Subject: Re: Proposed change to U++ to allow owning children.
Posted by mirek on Fri, 18 Mar 2011 18:32:51 GMT
View Forum Message <> Reply to Message

Lance wrote on Wed, 16 March 2011 12:18Newbie proposes changes to the library? Well, this is another proof of how well U++ is designed and implemented.

But will this break the "Everything belongs somewhere" principle? I don't think so. Owned Ctrls will be taken care of by their parents who own them. So they belong to their parents. As it's clearly defined and easily determineable, the principle is actually perfectly confirmed.

Sorry, but you would have hard time to convince me to go this way.

I guess the 'U++ legit' solution to this problem is to use Array.

```
struct Parent : ParentCtrl {
   Array<Ctrl> child;

   template<class T> T& Create() { T& x = child.Create<T>(); Add(x); return x; }
};
```

I believe that the moment you encourage using manual heap usage, the whole idea collapses. You start adding flags to what is owned and what is not everywhere and end in regular C/C++ heap mess, moments later wishing that you had garbage collector to deal with it...

Mirek