Subject: Re: 16 bits wchar
Posted by cbpporter on Tue, 05 Aug 2008 13:57:09 GMT
View Forum Message <> Reply to Message

luzr wrote on Tue, 05 August 2008 16:12
Well, I was studying this as well and came to conclusion that combining is of little concern.

First, AFAIK, basic Unicode "compliance" does not requite it.

Second, all important ("real") combining codepoints have characters in Unicode.

Sure, it does not require it, but is relatively easy to implement. I have a pretty clear idea on how to do it. But you are right, it's not a priority right now.

Quote:
This is because there is no general way how to create combined glyph....

Compute size of base character, retrieve align of character that is combined with, align in a rect that has the size as a maximum of both and draw. Basically in pseudocode:

```
draw(curx, cury, basechar);
draw(curx + deltax, cury + deltay, combinedchar);
```

Finding out delta is not that easy, but doable. This is pretty much what Qt (empirically determined) does and is near perfect.

Quote:
Does not make sense to me...

Was doesn't make sense. Basically what I said is that you cant feed composed characters to an editable control. And expect align and keyboard/mouse navigation to work.

Quote:
IMO, this would be pretty hard to address in fact. Or result in confusing user interface.

I don't understand why this affects user interface. Everything looks the same from the point of view of the user.

Quote:
Actually, we do not. Interface accepts strings. But I doubt it manages combining.

You are doing:

```
for(int i = 0; i < n; i++) {
 wchar h = text[i];
 XftDrawString16(...,(FcChar16 *)&h, 1);
```

That's drawing characters one at a time (if angle is not zero). If a composition system is available, in this case it will not trigger because drawing characters that are used for composition one at a

time doesn't make any sense.
But if angle is zero, you are right. I guess that it does not do composition.

The offline scan seems like a good idea.