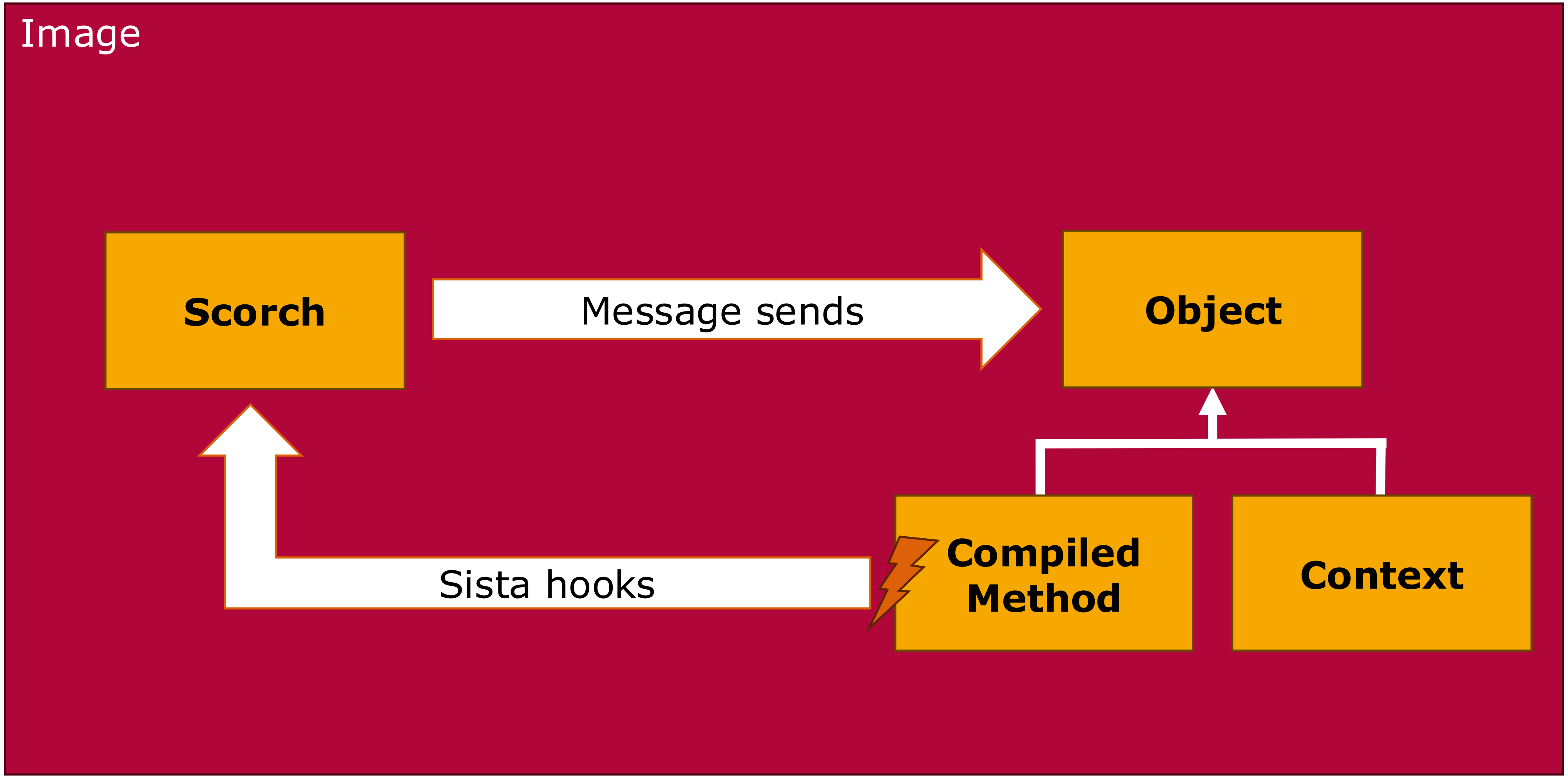
OSVM: Motivation

- **Sista VM** (**S**peculative **I**nline **S**mall**T**alk **A**rchitecture) [Bera, 2017a]
  - Optimizes frequent methods with **inlining** and **unsafe bytecodes**
  - Promises **speed-ups of 3x-5x**
  - **Adaptive optimizer** (Scorch) is implemented **in the image** to support live development and interactive debugging [Bera, 2017b]
- **Challenge: Bootstrapping** Sista/Scorch
  - **Frequent VM faults/crashes** impede initial development
- **Opportunity:** Use the **OSVM simulator** to debug the VM in another image [Miranda, 2018]
- **Questions:**
  - How can we run and develop Scorch in the **simulator?**
  - How can we **improve surrounding tooling** for exploratory programming in the simulator?

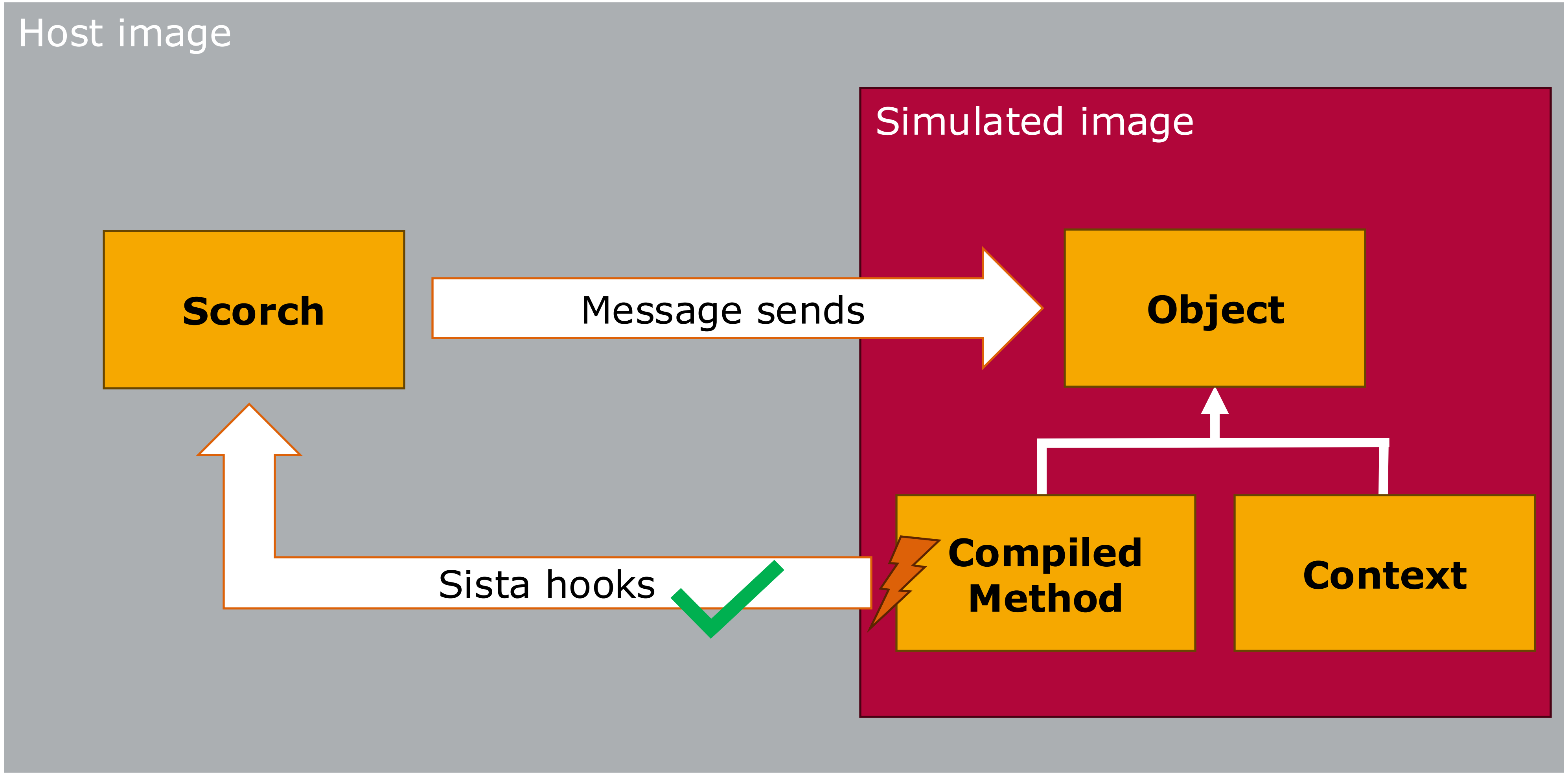
OSVM: Bootstrapping Scorch

Current Scorch/Image Interaction: Scorch Runs in Production Image



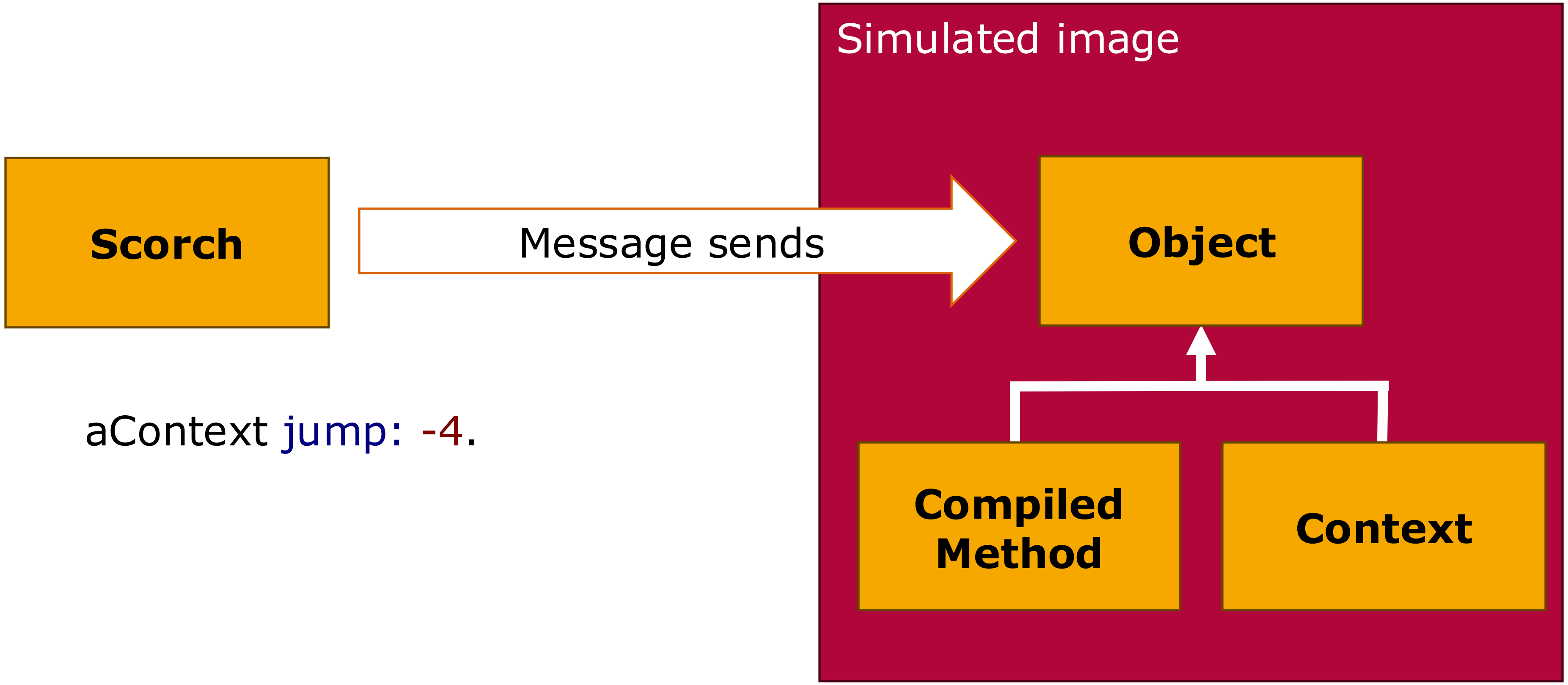
OSVM: Bootstrapping Scorch

Intended Scorch/Image Interaction: Develop Scorch Next to Simulated Image



OSVM: Bootstrapping Scorch

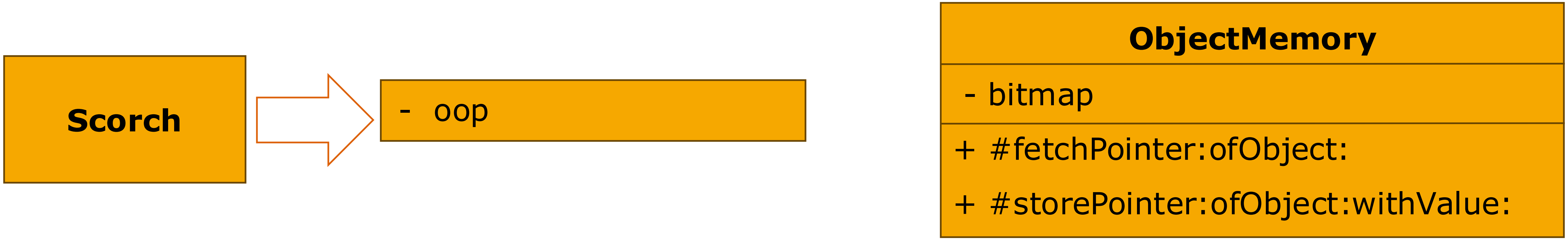
Intended Scorch/Image Interaction: Develop Scorch Next to Simulated Image



```
aContext jump: -4.
```

OSVM: Bootstrapping Scorch

Intended Scorch/Image Interaction: Develop Scorch Next to Simulated Image



Scorch

- oop

ObjectMemory

- bitmap

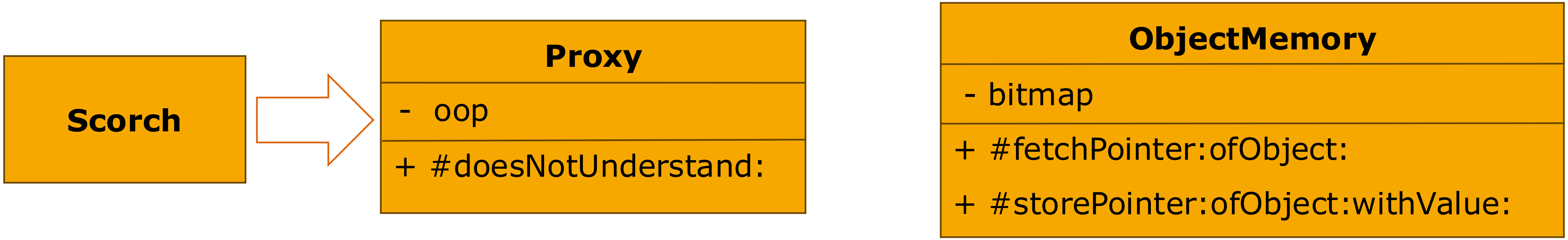
+ #fetchPointer:ofObject:

