

Pharo 11: 15 years and a shining future

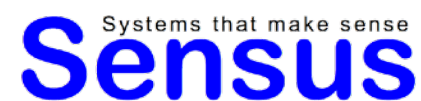
<http://stephane.ducasse.free.fr>

<http://www.pharo.org>

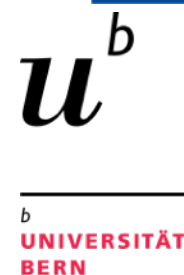




Yesplan
Let's make it happen



Accelerate Confidently



A word of presentation



Expert in language design

**Traits in Php7.0, Perl60
Influenced Traits in Scala**

Won the ECOOP'23 “Test of Time” award

Expert in maintenance

**Program analysis, metamodels,
visualization, testing, refactorings,
metrics,....**

...

- ...1996: LISP and reflective stuff
- 1994: discovered Smalltalk
- 1998: discovered Squeak
- ~2000: harvester for 3.xx, 3.6, 3.7, started to clean Squeak Responsible for Squeak3.9
- 2008: Pharo :)

Talk Outline

- Vision
- Goals
- (marketing thought)
- P11 and previous remarkable achievements :)
- Join and have fun with us

Pharo's Vision

- *High quality* and *robust* language and runtime to support large and complex software
- Support *business* growth
- *Modular* and *versatile* (xGb image to 200k)
- *Evolving* (revisit and improve Smalltalk)

Pharo's goals...

The background image shows a large, complex scientific instrument, likely a particle accelerator or fusion reactor, with a blue and yellow color scheme. It features a large, cylindrical central component surrounded by various pipes, cables, and structural elements. The overall scene is industrial and technical.

The best way to predict the future is to have

**A powerful engine
to (re)invent (y)our future**



Build/support/sustain

**An ecosystem where
innovation/business bloom**

... that ****you**** can make money with Pharo.

... to be able to invent solutions to existing problems.

... a powerful **innovative dynamic** language where we can build (y)**our** future.



**Super powerful live
programming language and
tools**

A(n eco-)system
that can **evolve**

**Time travel...
Why Pharo?**

**back to Smalltalk 2008 at Buenos
Aires**



I have a dream...

S. Ducasse stephane.ducasse@free.fr




INRIA & STIC Amsud



Let's stop to live in a
museum!

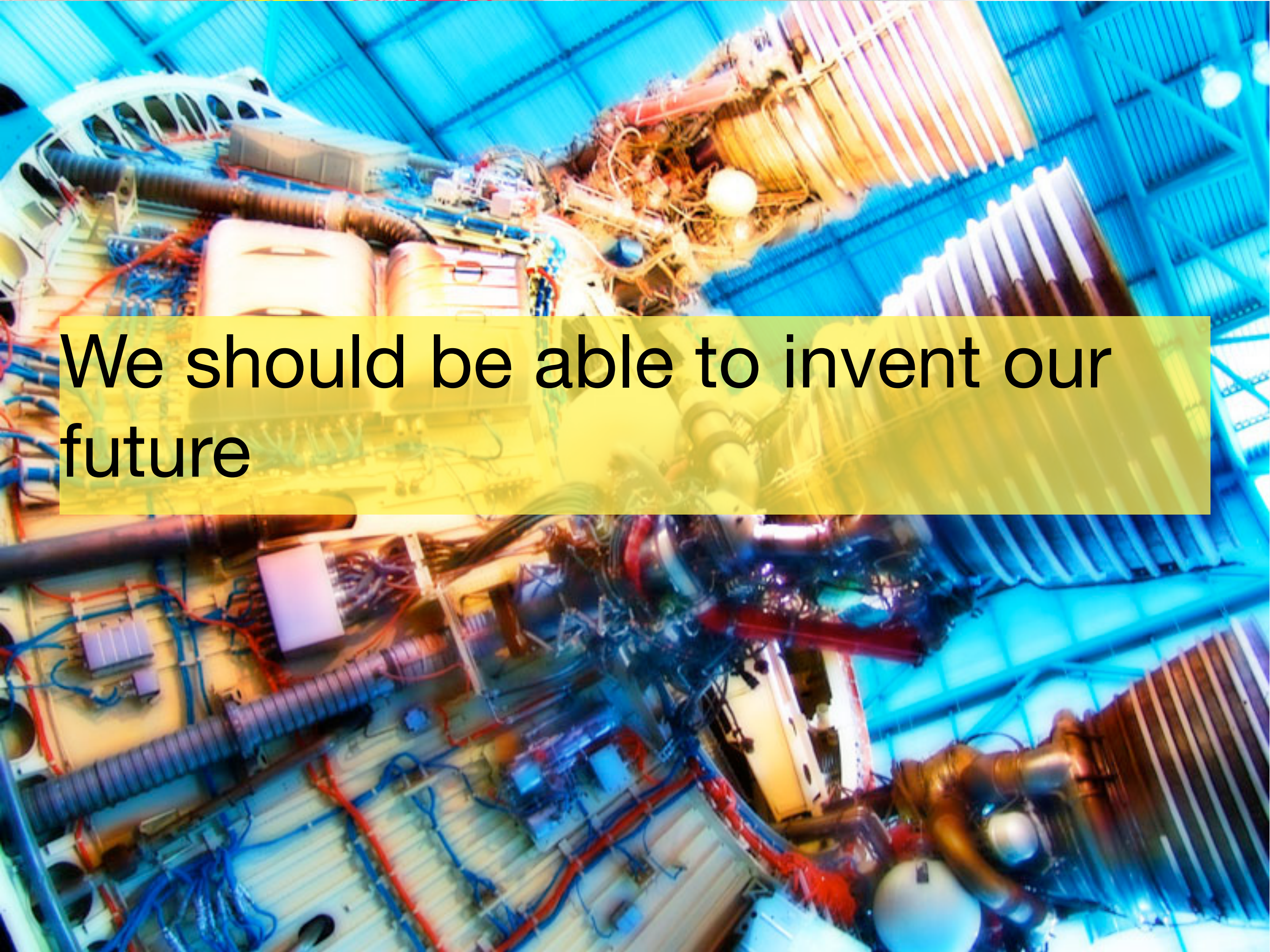




When do we ***really*** build on top of giant shoulders?

Thanks Dan and Alan. Now...

I want *more*!



We should be able to invent our future



When do we make sure that we could invent Smalltalk 3000?

first class instance variables, modules?, immutability bits?, visibility?,
image as cache?

The background is a light gray with a subtle, textured pattern. In the upper left quadrant, there is a cluster of colorful splatters in yellow, pink, and blue. A faint, light gray world map is visible in the lower half of the image, centered around the equator.

We need a new state of mind

Smalltalk as the language to define other languages

Advanced and stronger Mops

We deserve more than doesNotUnderstand:

New ways of defining languages

Domain specific language support

New generation of meta compilers

Software Engineering

Tests: which ones should I fix first?



