



ORACLE



Polyglot programming in **GraalVM**TM with Tooling, AOT compilation, and Embedding

Petr Chalupa

Principal Member of Technical Staff
Oracle Labs
November 28, 2019

Safe harbor statement

The following is intended to outline our general product direction. 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.

The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

GraalVM Native Image technology (including SubstrateVM) is early adopter technology. It is available only under an early adopter license and remains subject to potentially significant further changes, compatibility testing and certification.



Program agenda

- 1 GraalVM
- 2 Truffle
- 3 Polyglot
- 4 Embedding and native image generation
- 5 Instrumentation and Tooling
- 6 Language status



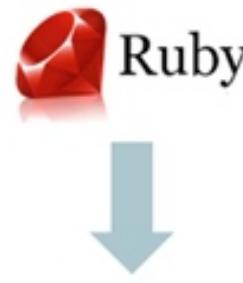
GraalVM™

What is GraalVM?

- Drop-in replacement for Oracle Java 8 and Java 11
 - Run your Java application faster
- Ahead-of-time compilation for Java
 - Create standalone binaries with low footprint
- High-performance JavaScript, Python, Ruby, R, ...
 - The first VM for true polyglot programming
 - Implement your own language or DSL

What is Graal?

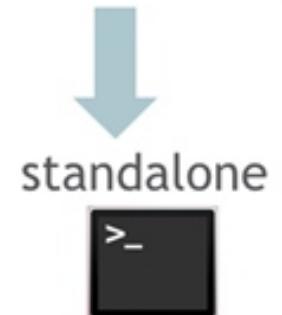
- A Java just in time compiler
 - At runtime, translates Java bytecodes into machine code
- Modern design, implemented in Java
 - Maintainable
 - Extendable
- Fully Java compliant => production ready
- High performance



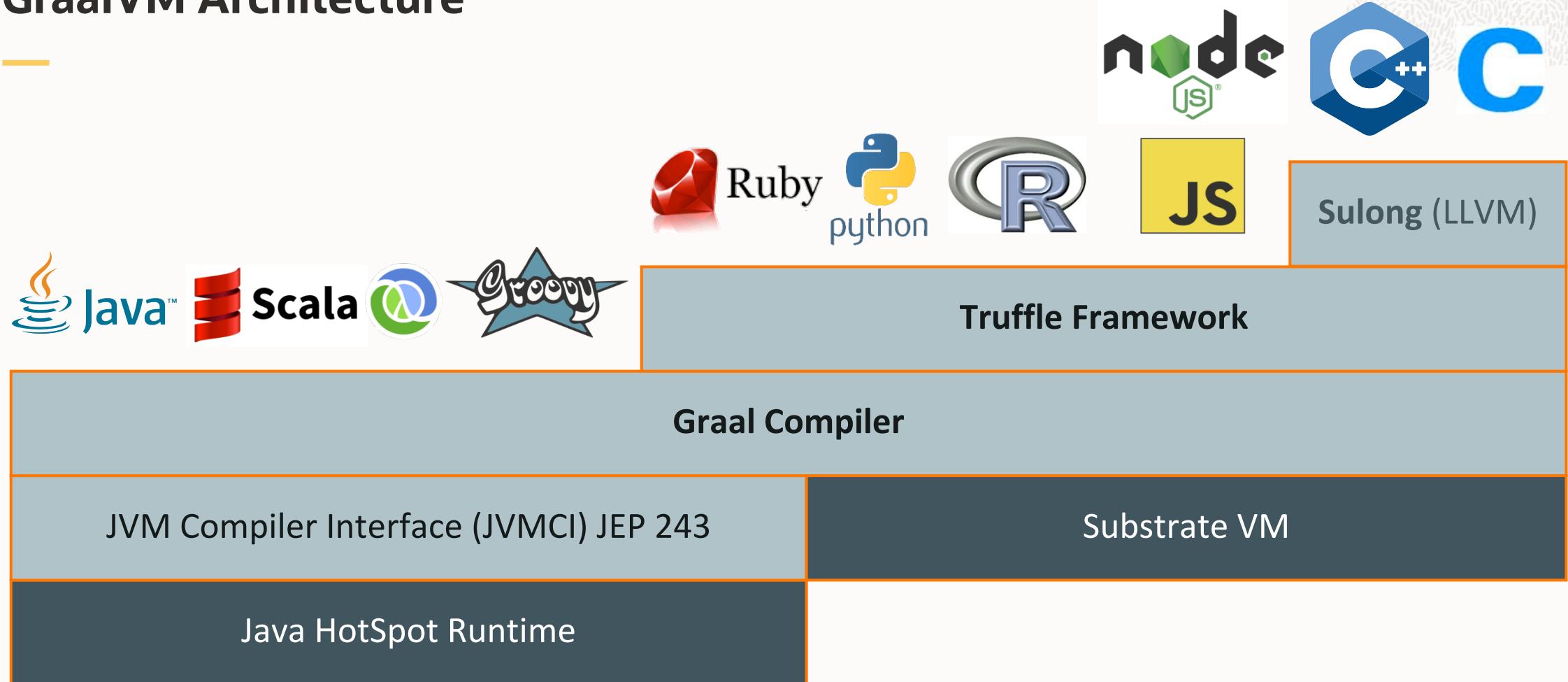
Automatic transform of interpreters to compiler

GraalVM™

Engine integration native and managed



GraalVM Architecture



Community Edition

GraalVM Community is available for free for evaluation, development and production use. It is built from the GraalVM sources available on [GitHub](#). We provide pre-built binaries for Linux, macOS X, and Windows platforms on x86 64-bit systems. Windows support is [experimental](#).

[DOWNLOAD FROM GITHUB](#)

Enterprise Edition

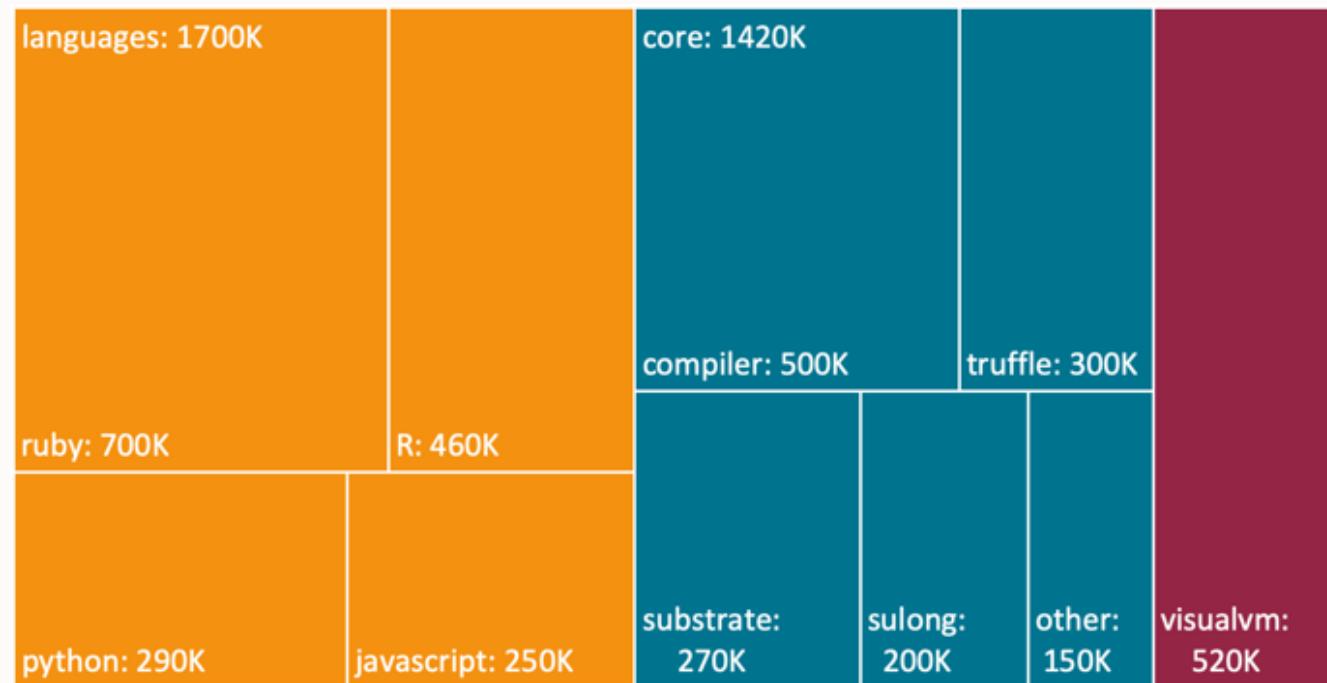
GraalVM Enterprise provides additional performance, security, and scalability relevant for running applications in production. It is free for evaluation uses and available for download from the [Oracle Technology Network](#). We provide binaries for Linux, macOS X, and Windows platforms on x86 64-bit systems. Windows support is [experimental](#).

[DOWNLOAD FROM OTN](#)

FREE on Oracle Cloud!

GraalVM Open Source

Open Source LOC actively maintained by GraalVM team

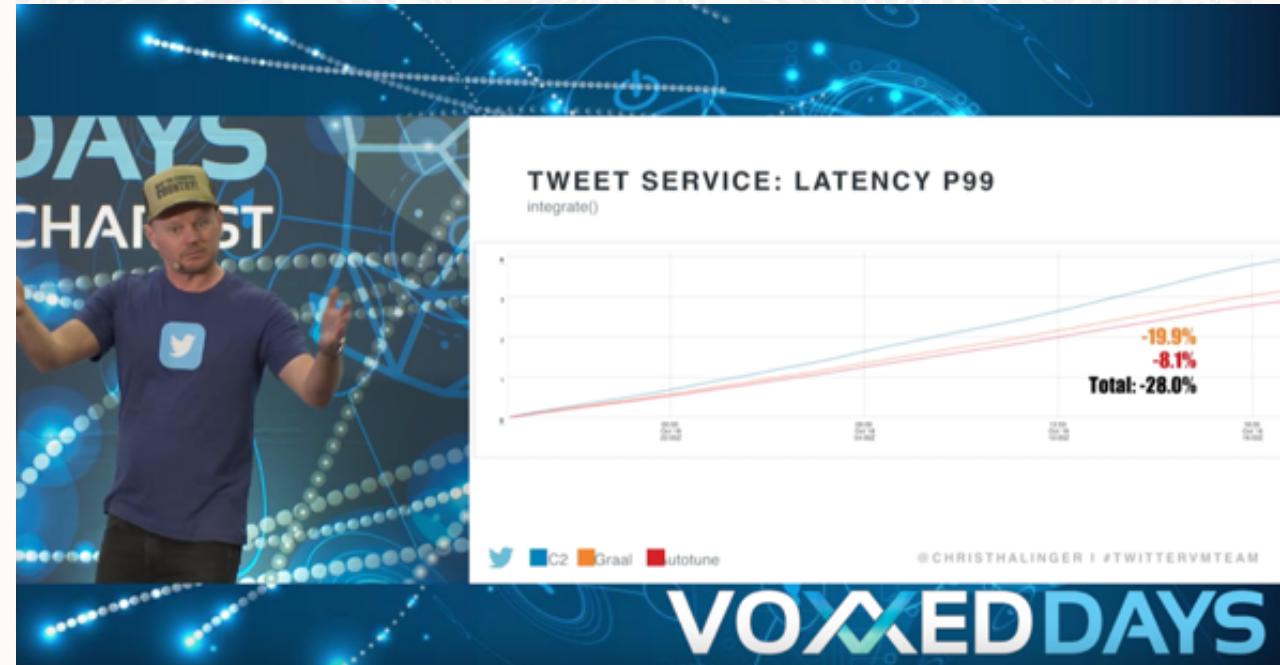


Total: 3,640,000 lines of code



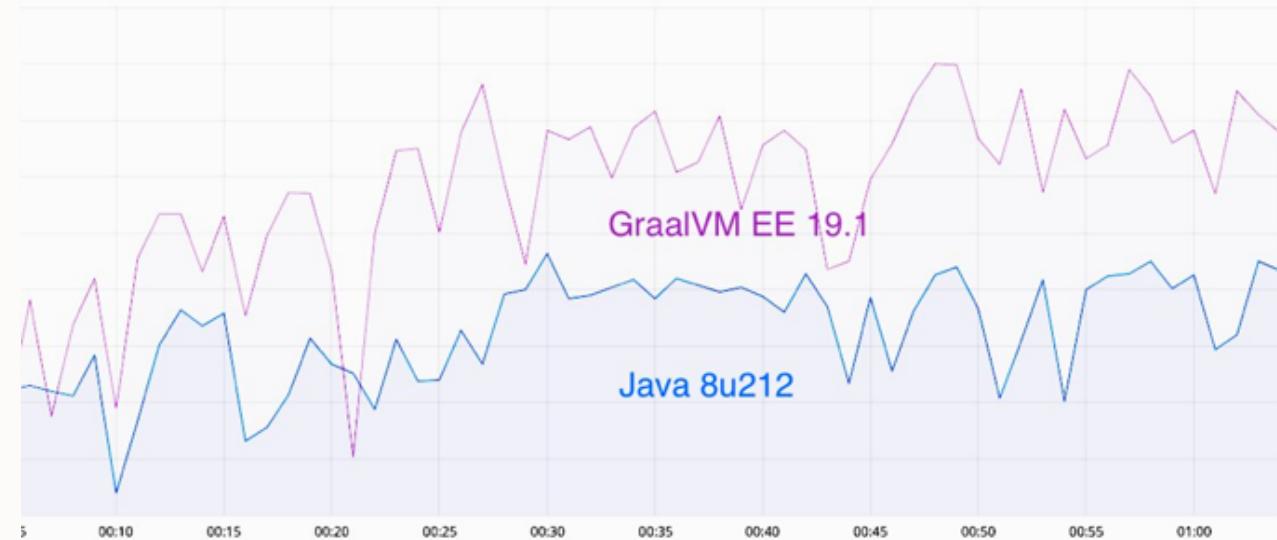
—

Twitter uses GraalVM compiler in production to run their Scala microservices



ORACLE® Cloud Infrastructure

- Peak performance: +10%
- Garbage collection time: -25%
- Seamless migration



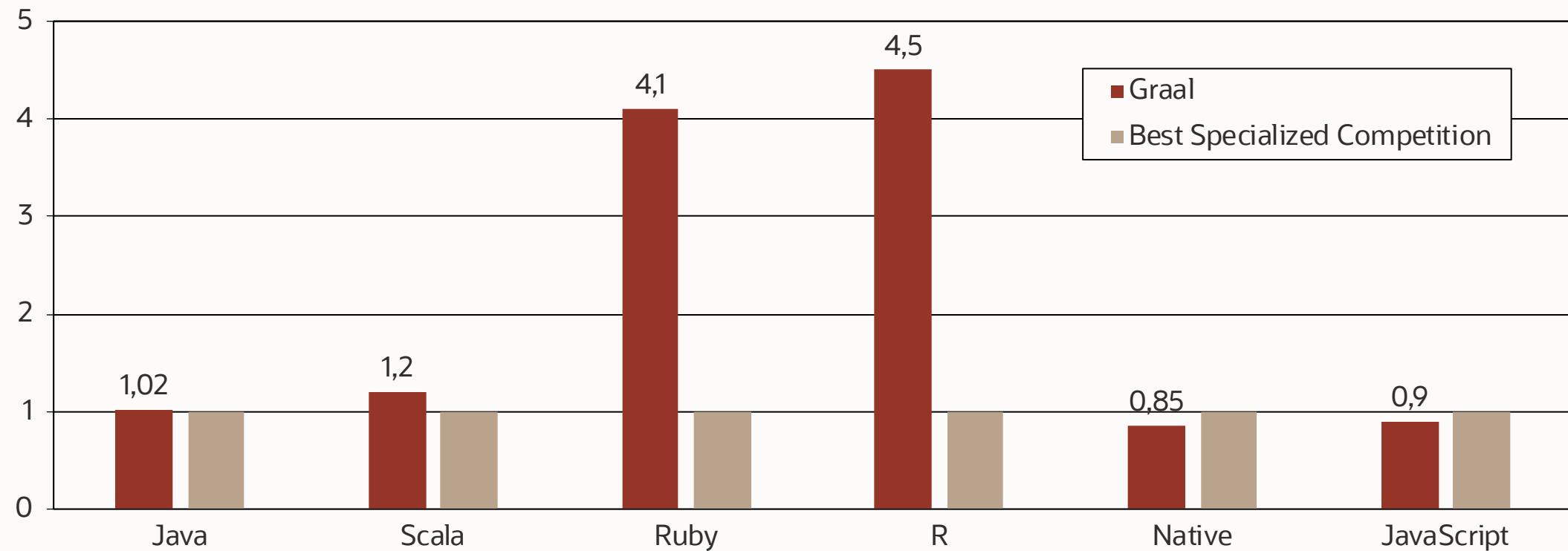


The rich ecosystem of CUDA-X libraries is now available for GraalVM applications.

GPU kernels can be directly launched from GraalVM languages such as R, JavaScript, Scala and other JVM-based languages.