+ #storePointer:ofObject:withValue:

```
aContext jump: -4.
```

OSVM: Bootstrapping Scorch

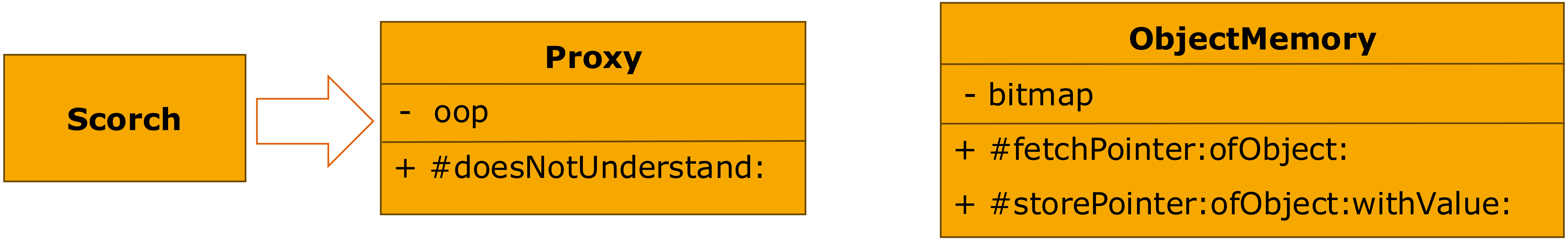
Implementing a Transparent Proxy Framework

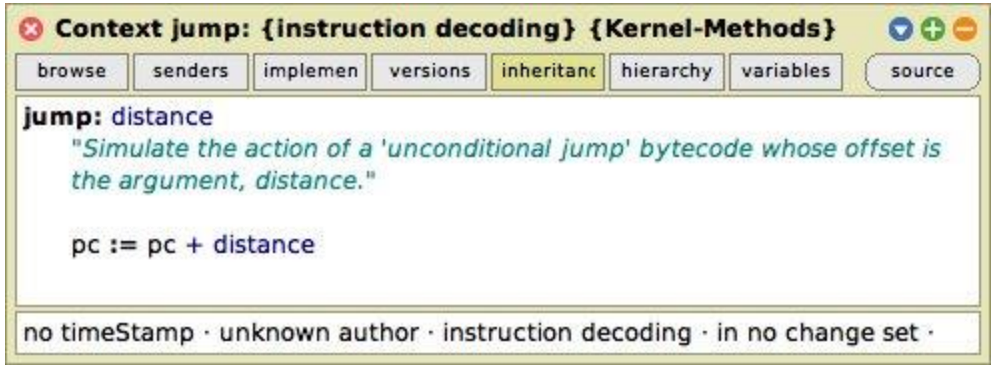

Scorch
Proxy
- oop
+ #doesNotUnderstand:
ObjectMemory
- bitmap
+ #fetchPointer:ofObject:
+ #storePointer:ofObject:withValue:

```
aContextProxy jump: -4.
```

OSVM: Bootstrapping Scorch

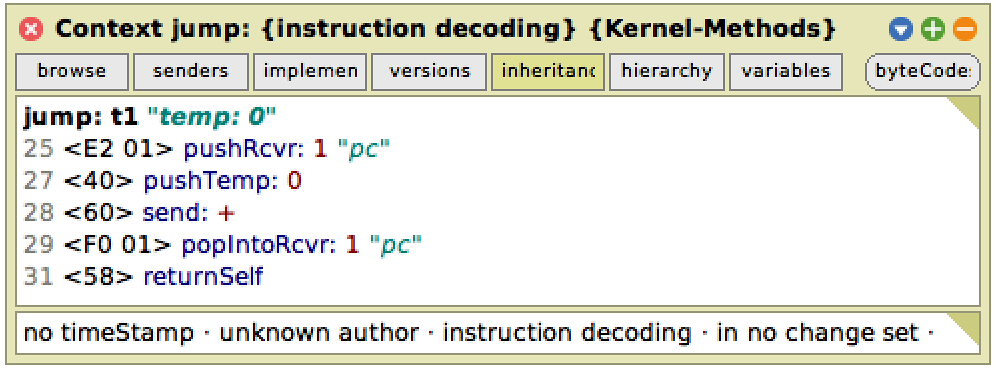
Implementing a Transparent Proxy Framework


Scorch
Proxy
- oop
+ #doesNotUnderstand:
ObjectMemory
- bitmap
+ #fetchPointer:ofObject:
+ #storePointer:ofObject:withValue:



```
jump: distance
	"Simulate the action of a 'unconditional jump' bytecode whose offset is
	the argument, distance."

	pc := pc + distance
```

no timeStamp · unknown author · instruction decoding · in no change set ·

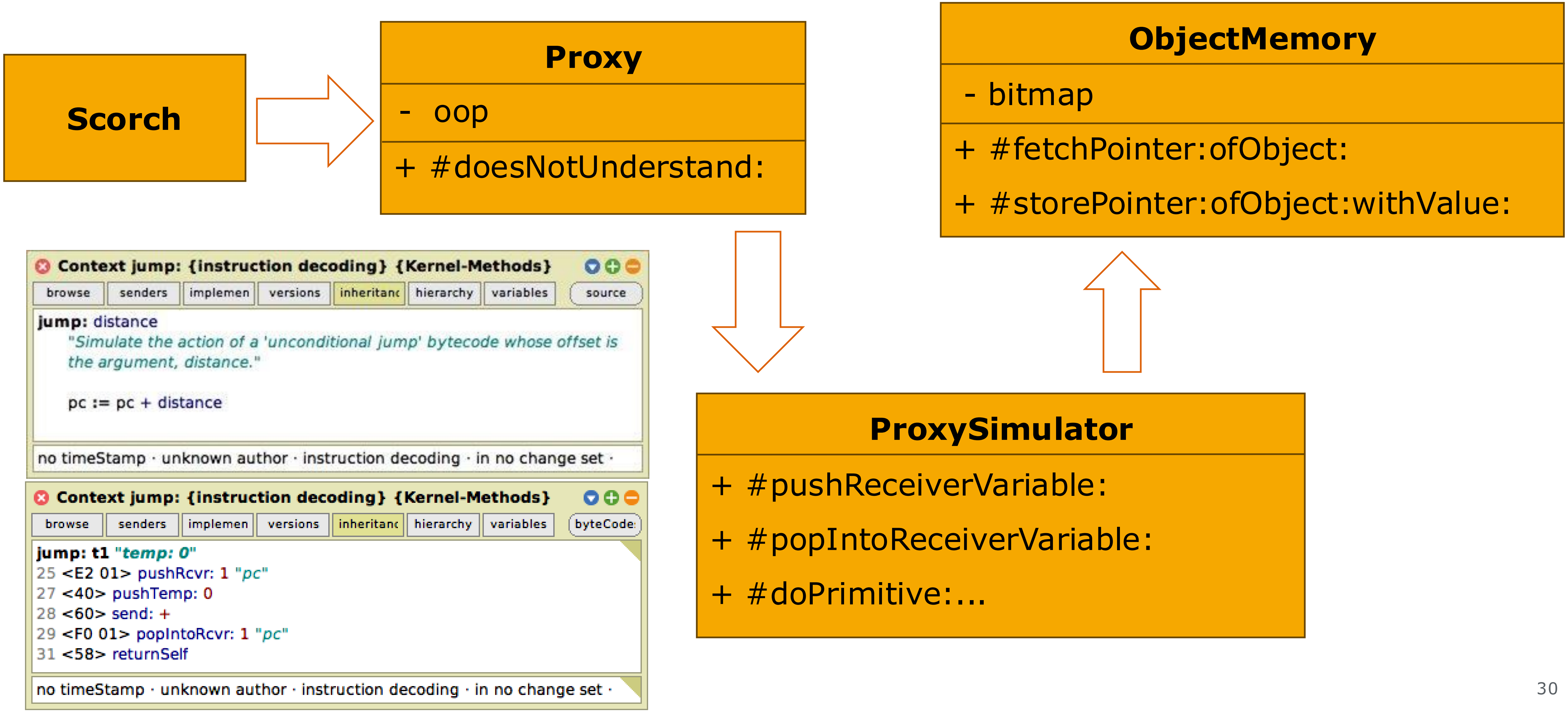


```
jump: t1 "temp: 0"
25 <E2 01> pushRcvr: 1 "pc"
27 <40> pushTemp: 0
28 <60> send: +
29 <F0 01> popIntoRcvr: 1 "pc"
31 <58> returnSelf
```

no timeStamp · unknown author · instruction decoding · in no change set ·

OSVM: Bootstrapping Scorch

Implementing a Transparent Proxy Framework



Context jump: {instruction decoding} {Kernel-Methods}

```
jump: distance
	"Simulate the action of a 'unconditional jump' bytecode whose offset is
	the argument, distance."

	pc := pc + distance
```

no timeStamp · unknown author · instruction decoding · in no change set ·

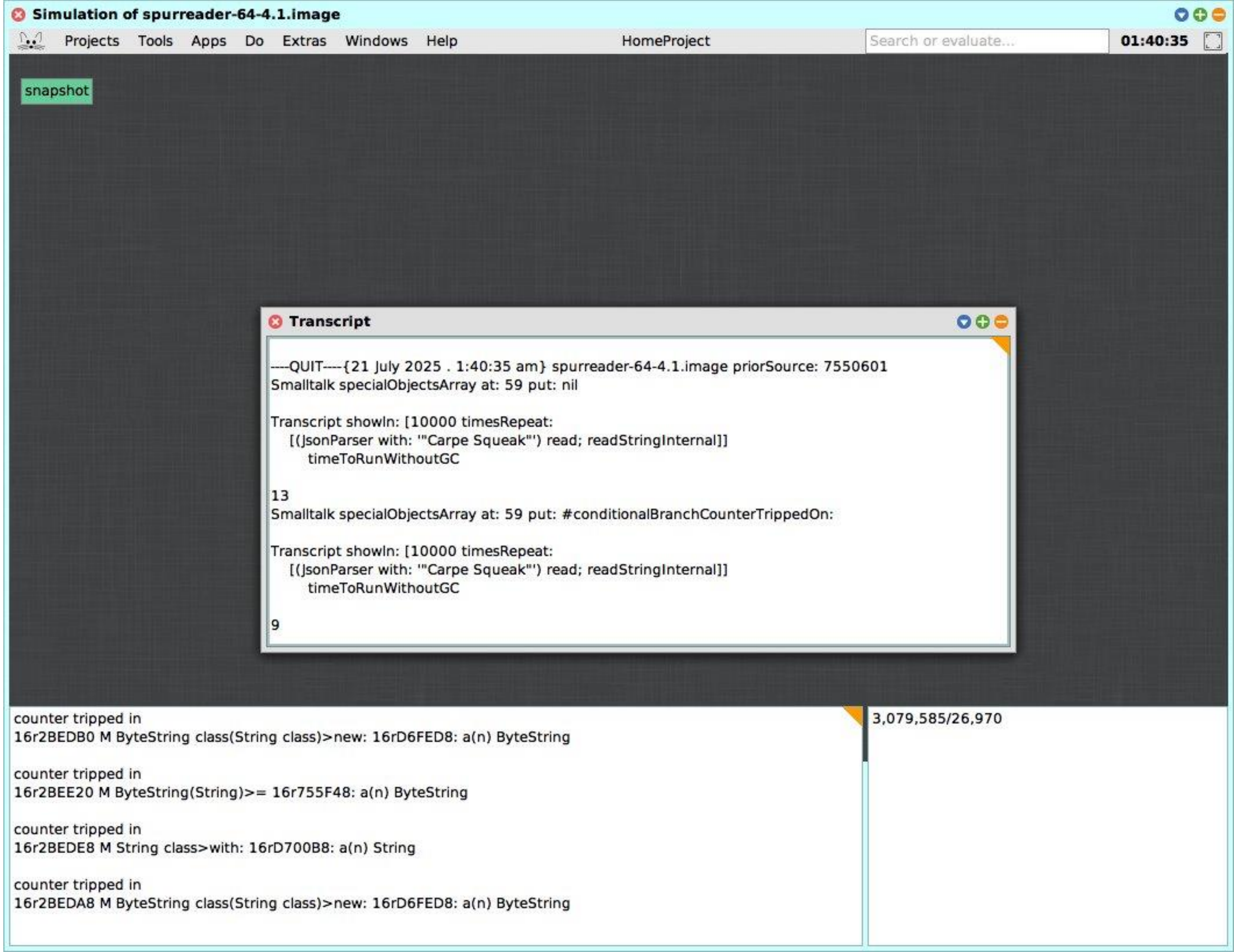
Context jump: {instruction decoding} {Kernel-Methods}

```
jump: t1 "temp: 0"
25 <E2 01> pushRcvr: 1 "pc"
27 <40> pushTemp: 0
28 <60> send: +
29 <F0 01> popIntoRcvr: 1 "pc"
31 <58> returnSelf
```

no timeStamp · unknown author · instruction decoding · in no change set ·

OSVM: Bootstrapping Scorch

Running Benchmarks



```
----QUIT----{21 July 2025 . 1:40:35 am} spurreader-64-4.1.image priorSource: 7550601
Smalltalk specialObjectsArray at: 59 put: nil

Transcript showln: [10000 timesRepeat:
    [(JsonParser with: '"Carpe Squeak"') read; readStringInternal]]
        timeToRunWithoutGC

13
Smalltalk specialObjectsArray at: 59 put: #conditionalBranchCounterTrippedOn:

Transcript showln: [10000 timesRepeat:
    [(JsonParser with: '"Carpe Squeak"') read; readStringInternal]]
        timeToRunWithoutGC

9
```

