meta−class subclass relationships

Source: http://newsgroups.derkeiler.com/Archive/Comp/comp.lang.ruby/2007−11/msg02212.html

- *From*: Greg Weeks <greg.weeks@xxxxxxx>
- *Date*: Fri, 16 Nov 2007 04:20:16 −0500

Ruby exposes its singleton meta−classes, eg:

class <<B ; $Meta_B = self ; end

But their relationships are not quite what I expected. For example:

Suppose that A inherits from B (and B inherits from Object). Of course, this means that A instances respond to all the B instance messages. So:

A.new.is_a? B −> true

At the same time, the A class object responds to all the B class object messages. So:

```
class <<Object ; $Meta_Object = self ; end
class <<B ; $Meta_B = self ; end
class <<A ; $Meta_A = self ; end

A.is_a? $Meta_A −> true
B.is_a? $Meta_A −> false
A.is_a? $Meta_B −> true
B.is_a? $Meta_B −> true
```

Since everything that is_a $Meta_A also is_a $Meta_B, I expected a subclass/superclass relationship. Indeed, I expected the following to be true:

```
$Meta_A.superclass == $Meta_B
$Meta_B.superclass == $Meta_Object
$Meta_Object.superclass == Class
Class.superclass == Module
Module.superclass == Object
Object.superclass == nil
```

The above superclass chain reflects how a message to the class A object is looked up. What surprised me is that the first two equalities above are false.

Should they have been true?

FWIW: The superclass of both $Meta_A and $Meta_B is something called #<Class:Class>, which is its own superclass and is a subclass of $Meta_Object:

$X = $Meta_A.superclass −> #<Class:Class>
$X == $Meta_B.superclass −> true
$X == $X.superclass −> true
$X < $Meta_Object −> true
−−
Posted via http://www.ruby−forum.com/.

.