Pharo Status

Marcus Denker

http://www.pharo-project.org
Pharo3: Release April `14

- Started March 2013
- 2390 Issue tracker entries with Pharo3 tag closed
- 854 Updates
Iceberg

- A lot of Changes!
- Not everything visible
Lots of Activity
A lot of Change!
Yet easy to adopt

- Moose switched in two afternoons (two people)
- Others: “I just loaded my packages”
Infrastructure: CI

- https://ci.inria.fr is stable and used a lot

- Every fix is validated automatically before human review
- Every update triggers test run on 3 Architectures
- over 80 projects in pharo-contribution
Pharo Contribution

The Pharo-Contribution CI server contains projects that are actively maintained by the Pharo Community.

Please note that each Jenkins job should fulfill the following requirements:
- A project description
- A contact person, either in the description or in the email configuration of the job

Please see the example job template for more details on how to add a new job.

<table>
<thead>
<tr>
<th>Helper</th>
<th>NBOpenGL</th>
<th>Pharo-Kernel-2.0</th>
<th>Pharo-Kernel-3.0</th>
<th>Phobos</th>
<th>ProtectedSmalltalk</th>
<th>Raspberry Pi</th>
<th>Raspberry Pi-Experimentation</th>
<th>VM</th>
<th>Versions and Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pharo Contribution-linux64.ci.inria.fr</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-linux64-3.ci.inria.fr</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-linux64-4.ci.inria.fr</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-win7.ci.inria.fr</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-raspberry-pi-linux</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-raspberry-pi-experimentation</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-win7.ci.inria.fr</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
<tr>
<td>pharo-contribution-win7.ci.inria.fr</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
<td>Pharo-Source</td>
</tr>
</tbody>
</table>
Infrastructure: Misc

- [http://files.pharo.org](http://files.pharo.org)
- [http://get.pharo.org](http://get.pharo.org)
- SmalltalkHub: [http://smalltalkhub.com](http://smalltalkhub.com)
- 1411 users, >1200 repos
Small Stuff

- Lots of Cleanups
- Lots of tuning (performance, memory)
- Lots of small improvements
Lots of larger things

- Closure class now standard in Pharo3
- Terminal output for stderr
- Cleanup Source file related code
- AST Interpreter
- AST based Navigation in Browser

- Komitter
- Launcher
- Font
- Speedup
- ....
New Inspector

- Unify Inspector and Explorer
- Make specialised inspectors visible
Athens: Vector Graphics

- New API for Vector Graphics
- Independent of Backend
  - For now: Cairo
  - Balloon3D for Debugging
- Future: OpenGL
Athens: Demo
Opal Compiler

- Uses RB AST
- IRBuilder: Bytecode backend with high-level builder
- Much easier to change
- Basis for advanced Reflection
New ClassBuilder

- Replaces the old ClassBuilder
- Easier to understand and more flexible
- Basis for First Class Variables (Slots)
New Debugger

- Model now separate from View
- Model is scriptable
- Debugger is extensible with Commands
Command Line

denker$ ./pharo Pharo.image --list
Currently installed Command Line Handlers:
  Fuel          Loads fuel files
  config        Install Configurations
  save          Rename the image and changes file
  update        Load updates
  printVersion  Print image version
  st            Loads and executes .st source files
  test          A command line test runner
  clean         Run image cleanup
  eval          Directly evaluates one line scripts
A lot of change...
But just one iteration
Pharo4
Pharo4

- Again: To be released Spring 2015
- Already 175 updates
- 480 Issues closed
- Very stable
For example...

- Improved Refactorings
- 6MB Deployment Image
- ifTrue: on non-Booleans
- Browser and Tool cleanups
- Context Cleanup (MethodContext/ContextPart merge)
In Progress...
First Class Variables

- First class Instance Variables (Slots)
- First class globals + class variables
For what?

- Allows programmers to define behavior
- Easy reflection on variable access
  - Break on variable read, for example
Property Slots

Object

subclass: #PropertyObject
layout: PointerLayout
slots: {
    #ivar.
    #property1 => PropertySlot.
    #property2 => PropertySlot.
    ...
    #propertyN => PropertySlot.
}
Property Slots

Object
  subclass: #PropertyObject
  layout: PointerLayout
  instanceVariables: {
    #ivar.
    #property1 => PropertySlot.
    #property2 => PropertySlot.
    ...
    #propertyN => PropertySlot.
  }
Examples

- BitSlot
- BooleanSlot
- Alias
- Relationships (e.g. one-one, one-many)
- .... Your Domain level Slot!  ==> Magritte
Flexible Object Layouts
Enabling Lightweight Language Extensions by Intercepting Slot Access

Toon Verwaest Mircea Lungu
Oscar Nierstrasz
Software Composition Group, University of Bern, Switzerland
http://scg.unibe.ch

Camillo Bruni
RMoD, INRIA Lille - Nord Europe, France
http://rmod.lille.inria.fr

Abstract
Programming idioms, design patterns and application libraries often introduce cumbersome and repetitive boilerplate code to a software system. Language extensions and external DSLs (domain specific languages) are sometimes introduced to reduce the need for boilerplate code, but they

1. Introduction
Object-oriented programming languages (OOPL) are highly effective as modeling languages. Features including classes and inheritance can be used to model concepts at a high level of abstraction, normally leading to compact and precise code. Unfortunately, there are many situations in which
Advanced Reflection

- Partial Behavioral Reflection
- Associate MetaObject with structural object
  - Slots, Globals
  - AST nodes
The Meta Link

links

meta-object

activation condition

source code (AST)
Why?

- Change behaviour for selected AST Nodes or Variables
- “All variable reads”
- “this message send”

But without changing the program code!
Uses...

- Debugger
- BreakPoints, WatchPoints
- Profilers
- Coverage Analysis
- AOP
One File Pharo

- .sources, .changes, .image

- It is time to simplify that!
Epicea

- Replace .changes

- High level model:
  - aggregate changes (refactoring)
  - serialized to disk independent of source model
Epicea Log Browser

Event

- ExampleClass >> #fortyTwo
- ExampleClass >> #fortyTwo
- #fortyTwo --> #forty2
- ExampleClass >> #forty2
- ExampleClass >> #fortyThree

- 'I didn't like the name'
- "comment " trim
- EpEntryItem >> #displayWidget
- Snapshot: /Users/tinchodias/Downloads/Epicea/Epicea-1.image
- '312312321'
Sources

• It is 2014: Memory is cheap.

• Complexity is expensive

• Why not just put the sources in the image?
  • Externalize when needed (small devices)
  • Code history is in Monticello (or Git)
Bootstrap

- Create an image from a git repository
- Control what the image contains
- Easier to make changes
- Enforces Modularity
Bootstrap

• Working for Pharo3 as a prototype

• Can we even use this for Pharo4 on the build server?
And more...

- GT Tools
- VM related news
  - Spur, Sista, 64bit...
  - ............ there are lots of talks here
Questions ?