

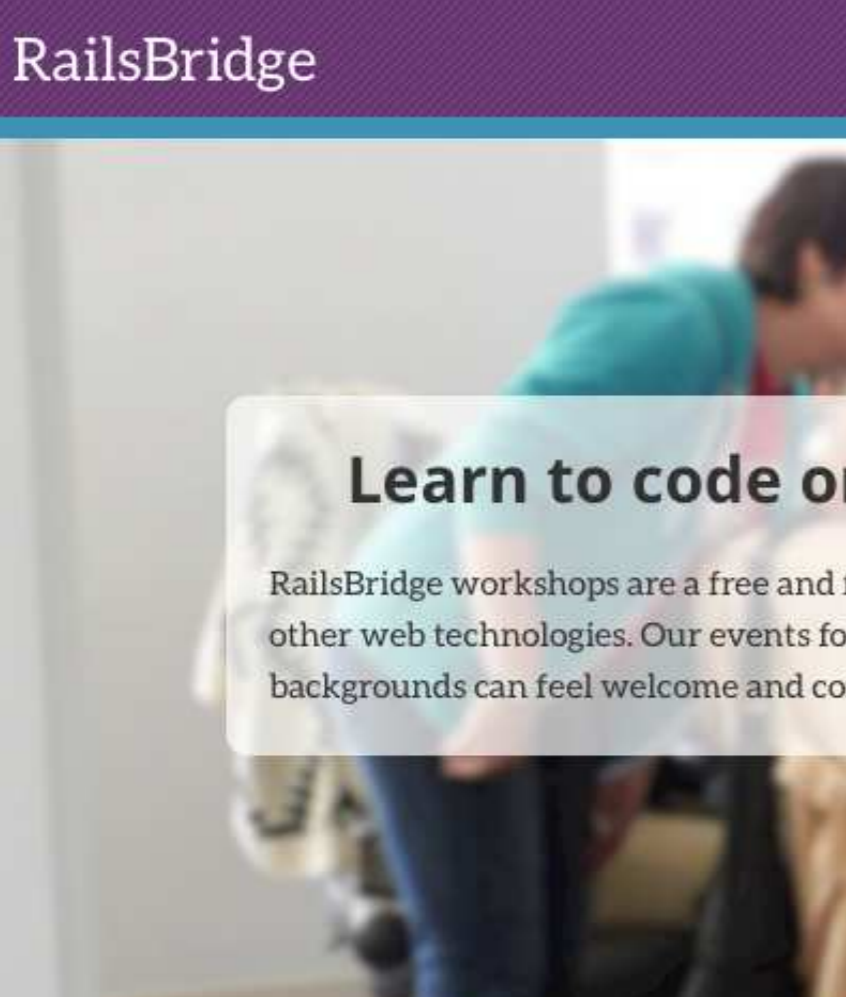
Code Is What

Code Does

@brixen

I <3 Ruby

RailsBridge



Learn to code on

RailsBridge workshops are a free and fun way to learn about other web technologies. Our events for people of all backgrounds can feel welcome and cool.



HOME ABOUT DOWNLOAD BLOG

Have fun and make games, or hack using Ruby! Just tell your parents or teachers you're learning Ruby programming... ;)