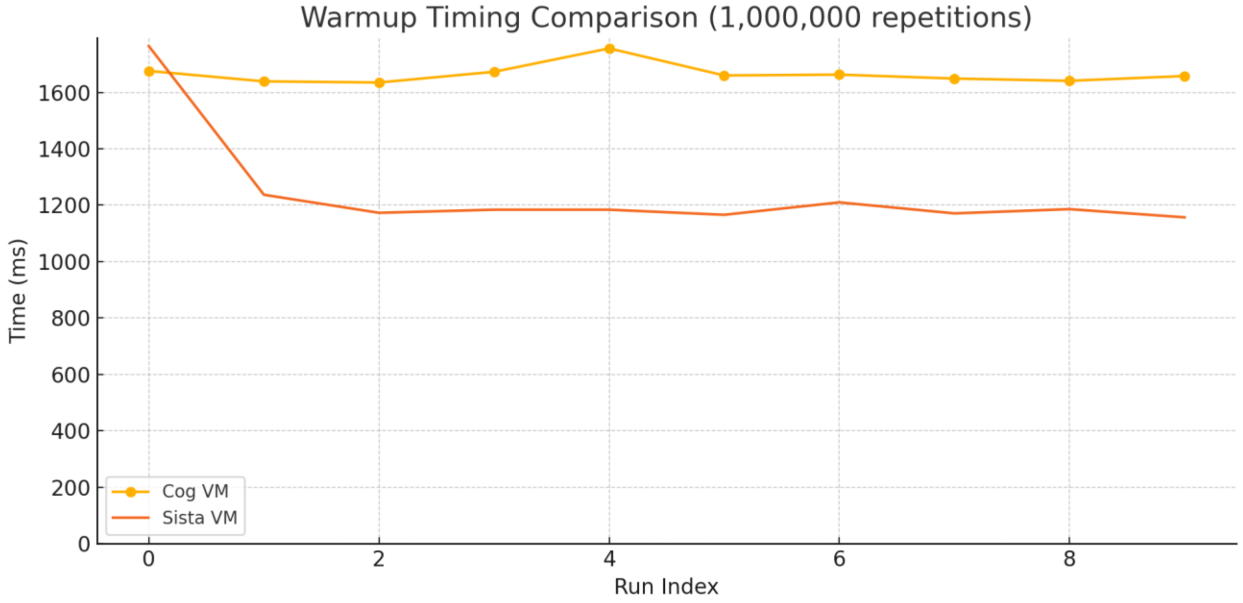
```
counter tripped in
16r2BEDB0 M ByteString class(String class)>new: 16rD6FED8: a(n) ByteString

counter tripped in
16r2BEE20 M ByteString(String)>= 16r755F48: a(n) ByteString

counter tripped in
16r2BEDE8 M String class>with: 16rD700B8: a(n) String

counter tripped in
16r2BEDA8 M ByteString class(String class)>new: 16rD6FED8: a(n) ByteString
```

```
3,079,585/26,970
```

OSVM: Bootstrapping Scorch

Running Benchmarks

```
[1000000 timesRepeat:
    [(JsonParser with: '"Carpe Squeak"') read; readStringInternal]]
        timeToRunWithoutGC
```



🥳
1.4x faster

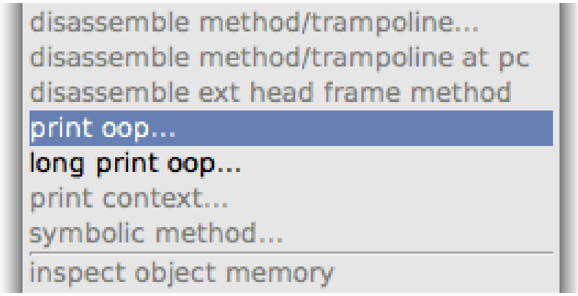
	Cog	Sista
Avg	1 665 ms	1 243 ms
Min	1 635 ms	1 157 ms
Max	1 756 ms	1 764 ms
Total	16 650 ms	12 432 ms

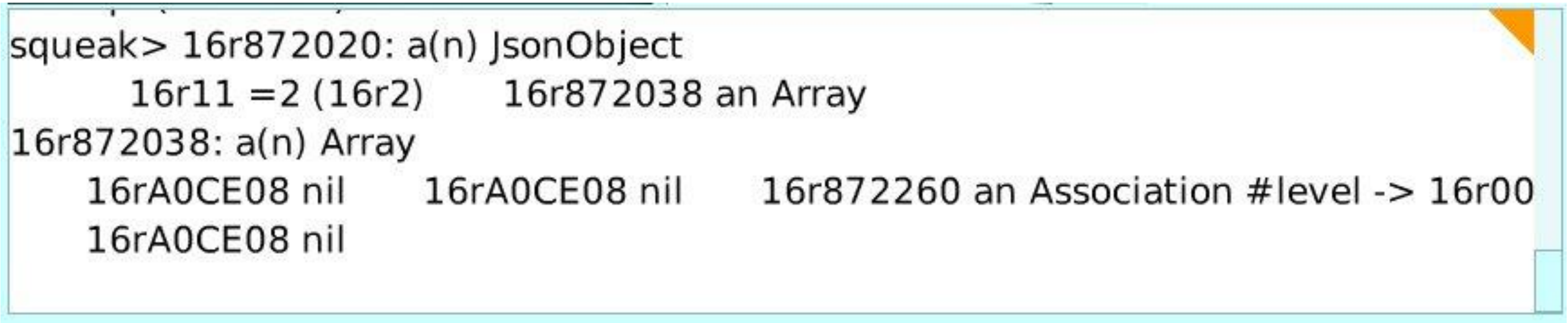
Benchmark Specs
i5-6267U @ 2.9 GHz x 4, 6 GB RAM, Ubuntu 18.04

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection

State of the art: **Low-level, non-interactive text output**


disassemble method/trampoline...
disassemble method/trampoline at pc
disassemble ext head frame method
print oop...
long print oop...
print context...
symbolic method...
inspect object memory

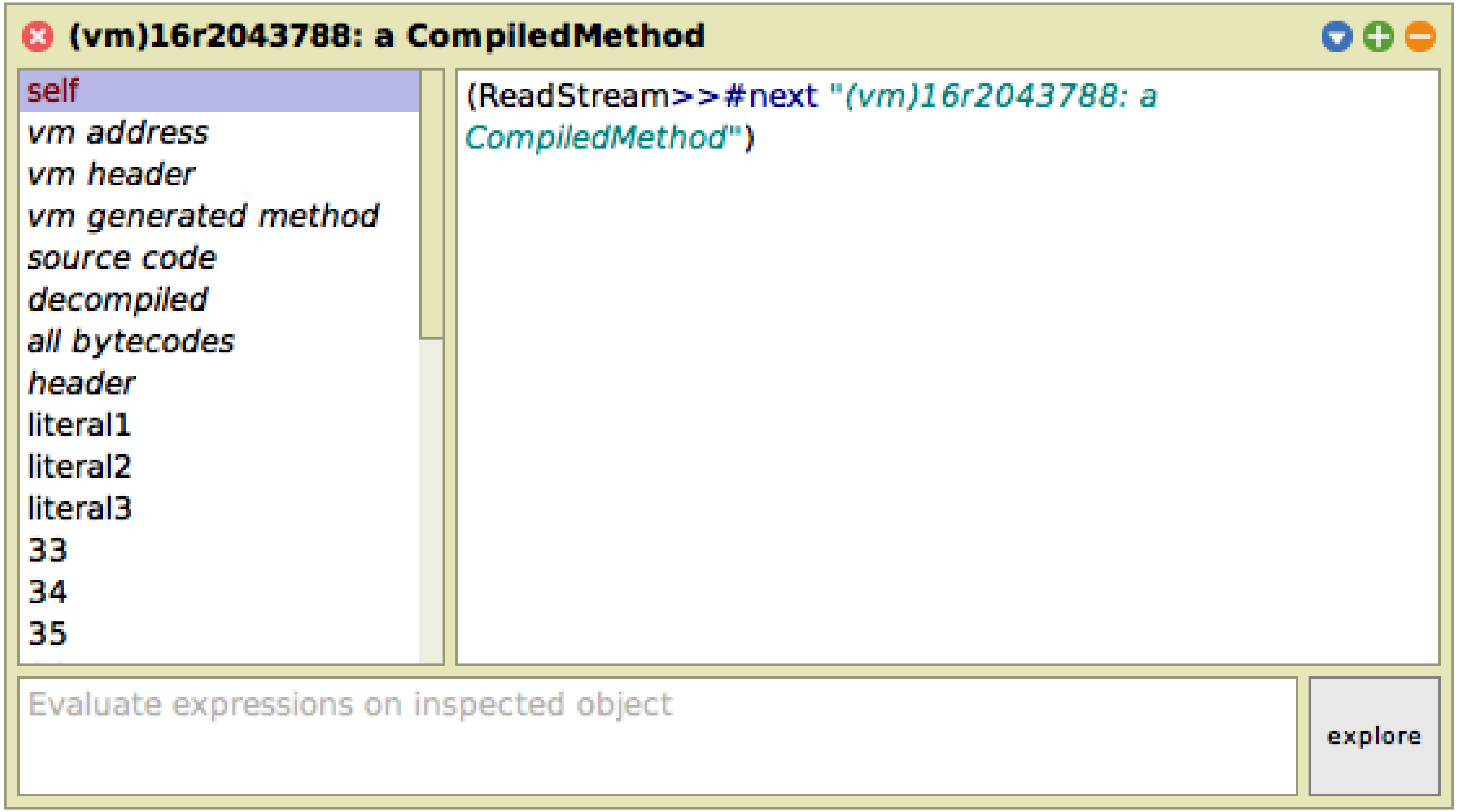


```
squeak> 16r872020: a(n) JsonObject
        16r11 =2 (16r2)      16r872038 an Array
16r872038: a(n) Array
     16rA0CE08 nil      16rA0CE08 nil      16r872260 an Association #level -> 16r00
     16rA0CE08 nil
```

Transcript output of simulator

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



(vm)16r2043788: a CompiledMethod

- self
- *vm address*
- *vm header*
- *vm generated method*
- *source code*
- *decompiled*
- *all bytecodes*
- *header*
- literal1
- literal2
- literal3
- 33
- 34
- 35

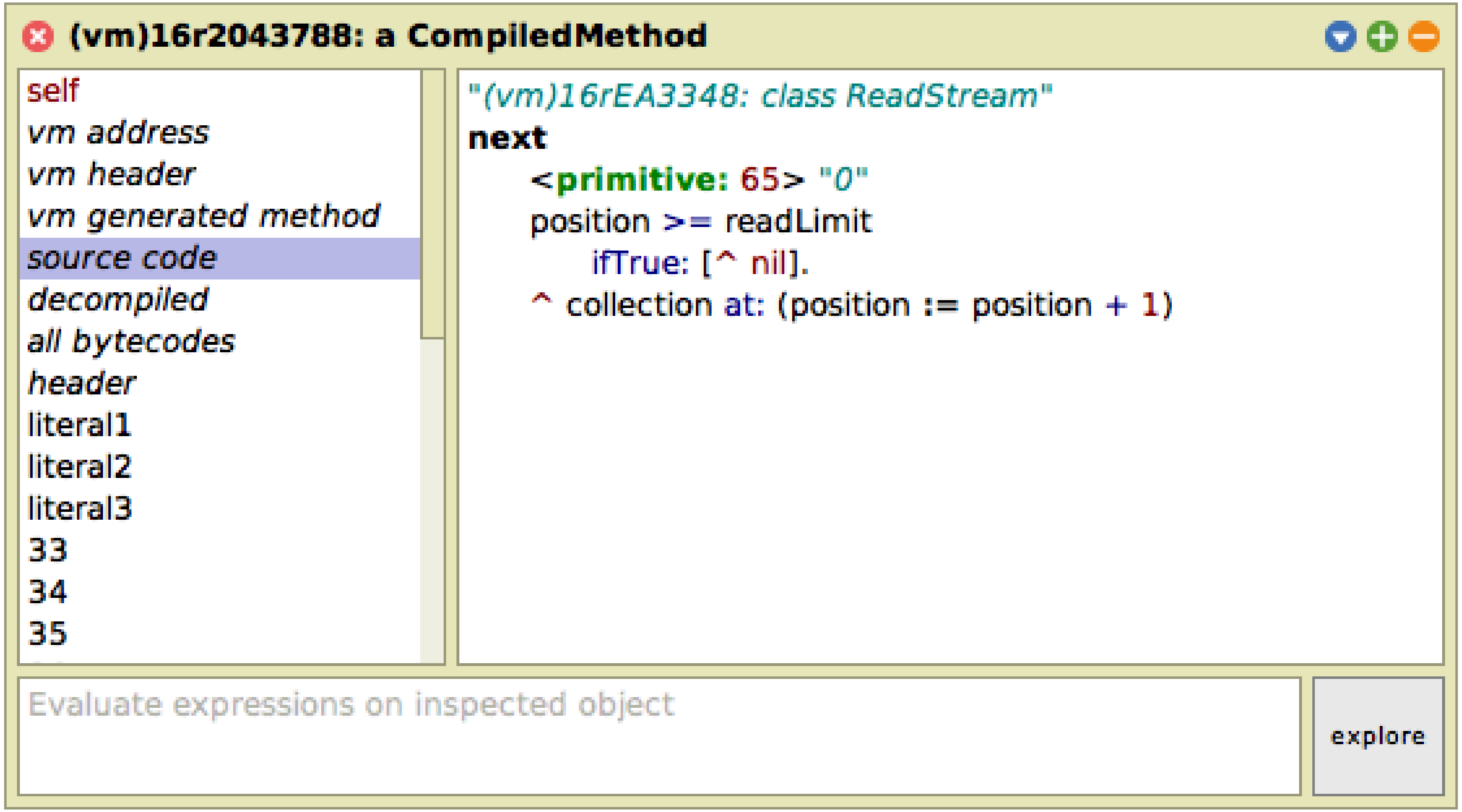
```
(ReadStream>>#next "(vm)16r2043788: a
CompiledMethod")
```

