Subject: Re: Proposal: add Vector::InsertPick(int i, pick_ T&)
Posted by Mindtraveller on Fri, 08 May 2009 22:05:54 GMT
View Forum Message <> Reply to Message

luzr wrote on Fri, 08 May 2009 23:21Mindtraveller wrote on Fri, 08 May 2009 04:30Actually it doesn`t. Differences are:
- you require M to have default constructor, which is not strong requirement for Vector element

1. Does it?

Quote:
- you have one redundant call (ctor and operator= instead of ctor), which sometimes is crucial (picking was made for efficiency, right?)

And IMO this code is little bit less clear than plain v.InsertPick(a).

2. While I can agree with symmetry point and clarity issue, I do not see how adding InsertPick should solve above problem.

3. Maybe you can rewrite mrjt's example with InsertPick?

Mirek
1. I`m not quite shure I understand what you mean by this question.
If I understand correctly and the question is about default ctor requirement, TheIDE help says:
Quote: General requirement: T is required to be moveable and must have either deep copy constructor, pick constructor or default constructor.

Adds new element to Vector and picks value of parameter to it.
Requires T to have pick constructor.
-- so it is not necessary for M to have default /deep copy ctor to have Vector<M> and do AddPick.

2. I hope that InsertPick won`t make me writing default constructors or optional deep copy to insert element into queue. This should solve the problem with deep copy where I want to pick only.

3.Vector<M> v;
M a;
..
v.InsertPick(a);
Or maybe you mean InsertPick(int i, pick_ Vector<M> &) ?