

---

Subject: Re: Choosing the best way to go full UNICODE

Posted by [mirek](#) on Wed, 31 May 2017 10:07:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

cbpporter wrote on Wed, 31 May 2017 11:30mirek wrote on Wed, 31 May 2017 12:00

If you can iterate linearly, Unicode is indexable...

Sorry, you don't get it. I'm using a very specific meaning of indexable, probably that is the problem.

A vector is indexable. You can reach `v[7]` without going through 0 to 6 and 0 through 6 can't do anything to change the "offset" of 7. A list is not. You need to traverse it to get `list[7]`.

I guess with "indexable" I was not specific enough.

Imagine that you traverse through Utf8 String and each codepoint (or grapheme cluster) you store in individual cell of

`Vector<String>`

Then the result is definitely indexable. Or am I missing something? (Of course, I would not use `Vector<String>` for optimized version, but that is just implementation issue).

mirek wrote on Wed, 31 May 2017 12:00

That sound to me like you trying to make String indexable. Which can't be done with Unicode. How are you going to determine the sequence of bytes that you must return for a position without traversing it linearly?

Traverse it and store positions. (It is functional equivalent to splitting to `Vector<String>` as described above).

mirek wrote on Wed, 31 May 2017 12:00

Not so sure about this - not that important IMO at this point. So I will not get correct `ToUpper` for many characters - that has little impact on most applications.

Half measures again...

[/quote]

I do not mean that it is not important at all. Just that choosing proper basic interfaces is more pressing at the moment.

---