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



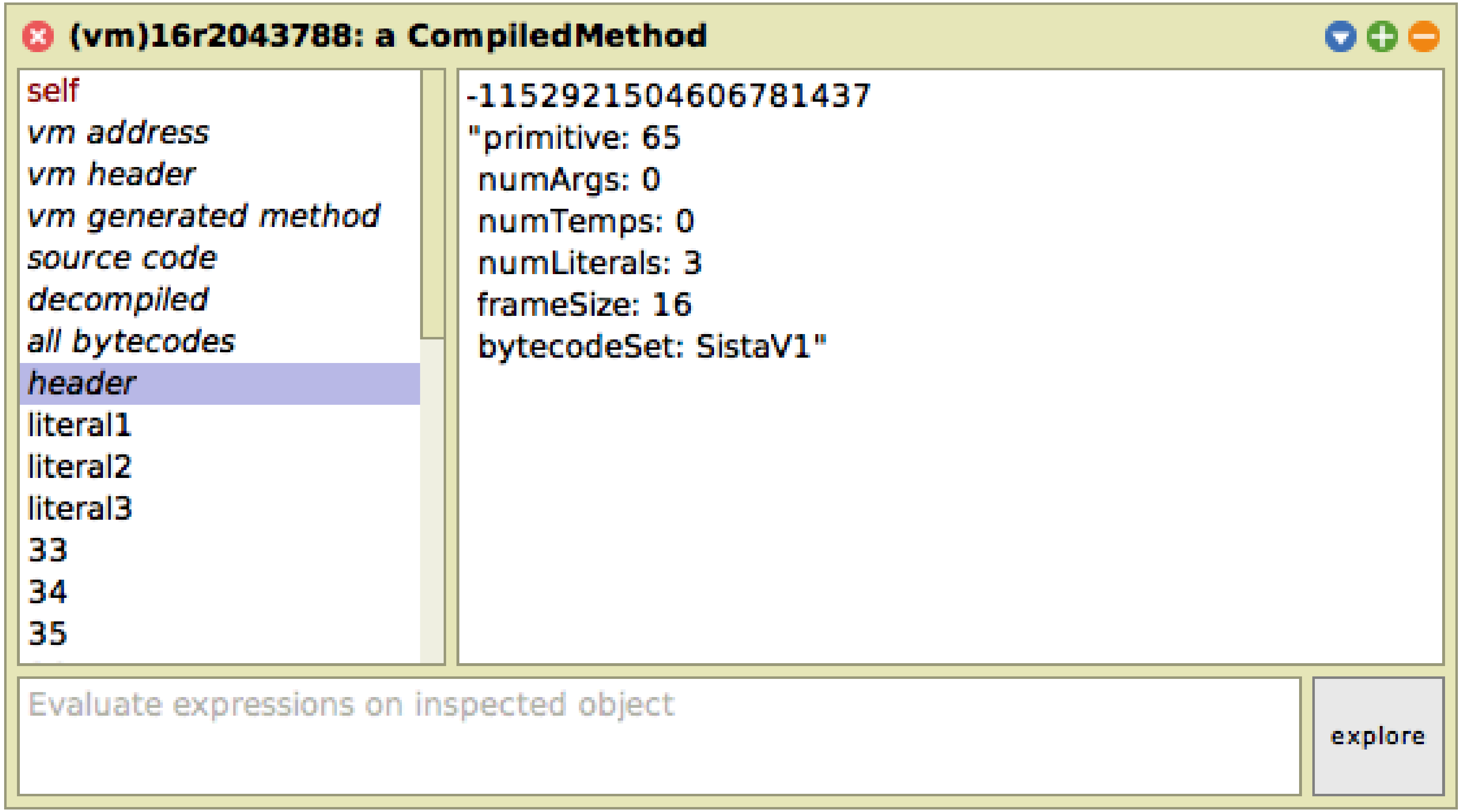
```
"(vm)16rEA3348: class ReadStream"
next
	<primitive: 65> "0"
	position >= readLimit
		ifTrue: [^ nil].
	^ collection at: (position := position + 1)
```

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



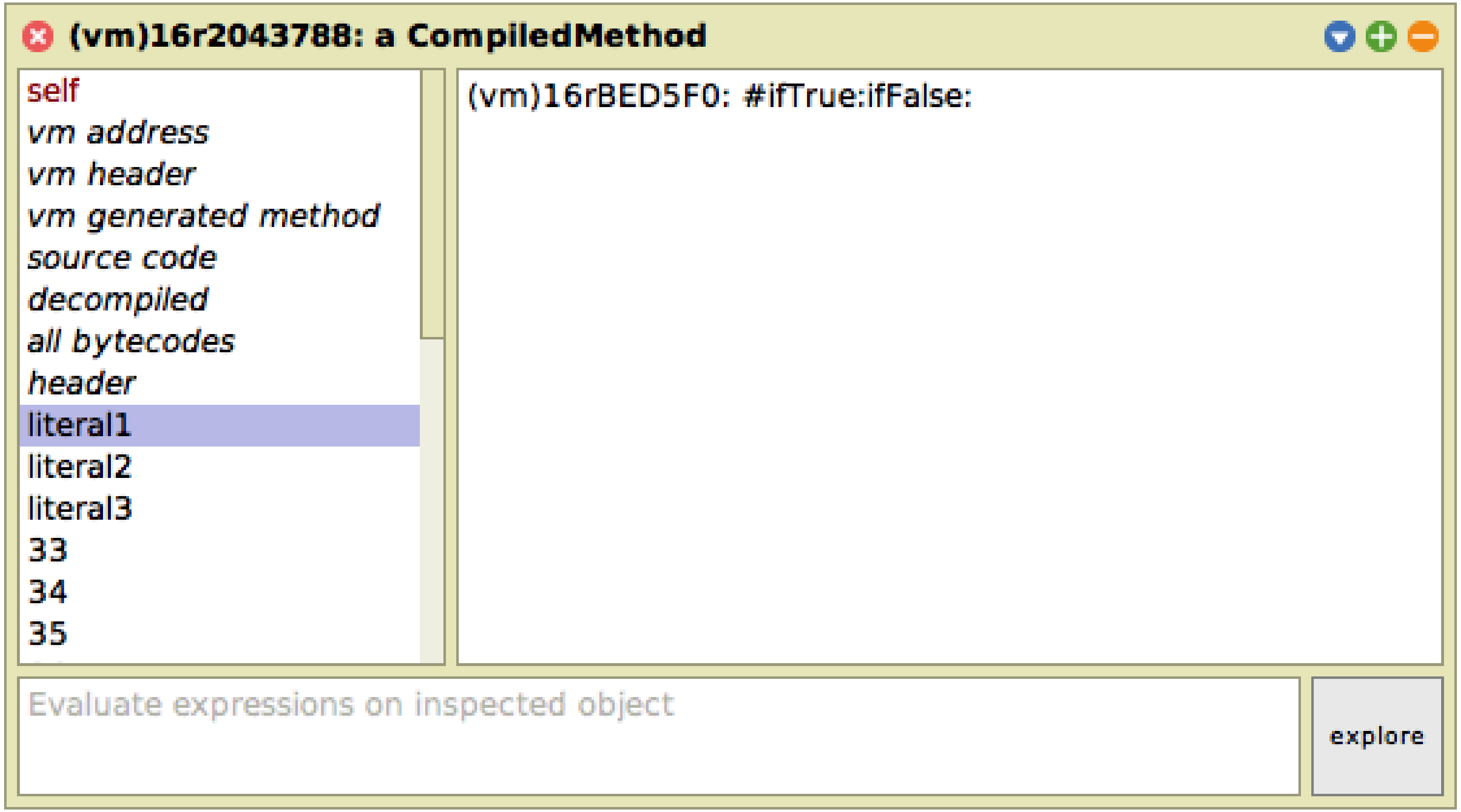
```
-1152921504606781437
"primitive: 65
 numArgs: 0
 numTemps: 0
 numLiterals: 3
 frameSize: 16
 bytecodeSet: SistaV1"
```

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



(vm)16r2043788: a CompiledMethod

self
vm address
vm header
vm generated method
source code
decompiled
all bytecodes
header
literal1
literal2
literal3
33
34
35

(vm)16rBED5F0: #ifTrue:ifFalse:

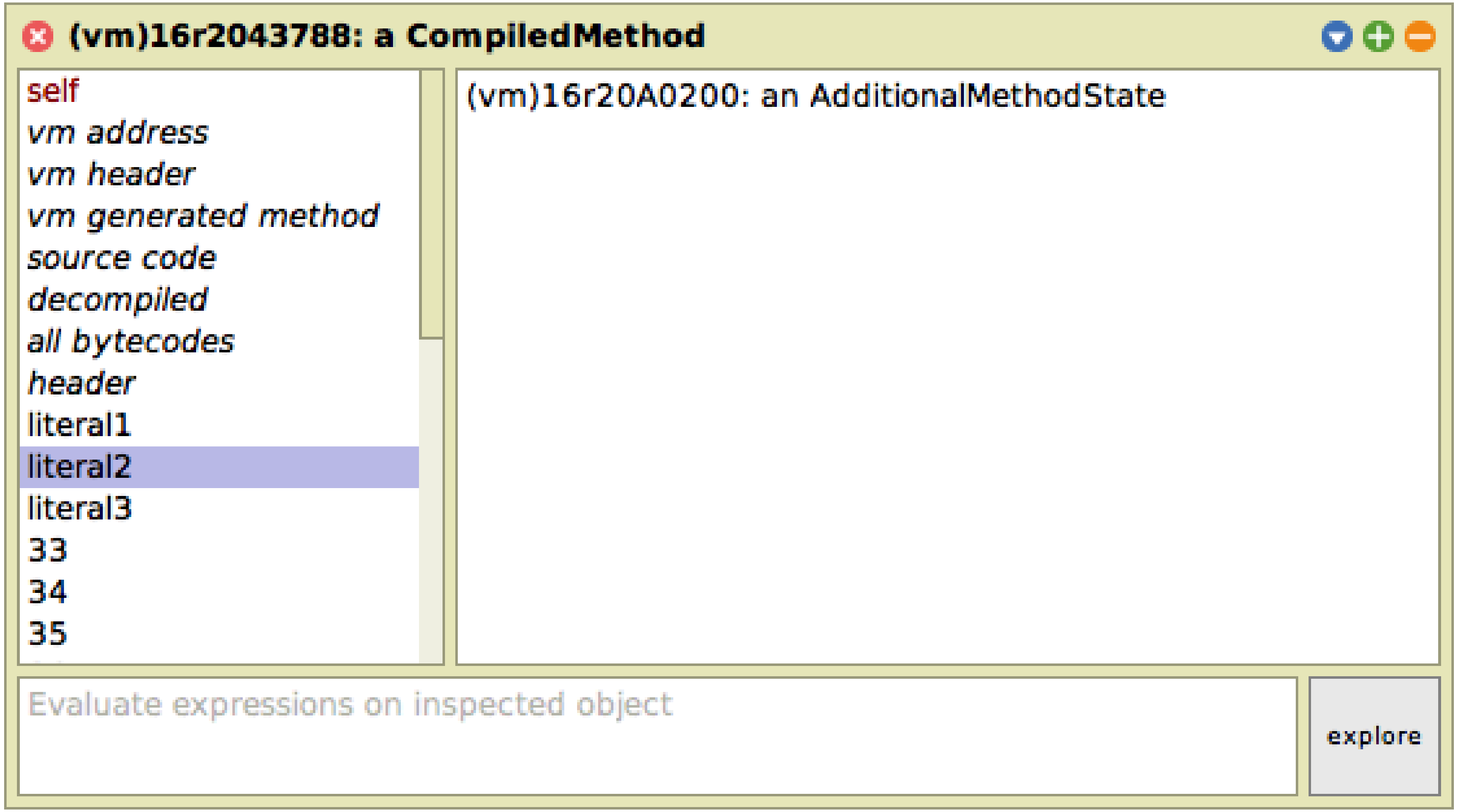
Evaluate expressions on inspected object

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



(vm)16r2043788: a CompiledMethod

self
vm address
vm header
vm generated method
source code
decompiled
all bytecodes
header
literal1
literal2
literal3
33
34
35

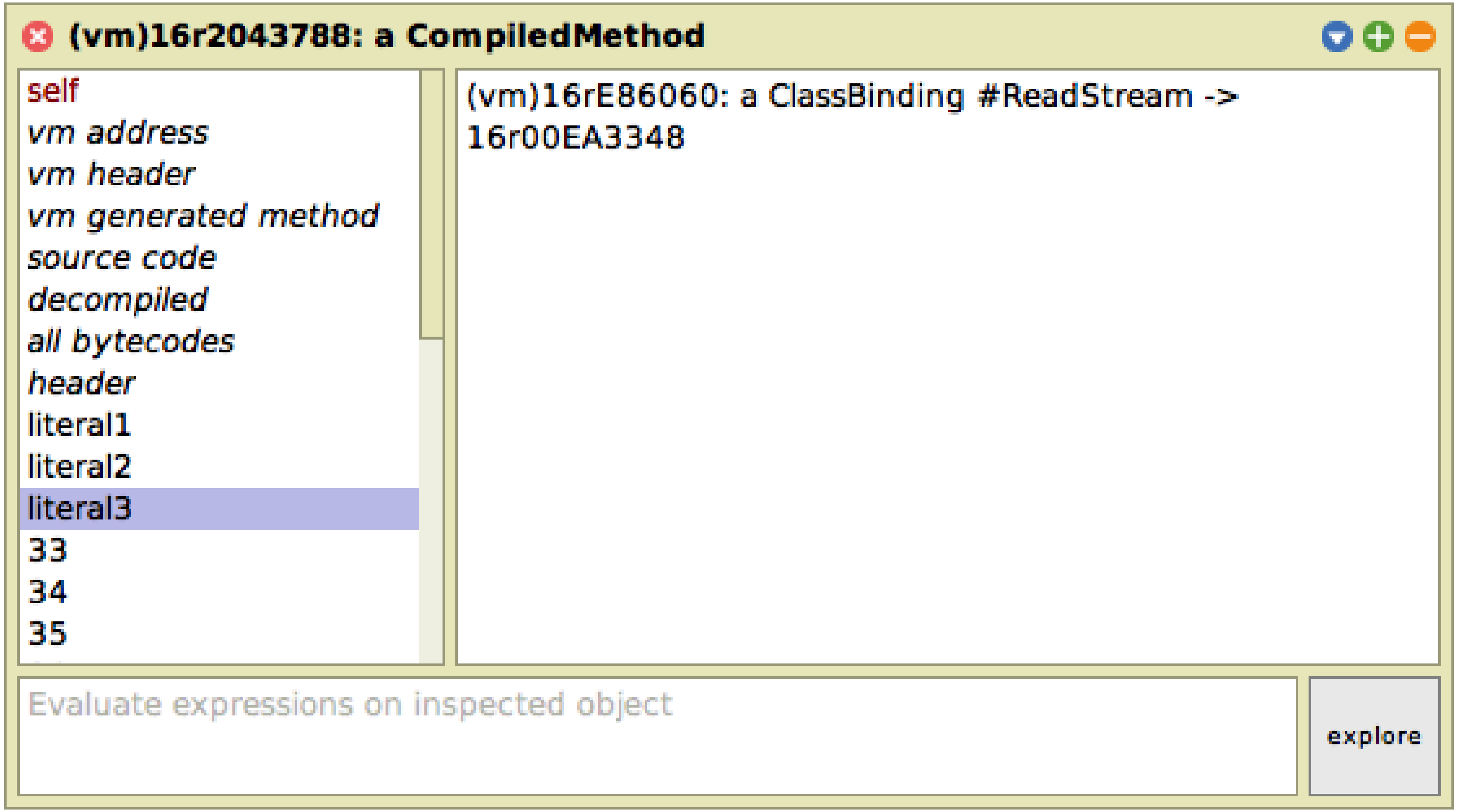
(vm)16r20A0200: an AdditionalMethodState