Performance: GraalVM Summary

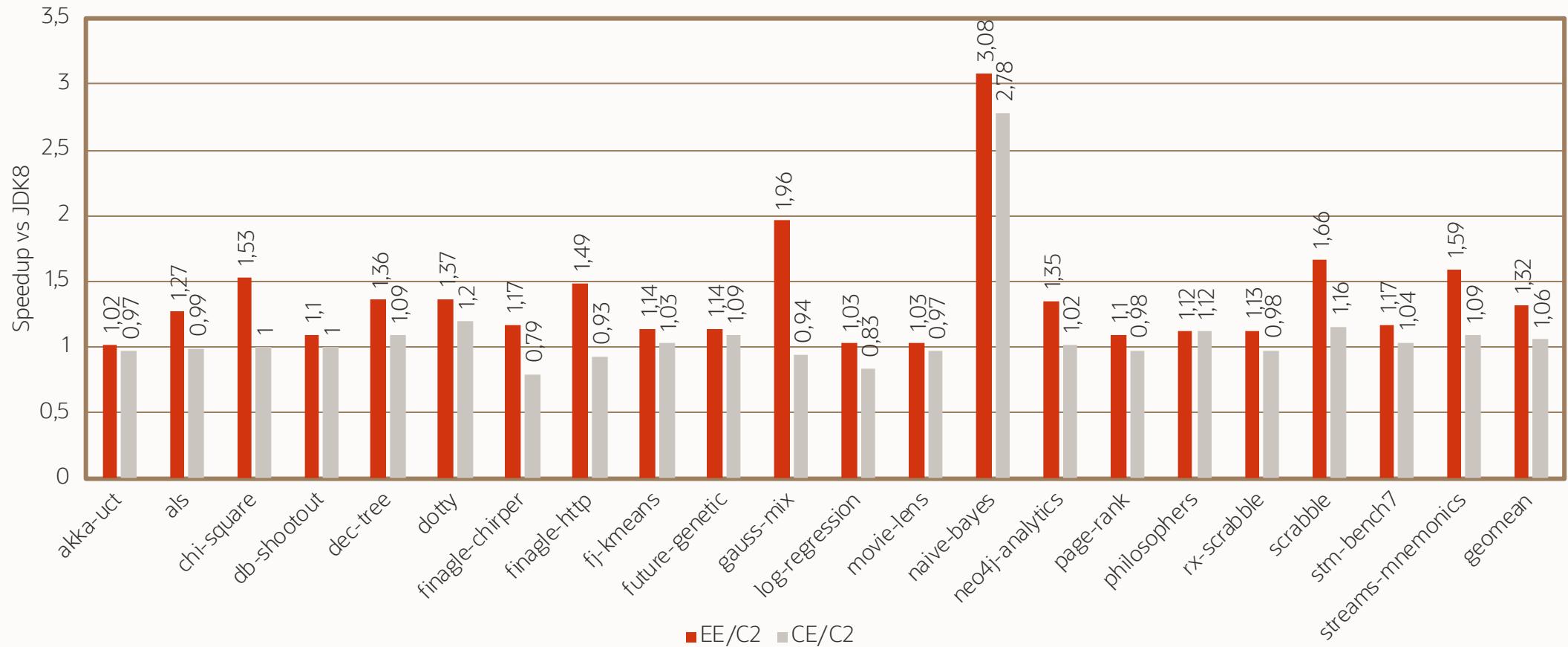
Speedup, higher is better



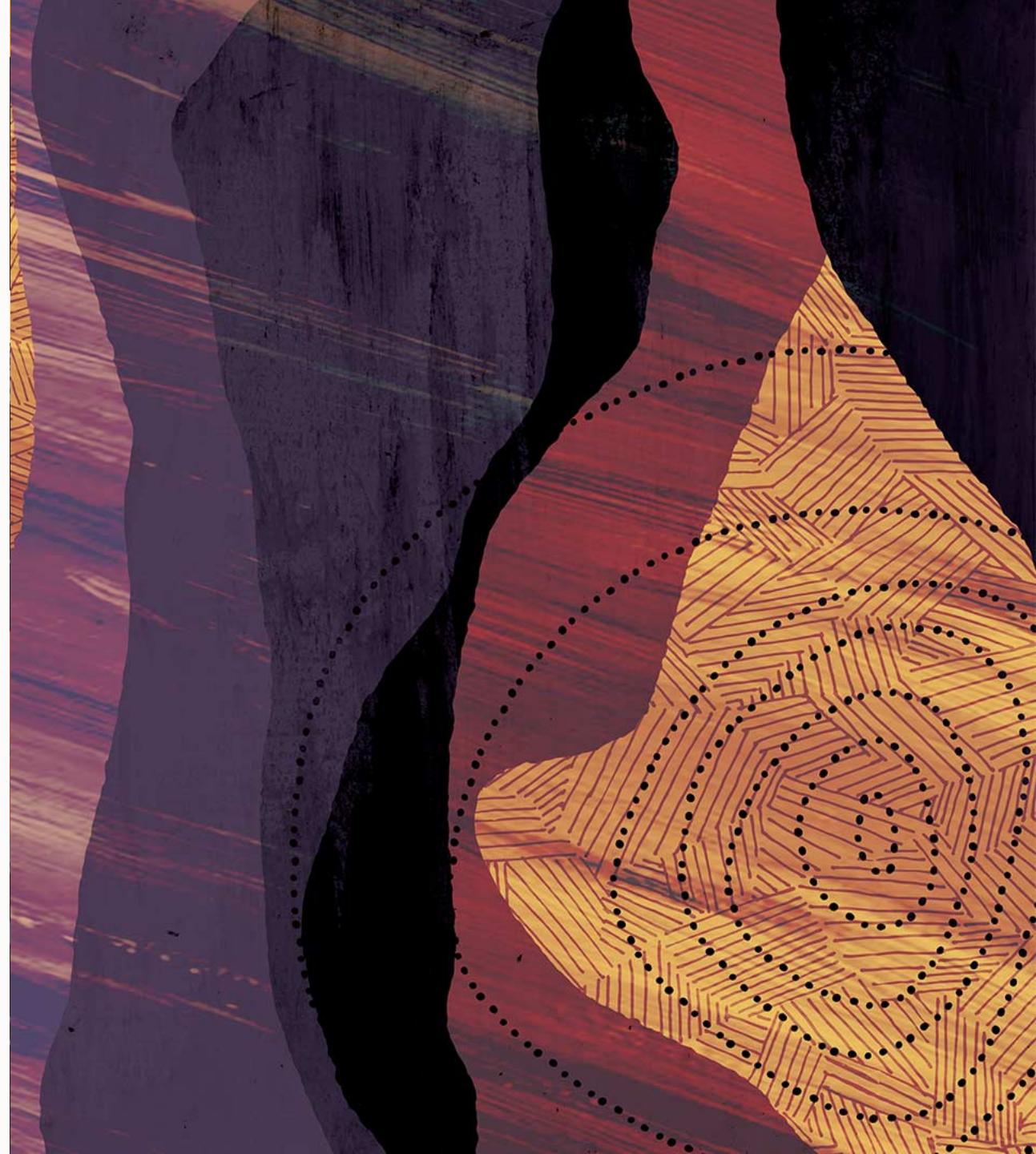
Performance relative to:

HotSpot/Server, HotSpot/Server running JRuby, GNU R, LLVM AOT compiled, V8

GraalVM JIT Performance: Renaissance.dev



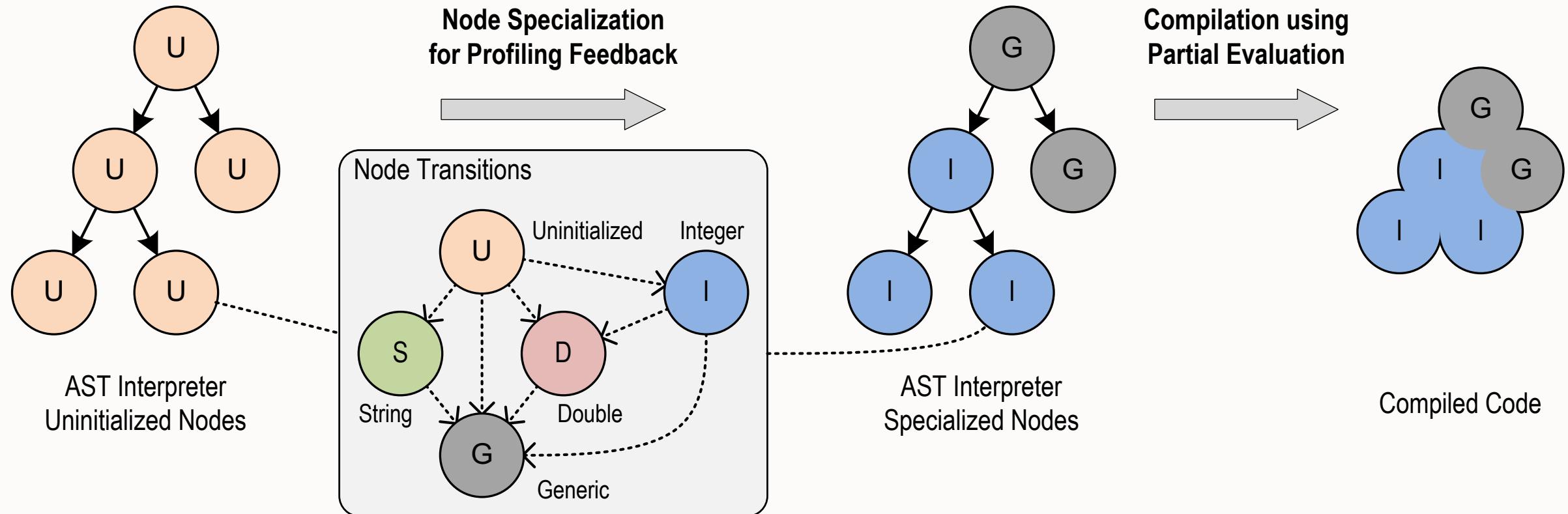
Truffle



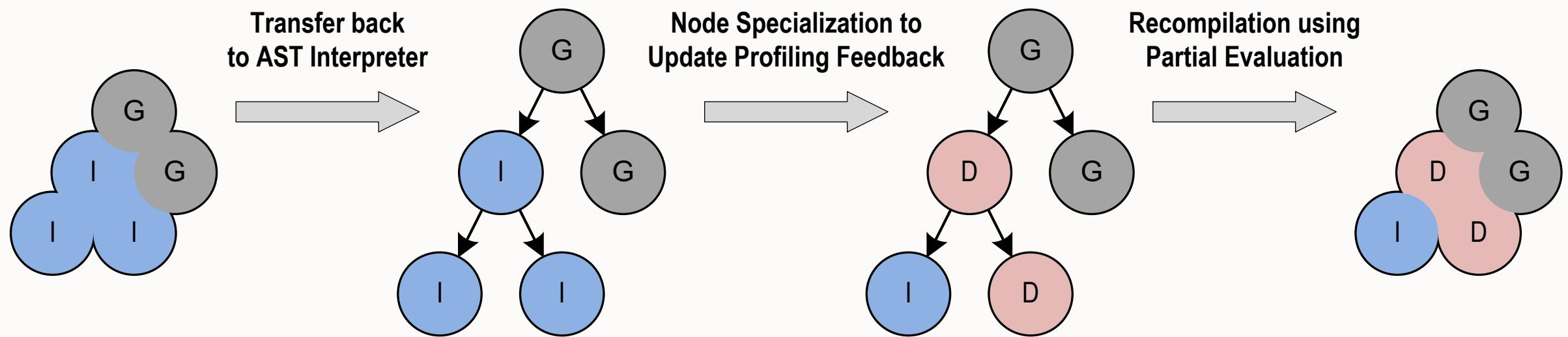
Truffle

- Self-optimizing abstract tree interpreter
 - Rewrites itself
- The simplest way to implement a language
- A Java library
 - Node classes
 - Execute methods

Speculate and Optimize ...



... and Transfer to Interpreter and Reoptimize!



Truffle

```
@GenerateUncached
public abstract class IsFrozenNode extends RubyBaseWithoutContextNode {

    public static IsFrozenNode create() {
        return IsFrozenNodeGen.create();
    }

    public abstract boolean execute(Object object);

    @Specialization
    protected boolean isFrozen(boolean object) {
        return true;
    }

    @Specialization
    protected boolean isFrozen(int object) {
        return true;
    }

    @Specialization
    protected boolean isFrozen(long object) {
        return true;
    }

    @Specialization
    protected boolean isFrozen(double object) {
        return true;
    }

    @Specialization
    protected boolean isFrozen(
        DynamicObject object,
        @Cached ReadObjectFieldNode readFrozenNode) {
        return (boolean) readFrozenNode.execute(object, Layouts.FROZEN_IDENTIFIER, false);
    }
}
```

Polyglot

Use cases

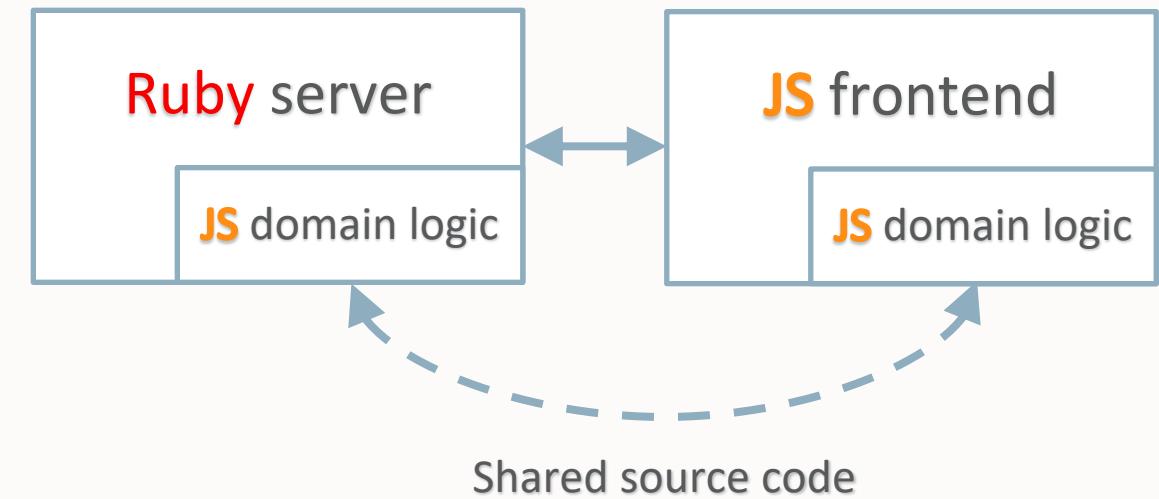


Polyglot use-cases

- Missing library
 - You are not limited to libraries only written in your language of choice
- Legacy code
 - Migration of legacy project from language A to B can be gradual
- Sharing code
 - Share domain logic between frontend (JS) and server code (Java, Ruby, ...)
- Execute on GPU
- User scripting
 - Run your user provided scaling rules
 - Run stored functions in DB
 - And more

Web application demo – Ruby

- JavaScript frontend
- Ruby backend
 - In-memory database
- **Shared** domain logic written in JavaScript
 - No duplication
 - Identical behavior
 - The same validation code runs in the webserver and on the server



Web application demo – Ruby

- Single table of people
 - With first name, last name, and age
- A form to add new person
 - With immediate validations

People

First name	Last name	Age
John	Doe	50
Adam	Bojacny	25

New person form

first name cannot be empty

last name cannot be empty

age has to be integer

Add person Add invalid person

[Quit the application.](#)

Web application demo – The shared JS domain

```
function Person(firstName, lastName, age) {
    this.firstName = firstName;
    this.lastName = lastName;
    this.age = Number.parseInt(age);
    this.errors = {};

    if (!this.firstName) { this.errors.firstName = 'cannot be empty'; }
    if (!this.lastName) { this.errors.lastName = 'cannot be empty'; }
    // ... age validation skipped (Integer, -1 > age < 151)

    Object.freeze(this);
}

Person.prototype.isValid = function () {
    return Object.keys(this.errors).length;
};
```

Web application demo – The Ruby server

```
class PolyglotApp < Sinatra::Base
  # ... configuration skipped

  Polyglot.eval_file('js',
    File.join(File.dirname(__FILE__), 'public', 'person.js'))
  JsPerson = Polyglot.eval('js', 'Person')

  PEOPLE_DB = []
  def self.initialize_db
    PEOPLE_DB.clear
    PEOPLE_DB.push JsPerson.new('John', 'Doe', 50)
  end

  initialize_db
```

Load and evaluate JS file

Get the defined class by evaluation and store it in Ruby constant

Create new instance of Person defined in JS

Web application demo – The Ruby server

```
get '/' do
  File.read(File.join(self.class.public_folder, 'index.html'))
end

def js_person_as_hash(js_user)
  { firstName: js_user[:firstName],
    lastName: js_user[:lastName],
    age: js_user[:age] }
end

get '/people.json' do
  data = PEOPLE_DB.map { |js_user| js_person_as_hash(js_user) }
  data.to_json
end
```

Read fields from a JS object

Web application demo – The Ruby server

```
post '/person.json' do
  person_json      = JSON.parse request.body.read
  js_person        = JsPerson.fromObject OpenStruct.new person_json
  js_person_is_invalid = js_person.invalid

  if js_person_is_invalid > 0
    [403, 'Invalid user.']
  else
    PEOPLE_DB.push js_person
    [200, 'Ok']
  end
end
```

Call function on a JS object.
Creates new instance.

Call function on a JS object.
Checks the validity.

