

# Stories About Renraku — the new Quality Model of Pharo

**Agent Status**  
As you seen for available Render Agent nodes, they may be used  
are currently in use by other Controllers. By setting the available  
Nodes, only those nodes that are available for ownership will be used.

**Resource Enforcement**  
Under Windows NS, the Render Agent gathers information about  
number of CPUs and the available memory directly from the operating  
system. On other platforms, this information is based on the Render Agent  
configuration file.

**When the Enforce Node Usage By Available Resource setting is active**  
When the Enforce Node Usage By Available Resource setting is active,  
then StealthNet attempts to prevent usage of each node beyond its capabilities.  
When Restrict By CPU is enabled, then no more instances of a node can  
be created than there are available CPUs. For Restrict By Available  
Memory, StealthNet will query each instance of a node prior to sending  
project to it to see how much available memory it has at that point in time.  
If the memory required for the project (based on the scene settings for  
frame size and segments) exceeds the available memory on the node,  
StealthNet will not send the project to that node.

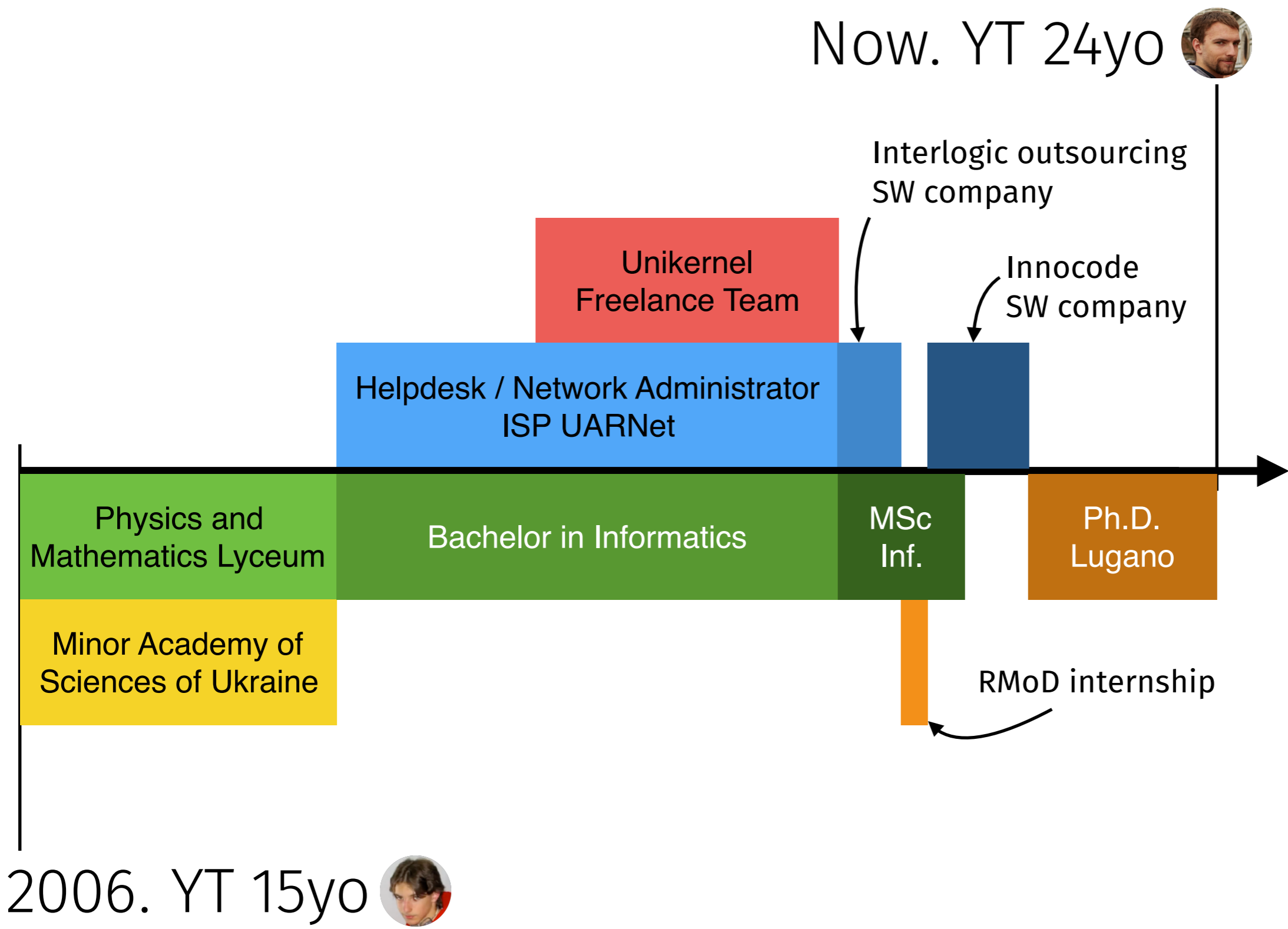
**Ownership Identification**  
When nodes are owned by a Controller, by default  
the name of that Controller's computer is listed in the Owner cell for a  
node. You can override this ownership identity by entering a value in the  
Owner ID field. You should use an identity that is easily associated with you as a  
person. Your name or perhaps your email address, are examples of this.



@yuriy\_tymchuk

Software Composition Group





Now. YT 24yo



Unikernel  
Freelance Team

Helpdesk / Network Administrator  
ISP UARNet

Interlogic outsourcing  
SW company

Innocode  
SW company

Physics and  
Mathematics Lyceum

Bachelor in Informatics

MSc  
Inf.

Ph.D.  
Lugano

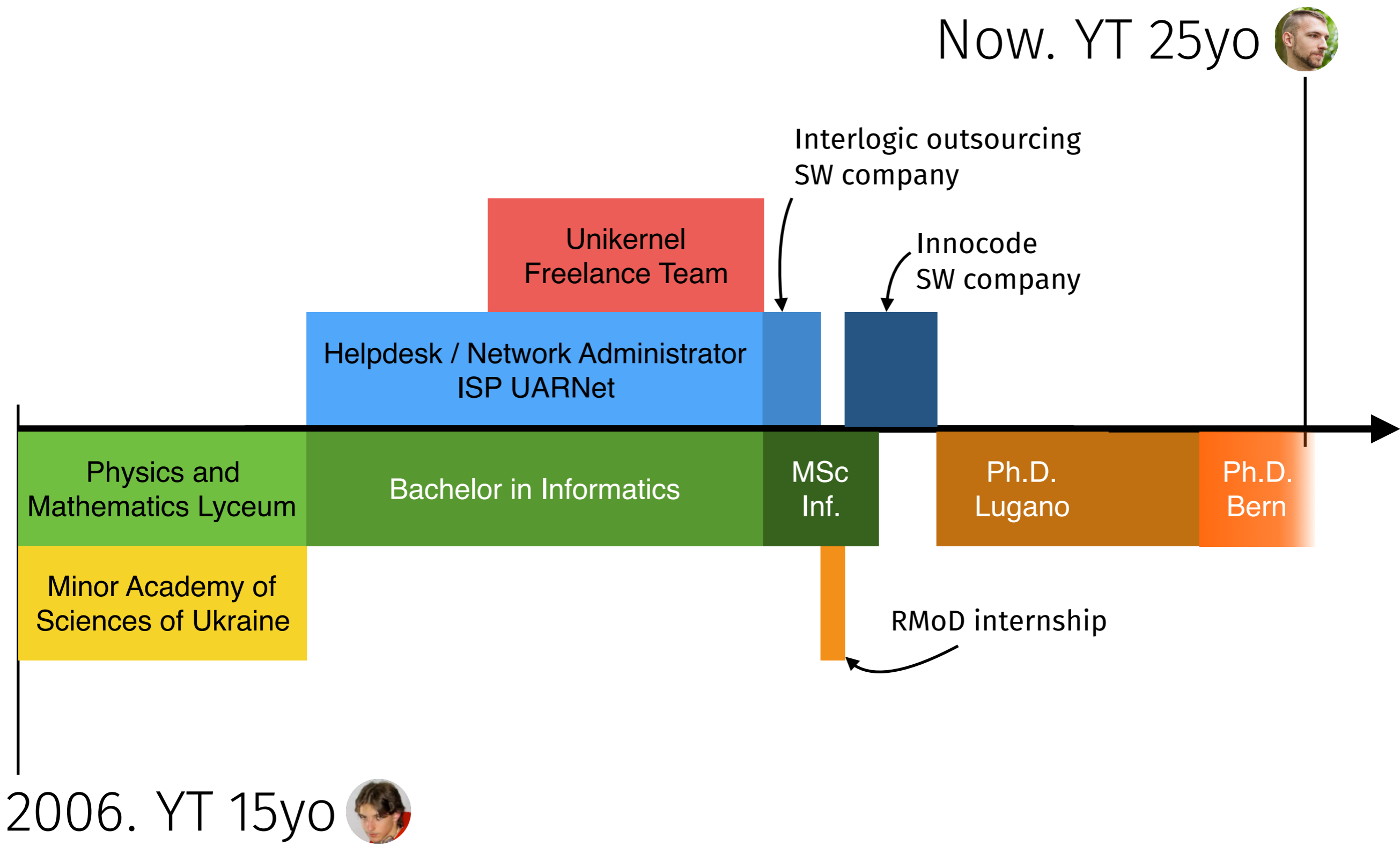
Minor Academy of  
Sciences of Ukraine

RMoD internship

2006. YT 15yo







Now. YT 25yo



Unikernel  
Freelance Team

Interlogic outsourcing  
SW company

Innocode  
SW company

Helpdesk / Network Administrator  
ISP UARNet

Physics and  
Mathematics Lyceum

Bachelor in Informatics

MSc  
Inf.

Ph.D.  
Lugano

Ph.D.  
Bern

Minor Academy of  
Sciences of Ukraine

RMod internship

RMod internship

2006. YT 15yo









# A Mistake

```
isForMethodHolder whenChangedDo: [ :bool |  
    self changed: #isForMethod: with: { bool } ]
```

```
self  
    beForCode;  
    aboutToStyle: true.
```

14/15 [13]

Format as you read W +L



# A Mistake can be Solved Early

```
isForMethodHolder whenChangedDo: [ :bool |  
  self changed: #isForMethod: with: { bool } ]
```

```
self  
  beForCode;  
  aboutToStyle: true.
```

13/15 [2] -- 13/15 [6]

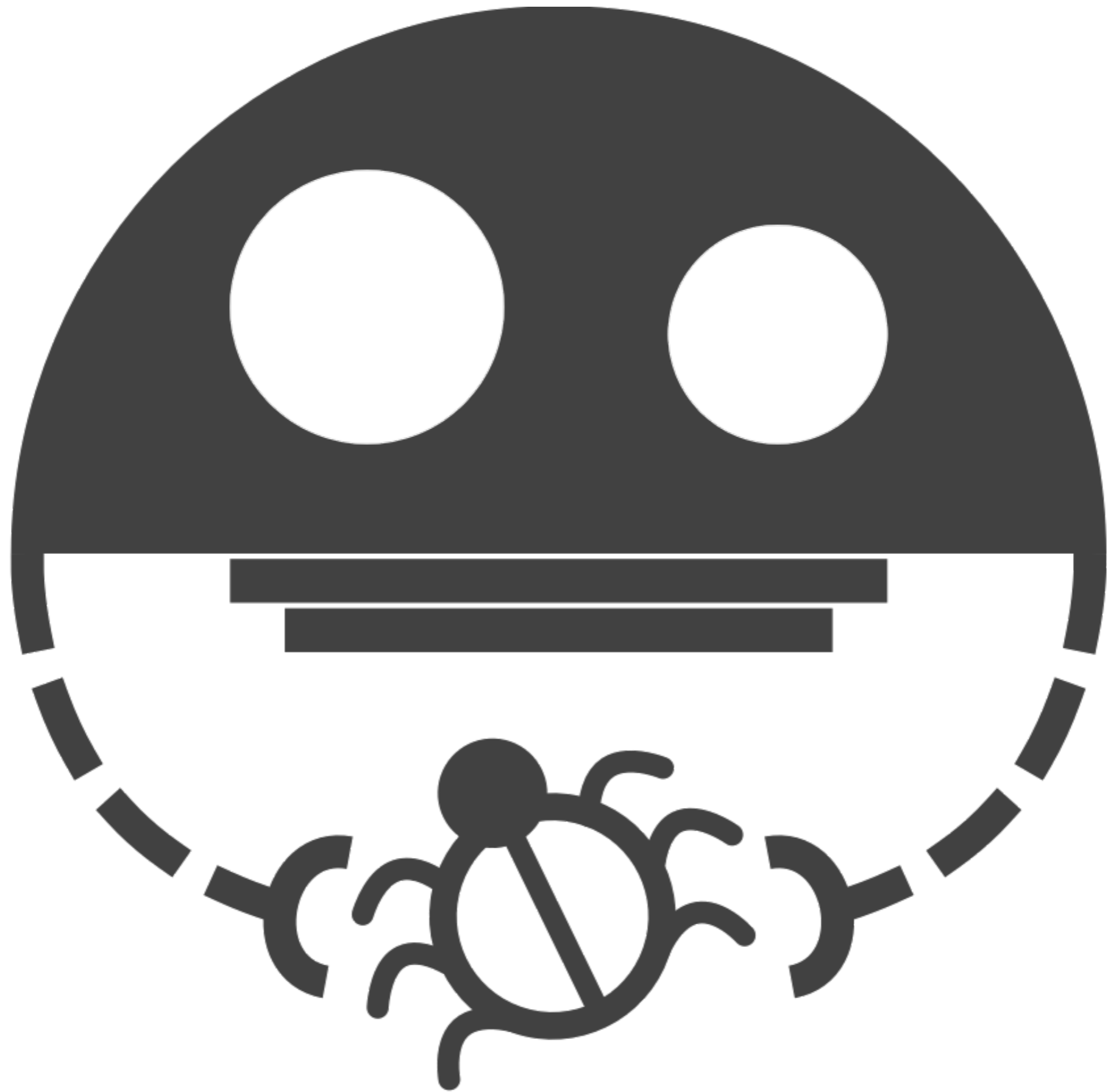
Format as you read W +L

⚠ Uncommon message send ? ✕

Helpful? 👍 👎

❗ [self] Messages sent but not implemented ? ✕

Helpful? 👍 👎



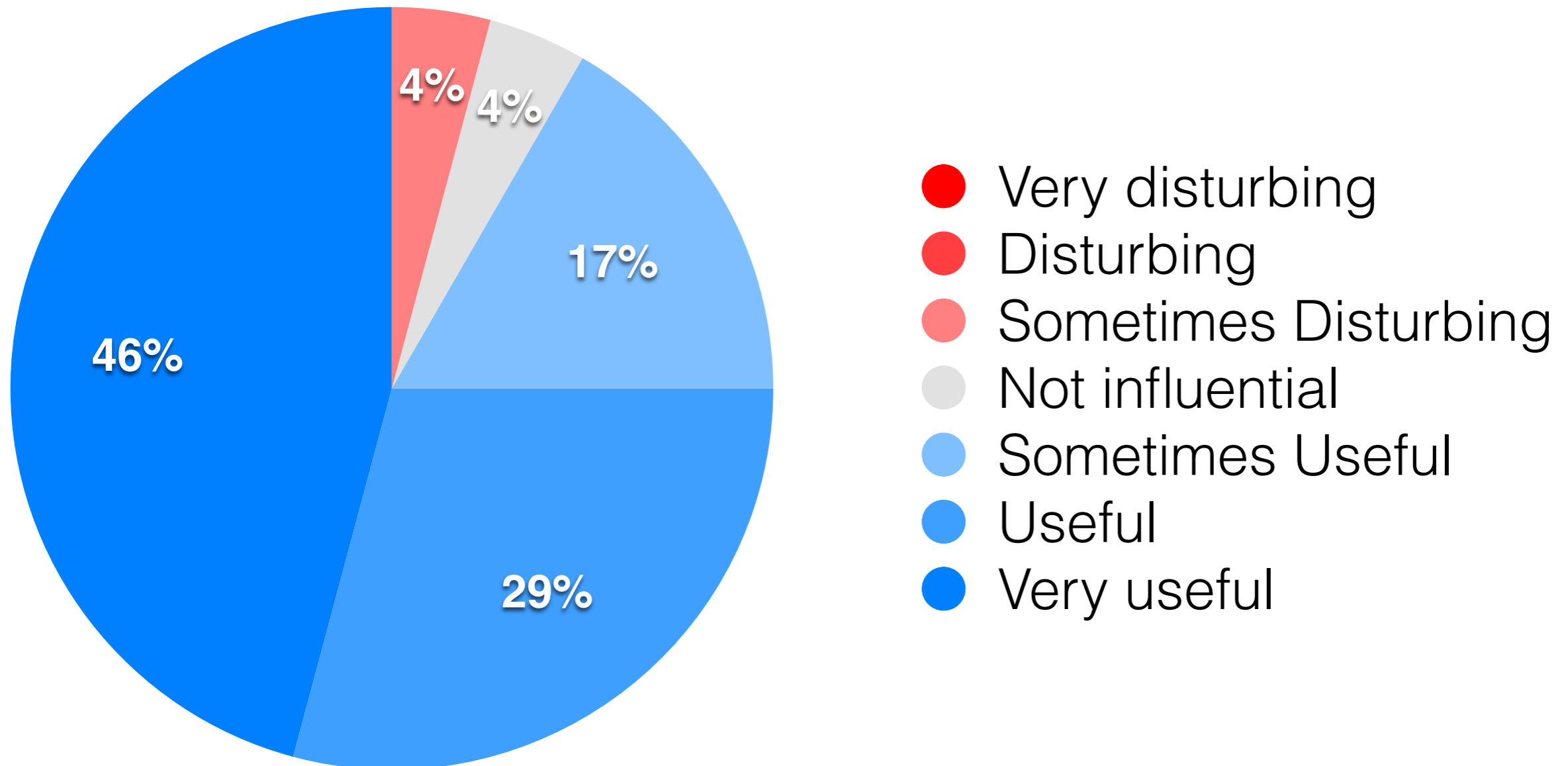
How do you find QualityAssistant?