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



(vm)16r2043788: a CompiledMethod

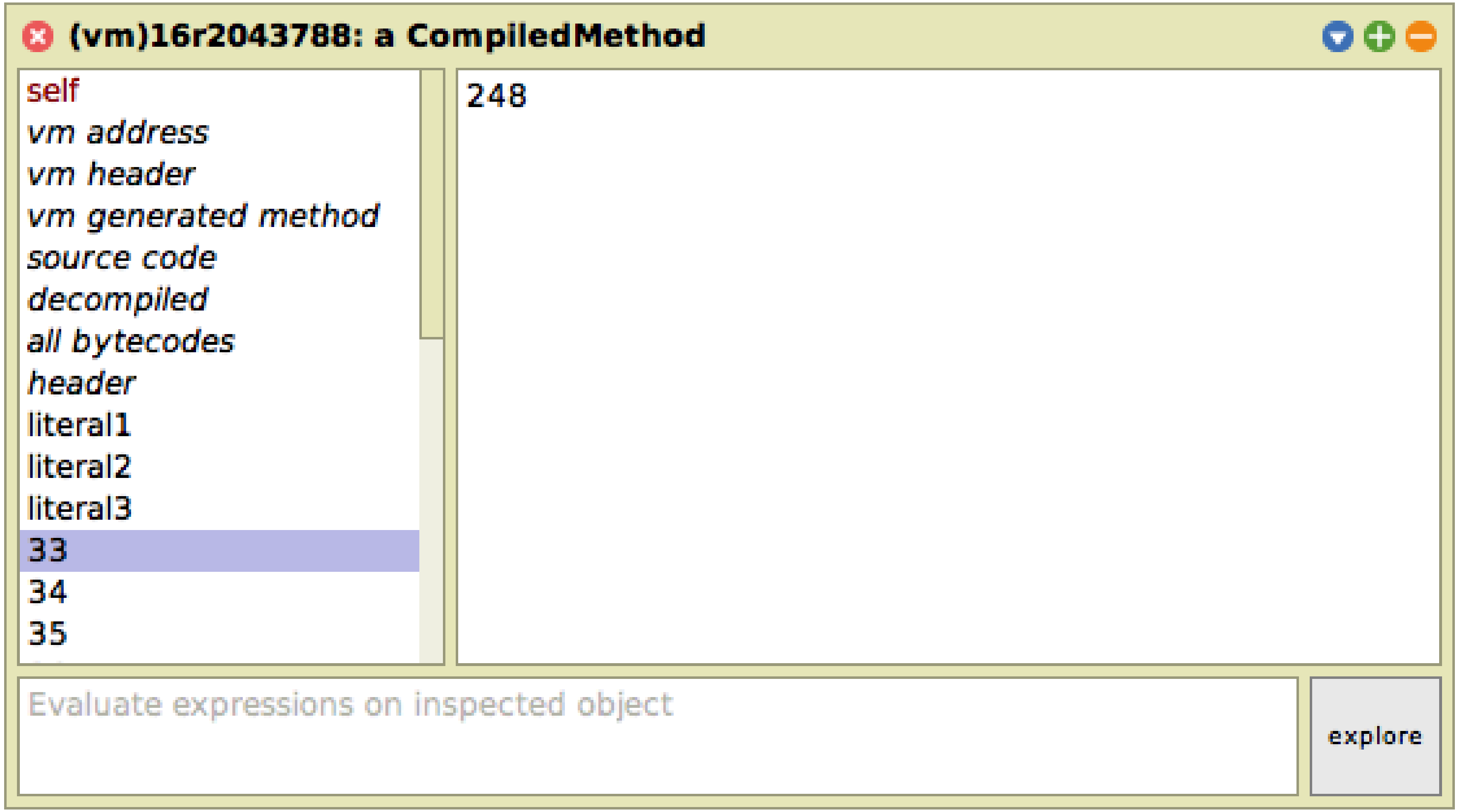
(vm)16rE86060: a ClassBinding #ReadStream -> 16r00EA3348

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



(vm)16r2043788: a CompiledMethod

- self
- *vm address*
- *vm header*
- *vm generated method*
- *source code*
- *decompiled*
- *all bytecodes*
- *header*
- literal1
- literal2
- literal3
- 33
- 34
- 35

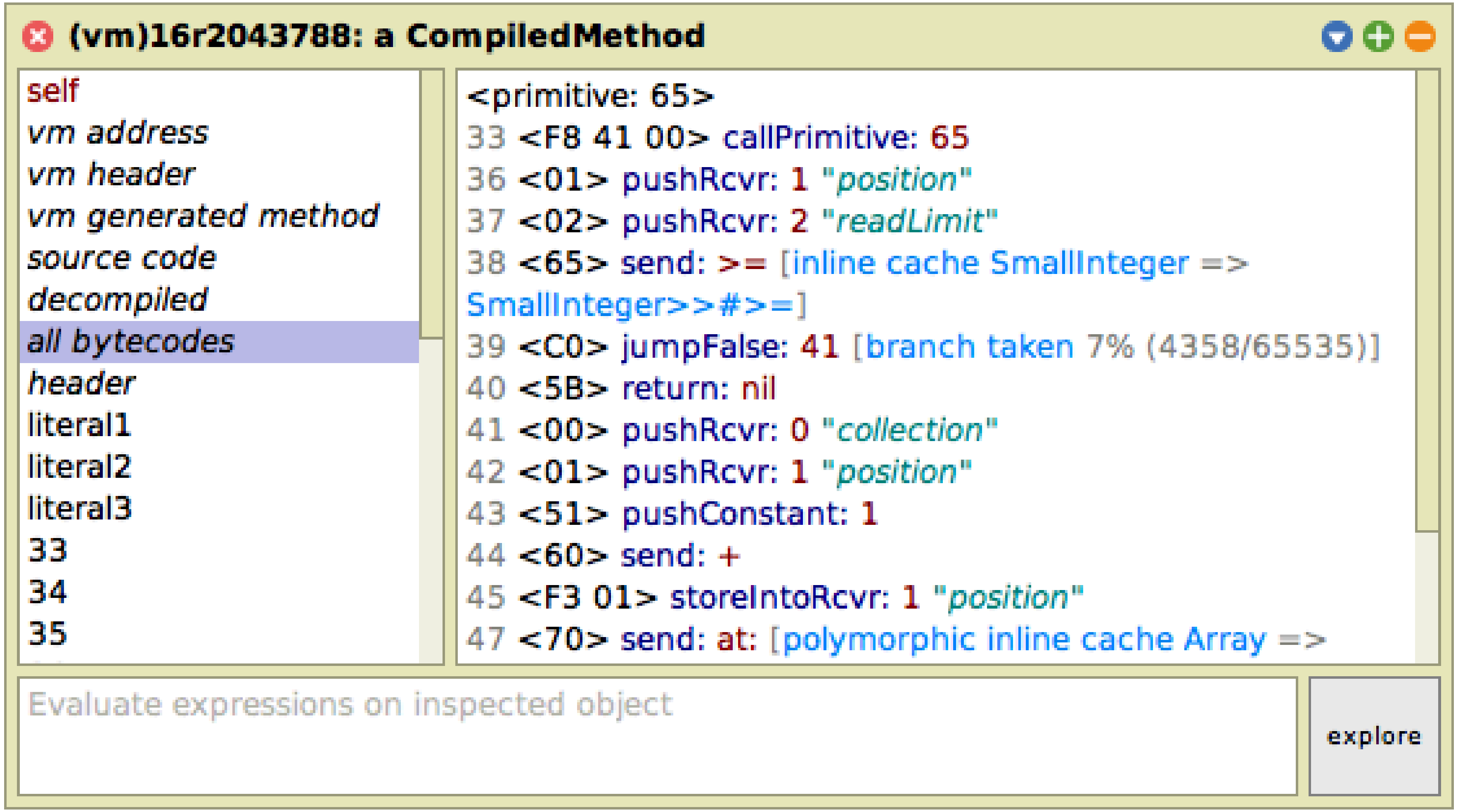
```
248
```

Inspectors on method proxies

Reuse rich, domain-specific tools to inspect and modify objects of from simulated image

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection



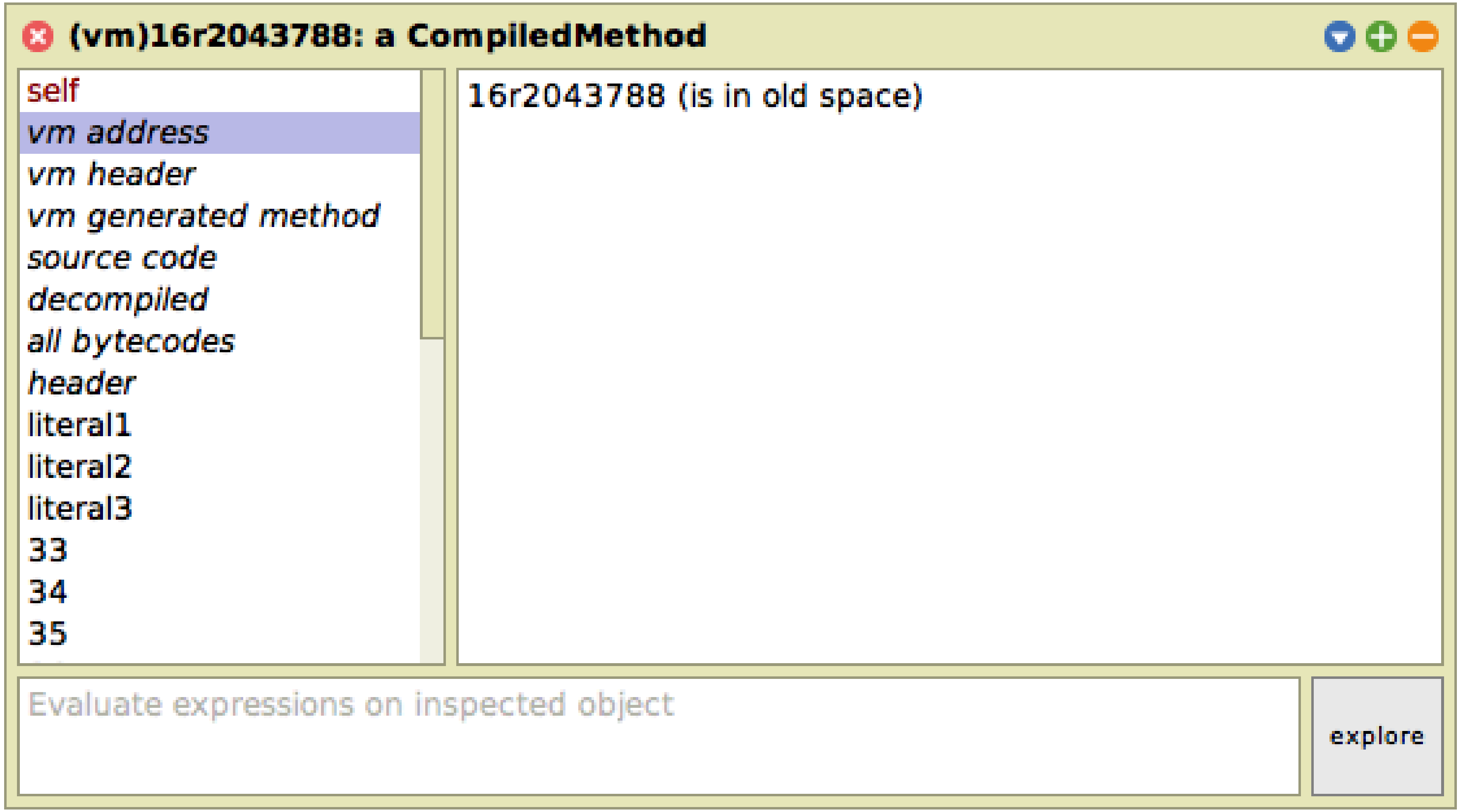
(vm)16r2043788: a CompiledMethod

```
<primitive: 65>
33 <F8 41 00> callPrimitive: 65
36 <01> pushRcvr: 1 "position"
37 <02> pushRcvr: 2 "readLimit"
38 <65> send: >= [inline cache SmallInteger =>
SmallInteger>>#>=]
39 <C0> jumpFalse: 41 [branch taken 7% (4358/65535)]
40 <5B> return: nil
41 <00> pushRcvr: 0 "collection"
42 <01> pushRcvr: 1 "position"
43 <51> pushConstant: 1
44 <60> send: +
45 <F3 01> storeIntoRcvr: 1 "position"
47 <70> send: at: [polymorphic inline cache Array =>
```

