xWinG VBX

Overview

The xWinG vbx custom control provides a way for VB programmers to access the functions of WinG device contexts. The control has a WinG bitmap and several buffers from which images or parts of images can be pasted into the bitmap. There are 6 transition effects to display the bitmap updates slowly.

Hotspots can be created on the bitmap. When the cursor is over a hotspot, it will change to the pointing finger. When a mouse button is depressed over a hotspot, a hotspot event will occur.

The control will have a multimedia timer with a resolution of 1/100 second. The timer event will have a count parameter to indicate how many timer events have occurred and an error parameter which will indicate the number of timer events that have failed. A timer event will fail if it occurs before the last timer event is done being processed.

The xWinG control supports the xSprite control which provides an easy way to create a multiple frame sprite with a backing buffer.

Limitations

All images must be 256 color bmps with the same color table. They may be RLE encoded but they will be expanded in memory when they are loaded. A separate control and palette editor application with source code will be provided to remap 8 and 24 bit images to a common palette.

All xWinG custom controls use the same palette. The palette is created when a control is initialized by loading an 8 bit bmp file. This palette will persist for the life of the application. The colors may be changed with palette animation but the palette will not be reset by loading further image files..

Only one application can use the VBX at a time. The application may have several controls on one or more forms. Other applications will fail to load the control. This is so that an application can assume that its palette gets created and to assure that it gets complete control of animating the palette colors.

The xWinG control is only intended for Visual Basic. Access has a static 256 color palette that is not compatible with the WinG identity palette concept. Visual C++ has an animation library provided on the developer CD which would be a better choice than a custom control.

The xWinG VBX is not friendly to other palette aware controls. Controls that follow the rules and only use system colors should be ok. Other palette aware controls will experience problems with background palettes not being realized. Controls which do not follow the VB design recommendation of only realizing their palette with the VB palette background flag may cause incorrect colors in the xWinG control when they are of a lower zorder.

The VB graphical controls, including the label control, will not work on the xWinG control. Neither will text and drawing functions and properties be provided. Instead, the API declarations and sample code will be provided for obtaining this functionality from GDI. This will include a technique for extracting the font handle from forms or other controls that support the font properties. The GDI functions will operate directly on the WinG bitmap.

Properties

Action

The action property can be set to one of the following values to perform an action similar to a custom method. These actions are explained in a later section.

XW_LOAD	' load the image from a file
XW_ERASE	' erase the rectangle with palindex
XW_RESERVE	' reserve a palette index
XW_SYNC	' match the WinG color table to the palette
XW_WIPE	' start a transition
XW_LOADBUFFER	' load the buffer from a file
XW_CLEARBUFFER	' delete the current buffer
XW_PUTRECT	' copy the rectangle to the image
XW_TRANRECT	' copy the rectangle to the image with transparency
XW_SETBUFFER	' copy the image pixels to the buffer
XW_SIZEFORM	' size the form to match the control
XW_HOTSPOT	' create a hotspot
XW_NOSPOTS	' delete all hotspots
XW_ENABLE	' enable hotspots with the current value
XW_DISABLE	' disable hotspots with the current value
XW_HALFTONE	'creates a WinG halftone palette

Buffer

Integer from 0 to 3

The Buffer property determines which buffer is referred to by the buffer actions and the buffer properties BuffWide and BuffHigh.

BuffLeft BuffTop

Integer greater than or equal to 0.

When pasting a rectangle from a buffer to the image, these properties determine where the source rectangle is located. The coordinates are in pixels.

BuffHigh BuffWide

Integers greater than or equal to 0. Read only at run time.

They indicate the size of the current buffer in pixels. Empty buffers will have a 0 value for these properties.

Color

Long available at run time.

If the **Palindex** property refers to a palette entry that has been reserved, setting this property will cause the color to be animated -- the hardware color register will be updated immediately. Setting this property will change the palette entry. System palette entries cannot be changed and setting this property when Palindex is below 10 or above 245 will generate an error. When the palette has been changed, the WinG software will try to map the colors that it uses to the nearest palette color. This will dramatically slow the screen updates and result in the color animation being ignored when the control is repainted. In order to avoid this, use the XW_SYNC action before the control is redrawn.

Filename

String

This property is used when using the Action property to read a BMP file to either the control image or a buffer.

hDC

Integer handle. Read only at run time.

This is the WinG device context handle. It may be used for API drawing functions. BitBlt will not work with this handle.

Interval

Integer greater than 9.
This is the timer interval in milliseconds. Valid values are 10 to 32768, which correspond to 1/100 to 32.768 seconds. Setting this value while the timer is running will generate an error.

Palette

Integer handle. Read only at run time.
This is the xWinG VBX palette. All controls will have the same value for this property. It may be used to set the palette for AVI playback.

Palindex

Integer from 0 to 255.
The palette index used for the XW_RESERVE and XW_ANIMATE actions.

PixHigh PixWide

Integers greater than or equal to 0.
This is the size of the WinG bitmap in pixels.

