Subject: anybody tried re-creating u++ for -std=c++20
Posted by piotr5 on Mon, 27 Feb 2023 22:16:04 GMT
View Forum Message <> Reply to Message

for now the only (obvious) problem is the warnings about [=] use being deprecated for handing over this. so I really have no problem with using current u++ and it is backwards compatible to all the old programs and compilers. but how about a new version that breaks compatibility and uses features like Modules and concepts like Moveable natively?

Subject: Re: anybody tried re-creating u++ for -std=c++20
Posted by mirek on Tue, 18 Apr 2023 11:29:30 GMT
View Forum Message <> Reply to Message

piotr5 wrote on Mon, 27 February 2023 23:16for now the only (obvious) problem is the warnings about [=] use being deprecated for handing over this. so I really have no problem with using current u++ and it is backwards compatible to all the old programs and compilers. but how about a new version that breaks compatibility and uses features like Modules and concepts like Moveable natively?

Modules are completely useless.

Concepts are not worth breaking the compatibility.

If you want to use c++20, go for it. We were discussing [this] in the past, every single workaround is worse than just disabling the damned warning (or just ignoring it)...

Mirek