
ORACLE®

John Tenniel illustrations public domain in the UK and US

Deoptimizing Ruby

JRuby+Truffle and the antidote to JITs

Chris Seaton
@ChrisGSeaton

Oracle Labs

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Safe Harbor Statement

The following is intended to provide some insight into a line of research in Oracle Labs. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. Oracle reserves the right to alter its development plans and practices at any time, and the development, release, and timing of any features or functionality described in connection with any Oracle product or service remains at the sole discretion of Oracle. Any views expressed in this presentation are my own and do not necessarily reflect the views of Oracle.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

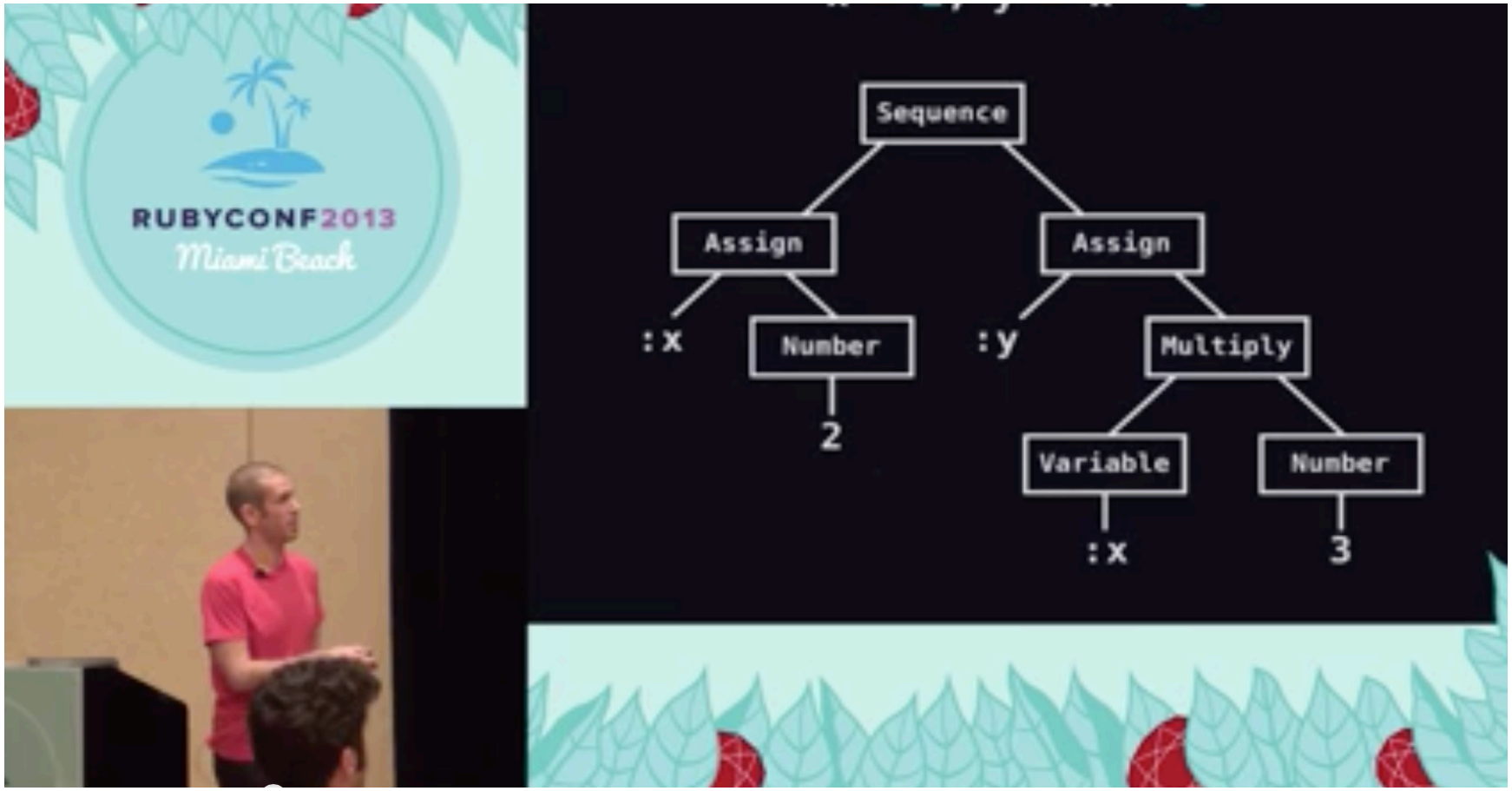
chrisseaton.com/rubytruffle/deoptimizing

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

JRuby+Truffle

A new open source implementation of Ruby by **Oracle Labs** with a **JIT** using **next-gen JVM** technology and **partial evaluation**, now part of **JRuby**

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



codon.com/compilers-for-free

Presentation, by Tom Stuart , licensed under a Creative Commons Attribution ShareAlike 3.0

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Why is Ruby hard to optimize?

