

TinySelf: Tiny object-oriented language Using the Self programming system

Tomas Petricek, Charles University

✉ tomas@tomasp.net

✈ [@tomasp.net](https://tomasp.net)

🌐 <https://tomasp.net>

🌐 <https://d3s.mff.cuni.cz/teaching/nprg077>



Everything is an object

Really everything

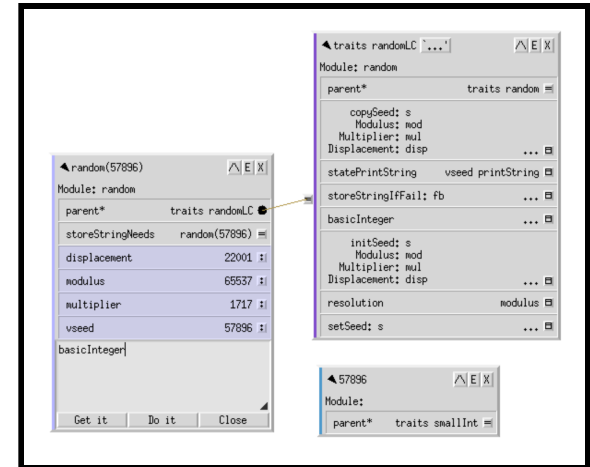
Objects, methods, lambdas, expressions, activation records

What is an object

Object has a list of slots and optionally contains code

Object = slots* + code?

- Data object has just slots
- Method object has code
- Closure has code and slots!
- Data object has methods as slots



```
// Object consists of zero or more
// slots and optionally code
type Objekt =
  { mutable Code : Objekt option
    mutable Slots : Slot list }

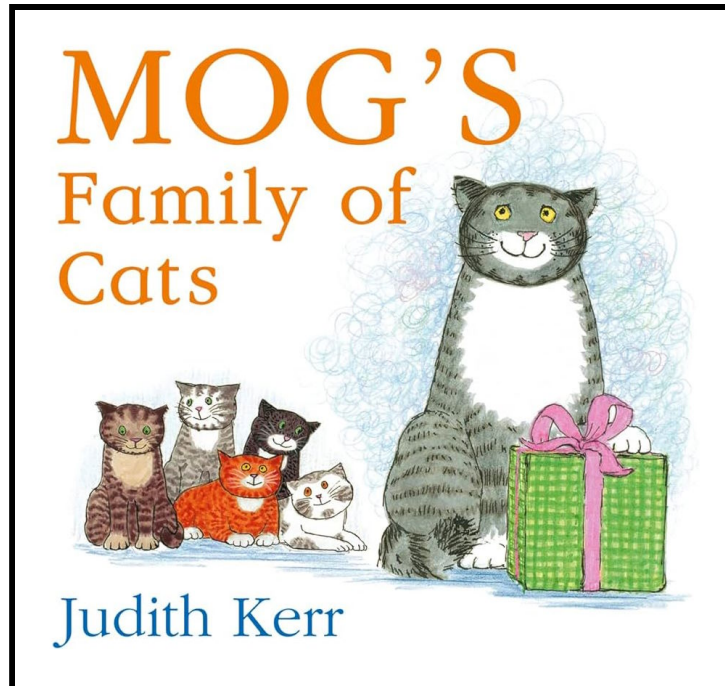
// A slot has name and contents;
// Some slots are parents
and Slot =
  { Name : string
    Contents : Objekt
    IsParent : bool }
```

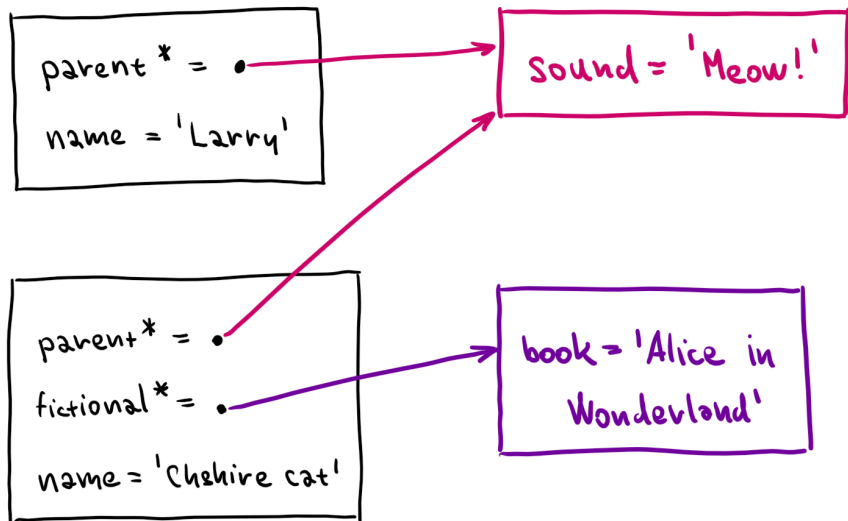
TinySelf objects

Object consists of zero or more slots and optional code!

In Self parent slot names end with *

TinySelf objects can also be special things





Prototypes and slots

Message send looks at target object first, then searches parents

```
cheshire name // OK  
cheshire book // OK
```

```
larry name // OK  
larry book // Fail
```

Message send fail if none or multiple slots found

Demo

Representing cats in Self

```

"""Data object with name"""
(| book = 'Alice in Wonderland' |)

"""Method with some code"""
( self name printLine )

"""Data object with parent
slot and a 'speak' method"""
(| parent* = cat
  name = 'Cheshire Cat'
  speak = (
    self sound printLine
  )
|)

"""Data access or method call"""
cheshire name
cheshire speak

```

Message sending

Lookup slot with a matching name, then:

- If it contains data object, it is returned
- If it contains method, the method is called

Assignment slots and special calls differ...

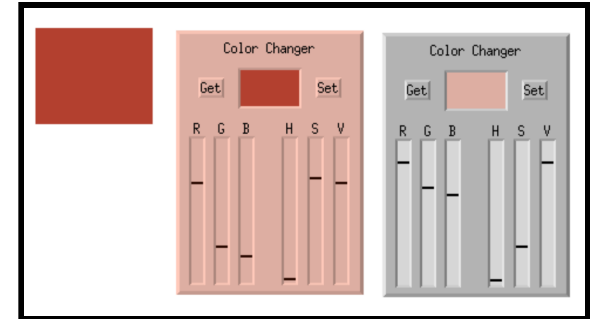
Demo

Hello world and traits

The power of simplicity...

Simplicity and uniformity

- All objects can be opened!
- Activation records for debugging
- Self-sustainable system



Morphic framework

- Things on screen are objects!
- Object with a morph prototype can draw itself
- User interface is just morphs - no special code!

Demo

Morphic and graphical objects