Subject: Re: 16 bits wchar
Posted by mirek on Mon, 04 Aug 2008 15:14:48 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 04 August 2008 09:53
Using multiple code units per character doesn't disable the use of a text editor or any means of manipulating Unicode texts. It just needs a little bit smarter methods for some operations. I know that using only one word is convenient, but Unicode says that there are up to two words per codepoint and there is no other work around than using 32 bits, which is not a lot better, because not even with UTF32 there isn't a 1:1 relationship between character and display operation of that character. Nonwhitespaces, separators, control characters, combining characters and others must be filtered out, and the end result is the same as if you would use 16 bit chars (where the same operations must be done and I don't think they are done right now).

Well, this rather sound like we should kick out WString altogether and keep just UTF-8:)

Quote:
This way there is no need for WString actually, except the fact that it helps as an optimization because Win32 uses it. In the end, we will probably need a full text layout engine, breaking text in multiple segments, and drawing them one by one to support composition, multichar composition, RTL.

Ah, right

OTOH, on logical level, I still see characters on the screen. And those characters should be edited on per-character basis.

Maybe we just need smarter encoding than UNICODE?

Makes me think - realistically, there is a lot of "reserved" positions in BMP. Could we just use them for this?

Mirek