Subject: Re: T* Detach() for ArrayMap
Posted by kohait00 on Wed, 03 Feb 2010 05:57:53 GMT

you're absolutely right, this one's better. should have looked before..

the goal was just to be able to switch from Array to ArrayMap, when i.e one gets the impression, well i need to speed up things, or need to change stuff, so just to enhance it in one place, and supply the keys in the other places, without the need of doing lot of workarounds to replace missing behaviour.

Detach() slowness was not the main goal in that, though having it done faster is precious. one should at least be able to separate an obejct from a ArrayMap again (which maybe were constructed outside the map and added).

this kind of aproach is not possible for Vector and VectorMap though, the objects are stored on a consequtive mem space, which can only be Removed(). but Vector/VectorMap requires to have a Mouvable object (objects are reorganized on storage space during growth/shrink of vector), so why not having same Detach() also here? which would simply copy the stored object to a decent new place and return pointer? maybe because its no real "detach".. but it would make life esier, when exchanging vector to array, array to map or waht ever. dont know for sure if it makes sense.

DetachUnlink also makes sense, to say simply "stay in background until reactivated..".. but how would you reactivate such one?