Subject: Re: Question about SubRange.
Posted by Novo on Mon, 08 Jan 2018 14:55:01 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 08 January 2018 05:52
Not true (not in U++ nor STL). That is basically a reason for those 2 overloads. E.g. try Sort with just const variant...

The "C& c" variant is definitely not needed. The "C&& c" variant is an universal reference and it is a complete replacement of the "C& c" variant. I tried to compile TheIDE with "C& c" commented out and it compiles just fine.

The problem with the "const C& c" variant is that when you try to construct the SubRangeClass you get a SubRangeClass<const I> version.
I tried to strip the constness like below

```
template <class I>
SubRangeClass<I> SubRange(I l, I h)
{
 return SubRangeClass<typename std::remove_const<I>::type>(l, h);
}

template <class I>
SubRangeClass<I> SubRange(I l, int count)
{
 return SubRangeClass<typename std::remove_const<I>::type>(l, count);
}
```

But I'm still getting weird compilation problems with TabBar::Tab.

And I'm still a little bit confused about what is IterSwap for. What is wrong with the regular Swap?

Something is not right about all this ...