

---

Subject: tray icon & balloon notification

Posted by [hojtsy](#) on Wed, 01 Feb 2006 15:32:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

I would like to to put an icon into the tray (notification area) on Windows. I also would like to display notification balloons even when my application is not in focus. Is this supported by U++?

---

---

Subject: Re: tray icon & balloon notification

Posted by [mirek](#) on Wed, 01 Feb 2006 17:01:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Define "supported"

Yes, you can do it with U++ app. Anyway, there is no direct cross-platform support, you will have to use Win32 API.

Mirek

---

---

Subject: Re: tray icon & balloon notification

Posted by [unodgs](#) on Thu, 02 Feb 2006 07:51:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

I have modified a TaskbarButton class posted by Tom to work with my Ultimate Player (it was not compatybile with newest upp). It is not finished, but you can use it

This is how I use it:

```
struct MyApp : TopWindow
{
    TaskbarButton tray;
}
```

main...

MyApp app;

```
app.Open()
app.tray.Icon(Mylcon()).Label("MyApp");
app.tray.Init(app.GetHWND());
app.Run();
```

Use WhenDouble and WhenBar callbacks to add action for double click and popup menu.

Here it is the question to Mirek. As you can see I have to open appilication first to get the HWND for tray. It isn't the upp way . I saw that Tom was using (overloading) CratelInit(HWND hwnd)

method which is not supported now and which was called when hwnd was known (I suppose so). This was very useful and made code much cleaner. Is it possible to get this method back?

---

## File Attachments

1) [TaskbarButton.zip](#), downloaded 1510 times

---

---

Subject: Re: tray icon & balloon notification  
Posted by [mirek](#) on Thu, 02 Feb 2006 19:38:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

What about something like

`app.tray.Open(app) ?`

Mirek

---

---

Subject: Re: tray icon & balloon notification  
Posted by [unodgs](#) on Thu, 02 Feb 2006 20:32:19 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hmm, what you mean by that?

Are you suggesting adding Open method to the TaskbarButton class like this one...

```
TaskbarButton::Open(TopWindow &wnd)
{
    wnd.Open()
}
```

.. to reduce main initialize code?

Frankly I would like to have a better solution. I had the same problem with other 3rd party libraries which required to pass to them hwnd value. What about adding WhenShow callback in TopWindow class which should be called when hwnd is accessible?

---

---

Subject: Re: tray icon & balloon notification  
Posted by [mirek](#) on Thu, 02 Feb 2006 22:00:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

How about overriding Ctrl::State and catching "OPEN"/"CLOSE" ?

Mirek

---

---

Subject: Re: tray icon & balloon notification  
Posted by [unodgs](#) on Fri, 03 Feb 2006 07:45:57 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Yes, I know, but to make the condition reason == OPEN true I need call Add in main application constructor:

```
App()
{
    Add(tray);
}
```

I don't want to do that, because tray control is not a layout control and the final result is a red box (in debug mode) displayed in the left-upper corner (it is ok as I did not set position and tray doesn't have a Paint method).

Could you add another value for reason - WINDOW? In this state GetHWND() should return a proper handle.

---

Subject: Re: tray icon & balloon notification  
Posted by [mirek](#) on Fri, 03 Feb 2006 10:42:27 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, I guess the confusion here is that HWND used for tray icon is quite unrelated to TopWindow. You would need to override TopWindow::WindowProc to handle it correctly.

I have tried different solution- tray icon with separate HWND and it seems to work fine - per popular demand, TrayIcon is now the part of U++.

Example:

```
struct App : TrayIcon {
    void Info() {
        PromptOK("TrayIcon launched this prompt!");
    }
}
```

```
void Menu(Bar& bar) {
    bar.Add("Info..", THISBACK(Info));
    bar.Separator();
    bar.Add("Exit", THISBACK(Break));
}
```

```
typedef App CLASSNAME;
```

```
App() {
    WhenBar = THISBACK(Menu);
}
```

```
WhenLeftDouble = THISBACK(Info);  
Icon(CtrlImg::File());  
Tip("This is U++ tray icon");  
}  
};
```

```
GUI_APP_MAIN  
{  
    App().Run();  
}
```

---

Subject: Re: tray icon & balloon notification  
Posted by [hojtsy](#) on Fri, 03 Feb 2006 11:45:29 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Can the icon of a TrayIcon be changed after construction?

Is there some wrapper function which can be used to hide the taskbar button of an application, so that the application can "minimize to the tray", or do I have to use Win32 for that?

I would like to monitor a web server from an u++ app and display an auto-hiding balloon notification if the server fails. Also the status could be displayed in a tooltip when the mouse is hovered over the tray icon. Are there methods for the notification and tooltip in the TrayIcon?

