
Subject: Re: New graph packages
Posted by [Didier](#) on Sun, 11 Mar 2012 18:50:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Koldo,

Quote: I have tried to understand your changes unsuccessfully. It is difficult for me to follow the changes in the files.

The changes I made are not functional modifications.

The idea is to move the common code from class A to a new template class: B.

The original class (A) then inherits from B: `class a : public B {};`

==> Class A therefore has the same methods as before, except for one (big) difference:

The return type of the methods : `B::x()` can only be of type B and not A

Usually in Upp method calls are chained: we write : `x0().x1().x2()`

So if I do `class a : public B {};` ==> the method chaining is then broken

To prevent this, we can use CRTP in order to have correct return types
CRTP

Better than long text here is an example:

----- Case 1 : the original code

```
class OriginalClass {
public:
    OriginalClass() {}
    OriginalClass& fctA() { .....; return *this; }
    OriginalClass& fctD() { .....; return *this; }
};
```

we can write `:OriginalClass inst;`
`isnt.fctA().fctB();` but all is contained in 1 class

----- Case 2 : naive separation (does not work)

```
class BaseClass {
public:
    BaseClass() {}
    BaseClass& fctA() { .....; return *this; }
};
class DerivedClass : public BaseClass {
public:
    DerivedClass() {}
    DerivedClass& fctD() { return *this; }
};
```

==> we **##CAN NOT##** write `:DerivedClass inst;`
`isnt.fctA().fctB();` ==> compilation error

----- Case 3 : separation using CRTP

```
template<class DERIVED>
class BaseClass {
public:
    BaseClass() {}
    DERIVED& fctA() { return *static_cast<DERIVED*>(this); }
};
class DerivedClass : public BaseClass<DerivedClass> {
public:
    DerivedClass() {}
    DerivedClass& fctD() { return *this; }
};
```

==> We CAN write :DerivedClass inst;
isnt.fctA().fctB(); ==> ScatterCtrl still works
and separation has been made ==> I can reuse it !!!

So please reconsider these changes since I only MOVED code and NOT CHANGED code.