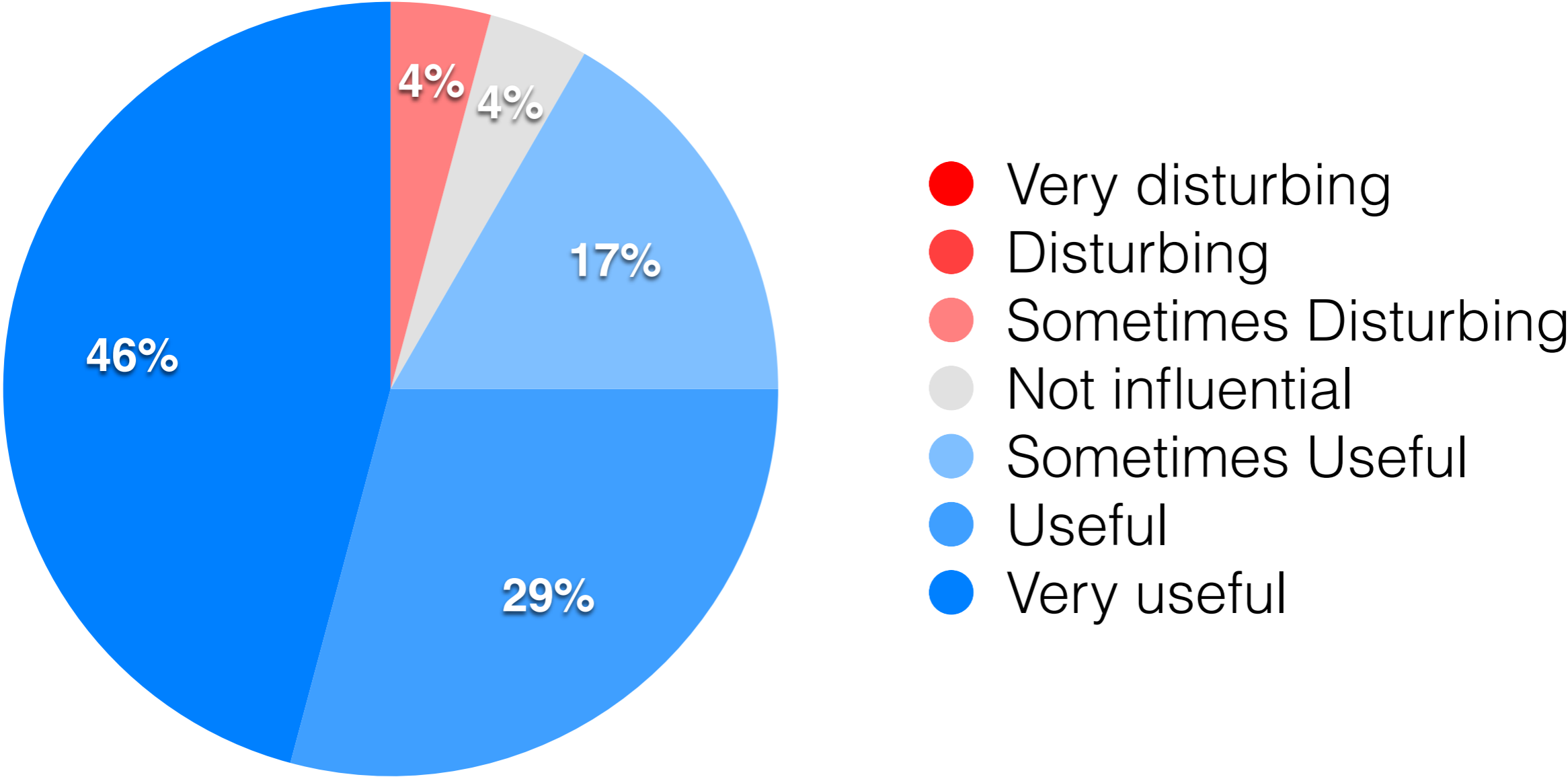
# How do you find QualityAssistant?

- Very disturbing
- Disturbing
- Sometimes Disturbing
- Not influential
- Sometimes Useful
- Useful
- Very useful

# How do you find QualityAssistant?

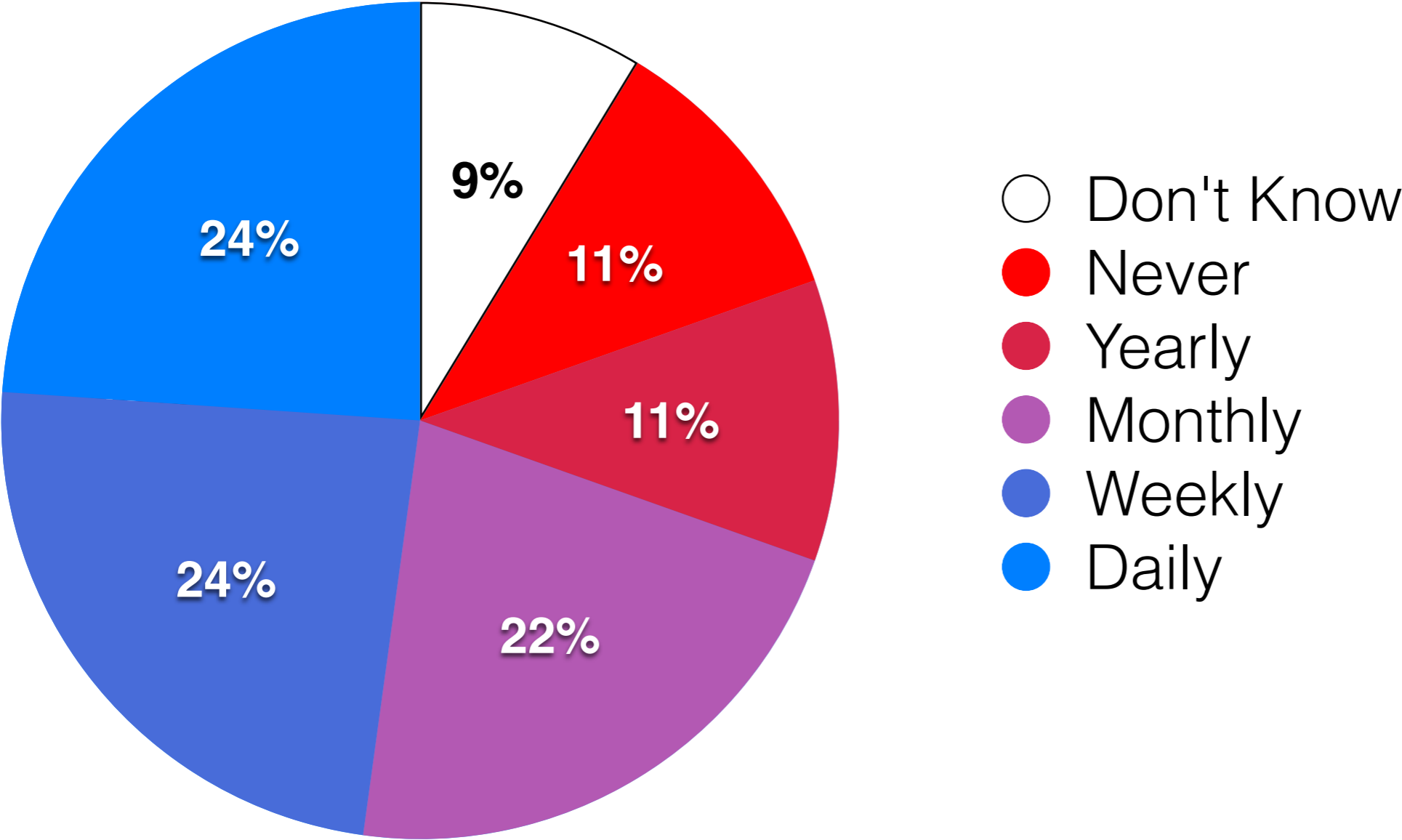


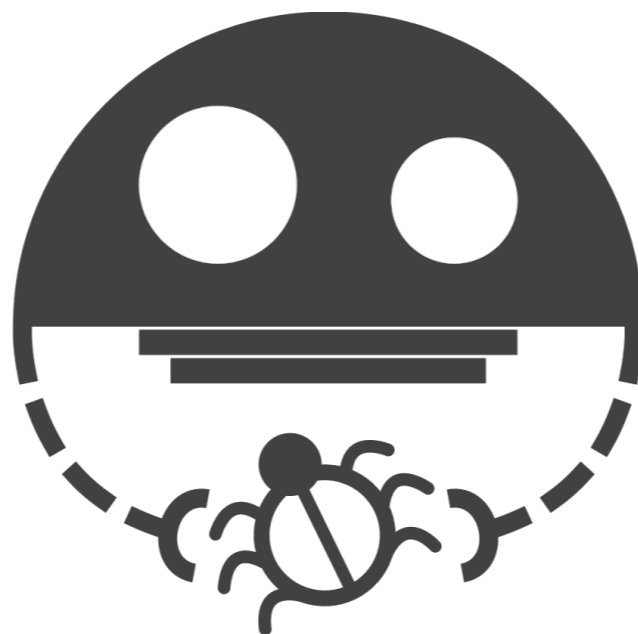
# How often do you use CriticBrowser?





# How often do you use CriticBrowser?









**ONE DOES NOT SIMPLY**



**“SHOW” RENRAKU**



# The Origin





 Renraku

# 連絡





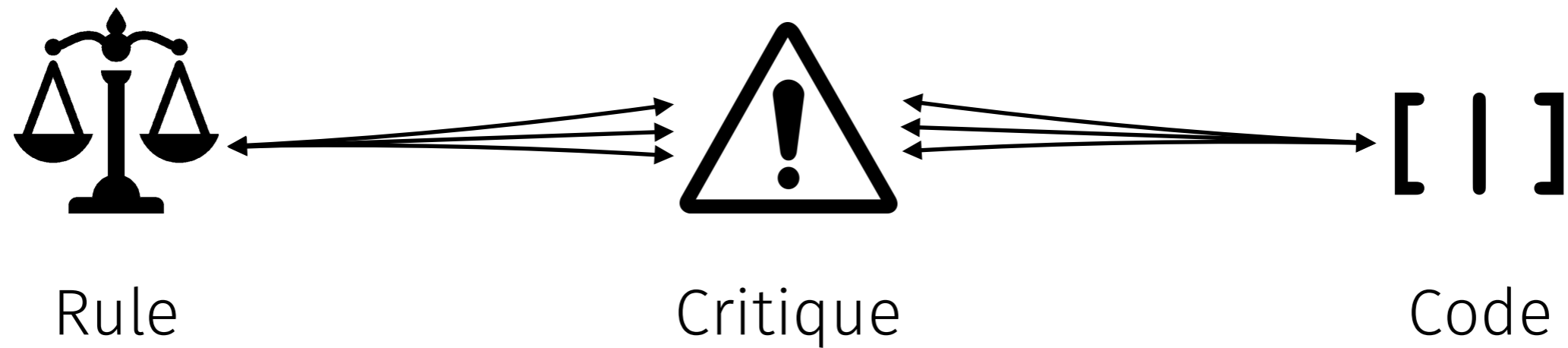
 Renraku

Rule

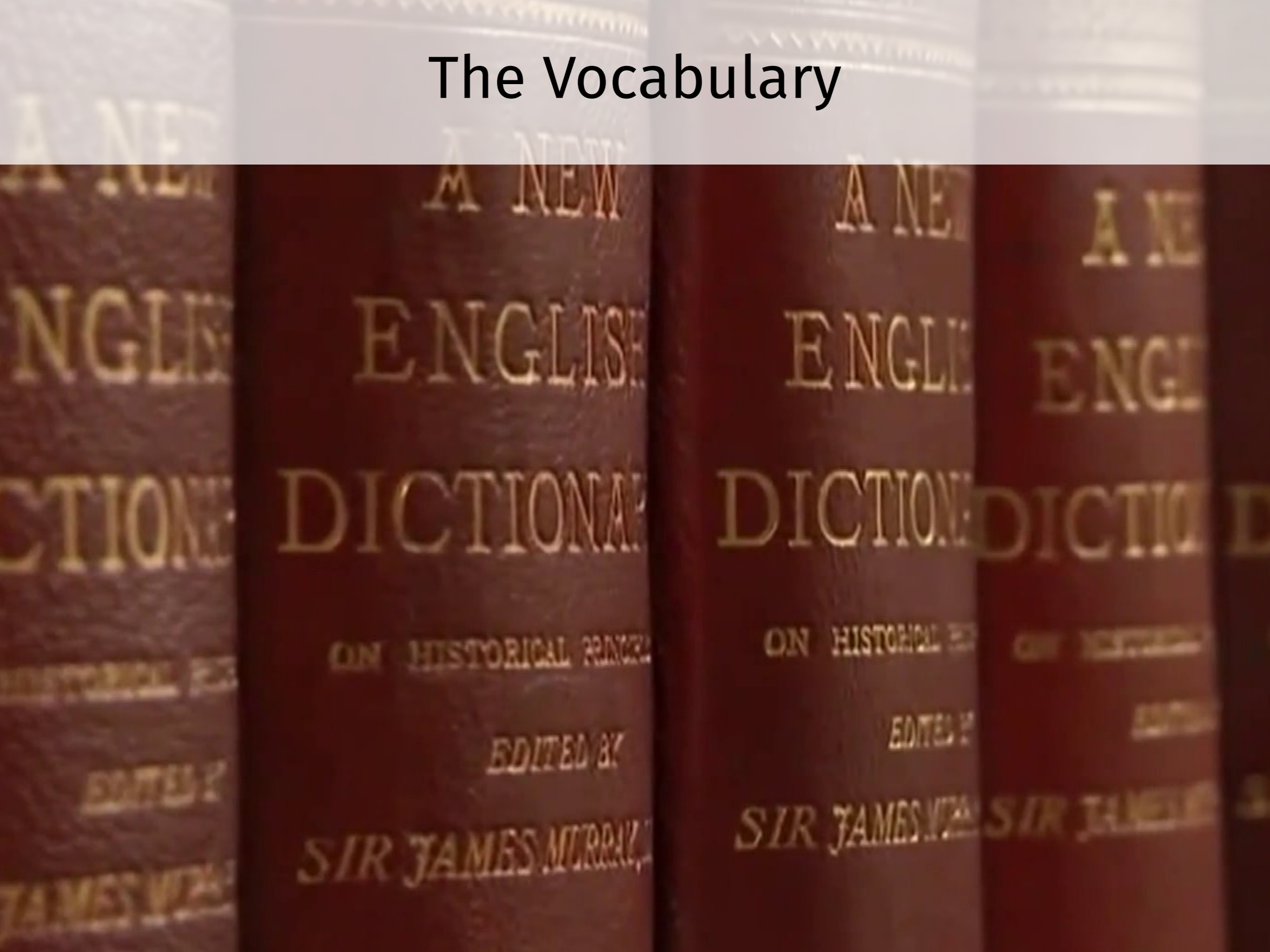


Code

# Renraku



# The Vocabulary



# Renraku





# English Language

Critic



Critique





# English Language



Critic



Critique



# English Language



Critic



Critique



# Huge Refactoring



Critic



Critique

ReTrivialCritic → ReTrivialCritique



# Our Dog Food is Good

```
argument := message arguments first.  
argument isBlock or: [ argument isLiteralNode ] ] ]  
thenCollect: [ :message |  
  aCriticBlock cull: (ReTrivialCritic  
    withAnchor: (ReIntervalSourceAnchor  
      entity: anAST  
      interval: message sourceInterval)  
    by: self) ])
```

13/19 [25] -- 13/19 [40]

Format as you read W +L

⚠ Re\*Critic -> Re\*Critique ? 🔄 ✕

Helpful? 👍 🗨







# Our Dog Food is Good!

```
argument := message arguments first.  
ar  
thenC  
aC
```

```
thenCollect: [ :message |  
  aCriticBlock cull: (ReTrivialCritique  
  aCriticBlock cull: (ReTrivialCritique  
    withAnchor: (ReIntervalSourceAnchor  
      entity: anAST  
      interval: message
```

13/19 [25] -- 13/19 [40]  Format as you read W +L

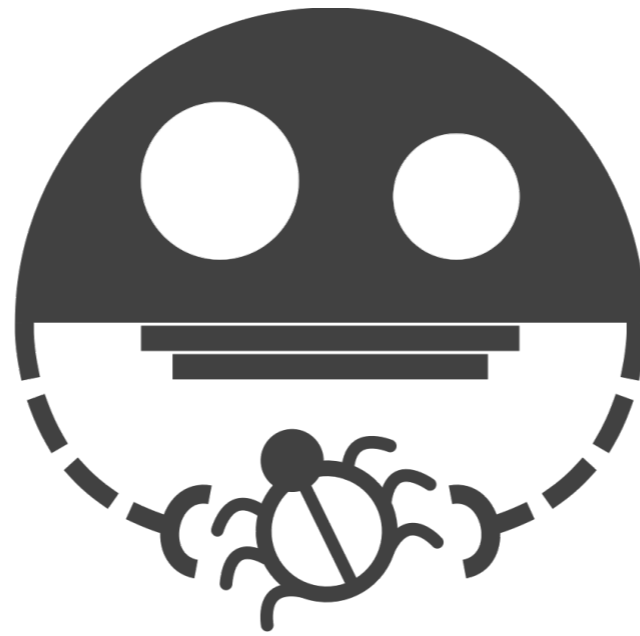
⚠ Re\*Critic -> Re\*Critique ?   Helpful?  



# Death and Life

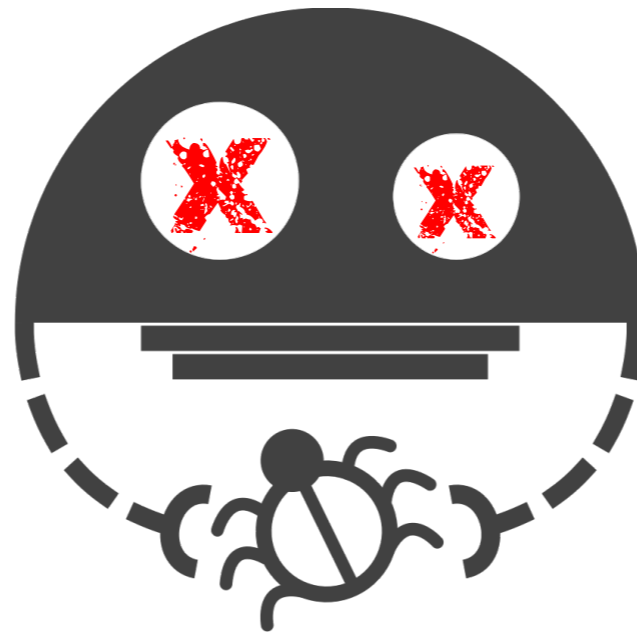








# Who Shoots the Assistant?





# Who is the Assistant?

⚠️ Uncommon message send ? ✖

Helpful? 👍 👎

❗ [self] Messages sent but not implemented ? ✖

Helpful? 👍 👎



# Who is the Assistant?

⚠️ Uncommon message send ? ✖️ Helpful? 👍 👎

❗ [self] Messages sent but not implemented ? ✖️ Helpful? 👍 👎

Playground

a CompiledMethod (SettingDeclaration>>#exportSettingAction)

Raw 4 Bytecode Source AST Ir Decompiled Header Meta

- ⚠️ Eliminate unnecessary not's Optimization
- ❗ Messages sent but not implemented Bugs
- ℹ️ Temporary variables not read AND written Optimization
- ⚠️ Sends "questionable" message Coding Idiom Violation

SettingDeclaration>>#exp...

