Ruby Metaclasses

Singleton, Eigen and Ghost Classes

How to do your head in.

Monday, 1 August 11

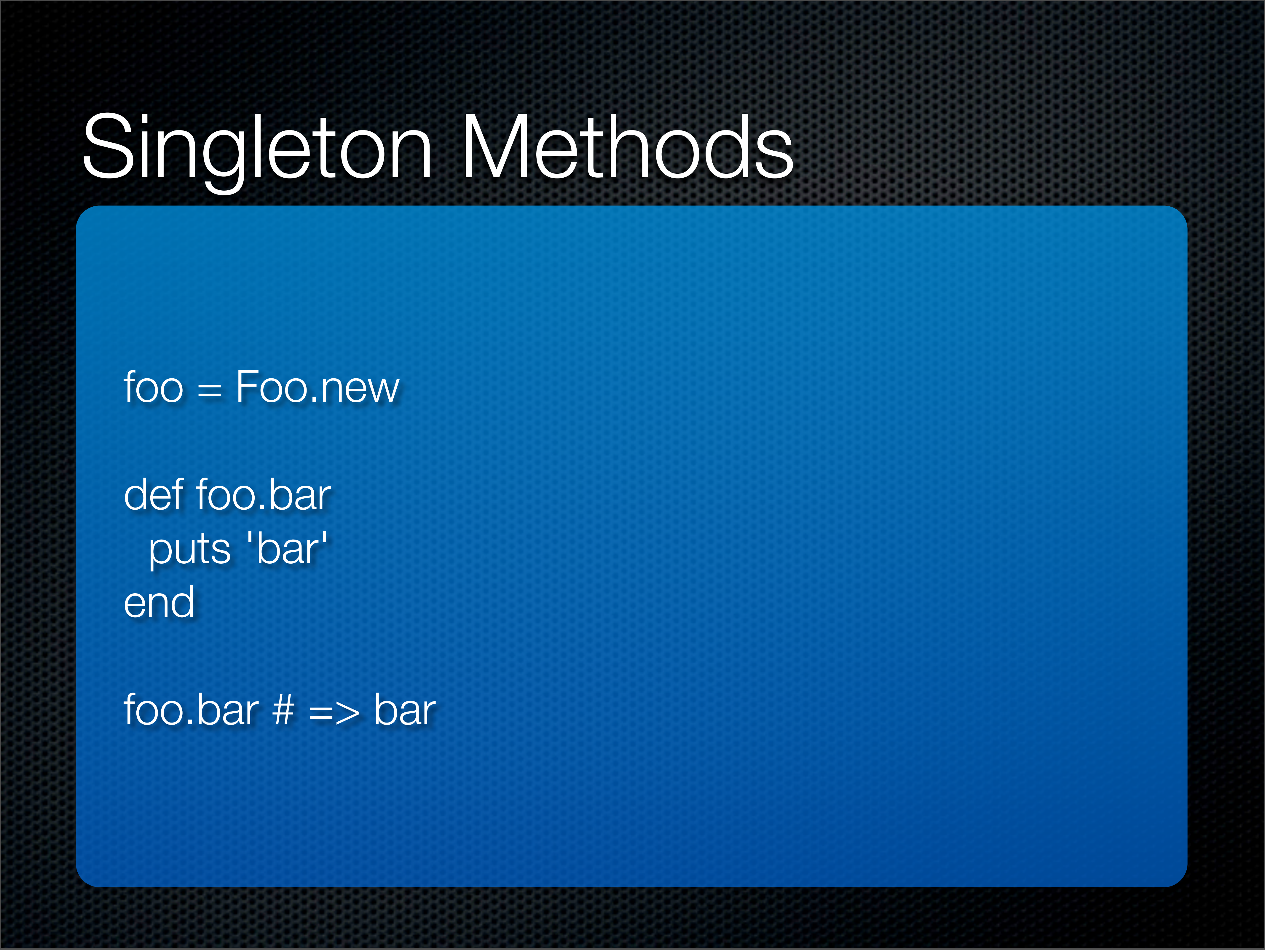
This stuff is hard to wrap your head around. Don’t worry if you don’t understand it the first time around.

