**VC++自绘窗体界面[转]**

2010-11-12 12:47:26|  分类： [默认分类](http://tosoo.blog.163.com/blog/#m=0&t=1&c=fks_087071084080084068087084087095085087083067084081086066092)|字号 订阅

一. 新建一个基于对话框的应用程序.  
二. 在对话框中添加Eeit ,StaticText ,Picture, Button 控件.  
三. 在对话框中添加如下成员变量:  
//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
BOOL m\_IsMax;         //是否处于最大化状态  
 int  m\_BorderWidth;   //边框宽度  
 int  m\_BorderHeight;  //边框高度  
 int  m\_CaptionHeight; //标题栏的高度  
 CString  m\_Caption;      //窗口标题  
 COLORREF m\_CapitonColor; //标题字体颜色  
 CFont    m\_CaptionFont;  //标题字体

 int m\_ButtonWidth;  //按钮位图宽度  
 int m\_ButtonHeight; //按钮位图高度  
   
 BOOL  m\_FirstShow; //窗口首次被显示

 CRect m\_OrigonRect;//原始窗口区域

 CRect m\_IniRect,m\_MinRect,m\_MaxRect,m\_CloseRect;//标题栏按钮的显示区域  
   
 CButtonState m\_ButtonState; //按钮状态

 BOOL m\_IsDrawForm;//是否需要绘制窗体

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
四.在对话框中添加DrawFrom 方法绘制窗体，代码如下:

//获取窗口设备上下文  
 CDC\* pWindowDC = GetWindowDC();  
 CBitmap LeftLine;  
 BITMAPINFO bitinfo;  
   
 CDC memDC;  
 memDC.CreateCompatibleDC(pWindowDC);

 CRect Clientrect;  
 GetClientRect(Clientrect);

 int leftwidth=0; //左标题的宽度  
 int rightwidth = 0; //右标题的宽度  
 int leftlinewidth = 0; //左边线宽度  
 LeftLine.LoadBitmap(IDB\_BITMAP3); //加载右标题  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 rightwidth = bitinfo.bmiHeader.biWidth;  
 LeftLine.DeleteObject();

 int x,y;  
 //绘制左边线

 //获取位图大小  
 LeftLine.LoadBitmap(IDB\_BITMAP4);

 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 leftlinewidth = x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;

 memDC.SelectObject(&LeftLine);  
 pWindowDC->StretchBlt(1-m\_BorderWidth,m\_CaptionHeight+1,x+1,Clientrect.Height()+2\*m\_BorderHeight+5,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
   
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制左标题\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP2);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 leftwidth = x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(-m\_BorderWidth,0,x,m\_CaptionHeight+4,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制左标题\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制中间标题\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP1);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(leftwidth-1,0,Clientrect.Width()-leftwidth-rightwidth ,m\_CaptionHeight+4,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制中间标题\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制右标题\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP3);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;

 pWindowDC->StretchBlt(Clientrect.Width()-x-1,0,x+m\_BorderWidth+9,m\_CaptionHeight+4,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制右标题\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制右边框\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP4);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(Clientrect.Width()+m\_BorderWidth+2,m\_CaptionHeight+1,x+m\_BorderWidth,Clientrect.Height()+2\*m\_BorderHeight+5,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制右边框\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制底边框\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP5);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(leftlinewidth-m\_BorderWidth,Clientrect.Height()+m\_CaptionHeight+2,Clientrect.Width()+m\_BorderWidth,y+2,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制底边框\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制初始化按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP6);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(m\_IniRect.left,m\_IniRect.top,m\_IniRect.right,m\_IniRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制初始化按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制最小化按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP6);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(m\_MinRect.left,m\_MinRect.top,m\_MinRect.right,m\_MinRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制最小化按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制最大化按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP6);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(m\_MaxRect.left,m\_MaxRect.top,m\_MaxRect.right,m\_MaxRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制最大化按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制关闭按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 LeftLine.LoadBitmap(IDB\_BITMAP6);  
 //获取位图大小  
 LeftLine.GetObject(sizeof(bitinfo),&bitinfo);  
 memDC.SelectObject(&LeftLine);  
 x = bitinfo.bmiHeader.biWidth;  
 y = bitinfo.bmiHeader.biHeight;  
 pWindowDC->StretchBlt(m\_CloseRect.left,m\_CloseRect.top,m\_CloseRect.right,m\_CloseRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
 LeftLine.DeleteObject();  
 m\_IsDrawForm = TRUE;  
 /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*绘制关闭按钮\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
 ReleaseDC(&memDC);  
 DrawFormCaption();  
//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

五. 处理窗体的WM\_NCMOUSEMOVE消息，确定标题栏按钮的状态，代码如下：

CDialog::OnNcMouseMove(nHitTest, point);  
 CRect tempIni,tempMin,tempMax,tempClose,ClientRect;  
   
 CDC\* pWindowDC = GetWindowDC();  
 CDC memDC;  
 memDC.CreateCompatibleDC(pWindowDC);

 BITMAPINFO bInfo;  
 CBitmap LeftLine;  
 int x,y;

