



# DrawCursor

## Summary

DrawCursor is a program that enables you to draw on the desktop with the mouse. DrawCursor has only been tested on Windows XP and probably doesn't work on Windows Vista.

## Usage

When you start DrawCursor the only thing that happens is that the DrawCursor Icon appears in the System Tray in the lower right corner.

If you right click on the Icon you can choose the different available configurations of DrawCursor.

DrawCursor has 3 different states:

1. Normal mode – the desktop works like it usually does.
2. Draw Mode – you can draw on the screen, if you press the left mouse button. You cannot have any other interaction with the desktop.
3. Semi transparent mode – you can see what is drawn, and can interact with the desktop. But you cannot draw anything.

DrawCursor always starts up in normal mode. But if you press F7 you toggle between the 3 modes.

## Configuration

In the Installation directory, typically C:\Program Files\JERN\DrawCursor\ there is a folder called DCXML.

In this directory there is a Schema file "DCConfiguration.xsd" and some XML files. The XML files have to comply with the Schema.

For each XML file in the DCXML folder a submenu is added to the Configuration Menu of the DrawCursor Tray Icon.

An example file is shown here

```
<?xml version="1.0" encoding="utf-8" ?>
<DCConfiguration>
  <Name>Small Orange</Name>
  <Default>true</Default>
  <MarkerSize>5</MarkerSize>
  <MarkerColor>OrangeRed</MarkerColor>
</DCConfiguration>
```

As you can see the XML file has 4 possible settings.



# DrawCursor

1. Name: The name of the Submenu.
2. Default: Denotes if the file contains the default configuration of DrawCursor. I.e. the one chosen by the system when DrawCursor starts. One and only one file has to be default. This is actually not validated by the program. If the Default setting is not present in the file, it means the same as if Default is set to false.
3. MarkerSize: the size in pixels of the Cursor.
4. MarkerColor: The color of the Cursor. Any Color that matches the `System.Media.Windows.Color` struct can be used.

## Inner workings

DrawCursor works by utilizing WPF, WinForms, and some Win32 API's.

For the Tray Icon it uses WinForms. WPF hasn't got a Tray Icon control yet.

For trapping the F7 Key, it uses a Win32 Keyboard Hook. This is because it is not possible to use .NET for trapping Keyboard events that are not tied to a specific window.

For the DrawMode it uses a WPF InkCanvas that encompasses the entire screen. First the screen is captured in a bitmap that gives the illusion that you can actually draw on the screen. For this capture another Win32 API is used.

For the semi transparent mode, the Window containing the InkCanvas, and bitmap is made 50% transparent. But this is not enough. To make it possible to interact with the desktop another Win32 API is used. `SetWindowLong(hwnd, GWL_EXSTYLE, ExtendedStyle | WS_EX_TRANSPARENT)`

The Configuration is done by reading each XML file into a simple object, using LinqToXML. Linq is also used to Create the Submenu Items and hooking up the Click events to the respective XML files.