Singleton Methods

- Defined on an instance of a class
- Held in a virtual class

Monday, 1 August 11

WTF is a virtual class? It’s how ruby allows you to over-ride things, it creates a class that sits in front of another class.



Singleton Methods

```
foo = Foo.new

def foo.bar
  puts 'bar'
end

foo.bar # => bar
```

Monday, 1 August 11

foo = instance of a class
.bar method exists only on the foo instance, not on the Foo class.

Class Methods

- Defined on a class
- Held in a virtual class

Class Methods

```
class Cat

  def self.says
    'Meow'
  end

end
```

Monday, 1 August 11

.says is defined on the Cat class.
Cat.says => ‘Meow’

Class Methods

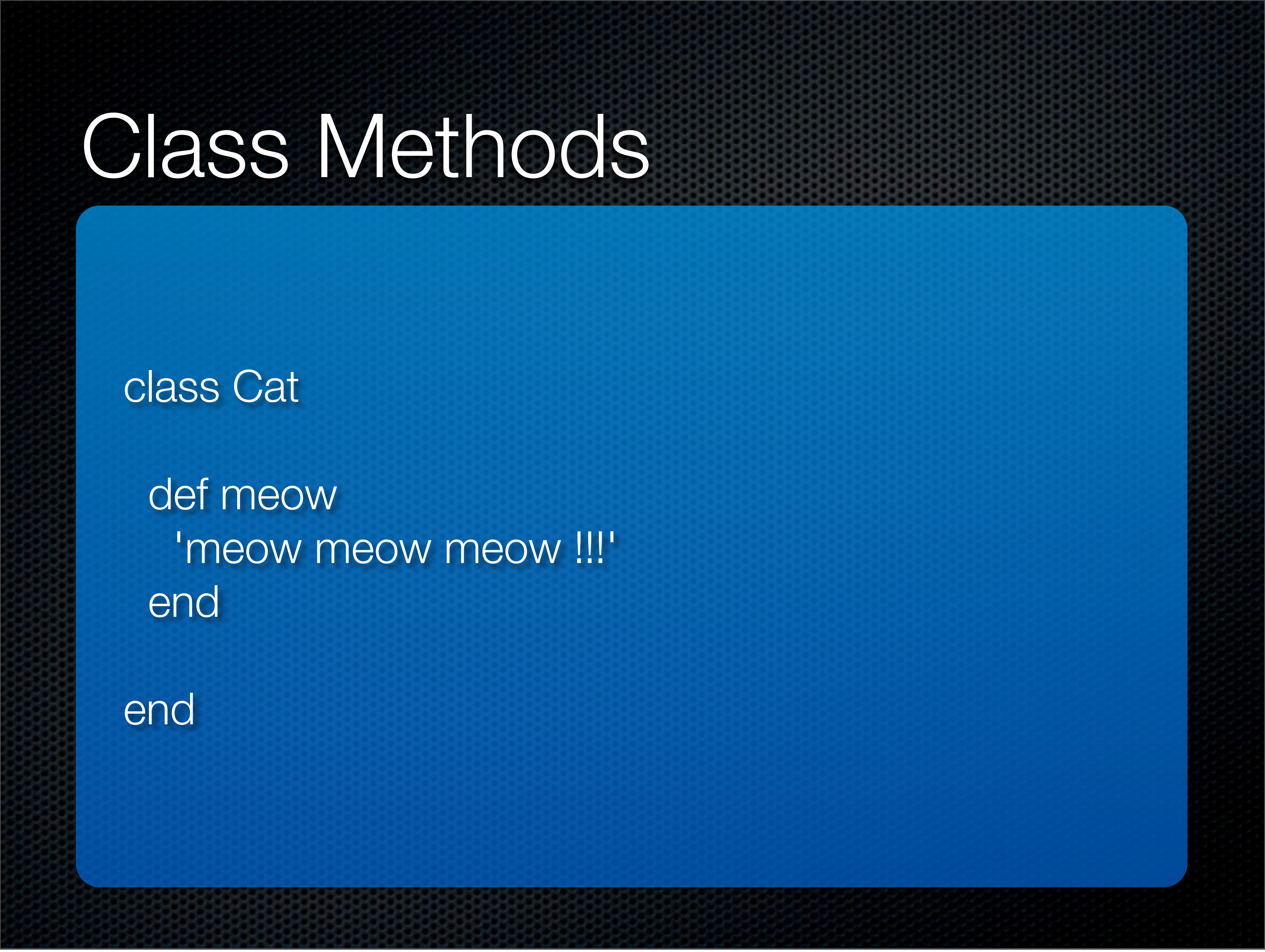
- Why is it held in a virtual class?
- Cat is an instance of Class

