

用C 品尝Vista美味[3] PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/245/2021\\_2022\\_\\_E7\\_94\\_A8C\\_\\_\\_E5\\_93\\_81\\_E5\\_c97\\_245993.htm](https://www.100test.com/kao_ti2020/245/2021_2022__E7_94_A8C___E5_93_81_E5_c97_245993.htm) 修改之后，底部的100像素现在是毛玻璃效果了。

在毛玻璃区域添加文本 在窗口中添加毛玻璃效果是比较简单的部分，但要把自己的界面元素（UI）添加到毛玻璃之上，就有点难度了。因为必须一直保持像素的alpha值，所以就要用到那些可以理解并适当设置alpha的绘图API。坏消息是，GDI函数差不多全部不理睬alpha--唯一剩下的API则为带有SRCCOPY光栅操作的BitBlt()函数了，因此，程序必须使用GDI 或主题API来进行绘图，这些API都是时刻不忘alpha的。在Vista中，有关毛玻璃的效果一般用在表示程序状态的区域（取代了通用控件中的状态栏），例如

，Windows Media Player 11就在窗口底部的毛玻璃区域显示播放控制与当前歌曲信息：以下，将演示怎样在毛玻璃区域上绘制文本，并怎样在文本上添加发光效果，以便文本在任何背景上都方便阅读。使用正确的字体 Vista已经彻底放弃使用MS Sans Serif与Tahoma字体，转而把Segoe UI作为默认的UI字体。我们的程序也应该使用Segoe UI字体，所以，将会在基于当前主题的情况下创建一个字体。如果主题被禁用（如用户正在使用Windows经典颜色方案），那我们就使用SystemParametersInfo() API。首先，需要在CMainFrame中添加主题支持，这一点非常简单，因为WTL已经有一个用于处理主题类：CThemeImpl。我们可把CThemeImpl添加到继承列表，并把消息链接至CThemeImpl，以便在当前主题改变时，程序可以得到相应的通知。class CMainFrame :public

```
CFrameWindowImpl,public CMessageFilter,public CThemeImpl{
// ... BEGIN_MSG_MAP(CMainFrame)
CHAIN_MSG_MAP(CThemeImpl) // ... END_MSG_MAP()
protected: CFont m_font. //用于绘制文本的字体}.
在CMainFrame的构造函数中，我们调用
了CThemeImpl::SetThemeClassList()，其指定了我们正在使用
哪一个主题的窗口类。对一般窗口来说（即不是普通控件的
窗口），名称为"globals"。CMainFrame::CMainFrame(){
SetThemeClassList ( L"globals" ).} 最后，在OnCreate()中，从主
题中读取字体信息，并创建一个字体自用：LRESULT
CMainFrame::OnCreate ( LPCREATESTRUCT lpcs ){ // ... //决定
在文本中使用哪一种字体 LOGFONT lf = {0}. if (
!IsThemeNull() ) GetThemeSysFont ( TMT_MSGBOXFONT,
&ncm, false ). lf = ncm.lfMessageFont. }
m_font.CreateFontIndirect ( &lf ). return 0.} 100Test 下载频
道开通，各类考试题目直接下载。详细请访问
www.100test.com
```