

The CLOS Metaobject Protocol

What is an object?

“An object has state, behavior, and identity.”

(Grady Booch, 1991)

State

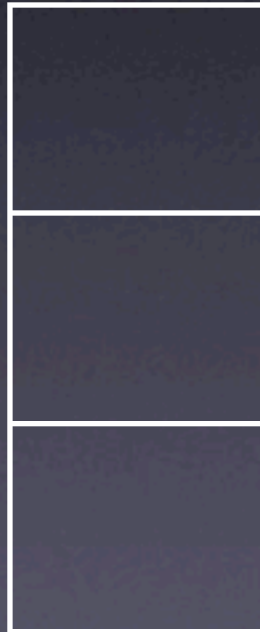
obj



(slot-value *obj* 'x) →

(slot-value *obj* 'y) →

(slot-value *obj* 'z) →



(aref *obj* 0)



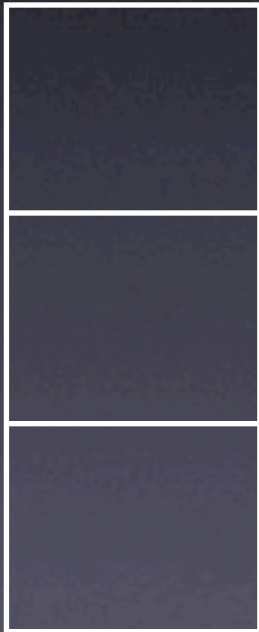
(aref *obj* 1)



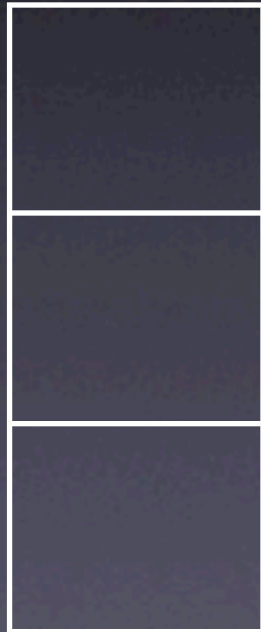
(aref *obj* 2)

Identity

obj1



obj2

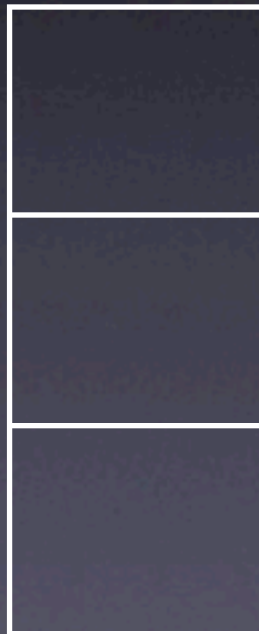
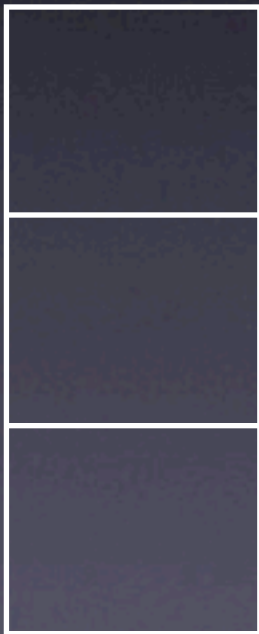


(eq *obj1* *obj2*)
=> nil

Identity

obj1

obj2



(eq *obj1* *obj2*)
=> nil

(setf *obj1* *obj2*)

(eq *obj1* *obj2*)
=> t

How to map slots?

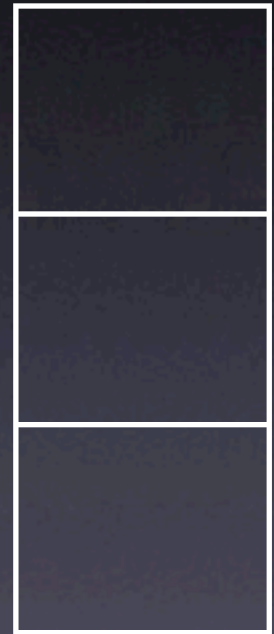
```
(defclass point ()  
  (x y))
```

```
(defclass point-3d (point)  
  (z))
```

x?

z?

y?



How to map slots?

