**C#WinForm程序设计——系统托盘NotifyIcon控件**

2011-06-23 17:04:27|  分类： [学习资料](http://yangyong2008620.blog.163.com/blog/#m=0&t=1&c=fks_087067092082088074083081086095092083088067083087080065) |  标签：[系统托盘](http://yangyong2008620.blog.163.com/blog/" \l "m=0&t=3&c=%E7%B3%BB%E7%BB%9F%E6%89%98%E7%9B%98)   |举报 |字号 订阅

1.如何实现托盘功能:  
在VS2005中直接添加notifyIcon控件,然后设置下icon属性,给其设置个图标即可,使用托盘功能.  
但是托盘并不能实现我们要求的功能,具体的功能实现,需要我们手工添加代码实现.  
2.如何最小化时自动到托盘

 private void Form1\_SizeChanged(object sender, EventArgs e)  
        {  
            if (WindowState == FormWindowState.Minimized)  
            {  
                this.Hide();  
                this.notifyIcon1.Visible = true;  
                this.notifyIcon1.ShowBalloonTip(30, "注意", "大家好，这是一个事例", ToolTipIcon.Info);  
            }

        }  
private void Form1\_SizeChanged(object sender, System.EventArgs e)   
{   
  if (this.WindowState == FormWindowState.Minimized)   
  {   
  this.Visible = false;   
  this.notifyIcon1.Visible = true;   
  }   
}   
3.如何双击托盘恢复原状  
private void notifyIcon1\_MouseDoubleClick(object sender, System.EventArgs e)   
{   
  this.Visible = true;   
  this.WindowState = FormWindowState.Normal;   
  this.notifyIcon1.Visible = false;   
}   
4.实现托盘的闪烁功能(如QQ有消息时的闪烁)  
  (1).首先我们在空白窗体中拖入一个NotifyIcon控件和定时控件  
  private System.Windows.Forms.NotifyIcon notifyIcon1;  
  private System.Windows.Forms.Timer timer1;  
  (2).其次,我们准备两张ico图片,用来显示在任务栏,其中一张可用透明的ico图片,分别叫做1.ico和2.ico;并且建立两个icon对象分别用来存放两个ico图片;  
  private Icon ico1 = new Icon("1.ico");  
  private Icon ico2 = new Icon("2.ICO");//透明的图标  
  (3).在Form\_load中初始化notifyicon:  
  private void Form1\_Load(object sender, System.EventArgs e)  
  {  
  this.notifyIcon1.Icon=ico1;//设置程序刚运行时显示在任务栏的图标  
  this.timer1.Enable = true;//将定时控件设为启用,默认为false;  
  }  
  (4).先设置一个全局变量 i ,用来控制图片索引,然后创建定时事件,双击定时控件就可以编辑  
  int i=0;  
  private void timer1\_Tick(object sender, System.EventArgs e)  
  {  
//如果i=0则让任务栏图标变为透明的图标并且退出  
  if(i<1)  
  {  
  this.notifyIcon1.Icon=ico2;  
  i++;  
  return;  
  }  
//如果i!=0,就让任务栏图标变为ico1,并将i置为0;  
  else  
  this.notifyIcon1.Icon=ico1;  
  i=0;  
  }

（2）NotifyIcon控件的doubleclick事件及两个menuitem的click事件：

 private void notifyIcon1\_DoubleClick(object sender, EventArgs e)  
        {  
            this.Show();  
            this.WindowState = FormWindowState.Normal;  
        }

        private void toolStripMenuItem2\_Click(object sender, EventArgs e)  
        {  
            this.Close();  
        }

        private void toolStripMenuItem1\_Click(object sender, EventArgs e)  
        {  
            this.notifyIcon1.Visible = false;  
            this.Show();  
            this.WindowState = FormWindowState.Normal;  
        }

 private void Form1\_FormClosing(object sender, FormClosingEventArgs e)//点关闭隐藏程序  
        {  
            if (e.CloseReason == CloseReason.UserClosing)  
            {  
                e.Cancel = true;

                notifyIcon1.Visible = true;  
                this.WindowState = FormWindowState.Minimized;  
                this.Visible = false;  
                notifyIcon1.ShowBalloonTip(1, "提示", "程序依然在运行！", ToolTipIcon.Info);  
            }  
        }

**最小化时 隐藏表单， 显示图标（默认是 显示） 并显示提示文字 ：**

private void Form1\_SizeChanged(object sender, EventArgs e)

{

Form fm = sender as Form;

if (fm.WindowState == FormWindowState.Minimized)

{

SingleInstance.ShowWindow(fm.Handle, SW.SW\_HIDE);

this.notifyIcon1.Visible = true;

this.notifyIcon1.ShowBalloonTip(3000, "Hello", "I am test icon", ToolTipIcon.Info);

}

}

private void timer1\_Tick(object sender, EventArgs e)

{

this.notifyIcon1.ShowBalloonTip(3000, "Message", "Interval 3000", ToolTipIcon.Warning);

}

**点击图标时， 切换显示与隐藏：**

private void notifyIcon1\_MouseClick(object sender, MouseEventArgs e)

{

if (SingleInstance.IsWindowVisible(this.Handle))

{

SingleInstance.ShowWindow(this.Handle, SW.SW\_HIDE);

this.notifyIcon1.Visible = true;

this.notifyIcon1.ShowBalloonTip(0, "Good", "Hide Again", ToolTipIcon.Info);

} else {

SingleInstance.ShowWindow(this.Handle, SW.SW\_SHOWMINIMIZED);

SingleInstance.SetForegroundWindow(this.Handle);

this.notifyIcon1.ShowBalloonTip(0, "Good", "Show Again", ToolTipIcon.Info);

}

}

**C++　显示与隐藏表单**

public enum SW

{

SW\_HIDE = 0,

SW\_SHOWNORMAL = 1,

SW\_NORMAL = 1,

SW\_SHOWMINIMIZED = 2,

SW\_SHOWMAXIMIZED = 3,

SW\_MAXIMIZE = 3,

SW\_SHOWNOACTIVATE = 4,

SW\_SHOW = 5,

SW\_MINIMIZE = 6,

SW\_SHOWMINNOACTIVE = 7,

SW\_SHOWNA = 8,

SW\_RESTORE = 9,

SW\_SHOWDEFAULT = 10,

SW\_MAX = 10

}

**导入Ｃ＋＋　函数**

[DllImport("User32.dll")]

public static extern bool ShowWindowAsync(IntPtr hWnd, SW cmdShow);

　[DllImport("User32.dll")]

public static extern bool ShowWindow(IntPtr hWnd, SW cmdShow);

　[DllImport("User32.dll")]

public static extern bool SetForegroundWindow(IntPtr hWnd);

　[DllImport("User32.dll")]

public static extern bool IsWindowVisible(IntPtr hWnd);