

---

Subject: Re: Proposed change to U++ to allow owning children.

Posted by [kohait00](#) on Tue, 19 Apr 2011 16:05:37 GMT

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

---

now got you..

though it's nice the way you planned it, the way u++ handles ownership in containers i.e. is a bit different.

there, the implicit rule of thumb is (assuming Array is used) that a pointer denotes a freshly created element (see `Add(T* x)` function in Array) for which the container needs to take ownership. moreover, there is the `Detach()` function, returning the pointer indicating that ownership has been stopped and the user needs to take care of that element again.

sth like this should also be expected from the ownership interface for Ctrl (if anytime to come).

IMHO ownership is not usefull in a C++ only / static environment (where no controls are created and thus nothing needs to be deleted). they are defined in Layout and done.

for dynamic environment OTOH, where controls with a reduced set of api (`GetData`, `SetData`, `WhenAction` Callback) are created and destroyed in a deliberate manner, ownership would improve codeability.. (especially thinking about scripting layouts, like my current intent: Python).

thus maybe mirek can help with some ideas. but it needs to be thought out well, since it touches the u++ philosophy..

BTW: as soon as the last little fix from CtrlLib is online, i'll post here a current environment, for dynamic control handling, like promised.