Subject: Re: Will UPP support full UNICODE (21bits long codepoint)?
Posted by mirek on Sat, 22 Aug 2020 16:01:03 GMT
View Forum Message <> Reply to Message

I am now investigating the whole thing in Win32. Let us say we would want just to support 32-bit codepoints as the first step.

Now current model needs following info: a) that the font has glyph for required, b) at least advance width for the glyph.

If we do not want to use Uniscribe, just good old GDI, so far I have only found solution to b), but unfortunately it returns just dimensions of that "box" character that is used in case glyph is missing instead of indicating anyhow that it is actually missing....

```
void Test(int ch, Font fnt)
{
 TIMING("Glyph");
 HFONT hfont = GetWin32Font(fnt, 0);
 VERIFY(hfont);
 if(hfont) {
  HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
  HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
  GLYPHMETRICS gm;
  memset(&gm, 0, sizeof(gm));
  MAT2 m_matrix;
  memset8(&m_matrix, 0, sizeof(m_matrix));
  m_matrix.eM11.value = 1;
  m_matrix.eM22.value = 1;
  int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED|GGO_METRICS, &gm,
0, NULL, &m_matrix);
  if(gsz == GDI_ERROR)
   DLOG("Failed " << ch);
  if(gm.gmCellIncX != 75)
   DLOG(ch << " " << gm.gmCellIncX << ", " << gm.gmCellIncY << ", " << gm.gmptGlyphOrigin.x);
  ::SelectObject(hdc, ohfont);
  ::DeleteDC(hdc);
 }
}

GUI_APP_MAIN
{
 for(int i = 32; i < 100000; i++)
  Test(i, Font().Height(100).FaceName("MingLiU-ExtB"));
}
```