Subject: Array : swapping element with a pointer
Posted by mdelfede on Sat, 10 Sep 2011 19:30:38 GMT
View Forum Message <> Reply to Message

Hi,

I'd need something like this :

```
Array[A] array;

A * a = new A;

array.Swap(someIndex, a);
```

where Array::Swap() should be so :

```
void Array<A>::Swap(int idx, A*& elem);
```

This would allow swapping contents of single elements of different arrays without insertion/deletion steps; example :

```
Array<A> arr1, arr2;

arr1.Add(new A);
arr1.Add(new A);

arr2.Add(new A);
arr2.Add(new A);

A *a = NULL;
arr1.Swap(1, a);
arr2.Swap(1, a);
arr1.Swap(1, a);
```

That one because I guess removing and then inserting an element in a 10000 elements array just to swap a couple of them would be quite slow....

A more nice way would be some sort of Detach() Set() that don't drop the element but replace it with a dummy (maybe null...) pointer. That would be :

```
Array<A> arr1, arr2;

arr1.Add(new A);
arr1.Add(new A);

arr2.Add(new A);
arr2.Add(new A);

A *a = arr1.NoDropDetach(1);
A *b = arr2.NoDropDetach(1);
arr1.NoInsertSet(1, b);
arr2.NoInsertSet(1, a);
```

Max