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Fixnum to Bignum promotion

Monkey patching methods

#binding

ObjectSpace

set_trace_func

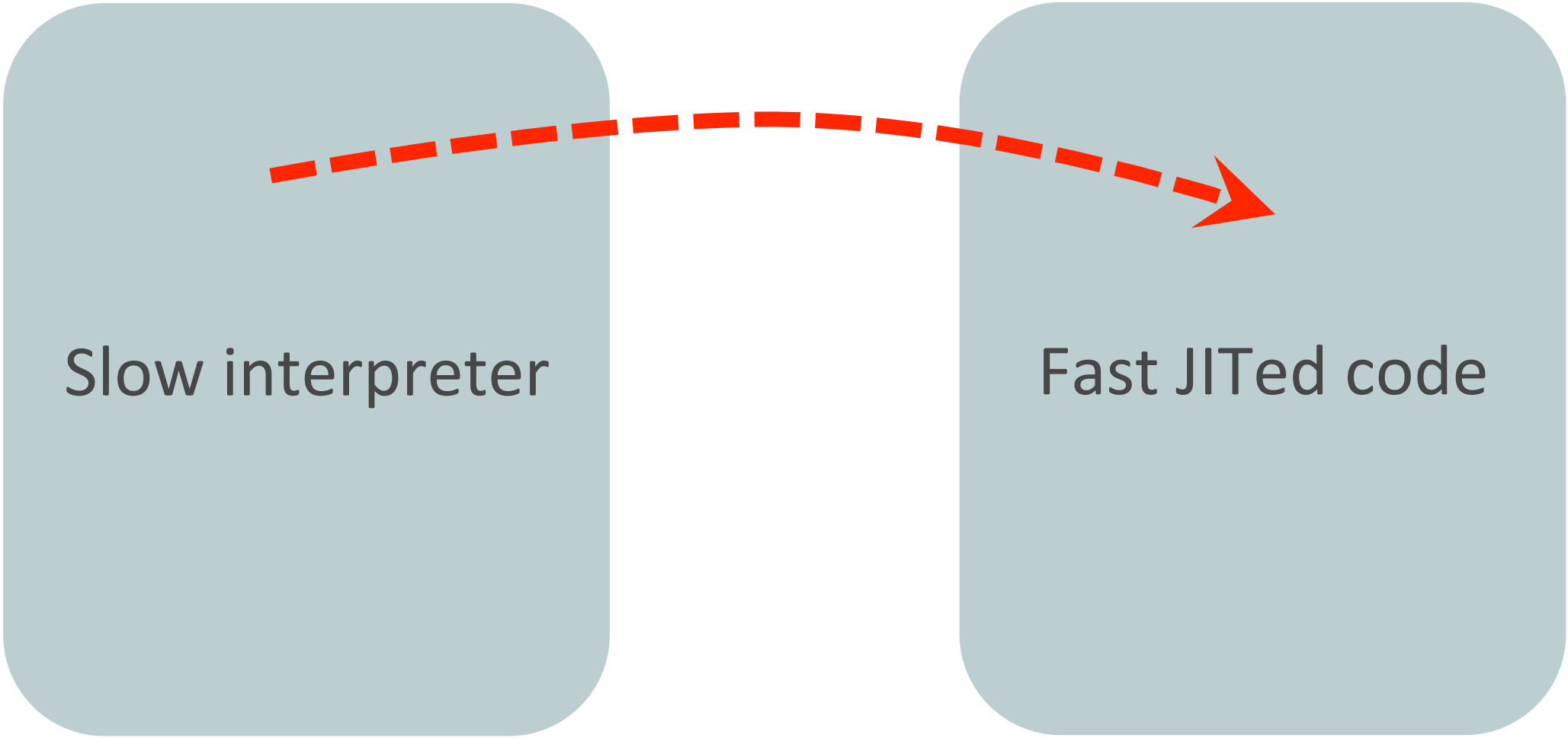
Thread#raise

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

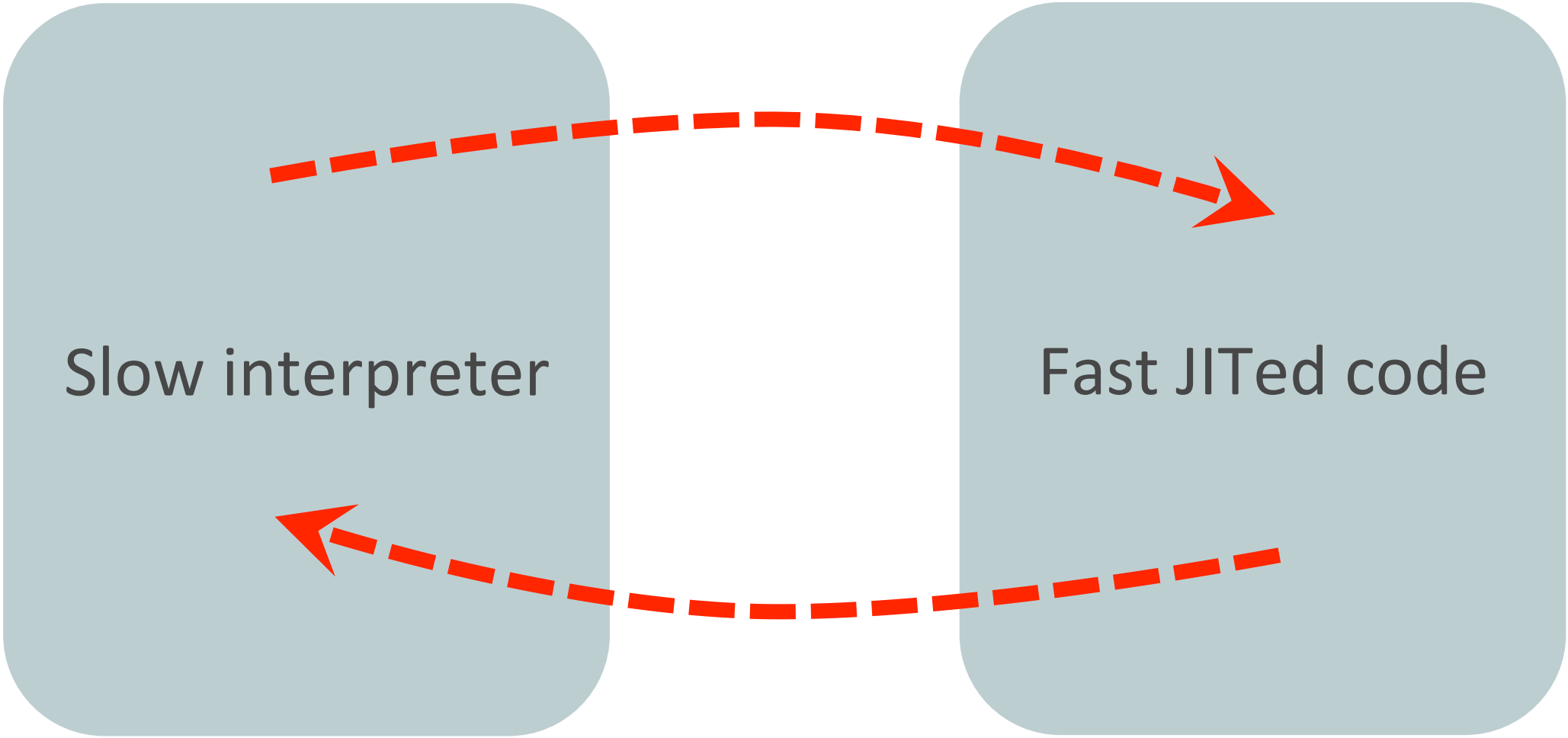
Deoptimization

elegantly solves all these problems

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Illustrating Deoptimization

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

John Tenniel illustrations public domain in the UK and US

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

John Tenniel illustrations public domain in the UK and US

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

John Tenniel illustrations public domain in the UK and US

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

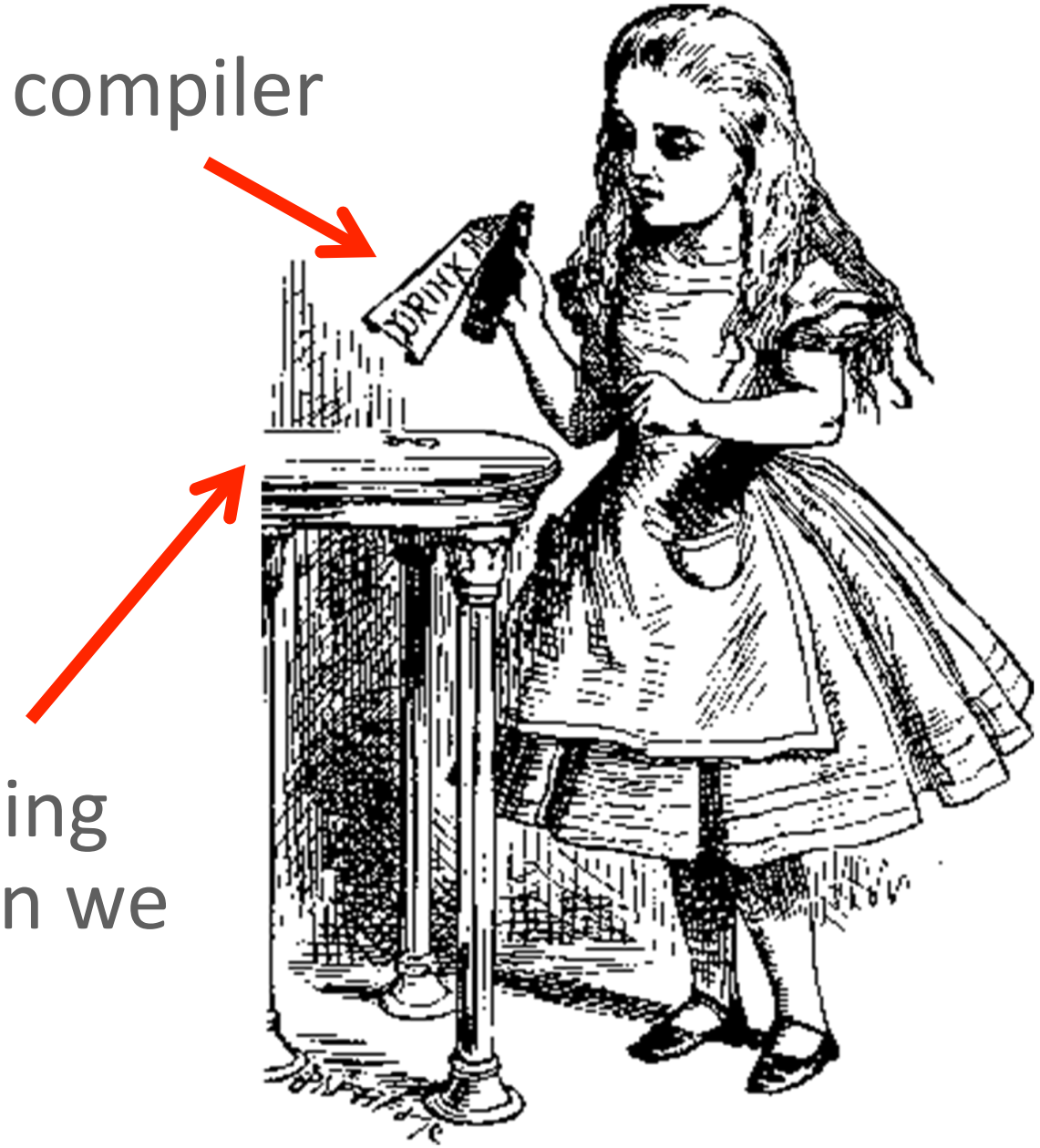
Ruby

Door to utopia of high performance

John Tenniel illustrations public domain in the UK and US

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Just-in-time compiler