Search

QA Critics 4/4 ➡

- ⚠️ Eliminate unnecessary not's
- ℹ️ Temporary variables not read AND w...
- ⚠️ Sends "questionable" message
- ❗ Messages sent but not implemented ✓ ➡

Senders 1/1

SettingBrowser>>#exportSettings

Implementors 2/2

PragmaSetting>>#exportSettingAction

SettingDeclaration>>#exportSetting...

Messages sent but not implemented

Checks for messages that are sent by a method, but no class in the system implements such a message. Further checks if messages sent to self or super exist in the hierarchy, since these can be statically typed. Reported methods will certainly cause a `doesNotUnderstand:` message when they are executed.



# This is not QualityAssistant

```
Implementors of showActions [1]
Morph (meta-actions) showActions [Morphic-Core]

Browse Users Senders Implementors Version Source

"The mechanism on a Button"
... (self respondsTo: #actionSelector)
  ifTrue: ["A button"
    selector := self actionSelector.
    cls := self target class whichClassIncludesSelector:
selector.
    cls
    ifNotNil: ["want more than default behavior"
      cls == Morph
        ifFalse: [adder value: cls value: selector]].
    Smalltalk tools messageList browse: list title: 'Actions of
' , self printString.
```



# This is not QualityAssistant

```
Morph (meta-actions) showActions [Morphic-Core]

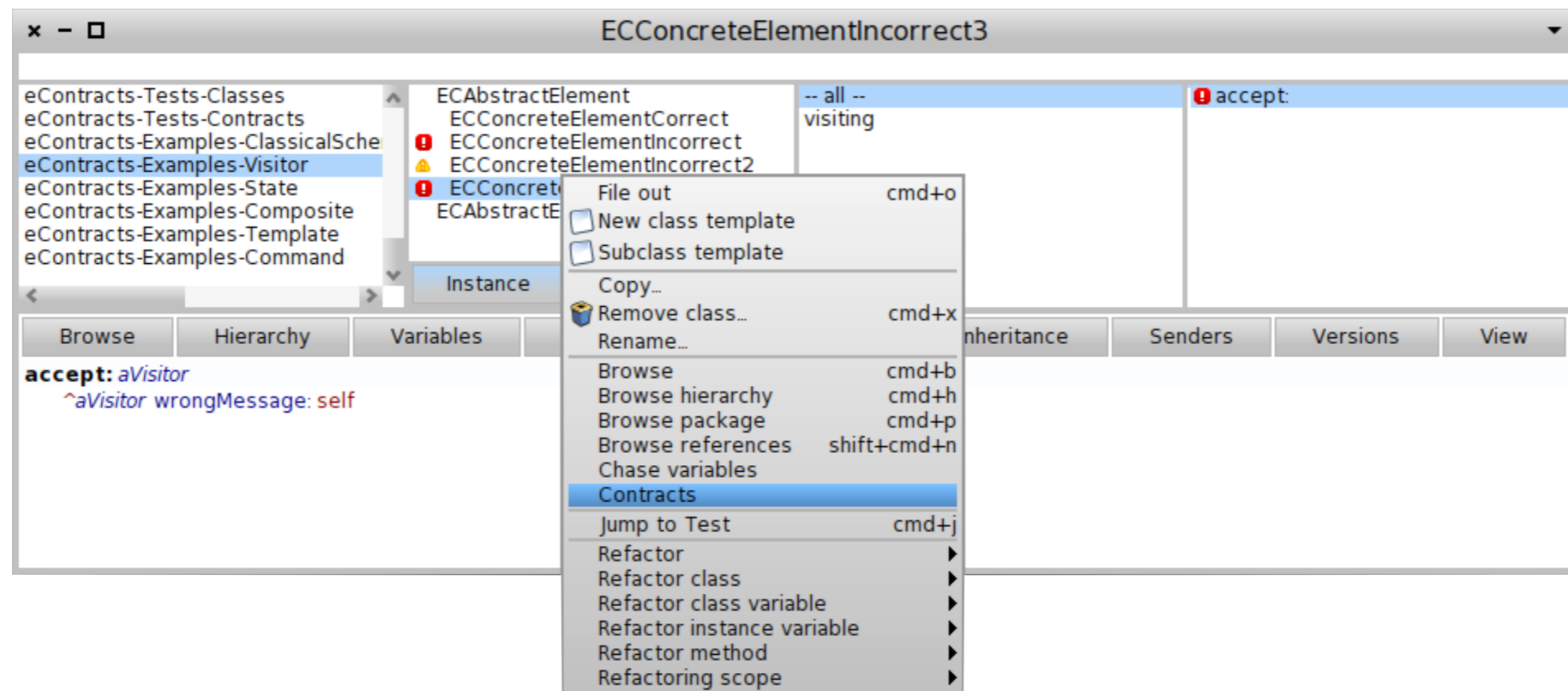
Browse Users Senders Implementors Version Source

"The mechanism on a Button"
... (self respondsTo: #actionSelector)
    ifTrue: ["A button"
            selector := self actionSelector.
            [target] Super and Self Messages sent but not implemented ? X Helpful? 👍 👎 r:
            selector.
            cls
            ifNotNil: ["want more than default behavior"
                    cls == Morph
                    ifFalse: [adder value: cls value: selector]].
            Smalltalk tools messageList browse: list title: 'Actions of
            ', self printString.
```





# Usage Contracts (2013)





# Usage Contracts (2013)

ECConcreteElementIncorrect3

ECAbstractElement	-- all --	! accept:
ECConcreteElementCorrect	visiting	
! ECConcreteElementIncorrect		
! ECConcreteElementIncorrect2		
! ECConcreteElementIncorrect3		
ECAbstractElement		

Instance

Variables

Inheritance Senders Versions

self

- File out cmd+o
- New class template
- Subclass template
- Copy...
- Remove class... cmd+x
- Rename...
- Browse cmd+b
- Browse hierarchy cmd+h
- Browse package cmd+p
- Browse references shift+cmd+n
- Chase variables
- Contracts**
- Jump to Test cmd+j
- Refactor ▶
- Refactor class ▶
- Refactor class variable ▶
- Refactor instance variable ▶
- Refactor method ▶
- Refactoring scope ▶



# Polycriticism





# Issue Tracker Connection





# Issue Tracker Connection



*We want to link FogBugs issues to Pharo's classes and methods.*

*Can we display this information in QualityAssistant?*

# Issue Tracker Connection



*Sure!*

# Issue Tracker Connection



*Sure!*

*Just follow the  
Renraku protocol*



# External Properties for Editor

The screenshot displays the RubyMine IDE interface for the `IERIssues` class. The main editor area shows the following object details:

```
Object subclass: #IERIssues
  instanceVariableNames: 'issues'
  classVariableNames: ''
  package: 'IssueEntityRelation-Core'
```

The History Navigator on the right lists the following actions:

- all --
- accessing
- adding
- enumerating
- gt-inspector-extension
- printing
- removing

The bottom status bar displays a list of issues:

- Issue #10 with 3 entities
- Issue #15 with 4 entities
- Issue #19 with 2 entities



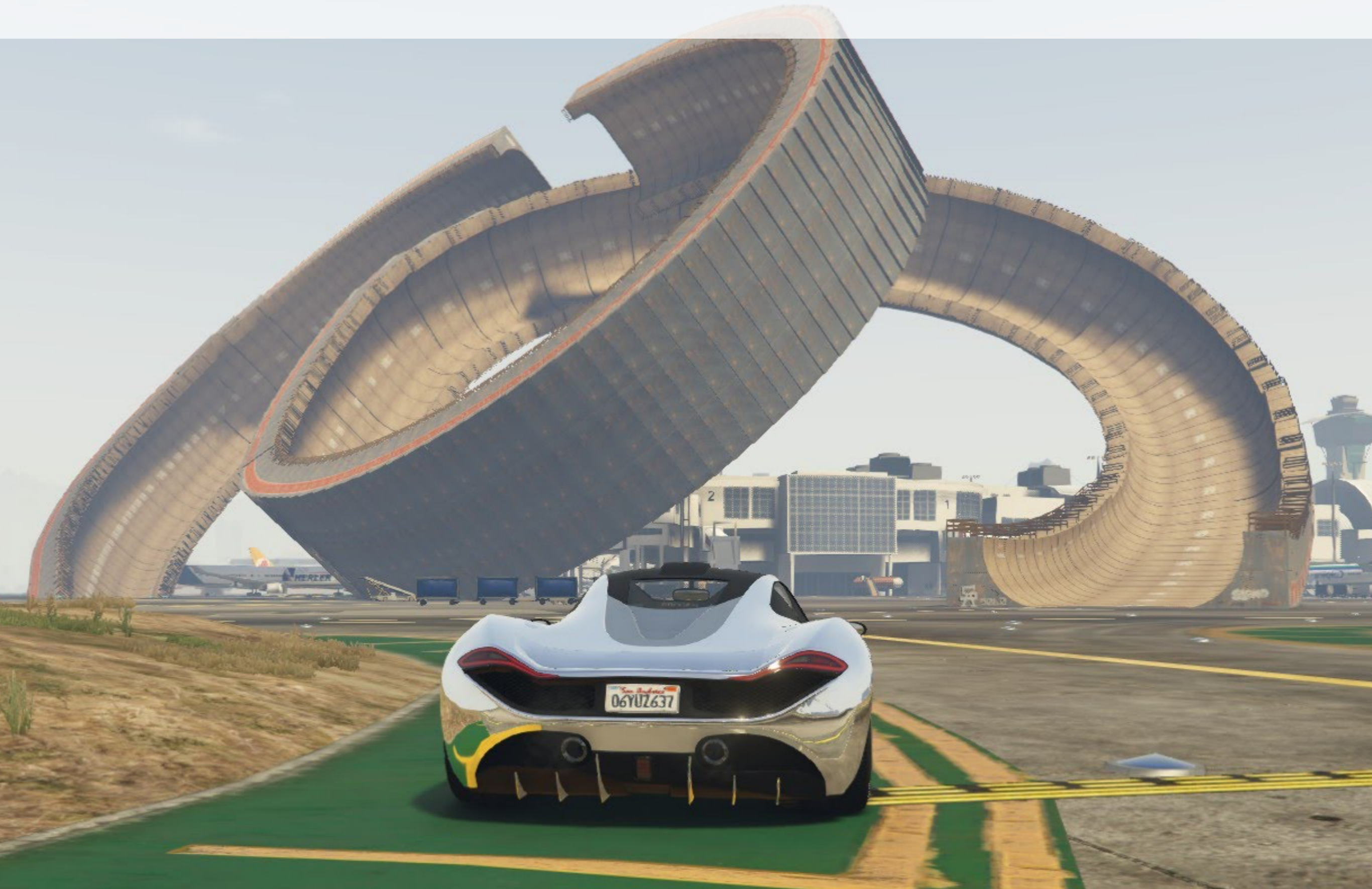
# External Properties for Editor

The screenshot shows an IDE interface with the following components:

- Class Hierarchy:** A tree view on the left shows the package structure: IssueEntityRelation (Core, UI, IssueTracking). A dropdown menu is open, listing classes: IERIssue, IERIssues (selected), IERIssuesRandomFactory, IERUITools, CompiledMethod, and TBehavior.
- Object Inspector:** The main area displays the details for an object subclass: #IERIssues. The details are: instanceVariableNames: 'issues', classVariableNames: '', and package: 'IssueEntityRelation-Core'.
- Bottom Panel:** A list of three objects: Issue #10 with 3 entities, Issue #15 with 4 entities, and Issue #19 with 2 entities. Each entry has a green bug icon, a blue exclamation mark, a trash icon, and a green plus icon.



# Feedback Loop





# Feedback

```
isForMethodHolder whenChangedDo: [ :bool |  
  self changed: #isForMethod: with: { bool } ]  
  
self  
  beForCode;  
  aboutToStyle: true.
```

13/15 [2] -- 13/15 [6]  Format as you read W +L

⚠️ Uncommon message send ? ✖️ Helpful? 👍 👎

❗ [self] Messages sent but not implemented ? ✖️ Helpful? 👍 👎



# RBInstVarInSubclassesRule



# RBInstVarInSubclassesRule

*„I have no idea what this rules does“*





# RBIInstVarInSubclassesRule

*„I have no idea what this rules does“*

```
RTInteraction subclass: #RTAbstractPopup  
instanceVariableNames: ''  
classVariableNames: ''  
package: 'Roassal2-Interactions'
```

1/4 [1]

Format as you read W +L

⚠ [text:] Method defined in all subclasses, but not in superclass ? 🗑️ ✖

📌 Instance variables defined in all subclasses ? ✖





# RBIInstVarInSubclassesRule

*„I have no idea what this rule does“*

```
RTInteraction subclass: #RTAbstractPo
instanceVariableNames: ''
classVariableNames: ''
package: 'Roassal2-Interactions'
```

## Instance variables defined in all subclasses



Checks classes for instance variables that are defined in all subclasses.

OK

1/4 [1]

Format as you read W +L

⚠ [text:] Method defined in all subclasses, but not in superclass ? 🔄 ✖

ⓘ Instance variables defined in all subclasses ? ✖



# RBInstVarInSubclassesRule

*„I have no idea what this rule does“*

```
RTInteraction subclass: #RTAbstractPo
instanceVariableNames: ''
classVariableNames: ''
package: 'Roassal2-Interactions'
```

[text] Same instance variable defined in ALL subclasses



Checks classes for instance variables that are defined in all subclasses.

OK

1/4 [1]

Format as you read W +L

⚠ [text:] Method defined in all subclasses, but not in superclass ? ✖

! [text] Same instance variable defined in ALL subclasses ? ✖



# RBInstVarInSubclassesRule

*„I have no idea what this rule does“*

```
RTInteraction subclass: #RTAbstractPo
instanceVariableNames: ''
classVariableNames: ''
package: 'Roassal2-Interactions'
```

[text] Same instance variable defined in ALL subclasses



All subclasses of this class define the same variable.  
Most likely this variable should be pulled up to the superclass.

OK

1/4 [1]

Format as you read W +L

⚠ [text:] Method defined in all subclasses, but not in superclass ? ✖

! [text] Same instance variable defined in ALL subclasses ? ✖



# RBInstVarInSubclassesRule

*„I have no idea what this rules does“*

The screenshot shows an IDE interface with several overlapping windows:

- Changes Browser:** A dialog box with the title "Changes Browser". It contains a list of changes with three checked items:
  - Remove instance variable text from RTPlatformPopup
  - Remove instance variable text from RTPopup
  - Add instance variable text to RTAbstractPopupBelow the list, a preview shows the text: "Remove instance variable text from RTPlatformPopup". At the bottom are "Ok" and "Cancel" buttons.
- Warning Message:** A message box with the text "[text] Same instance variable defined in ALL subclasses". It has an "OK" button at the bottom right.
- Code Editor:** Shows a snippet of code:

```
RTInteraction sub  
instanceVariabl  
classVariableNa  
package: 'Roass
```
- IDE Status Bar:** Shows "1/4 [1]" and "Format as you read W +L".
- Warning List:** A list of warnings at the bottom of the IDE:
  - [text:] Method defined in all subclasses, but not in superclass ? [icon] X
  - [text] Same instance variable defined in ALL subclasses ? [icon] XThe second warning is circled in red.



# The Demand Shapes the Model







# Started by the Message Browser

```
Implementors of showActions [1]
Morph (meta-actions) showActions [Morphic-Core]

Browse Users Senders Implementors Version Source

"The mechanism on a Button"
... (self respondsTo: #actionSelector)
  ifTrue: ["A button"
    selector := self actionSelector.
    cls := self target class whichClassIncludesSelector:
selector.
    cls
      ifNotNil: ["want more than default behavior"
        cls == Morph
          ifFalse: [adder value: cls value: selector]].
Smalltalk tools messageList browse: list title: 'Actions of
', self printString.
```



