
Subject: Re: cursor_image and byteorder

Posted by [Karl Schultz](#) on Wed, 10 Sep 2003 14:47:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

"Chad Bender" <cbender@mail.astro.sunysb.edu> wrote in message

news:pan.2003.09.09.19.28.53.322293.300@mail.astro.sunysb.edu...

> I recently upgraded my linux box and IDL distribution from a PII w/ RH
> 6.something and IDL 5.4 to a P4 w/ RH 7.3 and IDL 5.6. I have a routine
> that changes the cursor shape, and worked fine under my old setup. On the
> new machine, the cursors are swapped left to right, and I clearly don't
> understand enough of what's going on to figure out why.

>

> Here is exactly what I'm doing:

>

> On the old box with IDL 5.4,

> { x86 linux unix 5.4 Sep 25 2000 32 32}

>

> curs=BYTARR(16,16)

> curs[0:3,0:15]=1

> curs[4:6,2:13]=1

> curs[9:11,2:13]=1

> curs[12:15,0:15]=1

> curs[6,7:8]=0

> curs[9,7:8]=0

> m=BYTE(NOT FLOAT(curs))

> power=2L^INDGEN(16)

> cursor=curs#power

> mask=m#power

> DEVICE, Cursor_Image=cursor, Cursor_Mask=mask, Cursor_XY=[8,8]

>

> This turns my cursor into a sideways I-beam (or a tie fighter).

>

> Now on the new box:

> { x86 linux unix linux 5.6 Oct 26 2002 32 64}

>

> I run the above code and the right and left sides of the cursor are

> swapped. However, if I add the following lines before the device command,

> then everything is fine.

> BYTEORDER, cursor

> BYTEORDER, mask

>

> That's all well and good. I can now see my cursors correctly. However,

> two questions remain:

>

> 1) Why did this happen? I could understand the bytes being swapped if I

> was literally taking an array from one machine to another. But as the

> calculations are done individually on each machine, I'd think things would

> have the correct order. (this is probably just my lack of understanding).

>
> 2) What can I do about it in my program? It's all well and good that I
> can get this to work on one machine or another. But this code will
> probably end up being distributed to others who will be running a variety
> of different versions on different architectures. How can I make the
> program smart so that it knows when the swapping is needed?
>
> Thanks
> Chad Bender
> Dept. Physics and Astronomy
> SUNY Stony Brook

IDL 5.5 contained a release note on this topic. Hope this helps:

CURSOR_IMAGE and CURSOR_MASK keywords to DEVICE
IMAGE and MASK keywords to IDLgrWindow::SetCurrentCursor

On Unix little-endian platforms (Intel based), the two bytes within each integer were being used in the wrong order, resulting in cursors that were split in half. Now, on all platforms, the high-order byte gives the left half of the cursor, while the low-order byte gives the right half.

Example:

Given a 16x16 byte array containing a 0 or 1 bitmap pattern, you can construct a new

cursor by entering the following commands:

```
; Assume variable bitmap contains a 16x16 byte array.  
bytes = CVTTOBM(bitmap)  
; Left half becomes the high order, right becomes low order.  
image = 256*bytes[0,*] + bytes[1,*]  
; Direct graphics:  
DEVICE, CURSOR_IMAGE=image  
; Object graphics:  
oWindow->SetCurrentCursor, IMAGE=image
```

The above commands should now work on all platforms.
