

---

Subject: Thoughts about resource management

Posted by [Mindtraveller](#) on Sun, 23 Jan 2011 15:28:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

You of course know "everything belongs somewhere" approach. If you tried it, you most probably know it as simple and effective way of getting rid of resource management problems. Simply speaking, you equalize visibility of object and period of its life by defining resource as stack object of parent class.

This works good for simple cases where resource lives constantly, starting with constructor of its parent until parent destructor is called.

But sometimes we meet more complex scenario, when resource is destroyed and re-created many times while program is working. We of course have Ptr/Pte wrappers. And I use them havily in these cases. Of course we may use public parent member functions which manage this resource for it not to violate our general approach.

But in my opinion we should discuss "everything belongs somewhere" for complex scenarios and widen U++ manual a little. Because complex scenarios is where smart pointers live, and we have to avoid mixing them with "U++ style".

---