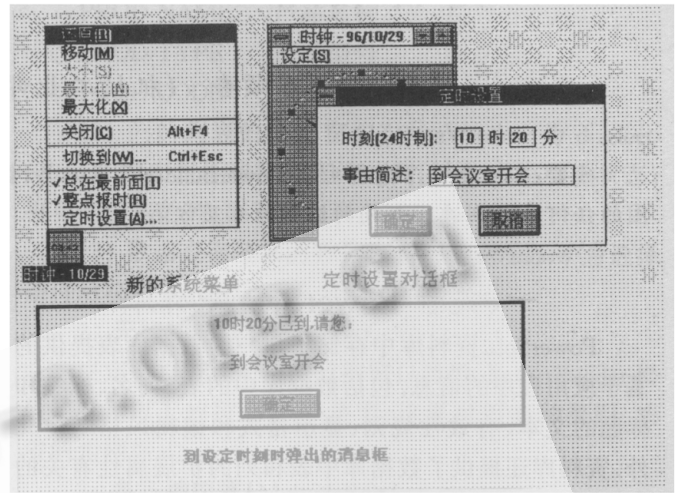


图 2 源程序执行效果图

hWndClock = FindWindow("CLOCK", NULL);
来获取时钟应用程序的窗口句柄。

有了时钟窗口句柄，调用 GetSystemMenu() 函数可得到其系统菜单的句柄。并在上面挂接“整点报时”和“定时设置”这两个菜单项。然后置换窗口函数，实现窗口子类化。

但并不能就此结束本应用程序返回 Windows 系统，因为时钟应用程序的子类化窗口函数是本应用程序创建的，结束本应用程序系统就会收回其申请的所有内存资源，其中也包括时钟程序此时正在使用的子类化窗口函数。所以接下来要进入消息循环，直到用户关闭时钟应用程序。这时时钟子类化窗口函数会向本应用程序发送 WM-QUIT 消息而退出消息循环，结束本应用程序。

在时钟子类化窗口函数 ClockSubClass() 中，不仅要对新添加的菜单项产生的消息 IDM-REPORT 和 IDM-ALART 进行处理，对时钟消息 WM-TIMER 在需要时(即选中了“整点报时”或“定时设置”选项)也要判断是否到了整点或所设定的时刻以进行相应的处理，然后传送给时钟程序原来的窗口函数来更新显示信息。另外对于消息 WM-DESTROY，则首先向本应用程序发送 WM-QUIT 消息来终止其消息循环，再传送给时钟程序原窗口函数处理。而对于其他消息则直接传送给时钟原窗口函数去处理。

根据上述编程思想用 Borland C++ 3.1 编制的源程序清单略，有兴趣的读者可与作者联系。

(来稿时间:1997年3月)

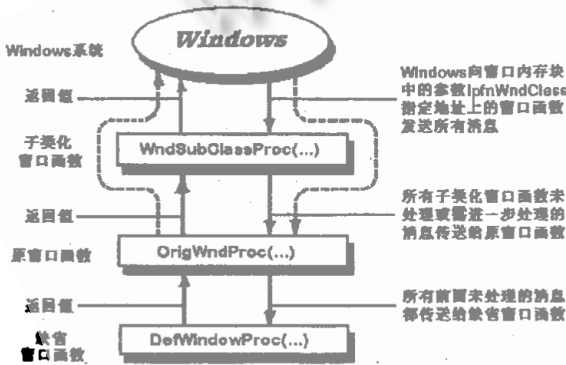


图 1 窗口子类化

三、增加时钟功能的编程思想

为了给时钟应用程序增加新的功能，首先要调用 FindWindow (lpzClassName, lpzWindow) 函数来获取时钟窗口的句柄。该函数需要两个参数：lpzClassName 是要寻找的窗口所属的窗口类名(如果为空则匹配所有的窗口类)，lpzWindow 是要获取的窗口的标题(如果为空则匹配所有的标题)。注意这里不能用 FindWindow (NULL, “时钟”)这样的调用形式来寻找时钟窗口句柄，因为它的标题并不总是“时钟”。如果选中了“指针式”和“日期”这两个菜单项，窗口标题便成“时钟 - 96/10/29”形式了(如图 2)，导致寻找失败。所以应该用其窗口类