Polyglot

How does it work



Language independent interoperability messages

- Messages
 - Array like
 - HAS_SIZE, GET_SIZE, READ, WRITE
 - Members
 - HAS_KEYS, KEYS, INVOKE, READ, WRITE,
 - etc.
- Each language defines its behavior for these messages only
 - Otherwise there would be language combination explosion
 - Otherwise adding new languages would be hard

Idiomatic syntax translates to interoperability messages

```
js_function = Polyglot.eval('js', <<-JS)
  function(arg) { return [1,2,3,5,8].slice(0,arg); }
```

JS

```
js_array      = js_function.call 4          # EXECUTE
puts js_array.size # => 4                  # GET_SIZE
puts js_array[3]  # => 5                  # READ
```

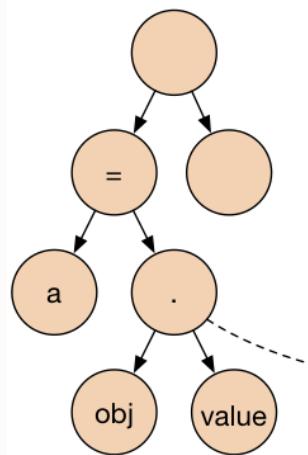
```
JavaIntArray     = Java.type 'int[]'
java_int_array   = JavaIntArray.new 1 # NEW
puts java_int_array.size # => 1      # GET_SIZE
java_int_array[0] = 42               # WRITE
puts java_int_array[0]    # => 42      # READ
```

Explicit message usage when necessary

- It's not possible to get a size of an array in C
- Therefore there is always a way to call the messages explicitly
 - `bool polyglot_has_array_elements(const void *value);`
 - `uint64_t polyglot_get_array_size(const void *array);`
 - `array[idx];`

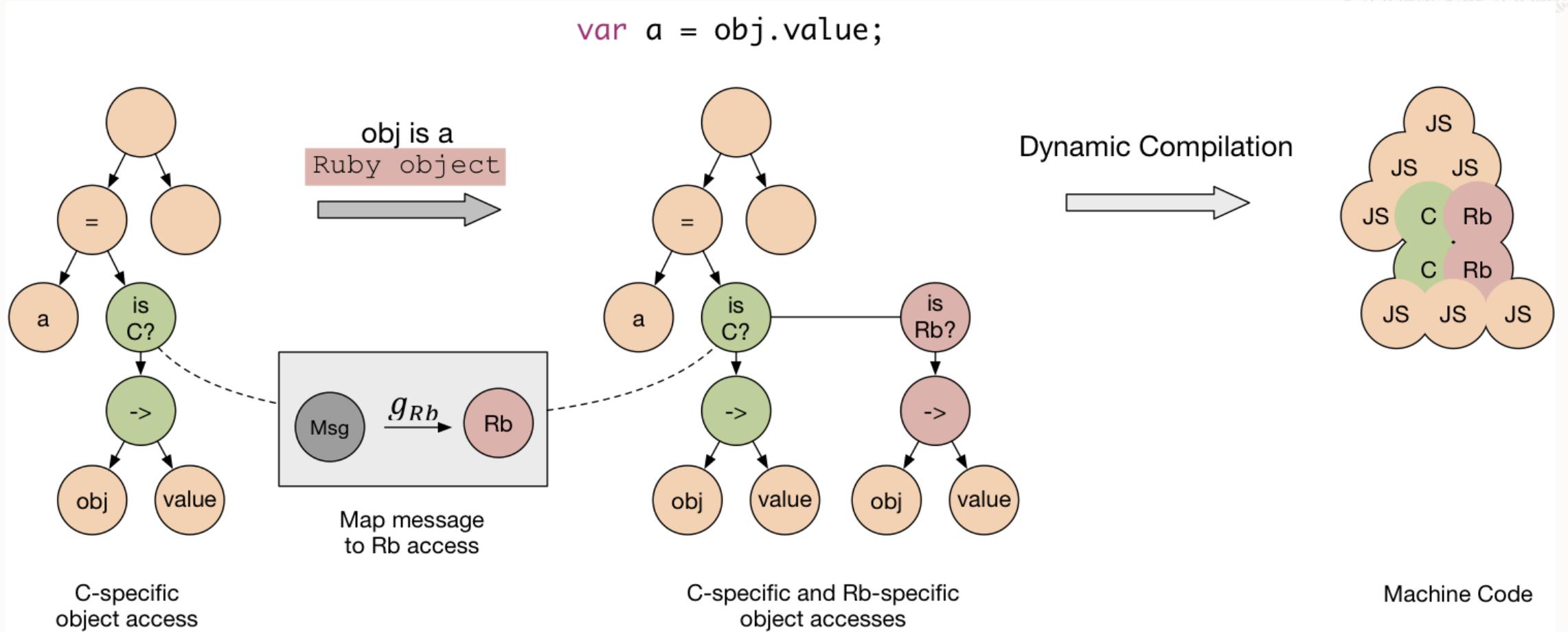
High-Performance Language Interoperability

```
var a = obj.value;
```



JS-specific
object access

High-Performance Language Interoperability



Polyglot performance

- Uses the same AST rewriting framework
- Optimizes through the language barrier
- The foreign language nodes make no difference to Truffle
- It can optimize as well as if it were nodes from the host language
 - Same inline cache principle as for regular method calls

Polyglot

More examples



Using Numpy from Java – Calling into Numpy

```
try (Context context = Context.newBuilder()
    .option("python.PythonPath", "/path/to/numpy-1.16.4-py3.7-macosx-10.14-x86_64.egg")
    .allowAllAccess(true).build()) {
    Value geomean = context.eval("python", "import numpy\n" + "import math\n" +
        "lambda x: math.pow(numpy.array(x).prod(), 1/len(x))");
}

anonymous function that calculates the geometric
mean using numpy and the math module

double[] values = new double[] { 1, 5, 8, 3, 5, 8, 8, 7, 5, 6 };
double mean = geomean.execute(values).asDouble();
System.out.println(mean);
// 4.905181164183902

}
```

Using CUDA to Access Nvidia GPUs

- Different binding libraries / APIs for CUDA in different programming languages
- Varying set of supported features
- Translation to/from unmanaged environment (in Java, C#, Python, etc.)

Python	Numba, cuPy, PyCUDA
Java	JCuda, jCUDA, CUDA4J
C / C++	CUDA C/C++ (language extension)
R	gpuR, indirectly through Rcpp
JS	gpu.js (WebGL), node-cuda, cuda-ts
C#	Hybridizer, ManagedCUDA, Alea GPU, ILGPU
Ruby	RbCUDA

Using grCUDA to Access Nvidia GPUs

- Efficient exchange of data between host language and GPU without burdening the programmer
- Expose GPU resources in ways that are native in the host language, e.g., as arrays
- Allow programmers to invoke existing GPU code from their host language
- Allow programmers to define new GPU kernels on the fly
- Polyglot interface: uniform bindings across all programming languages

- Implemented as a “Truffle Language”
(although “CUDA” is a platform, not a language)

- Developed by NVIDIA in collaboration
with Oracle Labs
- BSD 3-clause license



Creating and Using Device Arrays (Python)

```
import polyglot

# Get constructor function as callable
DeviceArray = polyglot.eval(language='grcuda', string='DeviceArray')

# Create 1D device array that can hold 1000 int values
dev_int_arr = DeviceArray('int', 1000)

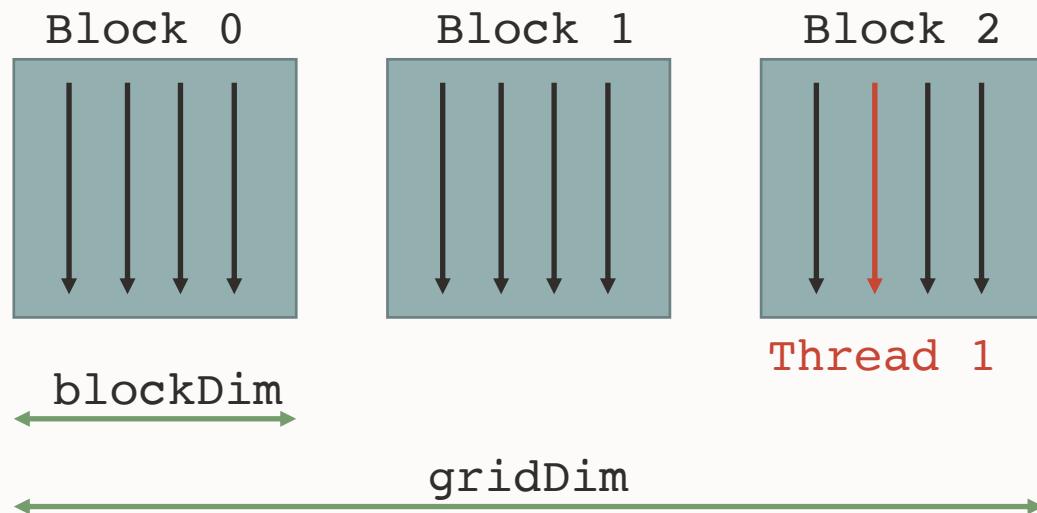
# Create 2D device array that can hold 1000 x 100 float values
dev_float_2d = DeviceArray('float', 1000, 100)

# Setting array elements
for i in range(len(dev_int_array)):
    dev_int_arr[i] = i
for i in range(len(dev_float_2d)):
    for j in range(len(dev_float_2d[i])):
        dev_float_2d[i][j] = i + j
```

As fast or faster than native language constructs (e.g., R vectors) because of simpler semantics (e.g., no NA values)

GPU Kernels in CUDA C++

```
__global__ void inc_kernel(float *out_arr, const float *in_arr, size_t num_elements) {  
    for (auto idx = blockIdx.x * blockDim.x + threadIdx.x; idx < num_elements;  
        idx += blockDim.x * blockDim.x) {  
        out_arr[idx] = in_arr[idx] + 1;  
    }  
}
```



Number of blocks and threads can
be configured in 3 dimensions (x, y, z)

Launching GPU Kernels (JS)

```
const DeviceArray = Polyglot.eval('grcuda', string='DeviceArray')
const N = 1000
const in_arr = DeviceArray('float', N)
const out_arr = DeviceArray('float', N)
for (let i = 0; i < N; i++)
  in_arr[i] = i
const code = '__global__ void inc_kernel(...)' ...
const buildkernel = Polyglot.eval('grcuda', string='buildkernel')
const incKernel = buildkernel(code, 'inc_kernel', 'pointer', pointer, uint64')

// Launch kernel in grid consisting of 160 blocks with 256 threads each
incKernel(160, 256)(out_arr, in_arr, N)

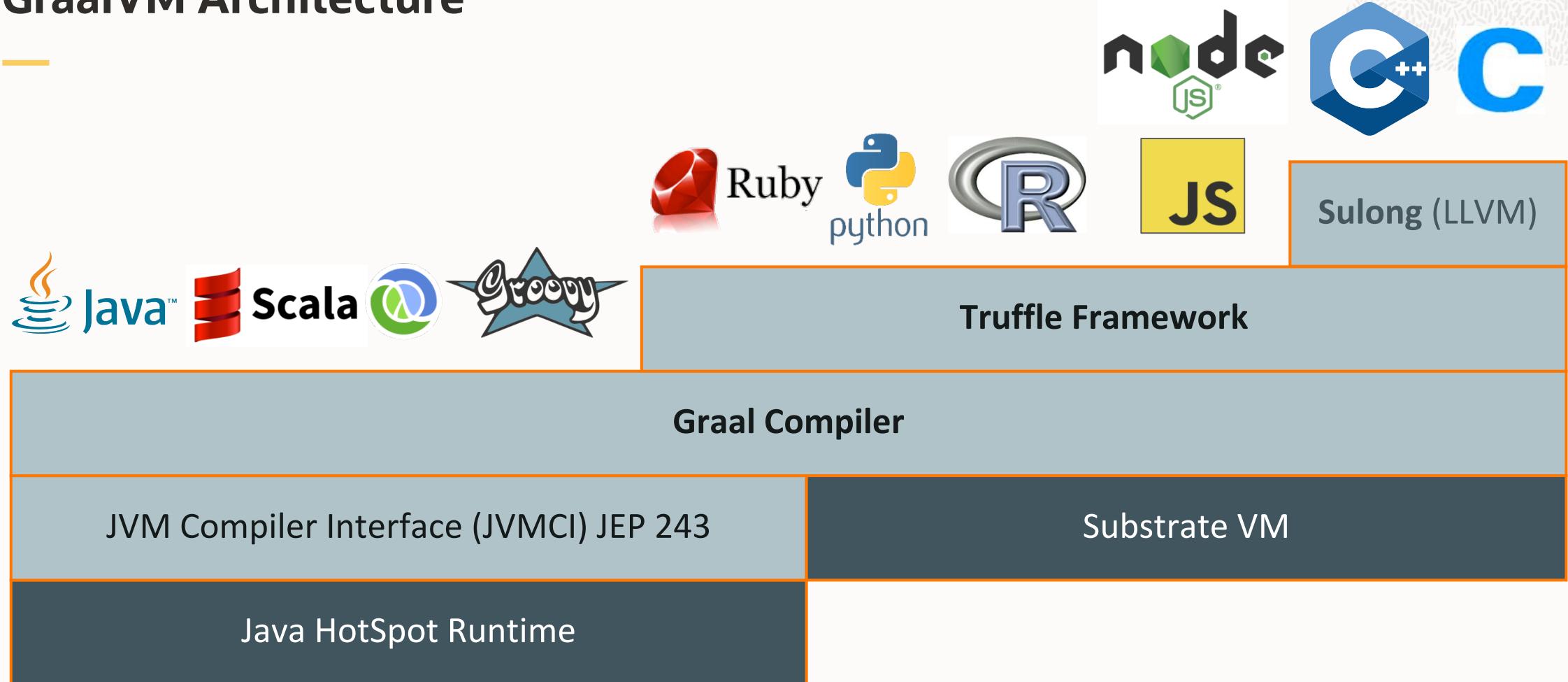
for (let i = 0; i < 10; i++) {
  console.log(out_arr[i]);
}
```

