Visualizing Graal

Chris Seaton
2020 Graal Workshop
22 Feb 2020

Context

shopify

- Build your own online business storefront
- Small and large merchants
- Capital, shipping, payments, fulfillment

83k

average r/s

10bn

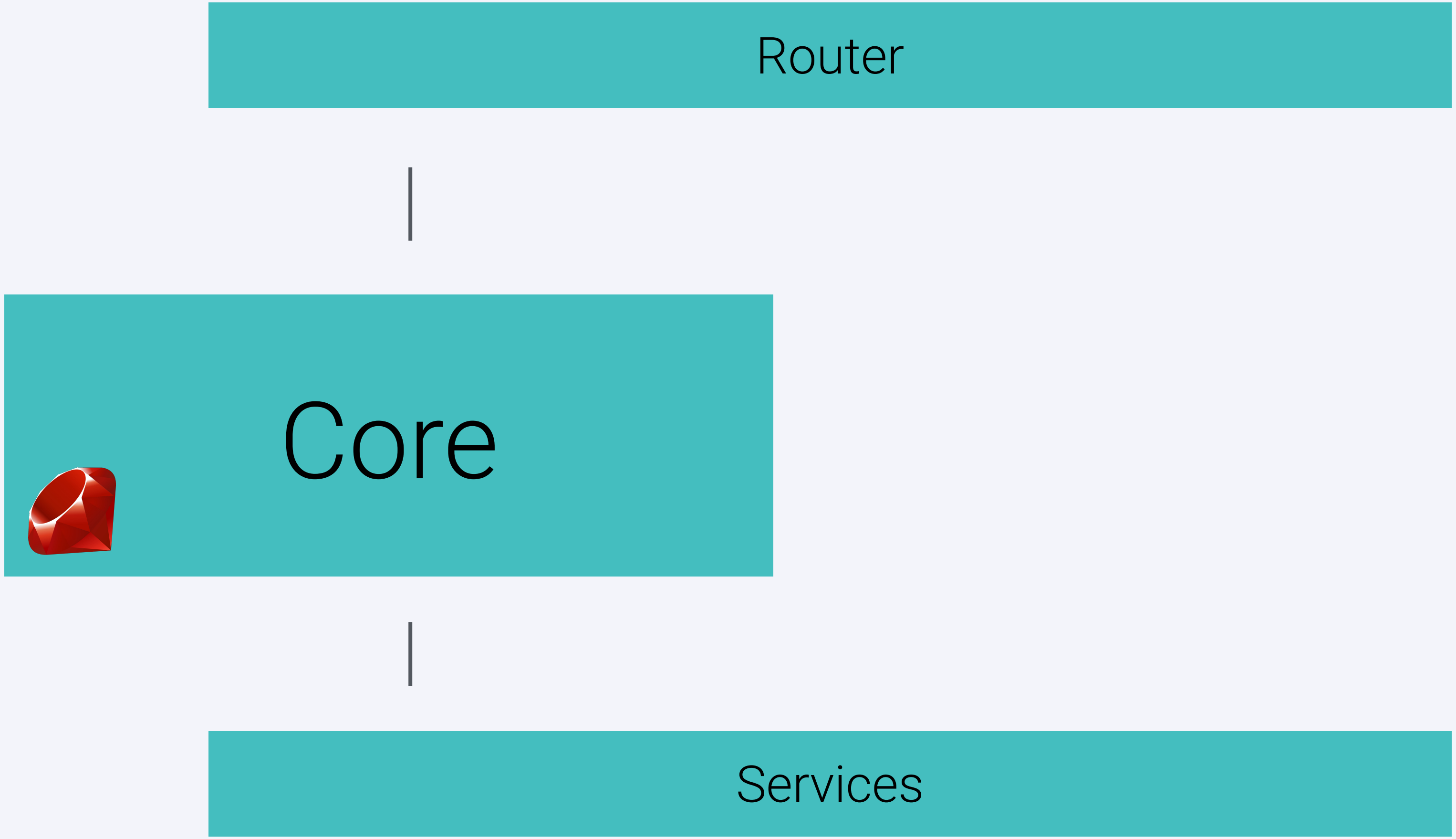
average events/day

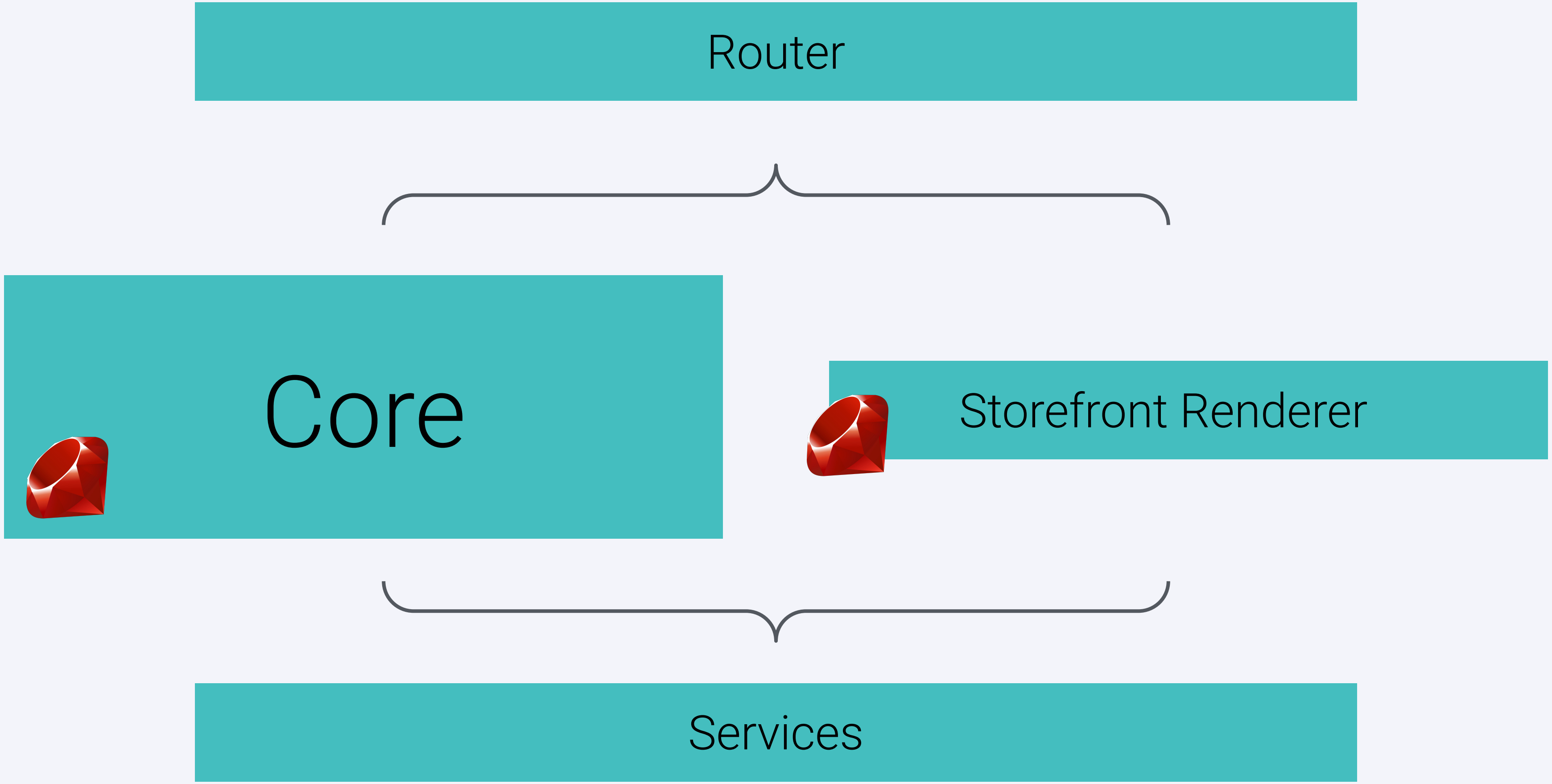
40

deploys/day

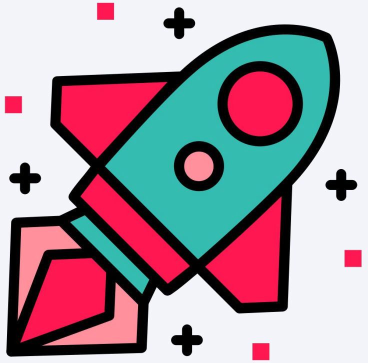
$130m

average merchant value/day


Router
Core
Services


Router
Core
Storefront Renderer
Services

TruffleRuby



TruffleRuby

- Ruby implemented in Java using the Truffle framework
- Interpreter AOT compiled to native by Graal Native Image
- Ruby JIT compiled to native by Graal PE
- Not the same as JRuby - that’s another Ruby in Java that can use Graal

Interpreter implemented in Java

```java
@Override
public Object execute(VirtualFrame frame) {
    if (conditionProfile.profile(condition.executeBoolean(frame))) {
        return thenBody.execute(frame);
    } else {
        return elseBody.execute(frame);
    }
}
```

Primitives implemented in Java

```
@CoreMethod(names = "clear", raiseIfFrozenSelf = true)
public abstract static class ClearNode extends CoreMethodArrayArgumentsNode {
    @Specialization(guards = "isNullHash(hash)")
    protected DynamicObject emptyNull(DynamicObject hash) {
        return hash;
    }
    @Specialization(guards = "!isNullHash(hash)")
    protected DynamicObject empty(DynamicObject hash) {
        Layouts.HASH.setStore(hash, null);
        Layouts.HASH.setSize(hash, 0);
        Layouts.HASH.setFirstInSequence(hash, null);
        Layouts.HASH.setLastInSequence(hash, null);
        return hash;
    }
}
```