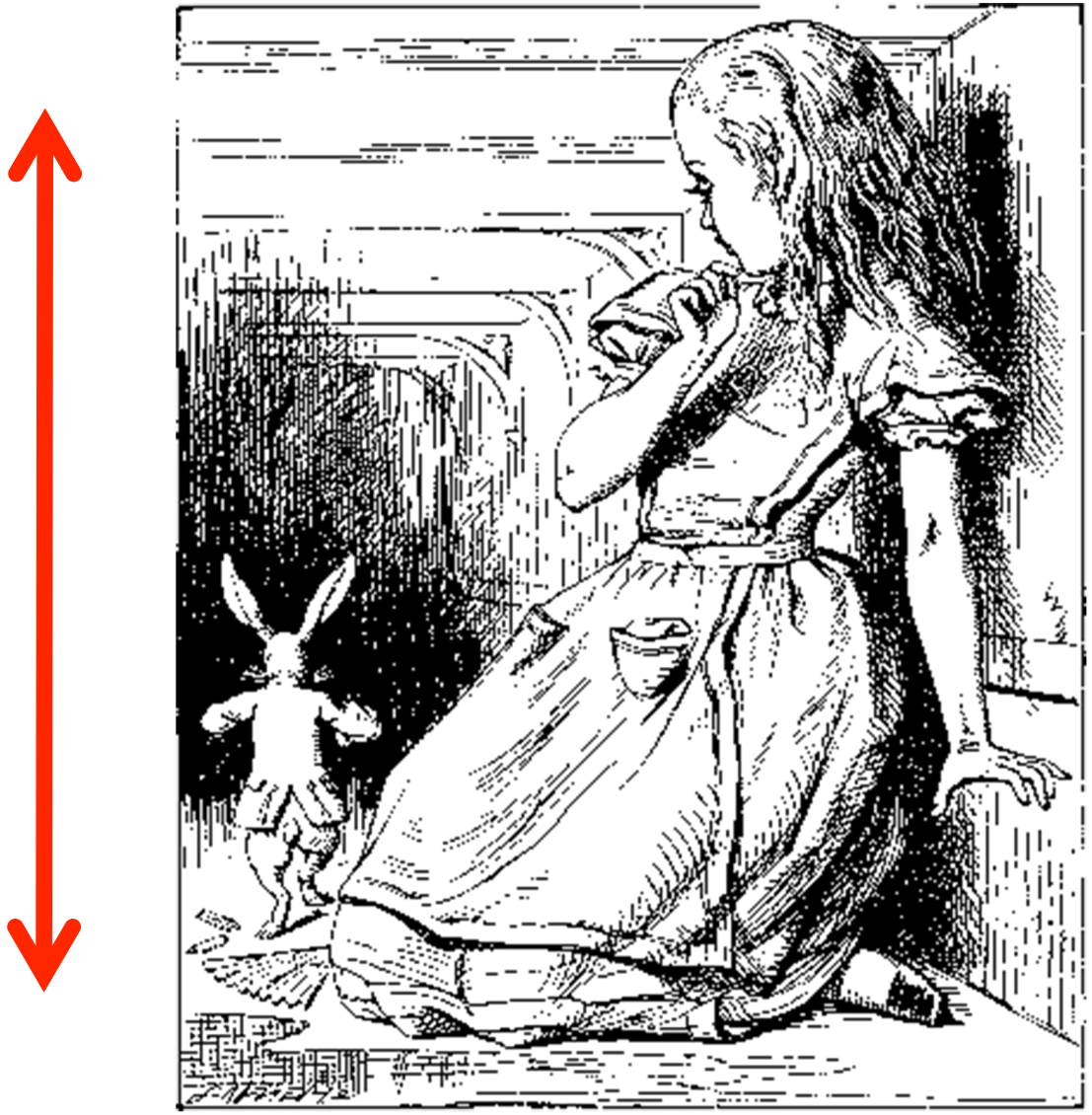
Left something behind when we compiled

John Tenniel illustrations public domain in the UK and US

ORACLE

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Deoptimization reverses the effects of the JIT



John Tenniel illustrations public domain in the UK and US

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

What does deoptimization do for Ruby?

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Fixnum to Bignum promotion

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

a + b + c

We'll assume we already know these are Fixnums

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
t1 = Fixnum(a) + Fixnum(b)
if t1.overflowed?
  t1 = Bignum(a) + Bignum(b)
  t2 = Bignum(t1) + Bignum(c)
else
  t2 = Fixnum(t1) + Fixnum(c)
  if t2.overflowed?
    t2 = Bignum(t1) + Bignum(c)
  end
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
t1 = Fixnum(a) + Fixnum(b)
deoptimize! if t1.overflowed?
t2 = Fixnum(t1) + Fixnum(c)
deoptimize! if t2.overflowed?
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
t1 = Fixnum(a) + Fixnum(b)