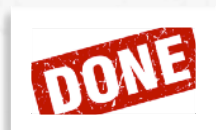
Continuous test running



Smoother refactorings and code manipulation



Continuous integration / Extreme integration



Better Lint Rules / Domain specific lint rules



Bug tracking

UI

DONE

Multitouch support

New UI

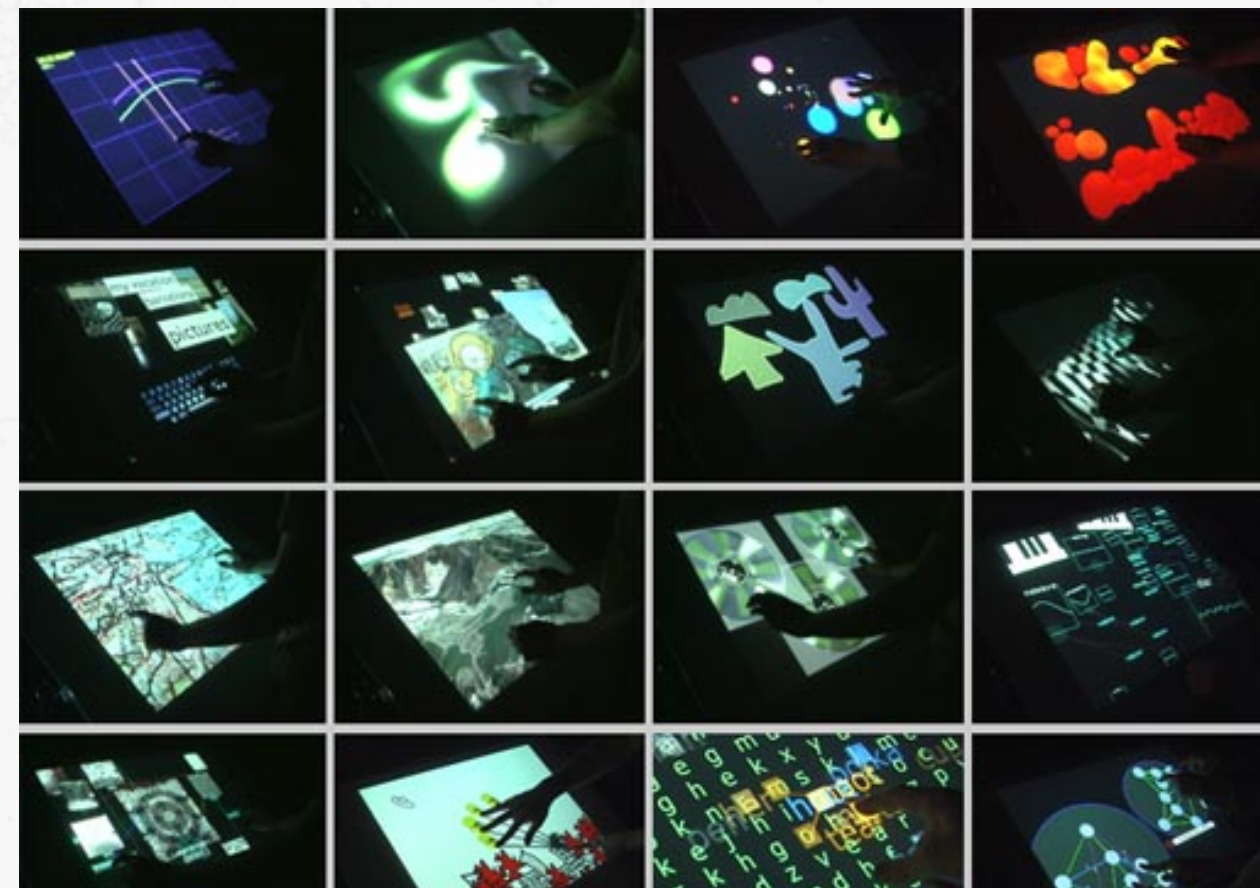
When will we be able to reinvent UI

DONE


Cairo and others?

DONE

New IDEs



Back to present :)



**An ecosystem where
innovation/business bloom**

trentosur

Soluciones móviles para retail y trade marketing

Nos enfocamos en lo que importa del negocio sin perder de vista los detalles de su implementación.

Primeros móviles, Plataforma Android, En la nube

PharoCloud

Overview Pricing Blog Login Sign Up

Pharo platform as a Service: put your Smalltalk web-application online at Pharocloud in just 3 clicks

Try it for FREE Watch how it works

Romax Technology

Wind Energy

Pioneering new ways of maximising sustainable wind energy yields. Our products and services optimise asset availability, wind turbine performance and drivetrain reliability. We work with owners, operators, manufacturers, insurers and service providers worldwide.

Get in touch

Related links

Wind Energy

WEBDRUCK.CH

Web-To-Print Solution

- Design and create individual printed matter
- eShop with credit card payment
- High quality PDF output with Printing Process integration
- Thousands of orders for seven Swiss printing companies

Quuve

Quuve is a powerful business intelligence tool that provides a comprehensive view of your company's performance. It features a wide range of charts, graphs, and tables that can be customized to suit your needs.

Some Success Stories

<http://pharo.org/success>

Background images include: iBizLog, Pinesoft MBagger, NTed: disaster scenario recovery, and various charts and graphs.

Dedicated and cost-effective tools for software evolution

Dedicated Analyses, Dedicated tools, Decision making

MOORE'S LAW

The world's first online platform fully supporting risk-based test management.

BETTER, FASTER, CHEAPER

COMPONENTS

Yesplan

Yesplan is veelzijdige software voor het efficiënt plannen van evenementen. Yesplan is uiterst gebruiksvriendelijk, flexibel en makkelijk te koppelen met andere software.

Event planning software interface

OBJECT PROFILE

Left to run / Right for code

Visual programming interface with various charts and graphs.

Das Content Management mit System

100% Inline-Editor

Drag & Drop, Copy / Paste

Content management system interface

2denker

Continuous API Testing keep your services under control 24/7

www.2denker.de

API testing interface with a map and various charts.

**if you have a 'success' story
please send it to us!**

Why Pharo?

**(An intermezzo
A marketing thought...)**

**Do you really remember
1980?**



1980...

1983



1983

1981



1981

“Smalltalk-80”
is so ‘sexy’

Yes let us scare the kids and
the managers

**We like smalltalk but
we love Pharo because
Pharo is not Pharo 1980 :)**

Pharo is Pharo!

Some Pharo aspects

- Bootstrap
 - Tests, tests, tests
 - Permanent space
- First class instance variables
 - Graphics back-ends
- Large libraries
 - Industrial consortium
 - Traits
- Real Git integration
- Automatic rewriting deprecation
- Ephemerons for real
 - Vectorial canvas
- Wonderful community
 - Books
- 11 platforms: ARM, X86, M1, RiscV

Places for improvements

- More database drivers
- Modern UI (HDPI + new 'Morphic')
 - More modular
 - Dynamic inliner
- Better Windows support

(I have tons of cool ideas, if you have 1 M\$ talk to me)

Pharo 11: a stabilization iteration

- No big bang
- More fixes, more tests
- More documentation
- Better compiler
- Better VM

(...)

We do maintain a LARGE code base and we do help people with old versions (recently we helped company with Pharo 7.0)!

