

Re: CPaintDC class

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.language/2006-10/msg00228.html>

- *From:* "John Carson" <jcarson_n o sp am @xxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 4 Oct 2006 22:20:41 +1000
-

"Alex Blekhman" <xfkt@xxxxxxxx> wrote in message
<news:OcBU4d65GHA.1188@xxxxxxxxxxxxxxxxxxxxxxxx>

NickP wrote:

Hi there,

I have some form drawing code that uses the CPaintDC class and it works fine. But I am just wondering if it is as good as using BeginPaint and EndPaint.

I know this class calls these 2 methods anyway, but I am unsure as to whether it takes in for account the invalid rectangle whilst calling it's BitBlt method.

For example, if I declare "CPaintDC dc(myWindowHandle)" and the paint event args specify that only a small portion of the entire surface is to be repainted does BitBlt only paint said region?

Let's say only the rectangle (100,100,150,150) needs repainting.... -----

```
CPaintDC dc(m_hWnd);
CRect rc; GetClientRect(&rc);

//Clear the buffer
m_memDC.FillRect(&rc,m_bkgrnd);

//Draw to the buffer here
...

//Blit the buffer
dc.BitBlt(rc.left,
rc.top,
rc.right,
rc.bottom,
m_memDC,
```

Re: CPaintDC class

```
rc.left,  
rc.top,  
SRCCOPY);
```

Does BitBlt actual perform a block copy of the entire rectangle specified, or only the one that is specified in the paint arguments?

BitBlt is very stupid. It just copies bits from source DC to destination DC. You can use CPaintDC::m_ps member to figure out exact paint rectangle.

Stupid or not, BitBlt will not paint outside the invalid region in a WM_PAINT call. There may be some performance difference when you use the PAINTSTRUCT variable to clip painting yourself, but I have no information on that.

--
John Carson

.