Subject: Re: Will UPP support full UNICODE (21bits long codepoint)?
Posted by mirek on Mon, 17 Aug 2020 12:04:20 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Mon, 17 August 2020 11:50Quote:It looks like most toolkits simply use HarfBuzz anyway...

Well, this seems to be the best option but I was even afraid of suggesting it, as it means another dependency (and possibly a lot of work) :)

By the way, If you think it's ok, In the meantime we can have better precomposition support. I've attached Charset.cpp with the patched UnicodeCombine for full precompositions support (for 16-bit UCS canonicals only).

(I can also send the extractor code for uppbox if needed)

Best regards,
Oblivion

I am sorry to say that because it is mostly due to lack of docs, but I think all this is already better covered with

int UnicodeDecompose(dword codepoint, dword t[MAX_DECOMPOSED], bool only_canonical);
Vector<dword> UnicodeDecompose(dword codepoint, bool only_canonical);

- the reason why this was not quite documented is that above functions are sort of abandoned effort in previous attempt at better Unicode support. Anyway, they are using quite effective z-compressed table (as not to increase .exe size too much). This (and other) tables are producced directly from Unicode tables by uppbox/unicode. And they should also support more than 1 combining marks...