# We Need an Interval

```
self target class
```

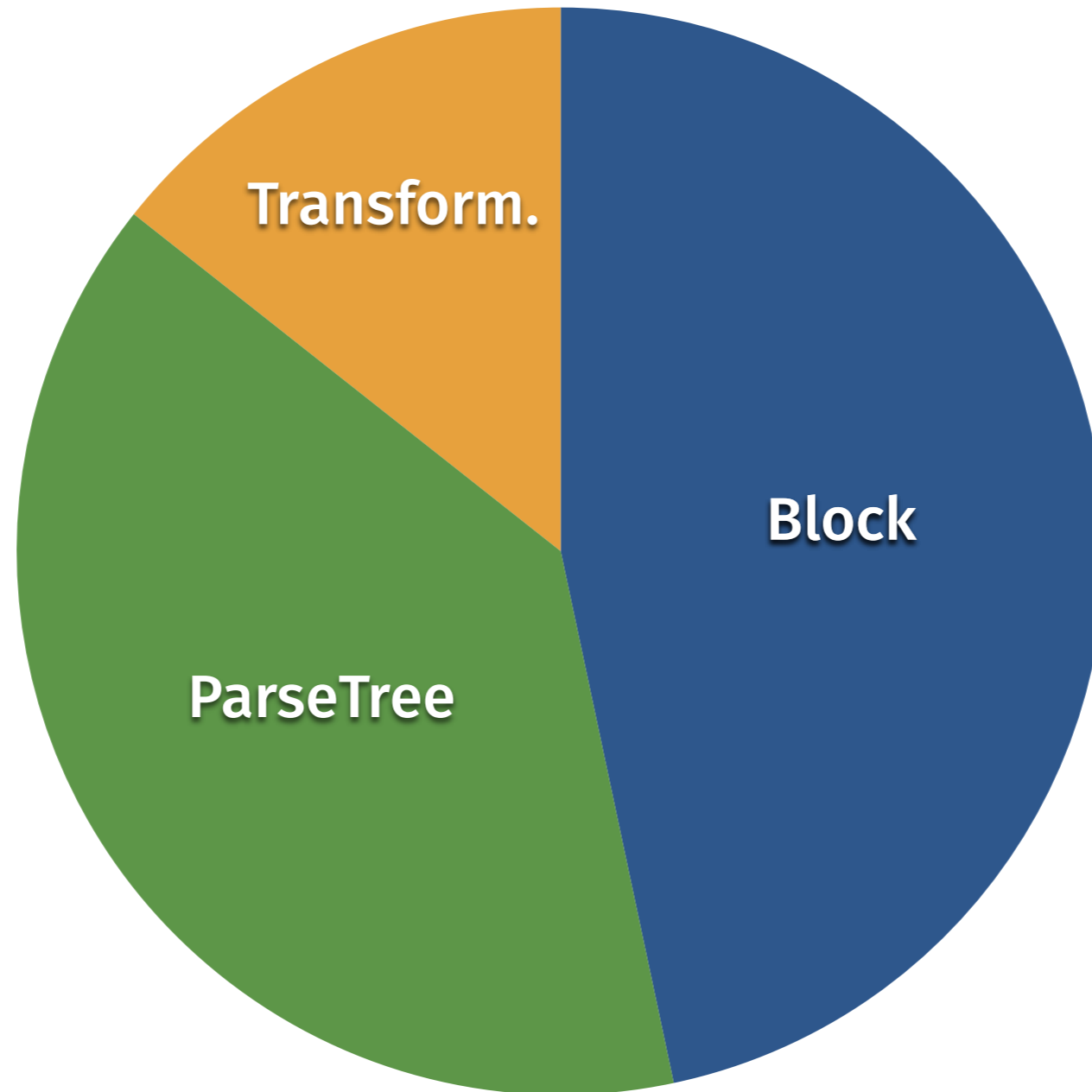


# We Need an Interval

`self` | `target` | `class`



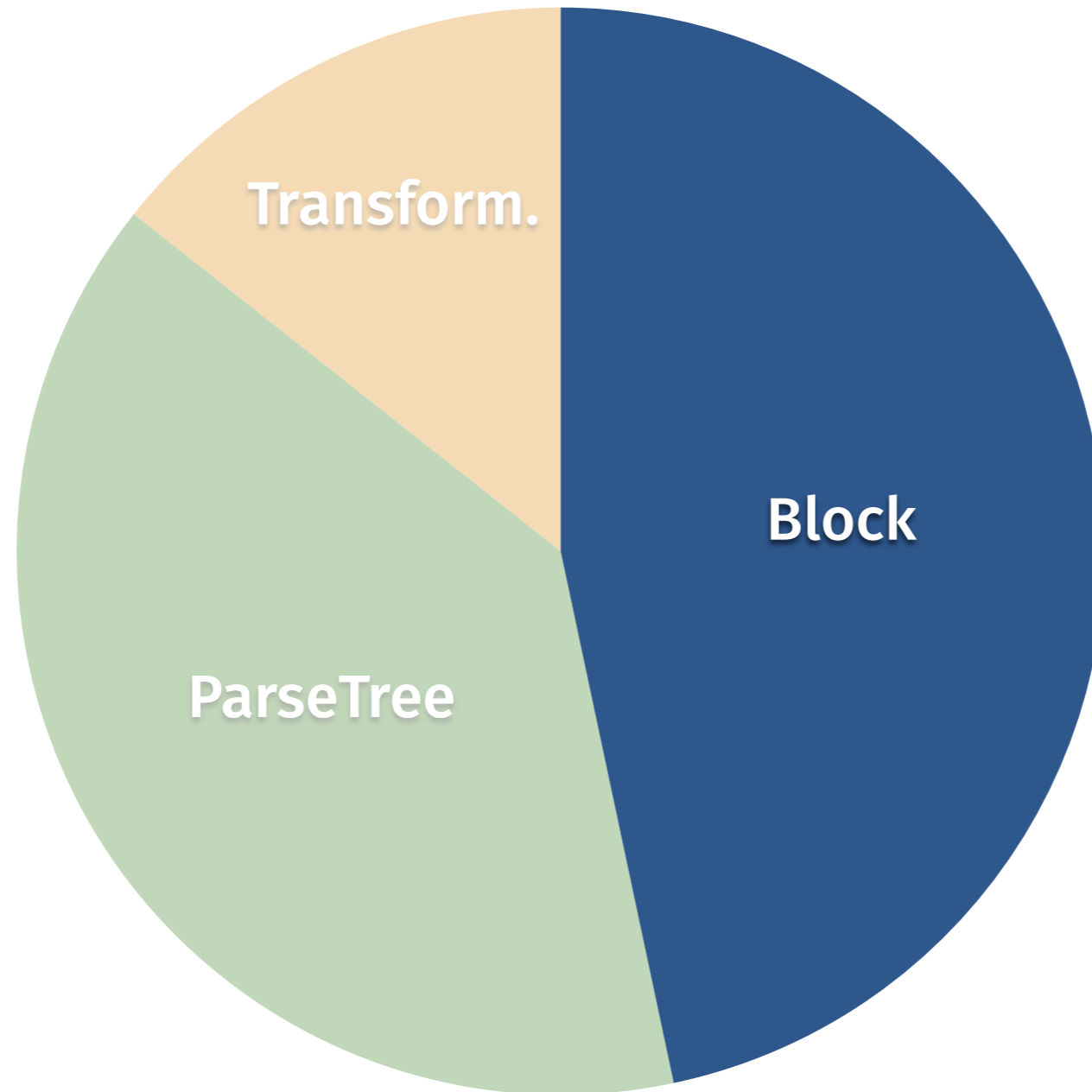
# Types of Rules





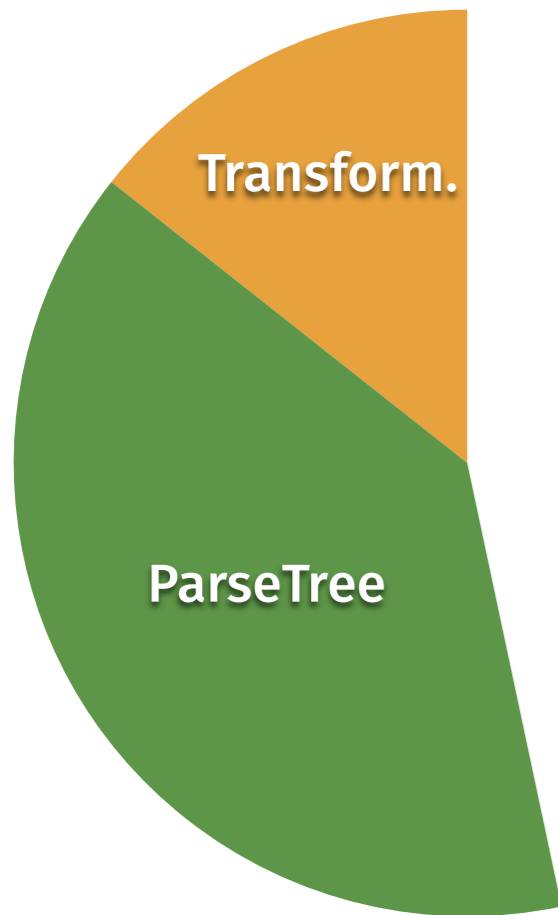


# Types of Rules



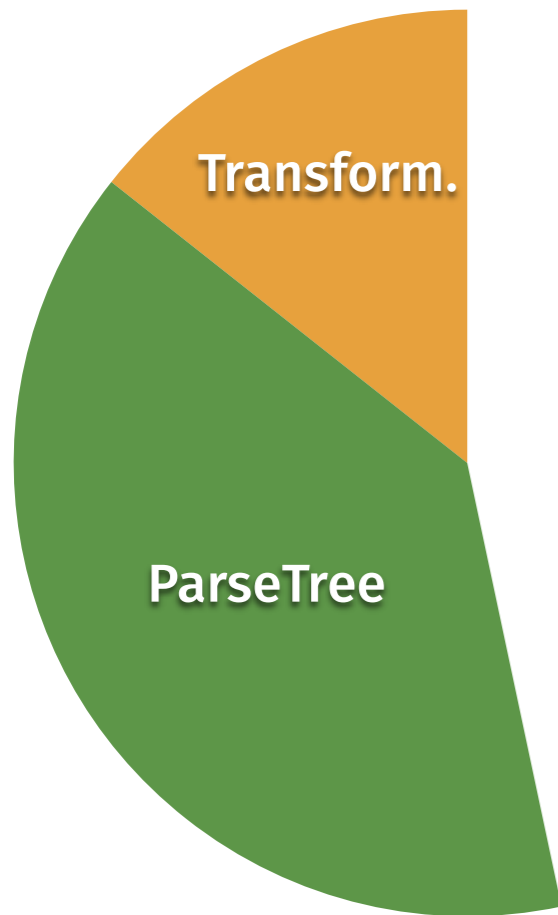


# Types of Rules





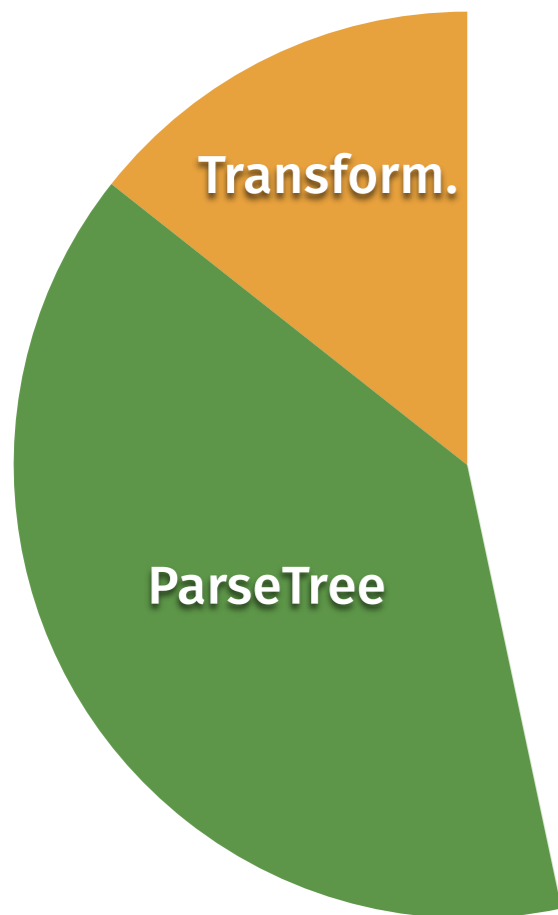
# Types of Rules



'Smalltalk ui icons iconNamed: `#aSymbol`'



# Types of Rules



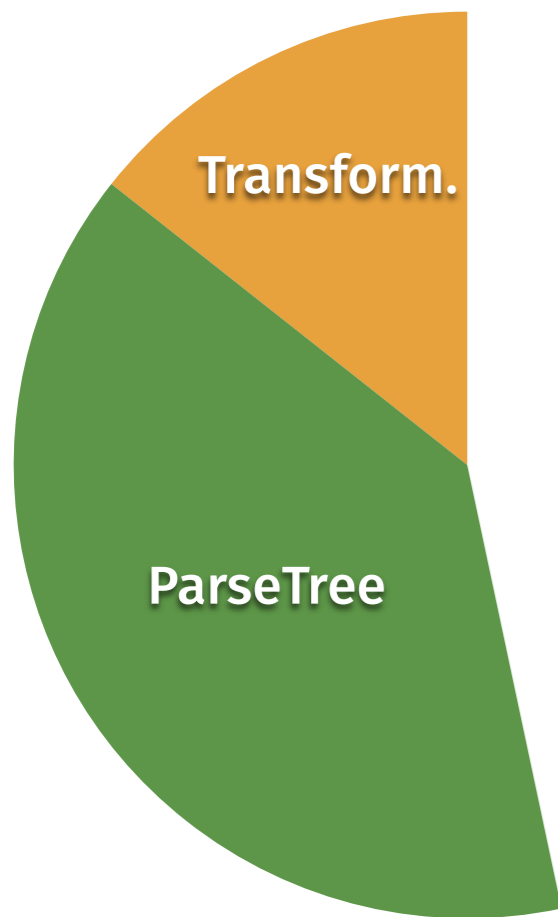
```
'Smalltalk ui icons iconNamed: `#aSymbol'
```

```
^ GLMGenericAction new  
  icon: (Smalltalk ui icons iconNamed: #stop);  
  action: [ self halt ].
```





# Types of Rules

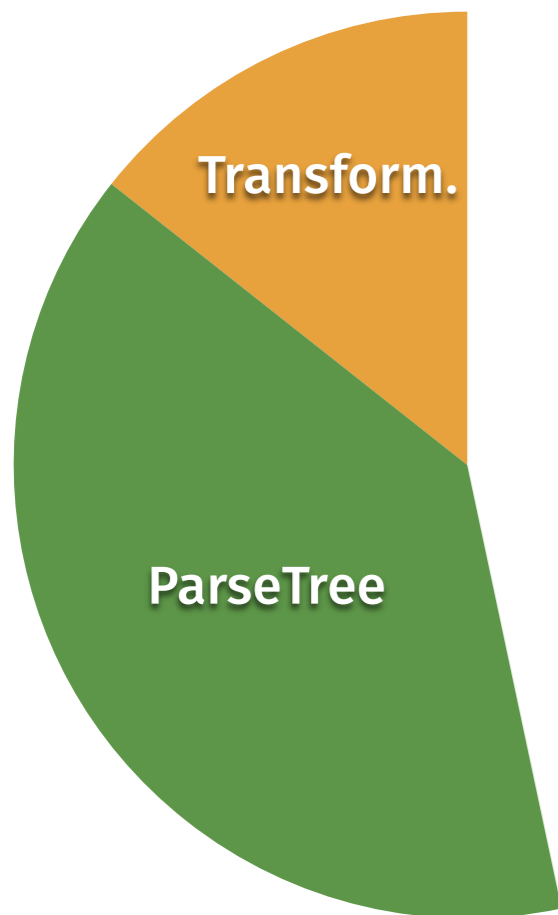


```
'Smalltalk ui icons iconNamed: `#aSymbol'
```

```
^ GLMGenericAction new  
  icon: (Smalltalk ui icons iconNamed: #stop);  
  action: [ self halt ].
```



# Types of Rules



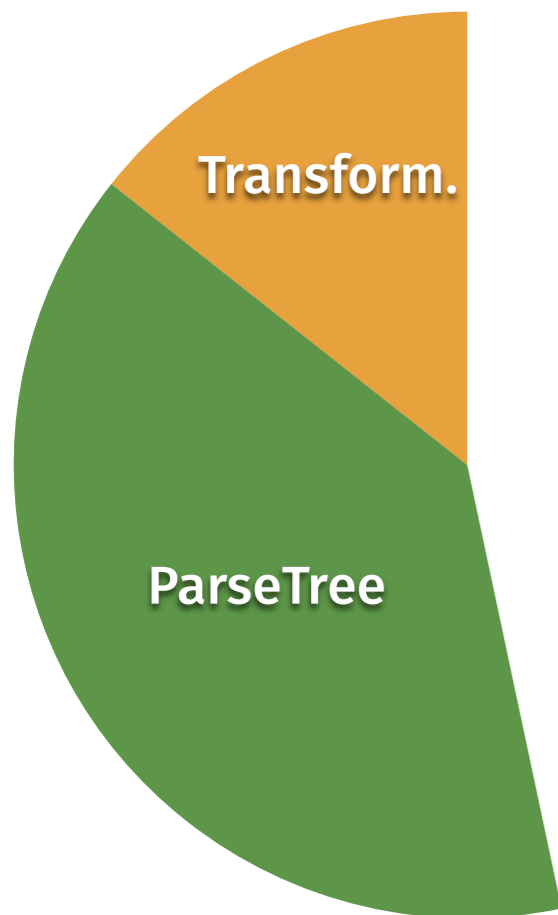
```
'Smalltalk ui icons iconNamed: `#aSymbol'
```

```
^ GLMGenericAction new  
  icon: (Smalltalk ui icons iconNamed: #stop);  
  action: [ self halt ].
```

```
'`#aSymbol asIcon'
```



# Types of Rules



```
'Smalltalk ui icons iconNamed: `#aSymbol'
```

```
^ GLMGenericAction new  
  icon: (Smalltalk ui icons iconNamed: #stop);  
  action: [ self halt ].
```

```
'`#aSymbol asIcon'
```

```
^ GLMGenericAction new  
  icon: (#stop asIcon);  
  action: [ self halt ].
```







# Node-based Critiques

```
a CompiledMethod (GLMPaneScroller>>#pane...
... | I... S... Ir AST He... Meta
paneWidthsToFit: limit
  | padded |
  padded := Array new: self
paneCount.
  padded atAllPut: (limit / self
sizing) floor.
  (1 to: limit - padded sum) do:
[:i | padded at: i put: (padded at:
i) + 1].
  ^ padded
```



# Node-based Critiques

a CompiledMethod (GLMPaneScroller>>#pane...)

... 1 I... S... Ir AST He... Meta

- ▼ RBMethodNode(paneWidthsToFit: limit | padded |
  - RBArgumentNode(limit)
  - ▼ RBSequenceNode(| padded | padded := Array ne
    - RBTemporaryNode(padded)
    - ▶ RBAssignmentNode(padded := Array new: sel
    - ▶ RBMessageNode(padded atAllPut: (limit / sel
    - ▶ RBMessageNode((1 to: limit - padded sum) c
    - ▶ RBReturnNode(^ padded)



# Node-based Critiques

The screenshot displays two side-by-side windows from a software development environment. The left window, titled "AST", shows the abstract syntax tree for a compiled method. The tree structure is as follows:

- ▼ RBMethodNode(paneWidthsToFit: limit | padded |
  - RBArgumentNode(limit)
  - ▼ RBSequenceNode(| padded | padded := Array new
    - RBTemporaryNode(padded)
    - ▶ RBAssignmentNode(padded := Array new: sel
    - ▶ RBMessageNode(padded atAllPut: (limit / sel
    - ▶ RBMessageNode((1 to: limit - padded sum) c
    - ▶ RBReturnNode(^ padded)

The right window, titled "Playground", shows a critique message: "Uses (to:)do: instead of to:do:". The message is preceded by a yellow warning triangle icon. The playground window also has a toolbar with icons for "Sc...", "Tree", "Sco...", and "Meta".



# Node-based Critiques

Playground

a RBMessageNode (RBMessageNode((1 to: limit - f...)

