# 载入资源创建DDB并显示的方法

附件中的代码封装了一个CKDDB类，可以完成从资源中载入位置，并显示位图，同时给出了示例程序。

## 一、程序要点

总的说来，有几下几步：载入位图，创建与屏幕兼容DC，选入位图，获取位图尺寸，将位图从兼容DC中拷贝到屏幕DC上。

1.1、载入位图，用到LoadBitmap函数，该函数从指定的ID资源中载入位图，返回Bitmap的句柄HBITMAP。

1.2、将位图选入DC，用到函数SelectObject，对不再需要的HBITMAP变量应销毁，用到函数DeleteObject函数。对于选入DC的HBITMAP，如果不想继续用变量保存位图句柄，可使用以下语句：

SelectObject(m\_hMemDC,m\_hOldBmp);

m\_hOldBmp=NULL;

1.3、获取位图的尺寸信息，用到函数GetObject，例如以下代码：

BITMAP bmp;

if (!GetObject(m\_hBitmap,sizeof(bmp),&bmp))

{

return false;

}

width=bmp.bmWidth;

height=bmp.bmHeight;

1.4、在绘制之前，检查是否已创建兼容DC，如果没有，则创建与屏幕DC兼容的兼容DC，并选入位图资源：

if (m\_hMemDC==NULL)

{

HDC hDC=GetDC(NULL);

m\_hMemDC=CreateCompatibleDC(hDC);

ReleaseDC(NULL,hDC);

m\_hOldBmp=(HBITMAP)SelectObject(m\_hMemDC,m\_hBitmap);

}

1.5、调用BitBlt函数将兼容DC上的位图绘制到屏幕上。

BitBlt(hDC,x0,y0,bmpwidth,bmpheight,m\_hMemDC,0,0,rop)

## 二、未封闭的例程

## 三、附录

### KDDB.h文件：

#pragma once

class CKDDB

{

protected:

HBITMAP m\_hBitmap;

HBITMAP m\_hOldBmp;

void ReleaseDDB(void);

public:

HDC m\_hMemDC;

bool Prepare(int & width, int & height);

typedef enum{draw\_normal,draw\_center,draw\_tile,draw\_stretch,draw\_stretchchprop};

HBITMAP GetBitmap(void)const

{

return m\_hBitmap;

}

CKDDB()

{

m\_hBitmap=NULL;

m\_hMemDC=NULL;

m\_hOldBmp=NULL;

}

virtual ~CKDDB()

{

ReleaseDDB();

}

BOOL Attach(HBITMAP hBmp);

BOOL LoadBitmap(HINSTANCE hInst,int id)

{

return Attach(::LoadBitmap(hInst,MAKEINTRESOURCE(id)));

}

BOOL Draw(HDC hDC,int x,int y,int w,int h, DWORD rop,int opt=draw\_normal);

};

### KDDB.CPP文件：

#include "StdAfx.h"

#include "KDDB.h"

bool CKDDB::Prepare(int & width, int & height)

{

BITMAP bmp;

if (!GetObject(m\_hBitmap,sizeof(bmp),&bmp))

{

return false;

}

width=bmp.bmWidth;

height=bmp.bmHeight;

if (m\_hMemDC==NULL)

{

HDC hDC=GetDC(NULL);

m\_hMemDC=CreateCompatibleDC(hDC);

ReleaseDC(NULL,hDC);

m\_hOldBmp=(HBITMAP)SelectObject(m\_hMemDC,m\_hBitmap);

}

return true;

}

void CKDDB::ReleaseDDB(void)

{

if (m\_hMemDC)

{

SelectObject(m\_hMemDC,m\_hOldBmp);

DeleteObject(m\_hMemDC);

m\_hMemDC=NULL;

}

if (m\_hBitmap)

{

DeleteObject(m\_hBitmap);

m\_hBitmap=NULL;

}

m\_hOldBmp=NULL;

}

BOOL CKDDB::Attach(HBITMAP hBmp)

{

if (hBmp==NULL)

{

return FALSE;

}

if (m\_hOldBmp) //deselect m\_hBitmap

{

SelectObject(m\_hMemDC,m\_hOldBmp);

m\_hOldBmp=NULL;

}

if (m\_hBitmap)

{

DeleteObject(m\_hBitmap);

}

m\_hBitmap=hBmp; //replace with new one

if (m\_hMemDC)

{

m\_hOldBmp=(HBITMAP)SelectObject(m\_hMemDC,m\_hBitmap);

return m\_hOldBmp!=NULL;

}

else

return TRUE;

}

BOOL CKDDB::Draw(HDC hDC,int x0,int y0,int w,int h, DWORD rop,int opt/\* =draw\_normal \*/)

{

int bmpwidth,bmpheight;

if (!Prepare(bmpwidth,bmpheight))

{

return FALSE;

}

switch (opt)

{

case draw\_normal:

return BitBlt(hDC,x0,y0,bmpwidth,bmpheight,m\_hMemDC,0,0,rop);

break;

case draw\_center:

return BitBlt(hDC,x0+(w-bmpwidth)/2,y0+(h-bmpheight)/2,

bmpwidth,bmpheight,m\_hMemDC,0,0,rop);

break;

case draw\_tile:

for (int j=0;j<h;j+=bmpheight)

{

for (int i=0;i<w;i+=bmpwidth)

{

if (!BitBlt(hDC,x0+i,y0+j,bmpwidth,bmpheight,

m\_hMemDC,0,0,rop))

{

return FALSE;

}

return TRUE;

}

}

return FALSE;

break;

case draw\_stretchchprop:

{

int ww=w;

int hh=h;

if (w\*bmpheight<h\*bmpwidth)

{

hh=bmpheight\*w/bmpwidth;

}

else

ww=bmpwidth\*h/bmpheight;

return StretchBlt(hDC,x0+(w-ww)/2,y0+(h-hh)/2,ww,hh,

m\_hMemDC,0,0,bmpwidth,bmpheight,rop);

}

break;

default:

return FALSE;

}

}

### 示例程序：

1、添加位图资源，此处ID为IDB\_BALLOONS。

2、在视图类中添加变量：CKDDB m\_DDB;

3、在OnInitialUpdate()中加入以下代码：

m\_DDB.LoadBitmap(AfxGetInstanceHandle(),IDB\_BALLOONS);

4、在OnDraw函数中：

int w,h;

m\_DDB.Prepare(w,h);

m\_DDB.Draw(pDC->GetSafeHdc(),0,0,w,h,SRCCOPY);