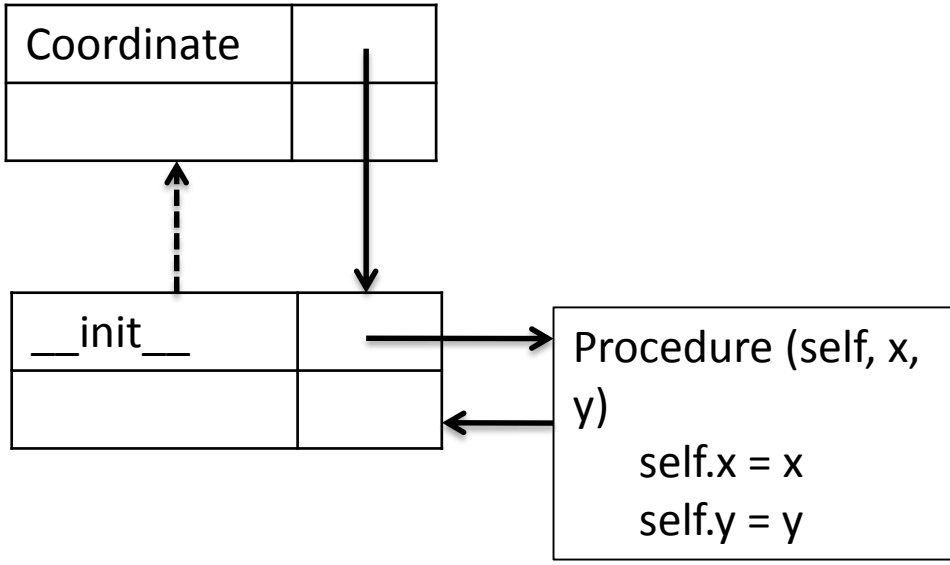
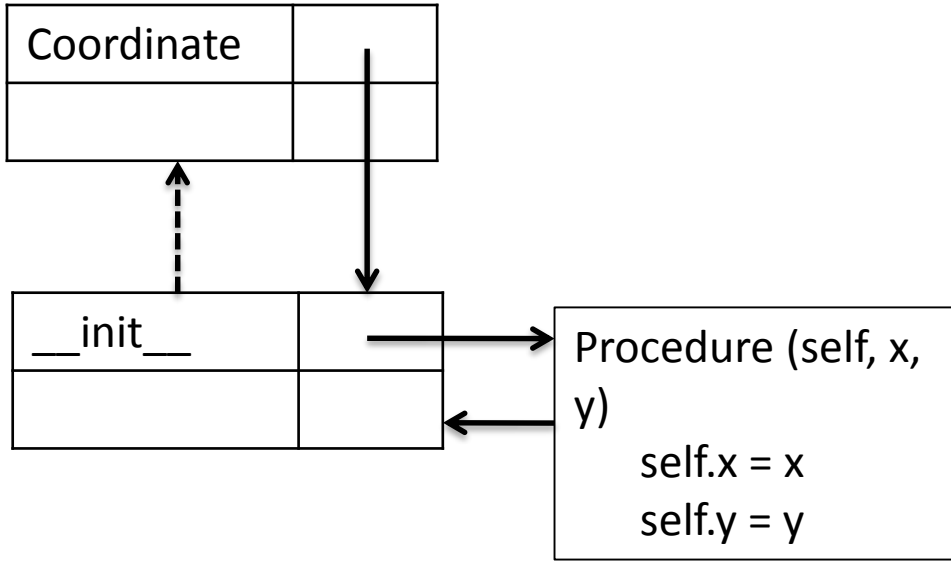
An environment view of classes

- Class definition creates a binding of class name in global environment to a new frame or environment
- That frame contains any attribute bindings, either variables or local procedures
- That frame also knows the parent environment from which it can inherit



An environment view of classes

- In this case, the only attribute is a binding of a name to a procedure
- But if a class definition bound local variables as part of its definition, those would also be bound in this new environment