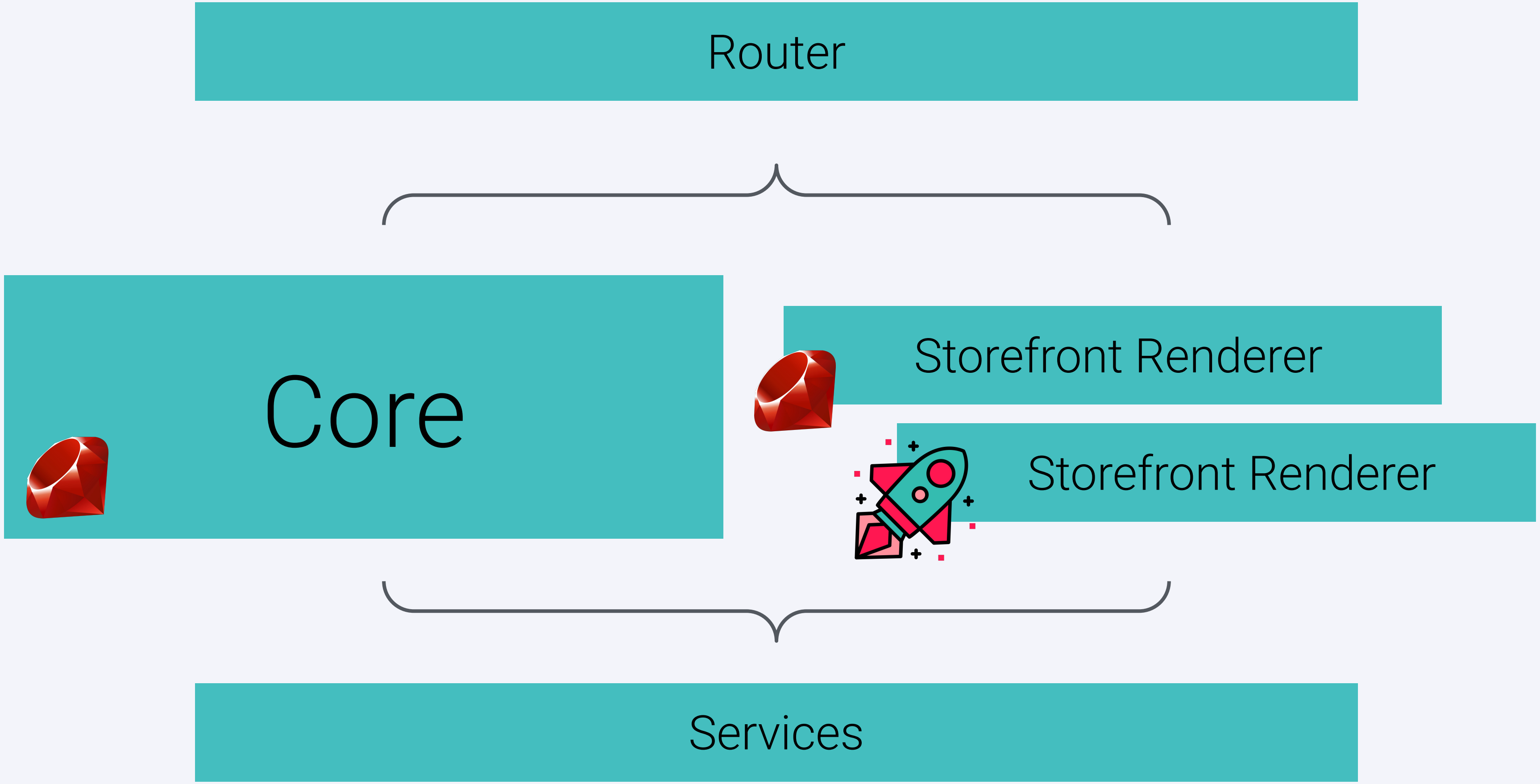
Core library re-implemented in Ruby

```ruby
def loop
  return to_enum(:loop) { Float::INFINITY } unless block_given?
  begin
    while true
      yield
    end
  rescue StopIteration => si
    si.result
  end
end
```

C extensions interpreted using Sulong

```
int rb_tr_obj_equal(VALUE first, VALUE second) {
 return RTEST(rb_funcall(first, rb_intern("equal?"), 1, second));
}
```

How Shopify is using TruffleRuby


Router
Core
Storefront Renderer
Storefront Renderer
Services

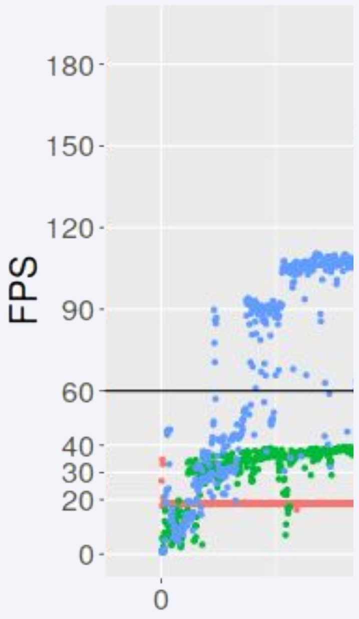
Challenges

Basic challenges

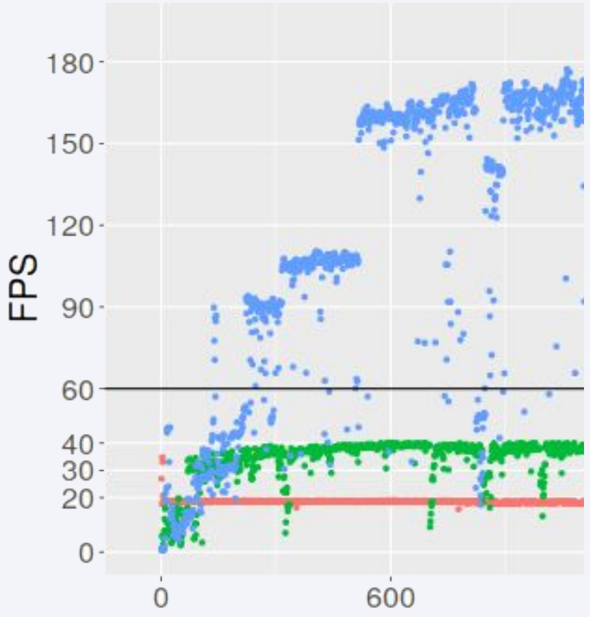
- Re-implementing a very large language with complex semantics
- Working against a developing language
- Working against a developing codebase
- Testing and preventing regressions
- Coordinating work between two companies

Understanding warmup

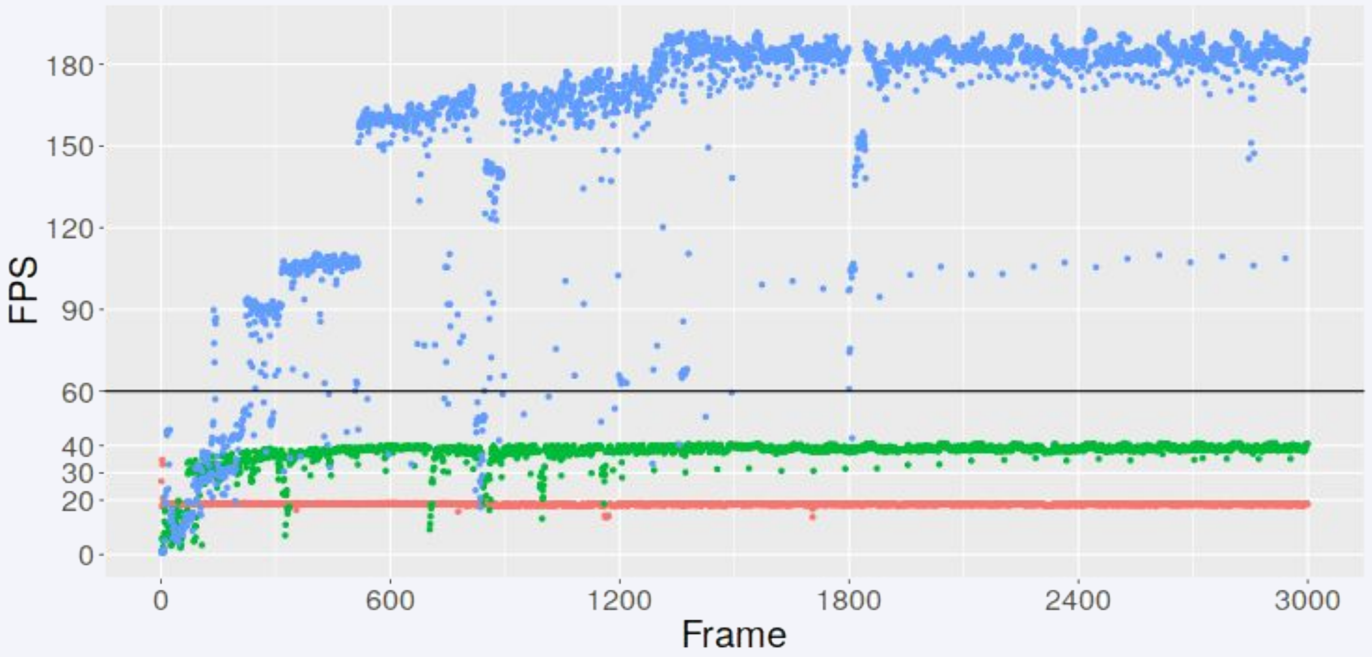
is it warm yet?



https://eregon.me/blog/2016/11/28/optcarrot.html



https://eregon.me/blog/2016/11/28/optcarrot.html



https://eregon.me/blog/2016/11/28/optcarrot.html

Understanding warmup

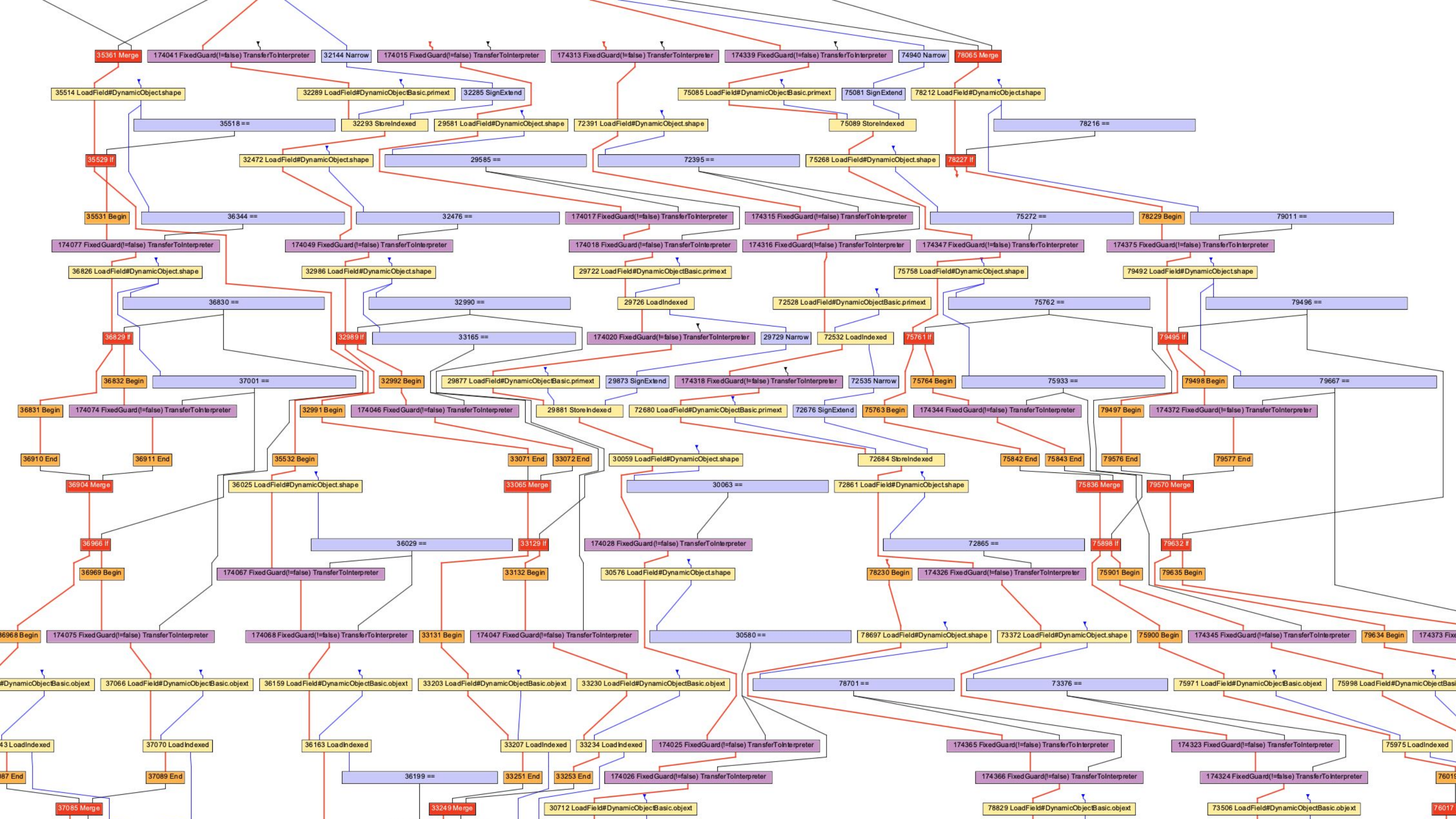
how do we get it warm in less time?

