Pharo: an immersive object-oriented system

Damien CASSOU, Stéphane DUCASSE and Luc FABRESSE

www.pharo.org
W1S02
Pharo?
Pharo!

- System: Pure object language + full IDE
- Inspired by Smalltalk
- Powerful, elegant and fun to program
- Great community
- Living system under your fingers
- Works on Mac OS X, Linux, iOS, Android, Windows, Pi
- 100% MIT
Elegant!

- Full syntax on a postcard
- Simple and powerful objet model
exampleWithNumber: x

“A method that illustrates every part of Smalltalk method syntax”

<menu>
| y |
true & false not & (nil isNil) ifFalse: [self halt].
y := self size + super size.
#($a #a ‘a’ 1 1.0)
do: [:each | Transcript
  show: (each class name);
  show: (each printString);
  show: ‘ ‘].
^ x < y
Full Model!

- Dynamically typed
- Everything is an object instance of a class
- All methods are public virtual
- All attributes are protected
- Single Inheritance
Fully Written in Itself

- Everything is written in Pharo
- Pharo itself
- One simple syntax/model to access everything
Immersive?
Immersive

- Pharo is not a blackbox
- Pharo is fully inspectable and reflective
- You can get immersed in objects
Immersed and interacting

- You can interact with objects
elements := (1 to: 5) collect: [ :ob |
R3CubeShape new) elementOn: ob ].
lay := R3WallLayout new.
lay on: elements.
UberPresenter present: elements
Pharo by Example

- http://books.pharo.org
- translated to french, merci!
- translated to spanish, gracias!
- translated to japanese, ありがとう!
- Currently updated to Pharo 50
Deep into Pharo

- http://books.pharo.org
- Some advanced topics
  - Libraries
  - Core language
  - Tools
Entreprise Pharo

- http://books.pharo.org
- Web related
  - Encodings,
  - Mustache
  - JSON
  - WebSockets
  - HTTP/HTTPS
Dynamic Web with Seaside

- Full framework explained
- DSL + Call:/answer:
- Automatic form generation
- Deployment
Numerical Methods

- https://github.com/SquareBracketAssociates/NumericalMethods
- Data mining
- Clustering
- Iterative algorithms
- Series
- Statistical analysis
A System to Learn From

 Pulitzer

Click on it

Cmd+Shift+Option
Ask the Finder

canStartMatch: aCharacter in: aMatcher

"Answer whether a match could commence at the given lookahead character, or in the current state of <aMatcher>. True answered by this method does not mean a match will definitely occur, while false answered by this method "does" guarantee a match will never occur."

aCharacter isNil ifTrue: ["true"],
~testBlock == nil or: [testBlock value: aCharacter value: aMatcher]
Ask Spotter

* Shift + Enter
From Examples to Messages

```
otherCollection
"Concatenate two Strings or Collections."

^self copyReplaceFrom: self size + 1
to: self size
  with: otherCollection
"

#(2 4 6 8), #(who do we appreciate)
((2989 storeStringBase: 16) copyFrom: 4 to: 6), 'boy'
```

About OO Design

Programming in Pharo will change the way you program and think OO
How not is implemented?

- false not -> true
- true not -> false

Why such question illustrates something so deep?
Try Pharo now!

http://pharo.org/download

Download Pharo

Version 4.0 for OS X, GNU/Linux, and Windows. The zip files contain all necessary files. Just download and run the executable.
Pure & elegant
Fun, simple
Highly productive
Excellent for teaching
Empowering
Full access