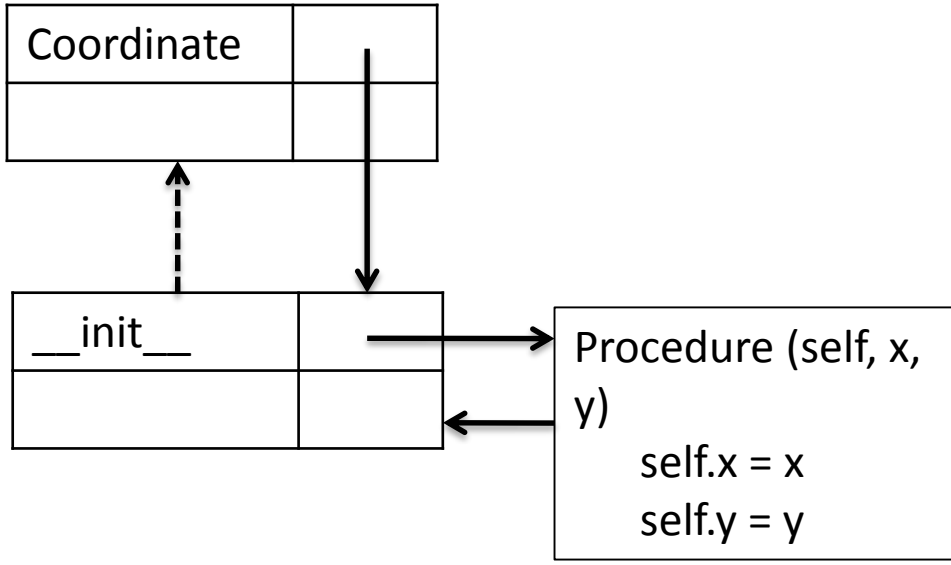


An environment view of classes

- We can access parts of a class using

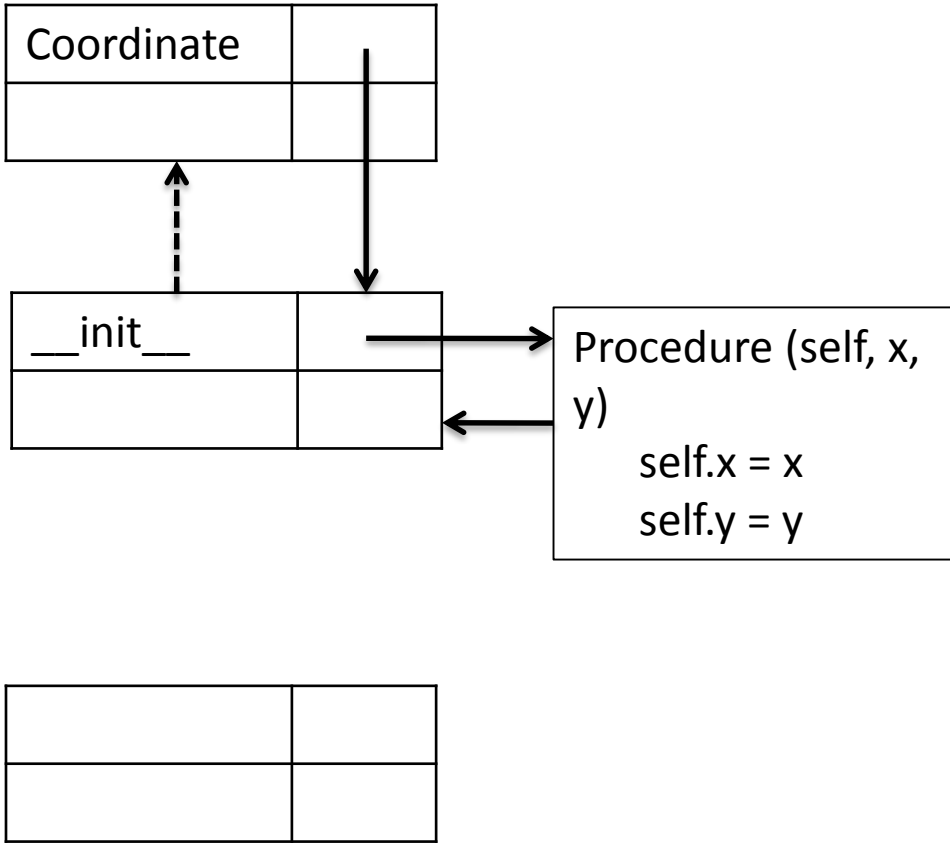
`Coordinate.__init__`

- Python interprets this by finding the binding for the first expression (which is a frame), and then using the standard rules to lookup the value for the next part of the expression in that frame