Understanding graphs

10,000+

node production graphs

after Truffle tier


35361 Merge
174041 FixedGuard(!=false) TransferToInterpreter
32144 Narrow
174015 FixedGuard(!=false) TransferToInterpreter
174313 FixedGuard(!=false) TransferToInterpreter
174339 FixedGuard(!=false) TransferToInterpreter
74940 Narrow
78065 Merge
35514 LoadField#DynamicObject.shape
32289 LoadField#DynamicObjectBasic.primext
32285 SignExtend
75085 LoadField#DynamicObjectBasic.primext
75081 SignExtend
78212 LoadField#DynamicObject.shape
35518 ==
32293 StoreIndexed
29581 LoadField#DynamicObject.shape
72391 LoadField#DynamicObject.shape
75089 StoreIndexed
78216 ==
35529 If
32472 LoadField#DynamicObject.shape
29585 ==
72395 ==
75268 LoadField#DynamicObject.shape
78227 If
35531 Begin
36344 ==
32476 ==
174017 FixedGuard(!=false) TransferToInterpreter
174315 FixedGuard(!=false) TransferToInterpreter
75272 ==
78229 Begin
79011 ==
174077 FixedGuard(!=false) TransferToInterpreter
174049 FixedGuard(!=false) TransferToInterpreter
174018 FixedGuard(!=false) TransferToInterpreter
174316 FixedGuard(!=false) TransferToInterpreter
174347 FixedGuard(!=false) TransferToInterpreter
174375 FixedGuard(!=false) TransferToInterpreter
36826 LoadField#DynamicObject.shape
32986 LoadField#DynamicObject.shape
29722 LoadField#DynamicObjectBasic.primext
75758 LoadField#DynamicObject.shape
79492 LoadField#DynamicObject.shape
36830 ==
32990 ==
29726 LoadIndexed
72528 LoadField#DynamicObjectBasic.primext
75762 ==
79496 ==
36829 If
32989 If
33165 ==
174020 FixedGuard(!=false) TransferToInterpreter
29729 Narrow
72532 LoadIndexed
75761 If
79495 If
36832 Begin
37001 ==
32992 Begin
29877 LoadField#DynamicObjectBasic.primext
29873 SignExtend
174318 FixedGuard(!=false) TransferToInterpreter
72535 Narrow
75764 Begin
75933 ==
79498 Begin
79667 ==
36831 Begin
174074 FixedGuard(!=false) TransferToInterpreter
32991 Begin
174046 FixedGuard(!=false) TransferToInterpreter
29881 StoreIndexed
72680 LoadField#DynamicObjectBasic.primext
72676 SignExtend
75763 Begin
174344 FixedGuard(!=false) TransferToInterpreter
79497 Begin
174372 FixedGuard(!=false) TransferToInterpreter
36910 End
36911 End
35532 Begin
33071 End
33072 End
30059 LoadField#DynamicObject.shape
72684 StoreIndexed
75842 End
75843 End
79576 End
79577 End
36904 Merge
36025 LoadField#DynamicObject.shape
33065 Merge
30063 ==
72861 LoadField#DynamicObject.shape
75836 Merge
79570 Merge
36966 If
36029 ==
33129 If
174028 FixedGuard(!=false) TransferToInterpreter
72865 ==
75898 If
79632 If
36969 Begin
174067 FixedGuard(!=false) TransferToInterpreter
33132 Begin
30576 LoadField#DynamicObject.shape
78230 Begin
174326 FixedGuard(!=false) TransferToInterpreter
75901 Begin
79635 Begin
6968 Begin
174075 FixedGuard(!=false) TransferToInterpreter
174068 FixedGuard(!=false) TransferToInterpreter
33131 Begin
174047 FixedGuard(!=false) TransferToInterpreter
30580 ==
78697 LoadField#DynamicObject.shape
73372 LoadField#DynamicObject.shape
75900 Begin
174345 FixedGuard(!=false) TransferToInterpreter
79634 Begin
174373 Fix
#DynamicObjectBasic.objext
37066 LoadField#DynamicObjectBasic.objext
36159 LoadField#DynamicObjectBasic.objext
33203 LoadField#DynamicObjectBasic.objext
33230 LoadField#DynamicObjectBasic.objext
78701 ==
73376 ==
75971 LoadField#DynamicObjectBasic.objext
75998 LoadField#DynamicObjectBasi
43 LoadIndexed
37070 LoadIndexed
36163 LoadIndexed
33207 LoadIndexed
33234 LoadIndexed
174025 FixedGuard(!=false) TransferToInterpreter
174365 FixedGuard(!=false) TransferToInterpreter
174323 FixedGuard(!=false) TransferToInterpreter
75975 LoadIndexed
087 End
37089 End
36199 ==
33251 End
33253 End
174026 FixedGuard(!=false) TransferToInterpreter
174366 FixedGuard(!=false) TransferToInterpreter
174324 FixedGuard(!=false) TransferToInterpreter
7601
37085 Merge
33249 Merge
30712 LoadField#DynamicObjectBasic.objext
78829 LoadField#DynamicObjectBasic.objext
73506 LoadField#DynamicObjectBasic.objext
76017

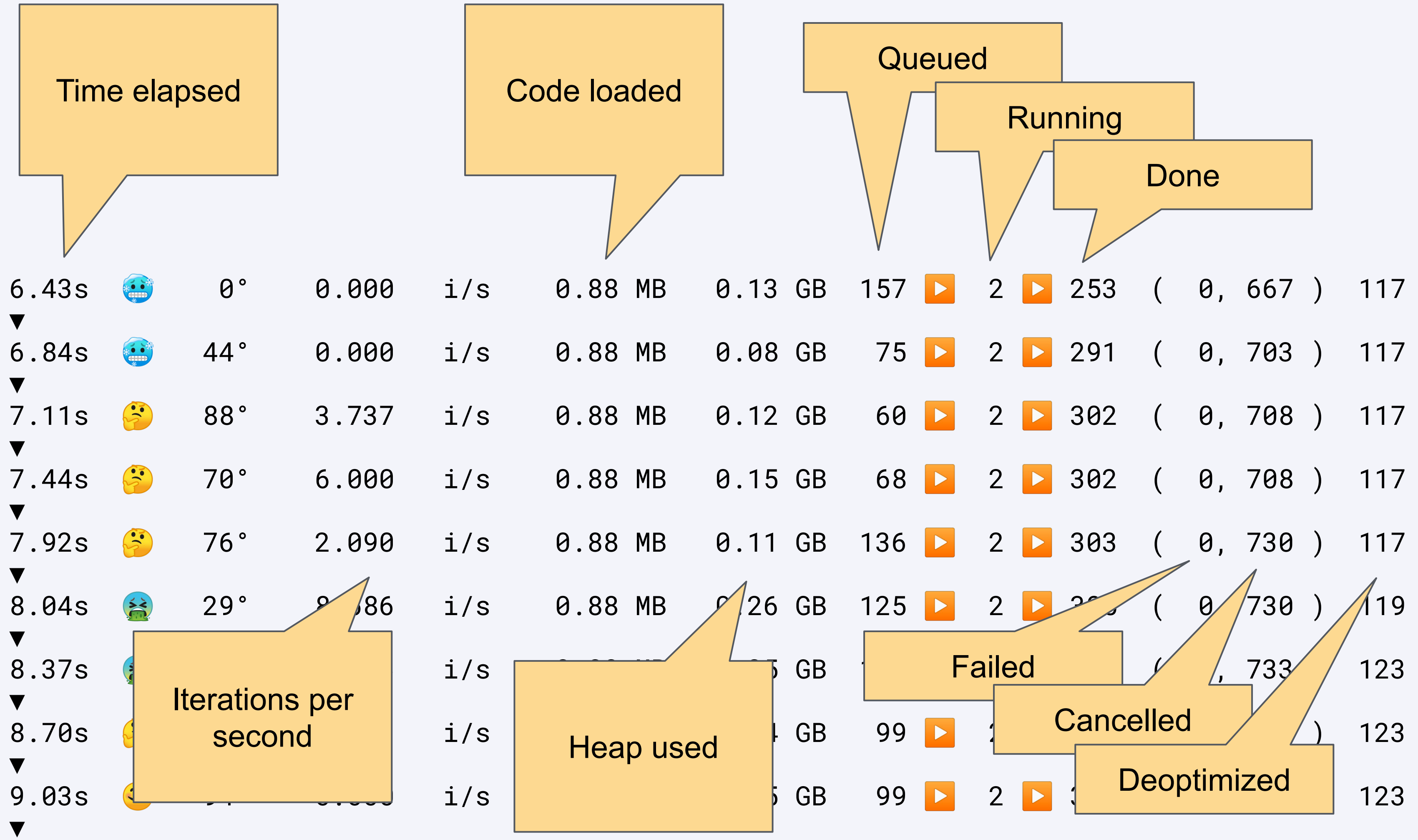
Taking Graal’s temperature

Goals

- Understand if TruffleRuby is warm yet
- Understand why it isn’t warm
- Understand how much longer it may take to become warm
- Be easier to use than watching a wall of text logs