1. compute class precedence list
2. compute slots
3. determine slot locations

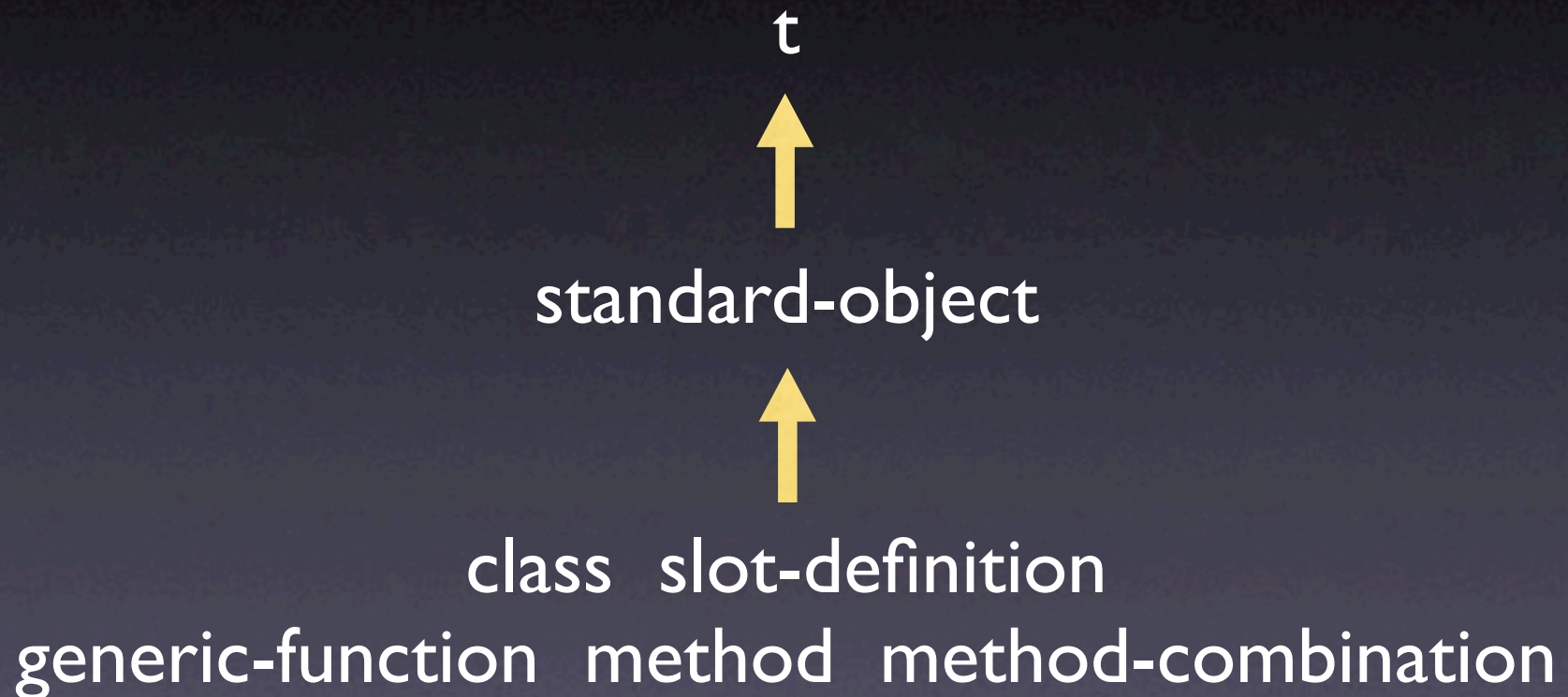
How to map slots?

1. (compute-class-precedence-list ...)
2. (compute-slots ...)
3. (slot-definition-location ...)

The Idea!

- Make compute-class-precedence-list, compute-slots, and so on, generic functions!
- Allow changes to the CLOS object model!
- Question: How to distinguish between standard and non-standard behavior?

Hierarchy for metaobject classes



Class

metaobject classes

class



built-in-class

standard-class

forward-referenced-class

Class

metaobject classes

```
(defclass persistent-class (standard-class)  
  ((database-connection ...)))
```

```
(defclass person ()  
  ((name ...)  
   (address ...))  
  (:metaclass persistent-class))
```

Class

metaobject classes

```
(defun slot-value (object slot)
  (slot-value-using-class
    (class-of object) object slot))
```

```
(defmethod slot-value-using-class
  ((class persistent-class) object slot)
  (fetch-slot-from-database ...))
```

Slot definition metaobject classes

slot-definition



standard-slot-definition



standard-direct-slot-definition
standard-effective-slot-definition

Method metaobject classes

method



standard-method



standard-accessor-method



standard-reader-method standard-writer-method

Subprotocols...

Example: The Python object model

1. Define a mix-in for hashtable-based slots.
2. Ensure that this mix-in is used.
3. Define a new allocation type `:hash`.
4. Modify the slot access protocol.