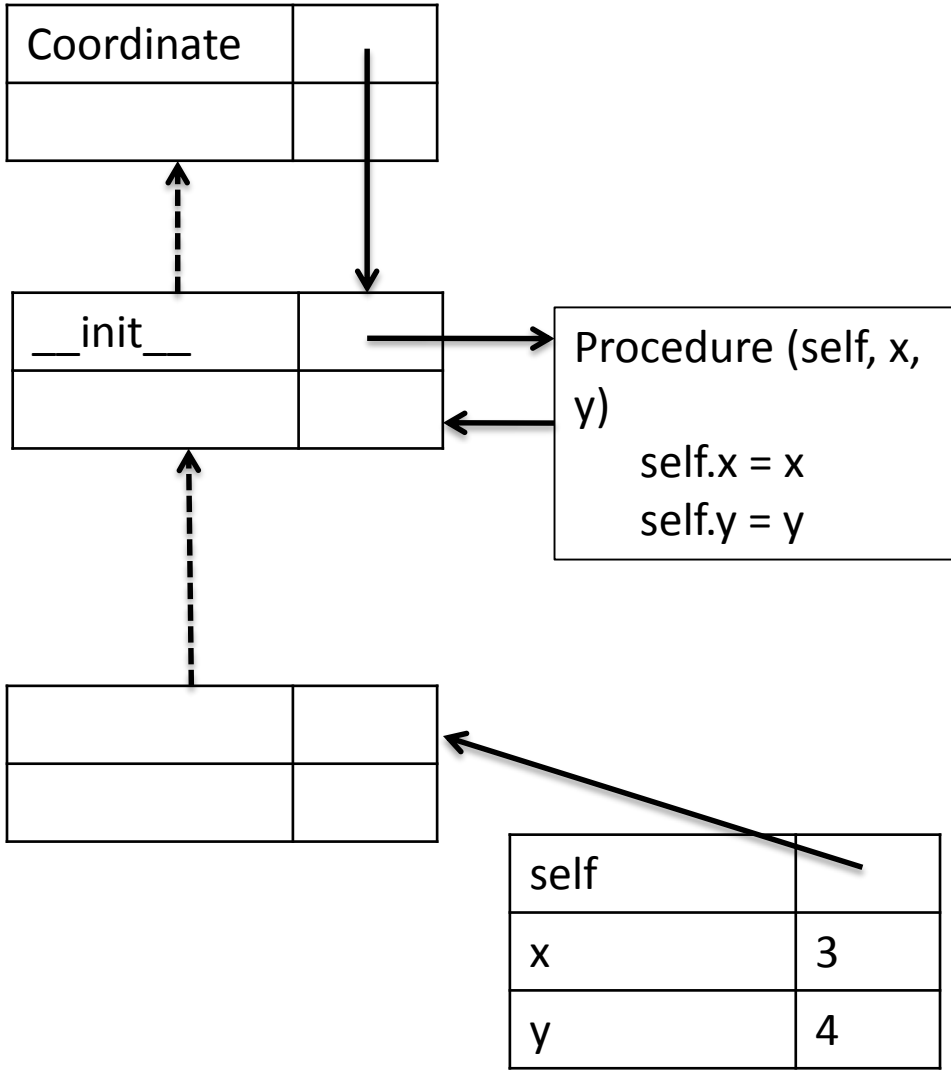
An environment view of classes

- Suppose the class is invoked
  - c = Coordinate(3,4)
- A new frame is created (this is the instance)



