Pharo 10 Status

Pharo Consortium Meeting

consortium-adm@pharo.org

09 December 2021

Pharo Days 2022

https://days.pharo.org

March 3rd and 4th 2021

Lille, France

Pharo 10
Ready To Launch!

Pharo 10
Ready To Launch!

Final Details!

Pharo 10
Ready To Launch!

After PharoDays

Pharo 10

Objectives

- Smaller Iteration

- Simplification

- Tooling

- Modularisation

- General Improvements

- VM Improvements

Pharo 10
Smaller Iteration

- Improving the Development process

- Shorter iteration to release

- Reduced Set of Objectives

- Better "Ready" definition

Pharo 10
Smaller Iteration

- Improving the Development process

- Shorter iteration to release

- Reduced Set of Objectives

- Better "Ready" definition

A result of a big introspection

Pharo 10

Simplification

- Removing duplicated / obsolete code

  - Old Tools

  - V3 Compiler Support

  - Old Blocks & Bytecode

  - VM based event handling

  - Glamour

  - GTTools

  - …

Pharo 10

Simplification

- Removing duplicated / obsolete code

  - Old Tools

  - V3 Compiler Support

  - Old Blocks & Bytecode

  - VM based event handling

  - Glamour

  - GTTools

  - …

10% Code Reduction

Keeping a good single one

- 48K LOC

Spec 2
Main Elements

- Core & Basic Layouts

- Basic Presenters

- Application Support

- Styles / Themes

- Code Presenter

Spec 2
Extended Features

- Different Layouts and Composition

- Extended support for Dynamic Layouts

- New Dialog Building

- Transmissions

- Direct Support for Roassal & Cairo

- Multiple Backends (GTK / Morphic)

- Spec Tests and Testing Support

Tooling

- Migrating final tools to Spec

- Improving existing ones

- Fixing issues and glitches

- Improving Refactorings, Deprecator & Rewriting tools.

- Improving Profilers

Tooling

- Migrating final tools to Spec

- Improving existing ones

- Fixing issues and glitches

- Improving Refactorings, Deprecator & Rewriting tools.

- Improving Profilers

In Tools we believe

Fluid Class Syntax

```
TestCase << #AIGraphReducerTest
    slots: { #graphReducer };
    tag: 'Tests';
    package: 'AI-Algorithms-Graph-Tests'
```

```
Trait << #TSetArithmetic
    traits: {};
    slots: {};
    tag: 'Traits';
    package: 'Collections-Abstract-Tests'
```

```
TestCase << #AIGraphReducerTest
    layout: FixedLayout;
    traits: {};
    slots: { #graphReducer };
    sharedVariables: {};
    sharedPools: {};
    tag: 'Tests';
    package: 'AI-Algorithms-Graph-Tests'
```

Fluid Class Syntax

```
TestCase << #AIGraphReducerTest
    slots: { #graphReducer };
    tag: 'Tests';
    package: 'AI-Algorithms-Graph-Tests'
```

```
Trait << #TSetArithmetic
    traits: {};
    slots: {};
    tag: 'Traits';
    package: 'Collections-Abstract-Tests'
```

```
TestCase << #AIGraphReducerTest
    layout: FixedLayout;
    traits: {};
    slots: { #graphReducer };
    sharedVariables: {};
    sharedPools: {};
    tag: 'Tests';
    package: 'AI-Algorithms-Graph-Tests'
```

Clear Syntax

Clean Implementation

Compiler Improvements

- Unifying objects variables into a single Hierarchy

- Improved Semantic Analysis

  - use Class and the Environment to lookup the variables

  - use Variable Hierarchy to model variables for name analysis.

  - Improved AST Visitor

- Pragma lookup speed-up

- Compiler Speed Improvements

Refactorings

- New Refactorings

  - Extract setUp method

  - Remove senders of method refactoring

  - Copy package as refactoring

  - Rename package (rename manifest)

  - Merge instance var x in y

  - Move to class side method

  - Create accessors with lazy initialization

Refactorings

- Improved ones

  - Deprecate method (simple version)

  - Deprecate class

  - Extract method refactoring

  - Replace senders by another

  - Rename vars in Traits

  - Convert temporary to instance variable

  - Push up method refactoring

  - Add access to pushUp and pushDown refactorings from source code

  - Permute parameters when add an argument

  - Abstract instance variable (now applies abstract methods when there is a reference to more than one variable in a method)

Other Improvements

- Sista Bytecodes w/ Full Block Closures

- Memory Management Configuration

- Integration with Windows

- Zinc

Lots of Bugs Fixed & Improvements

- + 1560 Pull Requests

- + 1100 Issues Closed

- 85 Different Contributors

Pharo 10

VM Improvements

- Sockets

- Clean up old code

- GC Improvements

- Logging

- Stability

- Speed

- Updated Dependencies

Pharo 10
VM Improvements

- 3 Operating Systems (OSX / Linux / Windows)

- 3 Architectures (ARM64 / ARM 32 / x86_64)

- Full Linux Packages through OBS

- Better FFI

Lots of Bugs Fixed & Improvements

- + 1560 Pull Requests

- + 1100 Issues Closed

- 85 Different Contributors

A great
Community!!

Pharo 10
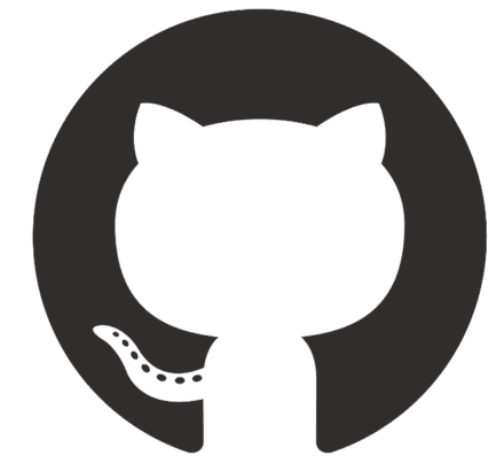## *Ready To Launch!*

After PharoDays

Thanks!!!

pharo.org

consortium

consortium-adm@pharo.org

pharo-project/pharo

discord.gg/QewZMZa

[DEV]

thepharo.dev