Instance Methods

- Defined on a class
- Held in the class

Monday, 1 August 11

No need for a virtual class here.



Class Methods

```
class Cat

  def meow
    'meow meow meow !!!'
  end

end
```

Every instance of the Cat class has the meow method.

instance_eval

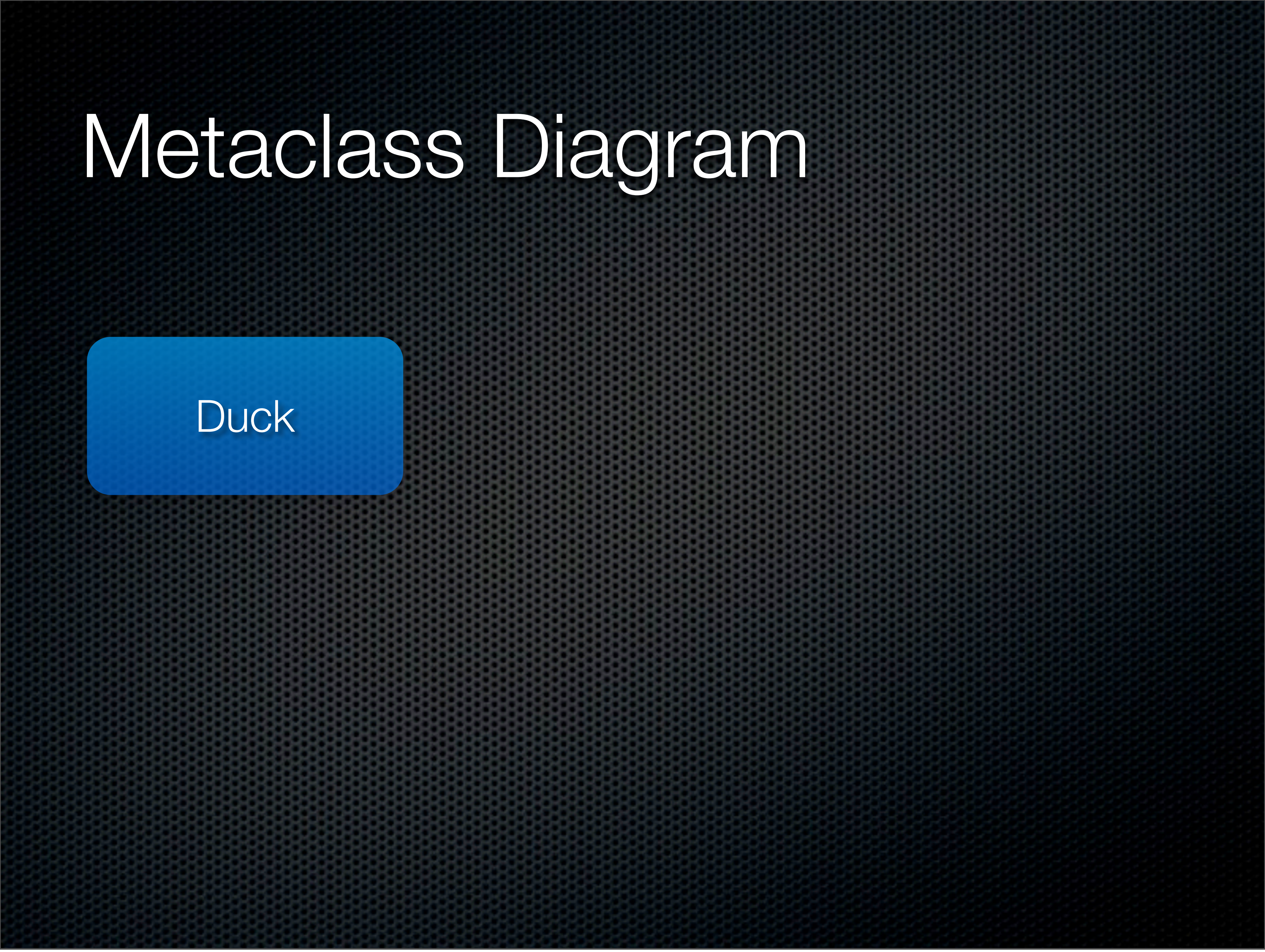
- Used on an instance of a class to create a singleton method
- Used on a class to create a class method