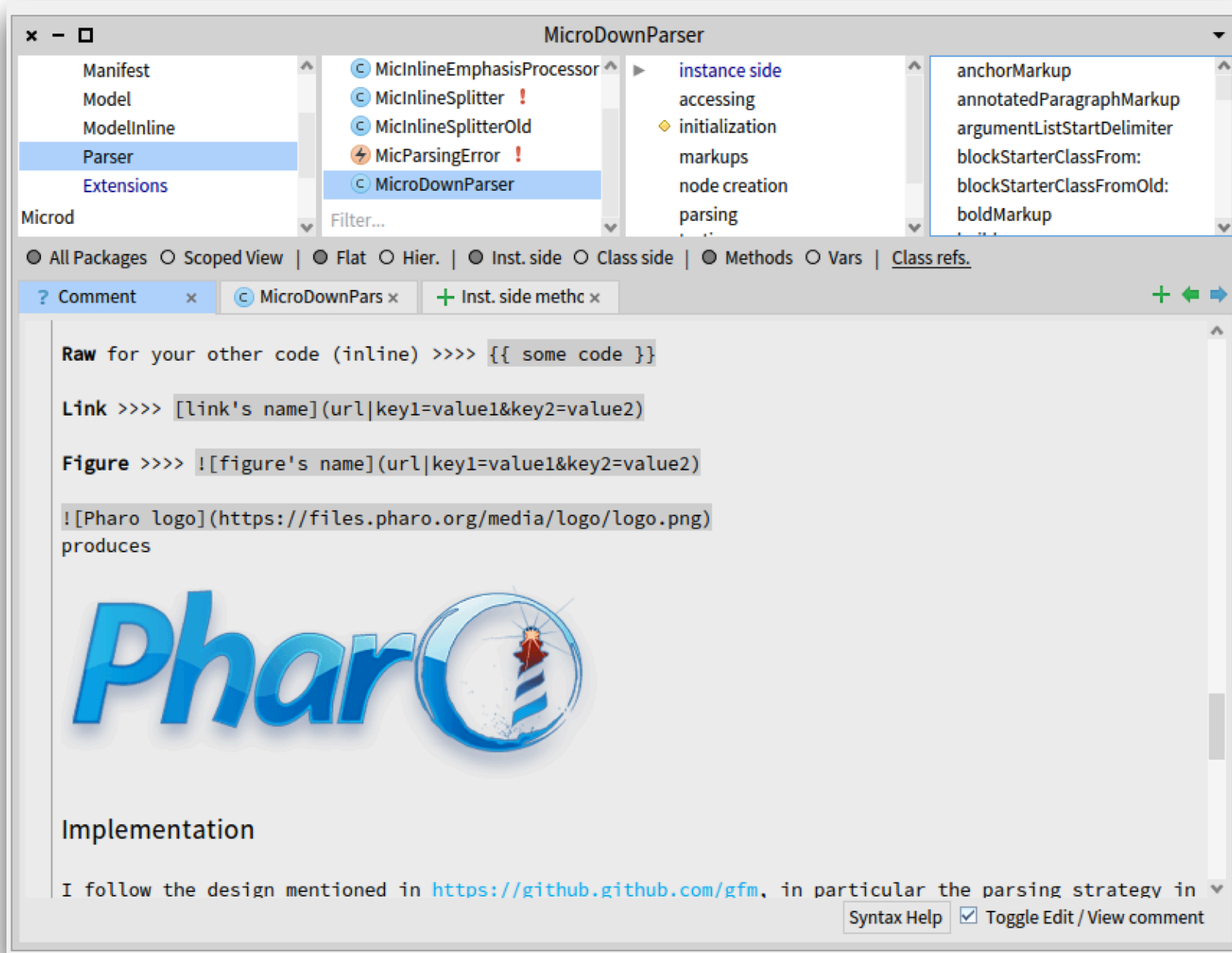
Pharo 11: Tools

- Iceberg/Git fix and adaptation to github changes!
- Better debugger
- Another step in adopting Fluid class definition
- Rewrite tools + better refactorings
- Better new tools
- DocumentationBrowser

Pharo 11: Documentation

- One format: microdown to rule them all
 - Github markdown 'compliant'
 - Class comments / Class comment templates
 - Documentation
 - Books

Rendering of Class and Package Comments



The screenshot shows a web browser window displaying the rendered HTML for the `MicroDownParser` class comment. The browser's address bar shows the URL `https://github.com/gfm/gfm`. The page title is `MicroDownParser`. The left sidebar shows a tree view of the `MicroDownParser` class, with `Parser` selected. The main content area shows the rendered HTML for the `MicroDownParser` class comment, which includes a `Raw` block, a `Link` block, a `Figure` block, and an `Implementation` section. The `Figure` block contains a `Pharo` logo. The `Implementation` section contains a paragraph of text.

Manifest
Model
ModelInline
Parser
Extensions
Microd

Filter...

instance side
accessing
initialization
markups
node creation
parsing

anchorMarkup
annotatedParagraphMarkup
argumentListStartDelimiter
blockStarterClassFrom:
blockStarterClassFromOld:
boldMarkup

All Packages | Scoped View | Flat | Hier. | Inst. side | Class side | Methods | Vars | Class refs.


? Comment x MicroDownPars x + Inst. side methc x

Raw for your other code (inline) >>> `{{ some code }}`

Link >>> `[link's name](url|key1=value1&key2=value2)`

Figure >>> `![figure's name](url|key1=value1&key2=value2)`

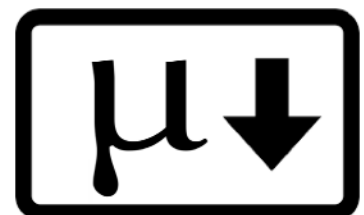
`![Pharo logo](https://files.pharo.org/media/logo/logo.png)`
produces



Implementation

I follow the design mentioned in <https://github.com/gfm/gfm>, in particular the parsing strategy in

Syntax Help ☒ Toggle Edit / View comment



Microdown

Manifest

Model

Microdown

MicSurfacicMicrodownToPillarTest

MicToPillarBasicTest

MicCodeBlockTest

Filter...

accessing

running

tests - anchor

tests - codeblock

codeBlockClass

factory

factory:

headerClass

All Packages Scoped View Flat Hier. Inst. side Class side Methods Vars Class refs.

Comment

*MicSurfacicMic

setUp

*visitHeader:

+ Inst. side methc

MicSurfacicMicrodownToPillarTest

Description

This test case uses the microdownSnippetFactory and test that the conversion to Pillar object trees is correct. This is why it is in this package

Microdown text -> Microdown trees -> Pillar trees

Te tests are just checking that object of the correct class is created. Future extensions should handle the details.

Tests

This test suite defines 56 test methods.

Locally defined tests are:

- MicSurfacicMicrodownToPillarTest>>#testSuperscriptFormatEmpty
- MicSurfacicMicrodownToPillarTest>>#testLineEnd
- MicSurfacicMicrodownToPillarTest>>#testScriptWithNewLine
- MicSurfacicMicrodownToPillarTest>>#testAnchorWithNewLine
- MicSurfacicMicrodownToPillarTest>>#testItalicFormatEmpty
- MicSurfacicMicrodownToPillarTest>>#testScriptParametersMultiple
- MicSurfacicMicrodownToPillarTest>>#testScriptParameterValue
- MicSurfacicMicrodownToPillarTest>>#testAnchorWithSpaceInside
- MicSurfacicMicrodownToPillarTest>>#testScriptParameter
- MicSurfacicMicrodownToPillarTest>>#testHeaderLevel3
- MicSurfacicMicrodownToPillarTest>>#testSubscriptFormat
- MicSurfacicMicrodownToPillarTest>>#testScriptTwoParametersNoValue


```
addItem: [ :item | item name: '3: ', loremIpsumWords atRandom ] ];  
yourself ].
```

^ presenter open

Factory method

You can use `SpMenuButtonPresenter` in your presenters by sending `SpPresenter>>#newMenuButton`.

Examples

- `SpMenuButtonPresenter class>>#example`

API Methods

- `SpMenuButtonPresenter>>#menu`
- `SpMenuButtonPresenter>>#menu:`

Events

- `SpMenuButtonPresenter>>#whenMenuChangedDo:`

Hierarchy

```
SpAbstractPresenter  
└─ SpPresenter  
   └─ SpAbstractWidgetPresenter  
      └─ SpAbstractButtonPresenter  
         └─ SpMenuButtonPresenter (this is me)
```




Fluid Class Syntax

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

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

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




Fluid Class Syntax

- Was sketched and presented in 2017 at ESUG
- First release in P10 (took longer than we wanted)
 - Nice design
 - Scale well with multiple and optional parameters
 - Extensible
 - Clean and *nice* implementation
- P11: the default Pharo syntax!
- P11: prepare to fade away of other class definitions

Pharo 11: Slots

- First class instance variables
- Appeared in P9
- P11: New iteration

Pharo 11: System

- Constant clean blocks (faster blocks :)
- More inlining and optimizations
- Ephemeron finalization production ready
 - Better VM support - killed vicious bugs
 - Image level takes advantages of them

