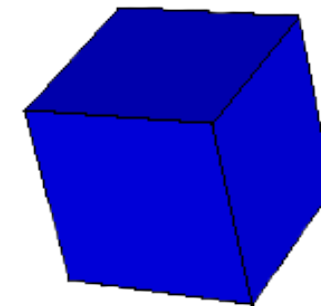
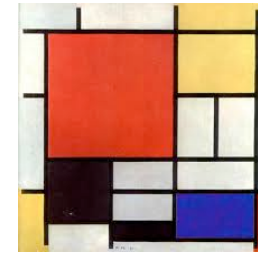


Object-centric profiling: Advanced Visualizations to Tame Wild Program Execution

Vanessa Peña, Juan Pablo Sandoval, Pablo Estefo,
Alexandre Bergel

Object Profile & University of Chile







Execution profiling with Kai

Problem:

Traditional code profilers are driven by the method stack, discarding the notion of sending messages

Why the problem is important:

How to answer to “Is there a slow function that is called too often?”

Solution:

An intuitive visual representation of the execution that visually compare the time spent and the number of executions

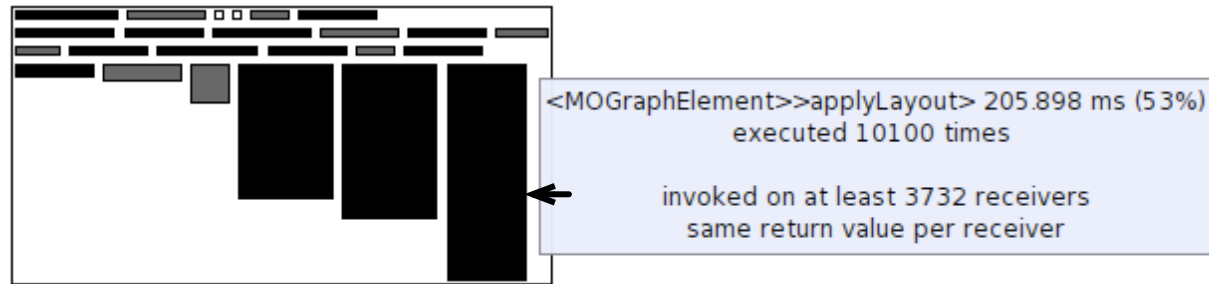


Execution sampling profiler

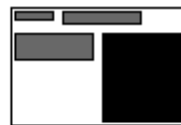
```
54.8% {11501ms} MOCanvas>>drawOn:  
  54.8% {11501ms} MORoot(MONode)>>displayOn:  
    30.9% {6485ms} MONode>>displayOn:  
      | 18.1% {3799ms} MOEdge>>displayOn:  
        ...  
      | 8.4% {1763ms} MOEdge>>displayOn:  
        | 8.0% {1679ms} MOStraightLineShape>>display:on:  
          | 2.6% {546ms} FormCanvas>>line:to:width:color:  
            ...  
    23.4% {4911ms} MOEdge>>displayOn:  
      ...
```



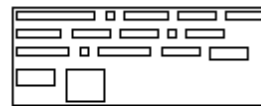
Structural profiling blueprint



MOGraphElement



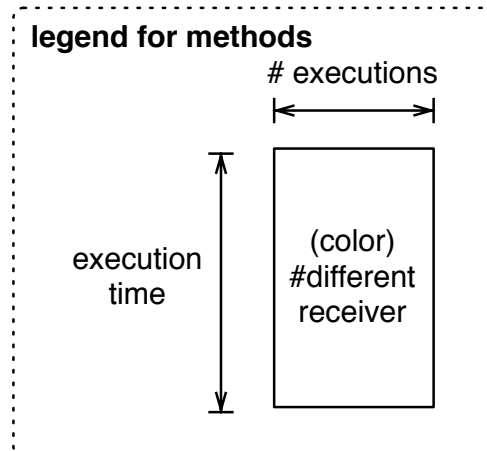
MONode



MOViewRenderer

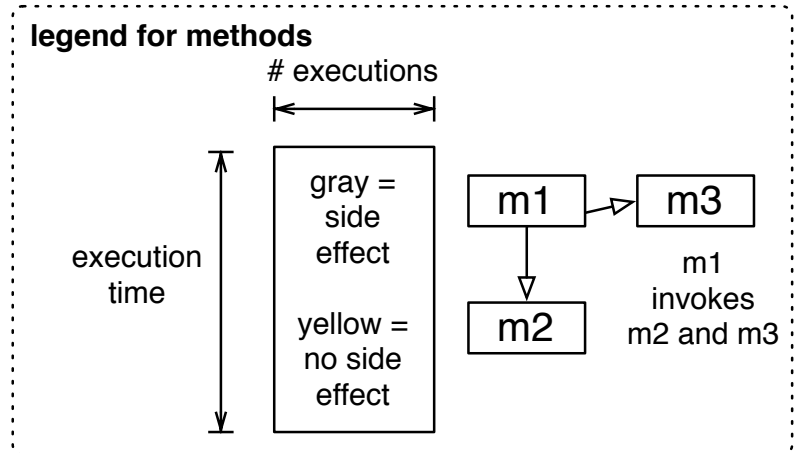
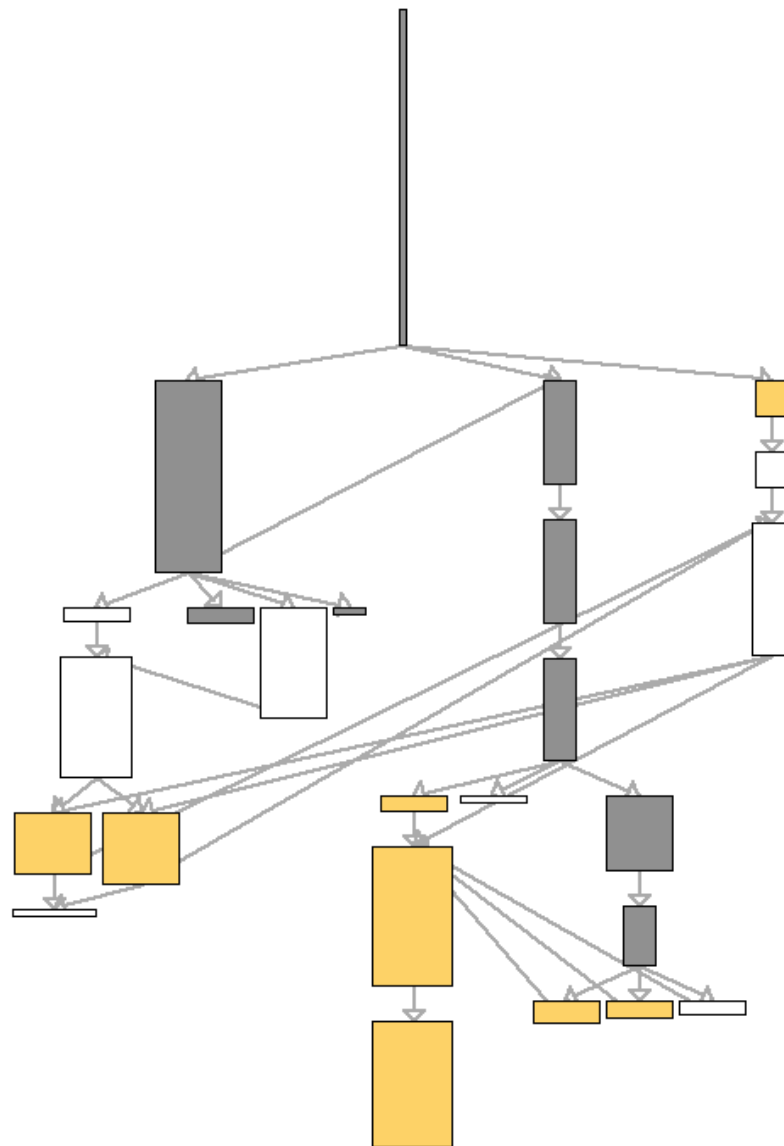