Advanced bytecode printer with Sista branch counters and syntax highlighting

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection

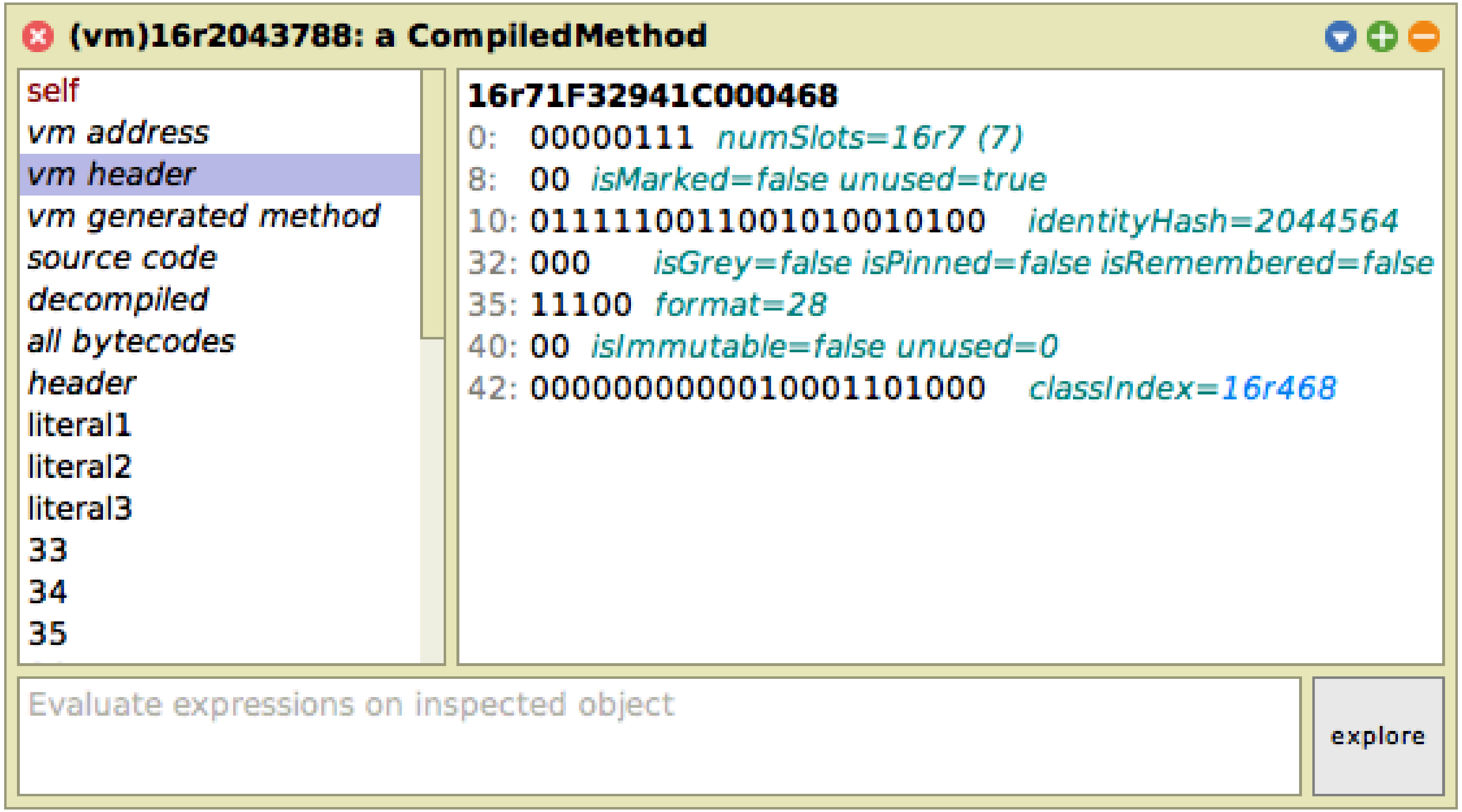


16r2043788 (is in old space)

Additional VM-specific inspection fields

OSVM: Improving Exploratory Programming in the Simulator

Object Inspection

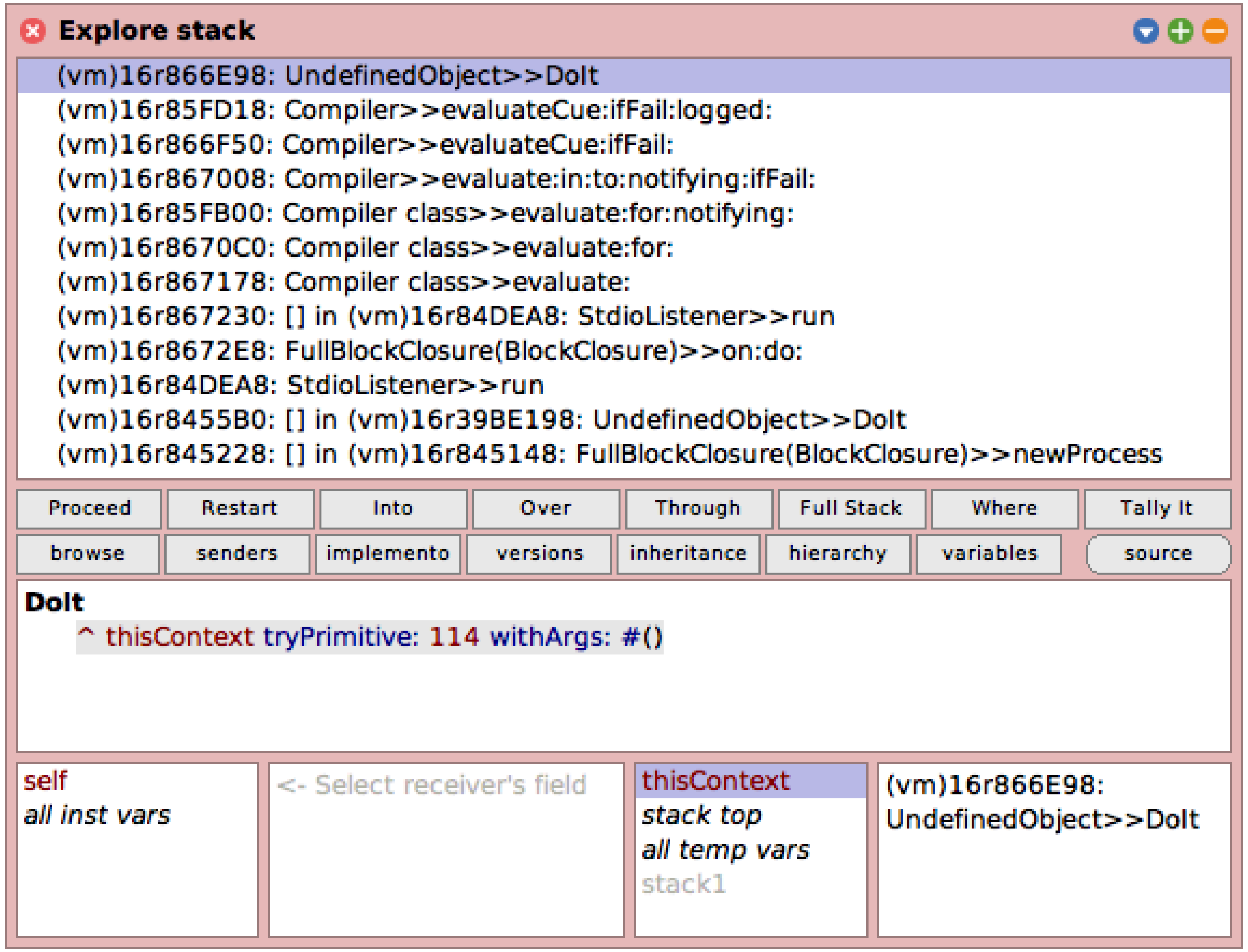


```
16r71F32941C000468
0:   00000111  numSlots=16r7 (7)
8:   00  isMarked=false unused=true
10: 0111110011001010010100    identityHash=2044564
32: 000       isGrey=false isPinned=false isRemembered=false
35: 11100  format=28
40: 00  isImmutable=false unused=0
42: 0000000000010001101000    classIndex=16r468
```

Additional VM-specific inspection fields

OSVM: Improving Exploratory Programming in the Simulator

Debugging



```
(vm)16r866E98: UndefinedObject>>DoIt
(vm)16r85FD18: Compiler>>evaluateCue:ifFail:logged:
(vm)16r866F50: Compiler>>evaluateCue:ifFail:
(vm)16r867008: Compiler>>evaluate:in:to:notifying:ifFail:
(vm)16r85FB00: Compiler class>>evaluate:for:notifying:
(vm)16r8670C0: Compiler class>>evaluate:for:
(vm)16r867178: Compiler class>>evaluate:
(vm)16r867230: [] in (vm)16r84DEA8: StdioListener>>run
(vm)16r8672E8: FullBlockClosure(BlockClosure)>>on:do:
(vm)16r84DEA8: StdioListener>>run
(vm)16r8455B0: [] in (vm)16r39BE198: UndefinedObject>>DoIt
(vm)16r845228: [] in (vm)16r845148: FullBlockClosure(BlockClosure)>>newProcess
```

```
DoIt
	^ thisContext tryPrimitive: 114 withArgs: #()
```

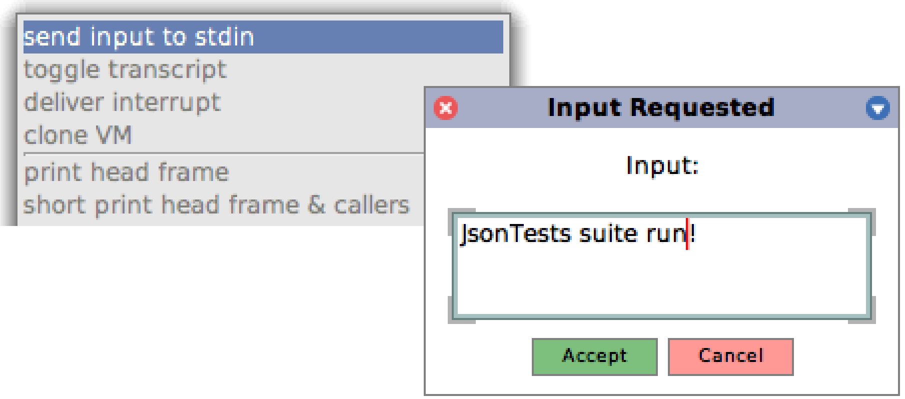
(vm)16r866E98: UndefinedObject>>DoIt

Debuggers on context proxies (currently read-only)

```
(Process forContext: aContextProxy)
    debugWithTitle: 'Explore stack'
```

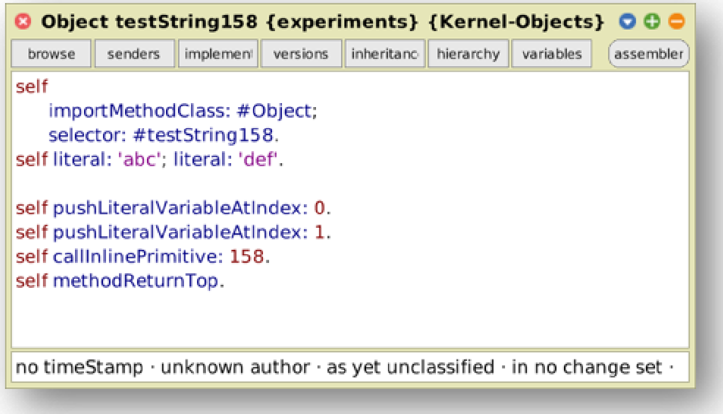
OSVM: Improving Exploratory Programming in the Simulator

Other Contributions



JsonTests suite run!

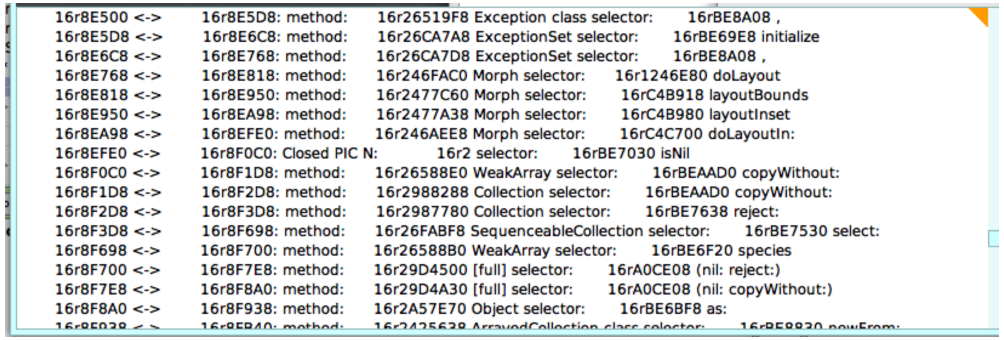
Added non-modal REPL interface



```
self
    importMethodClass: #Object;
    selector: #testString158.
self literal: 'abc'; literal: 'def'.

self pushLiteralVariableAtIndex: 0.
self pushLiteralVariableAtIndex: 1.
self callInlinePrimitive: 158.
self methodReturnTop.
```

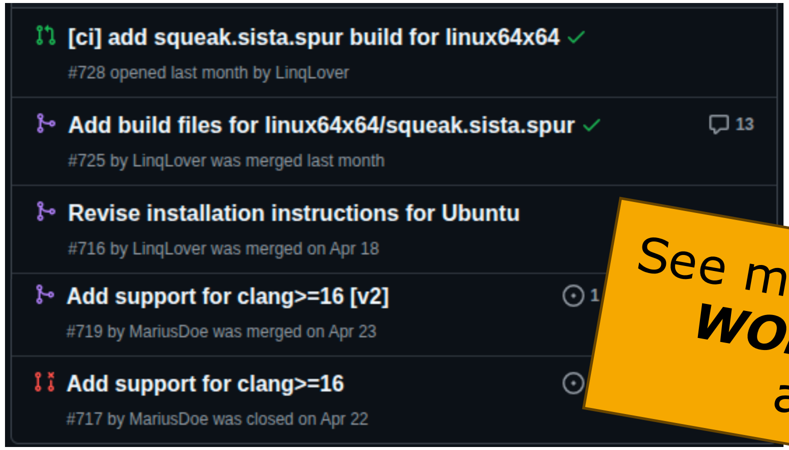
no timeStamp · unknown author · as yet unclassified · in no change set ·

