
Subject: properties like in C#

Posted by [kohait00](#) on Tue, 01 Feb 2011 13:45:29 GMT

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

see

<http://www.codeproject.com/KB/cpp/cppproperties.aspx>

which shows an implementation idea..

thanks to Callback we already can have stuff like that:

```
template<class T>
struct Property
{
    typedef Callback1<const T&> S;
    typedef Callback1<T&> G;

    Property(const S& s, const G& g)
        : set(s), get(g) {}

    const T& Set(const T& a) { ASSERT(set); set(a); return a; }
    T Get() const { ASSERT(get); T t; get(t); return t; }

    inline T operator= (const T& a) { return Set(a); }
    inline operator T() const { return Get(); }

public:
    const S set;
    const G get;
};

typedef Property<Value> PropertyValue;

...

//.h
class PropertyTest : public WithPropertyTestLayout<TopWindow> {
public:
    typedef PropertyTest CLASSNAME;
    PropertyTest();

    void GetD(Value& a) { a = "abc"; }
    void SetD(const Value& a) { RLOG(a); }

    PropertyValue vp;
};

//.cpp
```

```

PropertyTest::PropertyTest()

#pragma warning(push)
#pragma warning(disable:4355)
: vp( THISBACK(SetD), THISBACK(GetD) )
#pragma warning(pop)

{
    CtrlLayout(*this, "Window title");

    //setting, will call SetD
    vp = 123;

    //getting, will call GetD
    Value v = vp;
    RLOG(v);
}

GUI_APP_MAIN
{
    PropertyTest().Run();
}

```

now what's the benefit of it? imagine theCtrls beeing able to be parametrized like

```

EditInt ei;

ei.min = 100;
ei.max = 200;
ei.data = 150;
ei.show = true;
ei.rect = Rect(0,0, 100,200);
...

```

while i know it's not much better than current design rule of daisychaining methods, it makes code even bit more clean