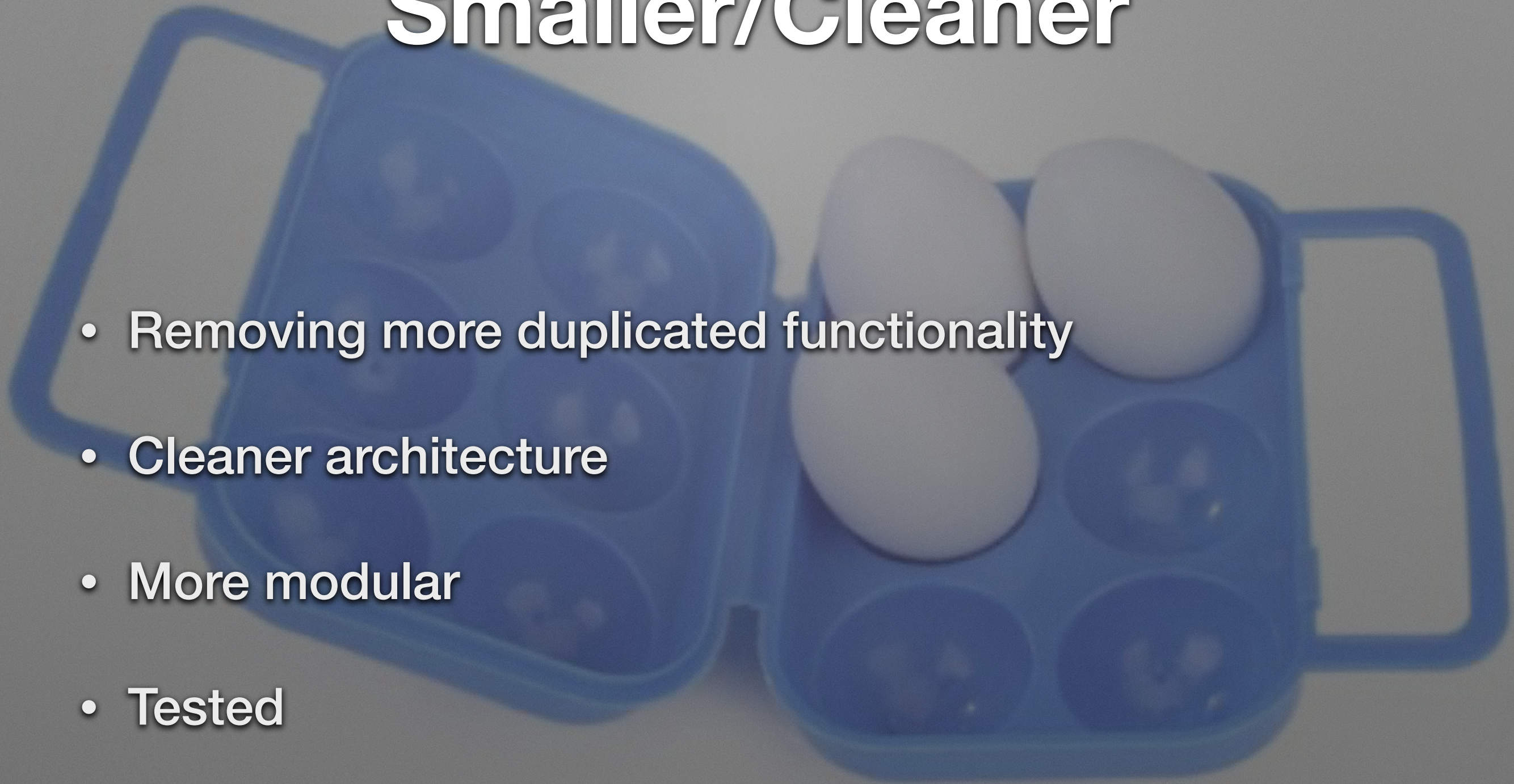
Pharo 11: VM

- GC bug fixes
- Permanent space (sponsored by Lifeware)
- Initial support for Single-Instruction Multiple-Data (SIMD)
- Third-Party Dependency Update (newer versions, Graphic Libraries using Hardware Acceleration)
- Clean Ups: Remove lots of old code, notably old experiments, and dead code

Realizing Vision for Pharo

Smaller/Cleaner

- Removing more duplicated functionality
- Cleaner architecture
- More modular
- Tested



Better tools

- Next generation debuggers
- Customisable

VM

- More tests / better build system
- VM under our control
- Threaded FFI for real
- P10: getting rid of the polling event loop
- P11: GC + permanent space
- Working on a AoT compiler
 - Nice ideas for P12 :)



Headless: true server side

- For real, no UI when no need it
- SDL back-end
- Embeddable

New graphics for real

- SDL 20 for events (no more hidden in the VM)
- Failed to deploy HDPI (solution was only for morphic)
- On the back burner
 - Bloc as a new bottom layer
 - Toplo new widget sets
 - Native windows (nearly done in P12 alpha)

Spec20: a Cornerstone!

- Large reimplementation from Spec1 to Spec2
- New widgets
- New layouts
- Multiple back ends (Morphic, GTK30, *Toplo*)

The trajectory

- Converting all existing tools to use Spec
- Support mix of back-ends
- Have backends (GTK, Morphic, Toplo) for Spec
- Remove Morphic and use Bloc/Toplo

Browser

▼ Morph

▶ BorderedMorph

HandMorph

▶ AbstractResizerMorph

AnimatedImageMorph

BracketMorph

▶ FTSelectableMorph

FTTableContainerMorph

halo:

copyToPasteBuffer:

savePatchFrom:

balloonHelp:

generateMouseEvent:

obtainHalo:

sendMouseEvent:

eventListeners:

moveToEvent:

obtainHalo: aHalo

"Used for transferring halos between hands"

self halo == aHalo

ifTrue: [^ self].

"Find former owner"

self world hands detect: [:hand | hand halo == aHalo] ifFound: [:formerOwner | formerOwner

releaseHalo: aHalo].

self halo: aHalo



Browser

▼ Morph

▶ BorderedMorph

HandMorph

▶ AbstractResizerMorph

AnimatedImageMorph

BracketMorph

▶ FTSelectableMorph

FTTableContainerMorph

▶ FTTableMorph

FullscreenMorph

halo:

copyToPasteBuffer:

savePatchFrom:

balloonHelp:

generateMouseEvent:

obtainHalo:

sendMouseEvent:

eventListeners:

moveToEvent:

mouseFocus

sendEvent:focus:

obtainHalo: aHalo

"Used for transferring halos between hands"

self halo == aHalo

ifTrue: [^ self].

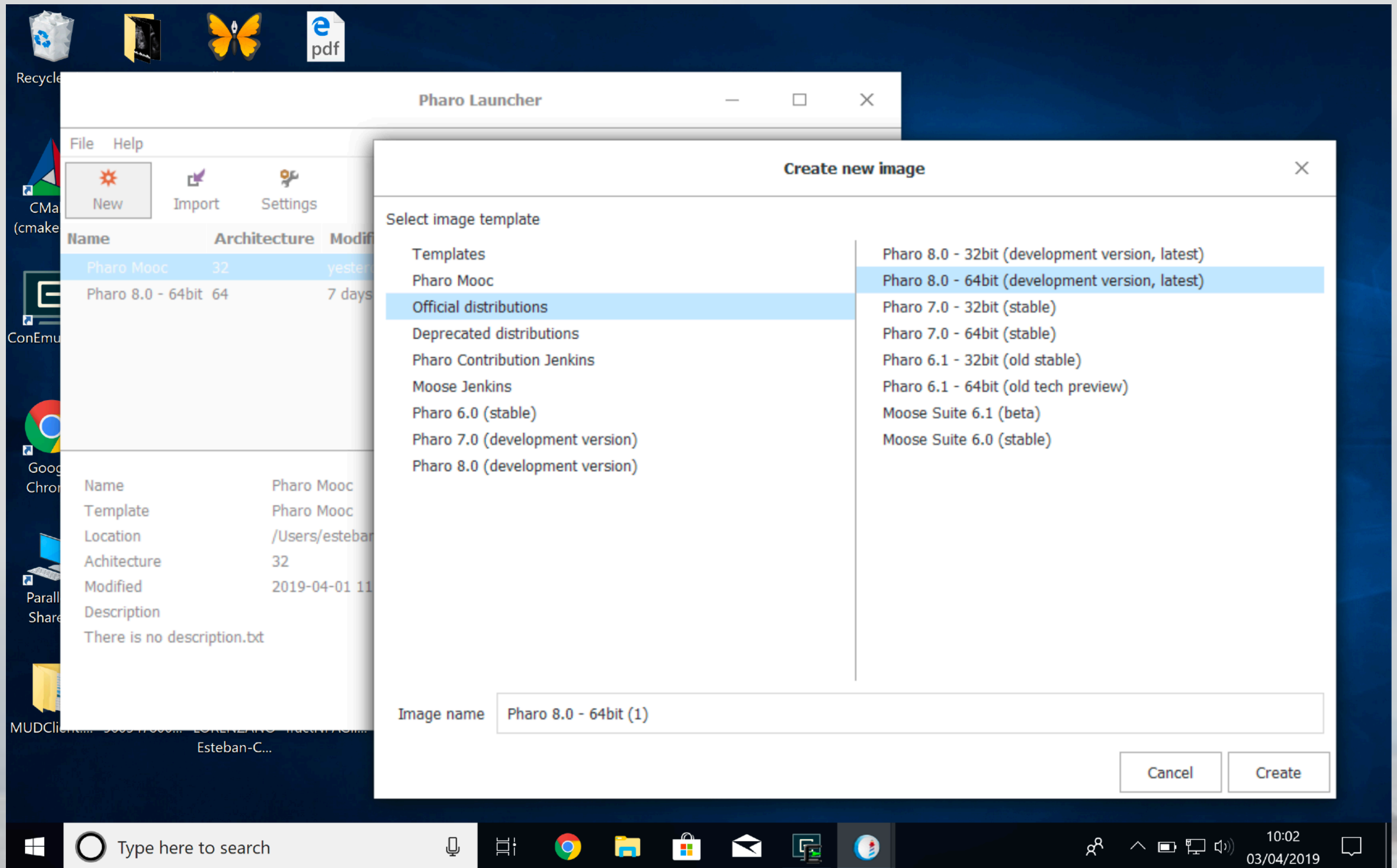
"Find former owner"