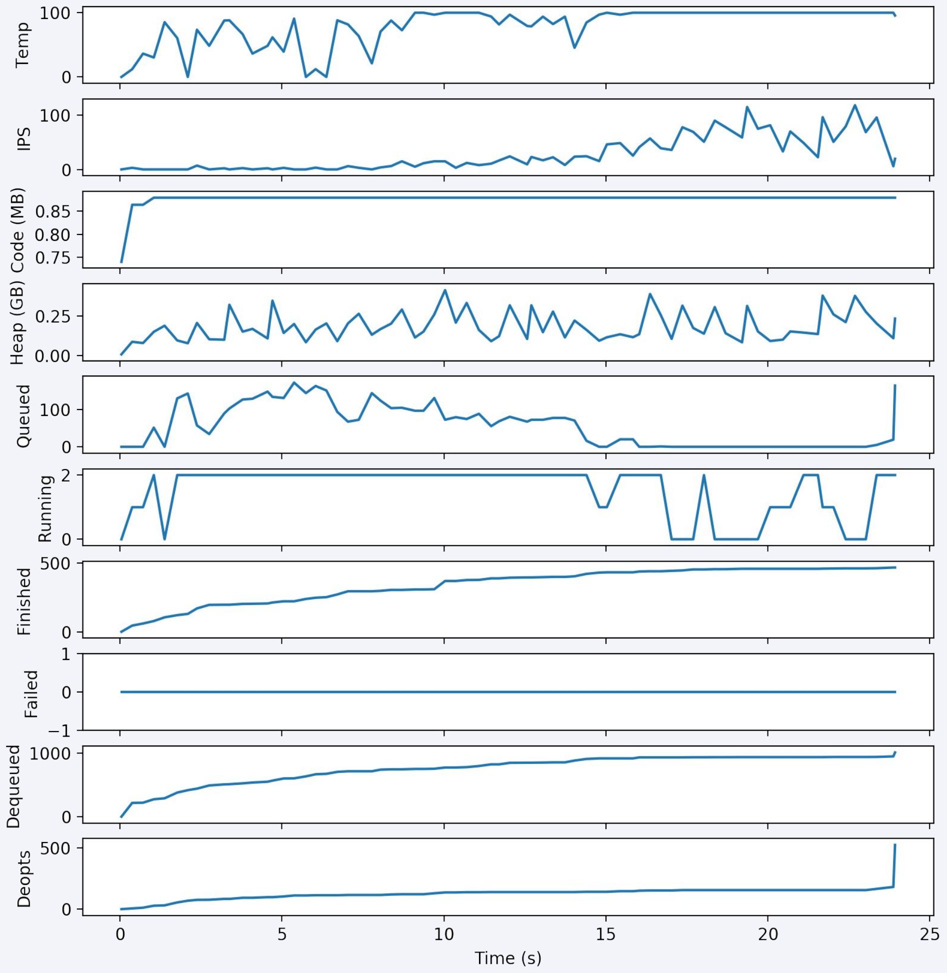
```
% ruby --thermometer --thermometer.IterationPoint=lib/optcarrot/nes.rb:42 \
      -v -r ./tools/shim bin/optcarrot --benchmark -f 100000 \
      examples/Lan_Master.nes
```

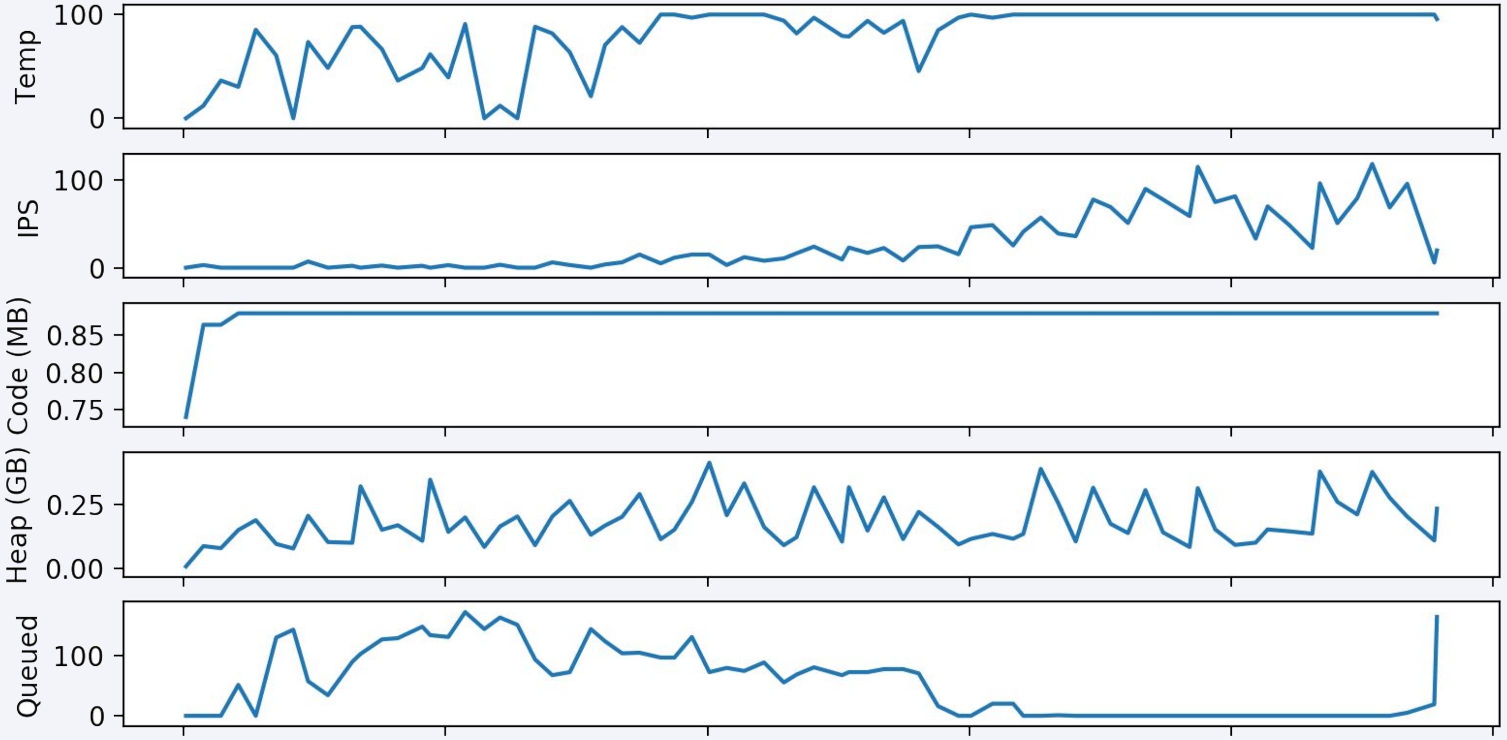
```
6.43s 🥶   0°  0.000  i/s  0.88 MB  0.13 GB  157 ▶️  2 ▶️ 253  (  0, 667 )  117
▼
6.84s 🥶  44°  0.000  i/s  0.88 MB  0.08 GB   75 ▶️  2 ▶️ 291  (  0, 703 )  117
▼
7.11s 🤔  88°  3.737  i/s  0.88 MB  0.12 GB   60 ▶️  2 ▶️ 302  (  0, 708 )  117
▼
7.44s 🤔  70°  6.000  i/s  0.88 MB  0.15 GB   68 ▶️  2 ▶️ 302  (  0, 708 )  117
▼
7.92s 🤔  76°  2.090  i/s  0.88 MB  0.11 GB  136 ▶️  2 ▶️ 303  (  0, 730 )  117
▼
8.04s 🤮  29°  8.686  i/s  0.88 MB  0.26 GB  125 ▶️  2 ▶️ 305  (  0, 730 )  119
▼
8.37s 🤮  91°  5.996  i/s  0.88 MB  0.35 GB  100 ▶️  2 ▶️ 311  (  0, 733 )  123
▼
8.70s 🤔  88°  8.991  i/s  0.88 MB  0.44 GB   99 ▶️  2 ▶️ 313  (  0, 736 )  123
▼
9.03s 😊  94°  0.000  i/s  0.88 MB  0.15 GB   99 ▶️  2 ▶️ 313  (  0, 736 )  123
▼
```