MORoot





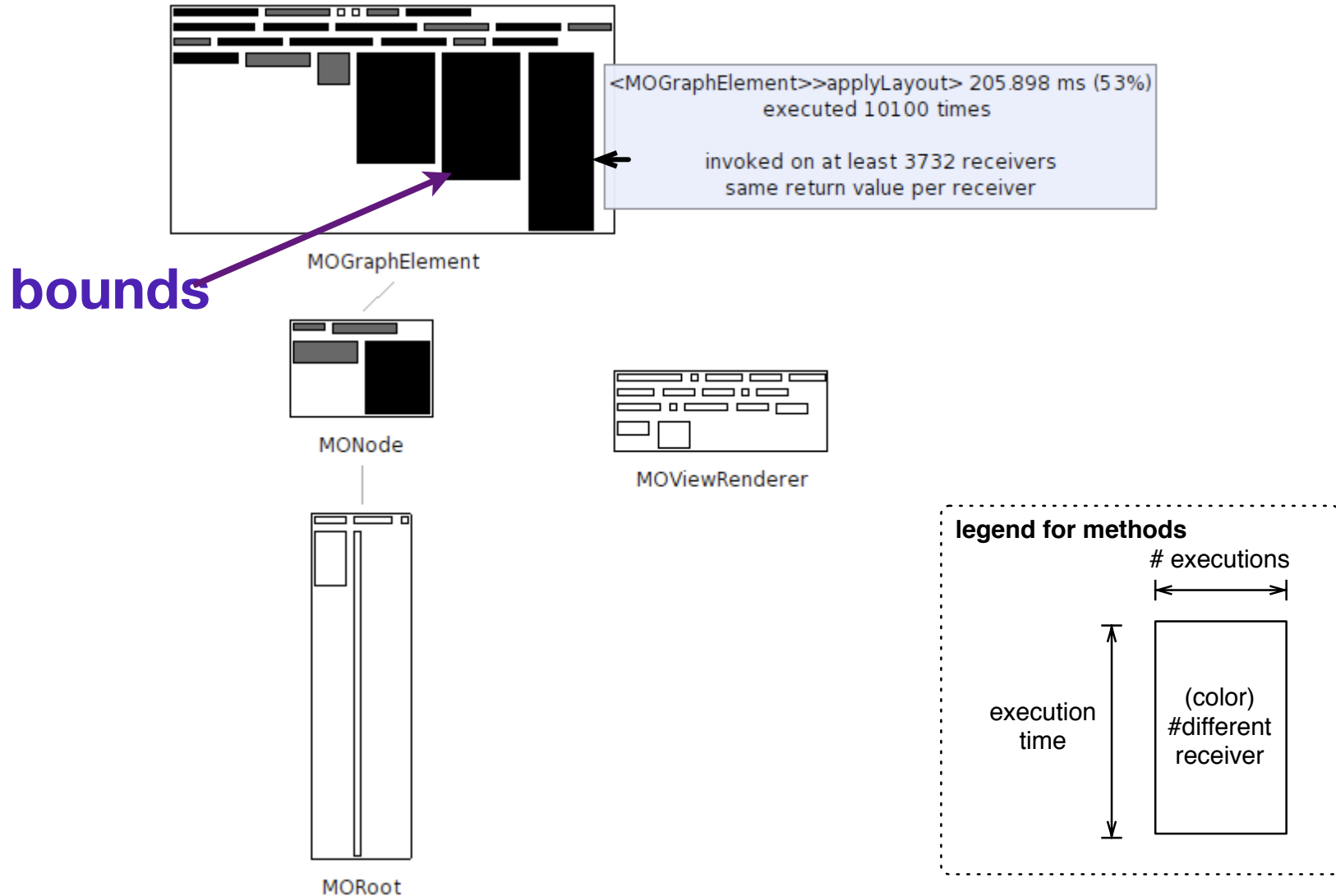
Behavioral profiling blueprint



DEMO

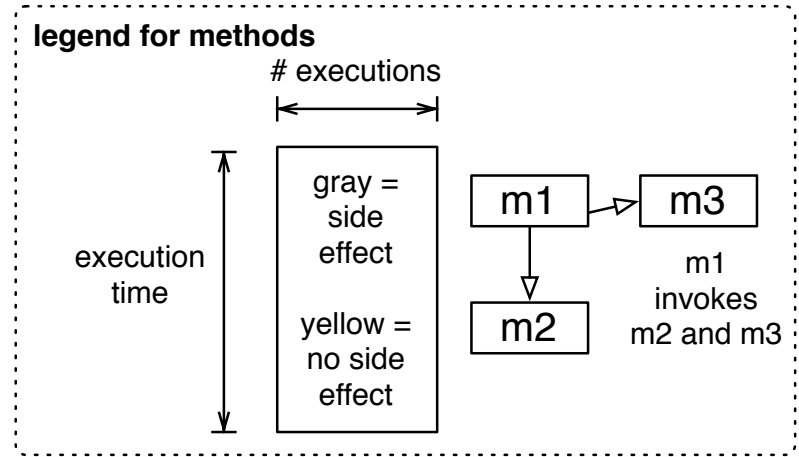
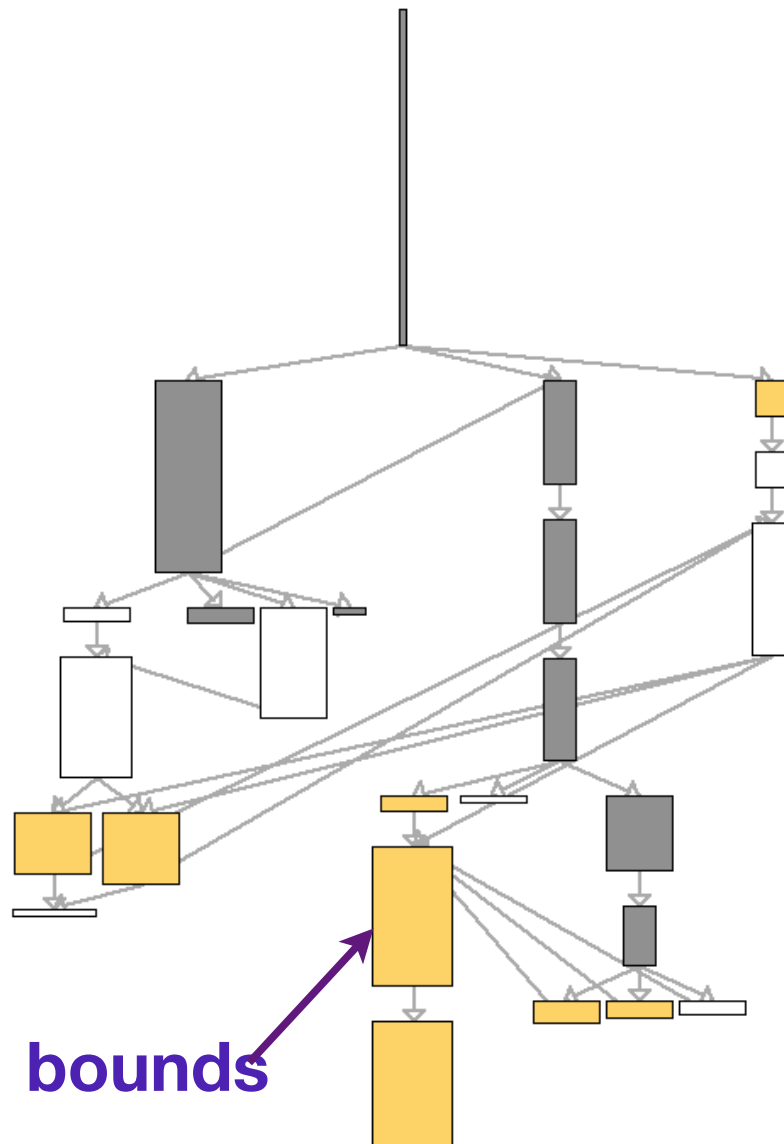


Structural profiling blueprint





Behavioral profiling blueprint





Code of the bounds method

```
MOGraphElement>>bounds
```

```
"Answer the bounds of the receiver."
```

```
| basicBounds |
```

```
self shapeBoundsAt: self shape ifPresent: [ :b | ^ b ].
```

```
basicBounds := shape computeBoundsFor: self.
```

```
self shapeBoundsAt: self shape put: basicBounds.
```

```
^ basicBounds
```



Memoizing

MOGraphElement>>bounds

"Answer the bounds of the receiver."

| basicBounds |

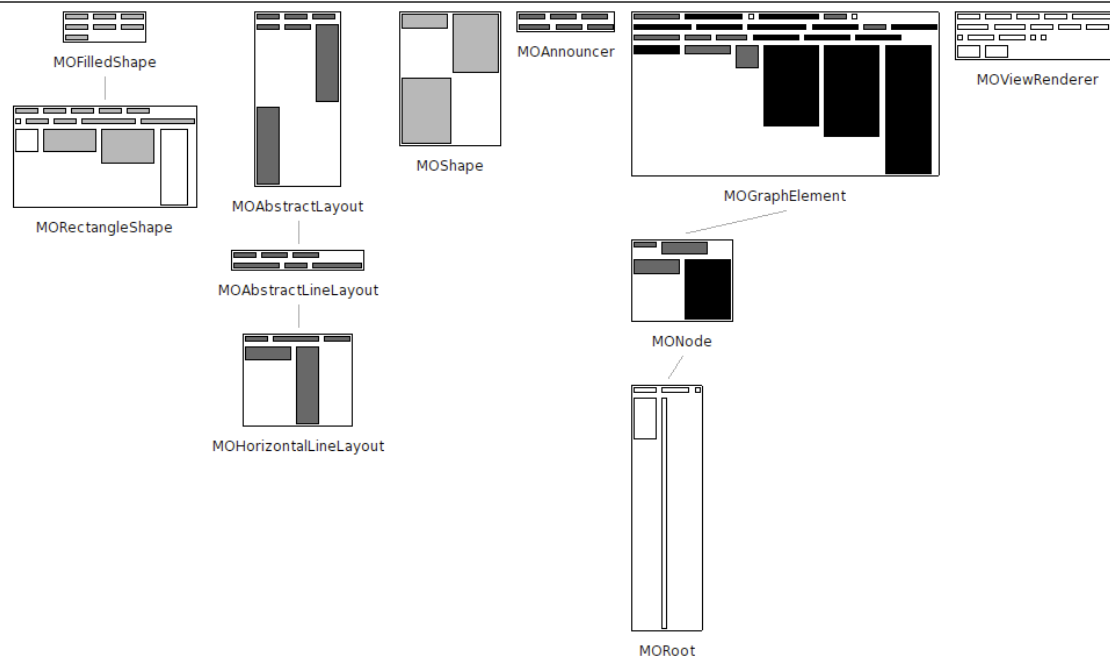
boundsCache ifNotNil: [^ boundsCache].

self shapeBoundsAt: self shape ifPresent: [:b | ^ b].

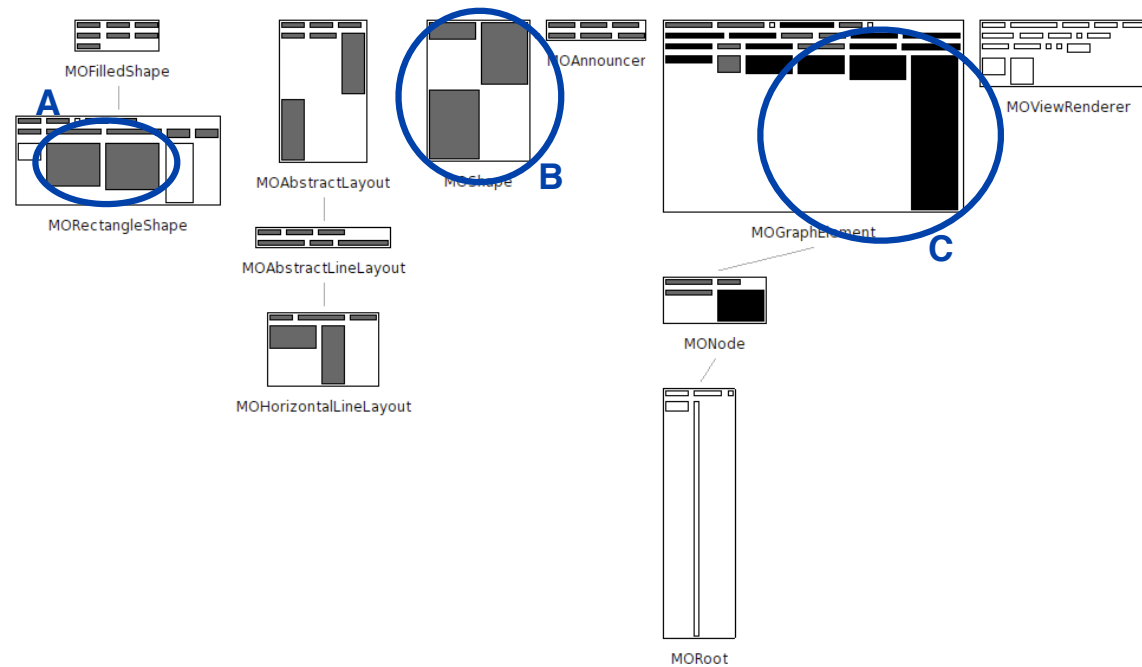
basicBounds := shape computeBoundsFor: self.

