Subject: Re: Vector<T>::Set(int i, T&& x) proposal
Posted by mirek on Wed, 31 Jan 2018 18:34:54 GMT
View Forum Message <> Reply to Message

Novo wrote on Tue, 30 January 2018 03:30mirek wrote on Sun, 28 January 2018 16:47I am considering this, but single element Set was always just v.At(i) = pick(src);

Well, yes. This will work as well. But this is significantly less intuitive. Your way of solving this problem definitely didn't come to my mind when I was looking for a solution. Adding of an rvalue-based version of Set definitely won't break API because you already have a reference-based version. This is just another performance optimization.

My solution is based on move-constructor and yours is based on move-operator.

OK. After further thinking, for some time now I think it is worthwhile to add

const T& Vector::Get(int i, const T& default) { return i >= 0 && i < GetCount() ? Get(i) : default; }

Set makes a nice complement to this, so it makes sense to add Set(int, T&&) too.

Do you see any problem with trivial implementation

void Vector::Set(int i, T&& x) { At(i) = pick(x); }

?

BTW, this is simple. But if I am about to add it here, I have to do that for every container where it makes sense and fill documentation too, add autotests. Thats just to explain my hesitation...