Monday, 1 August 11

Time for some brain warping.
Think about the second line a bit, we’re effectively creating a singleton method on an instance of class.

class_eval

- Used on a class to create an instance method

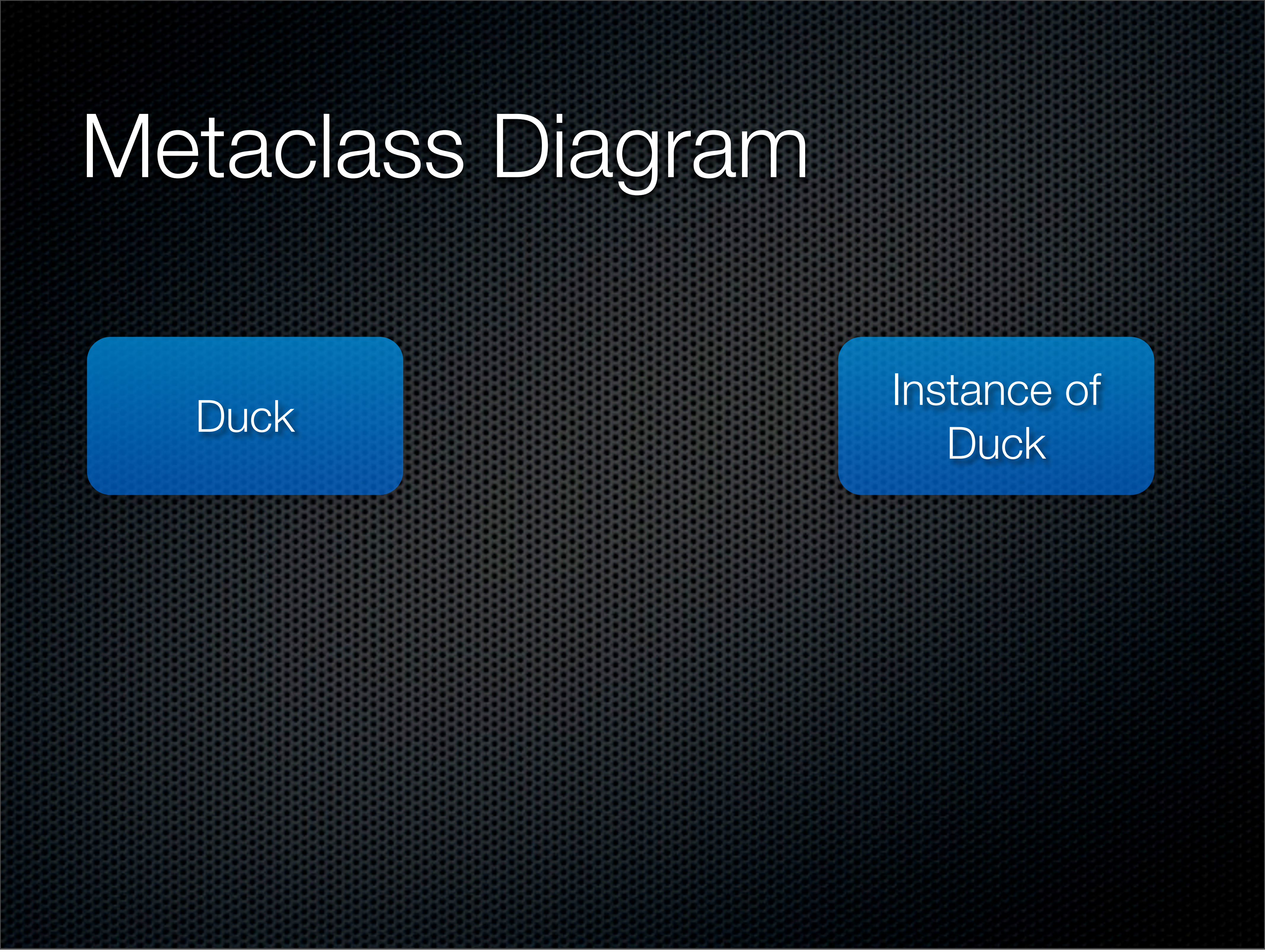


Metaclass Diagram