... 1 Sc... Tree Sco... Meta

⚠ Uses (to:)do: instead of to:do:

a ReTrivialCritique ×

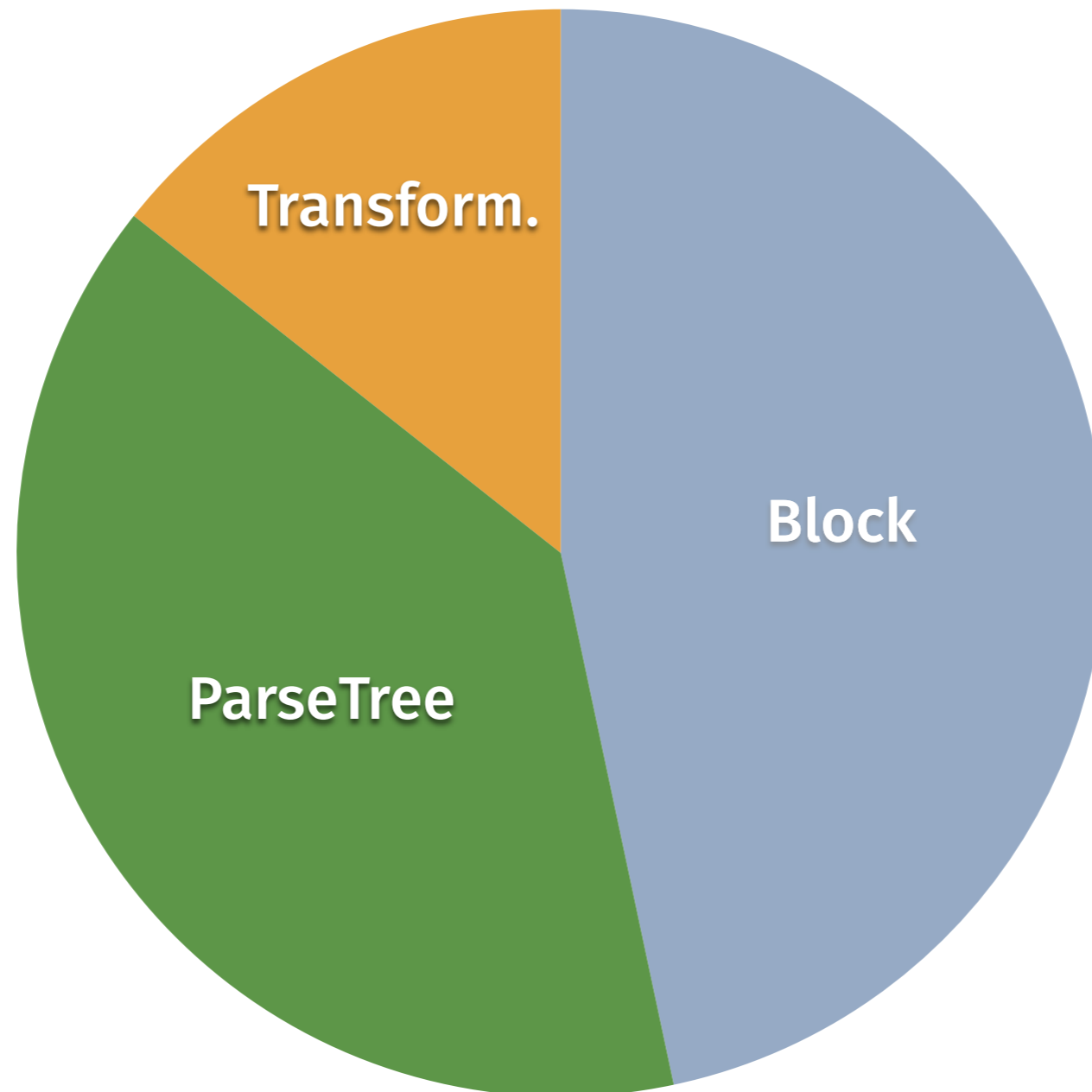
Sou... Descri... Raw Meta

```
paneWidthsToFit: limit
  | padded |
  padded := Array new: self
paneCount.
  padded atAllPut: (limit / self
sizing) floor.
  (1 to: limit - padded sum) do:
  [:i | padded at: i put: (padded at:
i) + 1].
    ^ padded
```





# Interval Support



# The Point of No Return







# CriticBrowser on Renraku

The screenshot shows the CriticBrowser application window titled "Sends a deprecated message to a known global". The window is divided into several sections:

- Design Flaws (To sort: 884, ToDo: 0, Wrong: 0):**
  - Method defined in all subclasses, but not in superclass
  - Sends a deprecated message to a known global (To sort: 884, ToDo: 0, Wrong: 0)** (highlighted)
  - Same instance variable defined in ALL subclasses (To sort: 884, ToDo: 0, Wrong: 0)
- Optimization (To sort: 353, ToDo: 0, Wrong: 0):**
  - Temporary variables not read AND written (To sort: 353, ToDo: 0, Wrong: 0)
- SUnit (To sort: 306, ToDo: 0, Wrong: 0)**

Below the list is a toolbar with buttons: "Run new configuration", "Save Critics", "Browse", "Transform", and "Mark as wrong".

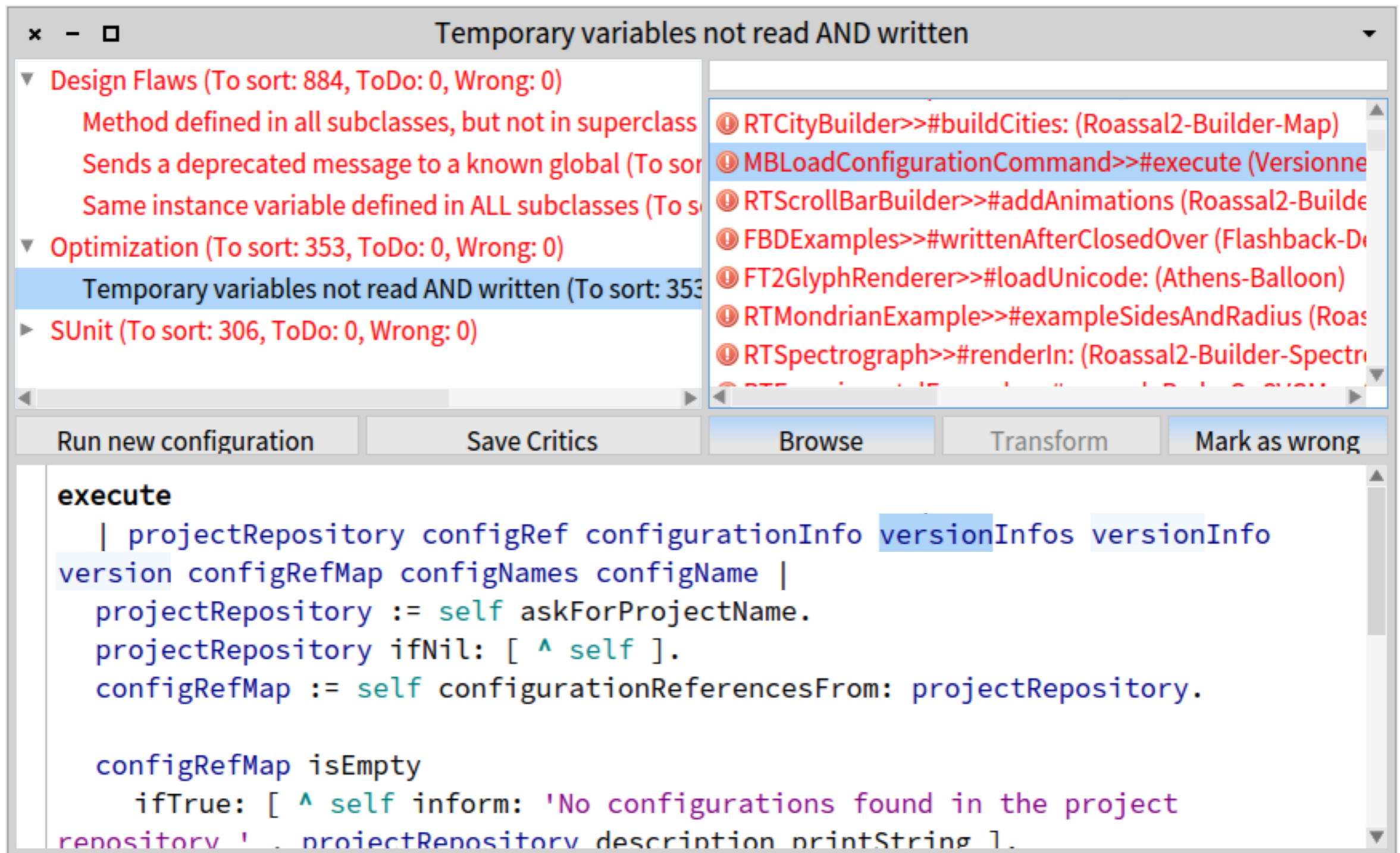
The main content area contains the following text:

```
Checks for sends of deprecated messages that might be removed in upcoming releases of Pharo.
```

On the right side of the window, there is a list of specific code issues, each preceded by a red warning icon:

- Form>>#pageWarp:at:forward: (Graphics-Display Objec
- Form>>#pageWarp:at:forward: (Graphics-Display Objec
- FormCanvas>>#privateWarp:transform:at:sourceRect:c
- Form>>#displayResourceFormOn: (Graphics-Display Ob
- Form>>#rotateBy:magnify:smoothing: (Graphics-Displa
- RTMondrianViewBuilder>>#edges:from:toAll: (Roassal2
- Form>>#rotateBy:magnify:smoothing: (Graphics-Displa
- Form>>#rotateBy:centerAt: (Graphics-Display Object)

# Old Highlight



The screenshot shows a window titled "Temporary variables not read AND written" with a list of design flaws on the left and a code editor on the right. The code editor shows the implementation of the `execute` method, with several lines highlighted in blue.

**Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)**

- Method defined in all subclasses, but not in superclass
- Sends a deprecated message to a known global (To sort: ...)
- Same instance variable defined in ALL subclasses (To sort: ...)

**Optimization (To sort: 353, ToDo: 0, Wrong: 0)**

- Temporary variables not read AND written (To sort: 353)

**SUnit (To sort: 306, ToDo: 0, Wrong: 0)**

**Code Snippet:**

```
execute
  | projectRepository configRef configurationInfo versionInfos versionInfo
  version configRefMap configNames configName |
  projectRepository := self askForProjectName.
  projectRepository ifNil: [ ^ self ].
  configRefMap := self configurationReferencesFrom: projectRepository.

  configRefMap isEmpty
    ifTrue: [ ^ self inform: 'No configurations found in the project
  repository ' . projectRepository description printString ].
```





# Highlight Based on AST

Temporary variables not read AND written

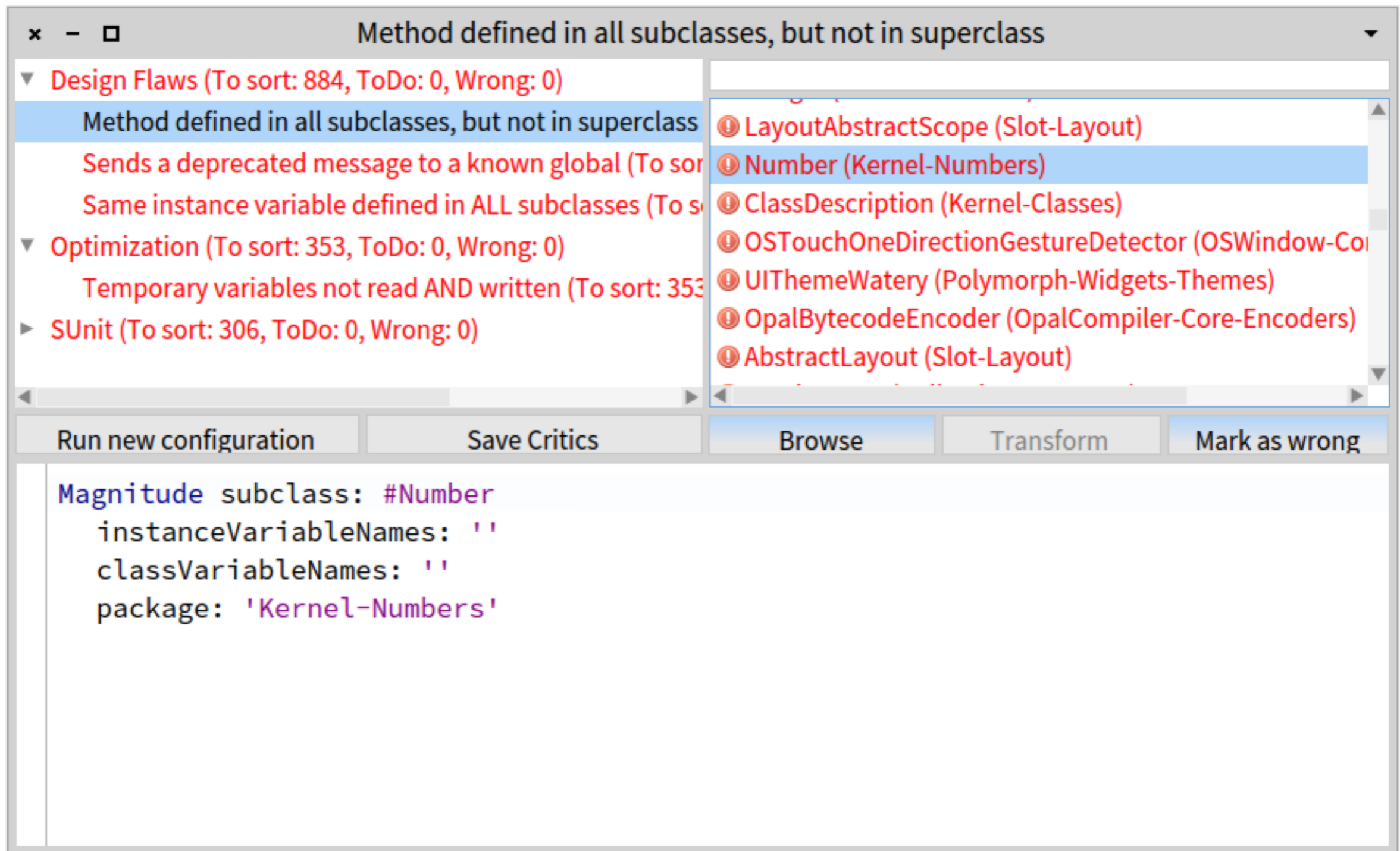
- Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)
  - Method defined in all subclasses, but not in superclass
  - Sends a deprecated message to a known global (To sort: ...)
  - Same instance variable defined in ALL subclasses (To sort: ...)
- Optimization (To sort: 353, ToDo: 0, Wrong: 0)
  - Temporary variables not read AND written (To sort: 353)**
- SUnit (To sort: 306, ToDo: 0, Wrong: 0)

Run new configuration   Save Critics   Browse   Transform   Mark as wrong

```
execute
  | projectRepository configRef configurationInfo versionInfos versionInfo
  version configRefMap configNames configName |
  projectRepository := self askForProjectName.
  projectRepository ifNil: [ ^ self ].
  configRefMap := self configurationReferencesFrom: projectRepository.

  configRefMap isEmpty
    ifTrue: [ ^ self inform: 'No configurations found in the project
  repository ' . projectRepository description printString ].
```

# Useless Output



The screenshot shows an IDE window titled "Method defined in all subclasses, but not in superclass". The window is divided into two main sections. The top section is a list of design flaws, and the bottom section is a detailed view of a selected flaw.

**Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)**

- Method defined in all subclasses, but not in superclass
  - Sends a deprecated message to a known global (To sort: 884, ToDo: 0, Wrong: 0)
  - Same instance variable defined in ALL subclasses (To sort: 884, ToDo: 0, Wrong: 0)
- Optimization (To sort: 353, ToDo: 0, Wrong: 0)
  - Temporary variables not read AND written (To sort: 353, ToDo: 0, Wrong: 0)
- SUnit (To sort: 306, ToDo: 0, Wrong: 0)

**Method defined in all subclasses, but not in superclass**

- LayoutAbstractScope (Slot-Layout)
- Number (Kernel-Numbers)
- ClassDescription (Kernel-Classes)
- OSTouchOneDirectionGestureDetector (OSWindow-Corner)
- UIThemeWatery (Polymorph-Widgets-Themes)
- OpalBytecodeEncoder (OpalCompiler-Core-Encoders)
- AbstractLayout (Slot-Layout)

Run new configuration    Save Critics    Browse    Transform    Mark as wrong

```
Magnitude subclass: #Number
instanceVariableNames: ''
classVariableNames: ''
package: 'Kernel-Numbers'
```



# Custom Autofix

Method defined in all subclasses, but not in superclass

▼ Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)

- Method defined in all subclasses, but not in superclass
- Sends a deprecated message to a known global (To sort: ...)
- Same instance variable defined in ALL subclasses (To sort: ...)

▼ Optimization (To sort: 353, ToDo: 0, Wrong: 0)

- Temporary variables not read AND written (To sort: 353)

► SUnit (To sort: 306, ToDo: 0, Wrong: 0)

LayoutAbstractScope (Slot-Layout)

Number (Kernel-Numbers)

ClassDescription (Kernel-Classes)

OSTouchOneDirectionGestureDetector (OSWindow-Co)

UIThemeWatery (Polymorph-Widgets-Themes)

OpalBytecodeEncoder (OpalCompiler-Core-Encoders)

AbstractLayout (Slot-Layout)

Run new configuration Save Critics Browse Transform Mark as wrong

```
asFraction
  ^ self subclassResponsibility
```



# Autofix by Refactoring...

The screenshot shows the 'Problems' window in an IDE. The title bar reads 'Same instance variable defined in ALL subclasses'. The left pane lists categories: 'Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)', 'Optimization (To sort: 353, ToDo: 0, Wrong: 0)', and 'SUnit (To sort: 306, ToDo: 0, Wrong: 0)'. Under 'Design Flaws', the issue 'Same instance variable defined in ALL subclasses (To sort: 884, ToDo: 0, Wrong: 0)' is selected. The right pane lists affected classes: 'AbstractCompiler (OpalCompiler-Core-FrontEnd)', 'AbstractFileReference (FileSystem-Core-Public)', 'GLMBrickLayoutTraverser class (Glamour-Morphic-Bric...', 'RTVisito...', 'Abstrac...', 'AthensF...', and 'MCVersi...'. A context menu is open over the selected issue, with options: 'Browse', 'References to...', 'Mark as wrong', 'Mark as TODO', 'View rationale behind the rule', 'Automatically resolve the issue' (highlighted), and 'Ban this rule'. At the bottom, a green button labeled 'a RBPullUpInstanceVariableRefactoring' is visible.





# Same Preview as in QA

Same instance variable defined in ALL subclasses

Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)

- Method defined in all sub
- Sends a deprecated mes
- Same instance variable c

Optimization (To sort: 353, ...)

- Temporary variables not

SUnit (To sort: 306, ToDo: 0, ...)

Run new configuration

a RBPullUpInstanceV

Changes Browser

- Remove instance variable stream from RTJavascriptVisitor
- Remove instance variable stream from RTSVGVisitor
- Add instance variable stream to RTVisitor

Mark as wrong

Ok Cancel



# Autofix by Refactoring...

The screenshot shows the 'Problems' window in an IDE. The title bar reads 'Same instance variable defined in ALL subclasses'. The left pane lists categories: 'Design Flaws (To sort: 884, ToDo: 0, Wrong: 0)', 'Optimization (To sort: 353, ToDo: 0, Wrong: 0)', and 'SUnit (To sort: 306, ToDo: 0, Wrong: 0)'. Under 'Design Flaws', the issue 'Same instance variable defined in ALL subclasses (To sort: 884, ToDo: 0, Wrong: 0)' is selected. The right pane lists affected classes: 'AbstractCompiler (OpalCompiler-Core-FrontEnd)', 'AbstractFileReference (FileSystem-Core-Public)', 'GLMBrickLayoutTraverser class (Glamour-Morphic-Bric...', 'RTVisito...', 'Abstrac...', 'AthensF...', and 'MCVers...'. A context menu is open over the selected issue, with options: 'Browse', 'References to...', 'Mark as wrong', 'Mark as TODO', 'View rationale behind the rule', 'Automatically resolve the issue', and 'Ban this rule'. The 'Automatically resolve the issue' option is highlighted in blue and circled in red. Below the list are buttons for 'Run new configuration', 'Save Critics', and 'Browse...'. At the bottom, a green link 'a RBPullUpInstanceVariableRefactoring' is visible.