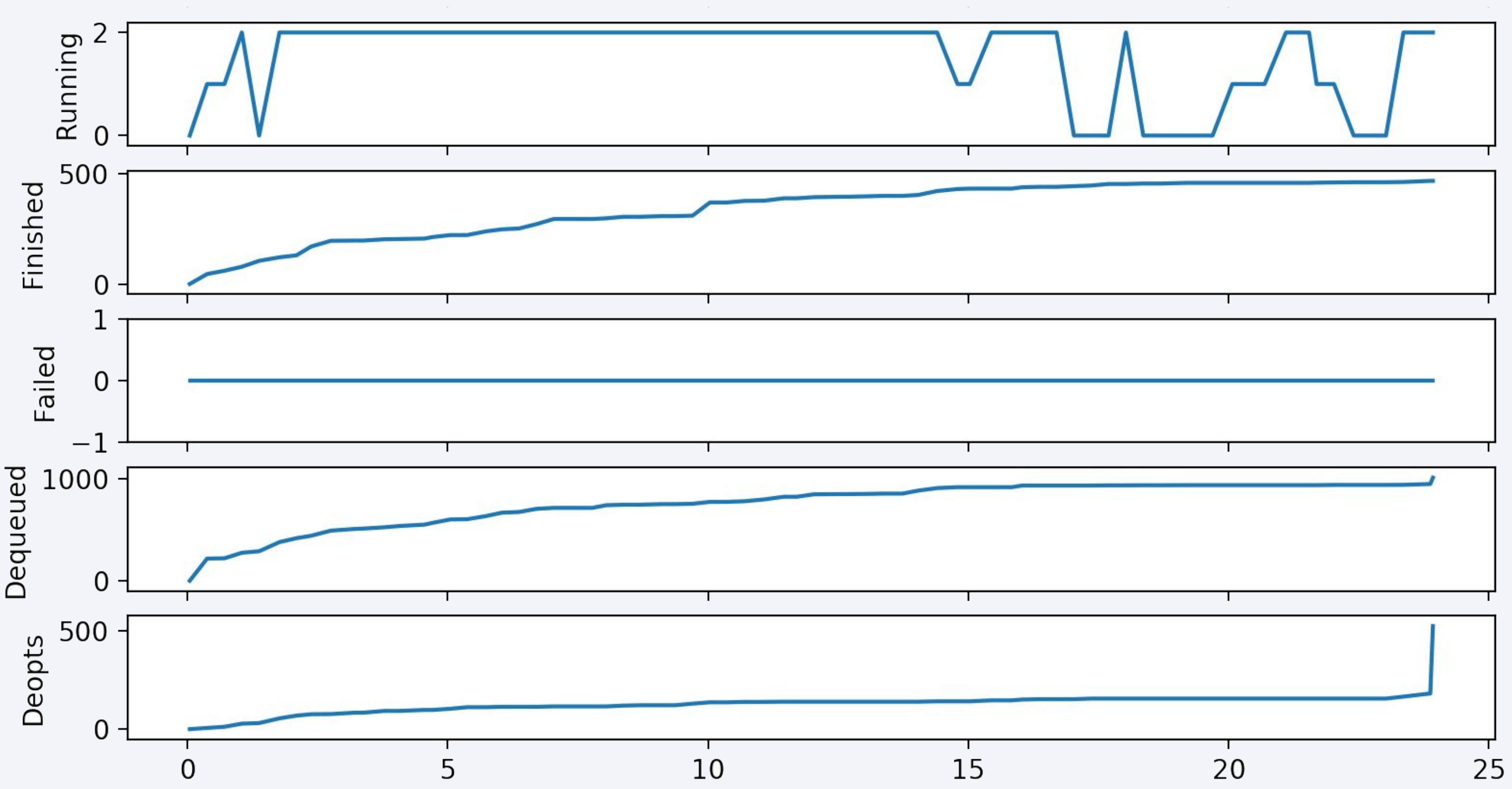


```
6.43s 🥶   0°   0.000  i/s  0.88 MB  0.13 GB  157 ▶️  2 ▶️ 253  (  0, 667 )  117
▼
6.84s 🥶  44°   0.000  i/s  0.88 MB  0.08 GB   75 ▶️  2 ▶️ 291  (  0, 703 )  117
▼
7.11s 🤔  88°   3.737  i/s  0.88 MB  0.12 GB   60 ▶️  2 ▶️ 302  (  0, 708 )  117
▼
7.44s 🤔  70°   6.000  i/s  0.88 MB  0.15 GB   68 ▶️  2 ▶️ 302  (  0, 708 )  117
▼
7.92s 🤔  76°   2.090  i/s  0.88 MB  0.11 GB  136 ▶️  2 ▶️ 303  (  0, 730 )  117
▼
8.04s 🤮  29°   8  86  i/s  0.88 MB  0 26 GB  125 ▶️  2 ▶️      (  0  730 )  119
▼
8.37s                i/s                5 GB                   ,  733        123
▼
8.70s                i/s                  GB   99 ▶️                    )  123
▼
9.03s                i/s                5 GB   99 ▶️  2 ▶️              123
▼
```

🥶	0°
🥶	44°
🤔	88°
🤔	70°
🤔	76°
🤮	29°
🤮	91°
🤔	88°
😊	94°


Temp
IPS
Code (MB)
Heap (GB)
Queued
Running
Finished
Failed
Dequeued
Deopts
Time (s)
0
5
10
15
20
25


Temp
100
0
IPS
100
0
Code (MB)
0.85
0.80
0.75
Heap (GB)
0.25
0.00
Queued
100
0



Running
Finished
Failed
Dequeued
Deopts

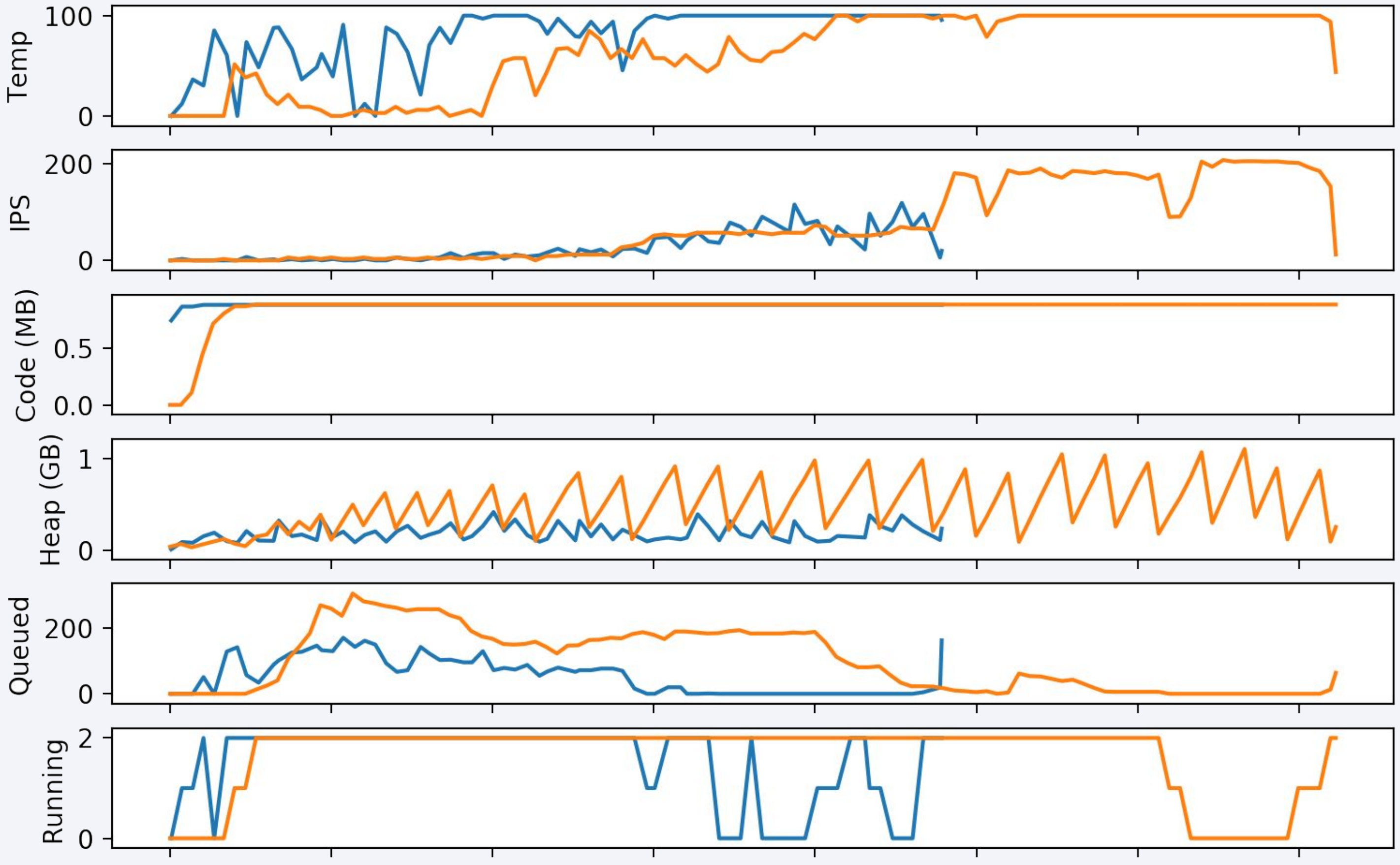
Taking a long time as we have a long queue!

Seems like a lot of cancellations

```
6.43s  🥶   0°   0.000  i/s   0.88 MB   0.13 GB  157 ▶️  2 ▶️ 253  (  0, 667 )  117
▼
6.84s  🥶  44°   0.000  i/s   0.88 MB   0.08 GB   75 ▶️  2 ▶️ 291  (  0, 703 )  117
▼
7.11s  🤔  88°   3.737  i/s   0.88 MB   0.12 GB   60 ▶️  2 ▶️ 302  (  0, 708 )  117
▼
7.44s  🤔  70°   6.000  i/s   0.88 MB   0.15 GB   68 ▶️  2 ▶️ 302  (  0, 708 )  117
▼
7.92s  🤔  76°   2.090  i/s   0.88 MB   0.11 GB  136 ▶️  2 ▶️ 303  (  0, 730 )  117
▼
8.04s  🤮  29°   8.686  i/s   0.88 MB   0.26 GB  125 ▶️  2     305  (  0, 730 )   19
▼
8.37s  🤮  91°   5.996  i/s   0.88 MB   0.35 GB               11  (
▼
8.70s  🤔  88°   8.991  i/s   0.88 MB   0.44 GB               13  (
▼
9.03s  😊  94°   0.000  i/s   0.88 MB   0.15 GB               13  (
▼
```

Seems like a lot of compilations

Are we speculating too much?


