
Subject: Re: MakeOne
Posted by [Klugier](#) on Sat, 30 Jun 2018 12:45:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

I have a problem with current MakeOne implementation. It doesn't work greate with auto:

```
One<Foo> CreateFoo() {
    auto pFoo = MakeOne<Foo>();

    int i = 0;
    if (i == 0) {
        return nullptr;
    }

    return pFoo; // Compilation error - /home/klugier/MyApps/MakeOneTest/MakeOneTest.cpp (19):
    error: use of deleted function 'Upp::One<T>::One(const Upp::One<T>&)' [with T = Foo]
}
```

The below example works as expected:

```
One<Foo> CreateFoo() {
    One<Foo> pFoo = MakeOne<Foo>();

    int i = 0;
    if (i == 0) {
        return nullptr;
    }

    return pFoo;
}
```

I am not sure it is the good design to limit it to only explicit types. std::make_shared and std::make_unique works in above case.

Sincerely,
Klugier