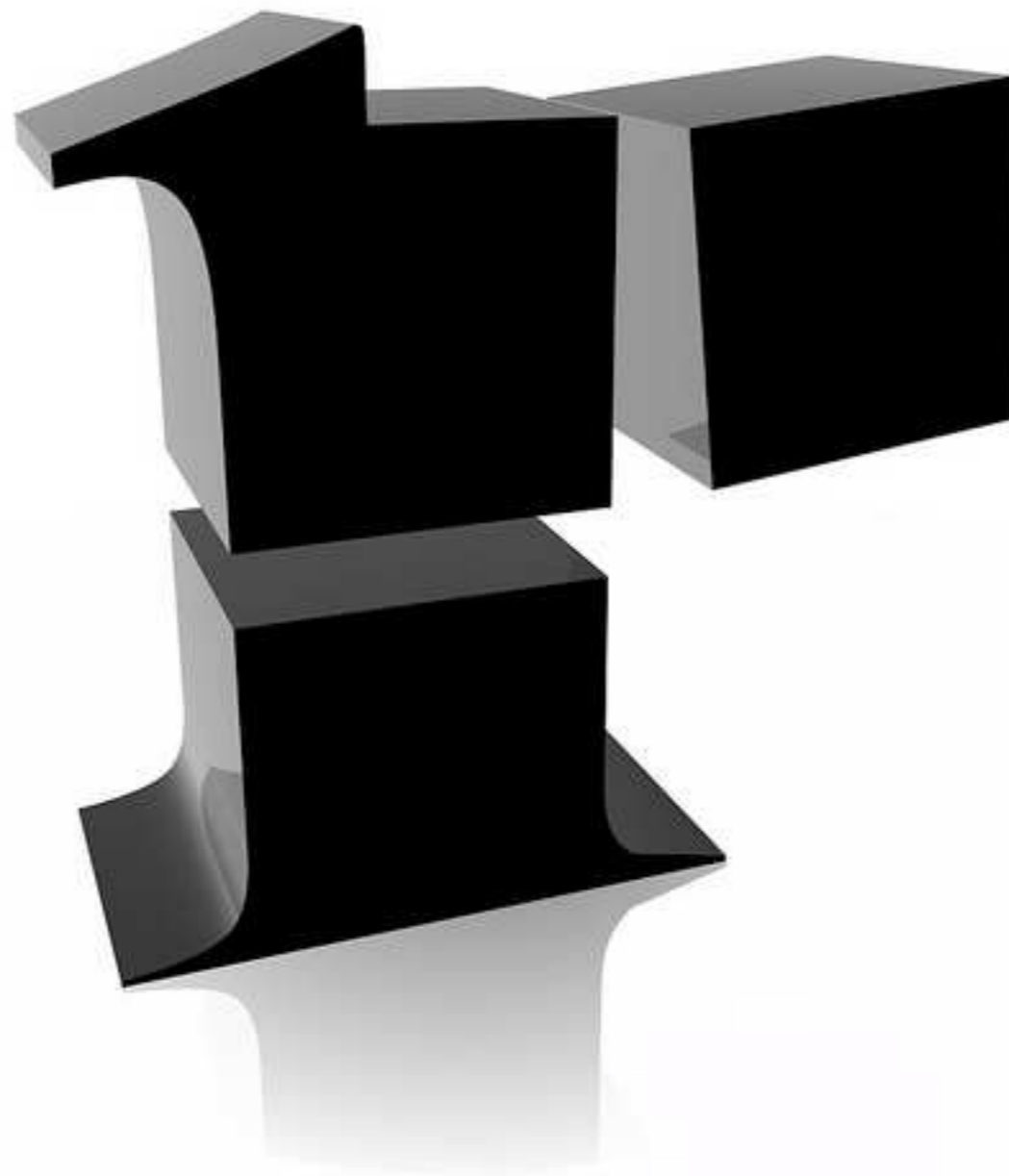
Rails Girls

Events ★ Materials ★ Projects

Press Kit

Rails Girls aims to give tools for women to build their ideas.







enova.®

**We live in a world filled
with hard problems**

[View Patients](#)[Donate Monthly](#)[About Us](#)[FAQ](#)[Sign Up](#)[Log In](#)

Directly fund low-cost, high-impact
medical care for people in need.

100% of donations fund patient care.

[View Patients](#)

New Patients

[View All Patients](#)