# Architectural Action

```
RePropertyAction new
  icon: #smallWindow asIcon;
  description: 'Open dependency analyzer';
  action: [ :crit :package |
    self openDependencyAnalyzerOn: package ];
  yourself
```



# Architectural Action

Illegal dependency of Bootstrap Layer

, Wrong: 0)

rap Layer (To sort: 1, ToDo: 0, Wro

**Kernel**

- Browse
- Senders
- Implementors
- Mark as wrong
- Mark as TODO
- View rationale behind the rule
- Ban this rule
- Open dependency analyzer**

Save Critics

Brow:

Mark as wrong





# Architectural Action

The screenshot displays an IDE window titled "Illegal dependency of Bootstrap Layer". The main area shows a table of architectural errors. The first row is highlighted in blue and contains the text "Kernel" with a red exclamation mark icon. A context menu is open over this row, listing several actions: "Browse", "Senders", "Implementors", "Mark as wrong", "Mark as TODO", "View rationale behind the rule", "Ban this rule", and "Open dependency analyzer". The "Open dependency analyzer" option is currently selected and highlighted in blue. Below the table, there are buttons for "Save Critics", "Browse", and "Mark as wrong". At the bottom of the IDE, the status bar shows "1/4 [1]" and a button labeled "Open dependency analyzer". A yellow warning icon is visible next to the error text in the status bar.

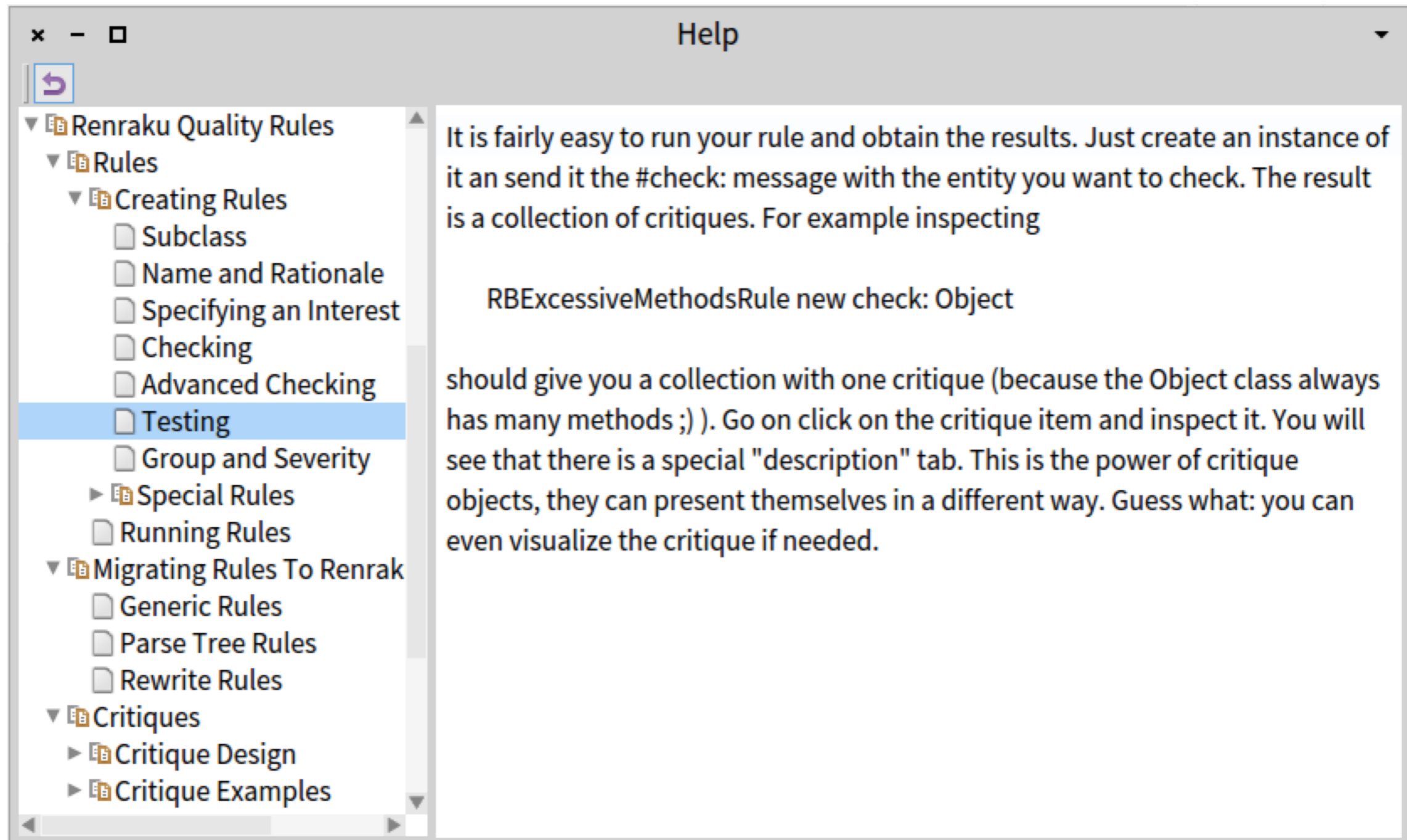


# Documentation :)





# System's Help Browser



Here's to the Contributors!



Here's to the Contributors!



Marcus Denker

# Here's to the Contributors!



Marcus Denker



Nicolai Hess



Henrik Nergaard

# Here's to the Contributors!



Marcus Denker



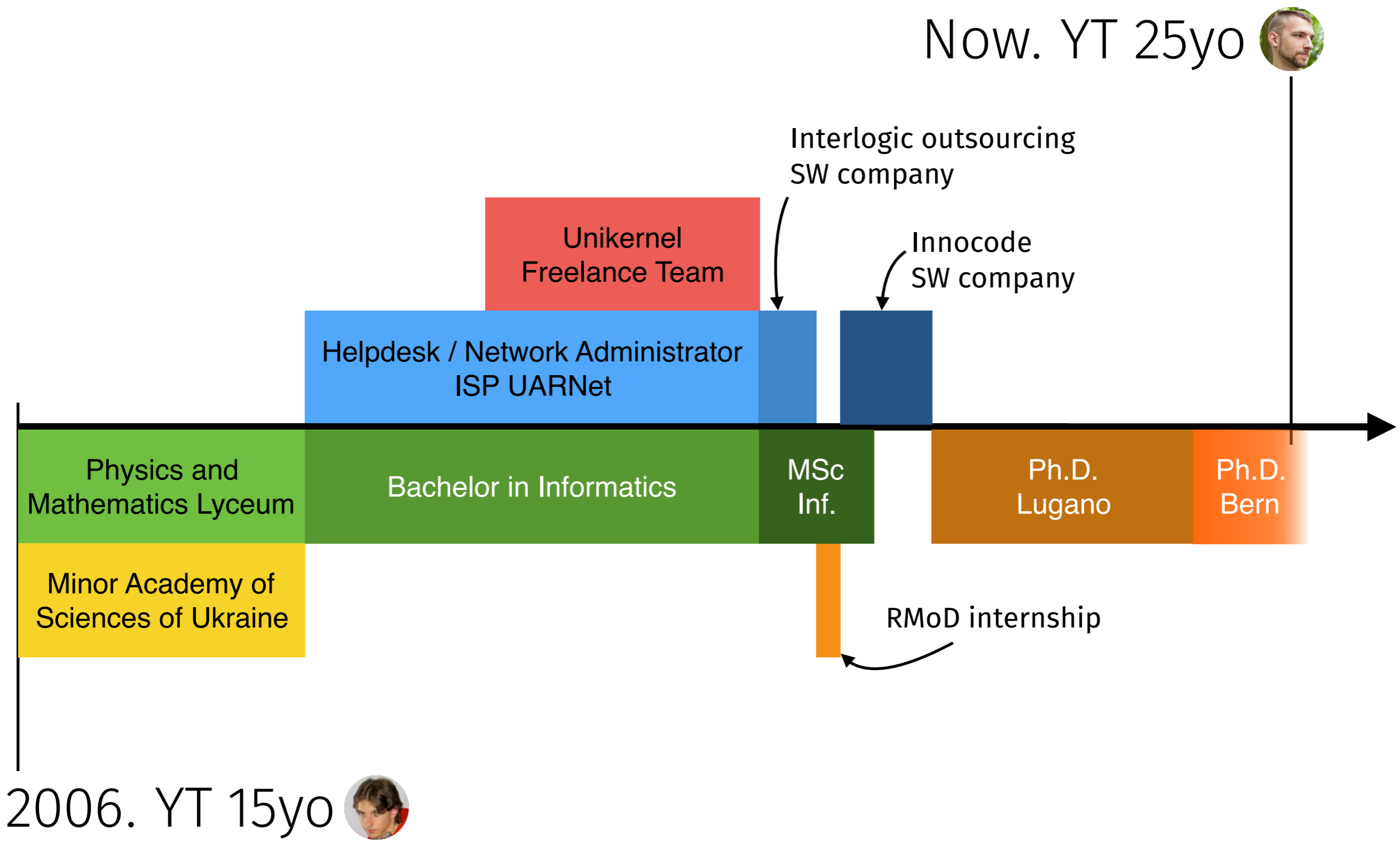
Nicolai Hess



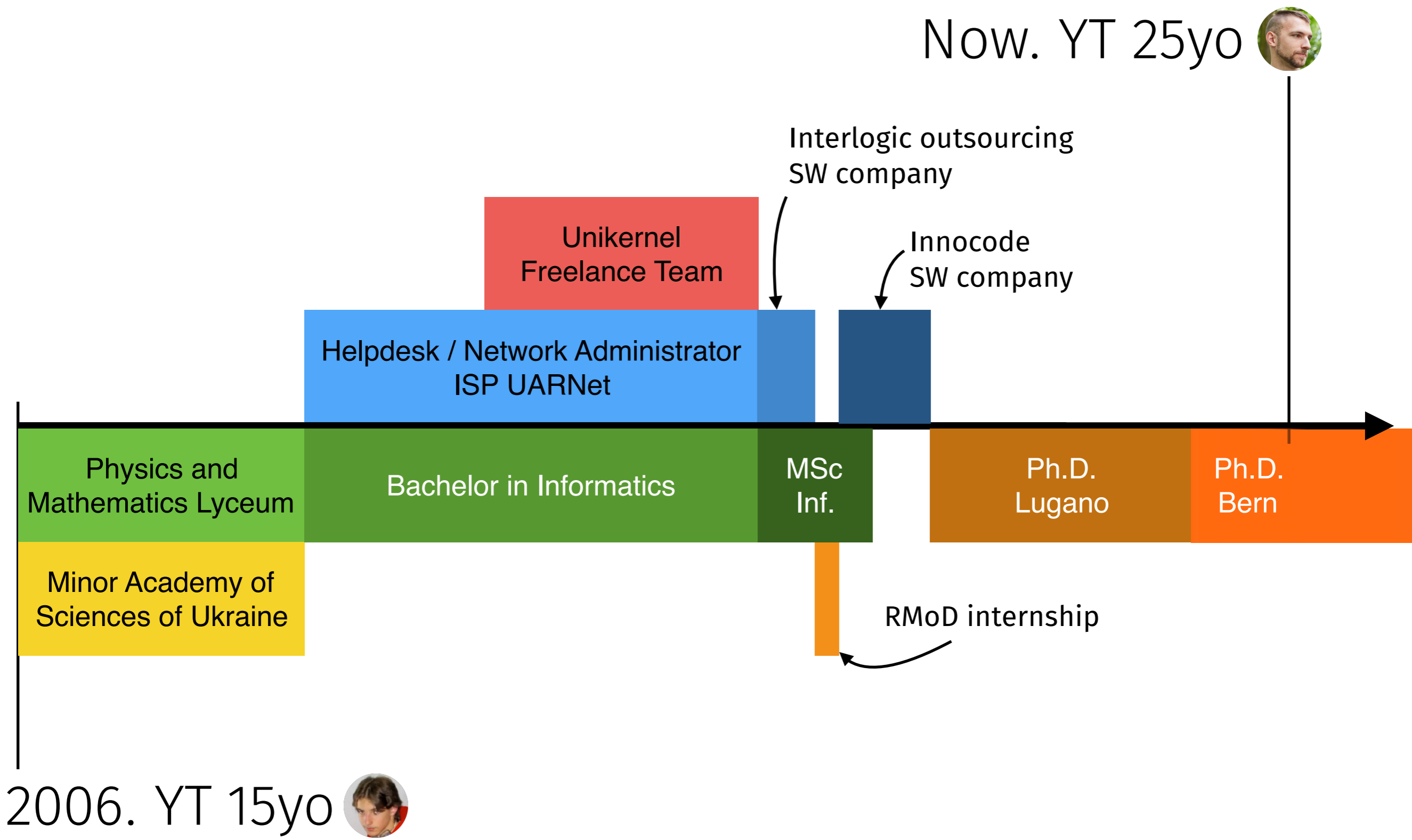
Henrik Nergaard

and others...









Now. YT 25yo 

Unikernel  
Freelance Team

Helpdesk / Network Administrator  
ISP UARNet

Interlogic outsourcing  
SW company

Innocode  
SW company

Physics and  
Mathematics Lyceum

Bachelor in Informatics

MSc  
Inf.

Ph.D.  
Lugano

Ph.D.  
Bern

Minor Academy of  
Sciences of Ukraine

RMoD internship

2006. YT 15yo 

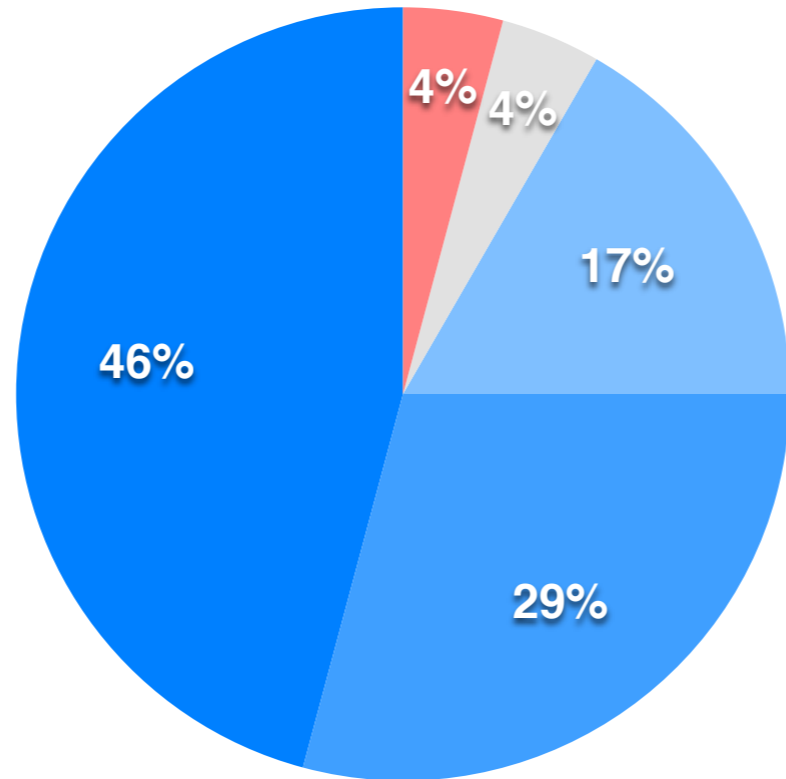




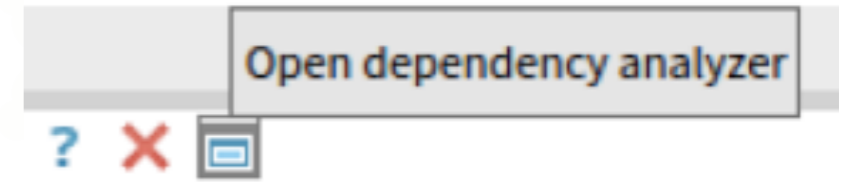
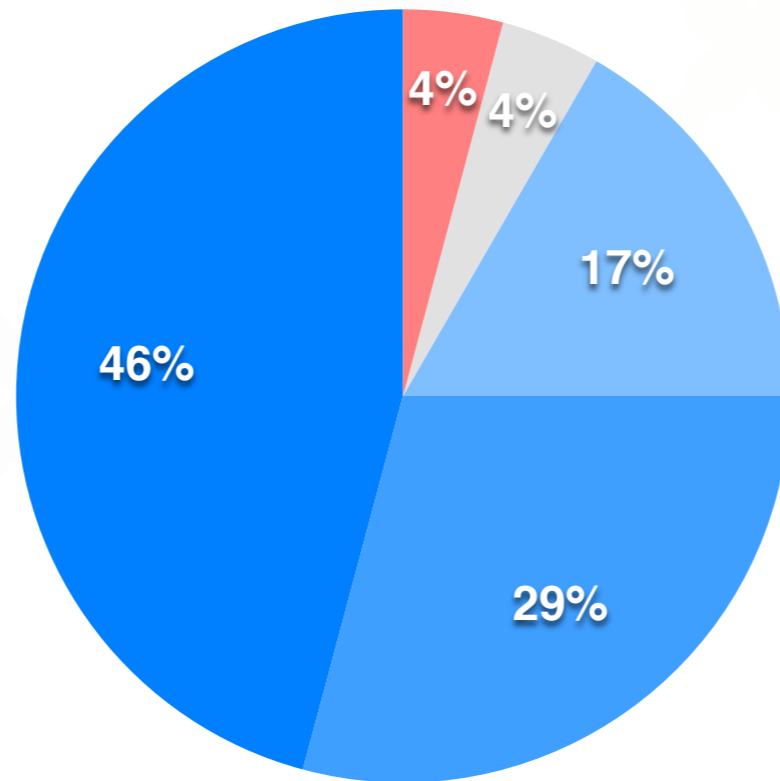