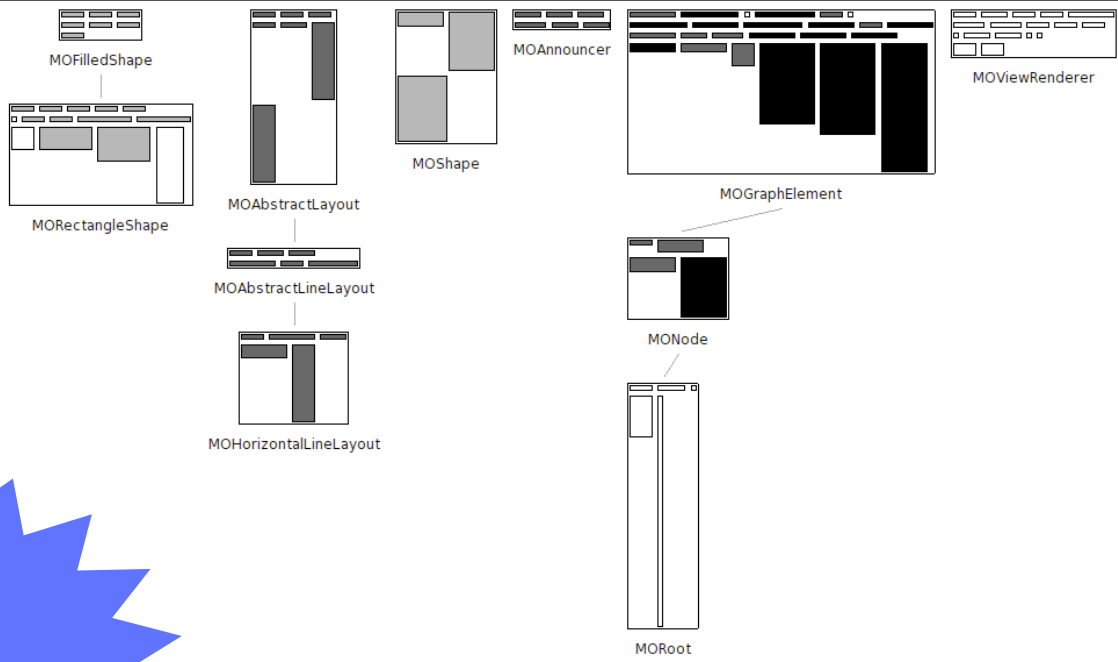
self shapeBoundsAt: self shape put: basicBounds.

