Subject: Re: why no 'Ctrl* Ctrl::Clone() const = 0' (virtual constructor)
Posted by kohait00 on Tue, 31 Aug 2010 11:45:47 GMT
View Forum Message <> Reply to Message

the new Ctrl would go to an Array<Ctrl>::Add(Ctrl* newt);
so it wont disappear..

this clone feature is quite well known from C# and i learned to like it..it enables you to manage your object containers from 'bottom', kind of flexibility.

something like the following woule be a layer to use maybe, but includes changes as well

```
template<class T>
class Clonable
{
public:
 virtual T* Clone() const { return DeepCopyNew<T>(*(T*)this); }
 virtual T* PartialClone() const { return new T(); }
};

class ValueC
 : public Value, public Clonable<ValueC>
{
public:
 ValueC() {}
 ValueC(const Value& v) : Value(v) {}
};

template<class T>
T* Clone(const T& c) { return DeepCopyNew(c); }

template<class T>
T* PartialClone(const T&) { return new T(); }
```