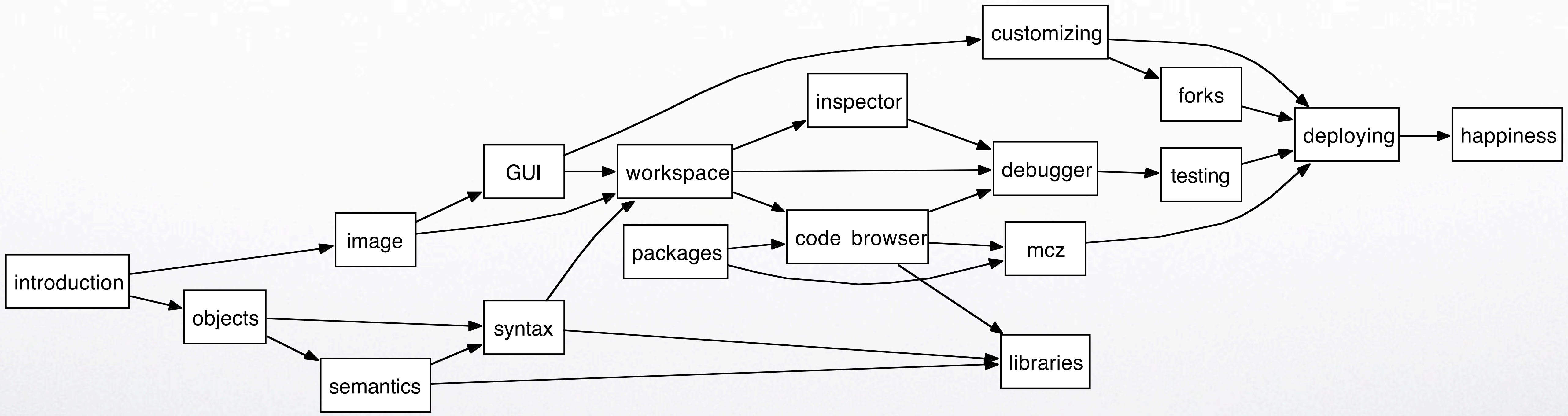
Introduction to Smalltalk

Randal L. Schwartz, merlyn@stonehenge.com

Version 1.01 on 20 July 2009

Roadmap


introduction
image
objects
semantics
syntax
GUI
workspace
packages
inspector
code browser
customizing
debugger
mcz
libraries
forks
testing
deploying
happiness

Introduction

- Not Avi Bryant
  - Co-creator of Seaside
  - Chief dude at dabbledb.com
- Randal Schwartz
  - Member of Squeak Oversight Board
  - Hacking Squeak since beginning
  - Hacking Smalltalk since beginning
    - Yes, long before Perl :)

Objects and classes

- Everything's an object
- An object belongs to a class
- An object has methods
  - The class (also an object) has methods
- A class inherits from a single superclass
  - Class-side and instance-side separately

Variables

- Alphanumerics
- Camelcased with initial lowercase:
  - rate, accelerationRate
- Value belongs to a class, variables don't care
- Assign to get a value:
  - rate := 30
- Special vars:
  - self, true, false, nil, super, thisContext

Methods

- Unary: single name, follows variable:
  - rate squared
- Binary: 1-2 punctuation chars:
  - rate * time
- Keyword: names and colons:
  - rate raisedTo: 2.5
  - rate between: 5 and: 10
- Simple precedence!

Literal data

- Strings: ‘hello world’
- Numeric data: 3 2.5 1.23e45 -2e-5
- Symbols: #size #foo:bar: #+

Classes

- Alphanumeric, initial uppercase
- Class methods are often constructors:
  - rates := Set new.
- But could also have other uses:
  - superclassOfSet := Set superclass.

