



Learning Object-Oriented Programming and Design with TDD

What is an Object?

entity with state and reacting to messages

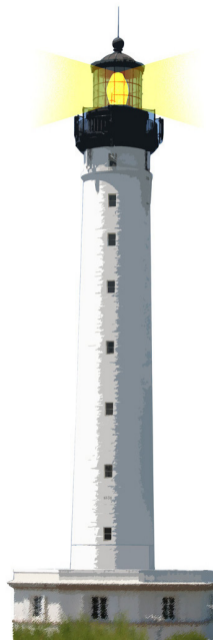
Stéphane Ducasse

<http://stephane.ducasse.free.fr>



<http://www.pharo.org>

W3S01



An object

A little computer entity with

- a specific state
- a behavior
- an identity

An entity collaborating with other entities



Object Example: a Die

aDie(6)

- aDie(6) has a state: 6 faces
- aDie(6) knows how to roll
- aDie(6) is part of aDieHandle (2D6 + 2D20)



Object Example: a Rectangle

(Rectangle width: 10 height: 20)

- has a state: an height and a width
- knows how to compute its area, edge closest to a point, overlap with other rectangles
- UI uses rectangles to redisplay

Object Example: a Contact

aContact

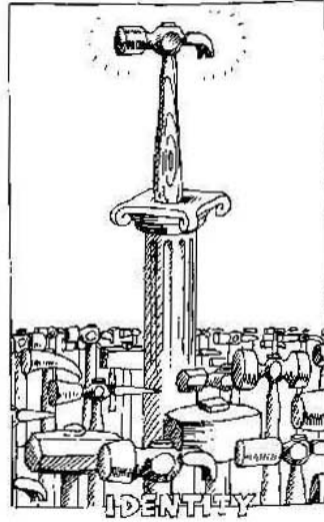
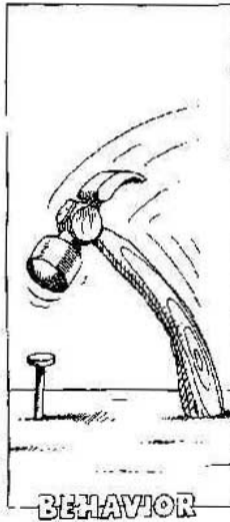
- aContact has a state: name, firstname, telephone, address
- aContact knows how to display itself
- aContact is in relation with a contact book



Object Example: an Item Collector

- aCollector has a state: lists of items
- aCollector knows how to save itself, search its items, modify its items
- aContact is relation with an object loader/saver, UI, list of items

State + Behavior + Identity



State + Behavior + Identity

- **State:**
 - Objects it contains or refers to (ex: point location)
- **Behavior:**
 - On object understands a given set of messages
- **Identity:**
 - On object can be the same (of the same class) than another one but it has still a different identity (location in memory)



About Equality and Identity

You enter a pizzeria and order a pizza

- **Equality** I want to eat the **same** kind of pizza
- **Identity** eat your pizza



Object Responsibilities

- An object has a set of responsibilities that he offers and maintains
- The responsibilities are realized by the execution of behavior
- Example
 - a turtle draws line, jumps, changes directions



An Object

- Encapsulates data
- Reacts to messages by executing methods

Messages

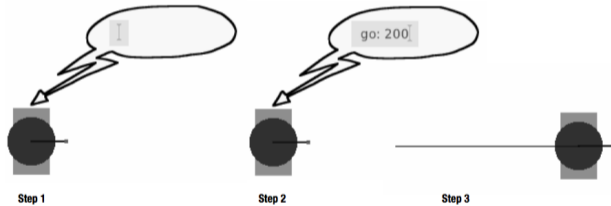
Methods

Data

A diagram illustrating the structure of an object. It consists of three concentric circles. The innermost circle is light purple and labeled "Data". The middle circle is light blue and labeled "Methods". The outermost circle is a darker blue and labeled "Messages".

An Object Reacts to Messages

```
aTurtle go: 200
```



Different Objects can Understand the Same Message

Each object may perform a different reaction

aWindow open

aFile open

aZip open



A Method Specifies a Behavior

```
Rectangle >> height  
"Answer the height of the receiver."  
^corner y – origin y
```

```
Rectangle >> area  
^ self width * self height
```

```
(Rectangle origin: 10@10 corner: 50@110) area  
>>> 4000  
(Rectangle origin: 10@10 corner: 50@60) area  
>>> 2000
```



A Method Can Send Other Messages

```
Rectangle >> topCenter
```

```
"Answer the point at the center of the receiver's top horizontal line."
```

```
^ self center x @ self top
```

```
Rectangle >> center
```

```
"Answer the point at the center of the receiver."
```

```
^ self topLeft + self bottomRight // 2
```

```
Rectangle >> top
```

```
"Answer the position of the receiver's top horizontal line."
```

```
^ origin y
```

```
(Rectangle origin: 10@10 corner: 50@110) topCenter
```

```
>>> 30@10
```

```
(Rectangle origin: 10@10 corner: 50@60) topCenter
```

```
>>> 2000
```

Delegating to Other Objects

An object may delegate to other objects

- An object can request another object to perform some tasks for him.
- The sender does not have to know how the delegatee is actually implementing the reaction to his message.



Delegation Example

- An item collector asks each of its items to save themselves.
- An item collector asks an object serialisation system to save itself.



What you should know

- An object is characterized by a state, a behavior and an identity
- Objects answer to messages by executing methods
- An object may delegate to other objects its tasks



A course by Stéphane Ducasse
<http://stephane.ducasse.free.fr>

Reusing some parts of the Pharo Mocc by

Damien Cassou, Stéphane Ducasse, Luc Fabresse
<http://mocc.pharo.org>



Except where otherwise noted, this work is licensed under CC BY-NC-ND 3.0 France
<https://creativecommons.org/licenses/by-nc-nd/3.0/fr/>