RectLeft RectTop RectHigh RectWide

Integers greater than or equal to 0.
These define a selection rectangle. The selection rectangle is used for pasting from a buffer, doing a transition on a selected area, and creating a hotspot.

SpotValue

Integer not equal to 0.
This is the value of hotspots. It is used to create, enable, or disable hotspots. When a hotspot is created it gets this value. A **Hotspot** event will return the value as an indicator of which hotspot was selected. several hotspots can have the same value.

Time

Long. Read only at run time.
This is the system time in milliseconds, the time that Windows has been running. This property is valid to 1/1000 second. It is provided because the normal VB time functions don't have a very high resolution.

Timer

Boolean. Available at run time.
This controls and indicates whether the multimedia timer is running.

WipeFull

Boolean.
When true, the transition will be on the entire image. When false, the transition will be on the rectangle defined by the rectangle properties.

WipeStyle

Enumerated from 0 to 6.
This determines which transition to apply. Values are:

0 - Fast
1 - Left to Right
2 - Right to Left
3 - Center to Sides
4 - Sides to Center
5 - Center to Edges
6 - Edges to Center

WipeTime

Integer from 0 to 32768.
This is the time in milliseconds that the transition will take. The exception is when **WipeStyle** is set to **Fast**. In this case the **WipeTime** property is ignored. The hardware can be a limiting factor when this value is set too low but the control will attempt to compensate by updating wider lines.

Events

Hotspot

SpotValue As Integer, X As Integer, Y As Integer

Occurs when a user pressed the mouse button when the cursor is over a hotspot. SpotValue is the value of the **SpotValue** property when the hotspot was created. X and Y are the position on the hotspot in pixels.

MouseDown MouseUp

X As Integer, Y As Integer, Button As Integer

X and Y are the window coordinates in pixels of the cursor when the mouse button was depressed. Button is equal to 0 unless the shift key is depressed. Either mouse button will trigger these events.

MouseMove

X As Integer, Y As Integer

X and Y are the window coordinates in pixels of the cursor.

Timer

nCalls As Integer, nErrors As integer.

nCalls is the number of times that this event has been fired since the timer was set. nErrors is the number of times that the event failed to fire. This event will fail to fire if the last timer event has not been processed or if the event is being blocked by some other means, a message box for instance.

Actions

XW_LOAD

Reads the image from the **Filename** bmp image. If the palette is not created then create the palette. The palette is created by using the system colors for the first 10 and last 10 colors. The bmp colors from 10 to 245 are used for the middle entries. Images must be mapped to reflect this palette. (See the Paledit application.) This action also sets the control size to match the image size in pixels.

XW_ERASE

Fills the selection rectangle with the **Palindex** palette color. The rectangle in not drawn in the control.

XW_RESERVE

Reserve the **Palindex** palette index. This value must be between 10 and 245. The system colors may not be reserved. Palette indexes may not be unreserved.

XW_SYNC

When palette animation occurs by setting the Color property, the WinG device will still retain the old colors. If the WinG bitmap is redrawn then the device will try to match the old colors as best it can. This will result in a much slower screen update and loss of the animated colors. This action will rematch the WinG colors to the new palette.

XW_WIPE

Perform a transition on the full control or on the selected rectangle, depending on the value of **WipeFull**. The transition will take **WipeTime** milliseconds unless **WipeStyle** is set to fast or zero.

XW_LOADBUFFER

The current buffer is loaded from a bmp file. No color information is kept in the buffer. The BuffWide and BuffHigh properties will reflect the size of the buffer.

XW_CLEARBUFFER

Deletes the contents of a buffer. The **BuffWide** and **BuffHigh** properties will be equal to zero.

XW_ PASTERECT XW_TRANSRECT

Paste a rectangle from the current buffer to the WinG bitmap. The rectLeft, rectTop, rectWide, and rectHigh properties will determine the destination on the bitmap. The BuffTop and BuffLeft properties will determine which part of the buffer will be pasted. Both rectangles must be contained in the images. XW_PASTERECT will copy the entire rectangle. XW_TRANSRECT will not copy pixels that match the first pixel in the buffer. The first pixel is the lower left pixel in a bmp file. If the lower left pixel must be copied with XW_TRNASRECT, it is suggested that one additional line be added to the image in order to define the transparent pixel index. The system palette indexes make good candidates for transparent indexes, since they are not copied, it makes no difference that the colors are determined at run time.

XW_SETBUFFER

Sets the buffer image equal to the bitmap image. This will create a backup image to restore the bitmap after pasting to it, etc.

XW_SIZEFORM

Size the form to match the control. Used to make sure the control exactly covers the form. Not done.

XW_HOTSPOT

Create a hotspot.

XW_NOSPOTS

Delete all hotspots.

XW_ENABLE

Enable all of the hotspots with **SpotValue**.

XW_DISABLE

Disable all the hotspots with **SpotValue**.