Subject: Re: How to mark std::array<T, N> moveable if only T is moveable
Posted by mirek on Thu, 18 Jul 2024 06:44:32 GMT
View Forum Message <> Reply to Message

busiek wrote on Tue, 02 July 2024 17:54Hi all,
U++ does not have a container with std::array functionality. Is there any simple way to mark std::array<T, N> moveable if only T is moveable?
I didn't come with anything better than implementing my own version of std::array derived from Moveable<MyArray<T, N>>. By using macro NTL_MOVEABLE one can only mark moveable a specific type not a template. Even if I want to mark specific template type, macro does not work with template types having two or more parameters and I have to expand macro myself:
template<> inline void AssertMoveable<std::array<double, 2>>(std::array<double, 2>*) {}

Correct me if I miss something. Shouldn't the mechanism of marking type moveable be done through partial class instantiation? Or concepts with C++20?

Wait a moment there: Are you sure that std::array is Moveable? There is no guarantee...

Mirek