\$195 To Go | 1 Donor



\$325 To Go | 7 Donors



\$525 To Go | 15 Donors

**People are leaving
Ruby**

Scala Javascript

Node.js

Clojure

Elixir

Go

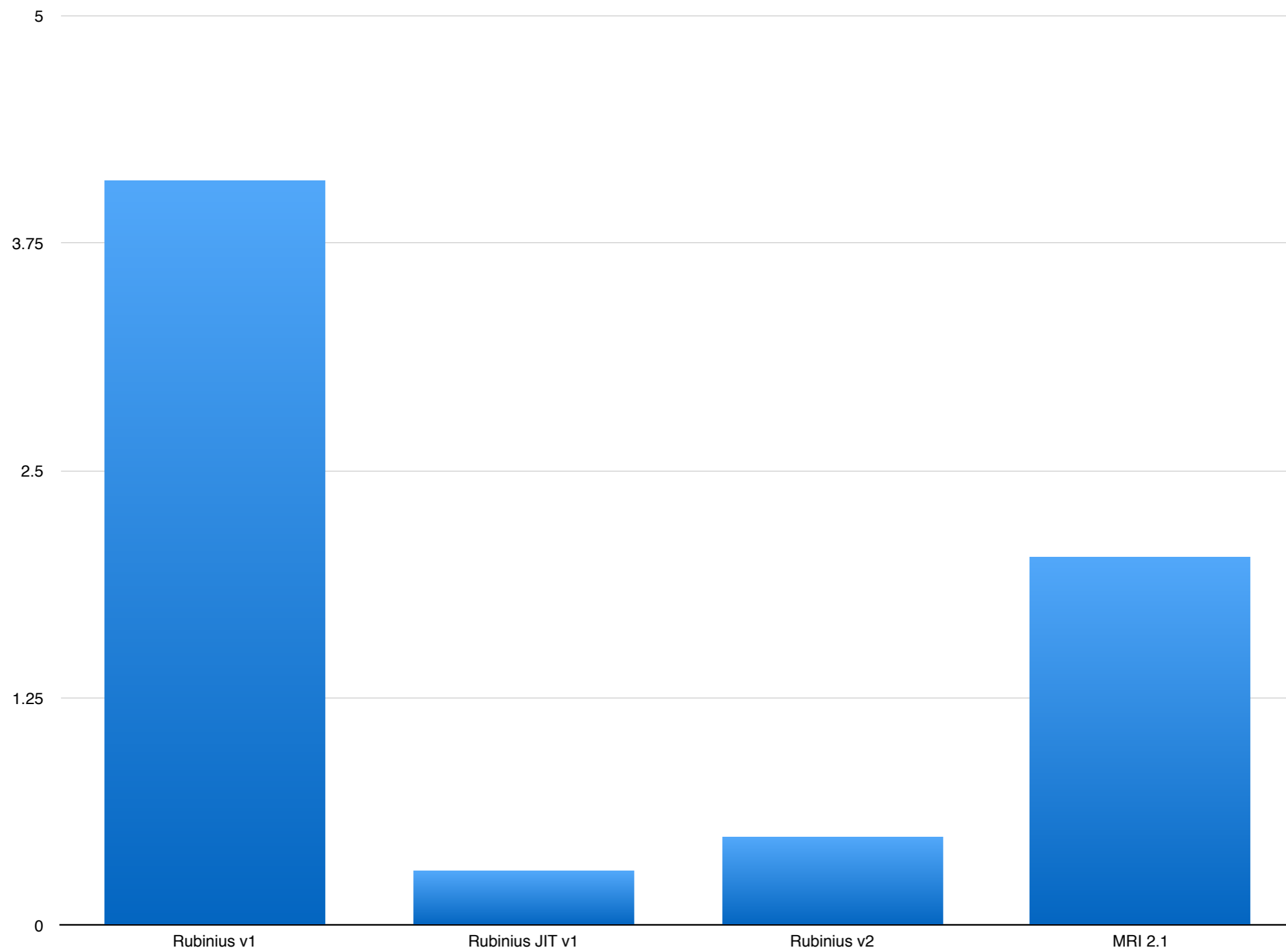
Rust

Haskell

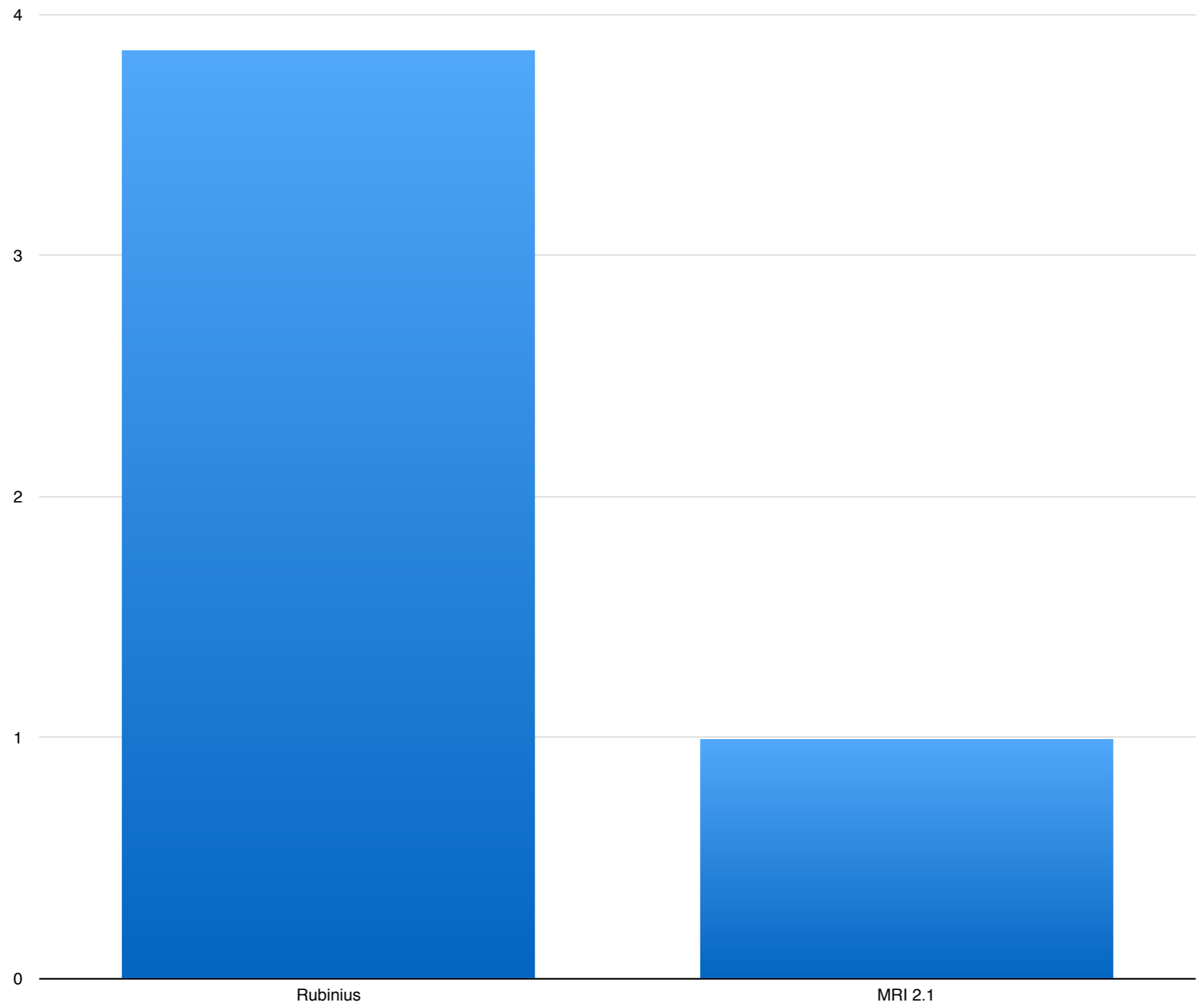
**Businesses are not
choosing Ruby**

Ruby has problems