Monday, 1 August 11

Here’s a quick diagram of what you get when you use each item.

Duck is a class, or an instance of Class.



Metaclass Diagram

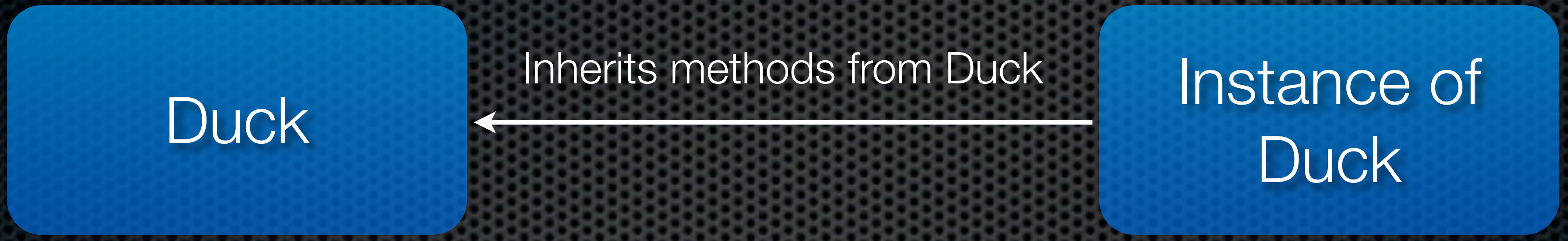
Duck

Instance of Duck

Monday, 1 August 11

This is an instance of the Duck Class.