Device arrays `in_arr` and `out_arr` can be passed to GPU kernel

Embedding and native image generation



GraalVM Architecture



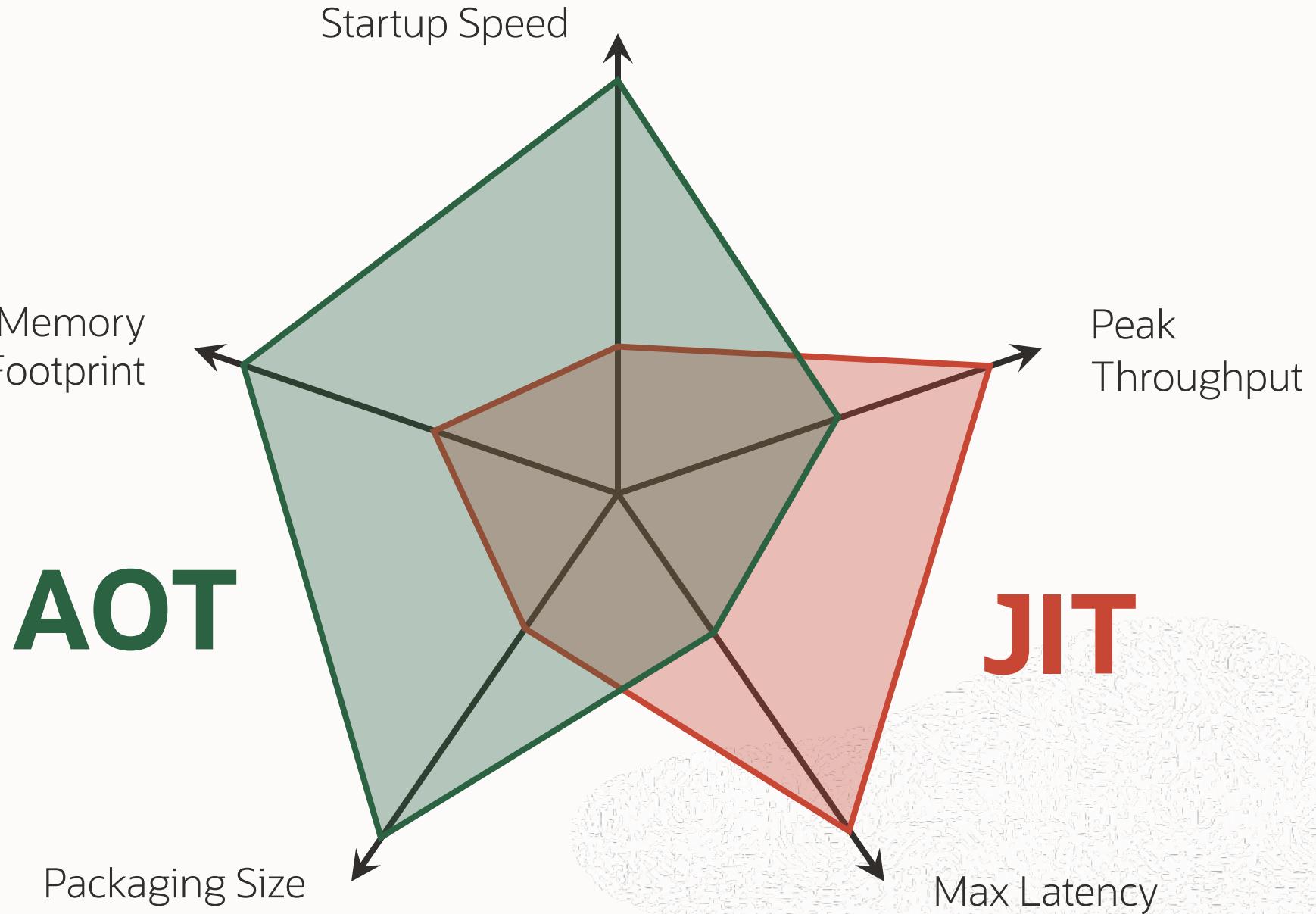


\$ java MyMainClass
OpenJDK™

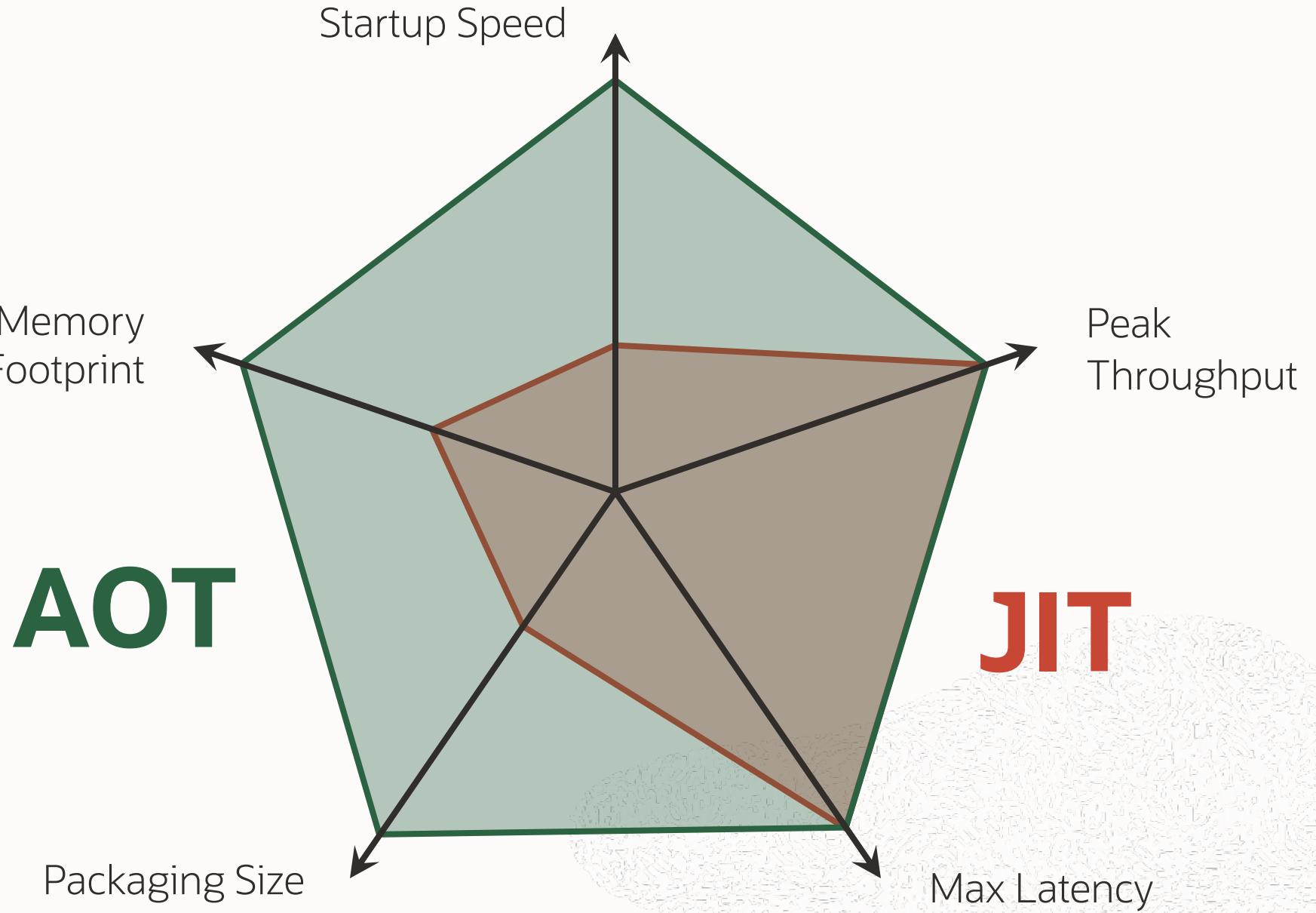
\$ native-image MyMainClass
\$./mymainclass



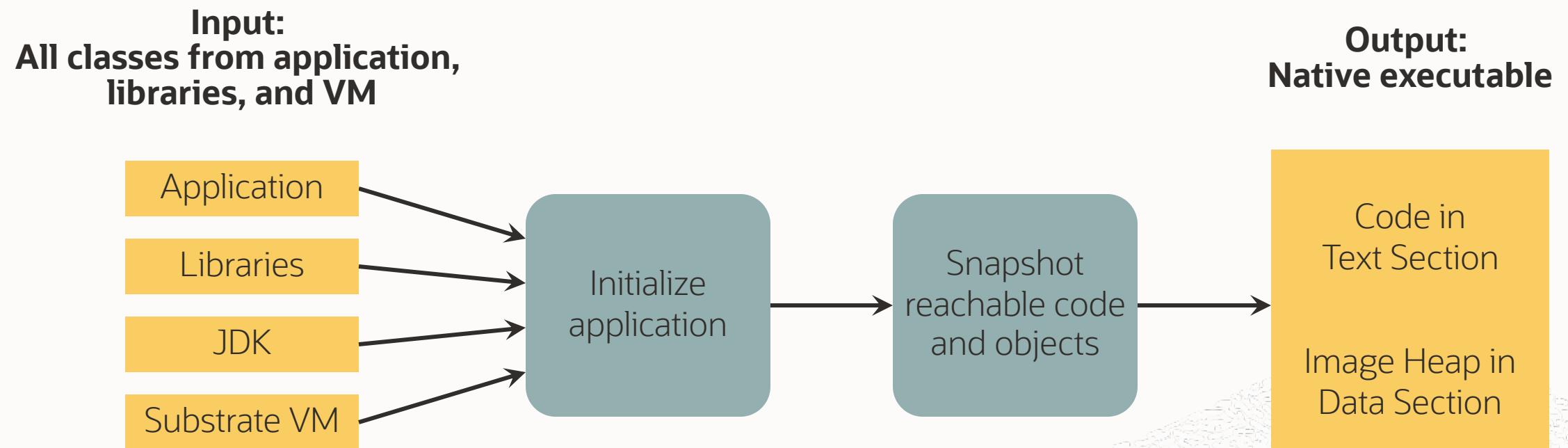
Currently



Goal

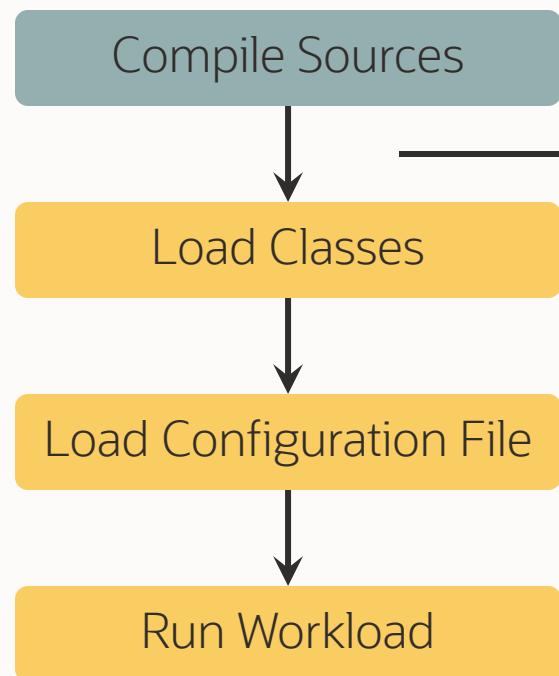


Native Image: Principle

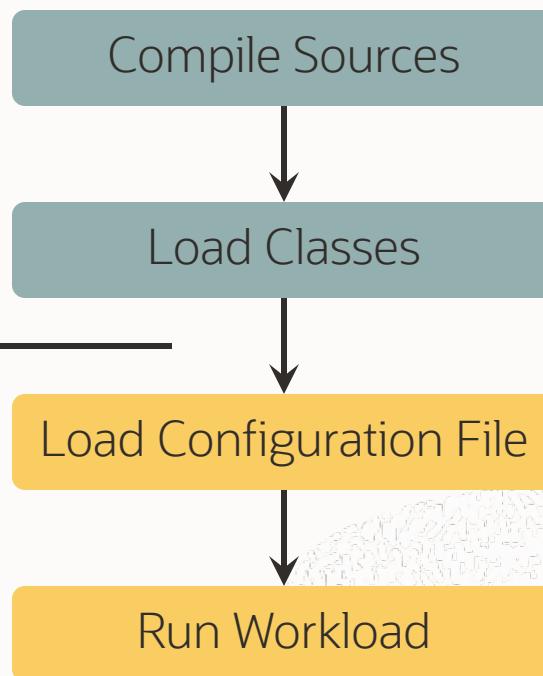


Benefits of the Image Heap

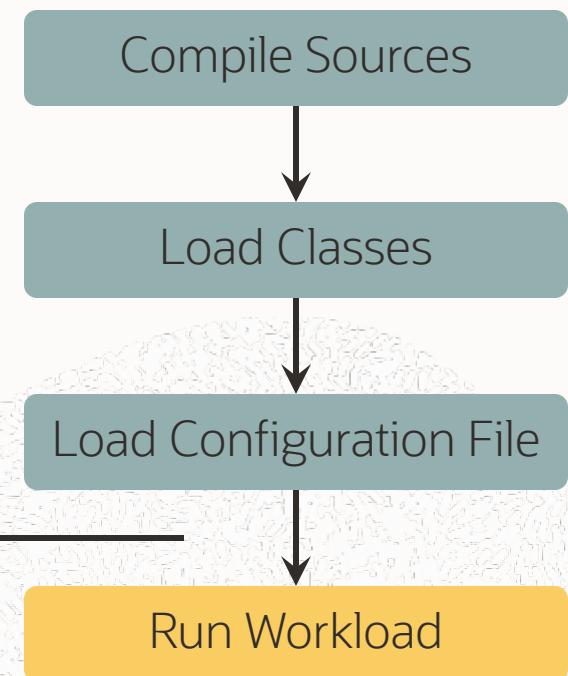
Without GraalVM Native Image



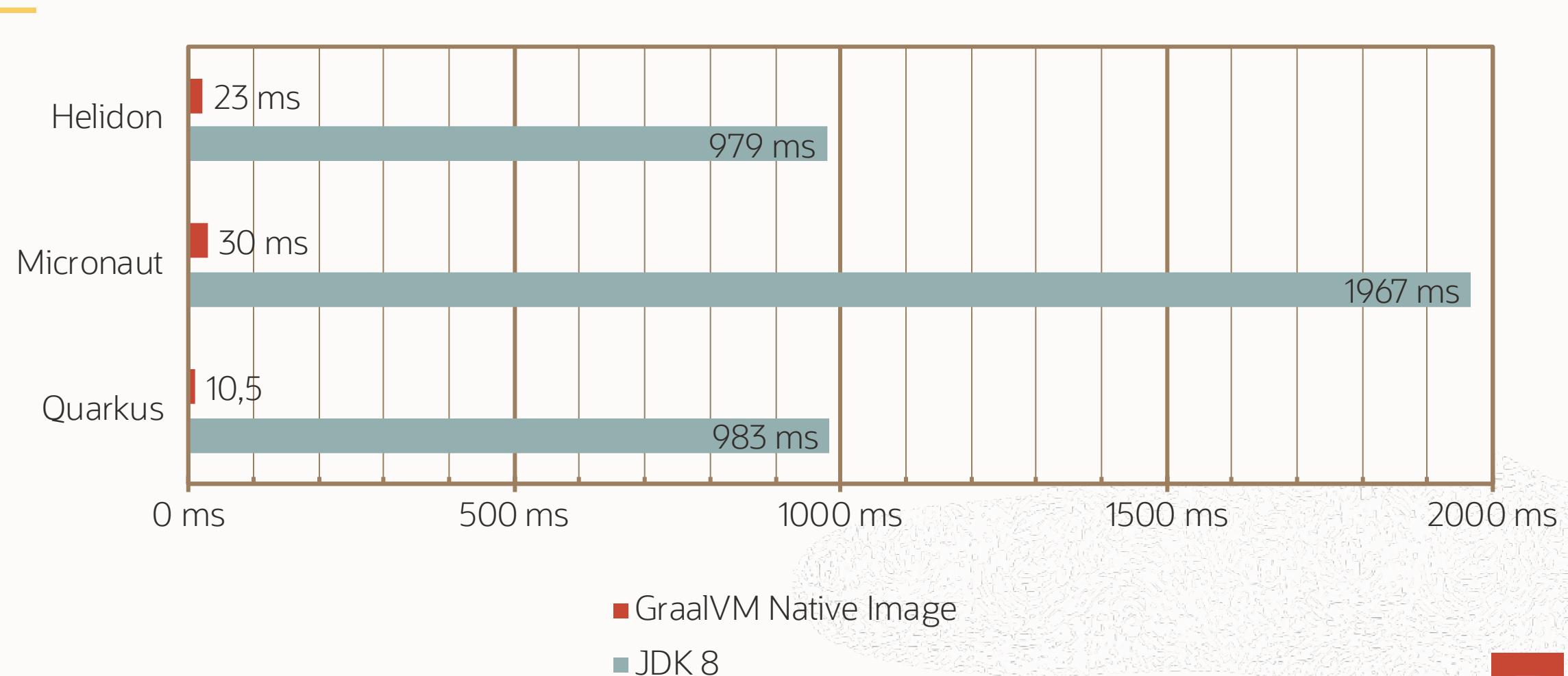
GraalVM Native Image (default)



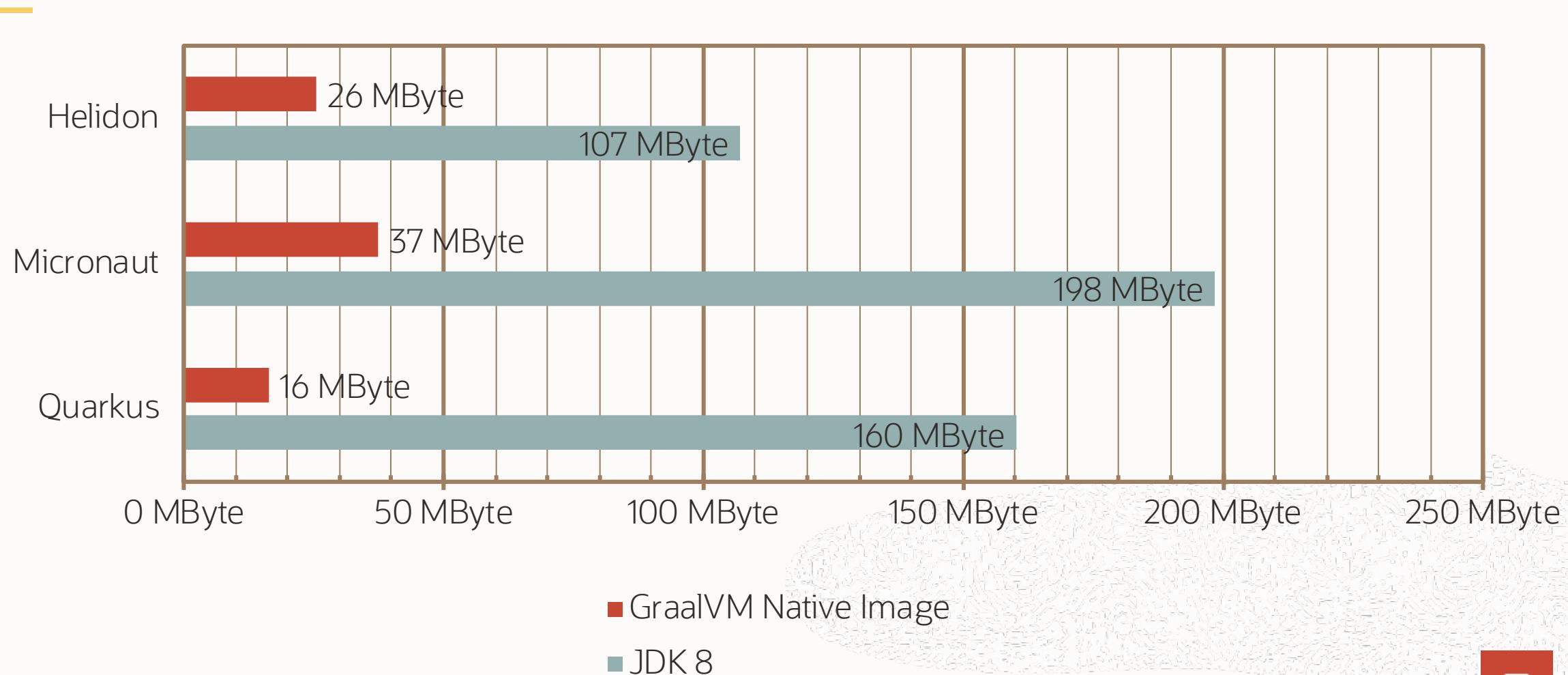
GraalVM Native Image: Load configuration file at build time