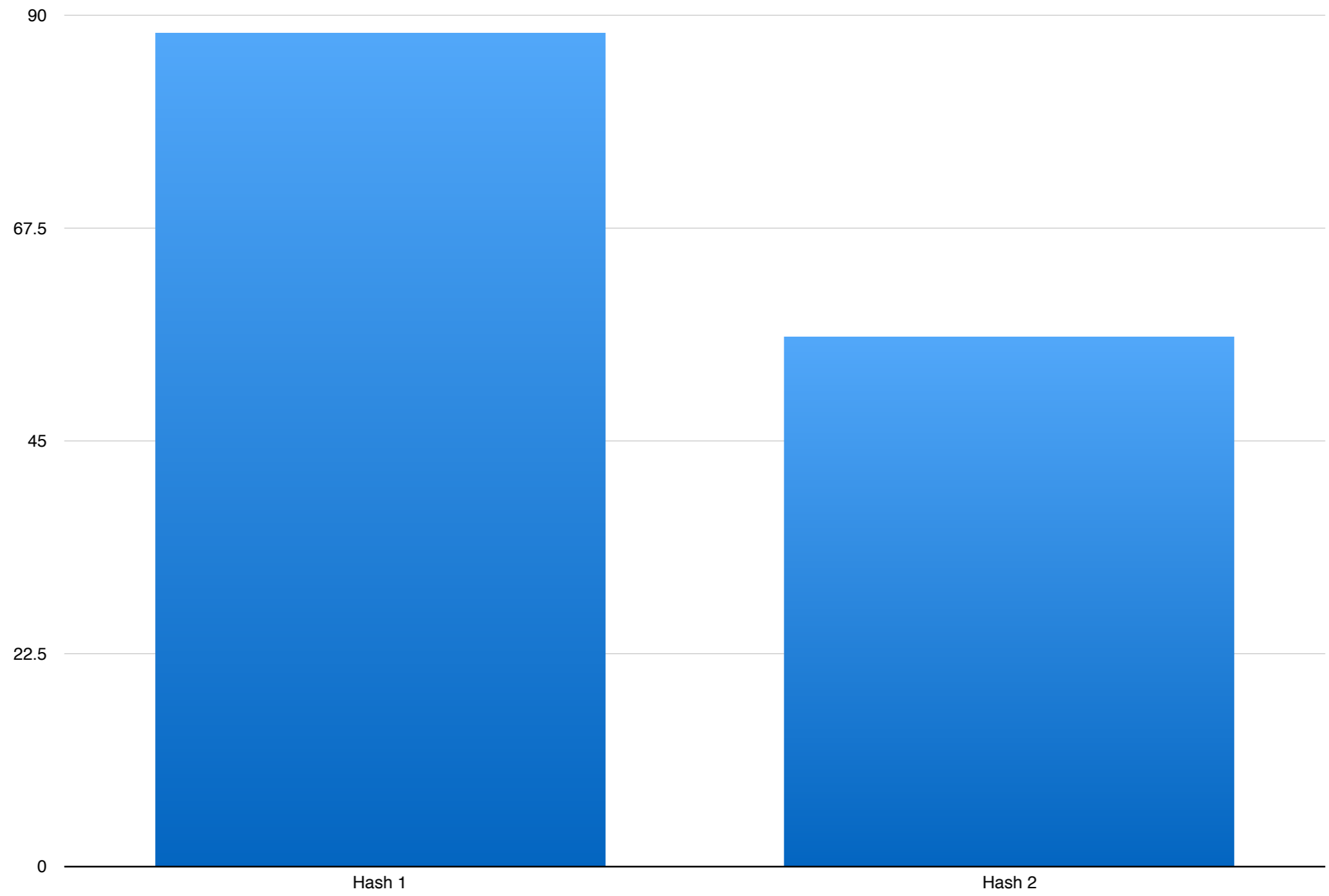
Too slow



No concurrency



Too much memory



Not compiled

Monkey patching

```
$ rbx
```

```
irb(main):001:0> "abc".clear
```

```
=> ""
```

```
irb(main):002:0> require "./some_gem.rb"
```

```
=> true
```

```
irb(main):003:0> "abc".clear
```

```
=> "nope"
```

```
irb(main):004:0>
```

```
irb(main):003:0> "abc".clear
```

```
=> "nope"
```

```
irb(main):004:0> "abc".method :clear
```

```
=> #<Method: String#clear (defined in String  
at some_gem.rb:2)>
```

```
irb(main):005:0>
```

```
1 class String
2     def clear
3         replace "nope"
4     end
5 end
6
```

Static types

Why use *static* types?