Metaclass Diagram



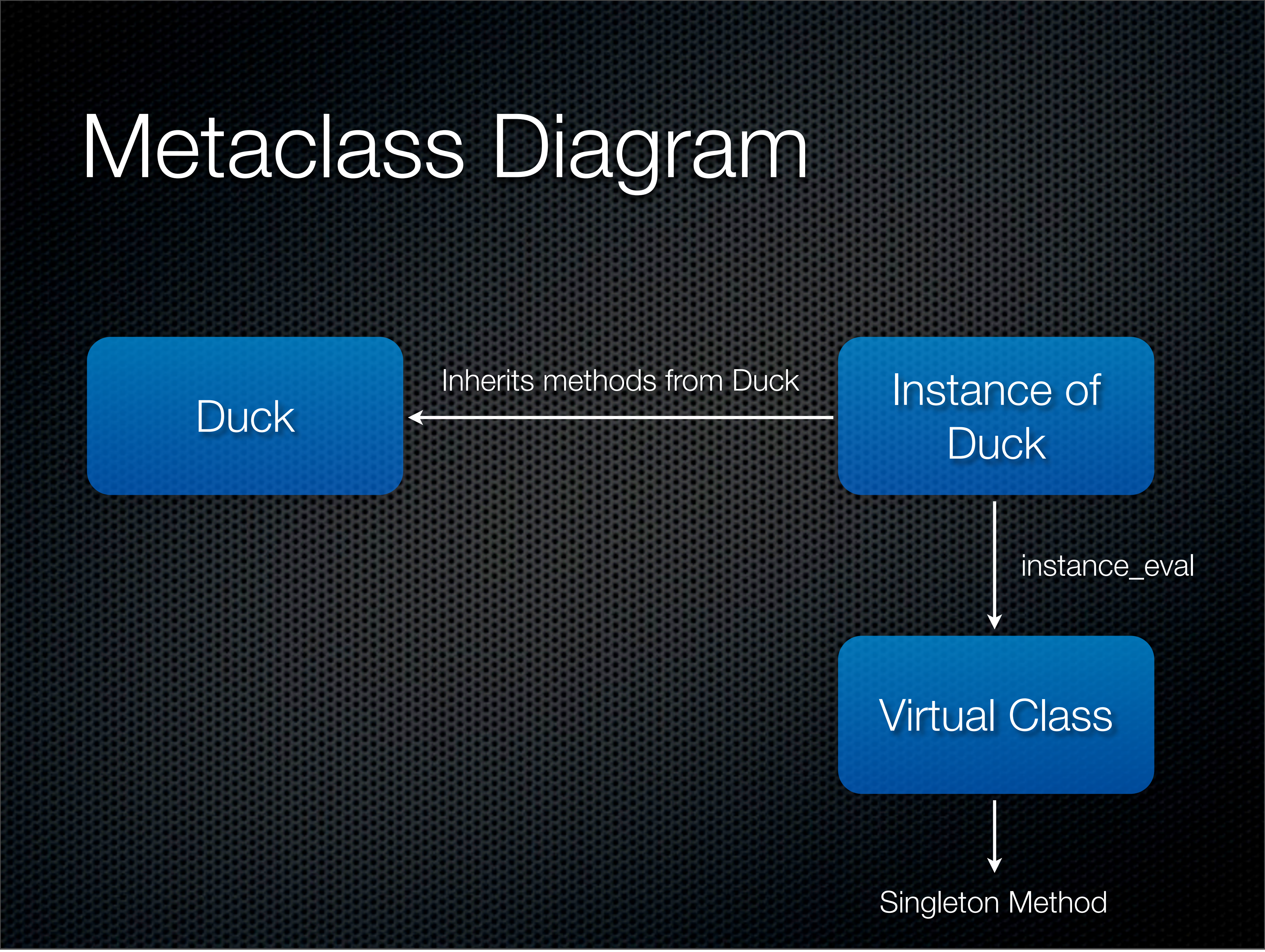
Duck

Inherits methods from Duck

Instance of Duck

Monday, 1 August 11

The instance of the Duck Class inherits all of its method from the Duck class.



