

---

Subject: Method Overrides

Posted by [Matthew Argall](#) on Mon, 04 Jun 2012 19:20:04 GMT

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

---

Hi,

I have a set of IDL objects and am running into method overriding problems at run-time. Any time I invoke a sub-class at run-time without explicitly compiling it before-hand, the class definition method gets compiled and executed, but not the over-riding methods. The super-class's methods still get used.

All methods are in the same file before/above the object definition method (`__define`), so I thought they would be compiled and recognized at the same time, but apparently not as I expected.

At run-time, I see

```
% Compiled module: CLASS__DEFINE
```

But if I compile it myself

```
IDL> .compile class__define
```

```
% Compiled module: CLASS__DEFINE::METHOD1
```

```
% Compiled module: CLASS__DEFINE::METHOD2
```

```
% Compiled module: CLASS__DEFINE::METHOD3
```

```
% Compiled module: CLASS__DEFINE
```

Could some one explain to me a little more clearly when or in which order to compile sub-classes and super-classes? Every time I use an over-riding method, do I either have to compile it or use the "obj -> classname::method" syntax to ensure that I get the right one?

Thanks

---