

SEPTEMBER DEVELOPER NEWS

INSIDE THIS ISSUE:

- Which Files do You Need to Install to Use BiAutoprint.ocx in a VB.NET Application?* **1**
- Printing Documents Programmatically* **1**
- How to Set Word Coordinates in the Text Output File* **2**
- How to Handle BiDisp Mouse Events in Mapping and SelectImageArea Modes* **3**

The BLACK ICE NEWSLETTER is published by Black Ice Software, LLC. The contents of this newsletter in its entirety are Copyright © 2007 by Black Ice Software, LLC. 292 Route 101, Salzburg Square, Amherst, NH 03031, USA. Black Ice Software, LLC. does hereby give permission to reproduce material contained in this newsletter, provided credit is given to the source, and a copy of the publication that the material appears in is sent to Black Ice Software at the above address.

Phone: (603) 673-1019

Fax: (603) 672-4112

sales@blackice.com

www.blackice.com

WHICH FILES DO YOU NEED TO INSTALL TO USE BIAUTOPRINT.OCX IN A VB.NET APPLICATION?

There are some misunderstandings about the files of the BiAutoprint control. Which files should you install in your install project? We can differentiate two cases:

1. Statically

In this case you use the BiAutoprint.ocx statically in your VB.NET application. This means you drag and drop the registered BiAutoprint control to a form in your project. When you compile the application the development environment automatically creates the following files:

```
AxInterop.BIAUTOPRINTLib.dll
Interop.BIAUTOPRINTLib.dll
```

These files are required to run your application. They should be placed at the same location as your application. Of course the BiAutoprint.dll and BiAutoprint.ocx are also required. The BiAutoprint.ocx must be registered before running the application. and

because the BiAutoprint.ocx calls functions from BiAutoprint.dll the ocx should be able to reach the dll, so the dll must be in the same directory as the ocx (or it can be in the path variable). The simplest solution is putting all dlls and ocxs in the same directory as your application.

2. Dynamically

In this case you use the BiAutoPrint control dynamically and you don't drag and drop the control to a form in your project. If you compile your project only the Interop.BIAUTOPRINTLib.dll will be automatically generated by the development environment.

In this case besides the Interop.BIAUTOPRINTLib.dll, BiAutoprint.dll and BiAutoprint.ocx you also need the BiAutoprint.lic file. The BiAutoprint.lic must also be placed at the same directory as the BiAutoprint.ocx.

The following VB.NET code snippet shows you how to use the BiAutoprint control dynamically.

```
' declaration
Private BiAutoPrint As BIAUTOPRINTLib.BiAutoPrint

' Initialize BiAutoPrint control
BiAutoPrint = New BIAUTOPRINTLib.BiAutoPrint()

' Calling a method from the ocx
BiAutoPrint.BIAPInitialize()
()
```

PRINTING DOCUMENTS PROGRAMMATICALLY

If you want to print a document programmatically you can try using the ShellExecute windows API function. The ShellExecute function tries to print the document using the application that is assigned to

the specified file type in the system registry. If an application is assigned to the file type and it supports the "print" or "printto" feature from the command line you can print the document easily.

The following example on page 2 demonstrates how to print a bitmap programmatically in C++:

Example from Printing Documents Programmatically, Page 1

```

SHELLEXECUTEINFO sei;

// Initialixe structure
ZeroMemory(&sei, sizeof(SHELLEXECUTEINFO));
sei.cbSize = sizeof(SHELLEXECUTEINFO );

sei.lpFile = _T("C:\\Test\\picture.bmp");
sei.fMask = SEE_MASK_NOCLOSEPROCESS | SEE_MASK_FLAG_NO_UI |
            SEE_MASK_FLAG_DDEWAIT;
sei.lpVerb = TEXT("printto");
sei.nShow = SW_HIDE;
sei.lpParameters = _T("\\Black Ice Color\\"); // Printer name
sei.hProcess = hProcess;

if(!ShellExecuteEx(&sei))
{
    // Error occurred. See GetLastError
    DWORD dwError = GetLastError();
    return ;
}

```

HOW TO SET WORD COORDINATES IN THE TEXT OUTPUT FILE

By default the Black Ice printer driver creates one "style and position" entry for text having the same style. For example:

```

[[1063,300,1488,344][10][,,]
[Courier New]New Text Document

```

If you would like to generate an entry for each word, you must add the following section to the INI file of the printer driver:

```

[Text Output Settings]
Display Word Coordinates=1

```

If this section is in the INI file, the generated text output file will contain style and position information separated for each word. For example:

```

[[1063,300,1138,344][10][,,]
[Courier New]New
[[1138,300,1163,344][10][,,]
[Courier New]
[[1163,300,1263,344][10][,,]
[Courier New]Text
[[1263,300,1288,344][10][,,]
[Courier New]
[[1288,300,1488,344][10][,,]
[Courier New]Document

```

This printer driver INI file is in the windows' driver directory (at 2 locations), for example C:\WINDOWS\system32\spool\drivers\w32x86\3\ and C:\WINDOWS\system32\spool\drivers\w32x86.