New bytecode editor



```
16r8E500 <->    16r8E5D8: method:    16r26519F8 Exception class selector:    16rBE8A08 ,
16r8E5D8 <->    16r8E6C8: method:    16r26CA7A8 ExceptionSet selector:    16rBE69E8 initialize
16r8E6C8 <->    16r8E768: method:    16r26CA7D8 ExceptionSet selector:    16rBE8A08 ,
16r8E768 <->    16r8E818: method:    16r246FAC0 Morph selector:    16r1246E80 doLayout
16r8E818 <->    16r8E950: method:    16r2477C60 Morph selector:    16rC4B918 layoutBounds
16r8E950 <->    16r8EA98: method:    16r2477A38 Morph selector:    16rC4B980 layoutInset
16r8EA98 <->    16r8EFE0: method:    16r246AEE8 Morph selector:    16rC4C700 doLayoutIn:
16r8EFE0 <->    16r8F0C0: Closed PIC N:       16r2 selector:    16rBE7030 isNil
16r8F0C0 <->    16r8F1D8: method:    16r26588E0 WeakArray selector:    16rBEAAD0 copyWithout:
16r8F1D8 <->    16r8F2D8: method:    16r2988288 Collection selector:    16rBEAAD0 copyWithout:
16r8F2D8 <->    16r8F3D8: method:    16r2987780 Collection selector:    16rBE7638 reject:
16r8F3D8 <->    16r8F698: method:    16r26FABF8 SequenceableCollection selector:    16rBE7530 select:
16r8F698 <->    16r8F700: method:    16r26588B0 WeakArray selector:    16rBE6F20 species
16r8F700 <->    16r8F7E8: method:    16r29D4500 [full] selector:    16rA0CE08 (nil: reject:)
16r8F7E8 <->    16r8F8A0: method:    16r29D4A30 [full] selector:    16rA0CE08 (nil: copyWithout:)
16r8F8A0 <->    16r8F938: method:    16r2A57E70 Object selector:    16rBE6BF8 as:
16r8F938 <->    16r8FB40: method:    16r2425638 ArrayedCollection class selector:    16rBE8830 newFrom:
```

Improved Transcript performance



[ci] add squeak.sista.spur build for linux64x64 ✓
#728 opened last month by LinqLover

Add build files for linux64x64/squeak.sista.spur ✓ 13
#725 by LinqLover was merged last month

Revise installation instructions for Ubuntu
#716 by LinqLover was merged on Apr 18

Add support for clang>=16 [v2] 1
#719 by MariusDoe was merged on Apr 23

Add support for clang>=16
#717 by MariusDoe was closed on Apr 22

See more demos at the **WORLD CAFÉ** at 3.30pm

Improved build chain & documentation

OSVM: Future Work

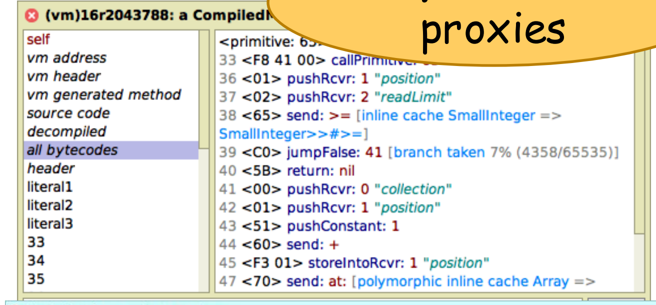
- **The road ahead for Sista/Scorch**
  - **Iteratively extend Sista/Scorch** with new unsafe bytecodes and optimizations using our proxy framework
  - **Harden and test** the new VM for a release
- **Open todos for our proxy framework**
  - Implement **missing edge cases** for context proxies
    - **Manipulation of top contexts** for external debugging
    - Complete trap support for **Scorch deoptimization**
  - **Accelerate** the proxy accesses by usings **bytecode rewriting** instead of context simulation

- How could we further **improve program exploration** in the simulator?
  - Introduce **additional VM inspectors** for (unmarried) frames, stack pages, immediate values, …?
  - Build an interactive, graphical **debugger for machine code?**

Literature

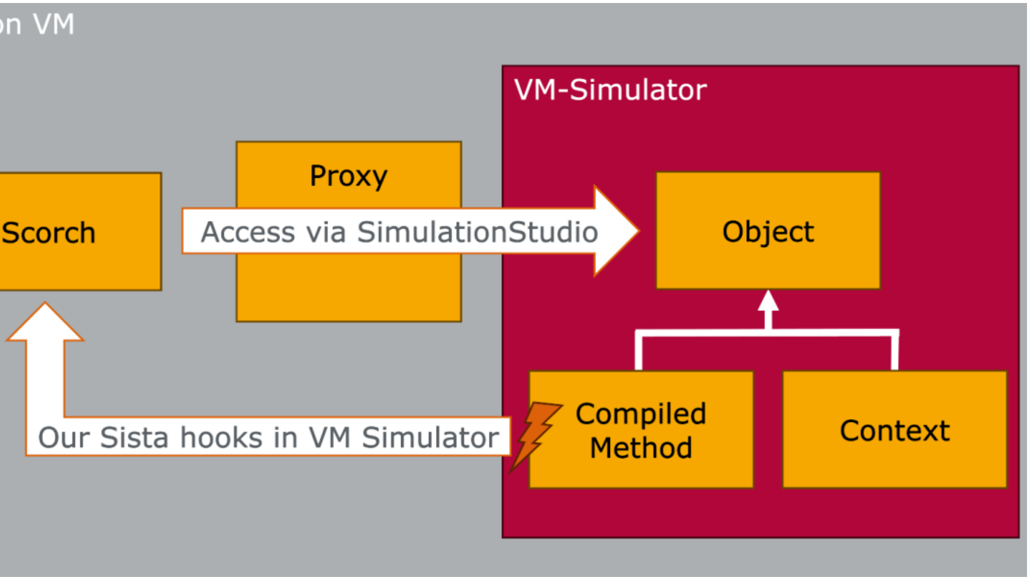
- [Bera, 2017a] Clément Béra, Eliot Miranda, Tim Felgentreff, Marcus Denker, and Stéphane Ducasse. 2017. **Sista: Saving Optimized Code in Snapshots for Fast Start-Up.** In *Managed Languages and Runtimes (ManLang 2017)*. 11. https://doi.org/10.1145/3132190.3132201
- [Bera, 2017b] Clément Béra. 2017. **Sista: A Metacircular Architecture for Runtime Optimisation Persistence.** Programming Languages [cs.PL]. Dissertation, Université de Lille 1. https://theses.hal.science/tel-01634137
- [Miranda, 2018] Eliot Miranda, Clément Béra, Elisa Gonzalez Boix, and Dan Ingalls. 2018. **Two Decades of Smalltalk VM Development: Live VM Development through Simulation Tools.** In *Proceedings of the 10th ACM SIGPLAN International Workshop on Virtual Machines and Intermediate Languages (VMIL '18)*, November 4, 2018, Boston, MA, USA. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3281287.3281295
- [Niephaus, 2022] Fabio Niephaus. 2022. **Exploratory Tool-building Platforms for Polyglot Virtual Machines.** Dissertation, Potsdam, Universität Potsdam. https://publishup.uni-potsdam.de/frontdoor/index/index/docId/57177
- [Würthinger, 2013] Thomas Würthinger et al. **One VM To Rule Them All.** In *Proceedings of the 2013 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming & Software*. 2013. https://dl.acm.org/doi/10.1145/2509578.2509581

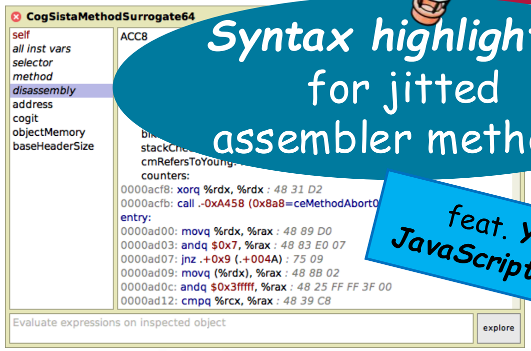
OOPSIE: Object Oriented PointerS Interaction Engine



```
(vm)16r2043788: a CompiledM
self
vm address
vm header
vm generated method
source code
decompiled
all bytecodes
header
literal1
literal2
literal3
33
34
35

<primitive: 65
33 <F8 41 00> callPrimitive:
36 <01> pushRcvr: 1 "position"
37 <02> pushRcvr: 2 "readLimit"
38 <65> send: >= [inline cache SmallInteger =>
SmallInteger>>#>=]
39 <C0> jumpFalse: 41 [branch taken 7% (4358/65535)]
40 <5B> return: nil
41 <00> pushRcvr: 0 "collection"
42 <01> pushRcvr: 1 "position"
43 <51> pushConstant: 1
44 <60> send: +
45 <F3 01> storeIntoRcvr: 1 "position"
47 <70> send: at: [polymorphic inline cache Array =>
```

Inspectors **on proxies**

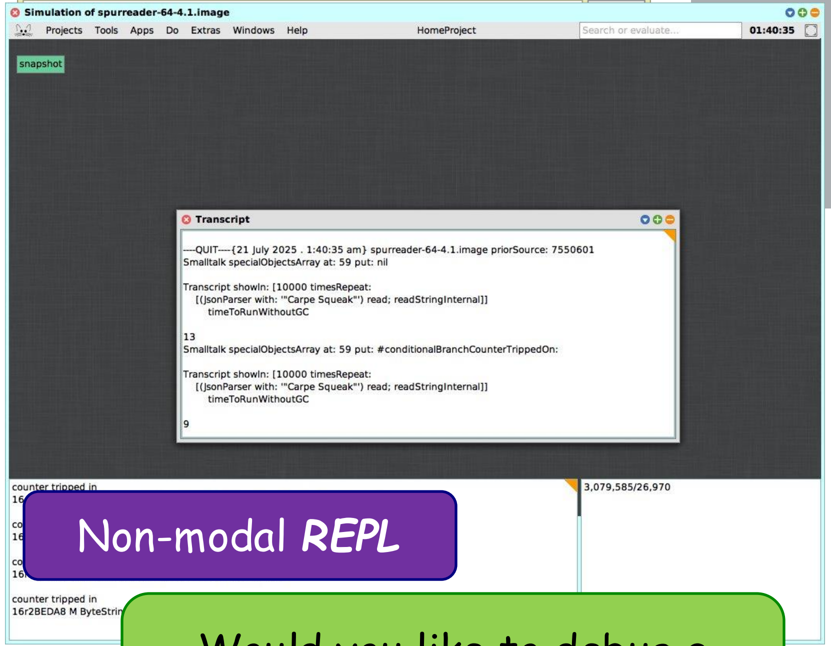




```
CogSistaMethodSurrogate64
self
all inst vars
selector
method
disassembly
address
cogit
objectMemory
baseHeaderSize

ACC8
   bl
   stackChe
   cmRefersToYoung:
   counters:
0000acf8: xorq %rdx, %rdx : 48 31 D2
0000acfb: call .-0xA458 (0x8a8=ceMethodAbort0
entry:
0000ad00: movq %rdx, %rax : 48 89 D0
0000ad03: andq $0x7, %rax : 48 83 E0 07
0000ad07: jnz .+0x9 (.+004A) : 75 09
0000ad09: movq (%rdx), %rax : 48 8B 02
0000ad0c: andq $0x3fffff, %rax : 48 25 FF FF 3F 00
0000ad12: cmpq %rcx, %rax : 48 39 C8
Evaluate expressions on inspected object
```

Syntax highlighting **for jitted assembler methods**

feat. ***Yaros*-*JavaScript*** **bindings**

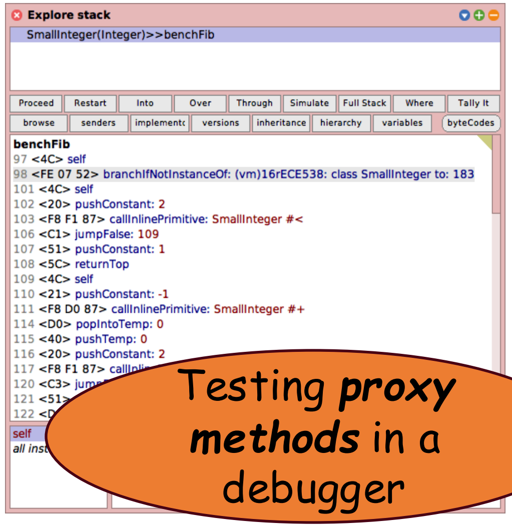


```
----QUIT----{21 July 2025 . 1:40:35 am} spurreader-64-4.1.image priorSource: 7550601
Smalltalk specialObjectsArray at: 59 put: nil

Transcript showln: [10000 timesRepeat:
    [(JsonParser with: '"Carpe Squeak"') read; readStringInternal]]
        timeToRunWithoutGC

13
Smalltalk specialObjectsArray at: 59 put: #conditionalBranchCounterTrippedOn:

Transcript showln: [10000 timesRepeat:
    [(JsonParser with: '"Carpe Squeak"') read; readStringInternal]]
        timeToRunWithoutGC

9
```

```
counter tripped in
16
co
16
co
16
counter tripped in
16r2BEDA8 M ByteStrin

3,079,585/26,970
```

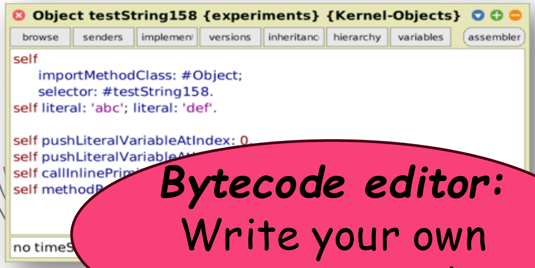
Non-modal *REPL*

Would you like to debug a ***proxy context?***



```
SmallInteger(Integer)>>benchFib

benchFib
97 <4C> self
98 <FE 07 52> branchIfNotInstanceOf: (vm)16rECE538: class SmallInteger to: 183
101 <4C> self
102 <20> pushConstant: 2
103 <F8 F1 87> callInlinePrimitive: SmallInteger #<
106 <C1> jumpFalse: 109
107 <51> pushConstant: 1
108 <5C> returnTop
109 <4C> self
110 <21> pushConstant: -1
111 <F8 D0 87> callInlinePrimitive: SmallInteger #+
114 <D0> popIntoTemp: 0
115 <40> pushTemp: 0
116 <20> pushConstant: 2
117 <F8 F1 87> callInli
120 <C3> jump
121 <51>
122 <D

self
all inst
```