Temp
100
0
IPS
200
0
Code (MB)
0.5
0.0
Heap (GB)
1
0
Queued
200
0
Running
2
0

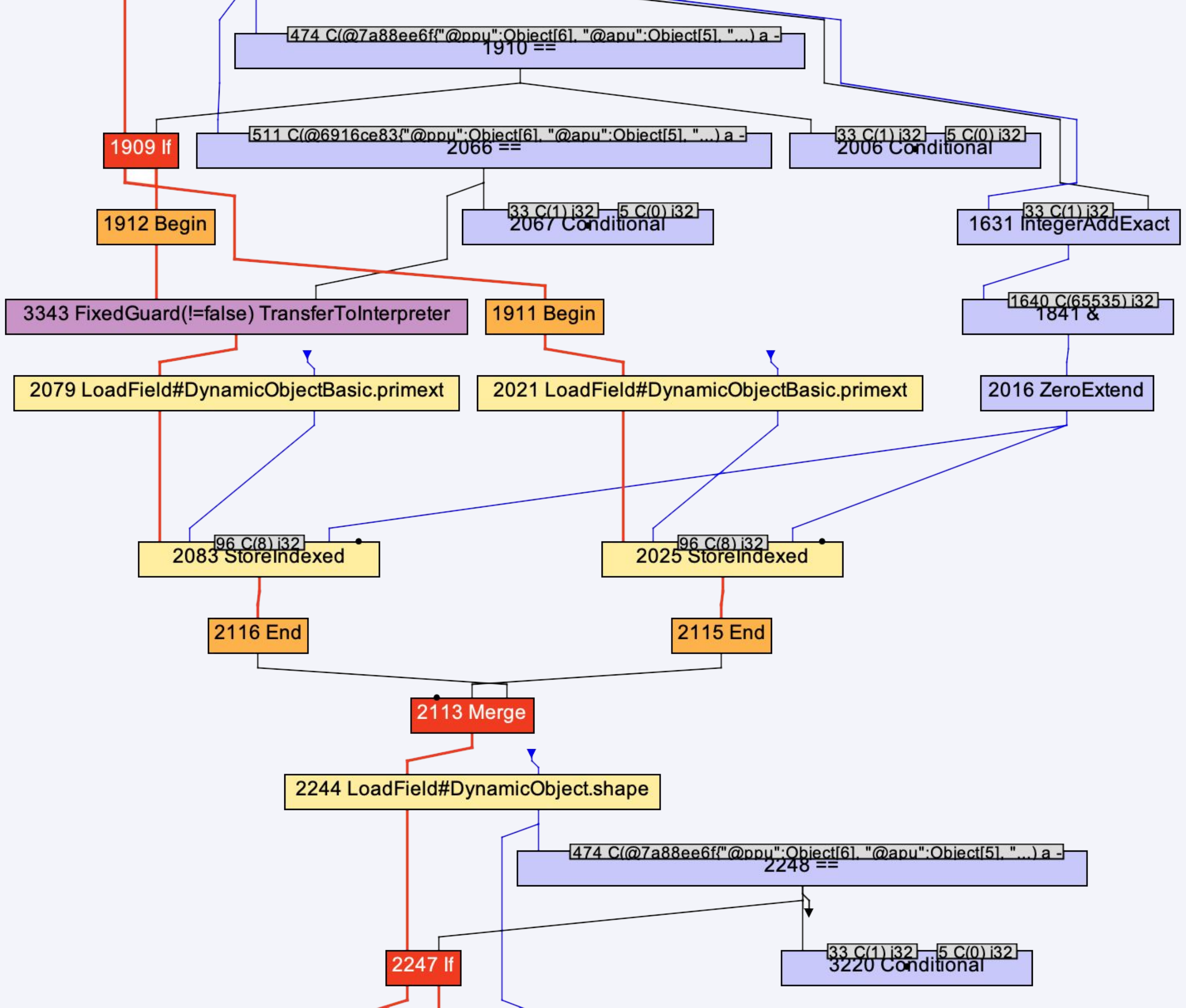
Cool things here

- Relationship between temperature and IPS
- Create a benchmark just by modifying the command line
- Could have this data in VisualVM?

Swimming in the sea

```ruby
def _rts
  @_pc = (pull16() + 1) & 0xffff
  @clk += CLK_6
end
```

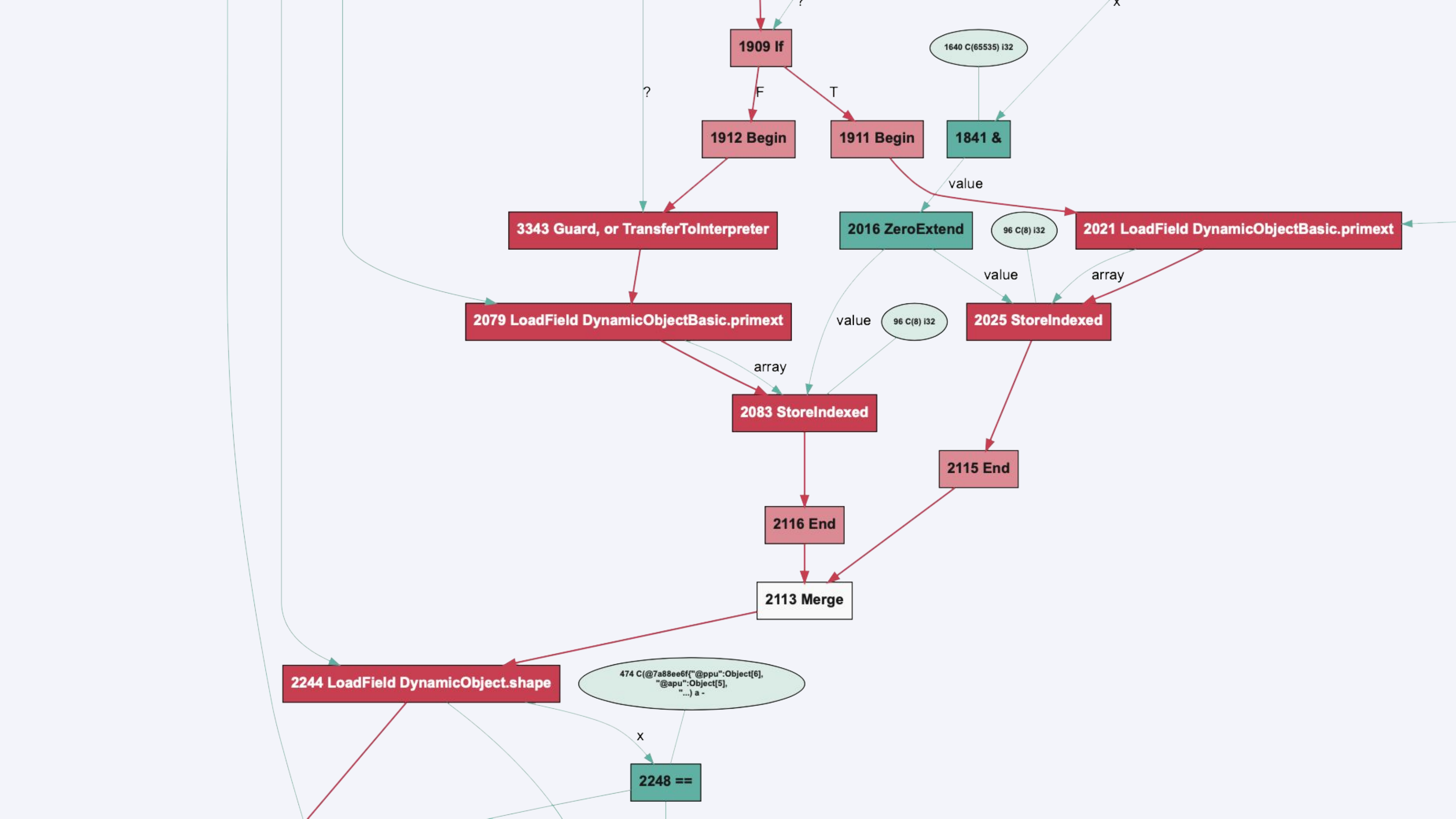
```
% ruby --experimental-options \
       --engine.Splitting=false --engine.Inlining=false \
       --vm.Dgraal.Dump=Truffle:2 \
       -v -r ./tools/shim bin/optcarrot --benchmark -f 100000 \
       examples/Lan_Master.nes
```


474 C(@7a88ee6f{"@ppu":Object[6], "@apu":Object[5], "...) a -
1910 ==
1909 If
511 C(@6916ce83{"@ppu":Object[6], "@apu":Object[5], "...) a -
2066 ==
33 C(1) i32
5 C(0) i32
2006 Conditional
1912 Begin
33 C(1) i32
5 C(0) i32
2067 Conditional
33 C(1) i32
1631 IntegerAddExact
3343 FixedGuard(!=false) TransferToInterpreter
1911 Begin
1640 C(65535) i32
1841 &
2079 LoadField#DynamicObjectBasic.primext
2021 LoadField#DynamicObjectBasic.primext
2016 ZeroExtend
96 C(8) i32
2083 StoreIndexed
96 C(8) i32
2025 StoreIndexed
2116 End
2115 End
2113 Merge
2244 LoadField#DynamicObject.shape
474 C(@7a88ee6f{"@ppu":Object[6], "@apu":Object[5], "...) a -
2248 ==
2247 If
33 C(1) i32
5 C(0) i32
3220 Conditional

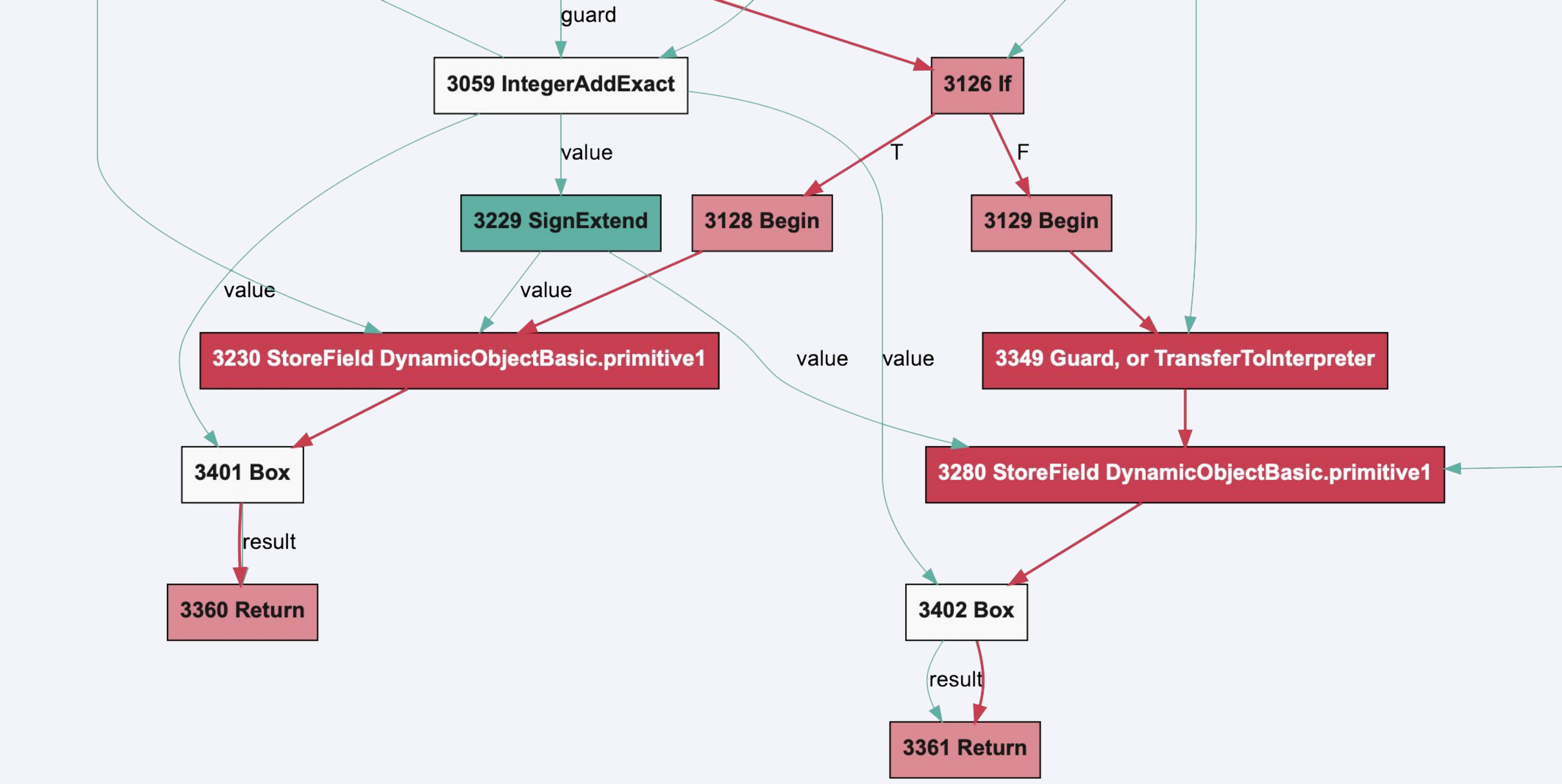
```
% seafoam rts.bgv list
rts.bgv:0  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/AST/After Profiling
rts.bgv:1  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Call Tree/After Profiling
rts.bgv:2  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After phase com.oracle.sv
rts.bgv:3  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After phase com.oracle.sv
rts.bgv:4  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After Partial Evaluation
rts.bgv:5  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After phase org.graalvm.c
rts.bgv:6  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After phase org.graalvm.c
rts.bgv:7  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After phase org.graalvm.c
rts.bgv:8  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After phase org.graalvm.c
rts.bgv:9  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/After TruffleTier
rts.bgv:10  Truffle::Optcarrot::CPU#_rts optcarrot/lib/optcarrot/cpu.rb:506/Graal Graphs/initial state
```

```
% seafoam rts.bgv:9:2079 props
{
  "uncheckedStamp": null,
  "relativeFrequency": 0.499995,
  "nodeCostSize": "SIZE_1",
  "stamp": "a# [J",
  "nodeToBlock": "B24",
  "nodeSourcePosition": {
    "method": {
      "declaring_class": "org.graalvm.compiler.truffle.runtime.OptimizedCallTarget",
      "method_name": "callRoot",
      "signature": {
        "args": [
          "[Ljava/lang/Object;"
        ],
        "ret": "Ljava/lang/Object;"
      },
      "modifiers": 20
    },
    "bci": -6,
...
```

```
% seafoam rts.bgv:9 render
```