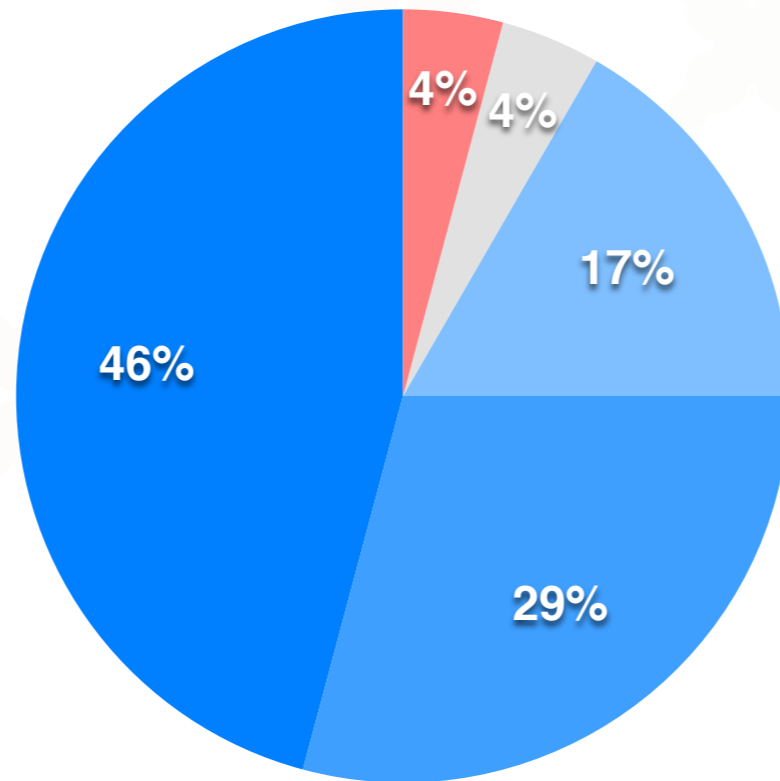
Actionability

Jenkins Job jenkins-Pharo-6.0-Issue-Validator-8864

### Console Output

```
The method Pragma>>selector has been deprecated.
```

Critiques in Monkey



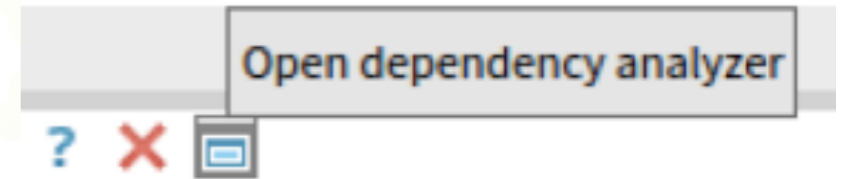
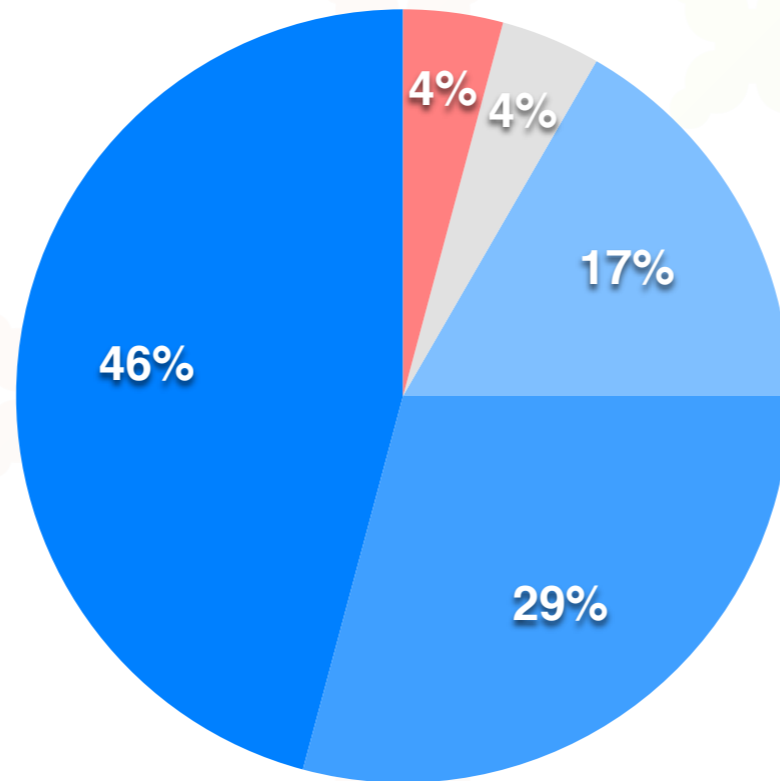
Open dependency analyzer

? X [icon]

Actionability

```
Jenkins Job jenkins-Pharo-6.0-Issue-Validator-8864  
Console Output  
The method Pragma>>selector has been deprecated.
```

Critiques in Monkey



Actionability

```
! (self respondsTo: #actionSe  
  ifTrue: ["A button"  
    selector := self actio  
! [target] Super and Self Messages sent but r
```

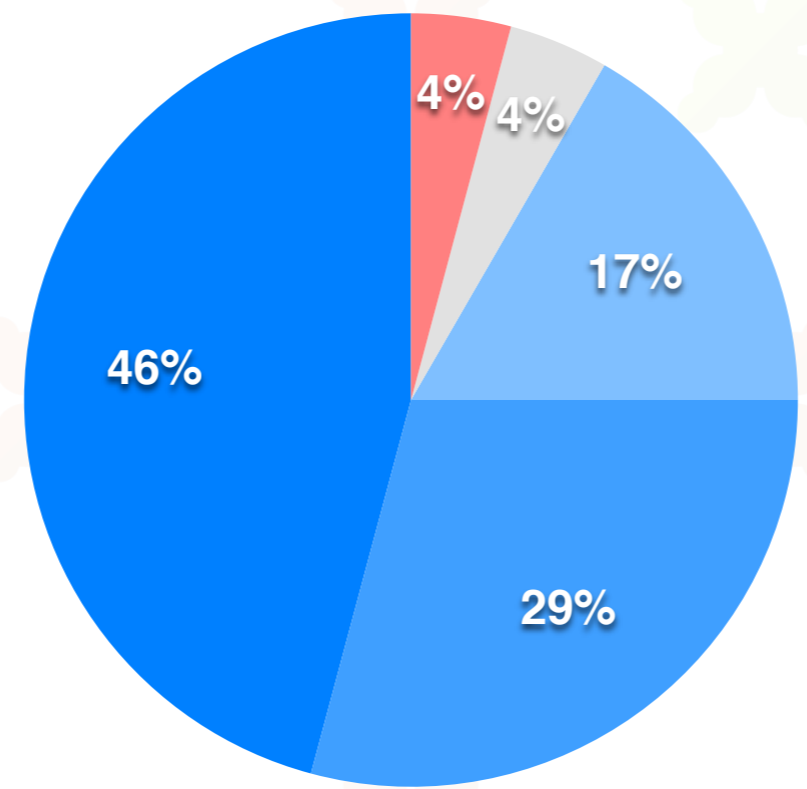
Better Feedback

Jenkins Job jenkins-Pharo-6.0-Issue-Validator-8864

### Console Output

```
The method Pragma>>selector has been deprecated.
```

Critiques in Monkey



Open dependency analyzer

?
X

Actionability

🟢 Issue #10 with 3 entities	🔔	🗑️	+
🟢 Issue #15 with 4 entities	🔔	🗑️	+
🟢 Issue #19 with 2 entities	🔔	🗑️	+

Code Properties

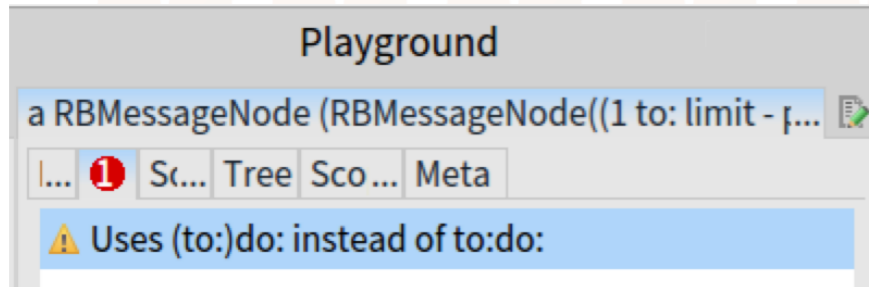
```

❗ (self respondsTo: #actionSe
    ifTrue: ["A button"
        selector := self actio
❗ [target] Super and Self Messages sent but r

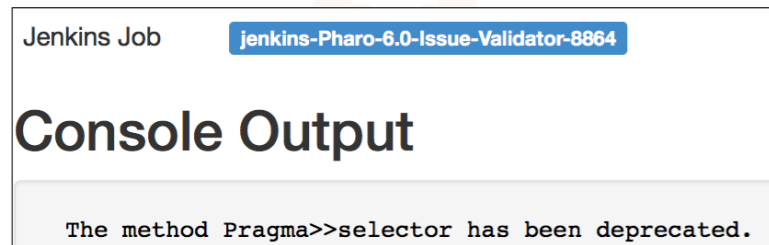
```

Better Feedback

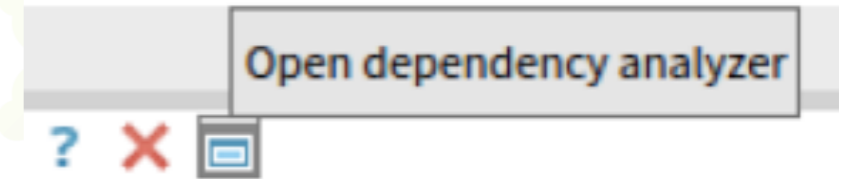
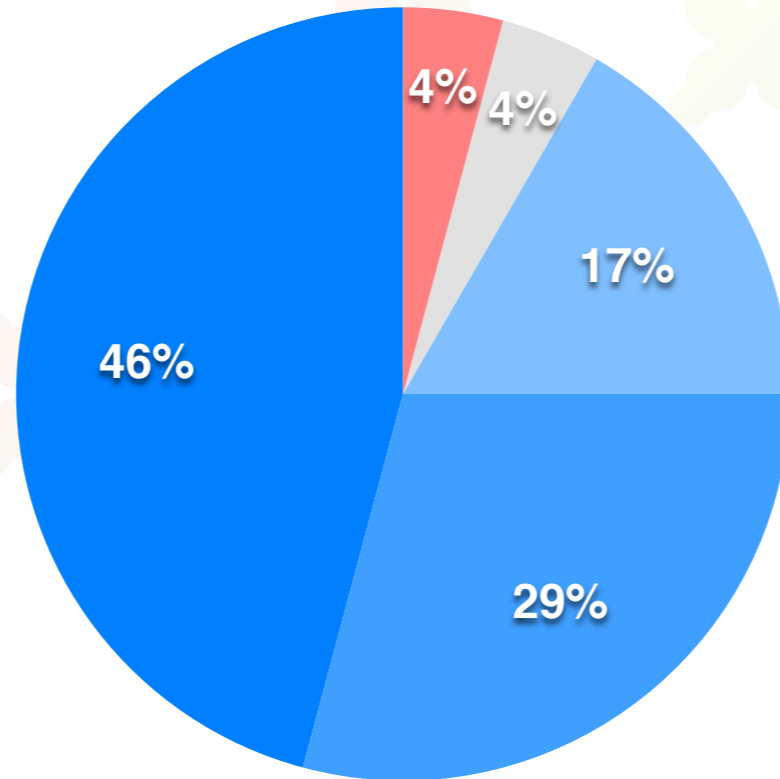




Object Critiques



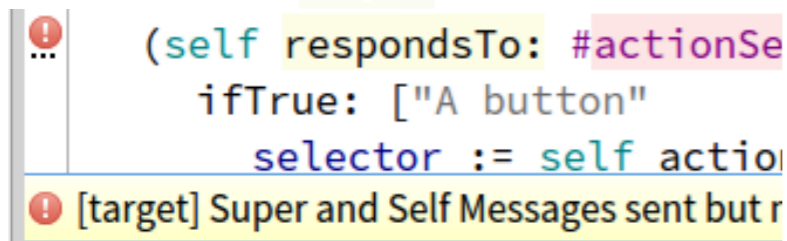
Critiques in Monkey



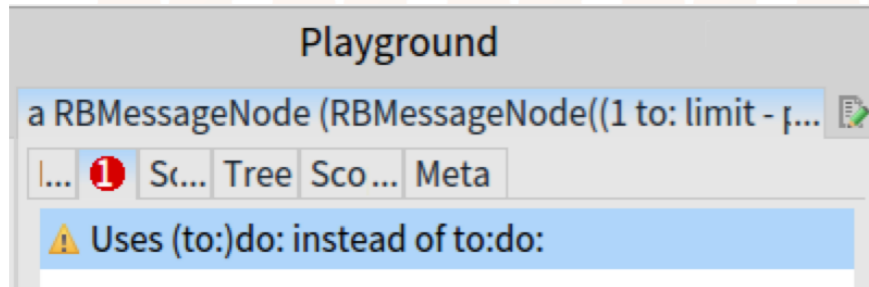
Actionability



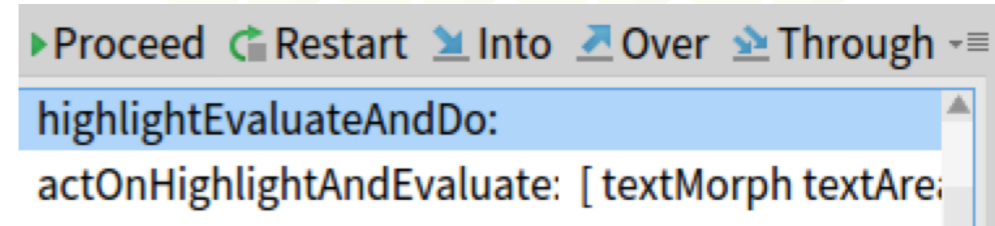
Code Properties



Better Feedback



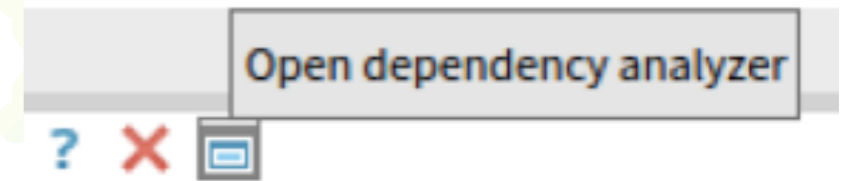
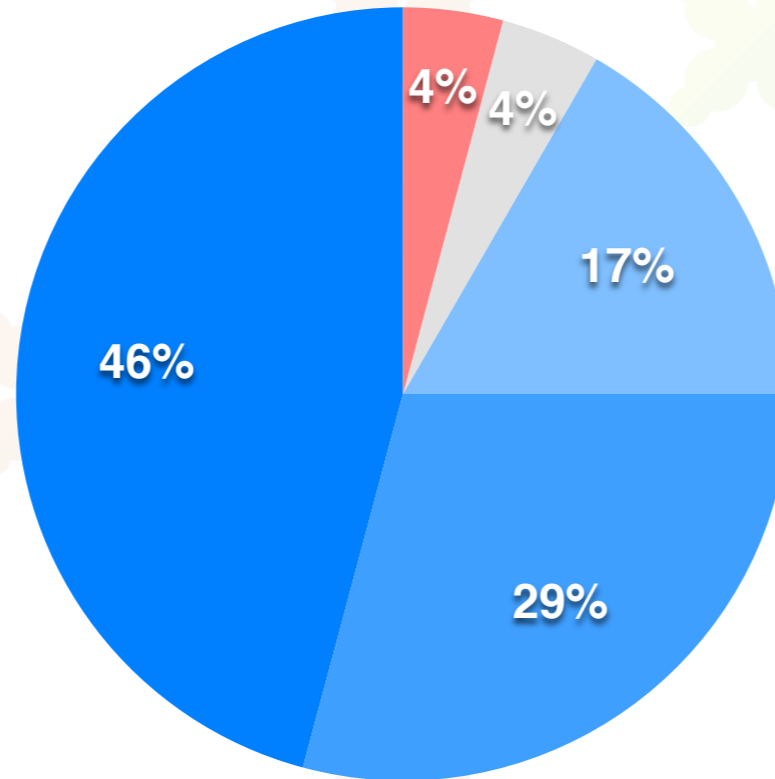
Object Critiques