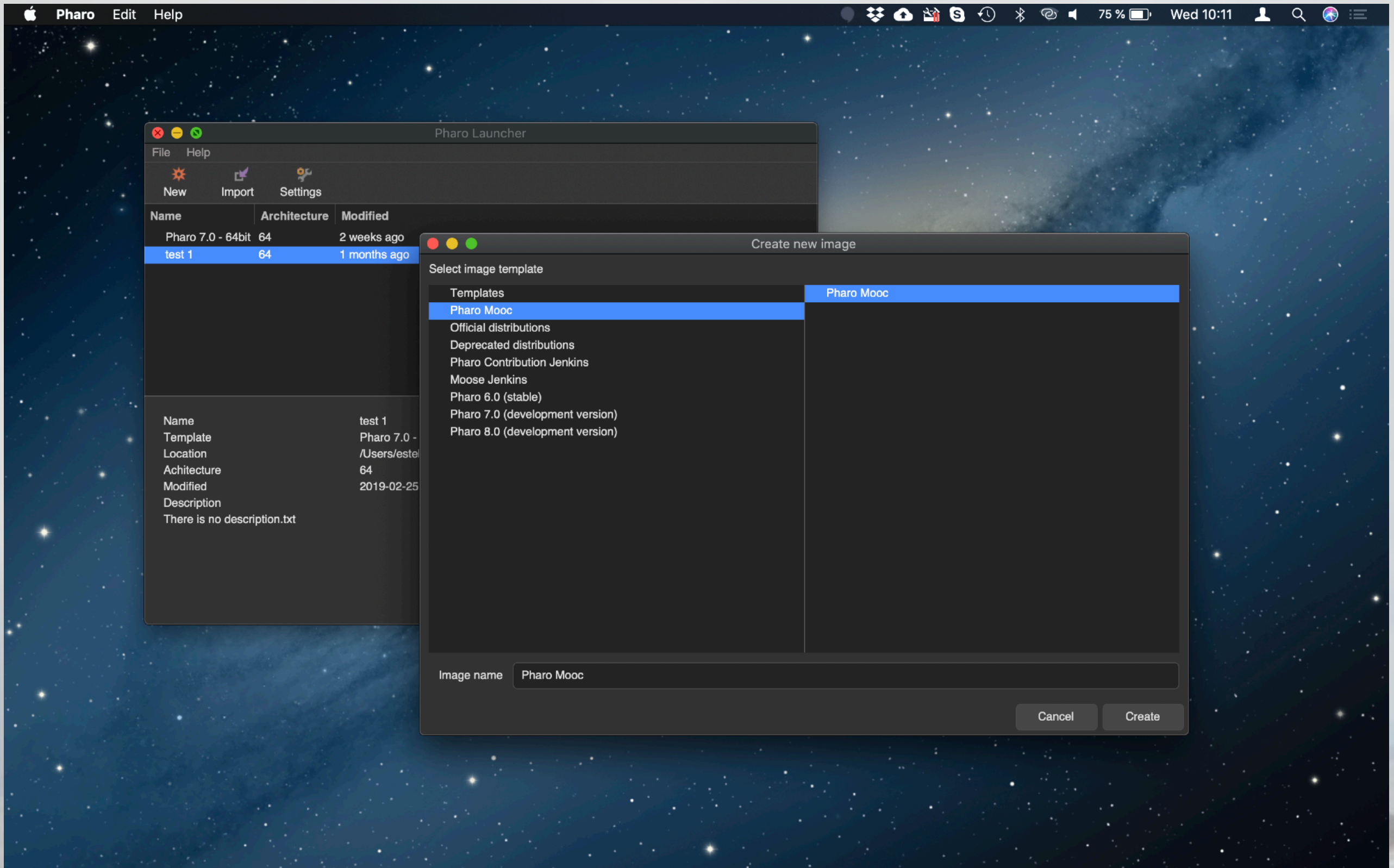
self world hands detect: [:hand | hand halo == aHalo] ifFound:

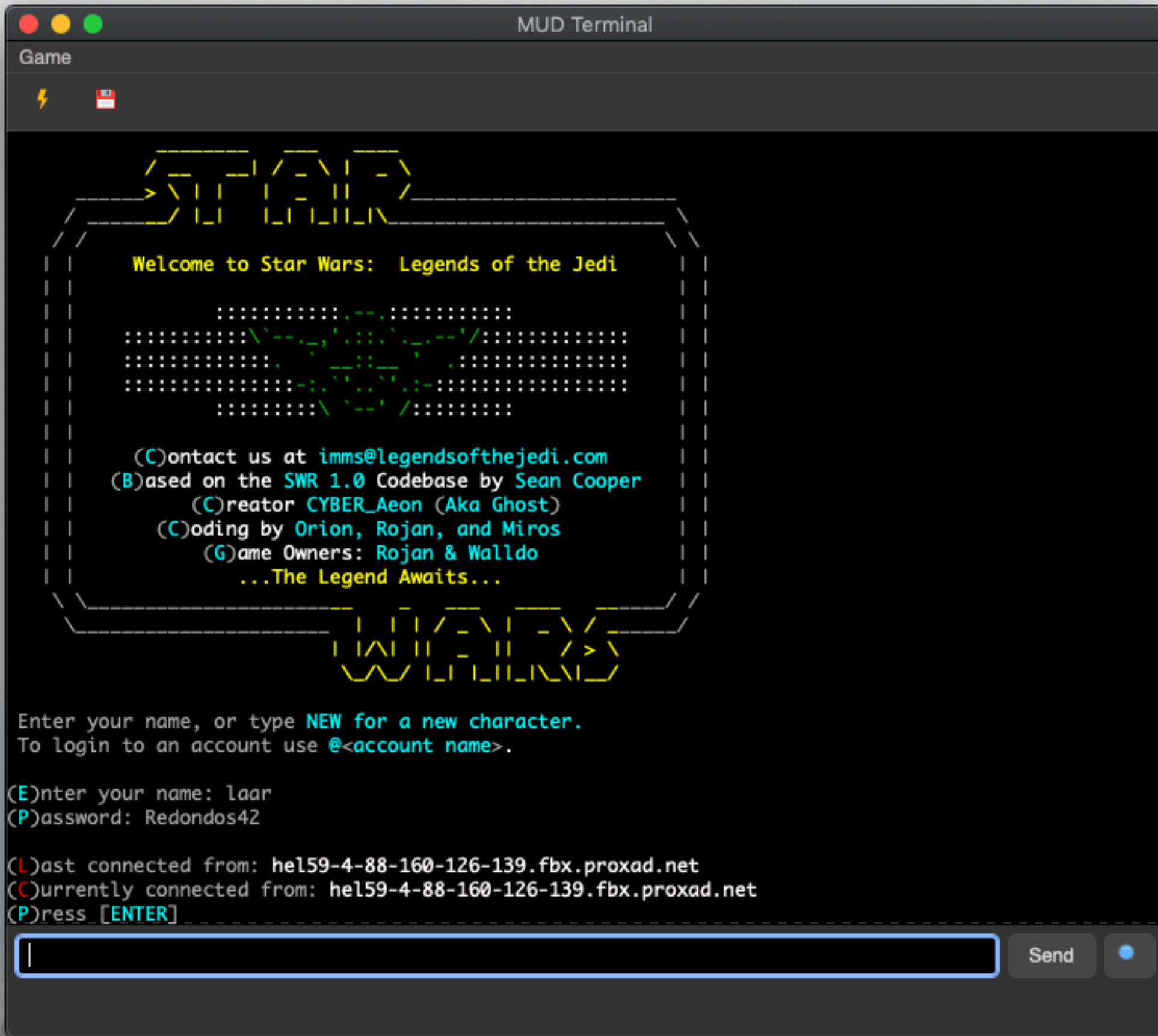
[:formerOwner | formerOwner releaseHalo: aHalo].

self halo: aHalo









Testing Spec

- Many tests
- Parameterized (same for Morphic and GTK30 and soon for TOPLO)



