
Subject: Vector<T>::Set(int i, T&& x) proposal
Posted by [Novo](#) on Tue, 19 Dec 2017 22:30:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

I propose to add method below to the Vector class.

```
template <class T>
T& Vector<T>::Set(int i, T&& x) {
    ASSERT(i >= 0);
    const int count = GetCount();
    if (i == count)
        return Add(pick(x));
    else if (i > count) {
        At(i - 1);
        return Add(pick(x));
    }
}
```

```
T* addr = vector + i;
addr->~T();
::new(addr) T(pick(x));
return *addr;
}
```

Or it can be implemented like this:

```
template <class T>
T& Vector<T>::Set(int i, T&& x) {
    ASSERT(i >= 0);
    At(i);
    T* addr = vector + i;
    addr->~T();
    ::new(addr) T(pick(x));
    return *addr;
}
```

Second implementation has less memory allocation stuff. It is hard to tell which one is better. I believe there is no need to check that x is already contained in Vector because it is impossible to get an rvalue of a Vectors's element.