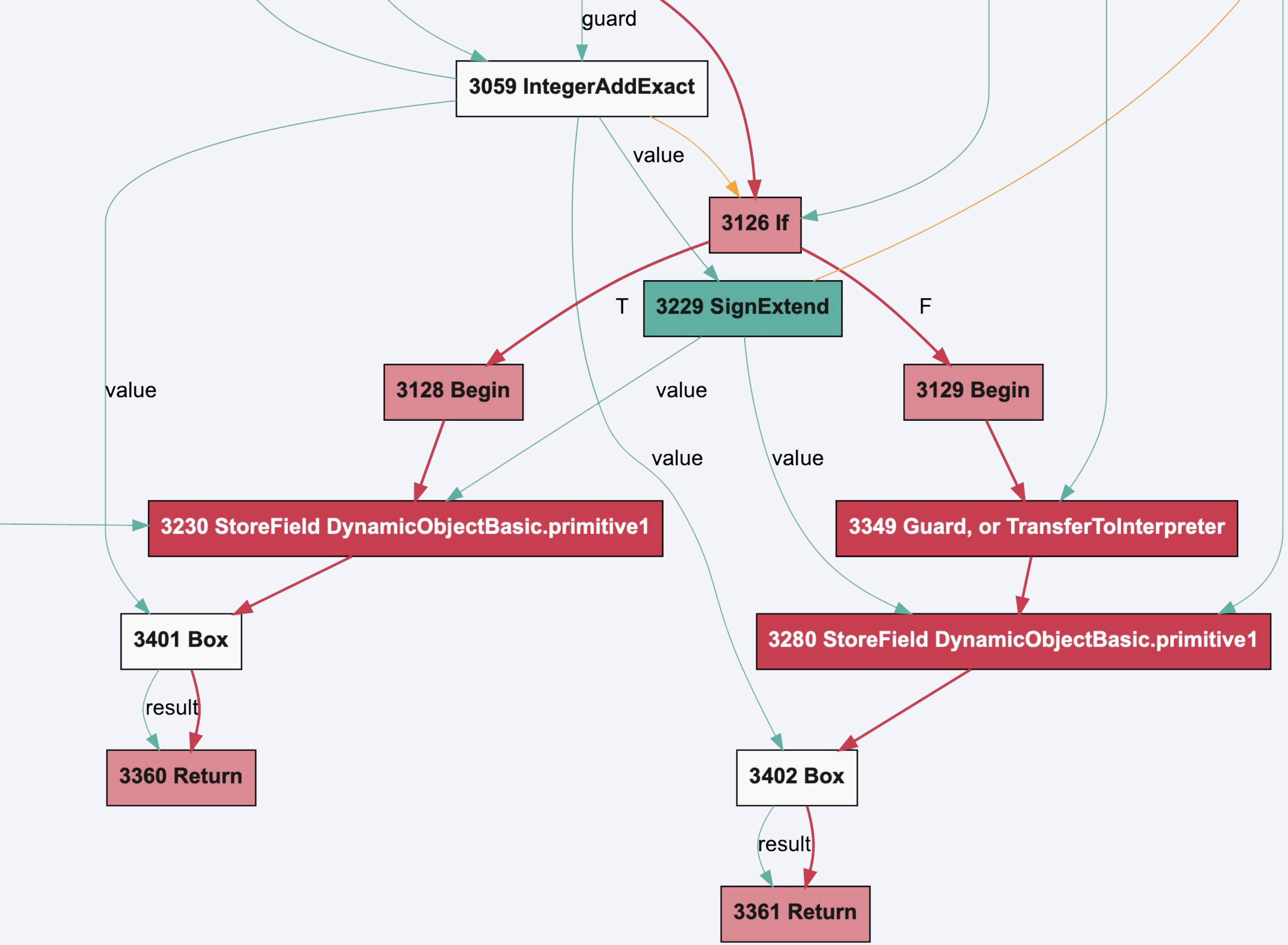

?
x
1909 If
1640 C(65535) i32
?
F
T
1912 Begin
1911 Begin
1841 &
value
3343 Guard, or TransferToInterpreter
2016 ZeroExtend
96 C(8) i32
2021 LoadField DynamicObjectBasic.primext
value
array
2079 LoadField DynamicObjectBasic.primext
value
96 C(8) i32
2025 StoreIndexed
array
2083 StoreIndexed
2115 End
2116 End
2113 Merge
2244 LoadField DynamicObject.shape
474 C(@7a88ee6f{"@ppu":Object[6], "@apu":Object[5], "...) a -
x
2248 ==

Decompiling Graal

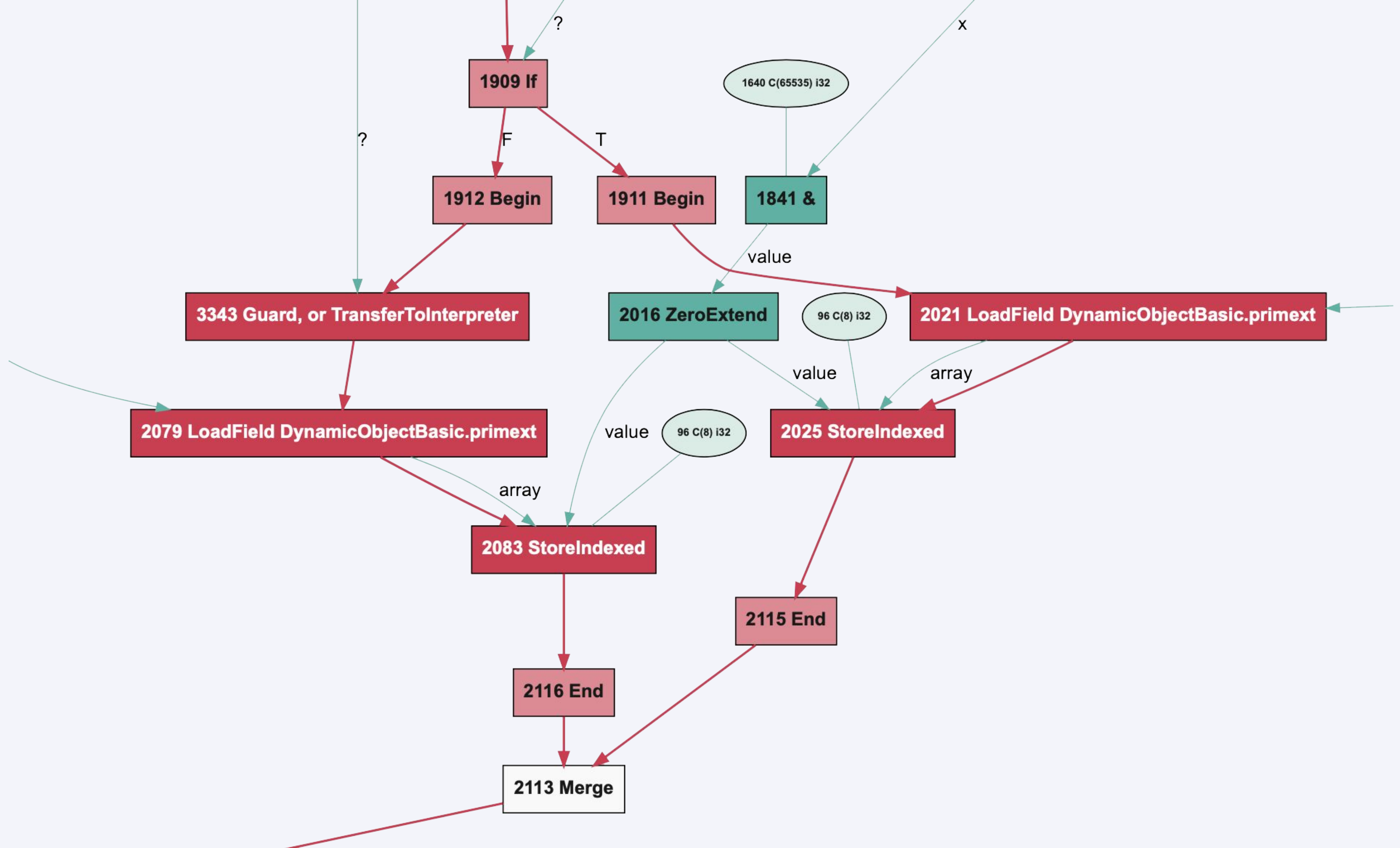

guard
3059 IntegerAddExact
3126 If
value
T
F
3229 SignExtend
3128 Begin
3129 Begin
value
value
3230 StoreField DynamicObjectBasic.primitive1
value
value
3349 Guard, or TransferToInterpreter
3401 Box
3280 StoreField DynamicObjectBasic.primitive1
result
3360 Return
3402 Box
result
3361 Return