if t1.overflowed?
  t1 = Bignum(a) + Bignum(b)
  t2 = Bignum(t1) + Bignum(c)
else
  t2 = Fixnum(t1) + Fixnum(c)
  deoptimize! if t2.overflowed?
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Monkey patching methods

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
my_object.my_method(x, y)
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
lookup my_method in my_object
call it with (x, y)
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
if my_object.changed?
  lookup my_method in my_object
  call it with (x, y)
else
  use cached my_method
  call it with (x, y)
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
if my_object.changed?
  deoptimize!
else
  use cached my_method
  call it with (x, y)
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
use cached my_method
call it with (x, y)
```

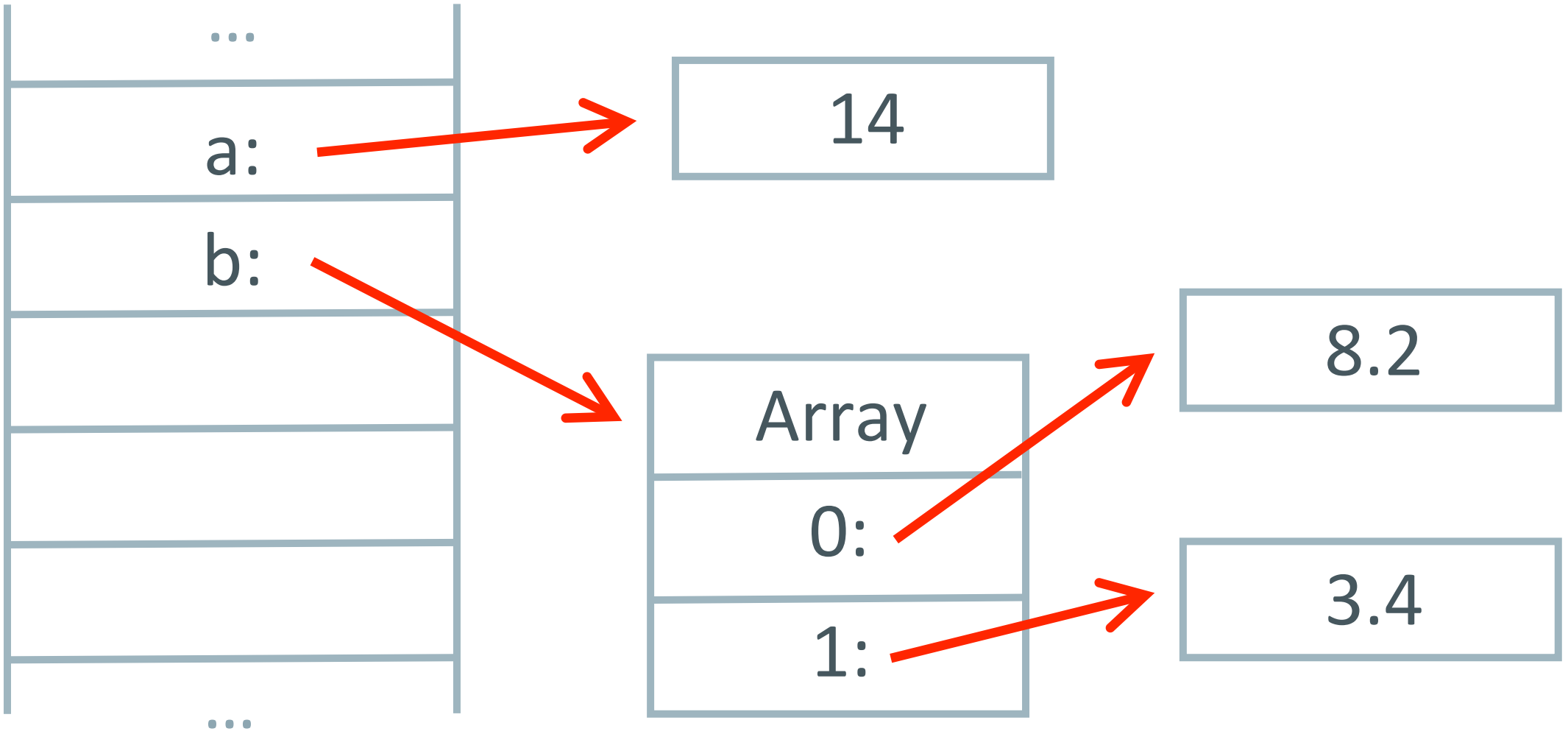
Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

#binding

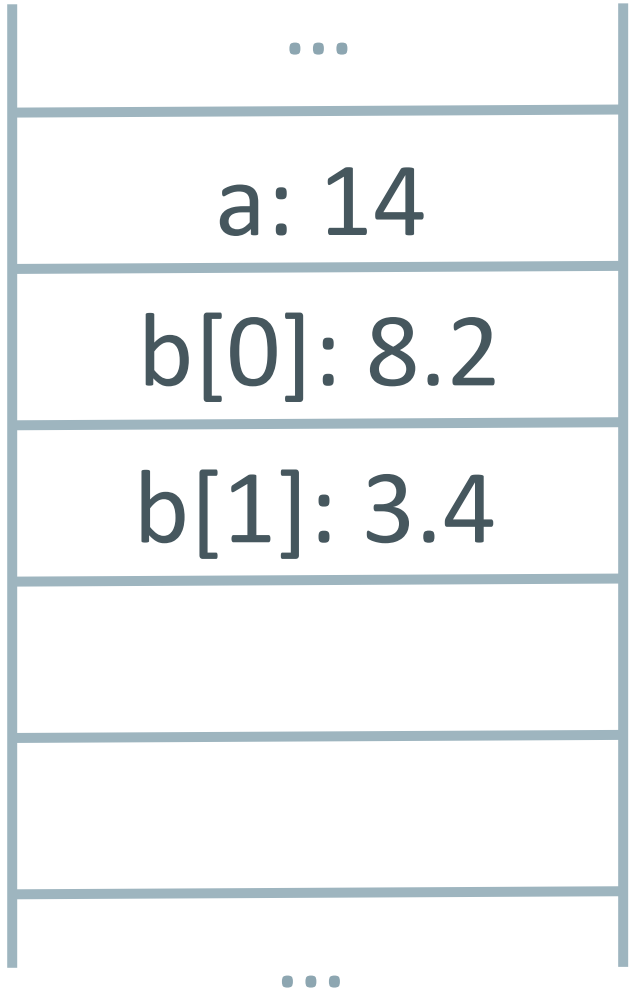
Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
a = 14
b = [8.2, 3.4]
```