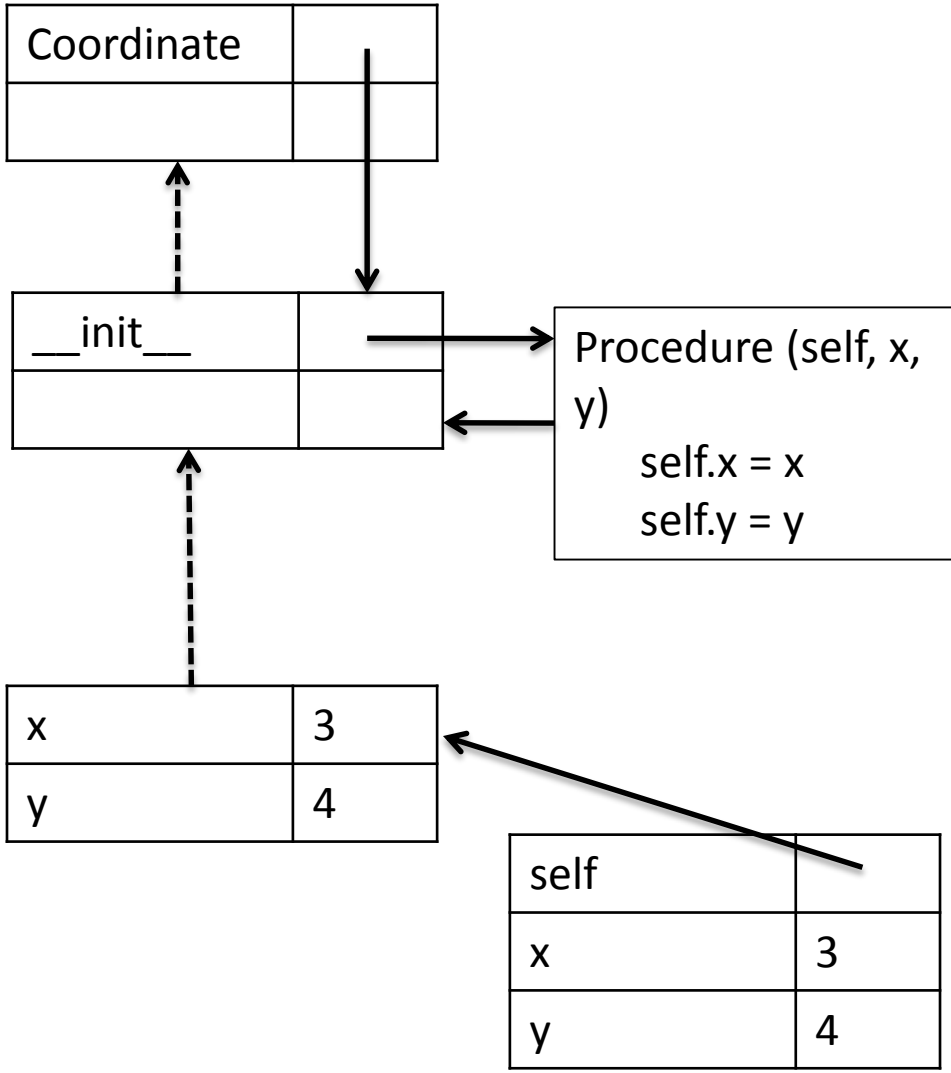
An environment view of classes

- Suppose the class is invoked
  - c = Coordinate(3,4)
- A new frame is created (this is the instance)
- The __init__ method is then called, with self bound to this object, plus any other arguments
- The instance knows about the frame in which __init__ was called



