

uNstring Arguments Shall Be UTF-N Encoded

Document #: P1880R0
Date: 2019-08-23
Project: Programming Language C++
SG16
LEWG
Reply-to: Zach Laine
<whatwasthataddress@gmail.com>

1 u8string, u16string, and u32string Don't Guarantee UTF Encoding

When SG16 is consulted about which string overloads should be provided for an interface, we will sometimes recommend providing UTF-encoded overloads. Those overloads are specified in terms of C++20's `u8string`, `u16string`, and `u32string` (though not all of them are usually used in the same interface).

When we give this recommendation, it always comes with the advice, “Those types don't guarantee encoding, so don't forget to add an *Expects*: that indicates that such arguments must be.”

Since this is probably universal (no exceptions have come up so far), it seems like something we should add to [res.on.arguments] in the library introduction.

2 Wording

Append this to the bulleted list in Function arguments [res.on.arguments]:

- A function argument whose type is (possibly cv-qualified) `u8string`, `u16string`, or `u32string` ([string.classes]), or a reference to one of those types, shall be respectively UTF-8, UTF-16, or UTF-32 encoded.