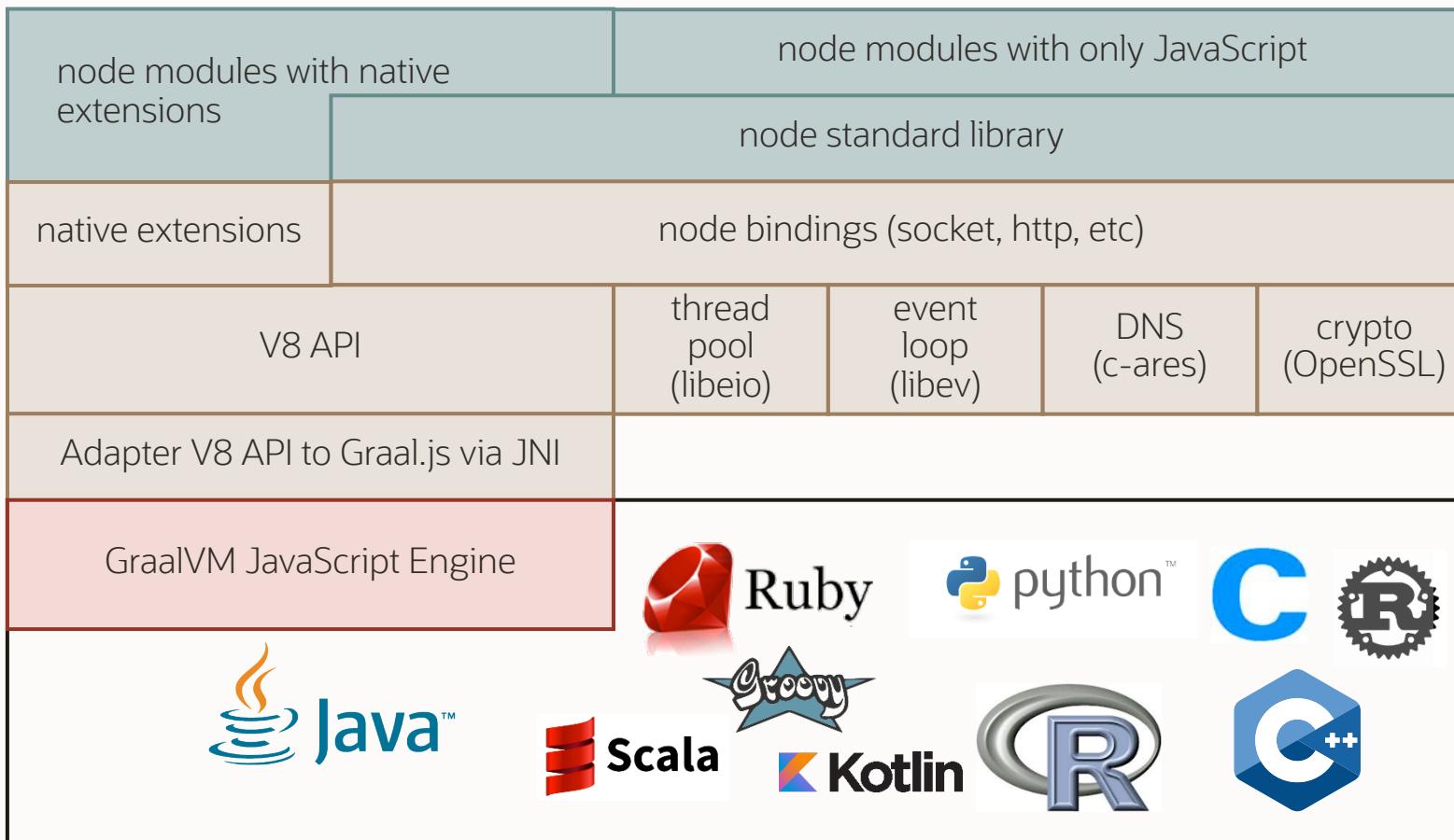
Startup Time of Java Microservice Frameworks



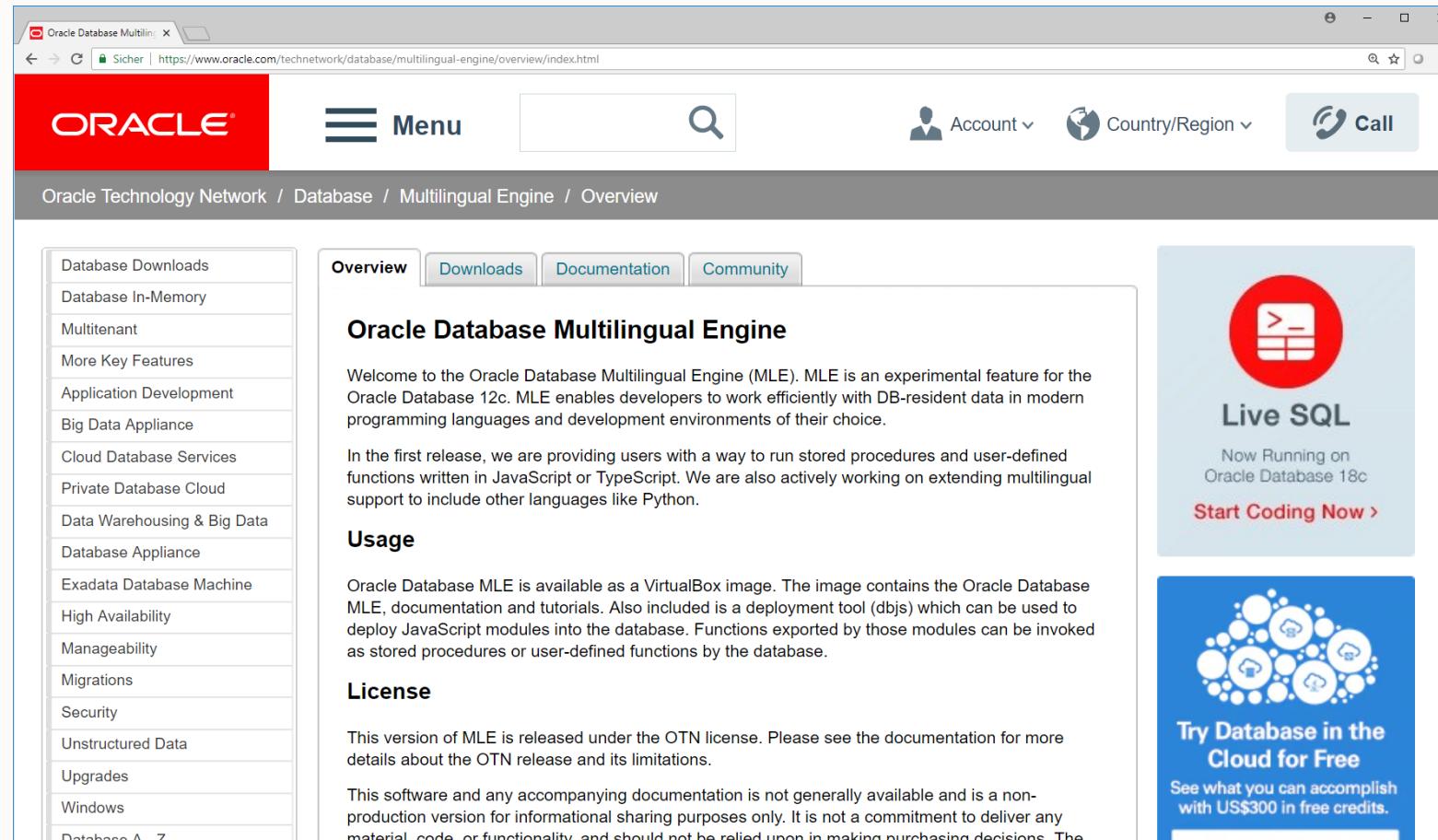
Memory Footprint



Architecture of Node.js running via GraalVM



Embedding Graal: MLE (OracleDB)



The screenshot shows the Oracle Database Multilingual Engine (MLE) overview page. The page has a navigation bar with links for Database Downloads, Database In-Memory, Multitenant, More Key Features, Application Development, Big Data Appliance, Cloud Database Services, Private Database Cloud, Data Warehousing & Big Data, Database Appliance, Exadata Database Machine, High Availability, Manageability, Migrations, Security, Unstructured Data, Upgrades, Windows, and Database A-Z. The main content area features tabs for Overview, Downloads, Documentation, and Community. The Overview tab is selected, displaying information about the Oracle Database Multilingual Engine. It includes sections for Usage and License. There are also promotional banners for "Live SQL" and "Try Database in the Cloud for Free".

```
module.exports.helloworld =  
  function () {  
    return "Hello World";  
}  
  
export function helloworld():string;  
shell> dbjs deploy helloworld.js  
sql> SELECT helloworld() from dual;
```

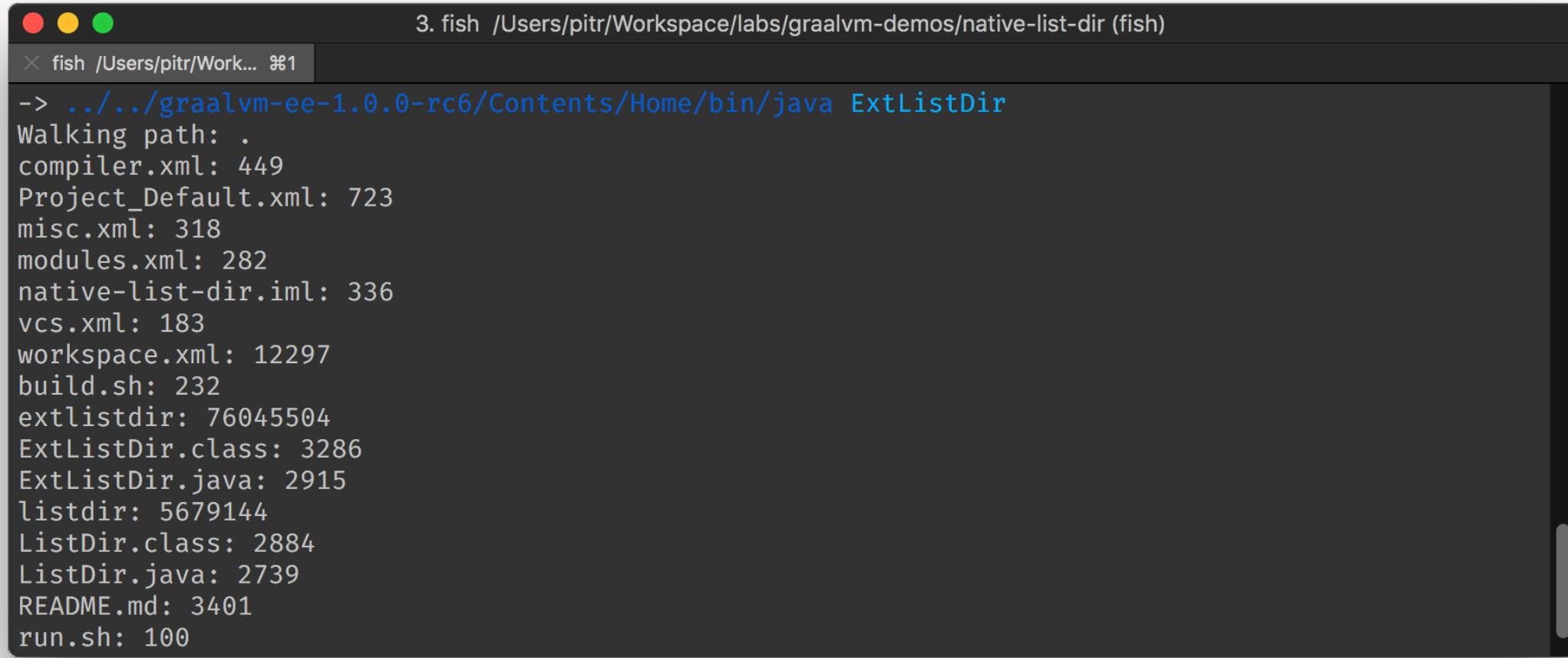
<https://www.oracle.com/technetwork/database/multilingual-engine/>

Embedding small JS snippet

```
import org.graalvm.polyglot.*;
// ...

public static void main(String[] args) throws java.io.IOException {
    System.out.println("Walking path: " + Paths.get(args.length > 0 ? args[0] : "."));
    final Context context = Context.create("js");
    final Value jsFn = context.eval("js",
        "function(name, size) { return name + ': ' + size}");
    try (Stream<Path> paths = Files.walk(Paths.get("."))) {
        paths.filter(Files::isRegularFile).
            forEach((Path p) -> {
                File f = p.toFile();
                Value v = jsFn.execute(f.getName(), f.length());
                System.out.println(v);
            });
    }
}
```

Embedding small JS snippet



A screenshot of a macOS terminal window titled "fish /Users/pitr/Workspace/labs/graalvm-demos/native-list-dir (fish)". The window shows the command "fish /Users/pitr/Work..." followed by a list of files and their sizes. The files listed are:

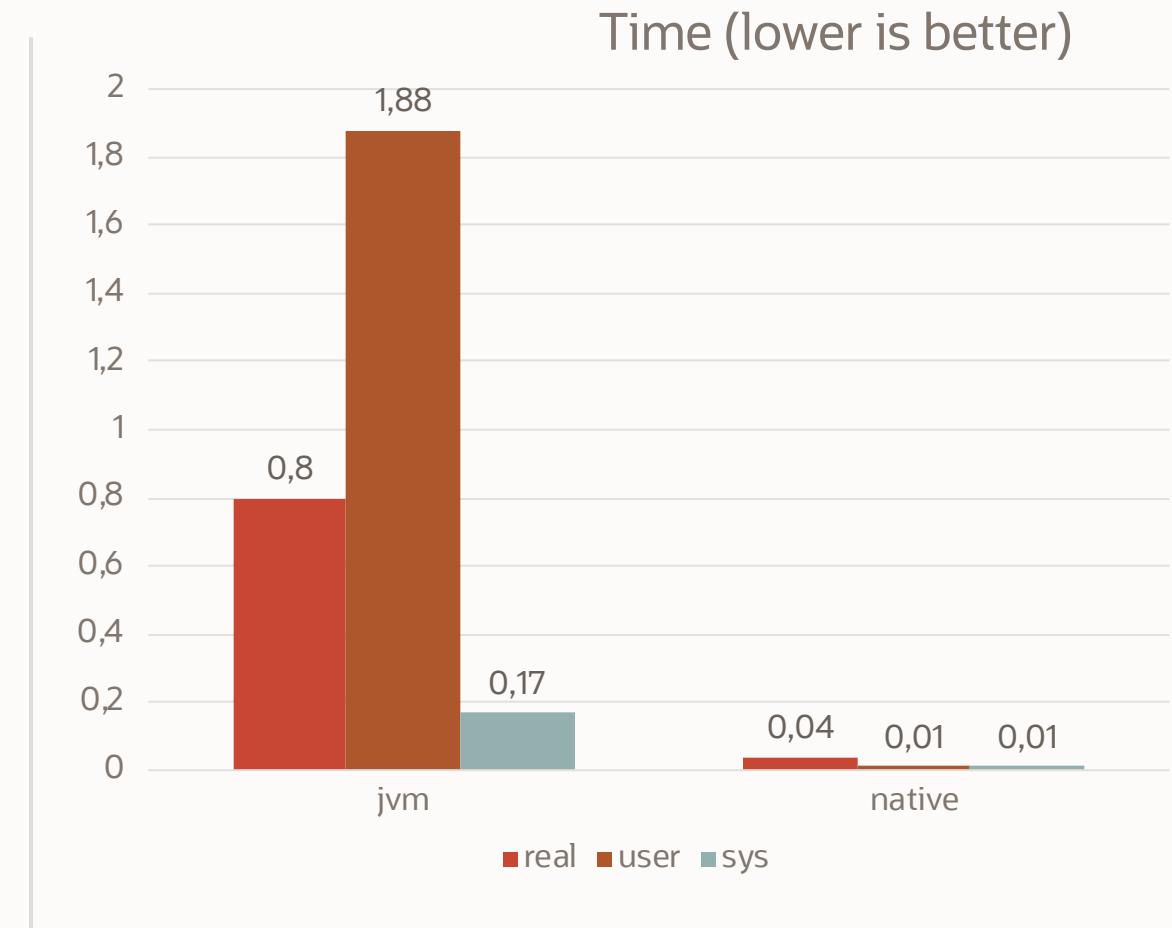
- > ../../graalvm-ee-1.0.0-rc6/Contents/Home/bin/java ExtListDir
- Walking path: .
- compiler.xml: 449
- Project_Default.xml: 723
- misc.xml: 318
- modules.xml: 282
- native-list-dir.iml: 336
- vcs.xml: 183
- workspace.xml: 12297
- build.sh: 232
- extlistdir: 76045504
- ExtListDir.class: 3286
- ExtListDir.java: 2915
- listdir: 5679144
- ListDir.class: 2884
- ListDir.java: 2739
- README.md: 3401
- run.sh: 100