Metaclass Diagram

Duck

Inherits methods from Duck

Instance of Duck

instance_eval

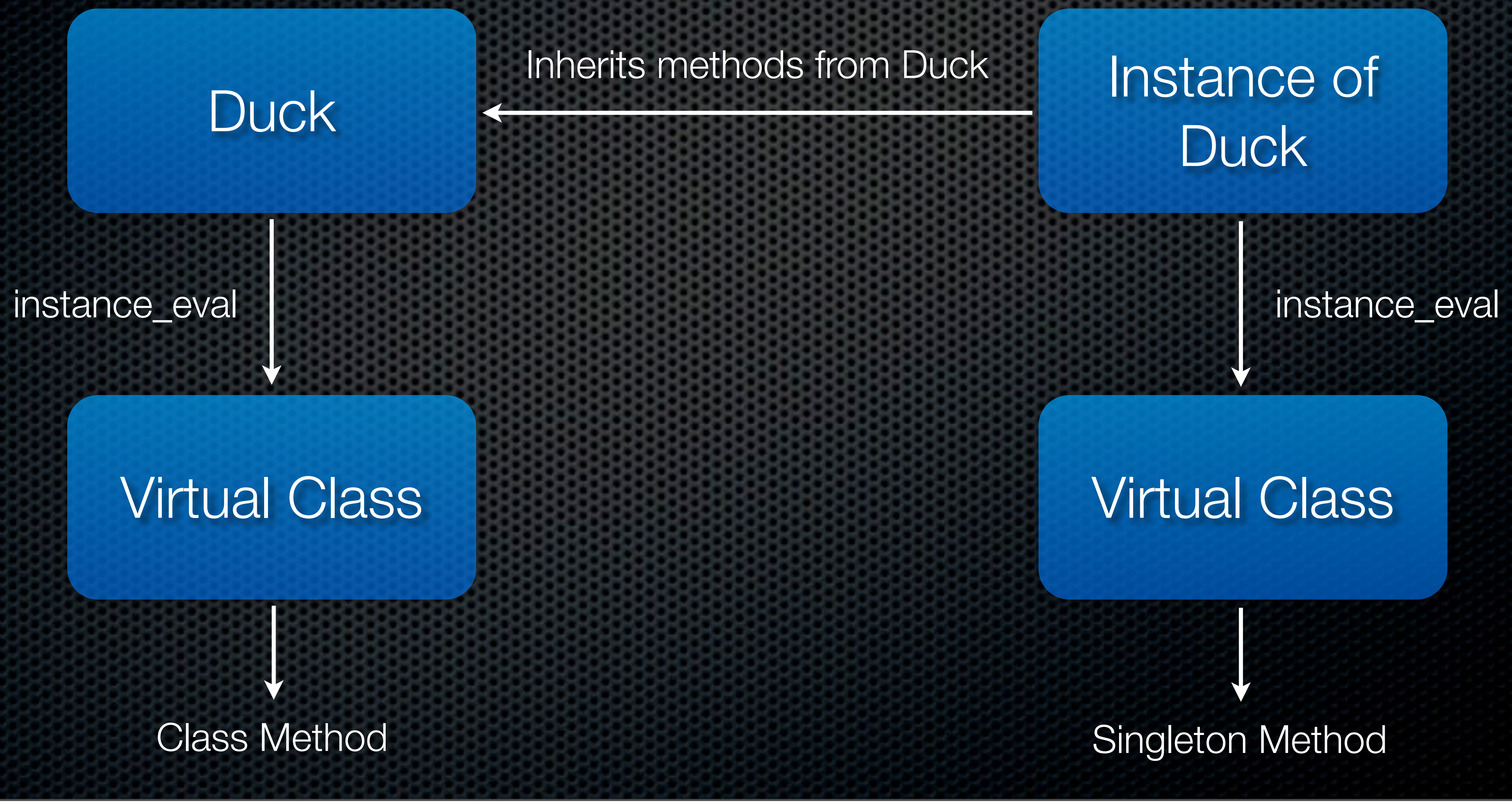
Virtual Class

Singleton Method

Monday, 1 August 11

We use instance_eval in an instance of a class to create a singleton method.
It’s stored in a virtual class.
Ruby first sends a message to the virtual class, then to the instance.

