
Subject: Understanding pick behaviour and containers
Posted by [Mindtraveller](#) on Thu, 01 May 2008 19:09:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Let`s say, one needs to have hashed container of some complex objects. OK, taking Index and putting some moveable and optional deep copy flavour to objects contained:

```
class AAA : public VectorMap<int, int>, public MoveableAndDeepCopyOption<AAA>
{
public:
    AAA() {}
    AAA(const AAA &, int) {}
    unsigned GetHashValue() const {return 0; /*proxy here*/;}
};
```

```
CONSOLE_APP_MAIN
{
    AAA a;
    a.Add(10,1);
    AAA a2(a);
    Cout() << a2.GetCount();

    Index<AAA> ai;
    ai.Add(a2);
    Cout() << " / " << ai[0].GetCount();
}
```

Output gives us 1 / 0. First value is OK, since first pick operation succeeded. Second one is zero, which is not right IMO: I added object to Index and it didn`t appear in the container. When we talk about Vector or even VectorMap - it`s OK, we have AddPick() there, which works in this case.

So, why add operation didn`t work and how to solve this?

P.S. Why (pick_ != const) for MSC compiler?