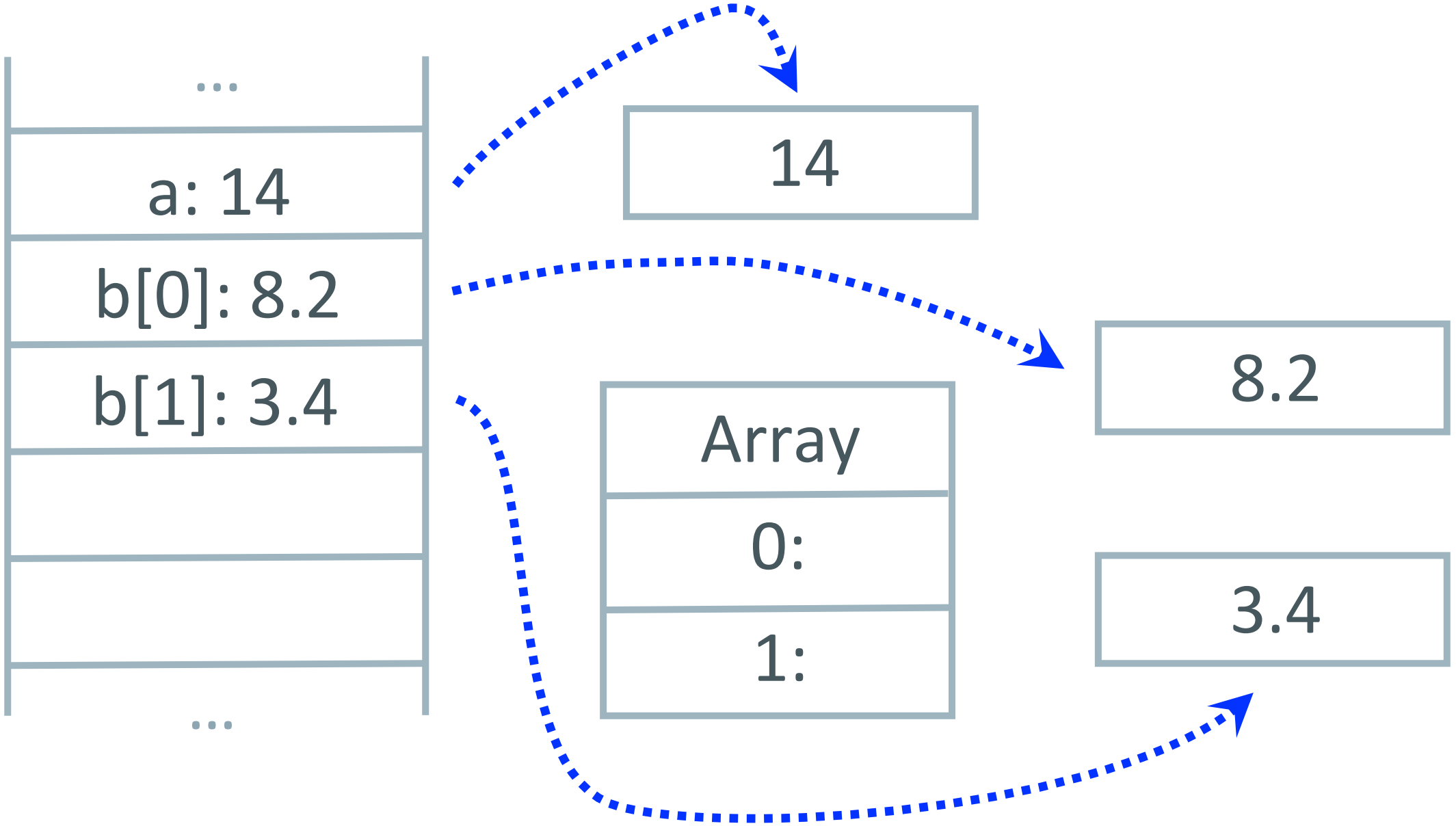
Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

ObjectSpace

set_trace_func

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

How JRuby+Truffle Deoptimizes

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

1. Recreate the interpreter stack frame
2. Jump from the JITed code into the interpreter
3. Allow us to force threads to do this

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
loop do
  a = 14
  b = 2
  a + b
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
loop do
  a = 14
  b = 2
  a + b
  deoptimize! if should_deoptimize?
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

```
loop do
  a = 14
  b = 2
  a + b
  read the safepoint page
end
```