Presenters

Layouts

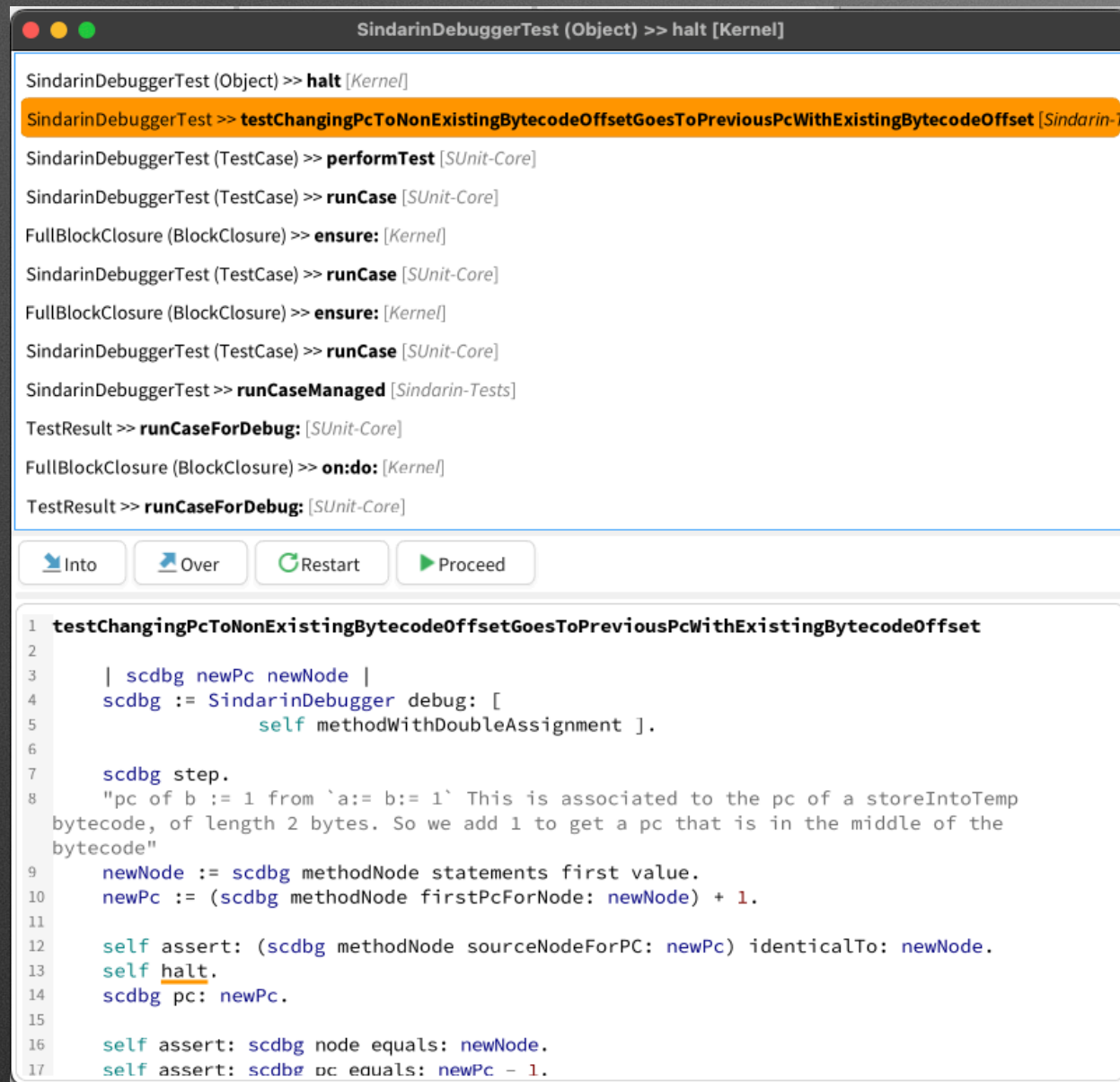
Widgets

Adapters

Toplo

- New widget library on TOP of bLOck
- Sponsored by Thales (should deploy products in 2023)
- Skins will be based on <https://ant.design/>
- Plan to have a Spec back-end soon
- Currently
 - All widgets except Tree/Table
 - Skin first iteration

Toplo



The screenshot displays the Toplo IDE interface. At the top, a window titled "SindarinDebuggerTest (Object) >> halt [Kernel]" contains a list of debugger actions. Below this is a toolbar with buttons for "Into", "Over", "Restart", and "Proceed". The main area is a code editor showing a Scala test function.

```
SindarinDebuggerTest (Object) >> halt [Kernel]
SindarinDebuggerTest >> testChangingPcToNonExistingBytecodeOffsetGoesToPreviousPcWithExistingBytecodeOffset [Sindarin-]
SindarinDebuggerTest (TestCase) >> performTest [SUnit-Core]
SindarinDebuggerTest (TestCase) >> runCase [SUnit-Core]
FullBlockClosure (BlockClosure) >> ensure: [Kernel]
SindarinDebuggerTest (TestCase) >> runCase [SUnit-Core]
FullBlockClosure (BlockClosure) >> ensure: [Kernel]
SindarinDebuggerTest (TestCase) >> runCase [SUnit-Core]
SindarinDebuggerTest >> runCaseManaged [Sindarin-Tests]
TestResult >> runCaseForDebug: [SUnit-Core]
FullBlockClosure (BlockClosure) >> on:do: [Kernel]
TestResult >> runCaseForDebug: [SUnit-Core]
```

Into Over Restart Proceed

```
1 testChangingPcToNonExistingBytecodeOffsetGoesToPreviousPcWithExistingBytecodeOffset
2
3   | scdbg newPc newNode |
4   scdbg := SindarinDebugger debug: [
5     self methodWithDoubleAssignment ].
6
7   scdbg step.
8   "pc of b := 1 from `a:= b:= 1` This is associated to the pc of a storeIntoTemp
   bytecode, of length 2 bytes. So we add 1 to get a pc that is in the middle of the
   bytecode"
9   newNode := scdbg methodNode statements first value.
10  newPc := (scdbg methodNode firstPcForNode: newNode) + 1.
11
12  self assert: (scdbg methodNode sourceNodeForPC: newPc) identicalTo: newNode.
13  self halt.
14  scdbg pc: newPc.
15
16  self assert: scdbg node equals: newNode.
17  self assert: scdbg pc equals: newPc - 1.
```


Login:
Password:

It can be used and subclassed to support comment and code editing, workspace implementation...
8 All the different kind of Album using should use the available basic client class or subclass it for specific purpose.
9 Here an example of a client for a method.

```
text model menu |
  text := self methodText asRopedText.
  model := ToAlbumModel new.
  model styler: (BlRBTextStyler new classOrMet
  model text: text copy.
17 model withSaveCapability.
18 model withRowNumbers.
19 model whenSaveRequestedDo: [ :saveRequested
```

File

Radio button icon first

☐ Right
☒ Center
☐ Left

Radio button label first

Right ☐
Center ☒
Left ☐

Radio buttons label first and justified

Right ☐
Center ☒
Left ☐

Yourname

Cheesecake ☐
Cheesecake ☒
Cheesecake ☐
☐ Cheesecake
☐ Cheesecake

Saving stuffs

Save All

Mini browser

Tools-CodeNavigation-Tests
Tools-Tests
Toplo
Toplo-IDE
Toplo-LookAndFeel
Toplo-LookAndFeel-Tests
Toplo-Tests
TraitsV2
TraitsV2-Compatibility
TraitsV2-Tests