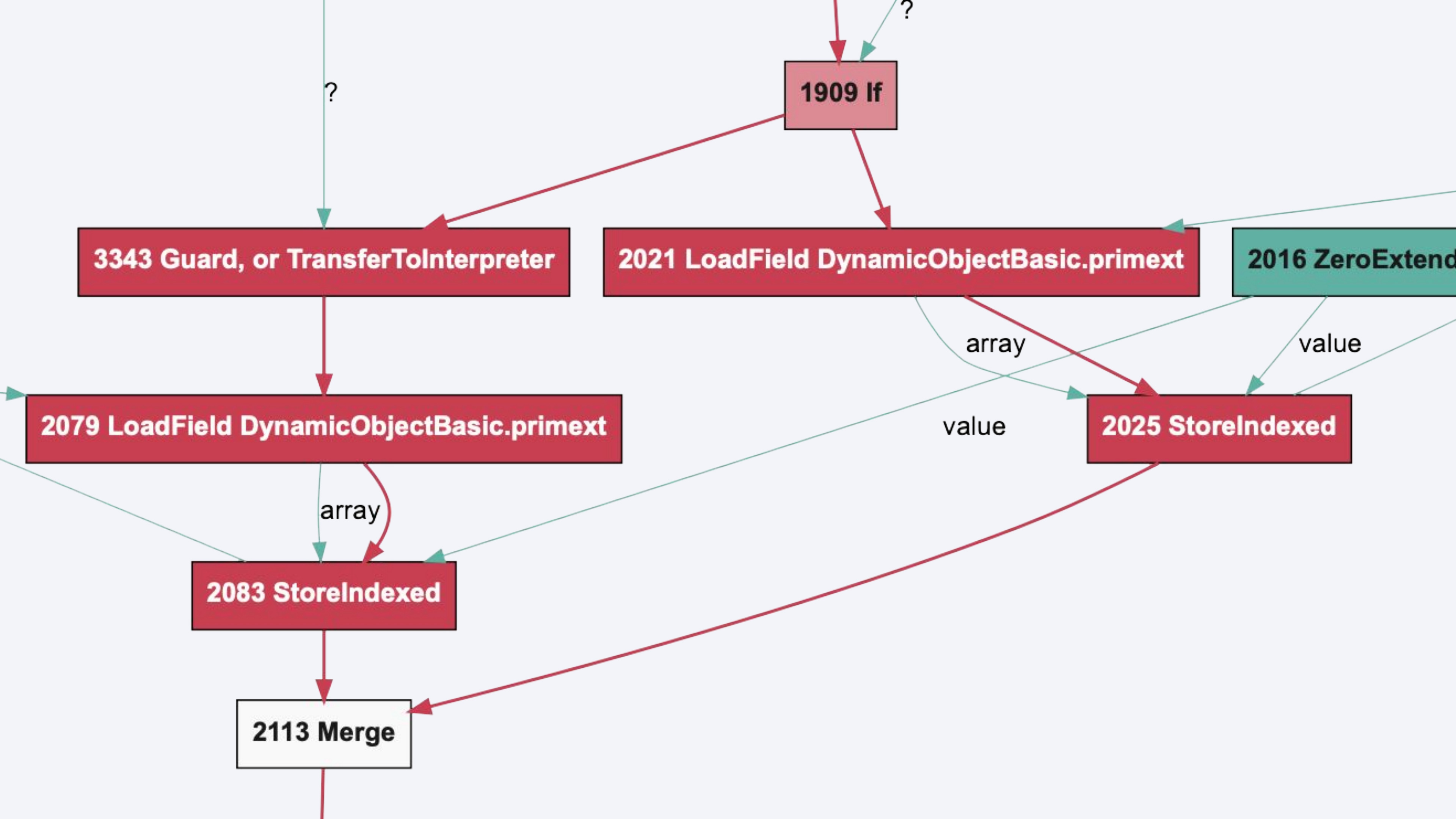
```
% seafoam rts.bgv:9 decompile
...
                v3057 = IntegerAddExactOverflow(v3396, C(72) i32)
                v3056 = Guard, or ArithmeticException(v3057)
                v3059 = IntegerAddExact(v3396, C(72) i32, v3056)
                if v2248 # node 3126
                  Begin() # node 3128
                  StoreField DynamicObjectBasic.primitive1(v3351, v3229, v3238) # node 3230
                  v3401 = Box(v3059)
                  return v3401 # node 3360
                else
                  Begin() # node 3129
                  Guard, or TransferToInterpreter(v3270) # node 3349
                  StoreField DynamicObjectBasic.primitive1(v3351, v3229, v3288) # node 3280
                  v3402 = Box(v3059)
                  return v3402 # node 3361
                end
              end
            end
...
```

```
% seafoam rts.bgv:9 render --schedule
```


guard
3059 IntegerAddExact
value
3126 If
T
3229 SignExtend
F
value
3128 Begin
value
3129 Begin
value
value
3230 StoreField DynamicObjectBasic.primitive1
3349 Guard, or TransferToInterpreter
3401 Box
3280 StoreField DynamicObjectBasic.primitive1
result
3360 Return
3402 Box
result
3361 Return

A more ideal ideal


?
x
1909 If
1640 C(65535) i32
?
F
T
1912 Begin
1911 Begin
1841 &
value
3343 Guard, or TransferToInterpreter
2016 ZeroExtend
96 C(8) i32
2021 LoadField DynamicObjectBasic.primext
value
array
2079 LoadField DynamicObjectBasic.primext
value
96 C(8) i32
2025 StoreIndexed
array
2083 StoreIndexed
2115 End
2116 End
2113 Merge


?
?
1909 If
3343 Guard, or TransferToInterpreter
2021 LoadField DynamicObjectBasic.primext
2016 ZeroExtend
array
value
2079 LoadField DynamicObjectBasic.primext
value
2025 StoreIndexed
array
2083 StoreIndexed
2113 Merge

Some disassembly required

Goals

- We want to use disassembly because we think it can be simpler than reading the graph, can be easier to see calls
- A disassembler we can ship
- A disassembler that doesn’t require a GUI tool
- A disassembler that works in native mode
- A disassembler we can add new annotations to

```
% ruby --experimental-options \
       --engine.Splitting=false --engine.Inlining=false \
       --engine.PrintDisassembly --engine.DisassembleOnly=_rts \
       -v -r ./tools/shim bin/optcarrot --benchmark -f 100000 \
       examples/Lan_Master.nes
```

```
[truffle] disassembly of Optcarrot::CPU#_rts
/Users/chrisseaton/src/github.com/mame/optcarrot/lib/optcarrot/cpu.rb:506 <opt> @ 0x1224d2000 for 2259 bytes
1224d2000:  subq  $72, %rsp
1224d2004:  movq  %rsi, 48(%rsp)
1224d2009:  movl  8(%rsi), %edi
1224d200c:  cmpl  $8, %edi
1224d200f:  jae   2089 <1224d283e>
1224d2015:  cmpl  $7, %edi
1224d2018:  jb    2208 <1224d28be>
1224d201e:  movq  56(%rsi), %rcx
1224d2022:  testq %rcx, %rcx
1224d2025:  je    2168 <1224d28a3>
1224d202b:  movq  $-8, %rdi
1224d2032:  andq  (%r14,%rcx), %rdi
1224d2036:  cmpl  $7145, 208(%r14,%rdi)
1224d2042:  jne   2148 <1224d28ac>
1224d2048:  movq  64(%rsi), %rdi
1224d204c:  testq %rdi, %rdi
1224d204f:  je    738 <1224d2337>
1224d2055:  movq  $-8, %rax
1224d205c:  andq  (%r14,%rdi), %rax
1224d2060:  movl  208(%r14,%rax), %eax
1224d2068:  cmpl  $7145, %eax
1224d206e:  sete  %al
```

```
1224d21ea:     movq   %r13, %rsi
1224d21ed:     movq   %rbp, %rax
1224d21f0:     callq  *%rax
                               ; com.oracle.svm.truffle.api.SubstrateOptimizedCallTarget.doInvoke(SubstrateOptimizedCallTarget.java:155)
                               ; org.graalvm.compiler.truffle.runtime.OptimizedCallTarget.callDirect(OptimizedCallTarget.java:349)
                               ; org.graalvm.compiler.truffle.runtime.OptimizedDirectCallNode.call(OptimizedDirectCallNode.java:67)
                               ; org.truffleruby.language.dispatch.CachedDispatchNode.call(CachedDispatchNode.java:130)
                               ; org.truffleruby.language.dispatch.CachedBoxedDispatchNode.executeDispatch(CachedBoxedDispatchNode.java:96)
                               ; org.truffleruby.language.dispatch.CachedBoxedDispatchNode.executeDispatch(CachedBoxedDispatchNode.java:86)
                               ; org.truffleruby.language.dispatch.DispatchHeadNode.dispatch(DispatchHeadNode.java:44)
                               ; org.truffleruby.language.dispatch.RubyCallNode.executeWithArgumentsEvaluated(RubyCallNode.java:117)
                               ; org.truffleruby.language.dispatch.RubyCallNode.execute(RubyCallNode.java:105)
                               ; org.truffleruby.core.inlined.InlinedAddNodeGen.execute(InlinedAddNodeGen.java:45)
                               ; org.truffleruby.core.inlined.InlinedBitAndNodeGen.execute(InlinedBitAndNodeGen.java:46)
                               ; org.truffleruby.language.objects.WriteInstanceVariableNode.execute(WriteInstanceVariableNode.java:41)
                               ; org.truffleruby.language.RubyNode.doExecuteVoid(RubyNode.java:60)
                               ; org.truffleruby.language.control.SequenceNode.execute(SequenceNode.java:33)
                               ; org.truffleruby.language.arguments.CheckArityNode.execute(CheckArityNode.java:41)
                               ; org.truffleruby.language.methods.CatchForMethodNode.execute(CatchForMethodNode.java:42)
                               ; org.truffleruby.language.methods.ExceptionTranslatingNode.execute(ExceptionTranslatingNode.java:33)
                               ; org.truffleruby.language.RubyRootNode.execute(RubyRootNode.java:61)
                               ; org.graalvm.compiler.truffle.runtime.OptimizedCallTarget.callProxy(OptimizedCallTarget.java:474)
                               ; org.graalvm.compiler.truffle.runtime.OptimizedCallTarget.callRoot(OptimizedCallTarget.java:449)
1224d21f2:     nop
1224d21f3:     movl   8(%rax), %edi
1224d21f6:     addl   $1, %edi
1224d21f9:     jo     1691 <1224d289a>
1224d21ff:     movq   40(%rsp), %rcx
```

Get in touch

- github.com/Shopify/truffleruby   .../graal
- chris.seaton@shopify.com
- @ChrisGSeaton
- Graal Slack

Thanks!

shopify