Embedding small JS snippet

```
3. fish /Users/pitr/Workspace/labs/graalvm-demos/native-list-dir (fish)
× fish /Users/pitr/Work... #1
-> ../../graalvm-ee-1.0.0-rc6/Contents/Home/bin/native-image --language:js ExtListDir
Build on Server(pid: 13992, port: 51063)*
[extlistdir:13992]    classlist:  4,906.11 ms
[extlistdir:13992]        (cap):  1,573.08 ms
[extlistdir:13992]        setup:   9,185.80 ms
Warning: Detected unnecessary RecomputeFieldValue.ArrayBaseOffset com.oracle.svm.core.jdk.Target_java_nio_DirectByteBuffer.arrayBaseOffset substitution field for java.nio.DirectByteBuffer.arrayBaseOffset. The annotated field can be removed. This ArrayBaseOffset computation can be detected automatically. Use option -H:+UnsafeAutomaticSubstitutionsLogLevel=2 to print all automatically detected substitutions.
[extlistdir:13992]    (typeflow): 24,695.47 ms
[extlistdir:13992]    (objects): 51,975.49 ms
[extlistdir:13992]    (features): 5,307.08 ms
[extlistdir:13992]    analysis: 99,034.79 ms
7022 method(s) included for runtime compilation
[extlistdir:13992]    universe:  4,572.60 ms
[extlistdir:13992]    (parse):  8,670.41 ms
[extlistdir:13992]    (inline): 13,117.23 ms
[extlistdir:13992]    (compile): 70,414.10 ms
[extlistdir:13992]    compile: 124,462.72 ms
[extlistdir:13992]    image: 14,027.42 ms
[extlistdir:13992]    write:   5,549.62 ms
[extlistdir:13992]    [total]: 296,971.65 ms
```

Embedding small JS snippet

- time graalvm-ee-1.0.0-rc6/.../bin/java ExtListDir
- time ./extlistdir

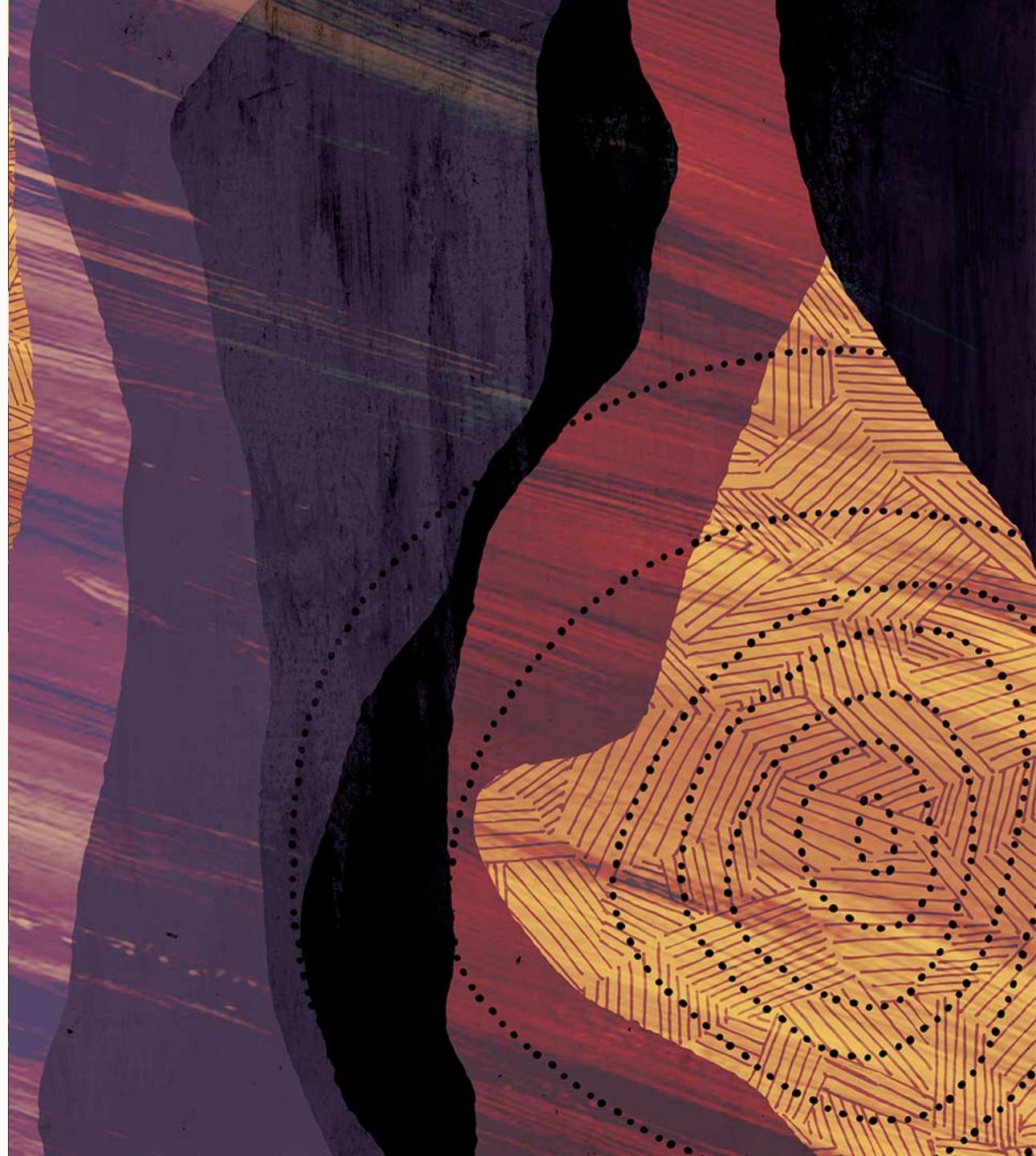


Embedding small JS snippet

- extlistdir is just 72MB
- It is a native image without dependencies on JVM
- It has the JS engine in it
- It executed almost as fast as native ls command

Instrumentation and Tooling

Debugging



Debugging with Chrome DevTools

- Run the server with *--inspect* option
 - ruby --polyglot --experimental-options --ruby.single-threaded --inspect app.rb
- Follow the instructions and open the *chrome-devtools:* link
 - Step through the languages

The screenshot shows the Chrome DevTools Sources tab for a Ruby application. The left pane displays the file structure and the content of `app.rb`. The right pane shows the call stack and the current execution context.

Filesystem:

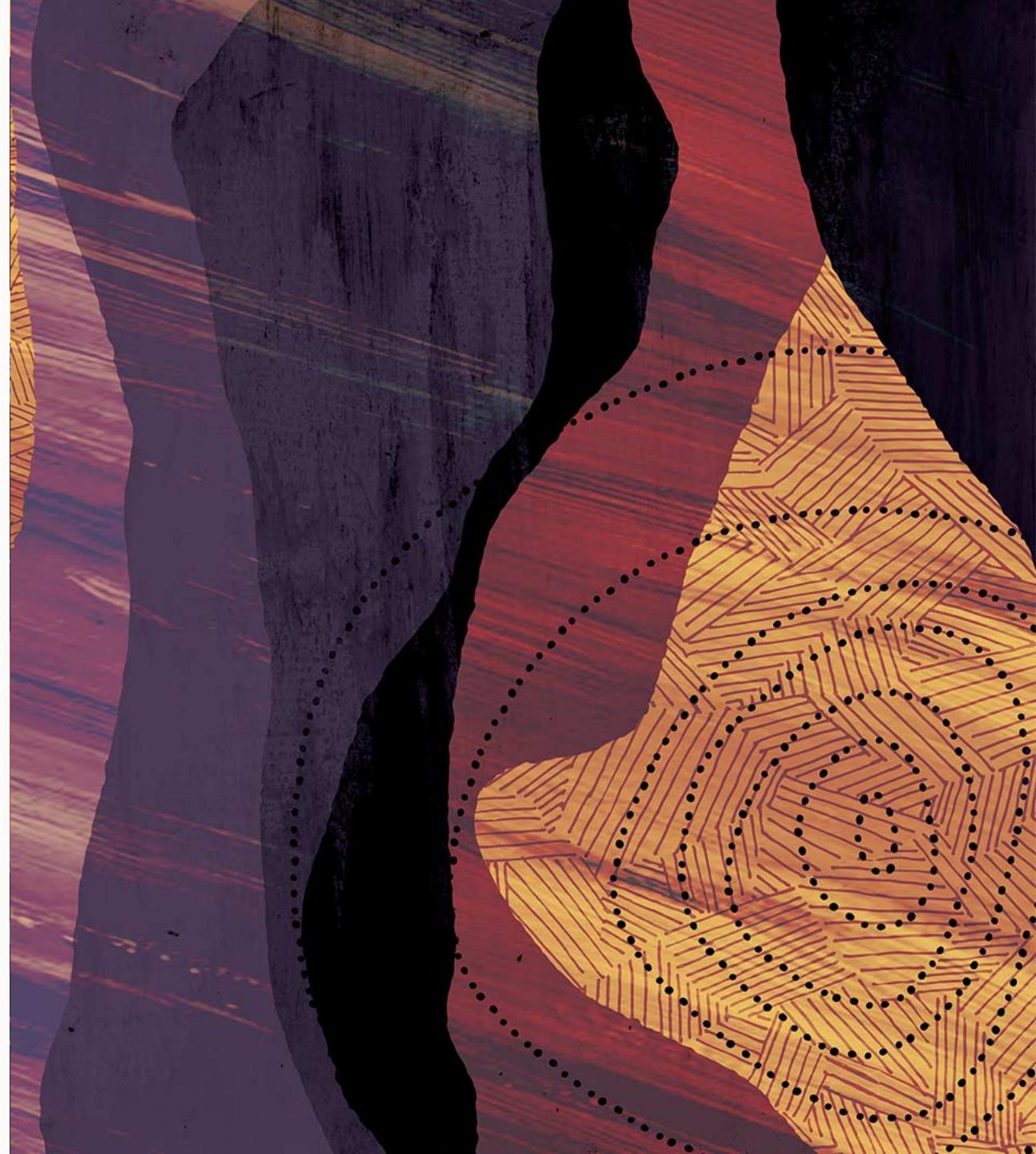
- + Add folder to workspace
- demo
 - public
 - Gemfile
 - Gemfile.lock
 - app.rb
 - demo.iml
 - examples.rb
 - single_threaded.rb
 - optcarrot
 - truffleruby