Notice: The Printer Drivers for Windows 95 / 98 / ME, and the Voice C++/ActiveX Toolkit have been discontinued as of the end of this year. Black Ice Software will no longer provide support for these products after January 1st, 2008.

HOW TO HANDLE BIDISP MOUSE EVENTS IN MAPPING AND SELECTIMAGEAREA MODES

If an application uses more than one display object (BiDisp.ocx control) a mouse event could be a request to handle the get focus, on click, mouse down, or mouse up events. This doesn't cause any problems if the Mapping property is set to false, and the SelectImageArea property is also set to false. In this case the user can use the event handler functions to handle the previously mentioned events. If either the Mapping or SelectImageArea properties is set to true, then the control handles these events and the user cannot catch them.

One of the easiest workarounds is the following:

The critical properties of the controls (Mapping and SelectImageArea) could be contained in global variables. If a control loses the focus, you can set these properties to false. In this case the user can handle the mouse events and when a control gets the focus again one can set the properties using the global variables.

So, you can set these properties to false when a LostFocus event is generated and set these properties to true when a GotFocus event is generated as you can see in the following VB.NET sample:

```
Public Class Form1

    Private Sub Form1_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

        ' DIB handler
        Dim hDib As Long
        ' display settings
        BiDisp1.ScaleMode = BIDISPLib.enumScaleMode.tifsmNoScale
        BiDisp1.ScrollBars = BIDISPLib.enumScrollBars.tifsbBoth
        BiDisp2.ScaleMode = BIDISPLib.enumScaleMode.tifsmNoScale
        BiDisp2.ScrollBars = BIDISPLib.enumScrollBars.tifsbBoth
        ' load the tiff file
        hDib = BITiff1.LoadTiffIntoDIB("test.tif", 0, False)
        BiDisp1.hDib = hDib
        BiDisp2.hDib = hDib
        ' set Mapping mode
        BiDisp1.Mapping = False
        BiDisp2.Mapping = False
    End Sub

    Private Sub BiDisp1_GotFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles BiDisp1.GotFocus
        ' set Mapping mode
        BiDisp1.Mapping = True
    End Sub

    Private Sub BiDisp1_LostFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles BiDisp1.LostFocus
        ' set Mapping mode
        BiDisp1.Mapping = False
    End Sub

    ...

End Class
```

This sample contains a form (Form1) with two BiDisp (BiDisp1, BiDisp2) and a BiTiff (BiTiff1) controls. It references just for the Mapping property, but handling of the SelectImageArea property is the same. The implementation of the BiDisp2_GotFocus and the BiDisp2_LostFocus event handler functions are the same as the implementation of the BiDisp1_GotFocus and the BiDisp1_LostFocus functions.

Time to upgrade?



Latest Version Numbers

Printer Drivers

Impact Products

Vista / XP Color, ColorPlus, EMF, Mono, PDF	10.03	5/9/07
2003 Terminal Servers / Citrix Color, ColorPlus, EMF, Mono, PDF	10.03	5/9/07

Impact Fax Server	7.31	5/18/07
Impact Fax Broadcast	5.11	11/29/06
Impact ColorFax	7.02	8/15/06

Fax, Voice, and Image Toolkits

Internet Tools

Fax & Voice C++/ActiveX	12.40	4/12/07
Document Imaging SDK/ActiveX	10.77	5/3/07
Image PDF Plug-in	10.77	5/3/07
PDF SDK Professional	2.0	6/11/07
Annotation SDK/ActiveX	10.77	5/3/07
Image SDK/ActiveX	10.77	5/3/07
Tiff SDK/ActiveX	10.77	5/3/07
Cover Page Generator SDK/ActiveX	10.77	5/3/07
Barcode SDK/ActiveX	5.10	5/7/07

Print2Email	5.13	10/19/06
Tiff Viewer Plug-in - Complete	8.01	5/3/07
Print Monitoring Server	3.00	9/16/04
Print2RDP	3.0	9/6/06
Print2FTP	2.02	8/15/06

Free Software

Impact ColorFax Lite	7.02	8/15/06
Tiff Viewer Plug-in - Free Version	8.01	5/3/07
ModemWeasel	2.00	8/01/02