



Using ActiveX Window Image Capture

1. Overview

There are so many cases in which window image captures are necessary. Especially for the DAS ActiveX components, many of our customers request the component image capture tools to capture their screen designs for documentation, advisement or brochures.

The Window Image Capture utility developed by Dragonfly Automation Software has following features,

- 1). Capture any part of the screen with any size;
- 2). Capture any active window image or active child window image;
- 3). Add any note underneath the image;
- 4). Capture any background (hidden) window image of the same application;
- 5). Capture the background hidden child window image of the sample application;
- 6). Copy any captured image into Clipboard;
- 7). Copy any Clipboard image back to screen image containers.
- 8). Save the window images into BMP files;
- 9). Save the window images into JPEG files;

The most powerful feature of this utility is capability to capture the background window image event the window is invisible, which is so useful for developers to capture all the screens or windows of their applications without bringing all windows to the front. In particular, sometimes it's impossible to bring all screens or windows of one application to the front to capture a group of screens that are captured at the same time.

By embedding this utility into applications of people, they can develop many kinds of powerful screen capture applications, or include the window image capture function in their own developments. They do not need to buy the other separate tools to manually capture the images to achieve their purposes.

This utility is developed as Microsoft ATL ActiveX DLL, so it's very easy for the developers to integrate this function into their own applications, and distribute the designed screen capture functions to their customers.

2. Interfaces

The designers can use the interface methods to implement their own screen capture tools or embed it into their other field applications. There is no COM object defined in this DLL, i.e., "SCRNCAPTURELib.ImgCapture", where the following interface functions are defined.

Function CaptureWinImage(hwnd As Long, bWinActive As Boolean, text As String, pPic As IPictureDisp) As Boolean

By calling this function, the application can capture the window image and attach it to *IPictureDisp* pointer, i.e., *pPic*, meanwhile the image will be copied to the clipboard. If the call is successful, the *VARIANT_TRUE* (*TRUE* in VB) is returned, otherwise *VARIANT_FALSE* is returned.

hwnd : the window handle of which the window image will be captured;
bWinActive : determine the capture mode, i.e., Active Mode or InActive Mode;
text : specify the note underneath the image, if it's NULL string, there should be no note underneath the image;
pPic : pointer of *IPictureDisp* to retrieve the image.

Note: ActiveX mode: when the window image is captured, it's assumed that the window *hwnd* was an active window, there is no other top window on the top of it. Therefore the function just directly copy the screen part of this window to *pPic*. If the window *hwnd* actually is not an active window, there are some other windows overlap this window, or it is not visible, but *bWinActive* is set to *true*, then the other overlapped window images will be captured.

InActiveX mode: It is assumed that the window *hwnd* was an inactive window (invisible, or hidden or overlapped by other windows, or it's a background window), then the function will apply another algorithm to capture the image of this window. This mode just works only when the calling process and the window process are same.

Function CopyWinImageToClipboard(hwnd As Long, bWinActive As Boolean, text As String) As Boolean

By calling this function, the application can capture the window image and copy it to the clipboard. If the call is successful, the *VARIANT_TRUE* (*TRUE* in VB) is returned, otherwise *VARIANT_FALSE* is returned.

hwnd : the window handle of which the window image will be captured;
bWinActive : determine the capture mode, i.e., Active Mode or InActive Mode;
text : specify the note underneath the image, if it's NULL string, there should be no note underneath the image;

Function CapturePartWinImage(hwnd As Long, bWinActive As Boolean, text As String, left As Long, top As Long, width As Long, height As Long, pPic As IPictureDisp) As Boolean

This function can be used to capture a rectangle part of the window image and attach it to *IPictureDisp* pointer, i.e., *pPic*, meanwhile the image will be copied to the clipboard. If the call is successful, the *VARIANT_TRUE* (*TRUE* in VB) is returned, otherwise *VARIANT_FALSE* is returned.

hwnd : the window handle of which the partial window image will be captured;
bWinActive : determine the capture mode, i.e., Active Mode or InActive Mode;
text : specify the note underneath the image, if it's NULL string, there should be no note underneath the image;

left :Specify the X coordinate of the left-top point of the rectangle of the part image, *left*=0 means the left side of rectangle of the part image is same as the left side of the client rectangle of window *hwnd*.

top :Specify the Y coordinate of the left-top point of the rectangle of the part image, *top*=0 means the top side of rectangle of the part image is same as the top side of the client rectangle of window *hwnd*.

width :Specify the width of the rectangle of the part image.

height :Specify the height of the rectangle of the part image.

pPic :pointer of IPictureDisp to retrieve the image.

Function CaptureChildWinImage(ParentHwnd As Long, pX As Long, pY As Long, bWinActive As Boolean, text As String, pPic As IPictureDisp) As Boolean

This function is used to capture the window image of the window that contains the specified point (*pX*,*pY*) or the child window belonging to the parent window *ParentHwnd* that contains the specified point (*pX*,*pY*), and attach it to *IPictureDisp* pointer, i.e., *pPic*, meanwhile the image will be copied to the clipboard. If the call is successful, the *VARIANT_TRUE* (*TRUE* in VB) is returned, otherwise *VARIANT_FALSE* is returned.

ParentHwnd : the parent window handle of which the child window image will be captured, if *ParentHwnd*=*NULL*, the desktop window is the parent window.

pX :Specify the X value of the point (*pX*,*pY*) contained in the window or the child window.

pY :Specify the Y value of the point (*pX*,*pY*) contained in the window or the child window.

bWinActive : determine the capture mode, i.e., Active Mode or InActive Mode;

text : specify the note underneath the image, if it's *NULL* string, there should be no note underneath the image;

pPic :pointer of IPictureDisp to retrieve the image.

Function GetClipboardImage(pPic As IPictureDisp) As Boolean

Attach the image contained in the clipboard to the *IPictureDisp* pointer *pPic*.

Function SaveImageToFile(pPic As IPictureDisp, fileName As String) As Boolean

Save the image attached to the *IPictureDisp* pointer *pPic* to the file named by *filename*. If the extension of the filename is “*jpg*”, then the file will be saved as JPEG file format, otherwise, it's saved as BMP format.

Function SaveWinImgToFile(hwnd As Long, bWinActive As Boolean, text As String, fileName As String) As Boolean

By calling this function, the application can capture the window image and save it to file *filename*. If the call is successful, the *VARIANT_TRUE* (*TRUE* in VB) is returned, otherwise *VARIANT_FALSE* is returned.

hwnd : the window handle of which the window image will be captured;
bWinActive : determine the capture mode, i.e., Active Mode or InActive Mode;
text : specify the note underneath the image, if it's NULL string, there should be no note underneath the image;
fileName :Specify the file name. If the extension of the filename is "jpg", then the file will be saved as JPEG file format, otherwise, it's saved as BMP format.

Function CaptureComponentImage(pUnk As Unknown, text As String, pPic As IPictureDisp) As Boolean

By calling this function, the application can capture the component image and attach it to *IPictureDisp* pointer, i.e., *pPic*, meanwhile the image will be copied to the clipboard. If the call is successful, the *VARIANT_TRUE* (*TRUE* in VB) is returned, otherwise *VARIANT_FALSE* is returned.

pUnk : Component object of which the image will be captured;
text : specify the note underneath the image, if it's NULL string, there should be no note underneath the image;
pPic :pointer of *IPictureDisp* to retrieve the image.

Function VerifyLicenseKey(LicenseKey As String) As Boolean

Only the correct *LicenseKey* is provided, the image can be normally captured and saved. Otherwise the "Dragonfly Automation Software Evaluation" will be displayed at the left-top side of the captured image.