^ boundsCache := basicBounds



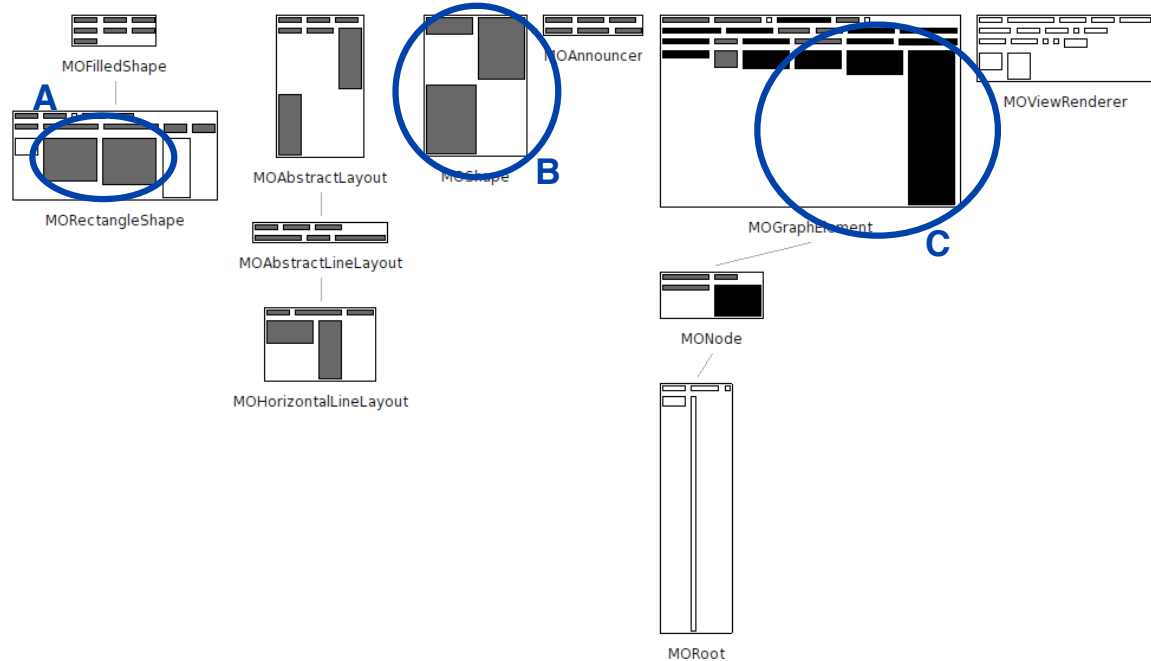
Upgrading
MOGraphElement >> bounds

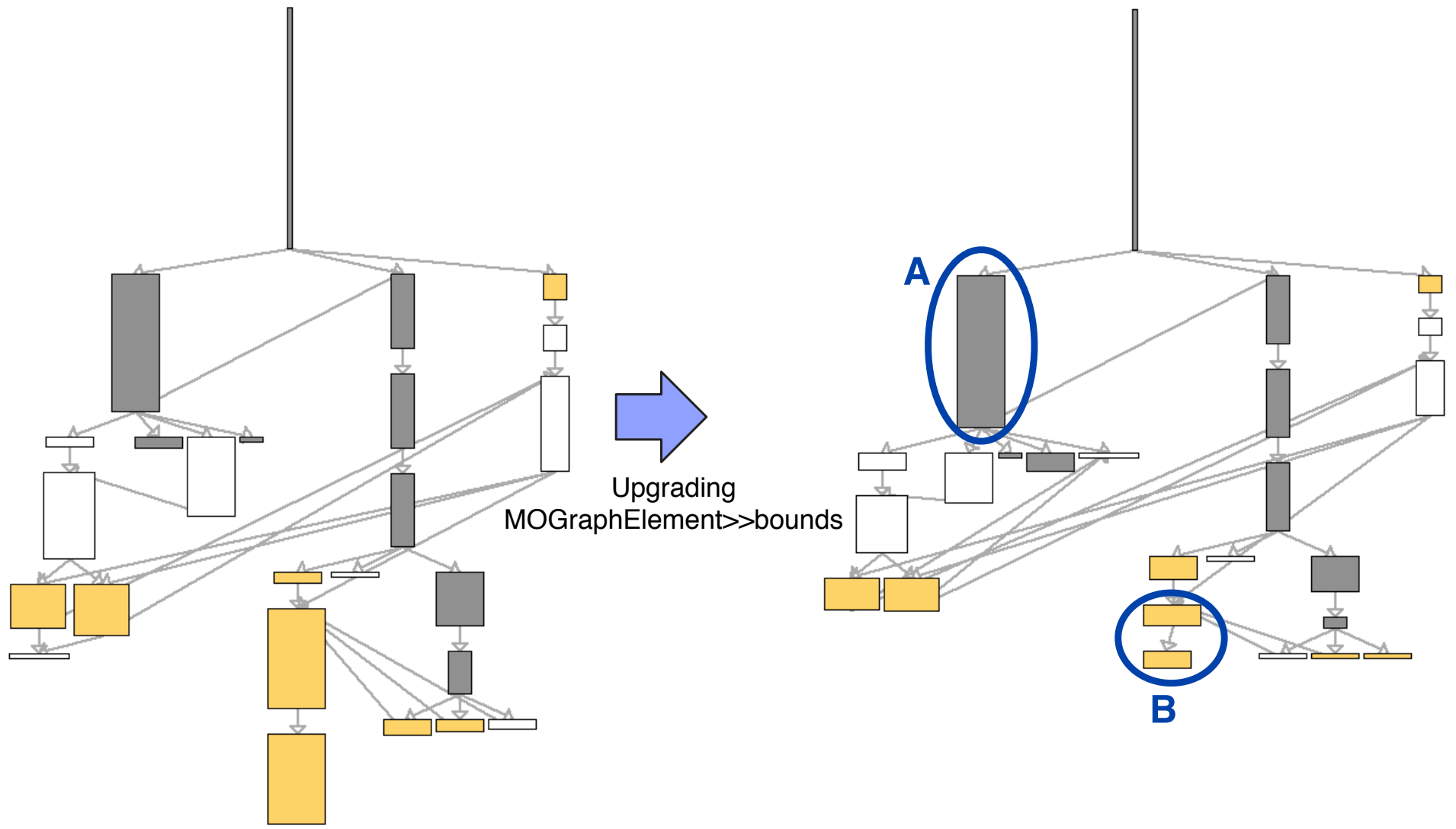


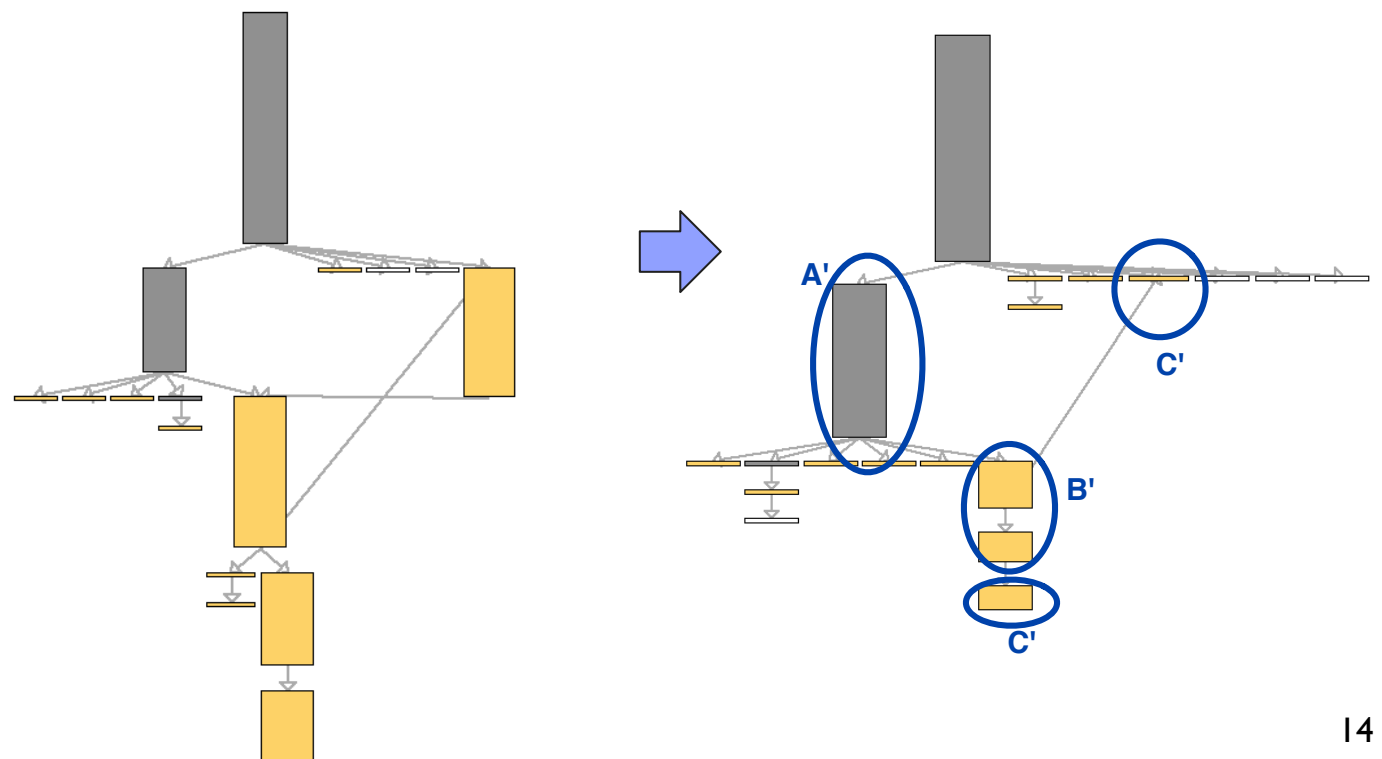
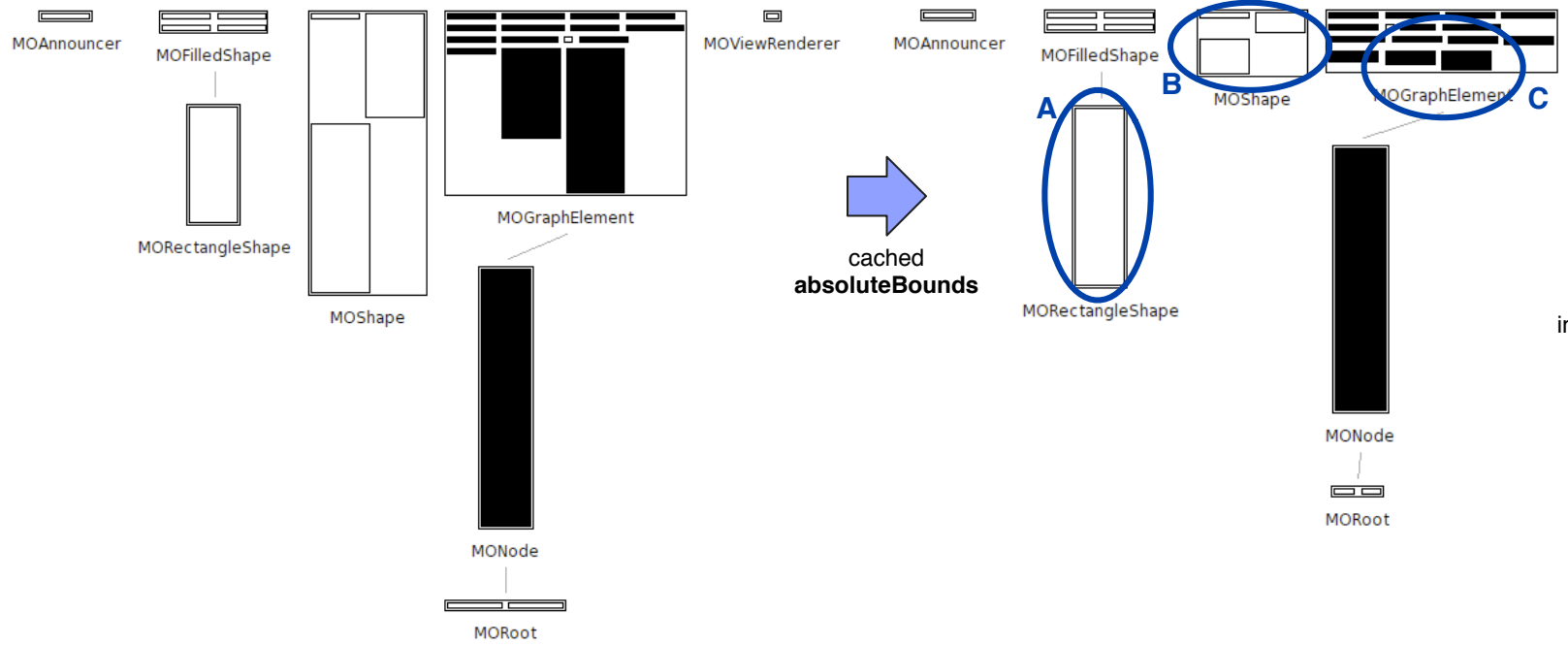


