Subject: Re: How to mark std::array<T, N> moveable if only T is moveable
Posted by busiek on Thu, 25 Jul 2024 15:32:03 GMT
View Forum Message <> Reply to Message

mirek wrote on Thu, 25 July 2024 09:32mirek wrote on Thu, 25 July 2024 09:19
Another thing to consider: Tuple is moveable if all its components are moveable. Are we able to express that somehow? (If not, no big deal, I can make Tuple moveable and request that elements are).

Figured it out:

```
struct not_moveable {};

template <class A, class B>
class Two : std::conditional<is_moveable<A> && is_moveable<B>, moveable<Two<A, B>>,
not_moveable>::type {
 A a;
 B b;
};
```

But what if one wants to use std::tuple instead of Upp::Tuple? How to mark it moveable if tuple components are moveable?
I sometimes prefer to use std::tuple because one can write:
auto [a, b] = some std::tuple
Alternatively, there is a way to implement Upp::Tuple that it works with structured binding?