 GetWindowRect(ClientRect);  
 //GetClientRect();  
 tempIni.CopyRect(CRect(m\_IniRect.left+ ClientRect.left,ClientRect.top+m\_IniRect.top,m\_IniRect.right+m\_IniRect.left+ ClientRect.left,m\_IniRect.bottom+m\_IniRect.top+ClientRect.top));  
 tempMin.CopyRect(CRect(m\_MinRect.left+ ClientRect.left,ClientRect.top+m\_MinRect.top,m\_MinRect.right+m\_MinRect.left+ ClientRect.left,m\_MinRect.bottom+m\_MinRect.top+ClientRect.top));  
 tempMax.CopyRect(CRect(m\_MaxRect.left+ ClientRect.left,ClientRect.top+m\_MaxRect.top,m\_MaxRect.right+m\_MaxRect.left+ ClientRect.left,m\_MaxRect.bottom+m\_MaxRect.top+ClientRect.top));  
 tempClose.CopyRect(CRect(m\_CloseRect.left+ ClientRect.left,ClientRect.top+m\_CloseRect.top,m\_CloseRect.right+m\_CloseRect.left+ ClientRect.left,m\_CloseRect.bottom+m\_CloseRect.top+ClientRect.top));  
   
 if (tempIni.PtInRect(point)) //鼠标在初始化按钮上移动时,更改按钮显示的位图  
 {

  LeftLine.LoadBitmap(IDB\_BITMAP7);  
  LeftLine.GetObject(sizeof(bInfo),&bInfo);  
    
  x = bInfo.bmiHeader.biWidth;  
  y = bInfo.bmiHeader.biHeight;  
  memDC.SelectObject(&LeftLine);  
  pWindowDC->StretchBlt(m\_IniRect.left,m\_IniRect.top,m\_IniRect.right,m\_IniRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
  m\_IsDrawForm = FALSE;  
  m\_ButtonState = bsIni;  
  LeftLine.DeleteObject();  
 }  
 else if(tempMin.PtInRect(point))//鼠标在最小化按钮上移动时,更改按钮显示的位图  
 {  
  LeftLine.LoadBitmap(IDB\_BITMAP7);  
  LeftLine.GetObject(sizeof(bInfo),&bInfo);  
    
  x = bInfo.bmiHeader.biWidth;  
  y = bInfo.bmiHeader.biHeight;  
  memDC.SelectObject(&LeftLine);  
  pWindowDC->StretchBlt(m\_MinRect.left,m\_MinRect.top,m\_MinRect.right,m\_MinRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
  m\_IsDrawForm = FALSE;  
  m\_ButtonState = bsMin;  
  LeftLine.DeleteObject();    
 }  
 else if (tempMax.PtInRect(point))  
 {  
  LeftLine.LoadBitmap(IDB\_BITMAP7);  
  LeftLine.GetObject(sizeof(bInfo),&bInfo);  
    
  x = bInfo.bmiHeader.biWidth;  
  y = bInfo.bmiHeader.biHeight;  
  memDC.SelectObject(&LeftLine);  
  pWindowDC->StretchBlt(m\_MaxRect.left,m\_MaxRect.top,m\_MaxRect.right,m\_MaxRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
  m\_IsDrawForm = FALSE;  
  if (m\_IsMax)  
  {  
   m\_ButtonState = bsMax;  
  }  
  else  
  {  
   m\_ButtonState = bsRes;  
  }  
    
  LeftLine.DeleteObject();    
 }  
 else if (tempClose.PtInRect(point))  
 {  
  LeftLine.LoadBitmap(IDB\_BITMAP7);  
  LeftLine.GetObject(sizeof(bInfo),&bInfo);  
    
  x = bInfo.bmiHeader.biWidth;  
  y = bInfo.bmiHeader.biHeight;  
  memDC.SelectObject(&LeftLine);  
  pWindowDC->StretchBlt(m\_CloseRect.left,m\_CloseRect.top,m\_CloseRect.right,m\_CloseRect.bottom,&memDC,0,0,x,y,SRCCOPY);  
  m\_IsDrawForm = FALSE;  
  m\_ButtonState = bsClose;  
  LeftLine.DeleteObject();     
 }  
 else  
 {  
  m\_ButtonState = bsNone;  
  if (m\_IsDrawForm==FALSE)  
   DrawForm();  
 // m\_IsDrawForm = TRUE;  
 }  
 ReleaseDC(&memDC);

//---------------------------------------------------------------------------------

六.处理窗体的WM\_NCLBUTTONDOWN消息，当用户在非客户区域单击时，根据按钮的状态执行相应的操作，代码如下：

 CDialog::OnNcLButtonDown(nHitTest, point);

 switch (m\_ButtonState)  
 {  
 case bsClose: //关闭窗口  
  {  
   DestroyWindow();  
  }  
  break;  
 case bsIni: //还原窗口到初始大小和位置  
  {  
   m\_IsMax = TRUE;  
   MoveWindow(m\_OrigonRect.left,m\_OrigonRect.top,m\_OrigonRect.Width(),m\_OrigonRect.Height());  
  }  
  break;  
 case bsMin: //  
  {  
   CWnd\* pDesk = GetDesktopWindow();  
   CRect rect;  
   pDesk->GetClientRect(rect);  
   SetWindowPos(0 ,(rect.Width()-m\_OrigonRect.Width())/2,2,m\_OrigonRect.Width(),0,SWP\_SHOWWINDOW);  
  }  
  break;  
 case bsMax:  
  {  
   m\_ButtonState = bsMax;  
   ShowWindow(SW\_SHOWMAXIMIZED);  
   m\_IsMax = FALSE;  
  }  
  break;  
 case bsRes:  
  {     
   ShowWindow(SW\_RESTORE);  
   m\_IsMax = TRUE;  
  }  
  break;  
 }

//------------------------------------------------------------------------------