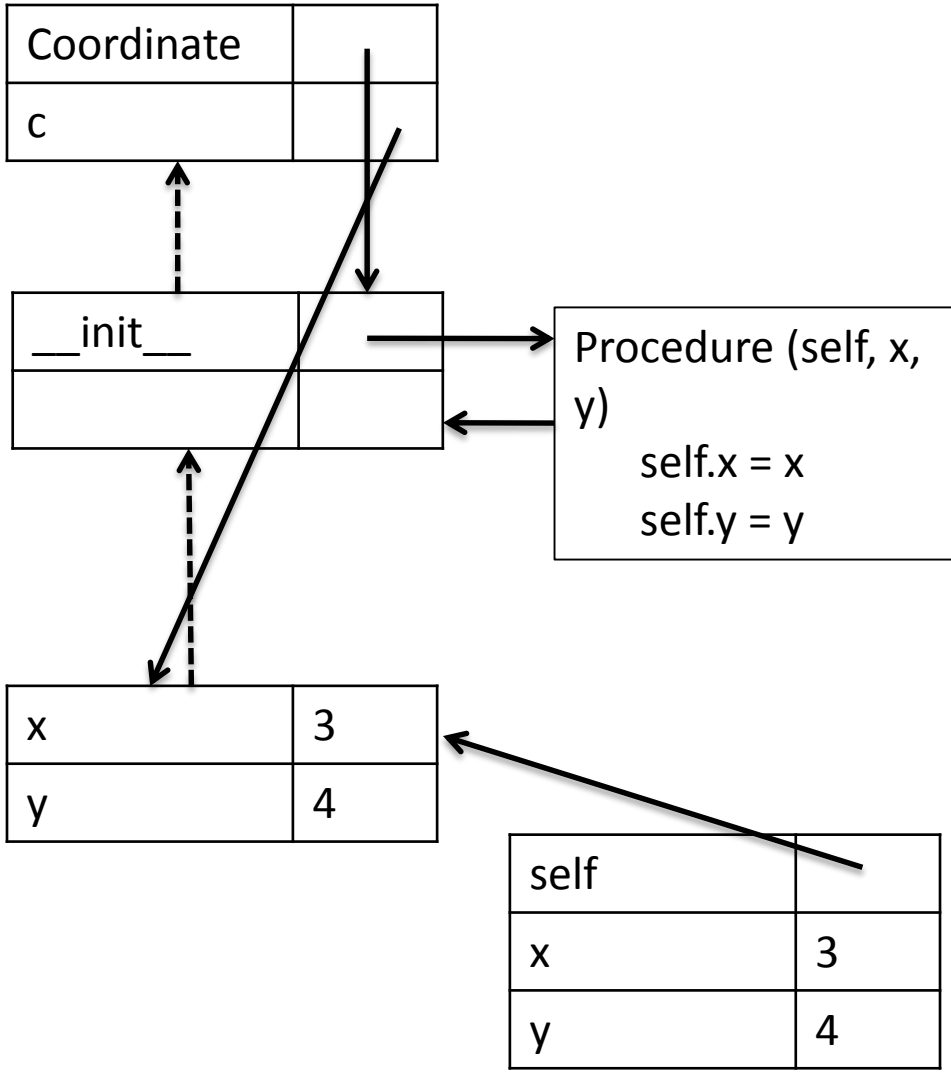
An environment view of classes

- Suppose the class is invoked
  - c = Coordinate(3,4)
- A new frame is created (this is the instance)
- The __init__ method is then called, with self bound to this object, plus any other arguments
- Evaluating the body of __init__ creates bindings in the frame of the instance