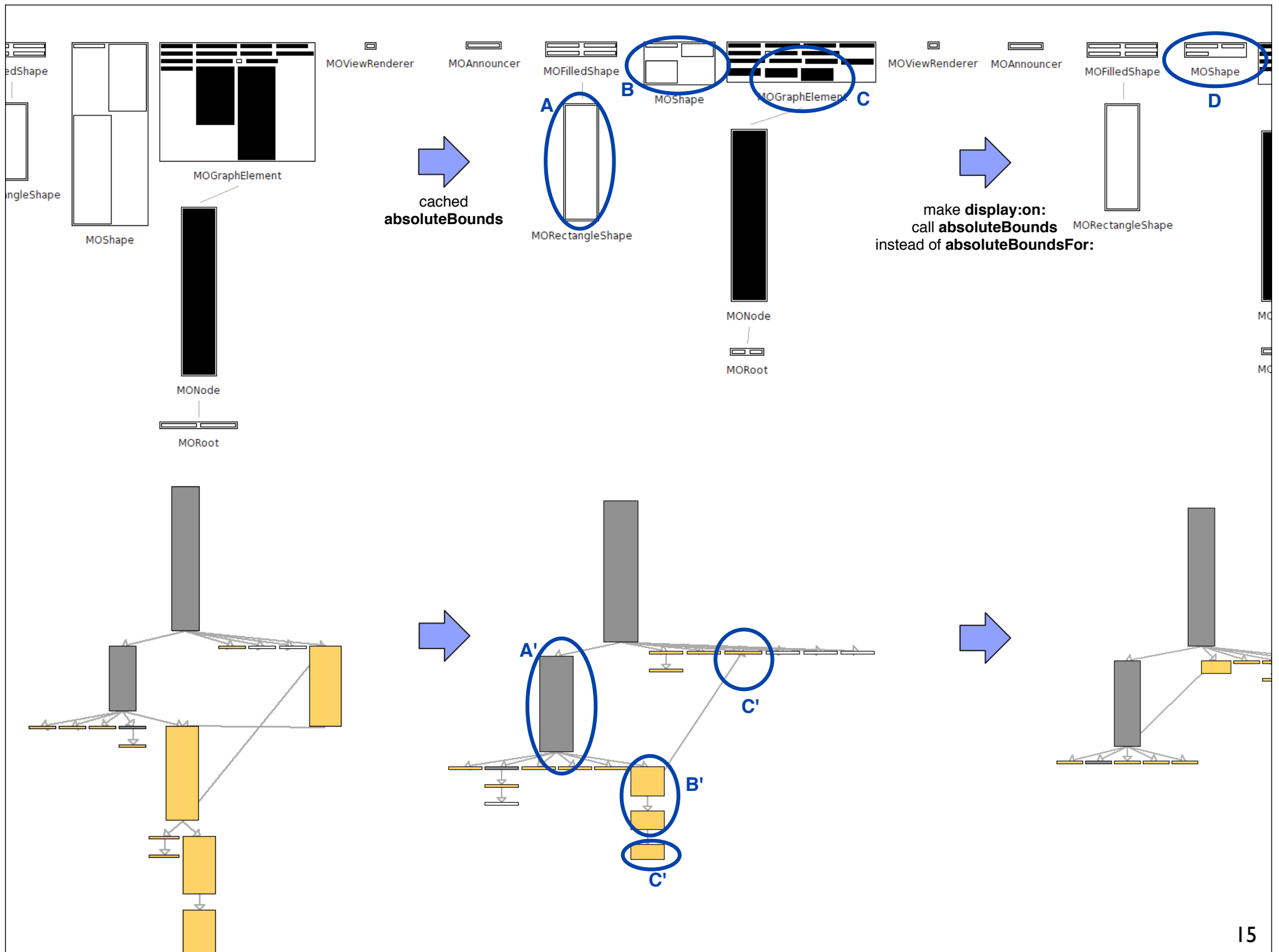
43% speedup

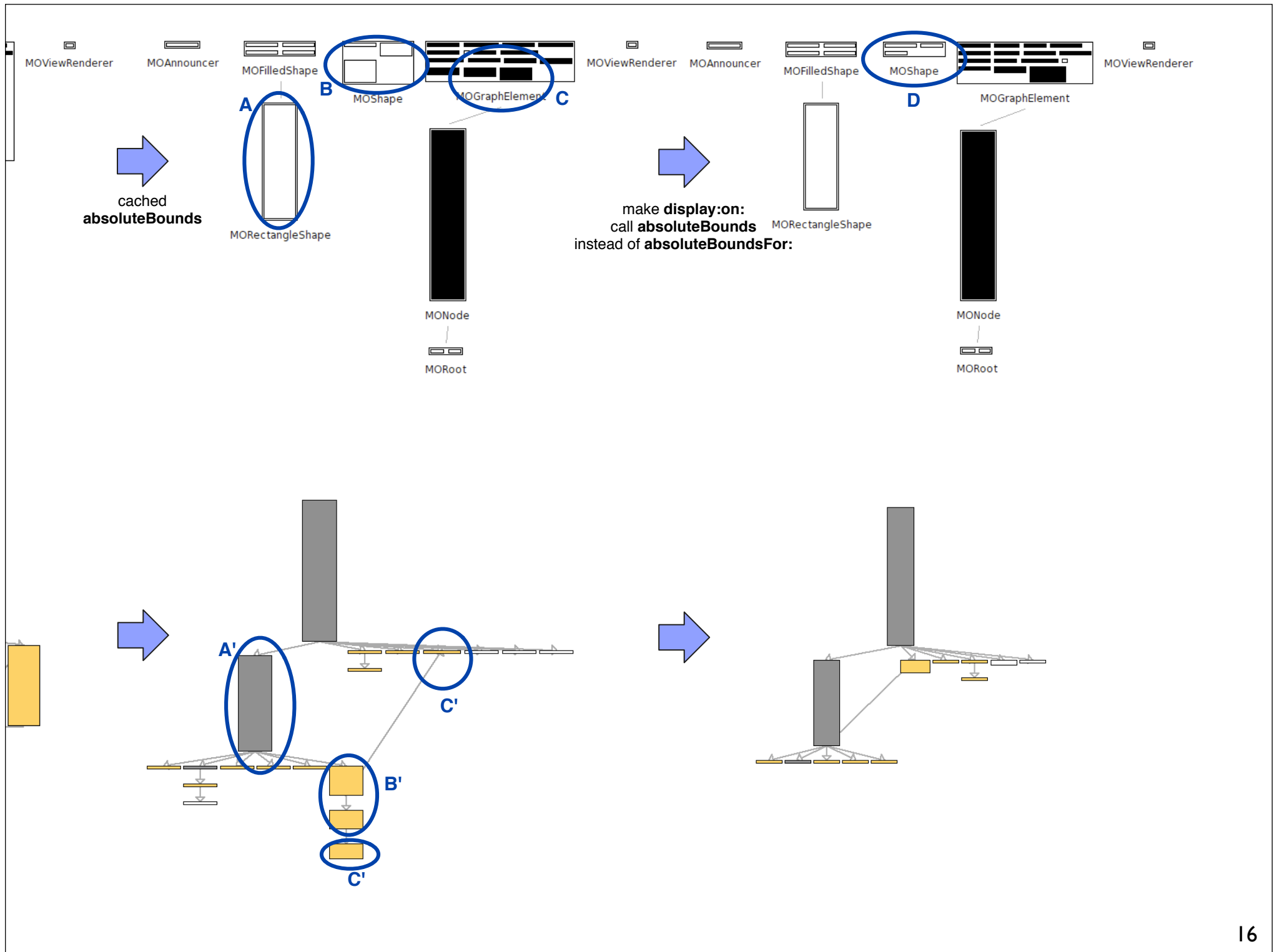
Upgrading MOGraphElement >> bounds













Test coverage with Hapao

Problem:

Traditional code coverage tools have a binary view of the world

Why the problem is important:

Which method should you test first in order to increase the coverage?

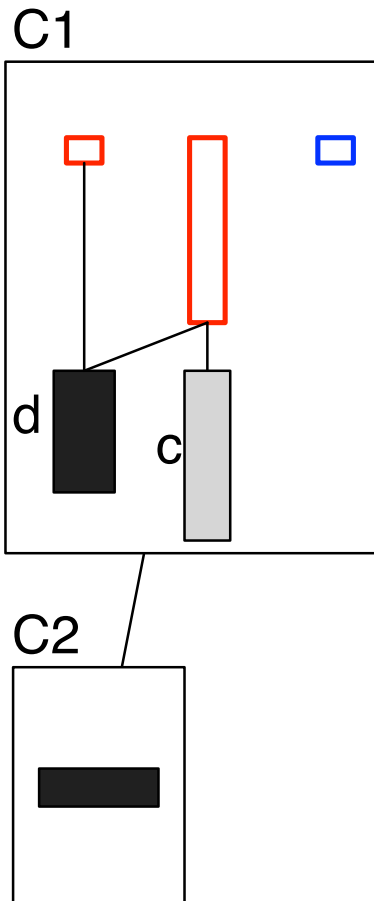
Is my code *well* covered or not?

Solution:

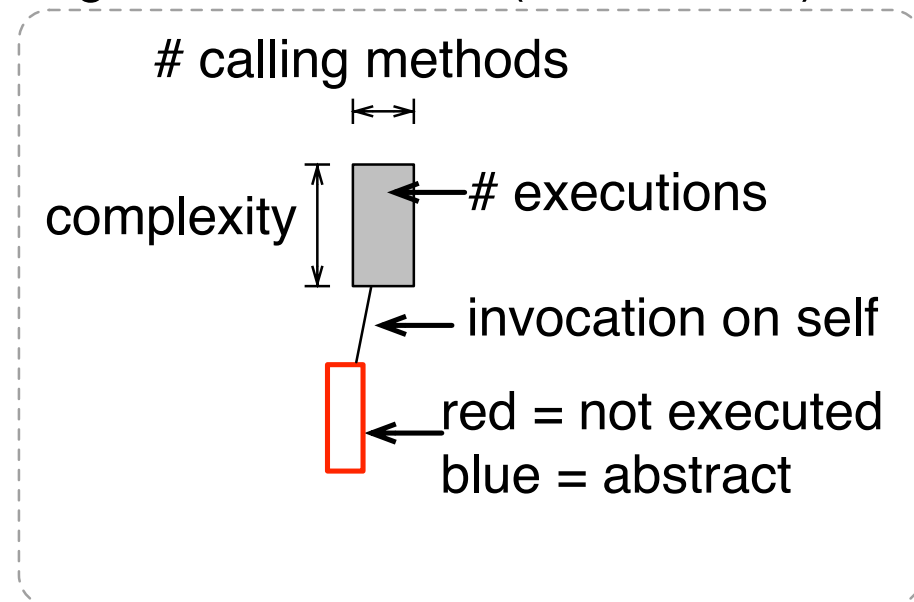
An intuitive visual representation of a qualitative assessment of the coverage



Test blueprint

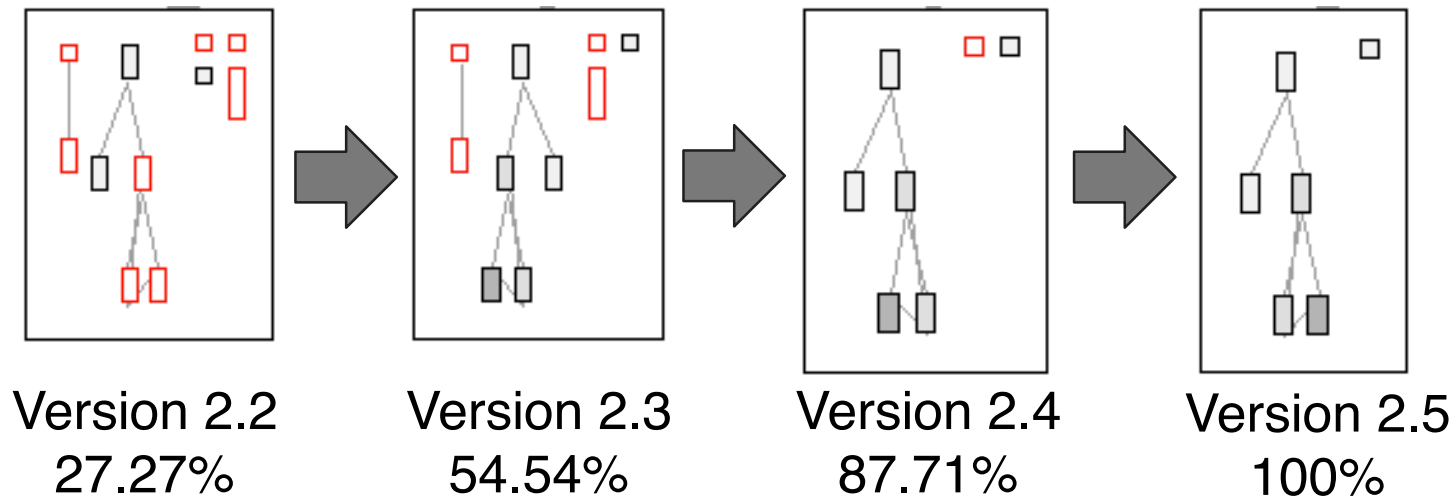


Legend for methods (inner boxes)



