
Subject: Re: Ptr improve

Posted by [tojocky](#) on Mon, 16 May 2011 14:06:54 GMT

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

mirek wrote on Mon, 16 May 2011 16:05Quote:Any comments are welcome!

No. This is going against U++ principles.

Mirek

Mirek,

As an alternative solution can you add a new virtual method OnPrecDelete() in class PteBase:

```
class PteBase {
protected:
    struct Prec {
        PteBase *ptr;
        Atomic n;
    };

    virtual void OnPrecDelete() {};

    volatile Prec *prec;

    Prec      *PtrAdd();
    static void PtrRelease(Prec *prec);
    static Prec *PtrAdd(const Uuid& uuid);

    PteBase();
    ~PteBase();

    friend class PtrBase;
};
```

and change in cpp file the method:

```
void PteBase::PtrRelease(Prec *prec){
    CriticalSection::Lock ____(sPteLock);
    if(prec && --prec->n == 0){
        if(prec->ptr){
            prec->ptr->prec = NULL;
            prec->ptr->OnPrecDelete();
        }
        delete prec;
    }
```

```

    prec = NULL;
}
}

```

In this case I can use for my specialized class this functionality by:

```
#include <Core/Core.h>
```

```
using namespace Upp;
```

```

struct Foo : public Pte<Foo> {
    String text;
    ~Foo();
protected:
    virtual void OnPrecDelete(){delete (Foo*)this;}
};

```

```

Foo::~~Foo(){
    Cout() << "deleted " << (void*)(this) << "\n";
}

```

```

Foo* factory(){
    return new Foo;
}

```

```

CONSOLE_APP_MAIN
{
    Ptr<Foo> ptr;
    {
        Ptr<Foo> ptr1 = new Foo;
        ptr1->text = "Text";
        ptr = ptr1;
        Cout() << (void*)~ptr << " -> " << ptr->text << "\n";
    }
    Cout() << "-----\n";
    Cout() << (void*)~ptr << "\n";
    ptr = factory();
    ptr = factory();
}

```