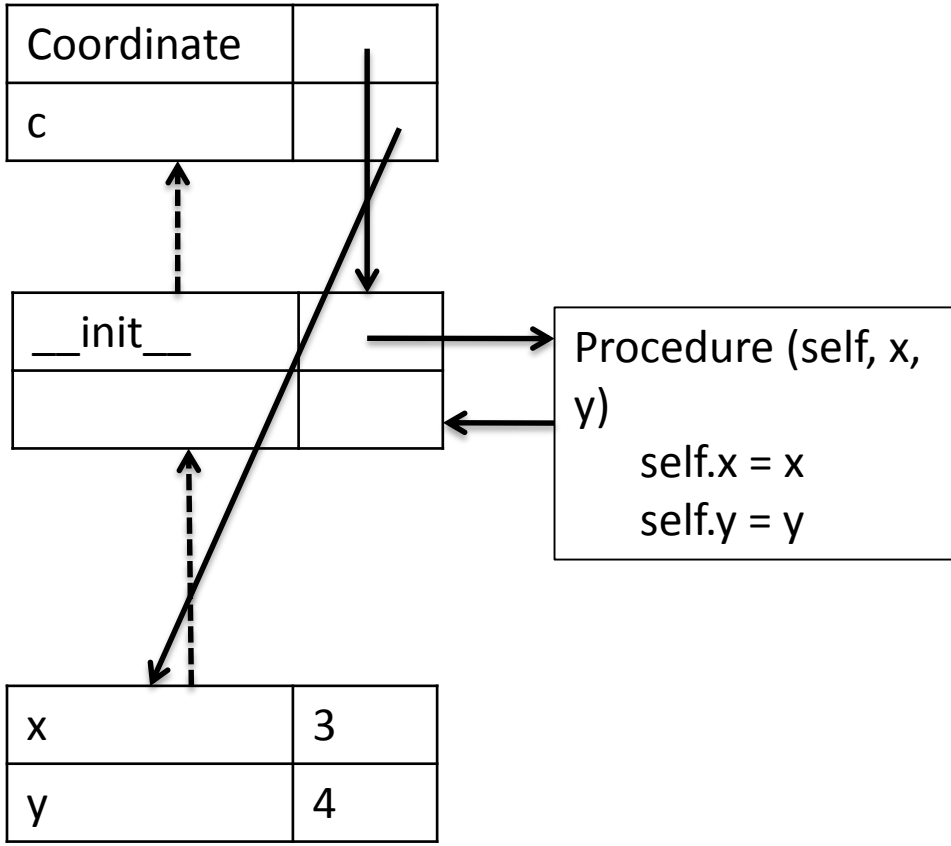
An environment view of classes

- Suppose the class is invoked
  - c = Coordinate(3,4)
- A new frame is created (this is the instance)
- The __init__ method is then called, with self bound to this object, plus any other arguments
- Evaluating the body of __init__ creates bindings
- Finally the frame created by the class call is returned, and bound in the global environment



An environment view of classes

- Given such bindings, calls to attributes are easily found
- c.x will return 3 because c points to a frame, and within that frame x is locally bound

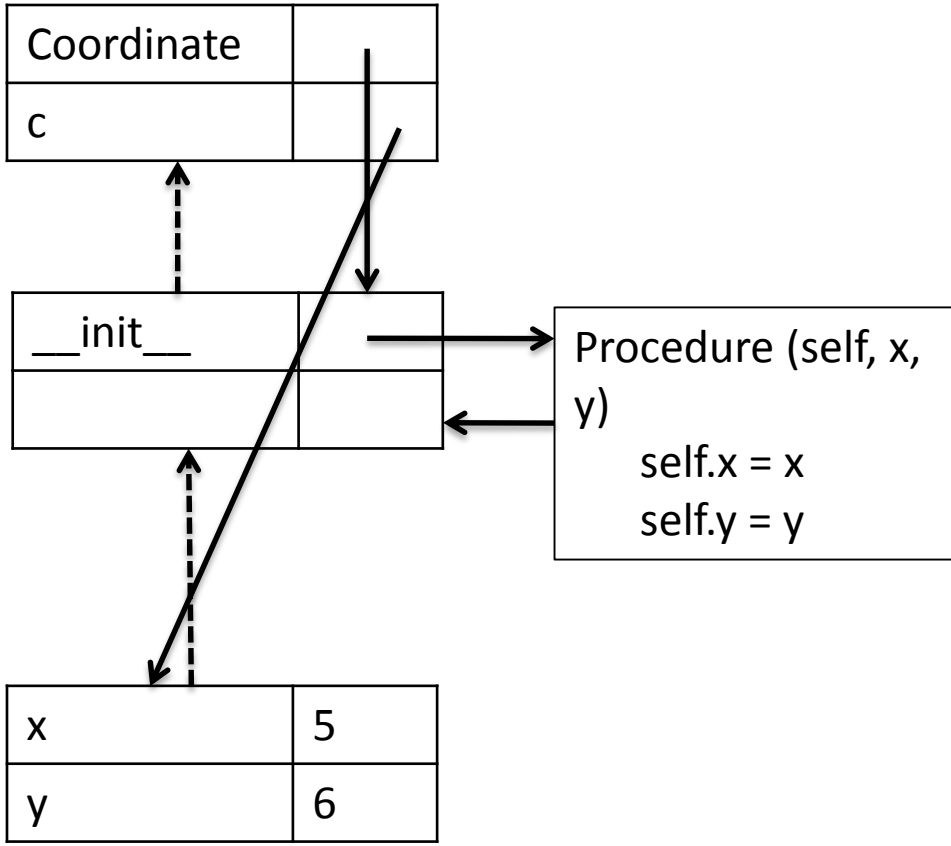


An environment view of classes

- Given such bindings, calls to attributes are easily found
- c.x will return 3 because c points to a frame, and within that frame x is locally bound
- Note that c has access to any binding in the chain of environments

```
c.__init__(5,6)
```

- will change the bindings for x and y within c



An environment view of classes

- Given such bindings, calls to attributes are easily found
- c.x will return 3 because c points to a frame, and within that frame x is locally bound
- Creating a new instance, creates a new environment, e.g.

```
Origin = Coordinate(0,0)
```

- This shares information within the class environment