OCTOBER DEVELOPER NEWSLETTER

Black Ice Software VOLUME 11, ISSUE 10 OCTOBER, 2006

Document Imaging SDK—Thumbnail Control

Black Ice Software has enhanced the thumbnail control to provide. better quality display. The new methods in the Black Ice Thumbnail control allow one to display images in three different ways, fast display with low quality, slow display with high quality, and mixed mode display.

What this means for programmers, is you can chose which method to use. One can use the low quality method for fast display, one can use the high quality display method with slower display or use the new "mixed" method, with which one can now have all of the thumbnails load in a low quality for speed, and then replace the thumbnails with a higher quality image afterwards. As the thumbnails get displayed and replace take place, users will experience a positive change of the image to a better quality. ■

Use of the new method:

SetLoadingMode(loadingType)

where loadingType can be one of the following:

LT_LOWQUALITY load thumbnails in low quality, but it's fast
LT_HIGHQUALITY load thumbnails in high quality, but it's slow
LT_MIXED load in low quality, then refresh in high quality

User can get this property too:

loadingType = GetLoadingMode()

Inside This Issue:

Printer Drivers—Watermark

Black Ice Software has integrated it's watermarking capability into it's redirect printing. With this new feature you are able to print a watermark from the Black Ice printer on a physical printer very easily. The sophisticated watermarking option of the Black Ice Printer Driver is redirected to a physical printer, or group of physical printers automatically.

Now you can easily make sure all documents printed have your company logo, without needing to alter the document itself. All you need to do is set the watermark in the Black Ice printer properties, and the driver does the rest! This will make your work easier, save you time and precious disk space, and keeps your documents' designs similar, while preserving the original document.

You can set the watermark to be used in the Black Ice printer driver's properties in a few different ways. You can have one watermark apply to every page printed, or you can have the watermark only show on the first page printed, or you can have 2 different watermarks: one watermark on the first page, and a different watermark for the rest of the printed pages.

You can use TIF, BMP, or JPG files as a watermark image, and you can specify the watermark's brightness as well as how it fits on the page. You can also select different images for portrait or landscaped pages. Download the latest demo version of the Printer Drivers to try it out! ■

Phone: (603) 673-1019

Fax: (603) 672-4112

Sales@blackice.com

How to Wait for DTMF Digits Using the Voice ActiveX Control in C#

There is a feature available in the Voice ActiveX control to wait for and receive DTMF digits for voice modems, Dialogic, NMS, and Brooktrout cards. There is a new sample, "RecDTMF_FaxOrVoice" for C# to demonstrate this feature.

If the user selects "Wait for DTMF" a window form will appear, which asks for the number of DTMF digits (nNum) to wait for, and also a delimiter digit (nDelim) after which no more DTMF digits will be received. DTMF digits can be the numbers (0 – 9) and the '*' and '#' symbols.

This is important for IVR applications to ensure they have all the necessary data before processing. Waiting for a certain number of DTMF digits can ensure that a user has entered a full phone number, bank account number, or their entire social security number for example.

Below is one of the sample application to demonstrate this feature. The "WaitForDTMF" method of the Voice ActiveX control begins to wait for digits until the value of "nNum" number of digits or a delimiter digit arrives. Below is a code snippet showing this.

```
short nNum;
short nDelim;

WaitforDTMF dlgDTMF = new WaitforDTMF();
dlgDTMF.ShowDialog();

nNum = dlgDTMF.GetNumberOfDigits();
nDelim = dlgDTMF.GetDelimiter();

if ( nNum > 0 )
{
  if ( axVoiceOCX1.WaitForDTMF(lModemID, nNum, nDelim,
    20)==0 )
  {
  AddToLog("Wait for DTMF digits");
  closeButton.Enabled = false;
  nProgressStatus = cnsWaitDTMF;
  }
else
AddToLog("WaitForDTMF failed");
}
```

After the DTMF digits arrive, the "GetReceivedDTMFDigits" method can retrieve the digits in this way:

```
string receivedDigits =
axVoiceOCX1.GetReceivedDTMFDigits(lModemID);
```

The "TerminateProcess" method of the Voice OCX can stop the wait and receive process on the current modem:

```
axVoiceOCX1.TerminateProcess(lModemID);
```

Here "lModemID" is created by the "CreateModemObject" method of the Voice OCX.

Events that can occur while waiting for DTMF digits and how to handle them:

DTMFDigit: occurs immediately when a port has received a DTMF digit. This works only with modems.

```
private void axVoiceOCX1_DTMFDigit(object sender,
AxVOICEOCXLib._DVoiceOCXEvents_DTMFDigitEvent e)
{
 nProgressStatus = 0;
  AddToLog("Received DTMF digit(s): [" +
  Convert.ToChar(e.digit) + "]");
}
```

DTMFReceived: occurs when the specified numbers of digits are received. It works for either voice modems or voice cards.

```
private void axVoiceOCX1_DTMFReceived(object sender,
AxVOICEOCXLib._DVoiceOCXEvents_DTMFReceivedEvent e)
{
AddToLog("Received DTMF digit(s): [" +
axVoiceOCX1.GetReceivedDTMFDigits(lModemID) + "]");
   nPrStatus = cnsOnline;
   isDTMFTerminated = true;
axVoiceOCX1.TerminateProcess(lModemID);
  nProgressStatus = 0;
  closeButton.Enabled = true;
}
```