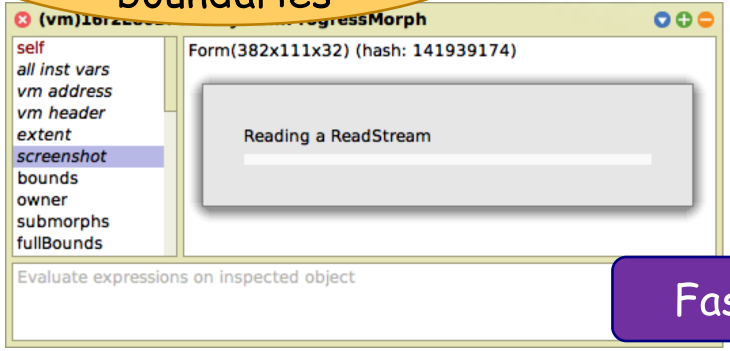
Testing *proxy methods* in a debugger



```
self
    importMethodClass: #Object;
    selector: #testString158.
self literal: 'abc'; literal: 'def'.

self pushLiteralVariableAtIndex: 0.
self pushLiteralVariableA
self callInlinePrim
self methodR

no timeS
```

Bytecode editor: **Write your own instructions!**

Morphic **across simulation boundaries**



```
(vm)16r2...ProgressMorph
self
all inst vars
vm address
vm header
extent
screenshot
bounds
owner
submorphs
fullBounds

Form(382x111x32) (hash: 141939174)

Reading a ReadStream

Evaluate expressions on inspected object
```

Faster *transcript*

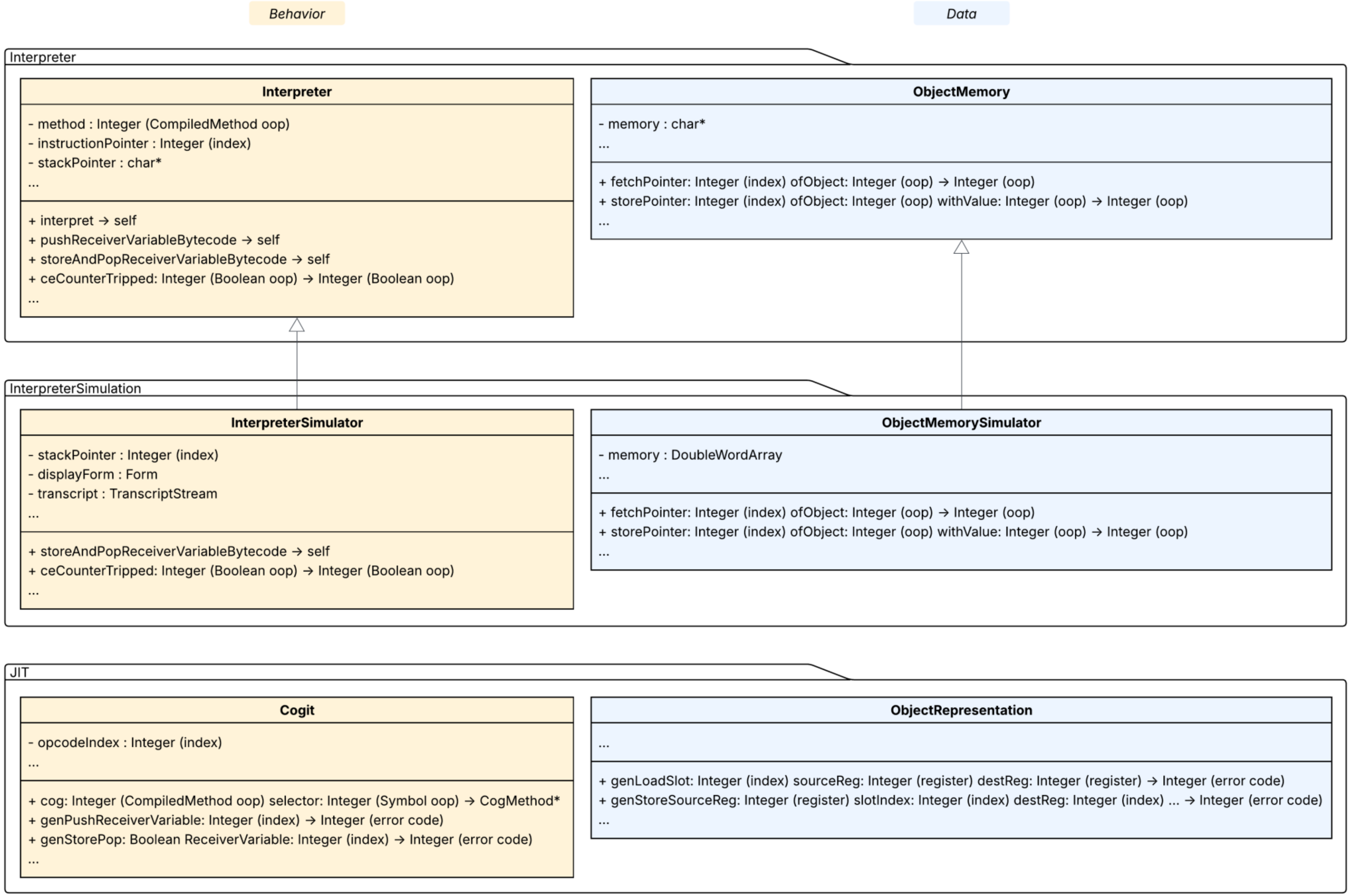
PROXIES ALL THE WAY DOWN

OSVM: Upstream Contributions

- **Scorch:** We ported the core of Scorch from Pharo to Squeak 6.1Alpha, made it compatible with our proxy simulator, and improved support for context simulation.
- **Squeak Trunk:**
- **Transparent Proxy Support**
  - resolveProxy.3.cs: Proposes a general pattern for resolving transparent proxies in primitive methods.
  - Kernel-ct.1604: Fixes Boolean support in ObjectViewer (which currently functions as the de-facto reference implementation of transparent proxies in Squeak).
  - Debugger step unexpected message sends.1.cs: Fixes stepping over ifTrue:/ifFalse: sends to Boolean proxies in the debugger.
- **Bytecode Representation and Execution**
  - Kernel-ct.1599: Fixes and revises CompiledCode constructors for SistaV1 bytecode set.
  - Kernel-ct.1600: Extends support for serializing CompiledCodes are storeStrings.
  - Kernel-ct.1601: Implements and documents the rare bytecode pushActiveProcess.
  - Kernel-ct.1605: Handles unusedBytecode trap from the VM by simulating unknown instructions in the context simulator.
  - Kernel-ct.1606: Adds context simulation of primitiveExitToDebugger.
- **Instruction Printing**
  - InstructionPrinter with style.3.cs: Adds text styling to all instruction printers in the Trunk and VMMaker.
  - Depends on:
    - Collections-ct.1087: Adds Stream>>#isTextStream.
    - addAttributesBack.7.cs: Adds reverse-ordered attribute accessors on Text and optimizes streaming of formatted texts to a lower complexity class.
- **VMMaker**
- VMMaker.oscog-mad.3558, VMMaker.oscog-mad.3559: Fixes for sendAndBranchData (primitiveSistaMethodPICAndCounterData).
- VMMaker.oscog-ct.3562: Adds breakpoint in simulator when encoutering an unknown instruction.
- VMMaker.oscog-ct.3541: Fixes and improves UI layout of VMMakerTool.
- VMMaker.oscog-ct.3556, VMMaker.oscog-mad.3557, VMMaker.oscog-ct.3563, Simulator-byteCountHelp.1.cs: Miscellaneous minor fixes and documentation improvements.
- **OpenSmalltalk VM**
- OpenSmalltalk/opensmalltalk-vm#716: Revise installation instructions for Ubuntu
- OpenSmalltalk/opensmalltalk-vm#719: Add support for clang>=16 [v2]
- OpenSmalltalk/opensmalltalk-vm#725: Add build files for linux64x64/squeak.sista.spur
- OpenSmalltalk/opensmalltalk-vm#728: [ci] add squeak.sista.spur build for linux64x64

OSVM: VMMaker Architecture

Simplified significantly, overview of most important classes only



Behavior

Data

Interpreter

Interpreter

- method : Integer (CompiledMethod oop)
- instructionPointer : Integer (index)
- stackPointer : char*
...

+ interpret → self
+ pushReceiverVariableBytecode → self
+ storeAndPopReceiverVariableBytecode → self
+ ceCounterTripped: Integer (Boolean oop) → Integer (Boolean oop)
...

ObjectMemory

- memory : char*
...

+ fetchPointer: Integer (index) ofObject: Integer (oop) → Integer (oop)
+ storePointer: Integer (index) ofObject: Integer (oop) withValue: Integer (oop) → Integer (oop)
...

InterpreterSimulation

InterpreterSimulator

- stackPointer : Integer (index)
- displayForm : Form
- transcript : TranscriptStream
...

+ storeAndPopReceiverVariableBytecode → self
+ ceCounterTripped: Integer (Boolean oop) → Integer (Boolean oop)
...

ObjectMemorySimulator

- memory : DoubleWordArray
...

+ fetchPointer: Integer (index) ofObject: Integer (oop) → Integer (oop)
+ storePointer: Integer (index) ofObject: Integer (oop) withValue: Integer (oop) → Integer (oop)
...

JIT

Cogit

- opcodeIndex : Integer (index)
...

+ cog: Integer (CompiledMethod oop) selector: Integer (Symbol oop) → CogMethod*
+ genPushReceiverVariable: Integer (index) → Integer (error code)
+ genStorePop: Boolean ReceiverVariable: Integer (index) → Integer (error code)
...

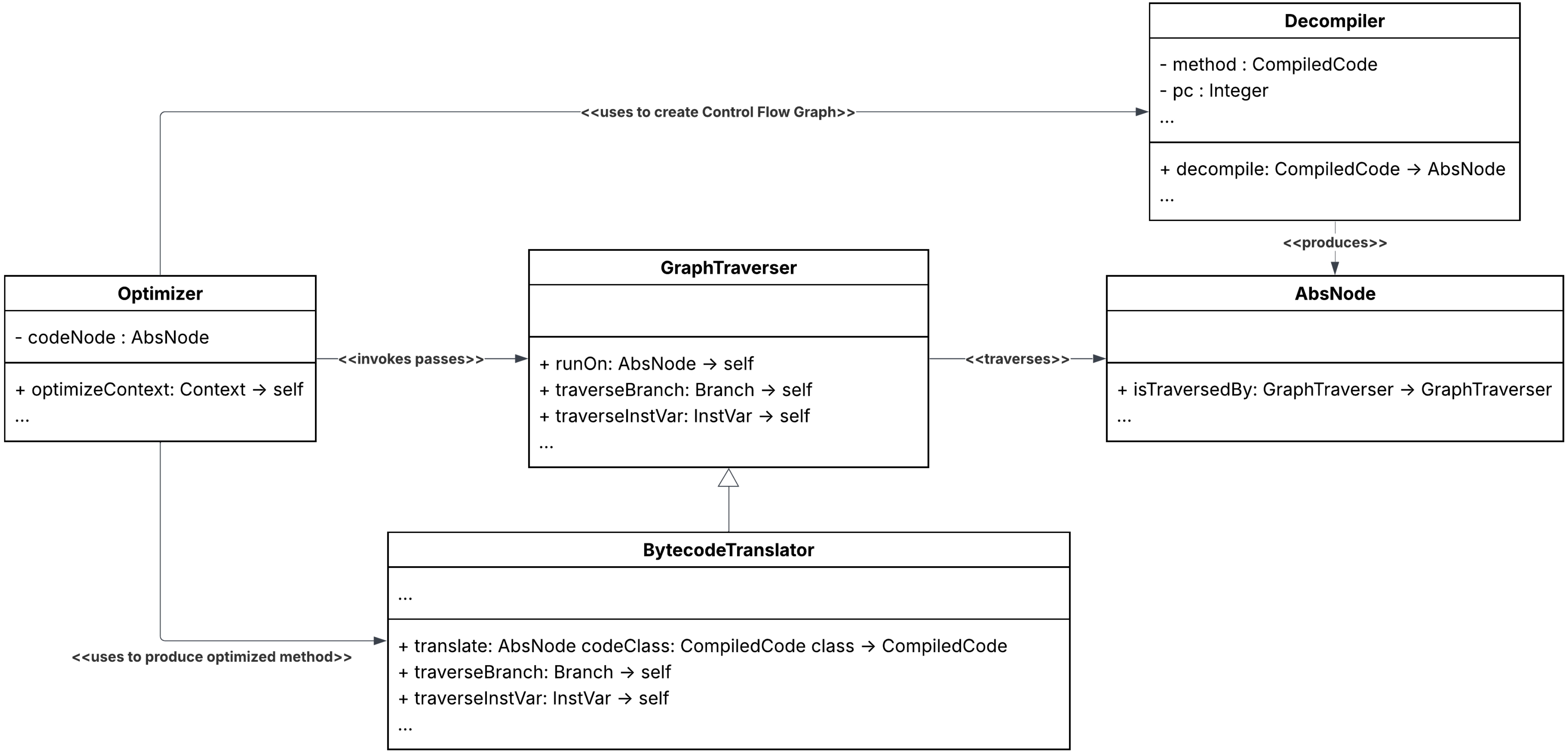
ObjectRepresentation

...

+ genLoadSlot: Integer (index) sourceReg: Integer (register) destReg: Integer (register) → Integer (error code)
+ genStoreSourceReg: Integer (register) slotIndex: Integer (index) destReg: Integer (index) ... → Integer (error code)
...

OSVM: Scorch Architecture

Simplified significantly, overview of most important classes for optimization only



Decompiler

- method : CompiledCode
- pc : Integer
...

+ decompile: CompiledCode → AbsNode
...

Optimizer

- codeNode : AbsNode

+ optimizeContext: Context → self
...

GraphTraverser

+ runOn: AbsNode → self
+ traverseBranch: Branch → self
+ traverseInstVar: InstVar → self
...

AbsNode

+ isTraversedBy: GraphTraverser → GraphTraverser
...

BytecodeTranslator

...

+ translate: AbsNode codeClass: CompiledCode class → CompiledCode
+ traverseBranch: Branch → self
+ traverseInstVar: InstVar → self
...