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

JRuby+Truffle Performance

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

86%

RubySpec language specs

rubyspec.org, Brian Shirai et al

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Method invalidation

#send

#binding

Float

Threads

Frame-local variables

C extensions

Encodings

ObjectSpace

Regexp

Thread#raise

#eval

Fixnum to Bignum promotion

set_trace_func

Proc#binding

Closures

Concurrency

Constant invalidation

Debugging

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Method invalidation

#send

#binding

Float

Threads

Frame-local variables

C extensions

Encodings

ObjectSpace

Regexp

Thread#raise

#eval

Fixnum to Bignum promotion

set_trace_func

Proc#binding

Closures

Concurrency

Constant invalidation

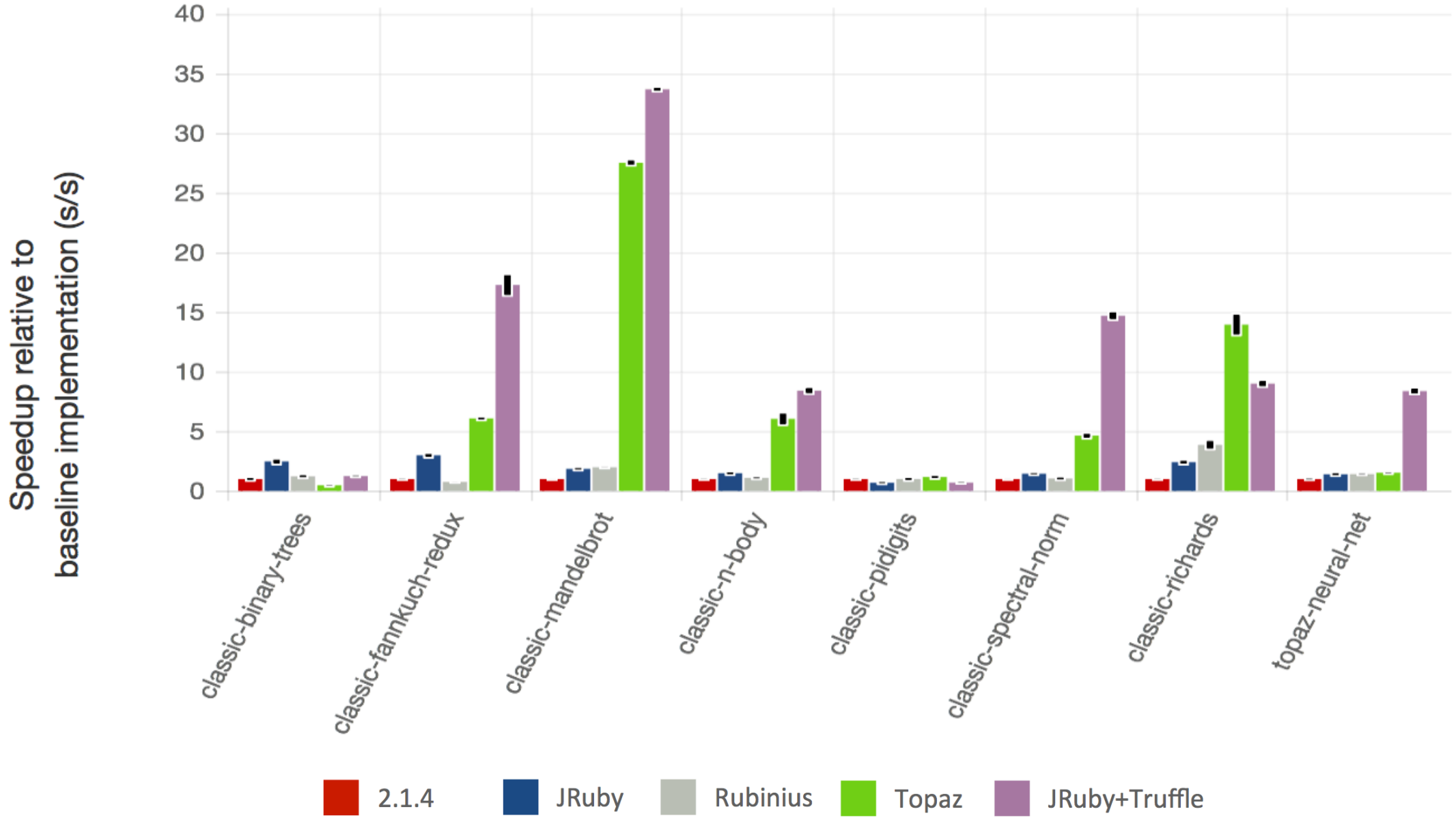
Debugging

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

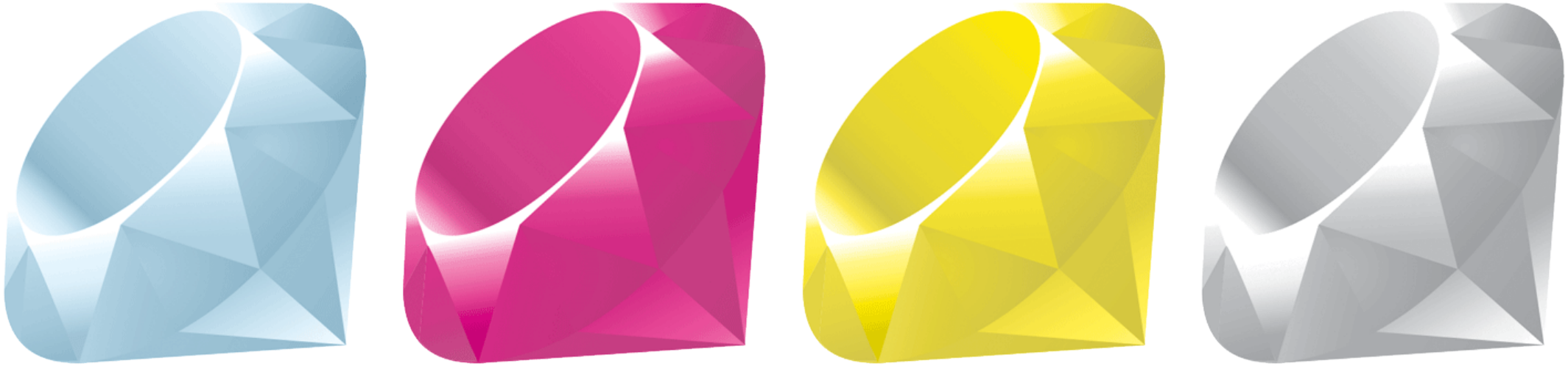
No, we can’t run Rails yet

but we’re working towards it

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

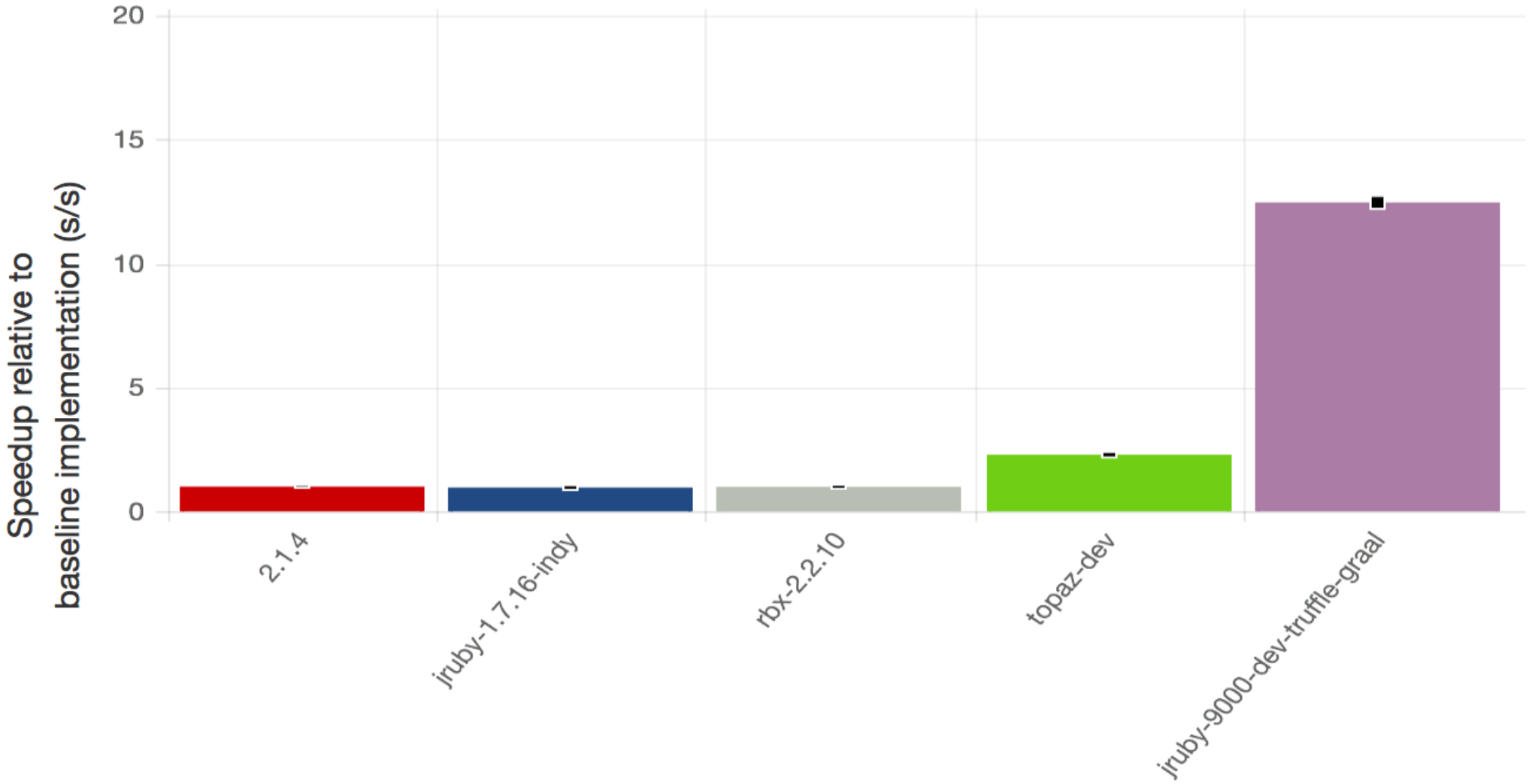


chunky_png and psd.rb

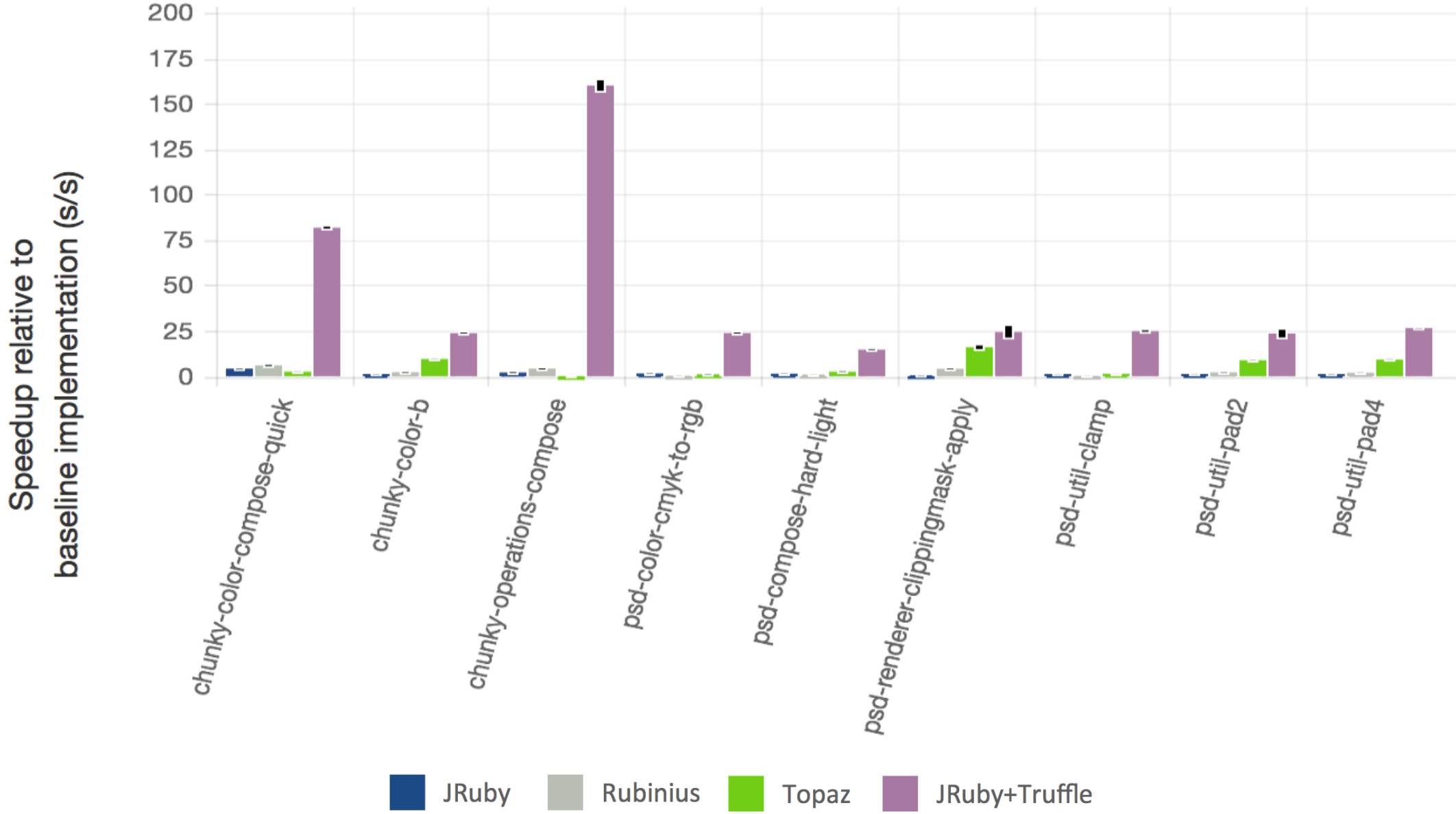
Willem van Bergen, Ryan LeFevre, Kelly Sutton, Layer Vault, Floorplanner et al