Method syntax

- Signature (like message send without self):
  - squared
  - * aNumber
  - raisedTo: aNumber
  - between: lowNumber and: highNumber
- Temporaries: | aDog aCat |
- Statements separated by periods
- Last statement can have ^ ("answer this")
- Comments are in "double quotes"

Control structures

- Conditionals:
  - aBoolExpr ifTrue: [some. code. here].
  - aBoolExpr ifFalse: [some. other. code].
  - #ifTrue:ifFalse:, #ifFalse:ifTrue:
- Loops:
  - [code. code. aBoolExpr] whileTrue.
  - [code. aBoolExpr] whileTrue: [code].
  - #whileFalse, #whileFalse:

Cascades

- Cascade omits common object:
  batallion selectTank target: enemy; fire.
- Same as (without the variable):
  aTank := batallion selectTank.
  aTank target: enemy.
  aTank fire.

Fancier Blocks

- Like methods in square brackets
- Argument list (if any)
  :arg1 :arg2 :arg3 |
- Temporaries (if any)
  | temp1 temp2 |
- Statements
  arg1 dothis. arg2 dothat. arg3 + arg1.
- ^ exits enclosing method, not block!

Example of Blocks

- add3ToDoubleOf := [:x | x * 2 + 3].
- a := add3ToDoubleOf.
- b := a value: 17.
- c := a value: (17 / 2).

Collection classes

- OrderedCollection
- Array (fixed size OrderedCollection)
- Set
- Bag (counted items of Set)
- SortedCollection (order from chaos)
- Dictionary (key/value mappings)
- Interval (5 to: 100 by: 3)

Collection protocols

- collect:
  - fractions := (1 to: 10) collect: [:n | 1 / n]
- do:
  - fractions do: [:each | html div: each]
- select:
  - overQuarter :=
    fractions select: [:f | f > 0.25]

Image

- Everything "above the VM"
  - But the VM is deliberately low-level
- Thus, nearly everything
- Includes core behaviors
- Also includes devel/debug tools
- Eventually includes your code
- Your code gets extracted for deploy
- "Minimal images" remove unneeded stuff

The Smalltalk GUI

- Workspace - run snippets of code
- Browser - edit and view code
- Debugger - “d” bugs
- Inspector - look at complex values
- Monticello - share and load code

The Workspace

- Evaluate code snippets
- “do it”: just run the code
- “print it”: run the code, show the result
- “debug it”: run the code in the debugger
- Operates on selection
  - If nothing selected, line cursor is on

Packages

- Collections of related classes
- Typical unit of development
- SCM tools can track package dependencies
- SCM repos store versions of packages
- Each class belongs to only one package
  - Monkeypatching allows other methods

The Code Browser

- Packages - groups of classes
- Classes
- Class/instance/comment toggle
- Method categories (including "all")
- Method names
- Lower pane views/edits selection
  - Sometimes preloaded with a template
- Lots of coding help available in menus

Inspector

- Simple inspect: sensible debug value
- Inspect view: simple first-level parts
- Explore view: triangle-expose hierarchy

The Debugger

- Debug notifier: proceed/cancel/full
- Full debugger:
  - Stack
  - Code pane (current line highlighted)
  - Instance vars
  - Temps and arguments
- Everything is live, editable, resumable
- Action buttons to step in, over, through

Testing

- Unit testing
- GUI test runners

Monticello (MCZ)

- Primary SCM for Squeak
- Distributed SCM (like git, hg, bzr)
  - Good merge tools
- Repos can be many formats
  - Simplest is directory on local disk

Libraries

- Many many published libraries for Squeak
- Some require specific VM plugins
- Some require other libraries
  - Most libs are published as MCZ
- Many libs are published in Squeakmap
- Many libs are published in Universes
- Many libs are just sitting in Squeaksource

Customizing

- Preferences pane
- Loading in new libs
- Starting with new images as a base

Forks

- FunSqueak
- Damien’s “dev” images (normal and web)
- Pharo
- Cuis
- Seaside one-click image
- FunSqueak

Deployment

- Need:
  - mywork.image
  - and its correlated mywork.changes
  - VM for your platform
  - Squeak3.9.sources (doesn’t change)

Happiness!

- www.squeak.org
- www.pharo-project.org
- www.jvuletich.org/Cuis/
- squeak-dev mailing list
- squeak-beginners mailing list
- MethodsAndMessages.vox.com