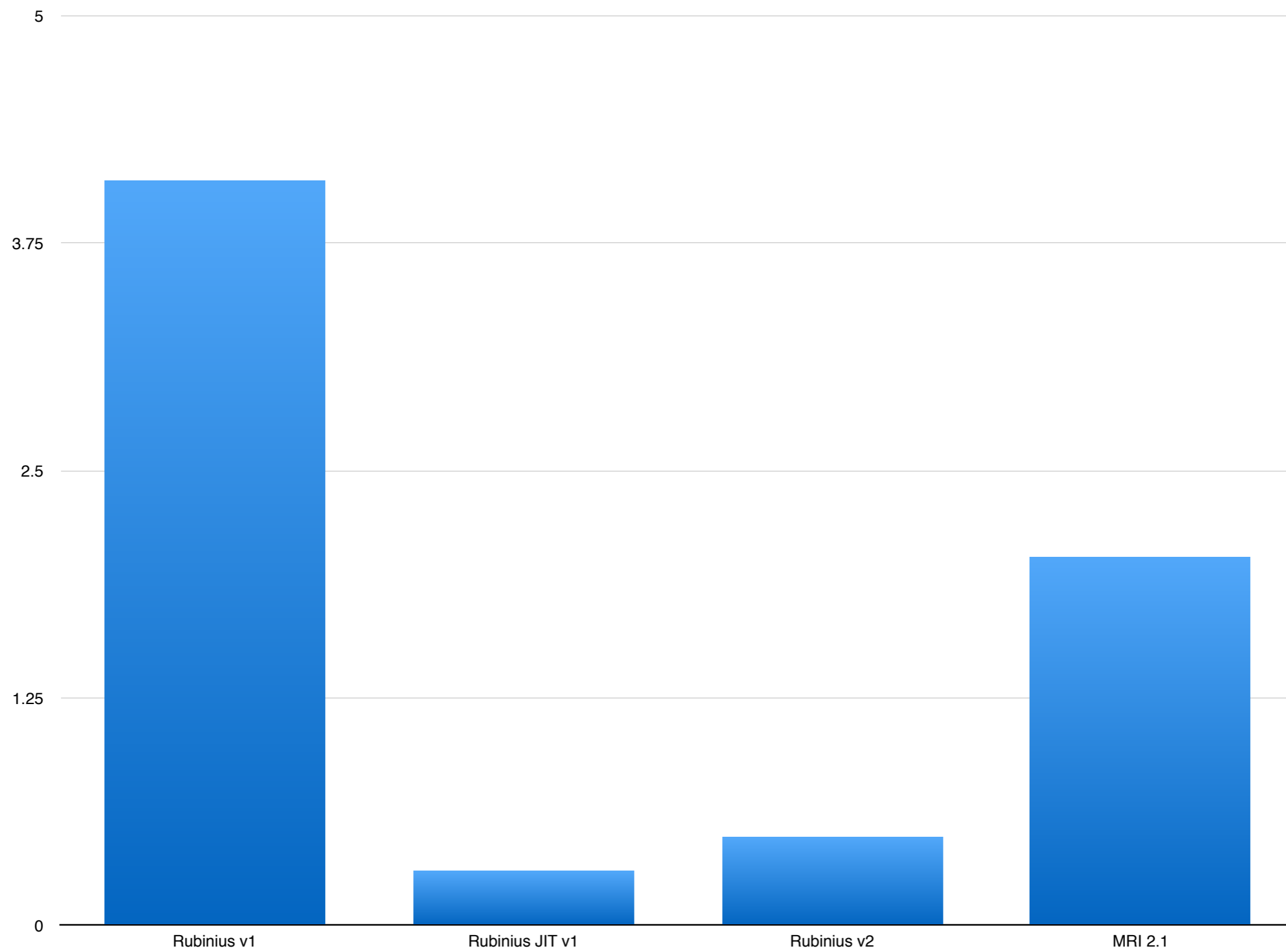
1

Correctness

**Adding types does not make
correct code more correct**

2

Speed



3

Maintenance

**Adding complexity does not
improve understanding**

**Ruby is a better
choice**

**Programming is
too hard**

We are too slow

We fail a lot

The Cynefin framework

1

Simple

Sense • Categorize • Respond

1

Simple

Sense • Categorize • Respond

Best Practice

2

Complicated

Sense • Analyze • Respond

2

Complicated

Sense • Analyze • Respond

Good Practice

3

Complex

Probe • Sense • Respond

3

Complex

Probe • Sense • Respond

Emergent Behavior

4

Chaotic

Act • Sense • Respond

4

Chaotic

Act • Sense • Respond

Novel Solutions

Clay vs concrete



B.C.

A.D.

B.C.

Before compilation

A.D.

After deployment

Read

Eval

Print

Loop

**Study
Experiment
Check
Loop**

**Assumptions help us
solve problems**

**Assumptions hinder
our efforts**

~~Technical debt~~

Cost to experiment

~~Refactoring~~

Rework

**We live in a world filled
with hard problems**

**Let's make
programming easier**

**Let's test our ideas
more quickly**

**Let's fail more
productively**

Improve the world

Try Rubinius

github.com/rubinius/rubinius

Thank you!

Credits

- http://winstonyw.com/2013/06/12/reddotrubyconf_2013_-_thank_you/
- <https://www.flickr.com/photos/veganstraightedge/6037027106/>
- http://www.europeancarweb.com/news/epcp_1010_audi_e_tron_spyder_creation/photo_13.html
- <http://thisisbuildingmaterials.blogspot.com/2012/03/examples-of-famous-concrete-buildings.html>