
Subject: Re: Gearing up for 2022.1 release...
Posted by [mr_ped](#) on Fri, 08 Apr 2022 10:42:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

IMHO:

- upp packages "should" compile also in C++20 mode, so user apps can use the C++20 (would be really nice to have this fixed before 2022.1 release)
- changing minimal requirement from C++14 to C++17 without using it is useless, just adding artificial hurdle (some users may be forced to C++14 in their projects)
- refactoring the code to use C++17 just before 2022.1 release just for the sake of using C++17 is also weird use of dev-time

My understanding is that U++ is almost ready for release, and unless there is some practical value in using C++17 for remaining tasks, it is IMHO much better to release it as C++14 compatible (but ideally with C++17 and C++20 compatibility too), that enables most options for users.

And things like bumping minimal version should be rather done at beginning of the release (ideally when some new features also make sense and will be used actively), not in last minute before release.