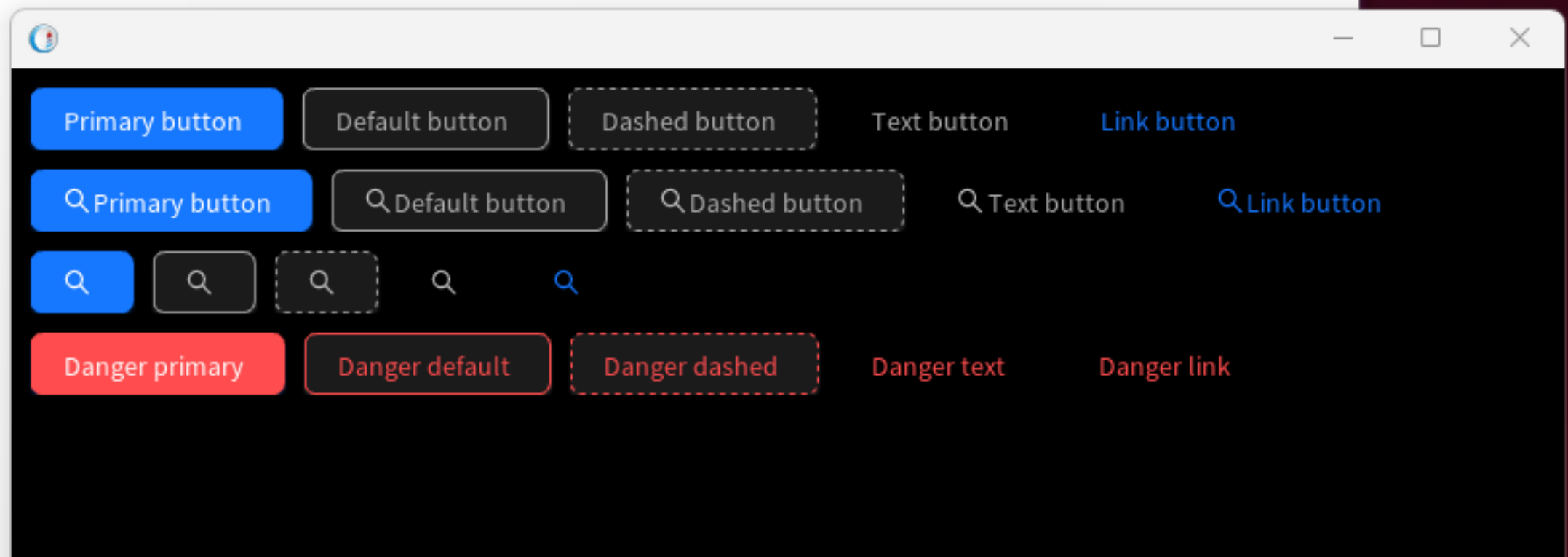
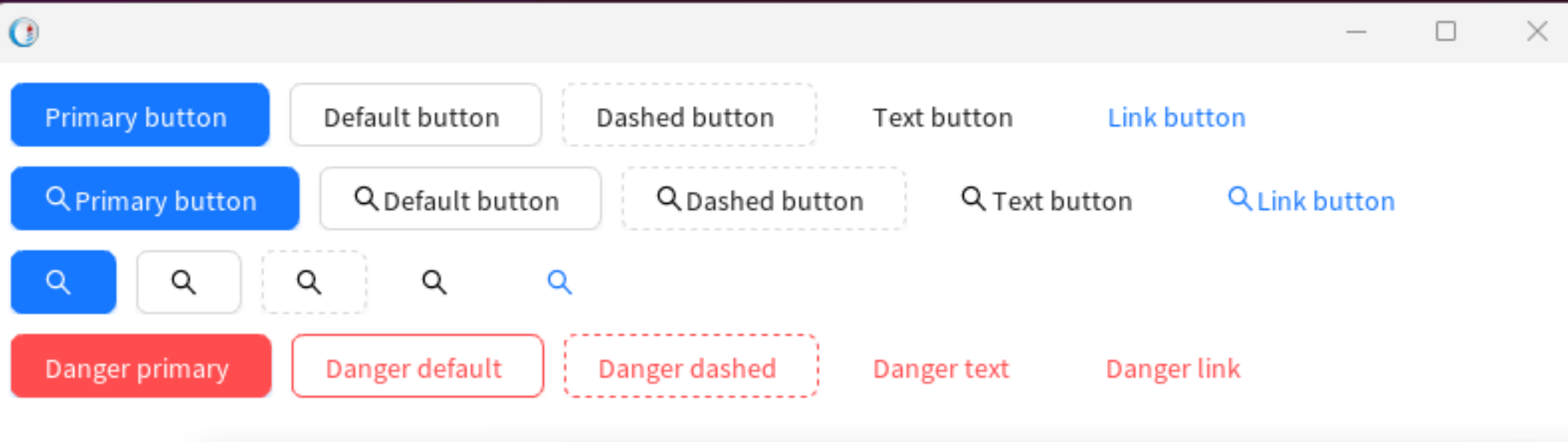
Class side ☒

! ToBottomRightResizeGrip
! ToButton
! ToButtonClickHandler
! ToButtonDresser
! ToButtonMenuItem
! ToButtonMenuItemDresser
! ToButtonModel
! ToCheckBoxDresser
! ToCheckMenuItem

-- all --
t - change hook
t - initialization dresser
t - labeled icon

defaultDresser
preInitializeDresser

```
1 preInitializeDresser
2
3 super preInitializeDresser.
4
5 self initializePair
```

Since may 2008

Pharo 11 - May 2023 - 70 committers (1412 closed)

Several subsets: (iceberg, microdown,...)

Pharo 7.0 - Feb 2019 - 75 committers (2142 closed)

Pharo 6.0 - Jan 2018 - 70 committers (1400 issues closed)

Pharo 5.0 - Apr 2016 100 committers (1600 issues closed)

Pharo 4.0 - Apr 2015 70 committers (1271 issues closed...)

Pharo 3.0 - Apr 2013 (2364 issues closed)

Pharo 2.0 - Apr 2013 (1727 issues closed)

Pharo 1.4 - Apr 2012 (860 updates)





I want to thank all
the contributors

We will continue ... :)

A word of teaching

Pharo on Exercism

Thank you guys for the work!
We owe you more than a beer

Excellent Mooc

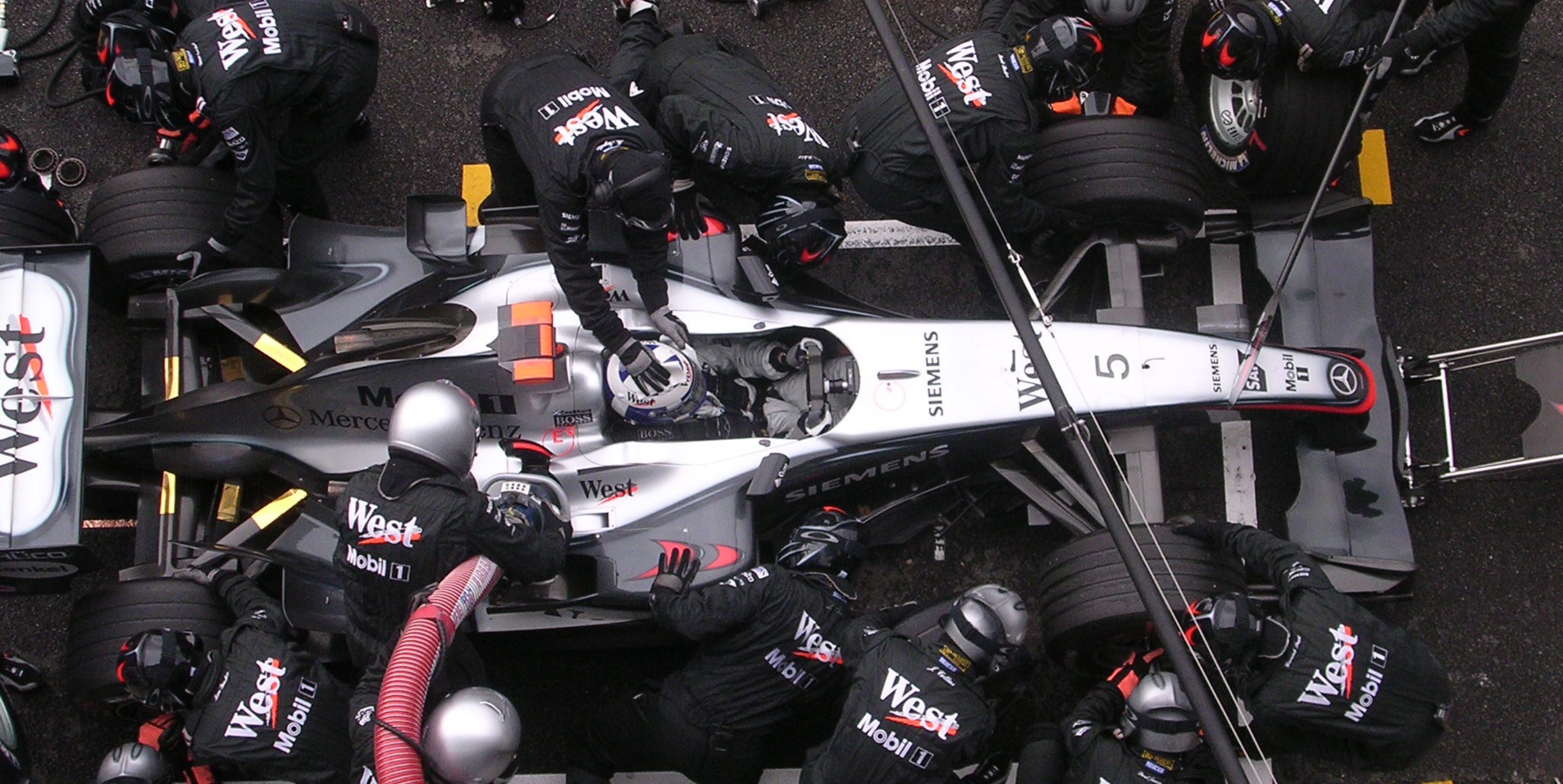
<http://mooc.pharo.org>

”I have just completed week seven of the Pharo Mooc I have already learned so much ! I have spent the last 20 years or so in software development and, following this Mooc, I realized I hadn't really grasped the essence of object oriented design”

