Pharo Status
(Fosdem15)
Marcus Denker
http://www.pharo.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”
A lot of change… (see FOSDEM14)
But just one iteration
The Platform

Jump too large
Nomadic Solution

- Do not build infrastructure
- Use resources until depleted
- Move on
The Platform

Jump Possible

Project1

Project2

Project3

Project4
Feedback Loop

Projects → Infrastructure
Every improvement has an effect
Even Trivial Changes
The only valid measurement of code quality: WTFs/minute

Good code.

Bad code.
Pharo4
Pharo4

- Again: To be released Spring 2015
- 470 updates
- >1200 Issues closed
- Very stable
Small stuff...

- Improved Refactorings
- 6MB Deployment Image
- ifTrue: on non-Booleans
- Browser and Tool cleanups
- Context Cleanup (MethodContext/ContextPart merge)
Dark Theme
GT Tools

- Start to replace all tools
- For now: Inspector + Spotter

http://gt.moosetechnology.org
Video: GTInspector

https://www.youtube.com/watch?v=wdYhM9byyPw
Demo: Inspect Methods
Demo: GTSpotter
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
In Progress...
Advanced Reflection

• Partial Behavioral Reflection

• Associate MetaObject with structural object
  • Slots, Globals
  • AST nodes
Why?

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

But without changing the program code!
Epicea

- Replace .changes

- High level model:
  - aggregate changes (refactoring)
  - serialized to disk independent of source model
A new GC: Spur

• New Object Layout
  • e.g. faster instanciation, larger hash
• Ephemerons, Fast Become:, Pinned Objects
• Double Performance
The Future
64 bit COG VM

- New Spur GC+Object layout has 64bit version
- 64bit in development
  - prototype starts to work in interpreter VM
  - 32bit images can be converted to 64bit
Sista: Runtime Optimizer

- Current JIT work per method
- Sista: detect hotspots, collect type info ==> Optimize
- Bytecode-2-Bytecode
- Optimizer implemented in Pharo Image
OSWindow+SDL2

- Move window logic to Smalltalk
- Backend uses libSDL2 via FFI
- Multiple OS Windows, OpenGL…
Block: Morphic, redone
Woden 3D

https://www.youtube.com/watch?v=1Nze9tnwYxY
Virtual GPU

- The VirtualGPU is a simple DSL/framework for parallel algorithms.

https://www.youtube.com/watch?v=6xrq4OkvSi4
Too much to show...

- Twitter: @pharoproject
- reddit Smalltalk sub-reddit
- Google+
Questions ?