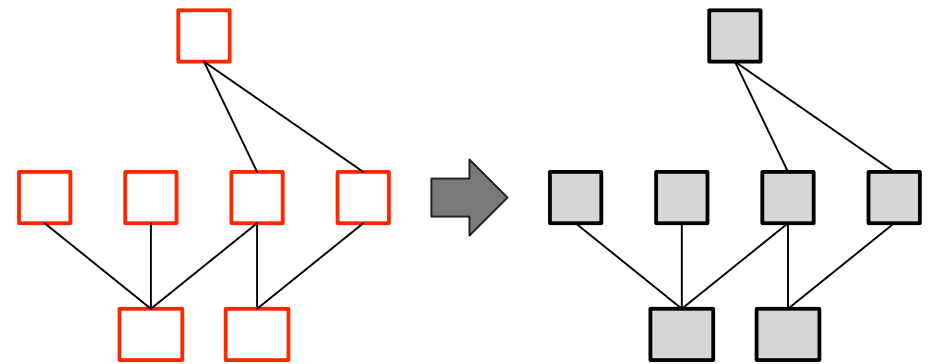
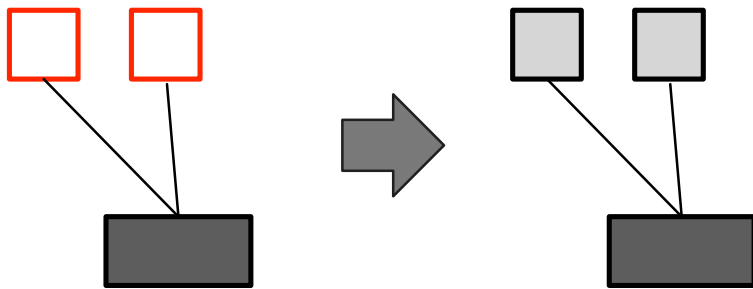
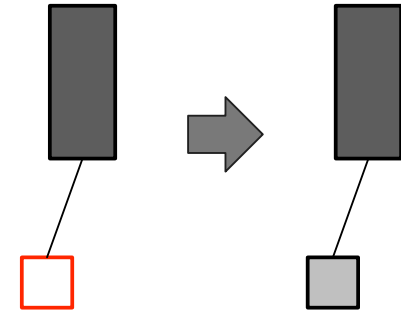
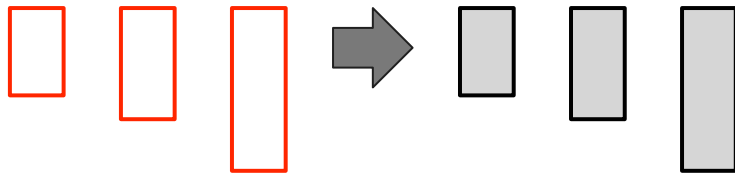


Successive improvement

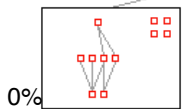
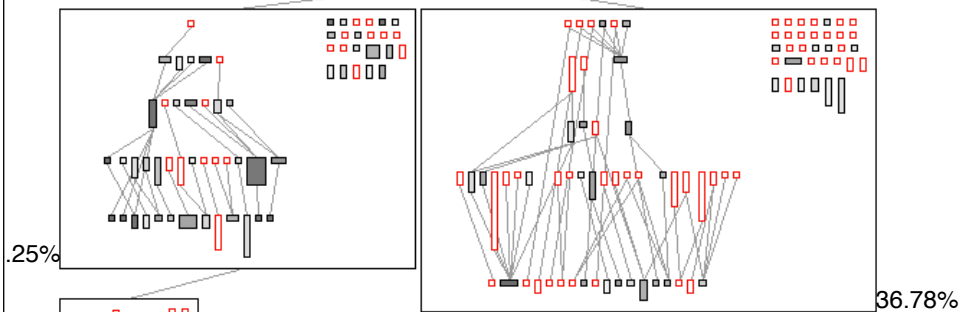
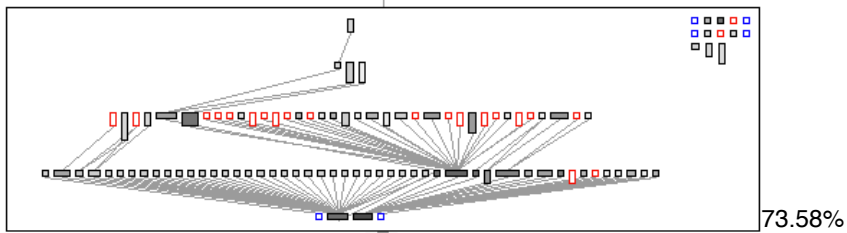
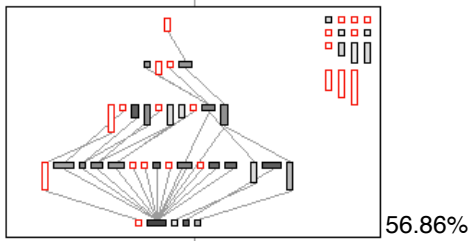
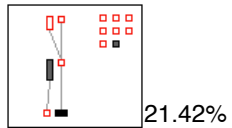




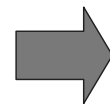
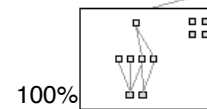
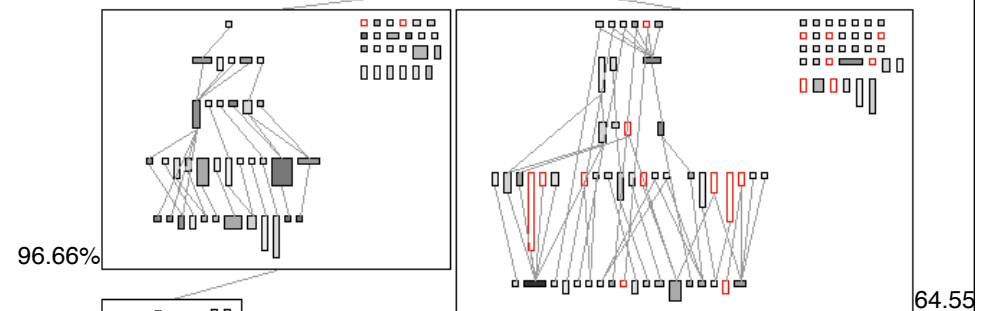
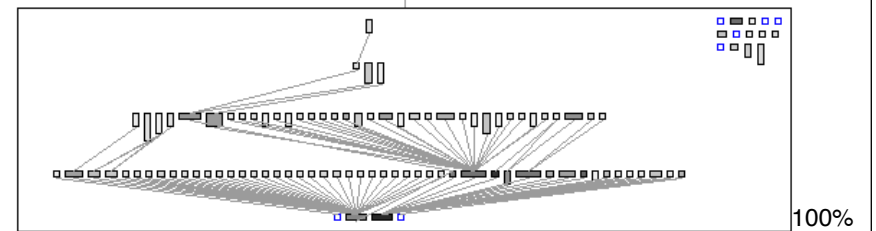
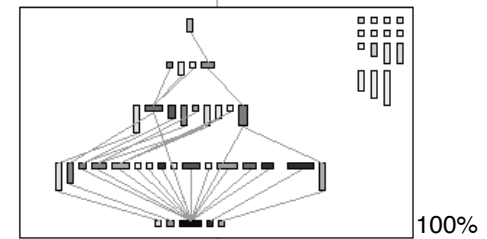
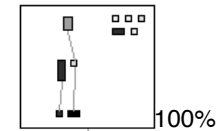
4 patterns



Moose-Test-Core.13
Moose-Core.313



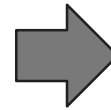
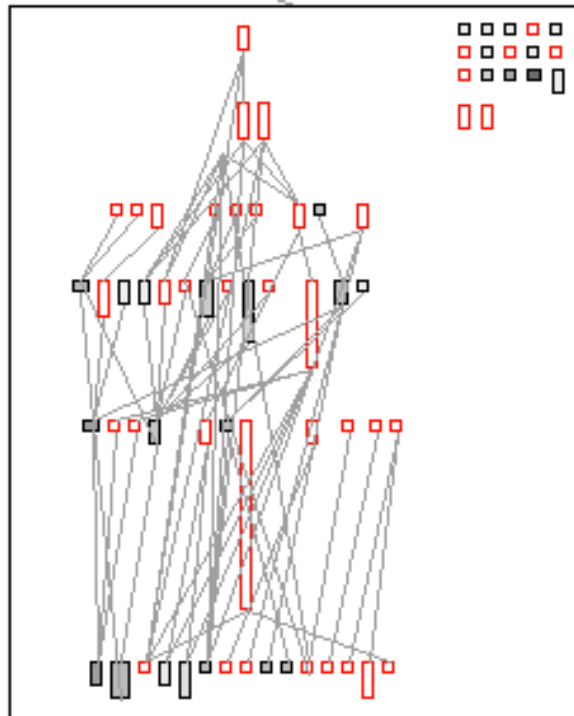
Moose-Test-Core.48
Moose-Core.326