Code Editor (app.rb):

```
1 require 'pp'
2 require 'sinatra/base'
3 require_relative 'single_threaded'
4
5 # noinspection RubyConstantNamingConvention,RubyParenthesesAfterMethodCallInspection
6 class PolyglotApp < Sinatra::Base
7
8   enable :static
9   set :server, 'webrick'
10  disable :logging
11
12  if defined? Polyglot
13    person_definition_path = File.join File.dirname(__FILE__),
14                                         'public', 'person.js'
15    Polyglot.eval_file('js', person_definition_path)
16    JsUser = Polyglot.eval('js', 'User')
17    PEOPLE_DB = [JsUser.new('John', 'Doe', 50)]
18  else
19    PEOPLE_DB = []
20  end
21
22  def js_user_as_hash(js_user)
23    [:firstName, :lastName, :age].reduce({}) do |hash, key|
24      hash.update key => js_user[key]
25    end
26  end
27
28  get '/' do
29    File.read(File.join(self.class.public_folder, 'index.html'))
30  end
31
32  get '/people.json' do
33    data = PEOPLE_DB.map do |js_user|
34      js_user_as_hash(js_user)
35    end
36    data.to_json
37  end
38
39  post '/person.json' do
40    request.body.rewind # in case someone already read it
41    person = JSON.parse request.body.read
42
43    if defined? Polyglot
44      js_user = JsUser.fromObject OpenStruct.new person
45      if !js_user.isValid!
46    end
47  end
48
49  get '/polyglot' do
50    "Polyglot is alive"
51  end
52
53  run Rack::Head
54
55  error 500 do
56    "Internal Server Error"
57  end
58
59  not_found do
60    "Not Found"
61  end
62
63  not_modified do
64    "Not Modified"
65  end
66
67  head :not_implemented do
68    "Not Implemented"
69  end
70
71  head :bad_gateway do
72    "Bad Gateway"
73  end
74
75  head :internal_server_error do
76    "Internal Server Error"
77  end
78
79  head :method_not_allowed do
80    "Method Not Allowed"
81  end
82
83  head :not_acceptable do
84    "Not Acceptable"
85  end
86
87  head :proxy_authentication_required do
88    "Proxy Authentication Required"
89  end
90
91  head :request_timeout do
92    "Request Timeout"
93  end
94
95  head :internal_server_error do
96    "Internal Server Error"
97  end
98
99  head :method_not_allowed do
100   "Method Not Allowed"
101 end
102
103 head :not_implemented do
104   "Not Implemented"
105 end
106
107 head :internal_server_error do
108   "Internal Server Error"
109 end
110
111 head :method_not_allowed do
112   "Method Not Allowed"
113 end
114
115 head :not_implemented do
116   "Not Implemented"
117 end
118
119 head :internal_server_error do
120   "Internal Server Error"
121 end
122
123 head :method_not_allowed do
124   "Method Not Allowed"
125 end
126
127 head :not_implemented do
128   "Not Implemented"
129 end
130
131 head :internal_server_error do
132   "Internal Server Error"
133 end
134
135 head :method_not_allowed do
136   "Method Not Allowed"
137 end
138
139 head :not_implemented do
140   "Not Implemented"
141 end
142
143 head :internal_server_error do
144   "Internal Server Error"
145 end
146
147 head :method_not_allowed do
148   "Method Not Allowed"
149 end
150
151 head :not_implemented do
152   "Not Implemented"
153 end
154
155 head :internal_server_error do
156   "Internal Server Error"
157 end
158
159 head :method_not_allowed do
160   "Method Not Allowed"
161 end
162
163 head :not_implemented do
164   "Not Implemented"
165 end
166
167 head :internal_server_error do
168   "Internal Server Error"
169 end
170
171 head :method_not_allowed do
172   "Method Not Allowed"
173 end
174
175 head :not_implemented do
176   "Not Implemented"
177 end
178
179 head :internal_server_error do
180   "Internal Server Error"
181 end
182
183 head :method_not_allowed do
184   "Method Not Allowed"
185 end
186
187 head :not_implemented do
188   "Not Implemented"
189 end
190
191 head :internal_server_error do
192   "Internal Server Error"
193 end
194
195 head :method_not_allowed do
196   "Method Not Allowed"
197 end
198
199 head :not_implemented do
200   "Not Implemented"
201 end
202
203 head :internal_server_error do
204   "Internal Server Error"
205 end
206
207 head :method_not_allowed do
208   "Method Not Allowed"
209 end
210
211 head :not_implemented do
212   "Not Implemented"
213 end
214
215 head :internal_server_error do
216   "Internal Server Error"
217 end
218
219 head :method_not_allowed do
220   "Method Not Allowed"
221 end
222
223 head :not_implemented do
224   "Not Implemented"
225 end
226
227 head :internal_server_error do
228   "Internal Server Error"
229 end
230
231 head :method_not_allowed do
232   "Method Not Allowed"
233 end
234
235 head :not_implemented do
236   "Not Implemented"
237 end
238
239 head :internal_server_error do
240   "Internal Server Error"
241 end
242
243 head :method_not_allowed do
244   "Method Not Allowed"
245 end
246
247 head :not_implemented do
248   "Not Implemented"
249 end
250
251 head :internal_server_error do
252   "Internal Server Error"
253 end
254
255 head :method_not_allowed do
256   "Method Not Allowed"
257 end
258
259 head :not_implemented do
260   "Not Implemented"
261 end
262
263 head :internal_server_error do
264   "Internal Server Error"
265 end
266
267 head :method_not_allowed do
268   "Method Not Allowed"
269 end
270
271 head :not_implemented do
272   "Not Implemented"
273 end
274
275 head :internal_server_error do
276   "Internal Server Error"
277 end
278
279 head :method_not_allowed do
280   "Method Not Allowed"
281 end
282
283 head :not_implemented do
284   "Not Implemented"
285 end
286
287 head :internal_server_error do
288   "Internal Server Error"
289 end
290
291 head :method_not_allowed do
292   "Method Not Allowed"
293 end
294
295 head :not_implemented do
296   "Not Implemented"
297 end
298
299 head :internal_server_error do
300   "Internal Server Error"
301 end
302
303 head :method_not_allowed do
304   "Method Not Allowed"
305 end
306
307 head :not_implemented do
308   "Not Implemented"
309 end
310
311 head :internal_server_error do
312   "Internal Server Error"
313 end
314
315 head :method_not_allowed do
316   "Method Not Allowed"
317 end
318
319 head :not_implemented do
320   "Not Implemented"
321 end
322
323 head :internal_server_error do
324   "Internal Server Error"
325 end
326
327 head :method_not_allowed do
328   "Method Not Allowed"
329 end
330
331 head :not_implemented do
332   "Not Implemented"
333 end
334
335 head :internal_server_error do
336   "Internal Server Error"
337 end
338
339 head :method_not_allowed do
340   "Method Not Allowed"
341 end
342
343 head :not_implemented do
344   "Not Implemented"
345 end
346
347 head :internal_server_error do
348   "Internal Server Error"
349 end
350
351 head :method_not_allowed do
352   "Method Not Allowed"
353 end
354
355 head :not_implemented do
356   "Not Implemented"
357 end
358
359 head :internal_server_error do
360   "Internal Server Error"
361 end
362
363 head :method_not_allowed do
364   "Method Not Allowed"
365 end
366
367 head :not_implemented do
368   "Not Implemented"
369 end
370
371 head :internal_server_error do
372   "Internal Server Error"
373 end
374
375 head :method_not_allowed do
376   "Method Not Allowed"
377 end
378
379 head :not_implemented do
380   "Not Implemented"
381 end
382
383 head :internal_server_error do
384   "Internal Server Error"
385 end
386
387 head :method_not_allowed do
388   "Method Not Allowed"
389 end
390
391 head :not_implemented do
392   "Not Implemented"
393 end
394
395 head :internal_server_error do
396   "Internal Server Error"
397 end
398
399 head :method_not_allowed do
400   "Method Not Allowed"
401 end
402
403 head :not_implemented do
404   "Not Implemented"
405 end
406
407 head :internal_server_error do
408   "Internal Server Error"
409 end
410
411 head :method_not_allowed do
412   "Method Not Allowed"
413 end
414
415 head :not_implemented do
416   "Not Implemented"
417 end
418
419 head :internal_server_error do
420   "Internal Server Error"
421 end
422
423 head :method_not_allowed do
424   "Method Not Allowed"
425 end
426
427 head :not_implemented do
428   "Not Implemented"
429 end
430
431 head :internal_server_error do
432   "Internal Server Error"
433 end
434
435 head :method_not_allowed do
436   "Method Not Allowed"
437 end
438
439 head :not_implemented do
440   "Not Implemented"
441 end
442
443 head :internal_server_error do
444   "Internal Server Error"
445 end
446
447 head :method_not_allowed do
448   "Method Not Allowed"
449 end
450
451 head :not_implemented do
452   "Not Implemented"
453 end
454
455 head :internal_server_error do
456   "Internal Server Error"
457 end
458
459 head :method_not_allowed do
460   "Method Not Allowed"
461 end
462
463 head :not_implemented do
464   "Not Implemented"
465 end
466
467 head :internal_server_error do
468   "Internal Server Error"
469 end
470
471 head :method_not_allowed do
472   "Method Not Allowed"
473 end
474
475 head :not_implemented do
476   "Not Implemented"
477 end
478
479 head :internal_server_error do
480   "Internal Server Error"
481 end
482
483 head :method_not_allowed do
484   "Method Not Allowed"
485 end
486
487 head :not_implemented do
488   "Not Implemented"
489 end
490
491 head :internal_server_error do
492   "Internal Server Error"
493 end
494
495 head :method_not_allowed do
496   "Method Not Allowed"
497 end
498
499 head :not_implemented do
500   "Not Implemented"
501 end
502
503 head :internal_server_error do
504   "Internal Server Error"
505 end
506
507 head :method_not_allowed do
508   "Method Not Allowed"
509 end
510
511 head :not_implemented do
512   "Not Implemented"
513 end
514
515 head :internal_server_error do
516   "Internal Server Error"
517 end
518
519 head :method_not_allowed do
520   "Method Not Allowed"
521 end
522
523 head :not_implemented do
524   "Not Implemented"
525 end
526
527 head :internal_server_error do
528   "Internal Server Error"
529 end
530
531 head :method_not_allowed do
532   "Method Not Allowed"
533 end
534
535 head :not_implemented do
536   "Not Implemented"
537 end
538
539 head :internal_server_error do
540   "Internal Server Error"
541 end
542
543 head :method_not_allowed do
544   "Method Not Allowed"
545 end
546
547 head :not_implemented do
548   "Not Implemented"
549 end
550
551 head :internal_server_error do
552   "Internal Server Error"
553 end
554
555 head :method_not_allowed do
556   "Method Not Allowed"
557 end
558
559 head :not_implemented do
560   "Not Implemented"
561 end
562
563 head :internal_server_error do
564   "Internal Server Error"
565 end
566
567 head :method_not_allowed do
568   "Method Not Allowed"
569 end
570
571 head :not_implemented do
572   "Not Implemented"
573 end
574
575 head :internal_server_error do
576   "Internal Server Error"
577 end
578
579 head :method_not_allowed do
580   "Method Not Allowed"
581 end
582
583 head :not_implemented do
584   "Not Implemented"
585 end
586
587 head :internal_server_error do
588   "Internal Server Error"
589 end
590
591 head :method_not_allowed do
592   "Method Not Allowed"
593 end
594
595 head :not_implemented do
596   "Not Implemented"
597 end
598
599 head :internal_server_error do
600   "Internal Server Error"
601 end
602
603 head :method_not_allowed do
604   "Method Not Allowed"
605 end
606
607 head :not_implemented do
608   "Not Implemented"
609 end
610
611 head :internal_server_error do
612   "Internal Server Error"
613 end
614
615 head :method_not_allowed do
616   "Method Not Allowed"
617 end
618
619 head :not_implemented do
620   "Not Implemented"
621 end
622
623 head :internal_server_error do
624   "Internal Server Error"
625 end
626
627 head :method_not_allowed do
628   "Method Not Allowed"
629 end
630
631 head :not_implemented do
632   "Not Implemented"
633 end
634
635 head :internal_server_error do
636   "Internal Server Error"
637 end
638
639 head :method_not_allowed do
640   "Method Not Allowed"
641 end
642
643 head :not_implemented do
644   "Not Implemented"
645 end
646
647 head :internal_server_error do
648   "Internal Server Error"
649 end
650
651 head :method_not_allowed do
652   "Method Not Allowed"
653 end
654
655 head :not_implemented do
656   "Not Implemented"
657 end
658
659 head :internal_server_error do
660   "Internal Server Error"
661 end
662
663 head :method_not_allowed do
664   "Method Not Allowed"
665 end
666
667 head :not_implemented do
668   "Not Implemented"
669 end
670
671 head :internal_server_error do
672   "Internal Server Error"
673 end
674
675 head :method_not_allowed do
676   "Method Not Allowed"
677 end
678
679 head :not_implemented do
680   "Not Implemented"
681 end
682
683 head :internal_server_error do
684   "Internal Server Error"
685 end
686
687 head :method_not_allowed do
688   "Method Not Allowed"
689 end
690
691 head :not_implemented do
692   "Not Implemented"
693 end
694
695 head :internal_server_error do
696   "Internal Server Error"
697 end
698
699 head :method_not_allowed do
700   "Method Not Allowed"
701 end
702
703 head :not_implemented do
704   "Not Implemented"
705 end
706
707 head :internal_server_error do
708   "Internal Server Error"
709 end
710
711 head :method_not_allowed do
712   "Method Not Allowed"
713 end
714
715 head :not_implemented do
716   "Not Implemented"
717 end
718
719 head :internal_server_error do
720   "Internal Server Error"
721 end
722
723 head :method_not_allowed do
724   "Method Not Allowed"
725 end
726
727 head :not_implemented do
728   "Not Implemented"
729 end
730
731 head :internal_server_error do
732   "Internal Server Error"
733 end
734
735 head :method_not_allowed do
736   "Method Not Allowed"
737 end
738
739 head :not_implemented do
740   "Not Implemented"
741 end
742
743 head :internal_server_error do
744   "Internal Server Error"
745 end
746
747 head :method_not_allowed do
748   "Method Not Allowed"
749 end
750
751 head :not_implemented do
752   "Not Implemented"
753 end
754
755 head :internal_server_error do
756   "Internal Server Error"
757 end
758
759 head :method_not_allowed do
760   "Method Not Allowed"
761 end
762
763 head :not_implemented do
764   "Not Implemented"
765 end
766
767 head :internal_server_error do
768   "Internal Server Error"
769 end
770
771 head :method_not_allowed do
772   "Method Not Allowed"
773 end
774
775 head :not_implemented do
776   "Not Implemented"
777 end
778
779 head :internal_server_error do
780   "Internal Server Error"
781 end
782
783 head :method_not_allowed do
784   "Method Not Allowed"
785 end
786
787 head :not_implemented do
788   "Not Implemented"
789 end
790
791 head :internal_server_error do
792   "Internal Server Error"
793 end
794
795 head :method_not_allowed do
796   "Method Not Allowed"
797 end
798
799 head :not_implemented do
800   "Not Implemented"
801 end
802
803 head :internal_server_error do
804   "Internal Server Error"
805 end
806
807 head :method_not_allowed do
808   "Method Not Allowed"
809 end
810
811 head :not_implemented do
812   "Not Implemented"
813 end
814
815 head :internal_server_error do
816   "Internal Server Error"
817 end
818
819 head :method_not_allowed do
820   "Method Not Allowed"
821 end
822
823 head :not_implemented do
824   "Not Implemented"
825 end
826
827 head :internal_server_error do
828   "Internal Server Error"
829 end
830
831 head :method_not_allowed do
832   "Method Not Allowed"
833 end
834
835 head :not_implemented do
836   "Not Implemented"
837 end
838
839 head :internal_server_error do
840   "Internal Server Error"
841 end
842
843 head :method_not_allowed do
844   "Method Not Allowed"
845 end
846
847 head :not_implemented do
848   "Not Implemented"
849 end
850
851 head :internal_server_error do
852   "Internal Server Error"
853 end
854
855 head :method_not_allowed do
856   "Method Not Allowed"
857 end
858
859 head :not_implemented do
860   "Not Implemented"
861 end
862
863 head :internal_server_error do
864   "Internal Server Error"
865 end
866
867 head :method_not_allowed do
868   "Method Not Allowed"
869 end
870
871 head :not_implemented do
872   "Not Implemented"
873 end
874
875 head :internal_server_error do
876   "Internal Server Error"
877 end
878
879 head :method_not_allowed do
880   "Method Not Allowed"
881 end
882
883 head :not_implemented do
884   "Not Implemented"
885 end
886
887 head :internal_server_error do
888   "Internal Server Error"
889 end
890
891 head :method_not_allowed do
892   "Method Not Allowed"
893 end
894
895 head :not_implemented do
896   "Not Implemented"
897 end
898
899 head :internal_server_error do
900   "Internal Server Error"
901 end
902
903 head :method_not_allowed do
904   "Method Not Allowed"
905 end
906
907 head :not_implemented do
908   "Not Implemented"
909 end
910
911 head :internal_server_error do
912   "Internal Server Error"
913 end
914
915 head :method_not_allowed do
916   "Method Not Allowed"
917 end
918
919 head :not_implemented do
920   "Not Implemented"
921 end
922
923 head :internal_server_error do
924   "Internal Server Error"
925 end
926
927 head :method_not_allowed do
928   "Method Not Allowed"
929 end
930
931 head :not_implemented do
932   "Not Implemented"
933 end
934
935 head :internal_server_error do
936   "Internal Server Error"
937 end
938
939 head :method_not_allowed do
940   "Method Not Allowed"
941 end
942
943 head :not_implemented do
944   "Not Implemented"
945 end
946
947 head :internal_server_error do
948   "Internal Server Error"
949 end
950
951 head :method_not_allowed do
952   "Method Not Allowed"
953 end
954
955 head :not_implemented do
956   "Not Implemented"
957 end
958
959 head :internal_server_error do
960   "Internal Server Error"
961 end
962
963 head :method_not_allowed do
964   "Method Not Allowed"
965 end
966
967 head :not_implemented do
968   "Not Implemented"
969 end
970
971 head :internal_server_error do
972   "Internal Server Error"
973 end
974
975 head :method_not_allowed do
976   "Method Not Allowed"
977 end
978
979 head :not_implemented do
980   "Not Implemented"
981 end
982
983 head :internal_server_error do
984   "Internal Server Error"
985 end
986
987 head :method_not_allowed do
988   "Method Not Allowed"
989 end
990
991 head :not_implemented do
992   "Not Implemented"
993 end
994
995 head :internal_server_error do
996   "Internal Server Error"
997 end
998
999 head :method_not_allowed do
1000   "Method Not Allowed"
1001 end
1002
1003 head :not_implemented do
1004   "Not Implemented"
1005 end
1006
1007 head :internal_server_error do
1008   "Internal Server Error"
1009 end
1010
1011 head :method_not_allowed do
1012   "Method Not Allowed"
1013 end
1014
1015 head :not_implemented do
1016   "Not Implemented"
1017 end
1018
1019 head :internal_server_error do
1020   "Internal Server Error"
1021 end
1022
1023 head :method_not_allowed do
1024   "Method Not Allowed"
1025 end
1026
1027 head :not_implemented do
1028   "Not Implemented"
1029 end
1030
1031 head :internal_server_error do
1032   "Internal Server Error"
1033 end
1034
1035 head :method_not_allowed do
1036   "Method Not Allowed"
1037 end
1038
1039 head :not_implemented do
1040   "Not Implemented"
1041 end
1042
1043 head :internal_server_error do
1044   "Internal Server Error"
1045 end
1046
1047 head :method_not_allowed do
1048   "Method Not Allowed"
1049 end
1050
1051 head :not_implemented do
1052   "Not Implemented"
1053 end
1054
1055 head :internal_server_error do
1056   "Internal Server Error"
1057 end
1058
1059 head :method_not_allowed do
1060   "Method Not Allowed"
1061 end
1062
1063 head :not_implemented do
1064   "Not Implemented"
1065 end
1066
1067 head :internal_server_error do
1068   "Internal Server Error"
1069 end
1070
1071 head :method_not_allowed do
1072   "Method Not Allowed"
1073 end
1074
1075 head :not_implemented do
1076   "Not Implemented"
1077 end
1078
1079 head :internal_server_error do
1080   "Internal Server Error"
1081 end
1082
1083 head :method_not_allowed do
1084   "Method Not Allowed"
1085 end
1086
1087 head :not_implemented do
1088   "Not Implemented"
1089 end
1090
1091 head :internal_server_error do
1092   "Internal Server Error"
1093 end
1094
1095 head :method_not_allowed do
1096   "Method Not Allowed"
1097 end
1098
1099 head :not_implemented do
1100   "Not Implemented"
1101 end
1102
1103 head :internal_server_error do
1104   "Internal Server Error"
1105 end
1106
1107 head :method_not_allowed do
1108   "Method Not Allowed"
1109 end
1110
1111 head :not_implemented do
1112   "Not Implemented"
1113 end
1114
1115 head :internal_server_error do
1116   "Internal Server Error"
1117 end
1118
1119 head :method_not_allowed do
1120   "Method Not Allowed"
1121 end
1122
1123 head :not_implemented do
1124   "Not Implemented"
1125 end
1126
1127 head :internal_server_error do
1128   "Internal Server Error"
1129 end
1130
1131 head :method_not_allowed do
1132   "Method Not Allowed"
1133 end
1134
1135 head :not_implemented do
1136   "Not Implemented"
1137 end
1138
1139 head :internal_server_error do
1140   "Internal Server Error"
1141 end
1142
1143 head :method_not_allowed do
1144   "Method Not Allowed"
1145 end
1146
1147 head :not_implemented do
1148   "Not Implemented"
1149 end
1150
1151 head :internal_server_error do
1152   "Internal Server Error"
1153 end
1154
1155 head :method_not_allowed do
1156   "Method Not Allowed"
1157 end
1158
1159 head :not_implemented do
1160   "Not Implemented"
1161 end
1162
1163 head :internal_server_error do
1164   "Internal Server Error"
1165 end
1166
1167 head :method_not_allowed do
1168   "Method Not Allowed"
1169 end
1170
1171 head :not_implemented do
1172   "Not Implemented"
1173 end
1174
1175 head :internal_server_error do
1176   "Internal Server Error"
1177 end
1178
1179 head :method_not_allowed do
1180   "Method Not Allowed"
1181 end
1182
1183 head :not_implemented do
1184   "Not Implemented"
1185 end
1186
1187 head :internal_server_error do
1188   "Internal Server Error"
1189 end
1190
1191 head :method_not_allowed do
1192   "Method Not Allowed"
1193 end
1194
1195 head :not_implemented do
1196   "Not Implemented"
1197 end
1198
1199 head :internal_server_error do
1200   "Internal Server Error"
1201 end
1202
1203 head :method_not_allowed do
1204   "Method Not Allowed"
1205 end
1206
1207 head :not_implemented do
1208   "Not Implemented"
1209 end
1210
1211 head :internal_server_error do
1212   "Internal Server Error"
1213 end
1214
1215 head :method_not_allowed do
1216   "Method Not Allowed"
1217 end
1218
1219 head :not_implemented do
1220   "Not Implemented"
1221 end
1222
1223 head :internal_server_error do
1224   "Internal Server Error"
1225 end
1226
1227 head :method_not_allowed do
1228   "Method Not Allowed"
1229 end
1230
1231 head :not_implemented do
1232   "Not Implemented"
1233 end
1234
1235 head :internal_server_error do
1236   "Internal Server Error"
1237 end
1238
1239 head :method_not_allowed do
1240   "Method Not Allowed"
1241 end
1242
1243 head :not_implemented do
1244   "Not Implemented"
1245 end
1246
1247 head :internal_server_error do
1248   "Internal Server Error"
1249 end
1250
1251 head :method_not_allowed do
1252   "Method Not Allowed"
1253 end
1254
1255 head :not_implemented do
1256   "Not Implemented"
1257 end
1258
1259 head :internal_server_error do
1260   "Internal Server Error"
1261 end
1262
1263 head :method_not_allowed do
1264   "Method Not Allowed"
1265 end
1266
1267 head :not_implemented do
1268   "Not Implemented"
1269 end
1270
1271 head :internal_server_error do
1272   "Internal Server Error"
1273 end
1274
1275 head :method_not_allowed do
1276   "Method Not Allowed"
1277 end
1278
1279 head :not_implemented do
1280   "Not Implemented"
1281 end
1282
1283 head :internal_server_error do
1284   "Internal Server Error"
1285 end
1286
1287 head :method_not_allowed do
1288   "Method Not Allowed"
1289 end
1290
1291 head :not_implemented do
1292   "Not Implemented"
1293 end
1294
1295 head :internal_server_error do
1296   "Internal Server Error"
1297 end
1298
1299 head :method_not_allowed do
1300   "Method Not Allowed"
1301 end
1302
1303 head :not_implemented do
1304   "Not Implemented"
1305 end
1306
1307 head :internal_server_error do
1308   "Internal Server Error"
1309 end
1310
1311 head :method_not_allowed do
1312   "Method Not Allowed"
1313 end
1314
1315 head :not_implemented do
1316   "Not Implemented"
1317 end
1318
1319 head :internal_server_error do
1320   "Internal Server Error"
1321 end
1322
1323 head :method_not_allowed do
1324   "Method Not Allowed"
1325 end
1326
1327 head :not_implemented do
1328   "Not Implemented"
1329 end
1330
1331 head :internal_server_error do
1332   "Internal Server Error"
1333 end
1334
1335 head :method_not_allowed do
1336   "Method Not Allowed"
1337 end
1338
1339 head :not_implemented do
1340   "Not Implemented"
1341 end
1342
1343 head :internal_server_error do
1344   "Internal Server Error"
1345 end
1346
1347 head :method_not_allowed do
1348   "Method Not Allowed"
1349 end
1350
1351 head :not_implemented do
1352   "Not Implemented"
1353 end
1354
1355 head :internal_server_error do
1356   "Internal Server Error"
1357 end
1358
1359 head :method_not_allowed do
1360   "Method Not Allowed"
1361 end
1362
1363 head :not_implemented do
1364   "Not Implemented"
1365 end
1366
1367 head :internal_server_error do
1368   "Internal Server Error"
1369 end
1370
1371 head :method_not_allowed do
1372   "Method Not Allowed"
1373 end
1374
1375 head :not_implemented do
1376   "Not Implemented"
1377 end
1378
1379 head :internal_server_error do
1380   "Internal Server Error"
1381 end
1382
1383 head :method_not_allowed do
1384   "Method Not Allowed"
1385 end
1386
1387 head :not_implemented do
1388   "Not Implemented"
1389 end
1390
1391 head :internal_server_error do
1392   "Internal Server Error"
1393 end
1394
1395 head :method_not_allowed do
1396   "Method Not Allowed"
1397 end
1398
1399 head :not_implemented do
1400   "Not Implemented"
1401 end
1402
1403 head :internal_server_error do
1404   "Internal Server Error"
1405 end
1406
1407 head :method_not_allowed do
1408   "Method Not Allowed"
1409 end
1410
1411 head :not_implemented do
1412   "Not Implemented"
1413 end
1414
1415 head :internal_server_error do
1416   "Internal Server Error"
1417 end
1418
1419 head :method_not_allowed do
1420   "Method Not Allowed"
1421 end
1422
1423 head :not_implemented do
1424   "Not Implemented"
1425 end
1426
1427 head :internal_server_error do
1428   "Internal Server Error"
1429 end
1430
1431 head :method_not_allowed do
1432   "Method Not Allowed"
1433 end
1434
1435 head :not_implemented do
1436   "Not Implemented"
1437 end
1438
1439 head :internal_server_error do
1440   "Internal Server Error"
1441 end
1442
1443 head :method_not_allowed do
1444   "Method Not Allowed"
1445 end
1446
1447 head :not_implemented do
1448   "Not Implemented"
1449 end
1450
1451 head :internal_server_error do
1452   "Internal Server Error"
1453 end
1454
1455 head :method_not_allowed do
1456   "Method Not Allowed"
1457 end
1458
1459 head :not_implemented do
1460   "Not Implemented"
1461 end
1462
1463 head :internal_server_error do
1464   "Internal Server Error"
1465 end
1466
1467 head :method_not_allowed do
1468   "Method Not Allowed"
1469 end
1470
1471 head :not_implemented do
1472   "Not Implemented"
1473 end
1474
1475 head :internal_server_error do
1476   "Internal Server Error"
1477 end
1478
1479 head :method_not_allowed do
1480   "Method Not Allowed"
1481 end
1482
1483 head :not_implemented do
1484   "Not Implemented"
1485 end
1486
1487 head :internal_server_error do
1488   "Internal Server Error"
1489 end
1490
1491 head :method_not_allowed do
1492   "Method Not Allowed"
1493 end
1494
1
```