GMJuliet on discord June 2019

Forthcoming advanced design mooc

<https://advanced-design-mooc.pharo.org/>



**Pharo is our vehicle
We improve it everyday**

**There are plenty of place for
improvements**

Pharo

is yours

**You can get an impact
Pharo is not a closed box**

http://discord.gg/QewZMZa

The screenshot shows the Discord interface for a server named 'Pharo'. The left sidebar contains a list of channels: #learning-help, #mooc (selected), #general, #development, #offtopic, #iceberg, #bootstrap, #databases, #polymath, #roassal, #roassal-scriptoftheday, #seaside, #moose, #bloc, #ai, and #ai-scriptoftheday. The main chat area displays a conversation in the #mooc channel. A blue banner at the top of the chat area indicates '23 new messages since 9:59 AM on November 23, 2017' and includes a 'MARK AS READ' button. The chat history shows several messages from users akgrant0710, Peter Uhnak, and Marek Erben. The right sidebar lists the server's members, including administrators and channel administrators. The bottom of the interface shows the user 'Stephano' with a status of 'online' and a list of nearby servers.

Pharo **#mooc** **23 new messages since 9:59 AM on November 23, 2017** **MARK AS READ**

akgrant0710 Last Saturday at 11:21 AM
dict includesKey: 'a key'
Take a look through the testing protocol of Dictionary (and superclasses)

Peter Uhnak Last Saturday at 11:26 AM
@Marek Erben especially **at::ifPresent:**, **at::ifAbsentPut:** & co.

Marek Erben Last Saturday at 1:14 PM
Yeah, thank you **@Peter Uhnak**
One more question - is there some simple way to remove one occurrence of item in Bag collection?

Peter Uhnak Last Saturday at 1:34 PM
remove: should do it

Marek Erben Last Saturday at 1:35 PM
But it removes all the occurrences of item, doesn't it?

Yani Last Saturday at 2:20 PM
as in script of the day >url>

Peter Uhnak Last Saturday at 2:36 PM
@Marek Erben did you try? 😊
trying is for free 😊 (edited)

ADMIN—3
Esteban Lorenzano
Juraj Kubelka
kilon Playing Elite: Dangerous

CHANNEL-ADMIN—7
Alexandre Bergel
Clement Bera
PhilBack
Stephano
Torsten Bergmann ...
Tudor Girba
Yuriy Tymchuk

ONLINE—42
akgrant0710
AlejandroInfante
awarenhem
Benoit Verhaeghe

Stephano #1160

We are a cool community



More Pharo's Teachers

- Uni. of Bern • Uni. of Novi Sad • Uni. of Maroua • Uni. of Brussels • Ecole des Mines de Douai • Uni. de Savoie • Ivan Franko Nat. Uni. • Czech Technical Uni. • CULS Prague • Uni. of Buenos Aires • Uni. of Quilmes • Uni. of La Plata • Northern Michigan Uni. • Uni. Technologica Nacional (UTN) • Uni. of Santiago • Uni. Policnica de Catalunya • Uni. de Bretagne Occidentale • Uni. of Tomsk • Uni. of Fernhagen • IT University of Copenhagen • Uni. of Yaounde • INFTI • Uni. of Lyon • Uni. of Cordoba • Uni. of Lille • Uni. of Monterrey • Enit • Uni. of Yaounde • Vietnam France University • Uni. of Yamanashi Gakuin

Research Groups

Lafhis (AR)

SCG (CH)

CAR (FR)

RMOD (FR)

Ummisco (IRD)

Reveal (CH)

Lysic (FR)

CEA-List (FR)

Uqbar (AR)

OC (FR)

CCMI-FIT (CZ)

ASERG (BR)

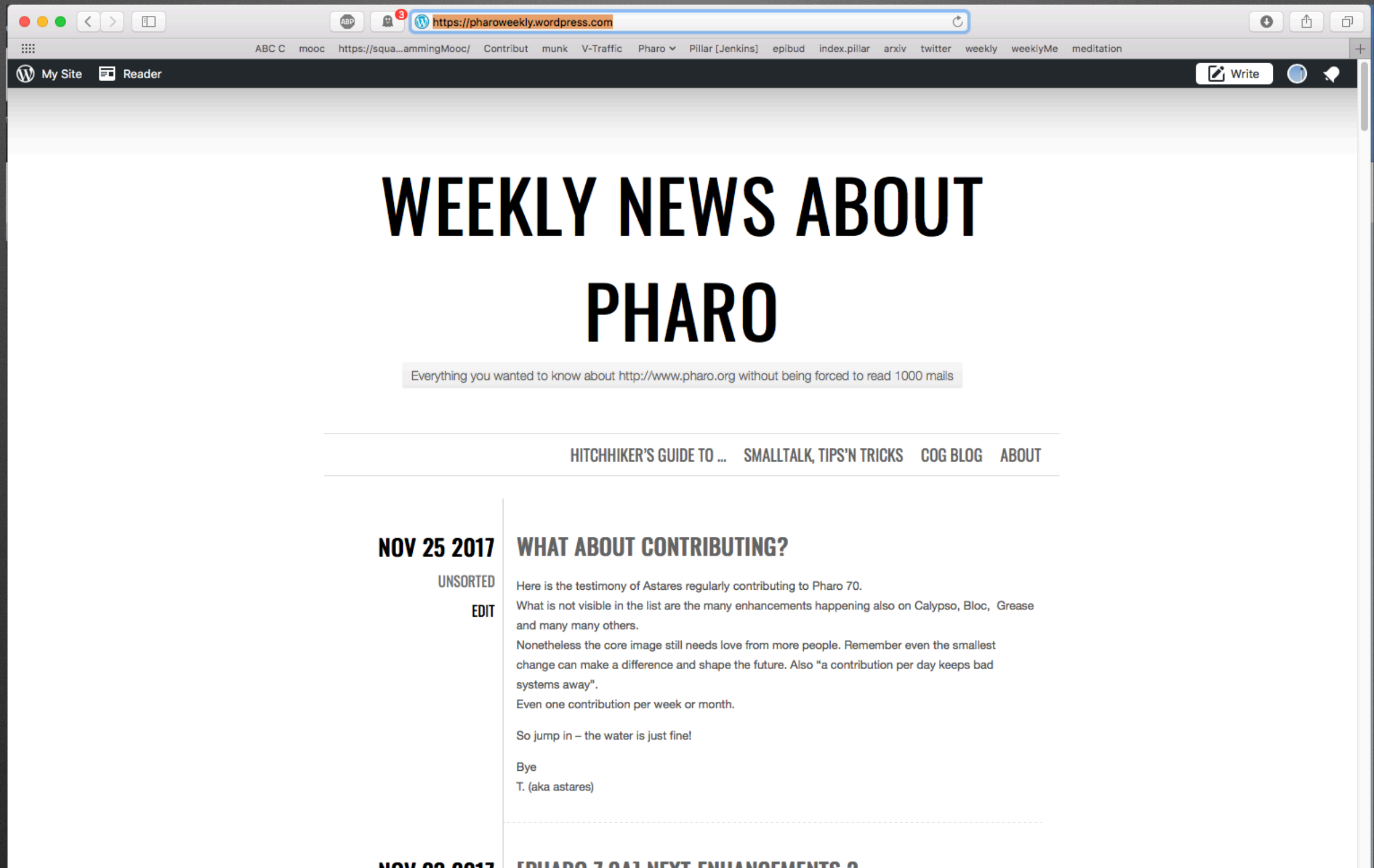
Pleiad (CL)

SoftwareQlab (Sr)

ENIT (FR)

SRA (Jap)

https:// pharowebly.wordpress.com





Join!

Creating good energy, software quality,
learning and having fun

<http://pharo-project.org>