

为 Windows 程序配置图标

赵忠伟 (山东雅泰电子公司)

Windows 是一个图形用户界面(GUI),各种操作对象都是以图形形式显示在屏幕上,通过键盘和鼠标器,用户可以在屏幕上直接操纵这些对象。这样,用户与程序之间的交互变的更加紧密了,不再是单一的用键盘输入信息到屏幕去显示,而是能直接与屏幕上的对象进行交互操作。

DOS 下程序是以文件名的形式呈现给用户,而在 Windows 下则是以图标的形式出现在屏幕上,即每个应用程序用一个小图案代表。用户只需用鼠标在其上双击即可调用该程序,使其运行。

所谓图标是一个小的图象文件,其扩展名是.ICO。程序人员可以用下面三种方法为 Windows 程序配置图标。

1.使用 Windows 内部提供的五种预定义图标。

在注册应用程序主窗口类时,利用第六个参数取得要用图标的句柄。

```
hIcon = LoadIcon(NULL,flag);
```

其中第一个参数 NULL,表示要取得 Windows 预定义的图标,第二个参数 flag,可以是下述五种标识之一。IDLAPLICATION,IDLASTERISK,IDLEXCLAMATION,IDLQUESTION,IDLHAND。这五种图标已由 Windows 内部作了定义,可以直接使用。

LoadIcon()函数返回要使用的图标句柄,存入 hIcon 变量中,类型为 hIcon,利用此返回的句柄即可取得该图标的使用权。

例如,要使用预定义的手形图标,则注册窗口类时第六个参数为:

```
hIcon = LoadIcon(NULL,IDLHAND);
```

2.利用 Windows 的程序管理器,将 Windows 系统已经装配的图标文件(.ICO),指定给某个应用程序,操作如下:

- (1)进入 Windows 程序管理器中
- (2)打开指定的应用程序工作组
- (3)选择要设置图标的应用程序
- (4)打开菜单中的文件选项
- (5)选择特性一栏
- (6)在出现的对话框中选择更改图标按钮
- (7)用鼠标拖动,选择一个满意的图标
- (8)按确定按钮,完成选择

完成上述操作后,工作组中该程序的图标就以指定的图标代替,以后当这个应用程序的主窗口缩到最小时,将由指定的图标来代表自己。

3.在应用程序中指定要使用的图标

(1)可以用 Windows 的 SDK 或 Borland C++3.1 的 Resource Workshop 来设计和制作.ICO 文件,假设已制作好的图形文件是 example.ico。

(2)资源文件(.RC)描述了应用程序要使用的各种资源。例如,光标资源、位图资源、字符串资源等。因图标也是一种资源,所以在资源文件中加入 ICON 语句,将图标文件标识出。

```
Example ICON example.ico
```

(3)在 C 语言源程序中,注册窗口类时,用下列语句获得资源文件中描述的图标文件的句柄。

```
hIcon = LoadIcon(hInstance,"Example");
```

其中 hInstance 是本实例的句柄。"Example"要与资源文件中 ICON 语句的标识一致。LoadIcon()函数将该图标的句柄返回,并存入 hIcon 变量中。通过以上三步即可为应用程序指定某一个图标文件。

如果想在程序运行过程中更换图标的图案。在窗口类注册完毕之后,则可采用改变窗口类字段的方法,具体通过使用 SetClassWord()函数来实现。假设资源描述文件中还有另外一个图标资源:

```
AnotherExam ICON another.ico
```

通过如下调用,就可以改变原来窗口类中所设置的图标:

```
hIcon = LoadIcon(hInstance,"AnotherExam");
```

```
SetClassWord(hWnd,GCW-HICON,hIcon);
```

这时如果再将窗口缩小成图标,则图标的图案便会是 "AnotherExam"所指定的图标图案。其中 hWnd 是所在窗口的句柄,hIcon 是指定图标的句柄。

通常图标是在窗口被缩小时,才显示于屏幕下方。若用户想在客户区显示图标,则可利用 DrawIcon()函数实现,调用如下:

```
hDC = GetDC(hWnd);
```

```
DrawIcon(hDC,x,y,hIcon);
```

GetDC()函数获得设备环境句柄,赋给变量 hDC。(x,y)是客户区坐标,hIcon 是指定图标的句柄。

这时,在客户区(x,y)坐标位置上,便会显示出一个以 hIcon 为句柄的图标图案。事实上,在 Windows 系统内部也是调用 DrawIcon()函数将程序的图标显示在屏幕下方。若在窗口右击鼠标后(RButtonDown),屏幕上图标图案(example.ico)旁边就会出现另一个图标图案(another.ico)。主窗口再缩小时,屏幕下方出现的图标图案也会发生变化。

