Subject: Re: Ptr improve
Posted by cbpporter on Tue, 24 May 2011 20:44:51 GMT
View Forum Message <> Reply to Message

Well, I guess you don't like my idea . It has been on my mind for some time. I am wondering if we could get a best of both worlds scenario out of this.

There is no need for writing compilers or libraries. Allocating memory without a container or a smart pointer and not freeing it would still be a memory leek. What I want is to delay the free operation. It does not have anything to do with pointer casting.

GC proponents have been hyping at least three things: no memory leaks, the advantage on parallel computing caused by a functional style combined with GC enabled more frequent allocations done to enable immutable data structures and and the responsiveness of allocation and deallocation. I was wondering if we could get some of that final point with what we have in U++ and test if it does bring an advantage or not. Seems like a fun experiment. We would need a very fast allocator, even at the price of a very slow deallocator.

And yes, my approach would need the use of Shared or a similar "rich pointer". Just using normal C pointer would be as bad of an idea as using them in normal U+ code to manage memory.