

Re: GDI does not provide all GLYPH handles for big font size

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2005-11/msg00228.html>

- *From:* "S. Mark Courter" <smcourter@xxxxxxxxxxxxx>
 - *Date:* Wed, 9 Nov 2005 10:55:22 -0600
-

"fussa",

Here is a more complete code snippet that works for larger fonts.
By the way, if the font gets even bigger (say 150 pts), DrvTextOut() is no longer called by the GDI. Instead, DrvFillPath() is called.

I am currently trying to solve this problem, since I get no font output in this case. Any help from anyone out there would be welcome.;

Mark

```
BOOL APIENTRY DrvTextOut(  
SURF_OBJ *pso,  
STROBJ *pstro,  
FONTOBJ *pfo,  
CLIPOBJ *pco,  
RECTL *prclExtra,  
RECTL *prclOpaque,  
BRUSHOBJ *pboFore,  
BRUSHOBJ *pboOpaque,  
POINTL *pptlBrushOrg,  
MIX mix  
)
```

.....

```
//Coming into DrvTextOut(), pstro->pgp can be NULL for large Fonts  
//So you need to do this  
if (pstro->pgp) // got the GLYPHPOS array, it's already all there  
{  
    pgpos = pstro->pgp;  
    MoreGlyphs = FALSE;  
    cGlyphs = pstro->cGlyphs;  
    VERBOSE(("--DrvTextOut: pstro->pgp is NOT NULL."));  
}  
else // Need to do more work to get the GLYPHS
```

Re: GDI does not provide all GLYPH handles for big font size

```
{
STROBJ_vEnumStart(pstro);
MoreGlyphs = TRUE;
VERBOSE(("--DrvTextOut: pstro->pgp is NULL, using new code"));
}

.....

//
// Now start drawing the glyphs, if we have MoreGlyphs = TRUE then we
// will do a STROBJ_bEnum first, in order to load up the Glyph data.
do
{
//
// We need to enum for more glyph data so do it now.
//

if (MoreGlyphs)
{
MoreGlyphs = STROBJ_bEnum(pstro, &cGlyphs, &pgpos);

if (MoreGlyphs == DDI_ERROR)
{
VERBOSE(("DrvTextOut: STROBJ_bEnum()=DDI_ERROR"));
return(FALSE);
}
}

// loop thru array of glyph info
for( gpInd = 0; gpInd < cGlyphs; gpInd++, pgpos++ )
{
GLYPHDATA gd;
GLYPHDATA *pgd; // pointer to glyph data

VERBOSE(("--DrvTextOut: Doing Glyph # %d", gpInd));

// If GCAPS_HIGHRESTEXT is set in DEVINFO, offset is actually
// in FIX coordinates, and needs to be converted
offset.x = pgpos->ptl.x;
offset.y = pgpos->ptl.y;

if (pfo)
{
//
// This is true type font, so query the bitmap
//
pgd = &gd;

count = FONTOBJ_cGetGlyphs( pfo, FO_GLYPHBITS, 1, &pgpos->hg,
(LPVOID)&pgd );
}
```

Re: GDI does not provide all GLYPH handles for big font size

Re: GDI does not provide all GLYPH handles for big font size

```
if (count == 1)
{
    pgb = pgd->gdf.pgb;
}
else
{
    count = 1;
    pgb = pgpos->pgdf->pgb; // For bitmap font, we already have the bitmap
}

if (count == 1)
{
    int x, y;
    int pos;
    int bitpos;
    BYTE set;
    COLOR24 *pPixel;
    int bitMask; // for checking pixels in 1 bpp glyph bitmap

    //Keep running total of boundary of text string
    bound.top = min( bound.top, offset.y + pgb->ptlOrigin.y + 1 );
    bound.left = min( bound.left, offset.x + pgb->ptlOrigin.x );
    bound.bottom = max( bound.bottom, offset.y + pgb->ptlOrigin.y + 1 +
    pgb->szlBitmap.cy );
    bound.right = max( bound.right, offset.x + pgb->ptlOrigin.x +
    pgb->szlBitmap.cx );

    //Can Process pixels in raster font if desired
    for( y = 0; y < pgb->szlBitmap.cy; y++ )
    {
        pPixel = (COLOR24*)((BYTE*)pso->pvScan0 + // the scan line
        ( y + offset.y + pgb->ptlOrigin.y + 1 ) * pso->lDelta);

        x = offset.x + pgb->ptlOrigin.x; // surface position

        for( bitpos = 0; bitpos < pgb->szlBitmap.cx; bitpos++ )
        {
            bitMask = 1 << ( 0x000007 ^ ( bitpos & 0x000007 ));

            if(((BYTE)pgb->aj[pos]) & bitMask)
            {
                // copy first 3 (lower order) bytes to the surface pixel
                memcpy( &pPixel[x], &color, 3 );
            }
            x++; // next pixel on surface

            if( bitMask == 1 ) // last bit of this byte
            {
                pos++;
            }
        }
    }
}
```

Re: GDI does not provide all GLYPH handles for big font size

```
}  
  
if( bitMask != 1 ) // if I didn't just move to next byte...  
{  
pos++; // do it now, b/c rest of this byte is padding  
}  
  
}  
} // printing raster glyph  
else  
{  
VERBOSE(("-- FONTOBJ_cGetGlyphs (bitmap) FAILED"));  
return(FALSE);  
}  
  
} // loop through glyphs  
  
} while (MoreGlyphs);
```

"fussa" <mailbox.AF@xxxxxxx> wrote in message
news:%23jfBrpJ5FHA.3188@xxxxxxxxxxxxxxxxxxxxxxxxxxxx
> Hello,
>
> I have a problem to obtain all glyphs to a STROBJ if the fontsize is to
> big, for instance size 96 in Word. I only get the first glyph. If I take
> a smaller font size like 12 all handles to glyphs can be accessed.
>
> Can someone of you help me? It would be very pleasefull.
>
> I try to obtain it by using the following code:
>
> STROBJ_vEnumStart(pstro);
> do {
> bMore = STROBJ_bEnum(pstro, &cGlyphs, &pgp);
> for (i=0 ; (UINT)i < cGlyphs; i++) {
> ...
> }
> } where (bMore);

• *Follow-Ups:*

- ◆ *Re: GDI does not provide all GLYPH handles for big font size*
◇ From: fussa

• *References:*

- ◆ *GDI does not provide all GLYPH handles for big font size*

Re: GDI does not provide all GLYPH handles for big font size

Re: GDI does not provide all GLYPH handles for big font size

◇ *From:* fussa

- Prev by Date: ***Re: What's wrong with IOCTL SERIAL WAIT ON MASK?***
- Next by Date: ***Re: accessing kernel driver from non admin user account***
- Previous by thread: ***GDI does not provide all GLYPH handles for big font size***
- Next by thread: ***Re: GDI does not provide all GLYPH handles for big font size***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***