

# Animate tray application windows as they open and close

---

Have you noticed that when you close the window of a tray application it often doesn't actually close the application but simply hides the window and leaves the application running in the tray: it effectively minimizes the application to the tray. There's usually no visual clue that this has happened as there would be if you minimized a normal application.

We can change this behaviour by using the *DrawAnimatedRects* Windows API function. (See *Tip #46* for an explanation of this function.)

Here is a little static class that your tray applications can use to animate the window caption, zooming to the tray when the window closes and zooming from the tray when it opens.

```
unit UTrayZoom;

interface

uses
  Windows;

type
  TTrayZoom = class(TObject)
  private
    class function GetTrayRect: TRect;
    class procedure DoZoom(const Wnd: HWND; const Src, Dest: TRect);
  public
    class procedure ZoomToTray(const Wnd: HWND);
    class procedure ZoomFromTray(const Wnd: HWND);
  end;

implementation

class procedure TTrayZoom.DoZoom(const Wnd: HWND; const Src, Dest: TRect);
begin
  DrawAnimatedRects(Wnd, IDANI_CAPTION, Src, Dest);
end;

class function TTrayZoom.GetTrayRect: TRect;
var
  TaskbarWnd, TrayWnd: HWND;
begin
  TaskbarWnd := FindWindow('Shell_TrayWnd', nil);
  TrayWnd := FindWindowEx(TaskbarWnd, 0, 'TrayNotifyWnd', nil);
  GetWindowRect(TrayWnd, Result);
end;

class procedure TTrayZoom.ZoomFromTray(const Wnd: HWND);
var
  WndRect: TRect;
begin
  GetWindowRect(Wnd, WndRect);
  DoZoom(Wnd, GetTrayRect, WndRect);
end;

class procedure TTrayZoom.ZoomToTray(const Wnd: HWND);
var
  WndRect: TRect;
begin
  GetWindowRect(Wnd, WndRect);
  DoZoom(Wnd, WndRect, GetTrayRect);
end;

end.
```

The *DoZoom* method simply wraps the call to *DrawAnimatedRects*. *GetTrayRect* gets the bounding rectangle of the tray area of the task bar. (See *Tip #47* for details of how to find the tray window handle.) The two public methods *ZoomToTray* and *ZoomFromTray* zoom the specified window to and from the tray respectively. The *Wnd* parameter to these methods should be the window handle of the form that you are zooming to and from the tray.

## Usage

---

When your tray application is opening from the tray call *TTrayZoom.ZoomFromTray*. You may possibly do this in a menu item's *OnClick* event handler. When your application is closing to the tray call *TTrayZoom.ZoomToTray*. This may be done in the form's *OnClose* or *OnCloseQuery* event handlers. Exactly where and when you call *TTrayZoom*'s methods will vary from application to application.

---

Author:	Peter Johnson
Contributor:	Peter Johnson
Added:	2007-08-14
Last updated:	2007-08-14

---

*Copyright* © Peter Johnson (*DelphiDabbler*) 2002-2018