Subject: U++ Allocator causing crashes (double-free/wrong size) with third party libraries
Posted by jjacksonRIAB on Mon, 26 Jun 2023 20:25:48 GMT

View Forum Message <> Reply to Message

Lately I've been finding myself increasingly having to work with third party closed source or complex build apps and I have to turn on USEMALLOC frequently just to be able to work with them because I get segfaults on delete, double-frees, etc all due to the U++ allocator. Disabling checks doesn't fix these problems.

I'm wondering whether it would just be a better option to do what EASTL does and modify NTL so that a custom allocator can be passed in instead of overriding the normal allocator.I'd like to be able to use the U++ allocator where it's still appropriate so that I can keep alignment and other performance improvements within NTL without having to turn everything off because of one C library that wants to do non-standard things with memory.

Is this a possibility? Is there some other way to do it?