DTMFReceivedExt2: occurs when the specified numbers of digits are received during recording a playback file. The Offset parameter contains the playback file position in bytes when the event was fired. It works for either voice modems or voice cards.

```
private void axVoiceOCX1_DTMFReceivedExt2(object sender,
AxVOICEOCXLib._DVoiceOCXEvents_DTMFReceivedExt2Event e)
{
AddToLog("Received DTMF digit(s): [" +
axVoiceOCX1.GetReceivedDTMFDigits(lModemID) + "]");
  AddToLog("Offset:" + Convert.ToChar(e.offset));
  nPrStatus = cnsOnline;
  isDTMFTerminated = true;
axVoiceOCX1.TerminateProcess(lModemID);
  nProgressStatus = 0;
  closeButton.Enabled = true;
}
```

■

Printer Driver Tips and Tricks: MSI Install for Black Ice Printer Driver

This section will show you how to create your own printer driver MSI installation. The most simple printer driver MSI install copies the driver files to the required locations, adds the printer driver, print processor and print monitor, and finally the printer to the system.

When installing the Black Ice printer drivers on a system, developers must first check that the following criteria are met:

- Administrative privileges are required.
- The Black Ice printer drivers must be installed from the local console, they cannot be installed remotely, such as through a terminal server client.
- If terminal services are present on the system, even if used only for remote administration, the terminal server version of the Black Ice printer drivers must be used.

1. Copy the driver files

The following files must be installed to the specified locations before installing the Black Ice printer driver. The sample filenames between parentheses show the filenames of the Black Ice TIFF NT printer drivers.

Driver file (BiMDrvNT.dll) Target location: <*DRIVERDIR*>
Print processor file (BiMProNT.dll) Target location: <*PROCESSORDIR*>
Port monitor file (BiMMonNT.dll) Target location: <*SYSTEM*>
User interface dll (BiMUifNT.dll) Target location: <*DRIVERDIR*>
Resource dll (BiMResNT.dll) Target location: <*SYSTEM*> and <*DRIVERDIR*>
Driver remove dll (BiMRmvNT.dll) Target location: <*SYSTEM*>
Driver initialization file (BiMIniNT.ini) Target location: <*DRIVERDIR*>
Tiff dll (BiTiff.dll) Target location: <*SYSTEM*> and <*DRIVERDIR*>
JPEG dll (Jpeg32.dll) Target location: <*SYSTEM*> and <*DRIVERDIR*>
Port monitor utility dll (BiImg.dll) Target location: <*SYSTEM*>
Application Starter (BiMAppNT.dll) Target location: <*SYSTEM*>

<SYSTEM>: The system folder for the operating system, by default Windows\System32 directory.
<DRIVERDIR>: For NT / Windows 2000, the default location for the driver files is the \WINNT\system32\spool\drivers\w32x86\ and the \WINNT\system32/spool\drivers\w32x86\2\ directories (the files should be copied in both these directories). To retrieve the location of this folder, use the "dGetPrinterDriverDirectory" function from the InstallDLL.
<PROCESSORDIR>: For NT / Windows 2000, the default location for the print processor files is the \WINNT\system32\spool\prtprocs\w32x86 directory. To retrieve the location of this folder, use the "dGetPrintProcessorDirectory" function from the InstallDLL.

After the necessary files are copied successfully, you still have to install the Black Ice printer driver.

2. Install the printer driver

To actually install the printer driver a "custom action" is used. The custom action is a function call from the Install DLL (MYDLL.DLL) which is included in the RTK. The function called by the MSI installation from the Install DLL makes all of the necessary Win32 API calls to install each printer driver component, including the print processor, print monitor, port, printer driver and the printer. To simplify the install, all component names are written to a registry key, and the Install DLL function reads the names of the printer driver components from that key. This way, the setup developer can easily change the name of the components by changing the registry key values. In this case you should call one function (the name of the function is PrinterInstallMSI_TIFFNT in case of TIFF NT printer driver) to install the Black Ice printer driver.

When the Black Ice printer driver is installed, the printer settings will be read by the driver from the INI file (BiMIniNT.ini in the case of the Black Ice TIFF driver). The settings specified in the INI file will be set as default settings for every user on the system. If a setting is not specified in the INI file, the printer driver will use the default values hard coded in the driver, such as the JPEG file format for the color driver.

Create your customized printer driver

If you want to install more Black Ice printer drivers on a single system, or you should change the names of the printer driver files or the names of the printer components you have to be sure to:

Rename the driver files:
Port monitor
Print processor
User interface dll
Resource dll
INI file

NOTE: The following files can be shared on a system:
Tiff dll
Jpeg dll
Application starter
Port monitor utility dll

Include the new files into your MSI installation.
Rename all of the components which you associate with the new files, including:

Name of the printer driver
Name of the printer
Name of the port monitor
Name of the print processor
Name of the port

These values are stored in the registry. You can change these values by changing the registry entries. This is easily done in your MSI install project.

There are several additional features for changing the settings of the Black Ice printer driver at installation time. For more information see the printer driver RTK documentation. There are several install samples in the printer driver resource toolkit for demonstrating the process of the printer driver install. Visual Basic 6.0, Visual Basic .Net 2005, InstallShield and MSI install samples are all available in the printer driver RTK. ■