---

---

Subject: Re: tray icon & balloon notification  
Posted by [mirek](#) on Fri, 03 Feb 2006 12:38:42 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Icon/Tip can be changed whenever you want.

To "minimize to tray", my guess is that simple minimize+close would work quite well.

"balloon" is, AFAIK, not related to tray icon (but I might be wrong).

Mirek

---

---

Subject: Re: tray icon & balloon notification  
Posted by [hojtsy](#) on Fri, 03 Feb 2006 14:54:17 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

luzr wrote on Fri, 03 February 2006 07:38 To "minimize to tray", my guess is that simple minimize+close would work quite well. I think I will rather hide it with WndShow(false). I would like

to keep the internal state of the window unchanged when "minimized" to the tray.

Other question: Could you add a TopWindow method to set the window topmost, which calls SetWindowPos(hWnd, HWND\_TOPMOST, ...) ?

---

Subject: Re: tray icon & balloon notification  
Posted by [mirek](#) on Fri, 03 Feb 2006 16:04:36 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, there is no WndShow in U++, but Hide calls it for TopWindow... And you are right, hiding is good as well.

OTOH, U++ is designed that it mostly does not matter. Only "internal state" that really exists is focus - and that one is stored as "active focus" in TopWindow anyway.

Other answer:

Well, I can adding TopMost modifier, but for now please use:

```
TopWindow& ExStyle(dword _exstyle);  
dword    GetExStyle() const;
```

(like ExStyle(GetExStyle()) | WS\_EX\_TOPMOST)

(note that it works the U++ way - does not matter whether TopWindow is open or not).

Mirek

---

Subject: Re: tray icon & balloon notification  
Posted by [hojtsy](#) on Fri, 03 Feb 2006 16:19:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

luzr wrote on Fri, 03 February 2006 11:04Well, there is no WndShow in U++It is a method of Ctrl.  
But I noticed now that it is private.  
ExStyle(GetExStyle()) | WS\_EX\_TOPMOST) is OK, no need for another method.

---

Subject: Re: tray icon & balloon notification  
Posted by [hojtsy](#) on Mon, 06 Feb 2006 12:38:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi I am having a problem with making a window topmost (floating above windows of other applications). With this function I can make a window topmost:

```
::SetWindowPos(GetHWND(), HWND_TOPMOST, 0,0,0,0, (UINT)(SWP_NOMOVE | SWP_NOSIZE));
```

But this other one you suggested does not work:

```
ExStyle(GetExStyle() | WS_EX_TOPMOST);
```

Additionally on linux the GetExStyle is undefined. It would be better to have a dummy function returning default values.

---

---

Subject: Re: tray icon & balloon notification

Posted by [mirek](#) on Mon, 06 Feb 2006 13:11:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

OK, I hope ExStyle is fixed now.

As for empty implementations in Linux, I do not agree - I do not like the idea of silently ignored problems.

Use `#ifdef PLATFORM_WIN32` conditional compilation instead.

Mirek

---

---

Subject: Re: tray icon & balloon notification

Posted by [hojtsy](#) on Tue, 07 Feb 2006 12:09:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

`ExStyle(GetExStyle() | WS_EX_TOPMOST);` Still not OK in 602 beta. It results in some kind of freezing of the window. It can no longer be resized, only moved, controls in the window no longer react to mouse clicks, and right click on the title bar opens up an incorrectly aligned menu, which contains the only way to close the application (not counting killing from Task Manager).

---

---

Subject: Re: tray icon & balloon notification

Posted by [mirek](#) on Tue, 07 Feb 2006 21:40:48 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Well, I hope I have fixed it, however, later I have found that it cannot work for `WS_EX_TOPMOST`:

[http://msdn.microsoft.com/archive/default.asp?url=/archive/e n-us/win9x/verdiff\\_2awj.asp](http://msdn.microsoft.com/archive/default.asp?url=/archive/e n-us/win9x/verdiff_2awj.asp)

so `SetWindowPos` is now the best option...

Mirek

---

Subject: Re: tray icon & balloon notification  
Posted by [hojtsy](#) on Fri, 10 Feb 2006 09:26:53 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

It is typical for popup menus created by tray icons to have a menu entry with bold font. Can this be accomplished with U++?

---

---

Subject: Re: tray icon & balloon notification  
Posted by [mirek](#) on Fri, 10 Feb 2006 09:45:05 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Yes, but just accidentally and not very nice way:

Typecast result of bar.Add... to MenuItemBase and use SetFont.

Putting to todo.... (but I do not see any nice way now..)

Mirek

---