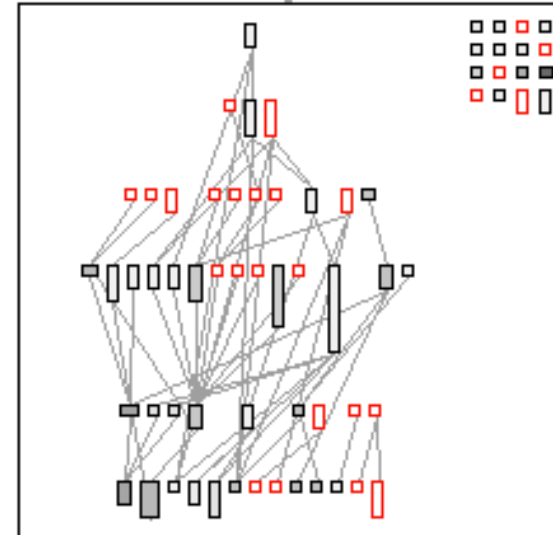


Reducing code complexity

Version 1.58.1
Coverage: 40.57%

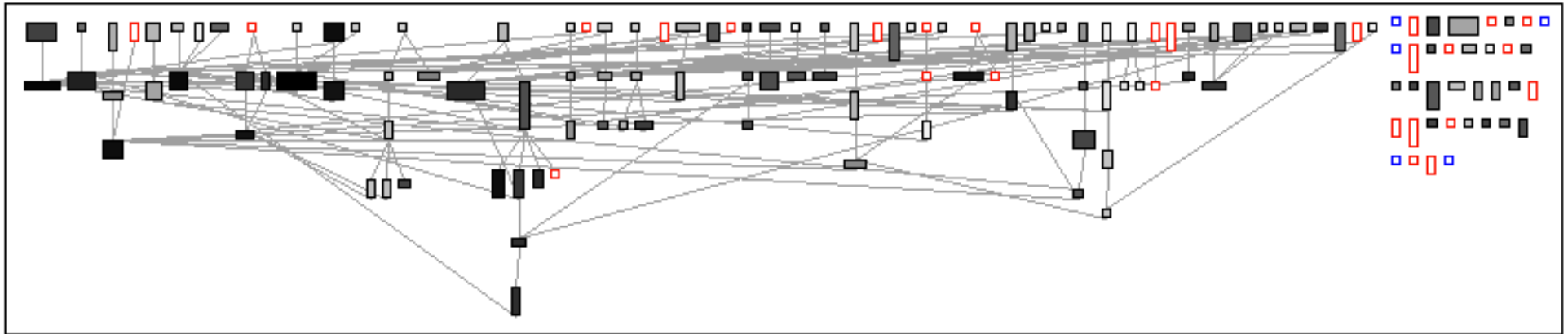


Version 1.58.9
Coverage: 60.60%

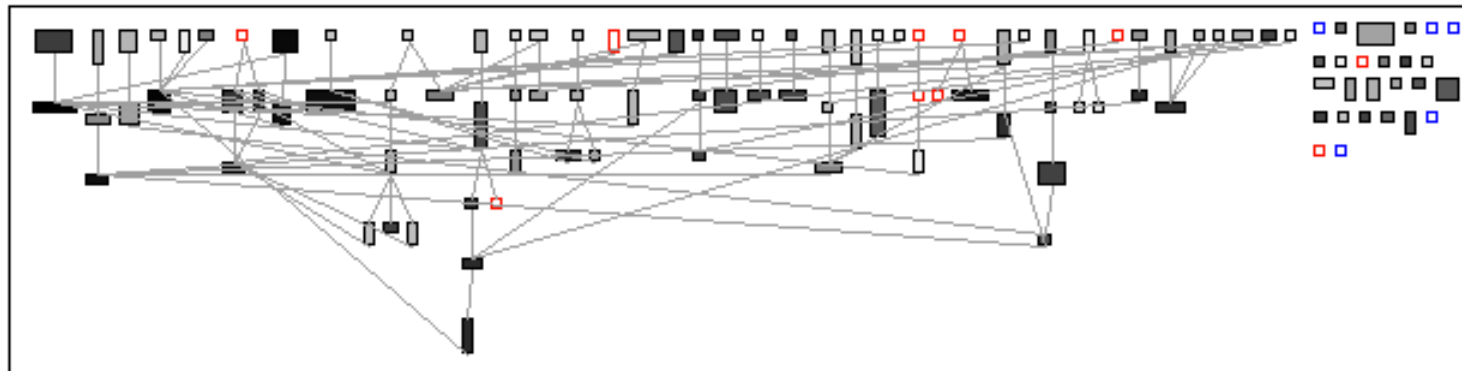




Reducing code complexity



Version 2.10



Version 2.17



Visualizing data with Roassal

Roassal is an *agile* and *interactive* visualization engine

view shape rectangle

width: #numberOfAttributes;

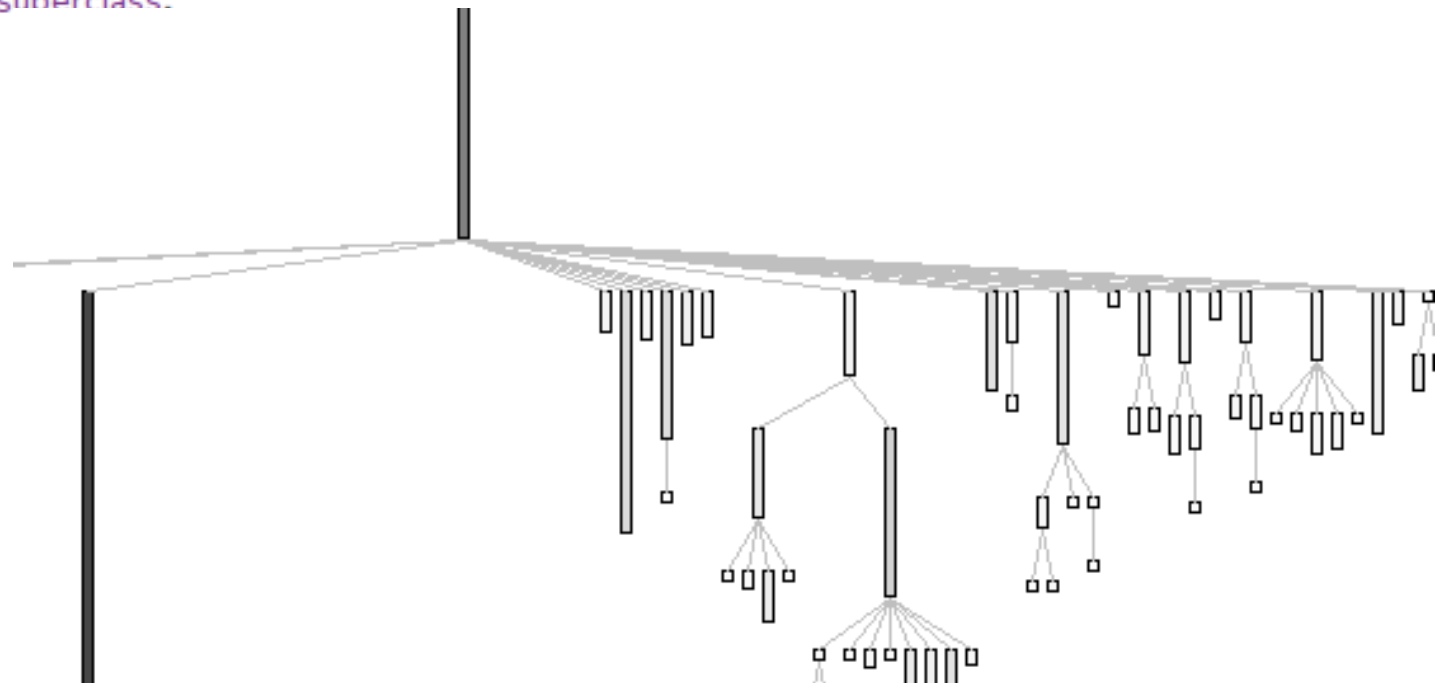
height: #numberOfMethods;

linearFillColor: #numberOfLinesOfCode within: Collection withAllSubclasses.

view nodes: Collection withAllSubclasses.

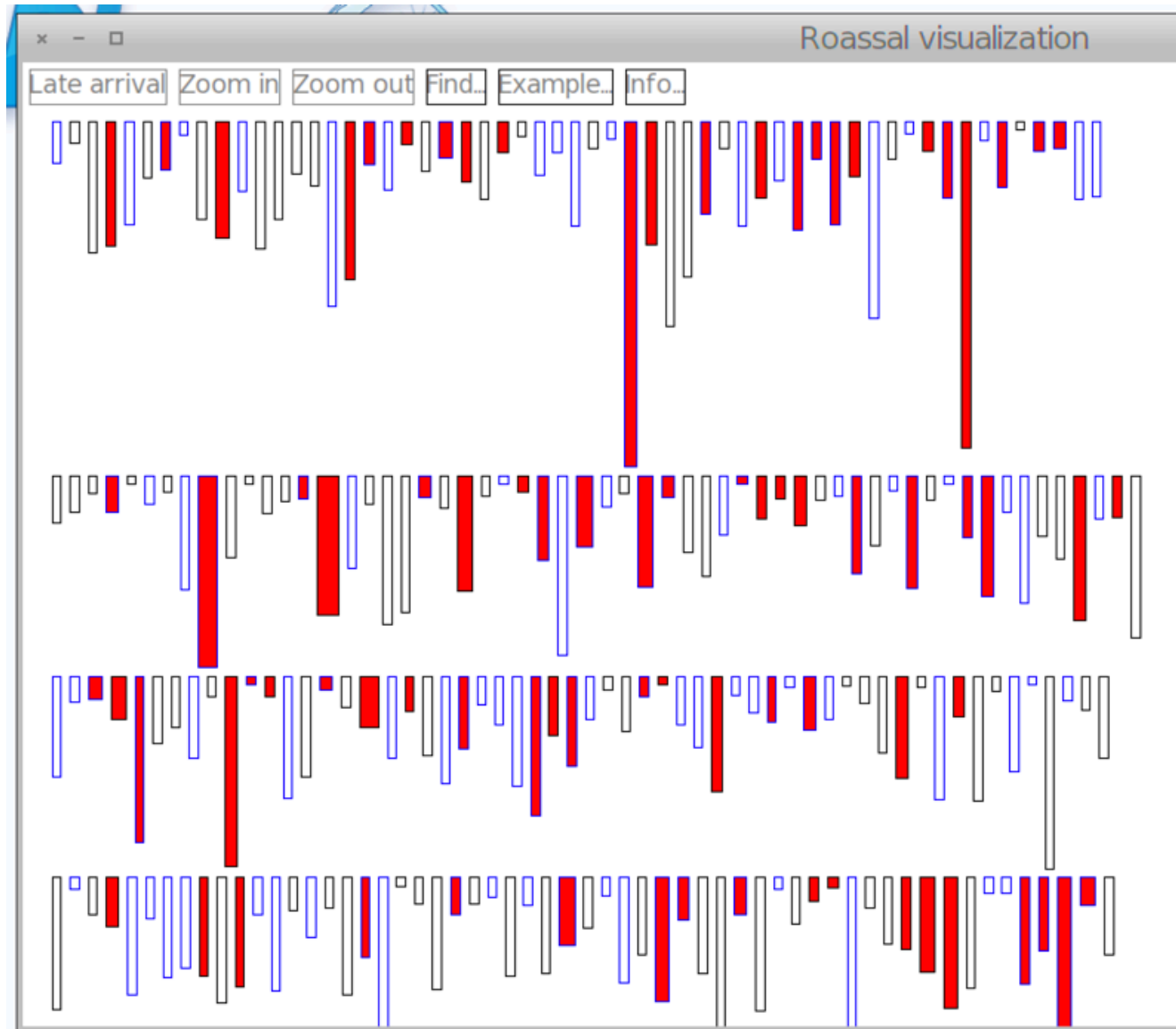
view edgesFrom: #superclass.

view treeLayout.