Ruby logo copyright (c) 2006, Yukihiro Matsumoto, licensed under the terms of the Creative Commons Attribution-ShareAlike 2.5 agreement

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

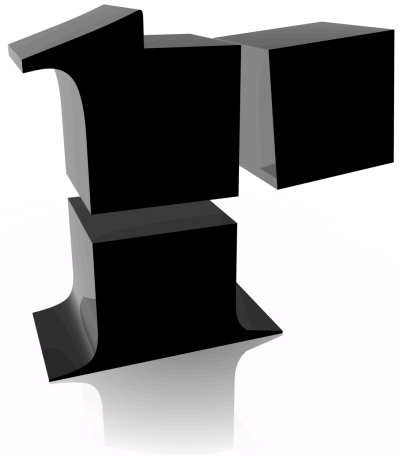


Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Conclusions

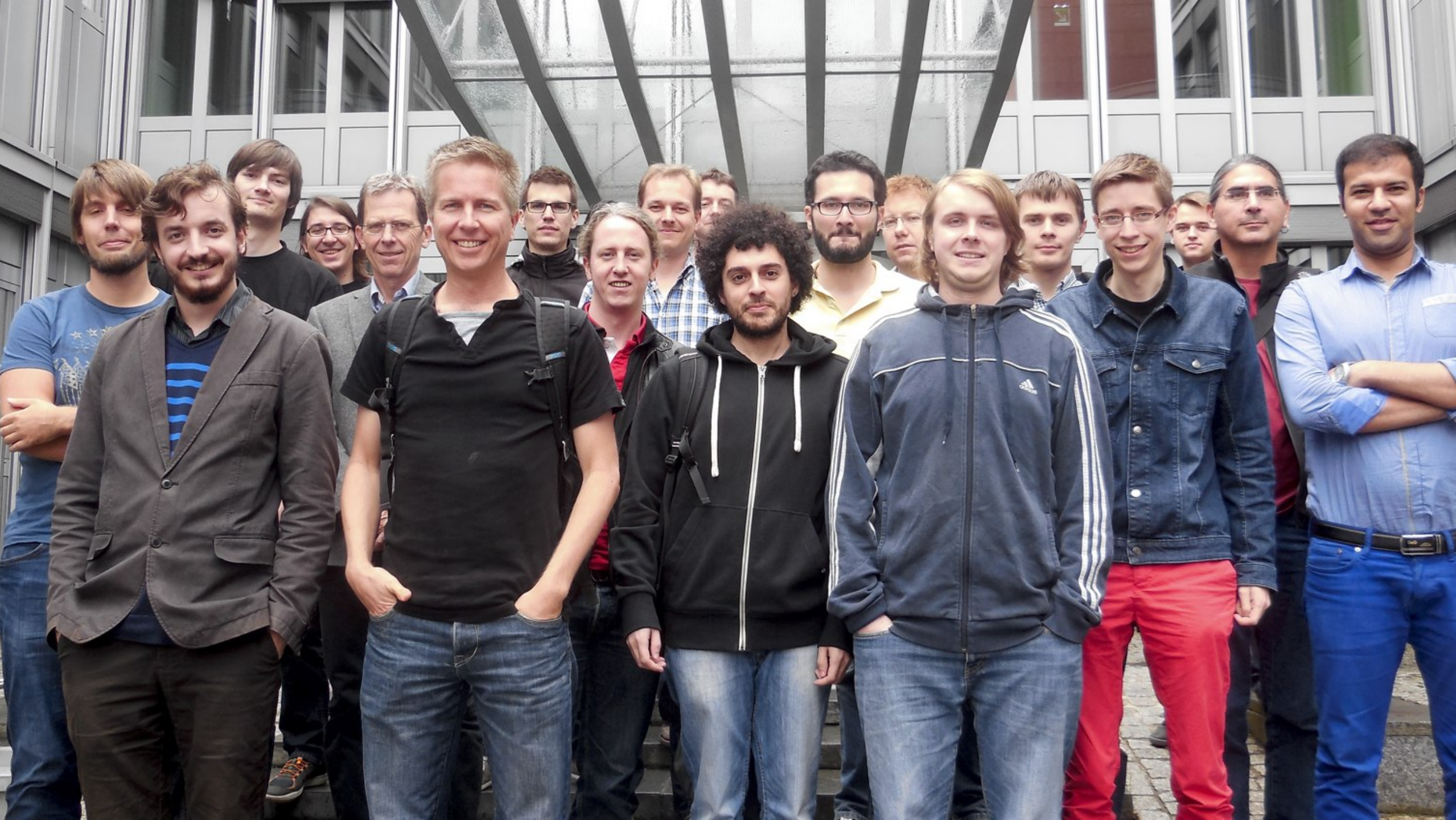
Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Building on other projects



JRuby logo copyright (c) Tony Price 2011, licensed under the terms of the Creative Commons Attribution-NoDerivs 3.0 Unported (CC BY-ND 3.0)
Ruby logo copyright (c) 2006, Yukihiro Matsumoto, licensed under the terms of the Creative Commons Attribution-ShareAlike 2.5 agreement
Rubinius logo licensed under the terms of the Creative Commons Attribution-NoDerivs 3.0 Unported

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |



@ChrisGSeaton

chrisseaton.com/rubytruffle/deoptimizing

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Safe Harbor Statement

The preceding is intended to provide some insight into a line of research in Oracle Labs. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. Oracle reserves the right to alter its development plans and practices at any time, and the development, release, and timing of any features or functionality described in connection with any Oracle product or service remains at the sole discretion of Oracle. Any views expressed in this presentation are my own and do not necessarily reflect the views of Oracle.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Hardware and Software
Engineered to Work Together

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |


ORACLE®