下面所附程序在 Borland C++3.1 下编译通过,可在中文 Windows 3.1 下运行。将应用程序加入工作组后,自动以指定图标显示该程序。主窗口被缩小时,也以指定图标显示于屏幕下方。

读者可直接将提供的源程序利用 Borland C++3.1 的集成环境(DOS 或 Windows 下都可)编译成 Windows 可执行文件。二个图标文件可利用工具制作,也可利用现有的图标文件,将其换名为 example.ico、another.ico。

附:C 语言源程序 drawicon.c

```

/* -----DRAWICON.C----- */
#include < Windows.h >
int PASCAL WinMain (HANDLE,HANDLE,LPSTR,int);
long FAR PASCAL MainWndProc(HWND,unsigned, unsigned,
LONG);
/* -----WinMain()----- */
HANDLE hGlobalInstance
int PASCAL WinMain(HANDLE hInstance,HANDLE hPrevInsta
nce,LPSTR lpszCmdLine,int nCmdShow)
{
    MSG        msg;
    HWND        hwnd;
    WNDCLASS    wndclass;
    hGlobalInstance = hInstance;
    if (!hPrevInstance)
    {
        /* set parameters of the window class
        used by application window */
        wndclass.style = CS_HREDRAW|CS_VREDRAW;
        wndclass.lpfnWndProc = MainWndProc;
        wndclass.cbClsExtra = 0;
        wndclass.cbWndExtra = 0;
        wndclass.hInstance = hInstance;
        wndclass.hIcon = LoadIcon(hInstance,"Example");
        wndclass.hCursor = LoadCursor(NULL,IDC_ARROW);
        wndclass.hbrBackground = GetStockObject(White_Brush);
        wndclass.lpszMenuName = NULL;
        wndclass.lpszClassName = "WindowClass";
        /* register this window class to Windows */
        if (!RegisterClass (wndclass))
            return FALSE;
    }

    /* create a window instance,send WM_CREATE message */
    hwnd = CreateWindow(
        "WindowClass", /* window class name */
        "Draw a Icon Program", /* window caption */
        WS_OVERLAPPEDWINDOW, /* window style */
        CW_USEDEFAULT, /* initial x position */
        CW_USEDEFAULT, /* initial y position */
        CW_USEDEFAULT, /* initial x size */
        CW_USEDEFAULT, /* initial y size */
        NULL, /* parent window handle */
        NULL, /* window menu handle */
        hInstance, /* program instance handle */
        NULL /* creation parameters */
    );
    /* return if creation failure */
    if (!hwnd)
        return FALSE;
    /* display this created window,send WM_PAINT message */

```

```

ShowWindow (hwnd,nCmdShow);
UpdateWindow(hwnd);
/* message loop,until WM_QUIT message */
while (GetMessage(&msg,NULL,0,0))
{
    TranslateMessage(&msg); /* translate virtual key code */
    DispatchMessage(&msg); /* dispatch to window procedure */
}
/* return the value returned from PostQuitMessage() */
return msg.wParam;
}
/* -----MainWndProc()----- */
long FAR PASCAL MainWndProc (HWND hwnd, unsigned
int message,unsignedint wParam,LONG lParam)
{
    HDC hDC;
    HICON hIcon;
    switch (message)
    {
        case WM_PAINT;
            hIcon = LoadIcon(hGlobal Instance,"Example");
            hDC = GetDC(hwnd);
            DrawIcon(hDC,10,10,hIcon);
            break;
        case WM_RBUTTONDOWN:
            hIcon = LoadIcon(hGlobal Instance,"AnotherExam");
            SetClassWord(hwnd,GCW_HICON,hIcon);
            hDC = GetDC(hwnd);
            DrawIcon(hDC,60,10,hIcon);
            break;
        case WM_DESTROY:
            PostQuitMessage(0);
            return 0;
        /* ignore all other messages */
        default:
            break;
    }
    return DefWindowProc (hwnd,message,wParam,lParam);
}
/* -----End of Windows Programming----- */
资源描述文件 drawicon.rc
#include < windows.h >
Example ICON example.ico
AnotherExam ICON another.ico
模块定义文件 drawicon.def
;DRAWICON.DEF module definition file
NAME DRAWICON
DESCRIPTION 'Drawicon a Icon Windows Program'
EXETYPE WINDOWS
STUB 'WINSTUB.EXE'
CODE PRELOAD MOVEABLE DISCARDABLE
DATA PRELOAD MOVEABLE MULTIPLE
HEAPSIZE 1024
STACKSIZE 8192
EXPORTS MainWndProc

```