Visualizing Incidents at the Fire Department of NY



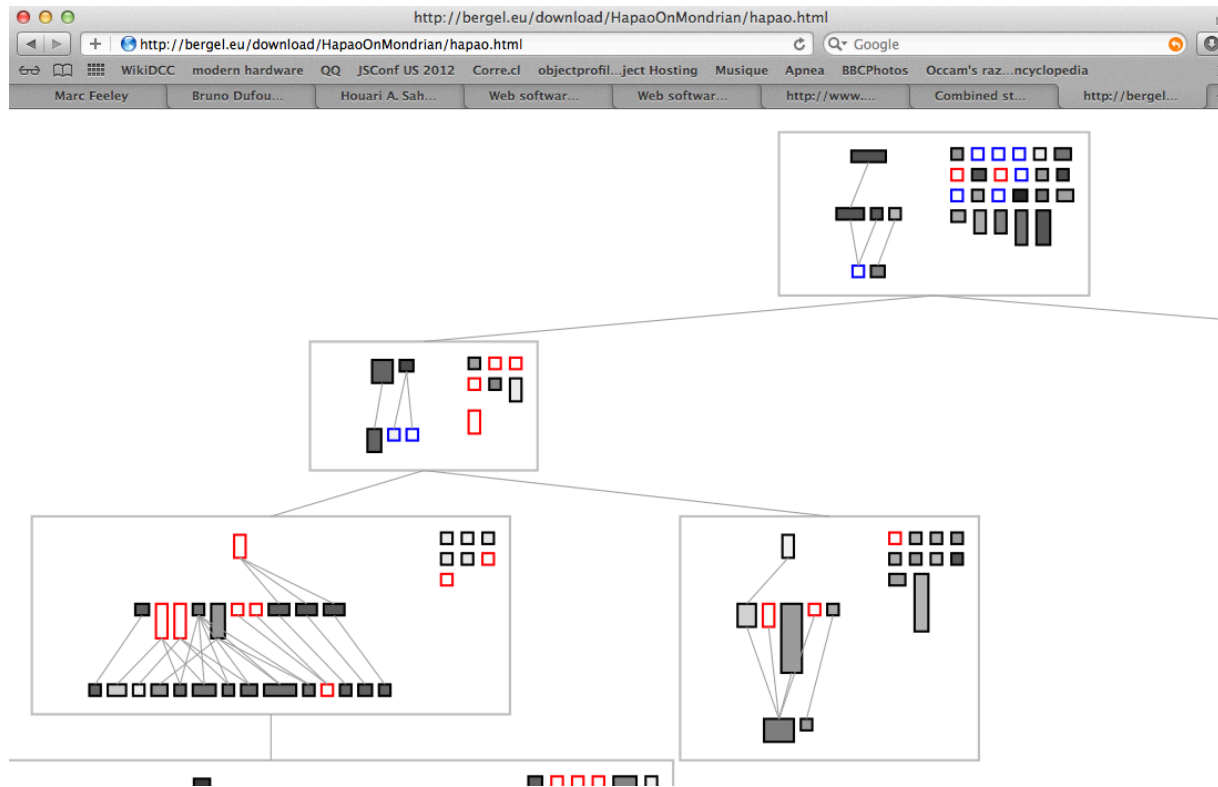
2 7TH ST : alias 82 7 ST NEW_ROCHELLE NEW_ROCH
2 7TH ST : alias 82 7 ST NEW_ROCHELLE NEW_ROCH
00 PELHAM RD : APT3R NEW_ROCHELLE
35 WEBSTER AVE : IFO NEW_ROCHELLE
35 WEBSTER AVE : IFO NEW_ROCHELLE
1 WINTHROP AVE : 4E NEW_ROCHELLE
WEBSTER AVE/LINCOLN AVE NEW_ROCHELLE
7 1ST ST : alias 37 1 ST NEW_ROCHELLE NEW_ROCH
7 1ST ST : alias 37 1 ST NEW_ROCHELLE NEW_ROCH
83 WASHINGTON AVE NEW_ROCHELLE
5 MAPLE AVE : @MAPLE TERRACE SENIOR APARTMEN
4 6TH ST : alias 14 6 ST NEW_ROCHELLE NEW_ROCH
47 UNION AVE NEW_ROCHELLE
0 KEOGH LN : @BAYBERRY NURSING HOME NEW RO



Supported Platforms

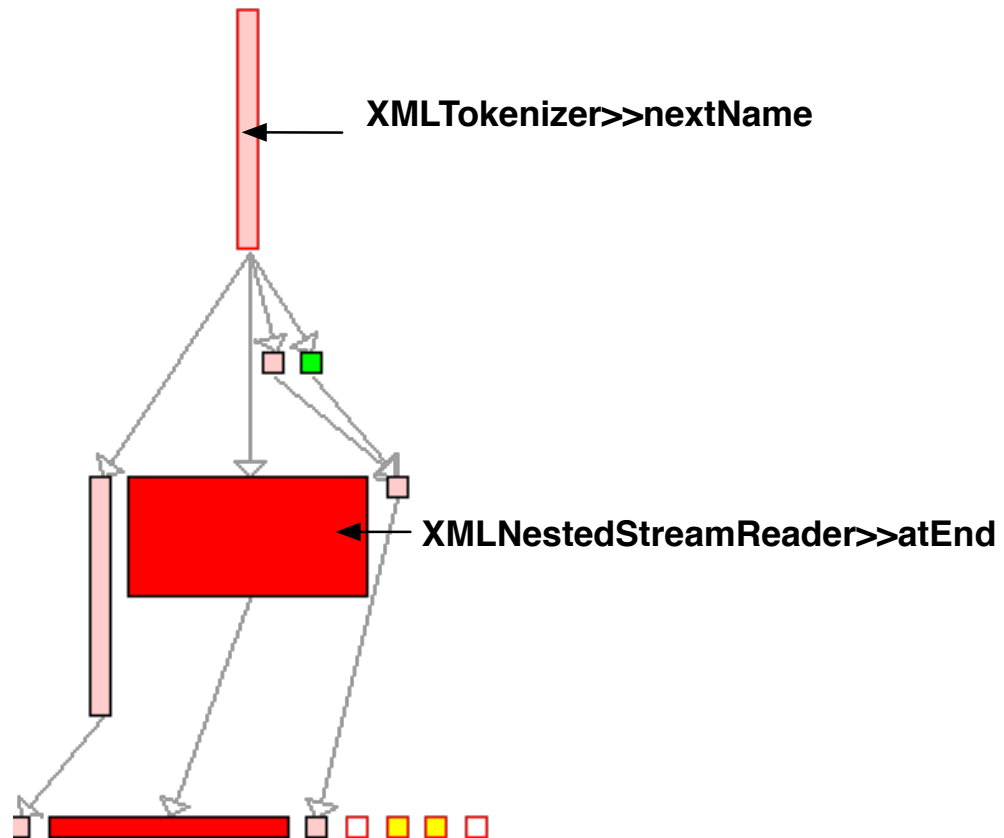


Amber
Smalltalk brought to the web





Multidimensional profiling



Differences between tests

TestSurgeon

GLMPresentationTest
GLMScriptingTest
GLMPaneTest
GLMCompositePresentationMorpl
GLMFinderTest
GLMBrowserTest
GLMFinderMorphicTest
GLMTreeMorphicTest

Red Test

- testActions
- testAllActions
- testAllSelectionActions
- testAnnouncement
- testContext
- testCopyAnnouncementsDoNot
- testCopyNotTheSameRegistry
- testCopyRootPrototype**
- testCopyTheTransformation
- testDefaultPane
- testDisplay
- testEquality
- testExplicitNotNilMultipleValues
- testHasTitle
- testHasTitleIcon
- testImplicitNotNilMultipleValues
- testMatching
- testSelectionActions
- testSetup
- testTitle
- testToRawSelection
- testToSend
- testToSendAs
- testToSendMultipleValue

Blue Test

- testCopyRootPrototype
- testEquality
- testCopyNotTheSameRegistry
- testCopyAnnouncementsDoNot
- testWith
- testSelectionActions
- testHasTitle
- testActions
- testHasTitleIcon
- testAllSelectionActions
- testAllActions
- testCopyTheTransformation
- testSetup
- testDefaultPane
- testUnregisterAnnouncements
- testToSendMultipleValue**
- testExplicitNotNilMultipleValues
- testImplicitNotNilMultipleValues
- testTitle
- testToSend
- testValidate
- testToRawSelection
- testDisplay
- testContext

```
testCopyRootPrototype
| presentation newPresentation yetAnotherPresentation |
presentation := GLMPresentation new.
newPresentation := presentation copy.
self assert: newPresentation rootPrototype equals: presentation
.
self assert: newPresentation parentPrototype equals: presentati
on.
yetAnotherPresentation := newPresentation copy.
self assert: yetAnotherPresentation rootPrototype equals: prese
ntation.
self assert: yetAnotherPresentation parentPrototype equals: ne
```

```
testToSendMultipleValue
| presentation |
presentation := GLMPresentation new.
presentation selection: (GLMMultiValue withAll: #(#foo #bar)).
self assert: presentation selection equals: #foo.
presentation selectionTransformation: [ :first :second | (first , seco
nd) asSymbol ].
```

Conclusion

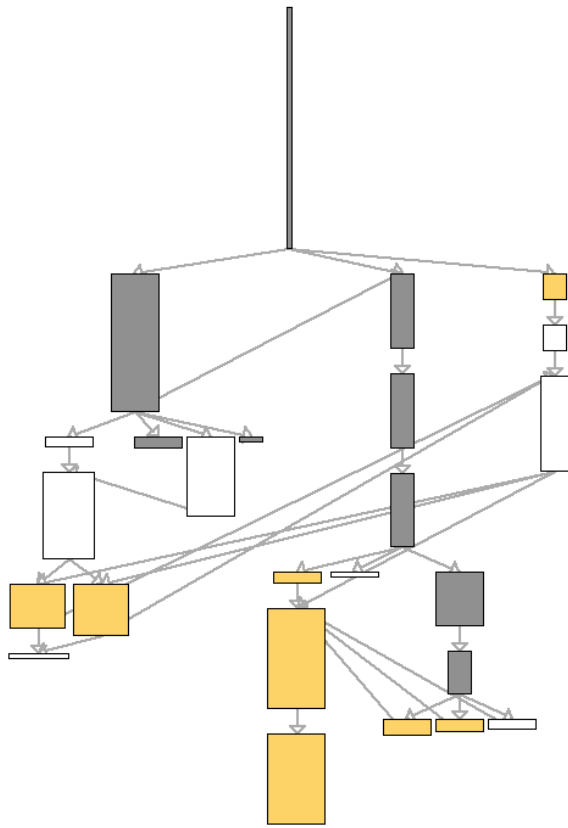
Little innovation in the tools we commonly use

Profilers, debuggers, testing tools have not significantly evolved

Fantastic opportunities for improvement

Kai, Hapao and Roassal are just a beginning

Feel free to provide feedback on our tool



Spy @ Cincom Store
 Spy @ SqueakSource
 Roassal @ ...

ObjectProfile.com

Thanks to
 ESUG
 Chris T
 Cincom
 All of you guys!