Metaclass Diagram



Duck

Inherits methods from Duck

Instance of Duck

instance_eval

instance_eval

Virtual Class

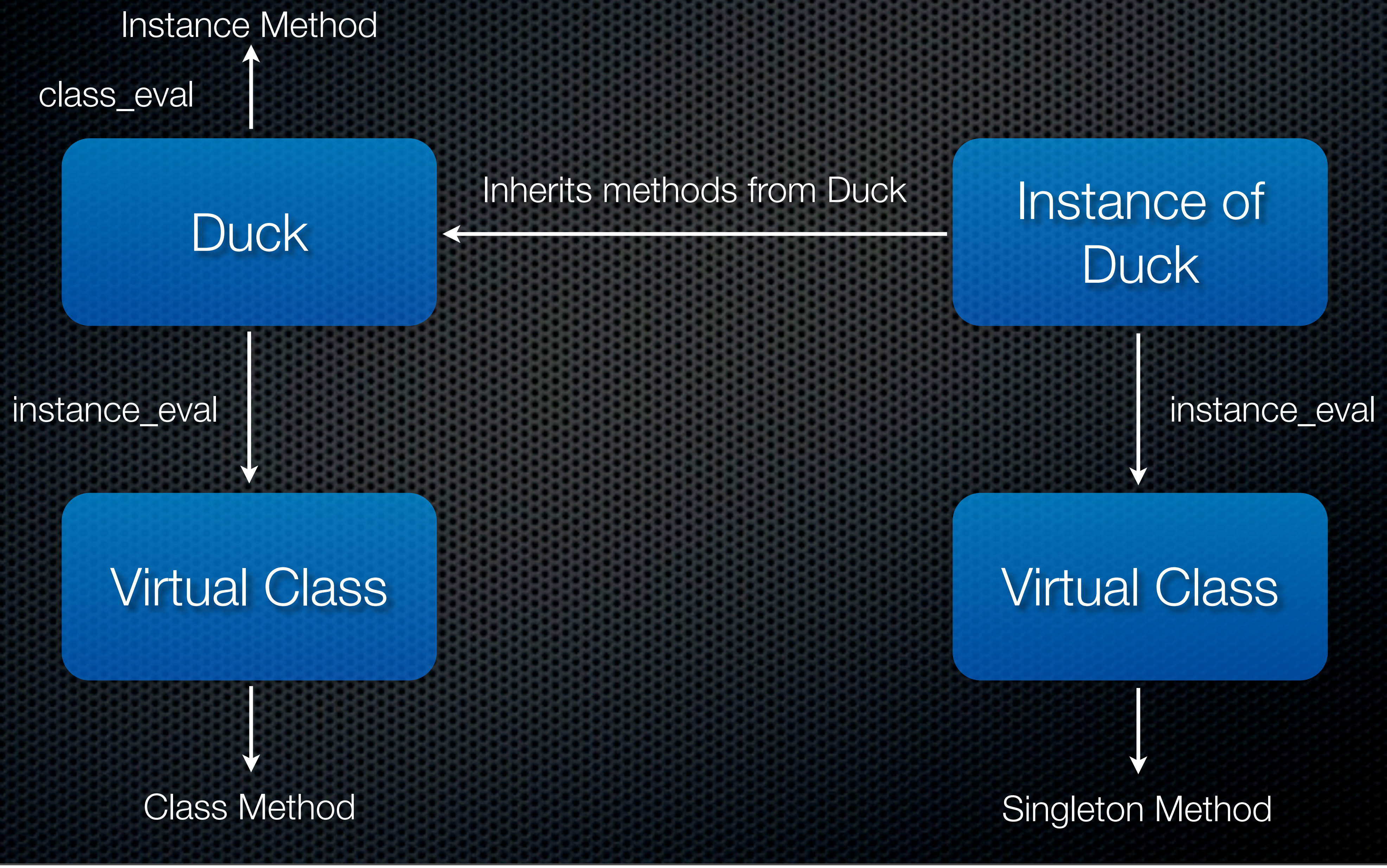
Virtual Class

Class Method

Singleton Method

We use instance_eval on the the Duck Class (remember it’s an instance of Class) to create a Class Method.
Stored in a virtual class.
Available only to the Duck instance of Class.

Metaclass Diagram



Instance Method

class_eval

Duck

Inherits methods from Duck

Instance of Duck

instance_eval

instance_eval

Virtual Class

Virtual Class

Class Method

Singleton Method

Monday, 1 August 11

We use class_eval to create an instance method on the Duck class.
Stored in the Duck instance of Class.
Available to all instances of Duck.