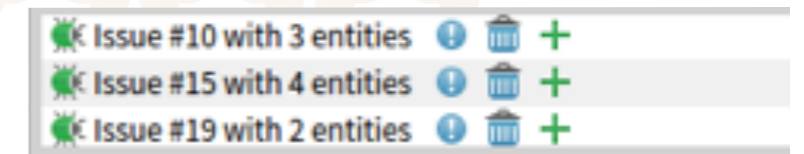
Critiques in Debugger



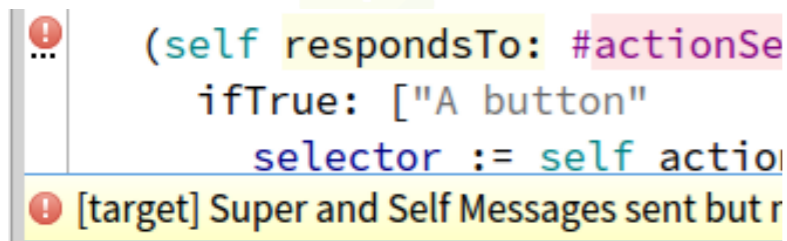
Critiques in Monkey



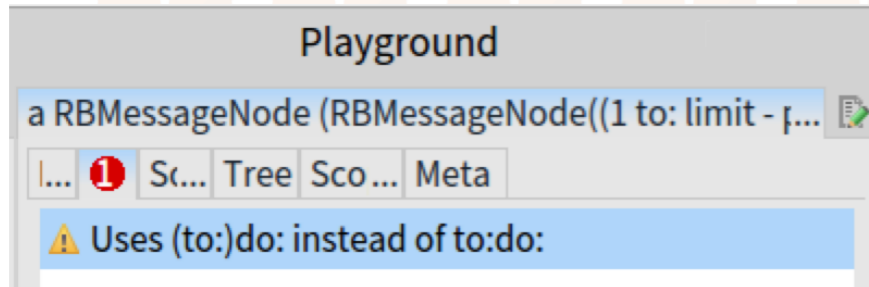
Actionability



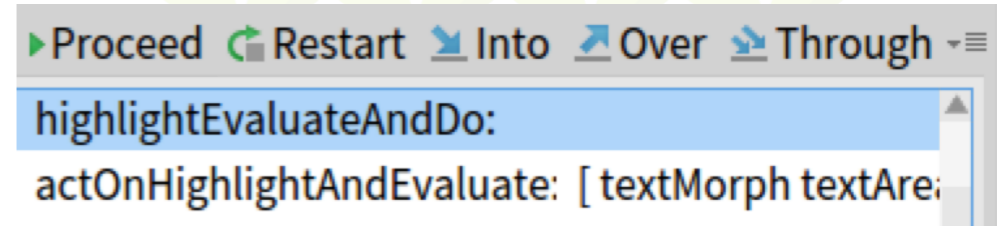
Code Properties



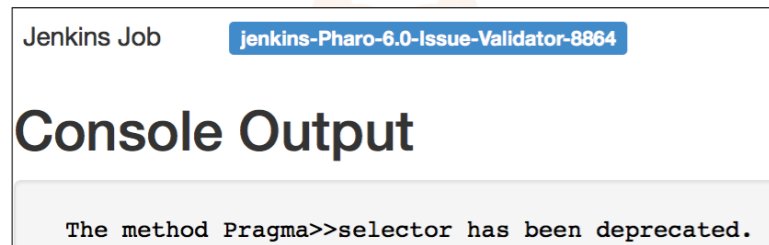
Better Feedback



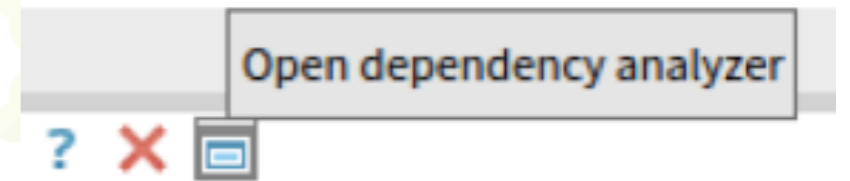
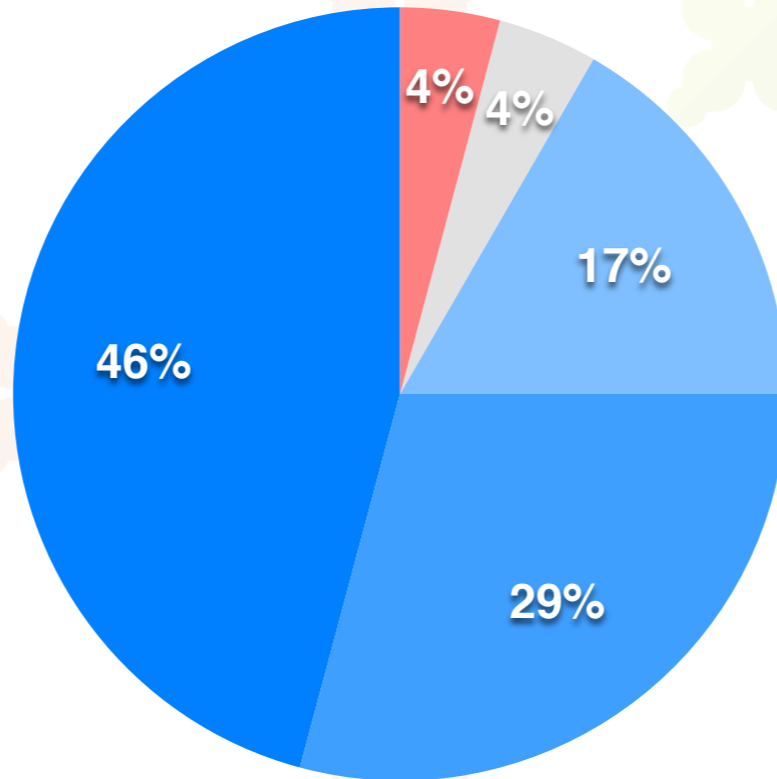
Object Critiques



Critiques in Debugger



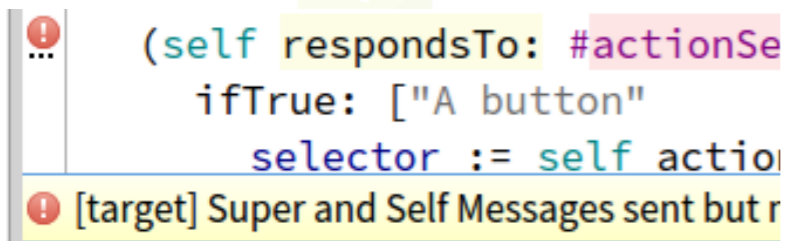
Critiques in Monkey



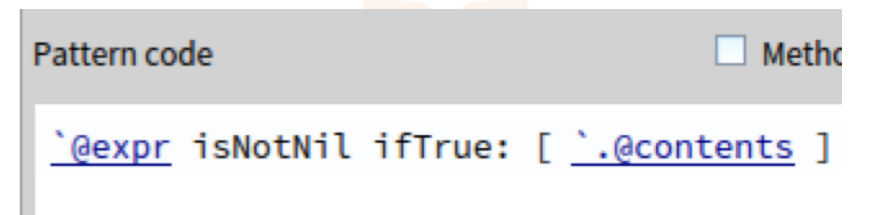
Actionability



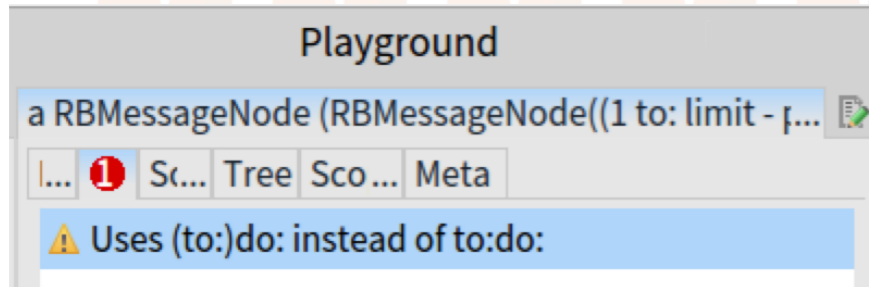
Code Properties



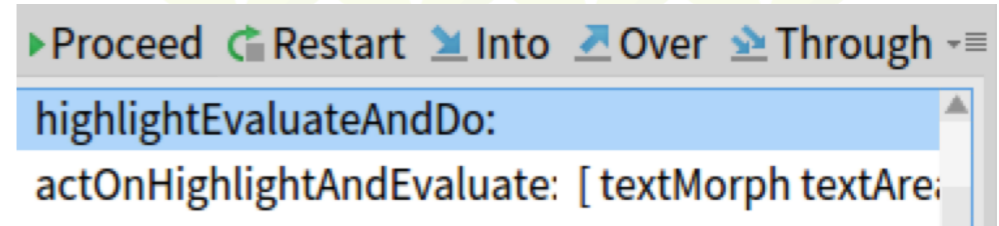
Better Feedback



Rule Developer-friendliness



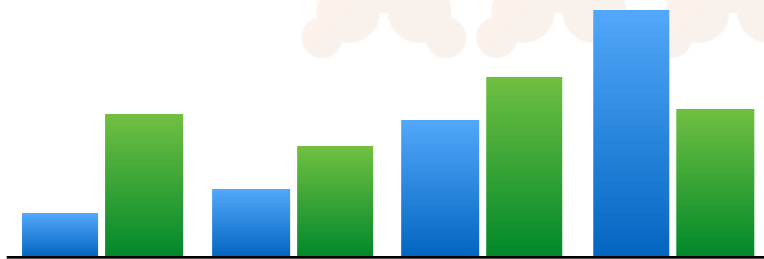
Object Critiques



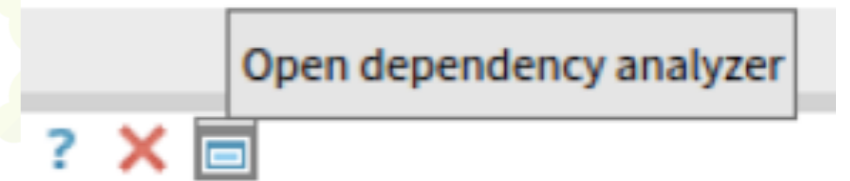
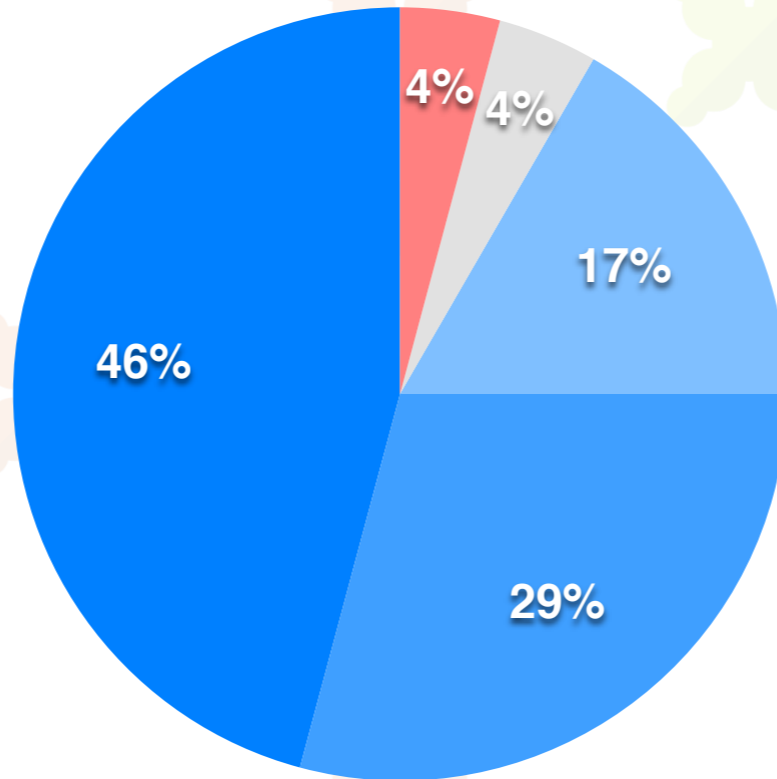
Critiques in Debugger



Critiques in Monkey



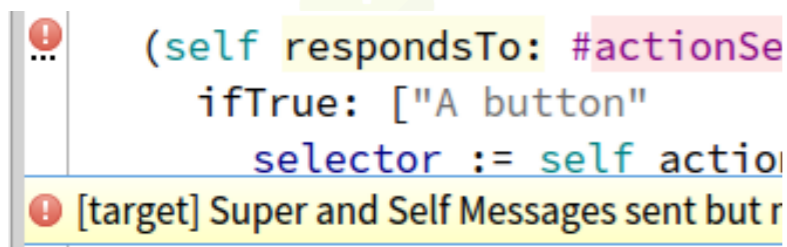
Usage Analysis



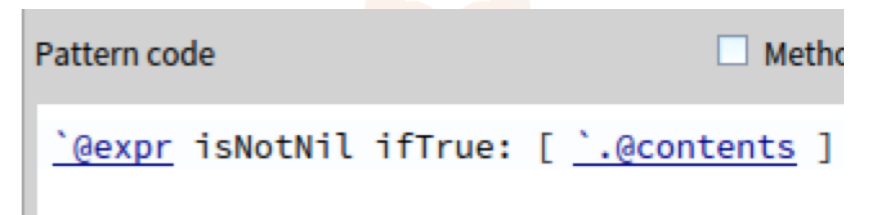
Actionability



Code Properties

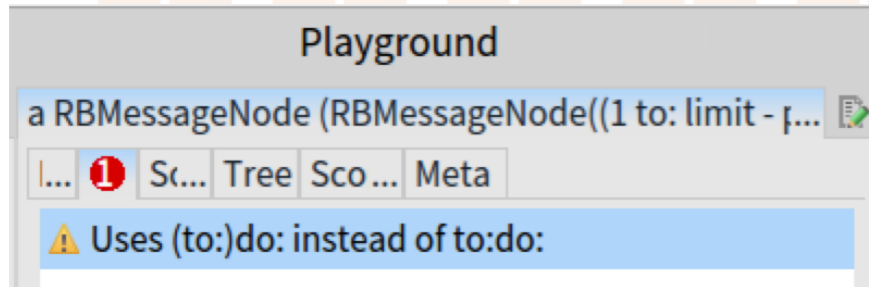


Better Feedback

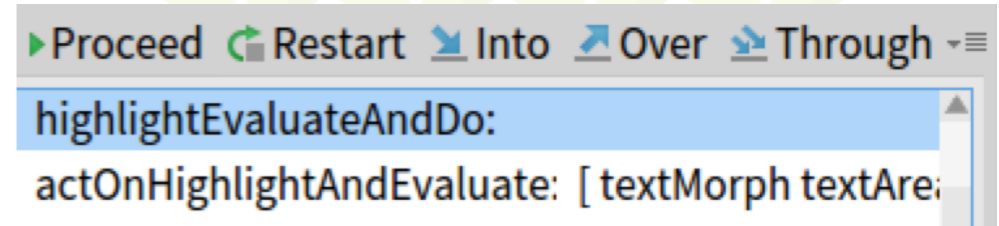


Rule Developer-friendliness

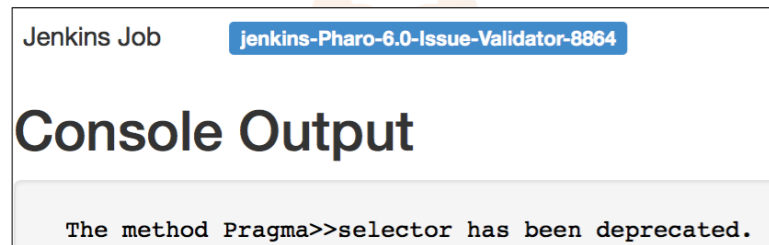




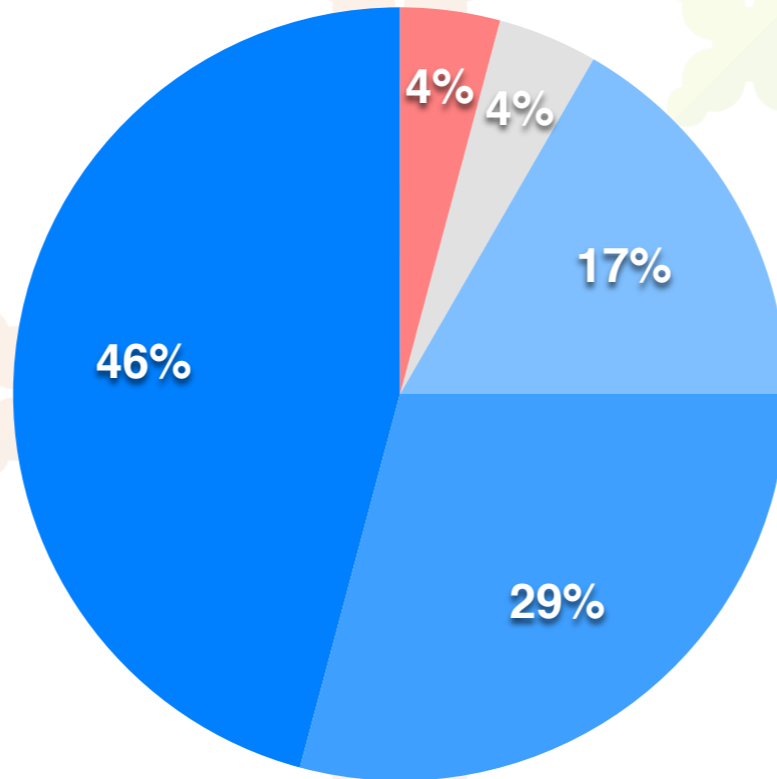
Object Critiques



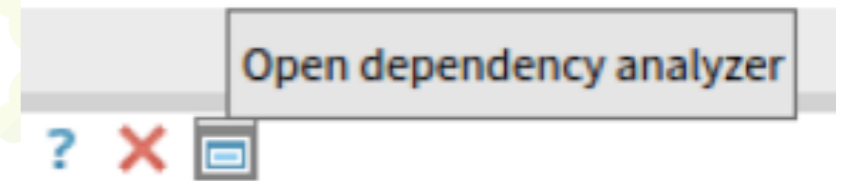
Critiques in Debugger



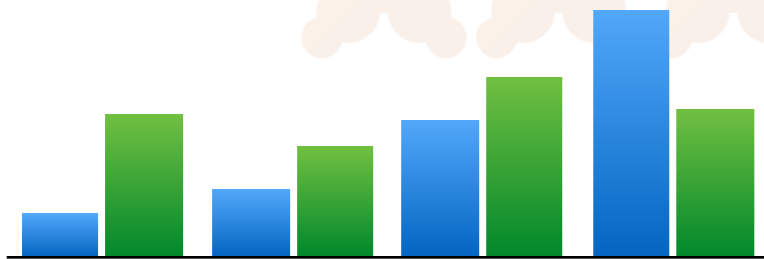
Critiques in Monkey



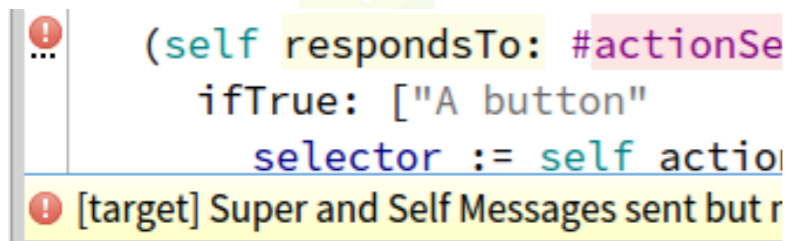
Usage Analysis



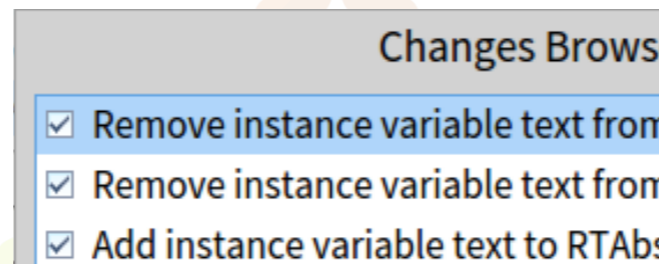
Actionability



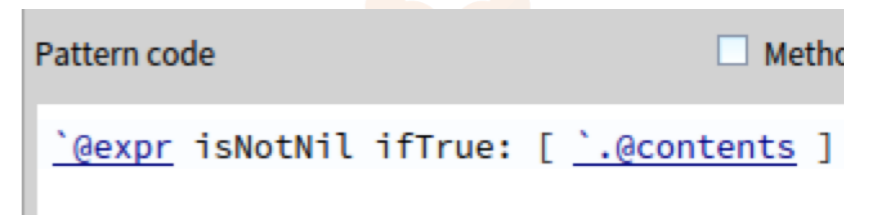
Code Properties



Better Feedback



Continuous Refactoring



Rule Developer-friendliness

# Stories About Renraku —

the new Quality Model of Pharo

tool-agnostic

object-agnostic

engine-agnostic

yet connecting  
everything

R  
E  
N  
R  
A  
K  
U

連  
絡