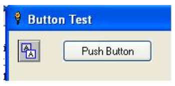
BmpButtons

-
Alyce

What are BmpButtons?

BmpButtons are buttons that display an image instead of text. The button on the left is a BmpButton. The button on the right is a regular push button.



How Do BmpButtons Work?

Bmpbuttons work a little differently than regular buttons. Instead of a caption, we must specify a filename for the bitmap that will appear on the button. This filename can have a relative path. If we run a program in the Liberty BASIC directory, it can find the bmps in the bmp directory if we specify the path like this:

```
bmpbutton #1.b, "bmp\copy.bmp",[hi],UL,10,10
```

The bmpbutton command also includes the name of the branch label or sub that will be the event handler for the button, the corner to which placement will be relative, and the x and y placement coordinates. We can't specify a width and height as we can with regular push buttons, because Liberty BASIC checks the size of the bitmap and makes the button that same size.

BmpButton Commands

Bmpbuttons understand a limited command set.

- print #handle.ext, "bitmap bitmapname"
- print #handle.ext, "locate x y width height"
- print #handle.ext, "setfocus"
- print #handle.ext, "enable"

- print #handle.ext, "disable"
- print #handle.ext, "show"
- print #handle.ext, "hide"

SETTING FOCUS TO A BMPBUTTON

Commands sent to a bmpbutton should not begin with a "!" character like commands sent to a regular button. We cannot change the text on a bmpbutton, because there IS no text on a bmpbutton. We can cause a bmpbutton to receive the input focus so that any keypresses will be directed to the bmpbutton. Here is the syntax:

```
print #handle.ext, "setfocus"
```

CHANGING THE BITMAP ON THE BUTTON

We can set the bitmap of the control to be the named bitmap that has been loaded previously with the LOADBMP command (not the filename of the bitmap).

```
loadbmp "bitmapname", "filename.bmp"
print #handle.ext, "bitmap bitmapname"
```

If the new bitmap is not the same dimensions as the original bitmap, it won't look right, because the bmpbutton will not automatically change size to match the new bitmap. You can find the dimensions of a bitmap by loading it into MS Paint and checking its attributes. You can also find the dimensions of a loaded bitmap at runtime by opening the file and reading the info, or via API calls.

MOVING AND RESIZING A BMPBUTTON

You can reposition the bmpbutton within the window that contains it. Use the "locate" command. This command requires the new x and y coordinates for the bmpbutton, plus the width and height desired. You must issue a "refresh" command to the window to cause it to update the display to show the new location of the button. To use the "locate" command, you must know the desired dimensions of the bmpbutton. See Issue #100 for a way to discover bitmap dimensions at runtime. Here is the syntax:

```
print #handle.ext, "locate x y width height"
print #handle, "refresh"
```

Remember that values inside the quotes are hard-coded. To use variables, place them outside of the quote marks and preserve the blank spaces inside the quote marks.

```
print #handle.ext, "locate 12 45 25 25"
or
x=12 : y=45 : width=25 : height=25
print #handle.ext, "locate ";x;" ";y;" ";width;" ";height
```

The "locate" command gives us the ability to specify a width and height for our bmpbuttons, so we are not limited to accepting the size given them by Liberty BASIC. We can change the look of our buttons this way, making them smaller if the window is resized by the user to be very small, for instance. We can also use the "bitmap" command to place a new image on the button, and then use the "locate" command to change the size of the button to match the size of the new bitmap.

DEMO

Here is a small demo that illustrates some of the possibilities of using the "bitmap" and "locate" commands with bmpbuttons.

```
'run this code from within your Liberty BASIC directory
'so that the bmps can be found by the program
nomainwin
WindowWidth=600:WindowHeight=400
bmpbutton #1.b, "bmp\copy.bmp",[hi],UL,10,10
button #1.move, "Move Button",[moveButton],UL,10,80,140,24
button #1.new, "New Bitmap",[new],UL,10,120,140,24
open "Move Bmpbutton" for window as #1
#1 "trapclose [quit]"
wait

[quit]
'if bmp was loaded, unload it
if pianoLoaded then unloadbmp "piano"
close #1:end

[moveButton]
'move bmpbutton to new coords

if not(pianoLoaded) then
```

```
'actual size of copy bmp is
'25x25, and we will preserve that:
#1.b "locate 200 10 25 25"
else
'if piano bmp is in use, use
'these dimensions:
#1.b, "locate 200 10 300 50"
end if

#1 "refresh"
wait


[new]
'change bmp and reposition
loadbmp "piano","bmp\piano6.bmp"
pianoLoaded=1   'flag that bmp was loaded
'actual size of piano.bmp is 600x100
'but we will make it 300x50 here:
#1.b "bitmap piano"
#1.b "locate 50 10 300 50"
#1 "refresh"
wait

[hi]
notice "Hi"
wait
```

BmpButtons | What are BmpButtons? | How Do BmpButtons Work? | BmpButton Commands | SETTING FOCUS TO A BMPBUTTON | CHANGING THE BITMAP ON THE BUTTON | MOVING AND RESIZING A BMPBUTTON | DEMO