Instrumentation and Tooling

Profiling



CPU-Sampler

- Let's introduce extra slowness affecting /people.js request
- CPU-Sampler can tell us which methods take the most time
 - ruby --jvm --polyglot -Xsingle_threaded --cpusampler app.rb

```

1 -----
2 Sampling Histogram. Recorded 225692 samples with period 1ms
3   Self Time: Time spent on the top of the stack.
4   Total Time: Time the location spent on the stack.
5   Opt %: Percent of time spent in compiled and therefore non-interpreted code.
6 -----
7 Thread: Thread[main,5,main]
8 Name | Total Time | Opt % || Self Time | Opt % | Location
9 -----
10 block in WEBrick::GenericServer#start | 218564ms 96.8% | 0.0% || 160833ms 71.3% | 0.0% | /Users/pitr/Workspace/labs/g
11 WEBrick::GenericServer#start_thread | 46639ms 20.7% | 0.0% || 28324ms 12.5% | 0.0% | /Users/pitr/Workspace/labs/g
12 block (2 levels) in WEBrick::GenericServer#start | 52570ms 23.3% | 0.0% || 5929ms 2.6% | 0.3% | /Users/pitr/Workspace/labs/g
13 Rack::Handler::WEBrick#service | 20936ms 9.3% | 35.7% || 2901ms 1.3% | 23.2% | /Users/pitr/Workspace/labs/g
14 <main> | 225443ms 99.9% | 0.0% || 1598ms 0.7% | 0.0% | app.rb-1-81:0-1740
15 Sinatra::Base#public_folder | 2972ms 1.3% | 43.1% || 1579ms 0.7% | 81.2% | /Users/pitr/Workspace/labs/g
16 Sinatra::Base#root | 1393ms 0.6% | 51.0% || 1388ms 0.6% | 51.2% | /Users/pitr/Workspace/labs/g
17 WEBrick::HTTPResponse#setup_header | 1335ms 0.6% | 0.4% || 1327ms 0.6% | 0.4% | /Users/pitr/Workspace/labs/g
18 Rack::Protection::PathTraversal#cleanup | 1190ms 0.5% | 16.9% || 1175ms 0.5% | 17.1% | /Users/pitr/Workspace/labs/g
19 Timeout#timeout | 1153ms 0.5% | 0.0% || 1153ms 0.5% | 0.0% | /Users/pitr/Workspace/labs/g
20 Sinatra::Base#process_route | 2879ms 1.3% | 35.4% || 1112ms 0.5% | 65.5% | /Users/pitr/Workspace/labs/g
21 Sinatra::Base#call | 12865ms 5.7% | 31.6% || 1047ms 0.5% | 93.3% | /Users/pitr/Workspace/labs/g
22 Sinatra::Base#static! | 2161ms 1.0% | 0.0% || 885ms 0.4% | 0.0% | /Users/pitr/Workspace/labs/g
23 <top (required)> | 1251ms 0.6% | 0.0% || 864ms 0.4% | 0.0% | /Users/pitr/Workspace/labs/g
24 PolyglotApp#GET /people.json | 998ms 0.4% | 0.0% || 847ms 0.4% | 0.0% | app.rb~36:821-843
25 Sinatra::Helpers#content_type | 1263ms 0.6% | 17.2% || 765ms 0.3% | 28.4% | /Users/pitr/Workspace/labs/g
26 <top (required)> | 1507ms 0.7% | 0.0% || 701ms 0.3% | 0.0% | /Users/pitr/Workspace/labs/g
27 Rack::Utils::HeaderHash#[]= | 567ms 0.3% | 0.0% || 567ms 0.3% | 0.0% | /Users/pitr/Workspace/labs/g
28 Rack::Utils::HeaderHash#[] | 551ms 0.2% | 0.0% || 551ms 0.2% | 0.0% | /Users/pitr/Workspace/labs/g
29 Rack::Protection::Base#html? | 760ms 0.3% | 34.2% || 549ms 0.2% | 47.4% | /Users/pitr/Workspace/labs/g
30 Rack::Protection::JsonCsrf#call | 16324ms 7.2% | 31.6% || 484ms 0.2% | 86.6% | /Users/pitr/Workspace/labs/g
31 Rack::Protection::JsonCsrf#has_vector? | 552ms 0.2% | 0.0% || 480ms 0.2% | 0.0% | /Users/pitr/Workspace/labs/g
32 Rack::Utils::HeaderHash#each | 871ms 0.4% | 27.4% || 420ms 0.2% | 56.9% | /Users/pitr/Workspace/labs/g
33

```

Line: 1 | Plain Text

Soft Tabs: 4

CPU-Sampler in Chrome DevTools

The screenshot shows the Chrome DevTools Profiler tab with a single profile named "Profile 1". The table displays the following data:

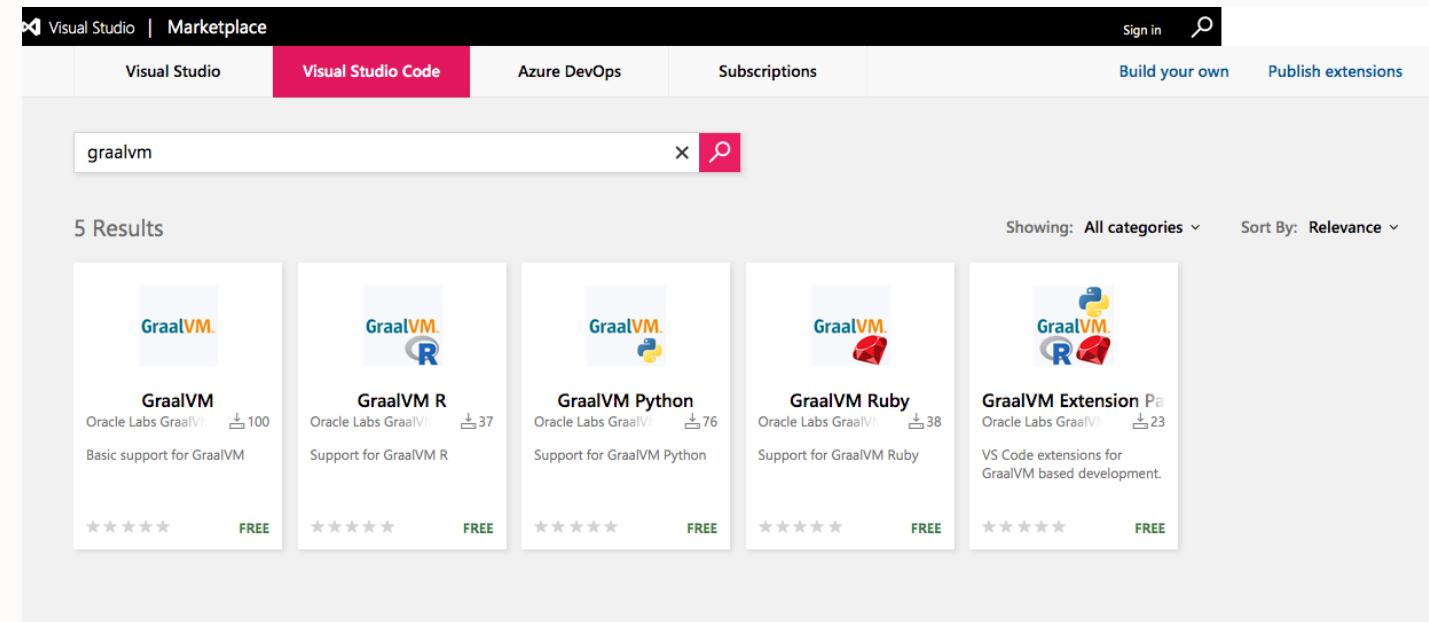
	Self Time	Total Time	Function	File
5688.0 ms	51.36 %	6315.0 ms	WEBrick::GenericServer#start_thread	single_threaded.rb:1
183.0 ms	1.65 %	251.0 ms	PolyglotApp#GET /people.json	app.rb:1
63.0 ms	0.57 %	558.0 ms	Rack::Handler::WEBrick#service	webrick.rb:1
53.0 ms	0.48 %	53.0 ms	Timeout#timeout	single_threaded.rb:13
52.0 ms	0.47 %	52.0 ms	block in PolyglotApp#js_person_as_hash	app.rb:1
39.0 ms	0.35 %	40.0 ms	Sinatra::Base#root	base.rb:18
30.0 ms	0.27 %	42.0 ms	PolyglotApp#GET /	app.rb:1
18.0 ms	0.16 %	18.0 ms	block in Rack::Handler::WEBrick#service	webrick.rb:1
16.0 ms	0.14 %	16.0 ms	WEBrick::HTTPResponse#setup_header	webrick.rb:1
13.0 ms	0.12 %	13.0 ms	Rack::Protection::PathTraversal#cleanup	path_traversal.rb:1
13.0 ms	0.12 %	65.0 ms	PolyglotApp#js_person_as_hash	app.rb:1
13.0 ms	0.12 %	14.0 ms	Rack::Protection::JsonCsrf#has_vector?	json_csrf.rb:1
12.0 ms	0.11 %	20.0 ms	Sinatra::Helpers#content_type	base.rb:3
11.0 ms	0.10 %	41.0 ms	Sinatra::Base#static!	base.rb:10
9.0 ms	0.08 %	13.0 ms	Rack::Protection::Base#html?	base.rb:12
8.0 ms	0.07 %	6323.0 ms	block (2 levels) in WEBrick::GenericServer#start	single_threaded.rb:1
7.0 ms	0.06 %	11.0 ms	Mustermann::Pattern#params	pattern.rb:2
6.0 ms	0.05 %	380.0 ms	Sinatra::Base#invoke	base.rb:10
6.0 ms	0.05 %	423.0 ms	Sinatra::Base#call!	base.rb:9
6.0 ms	0.05 %	6.0 ms	Rack::Utils::HeaderHash#[]	utils.rb:46

CPU-Tracer

```
1 Tracing Histogram. Counted a total of 51201 element executions.
2   Total Count: Number of times the element was executed and percentage of total executions.
3   Interpreted Count: Number of times the element was interpreted and percentage of total executions of this element.
4   Compiled Count: Number of times the compiled element was executed and percentage of total executions of this element.
5 -----
6   Name           |   Total Count |   Interpreted Count |   Compiled Count | Location
7 -----
8
9  block in Mustermann::AST::Translator#decorator_for |   6768 13.2% |   6508 96.2% |   260 3.8% | /Users/pitr/Workspace/labs/graalvm-e
10 Mustermann::AST::Node#payload=                      |   3305 6.5% |   3305 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
11 Mustermann::AST::Translator.dispatch_table          |   2392 4.7% |   2302 96.2% |   90 3.8% | /Users/pitr/Workspace/labs/graalvm-e
12 ossl_lock_unlock                                  |   2078 4.1% |   2078 100.0% |   0 0.0% | ../../graalvm-ee-1.0.0-rc6/Contents/
13 ossl_lock_callback                                |   2078 4.1% |   2078 100.0% |   0 0.0% | ../../graalvm-ee-1.0.0-rc6/Contents/
14 is_rstring_ptr                                    |   1576 3.1% |   1501 95.2% |   75 4.8% | /Users/graal/slave/e/truffleruby/src
15 rb_str_new                                       |   1576 3.1% |   1576 100.0% |   0 0.0% | /Users/graal/slave/e/truffleruby/src
16 is_managed_rstring_ptr                           |   1576 3.1% |   1501 95.2% |   75 4.8% | /Users/graal/slave/e/truffleruby/src
17 @strlen                                         |   1509 2.9% |   179 11.9% | 1330 88.1% | libsulong.bc:@strlen~1:0-19
18 rb_str_new_cstr                                 |   1497 2.9% |   1497 100.0% |   0 0.0% | /Users/graal/slave/e/truffleruby/src
19 Mustermann::AST::Translator::NodeTranslator#t     |   1074 2.1% |   1074 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
20 Mustermann::AST::Translator::NodeTranslator#translator |   1074 2.1% |   1074 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
21 rb_nativethread_self                            |   1043 2.0% |   1043 100.0% |   0 0.0% | /Users/graal/slave/e/truffleruby/src
22 Mustermann::AST::Translator#translate            |   910 1.8% |   910 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
23 Mustermann::AST::Translator#decorator_for       |   910 1.8% |   910 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
24 Mustermann::AST::Translator::NodeTranslator#initialize |   910 1.8% |   910 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
25 Mustermann::AST::Parser#buffer                  |   850 1.7% |   850 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
26 rb_define_method                                |   689 1.3% |   689 100.0% |   0 0.0% | /Users/graal/slave/e/truffleruby/src
27 Mustermann::AST::Node.□                         |   584 1.1% |   584 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
28 block in PolyglotApp#js_person_as_hash         |   564 1.1% |   564 100.0% |   0 0.0% | app.rb~27:623-681
29 Mustermann::AST::Parser#pos                     |   484 0.9% |   484 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
30 Mustermann::AST::Node#is_a?                     |   470 0.9% |   470 100.0% |   0 0.0% | /Users/pitr/Workspace/labs/graalvm-e
31
```

Line: 1 | Plain Text ⌂ | Soft Tabs: 4 ⌂ | ⚙ ⌂ |

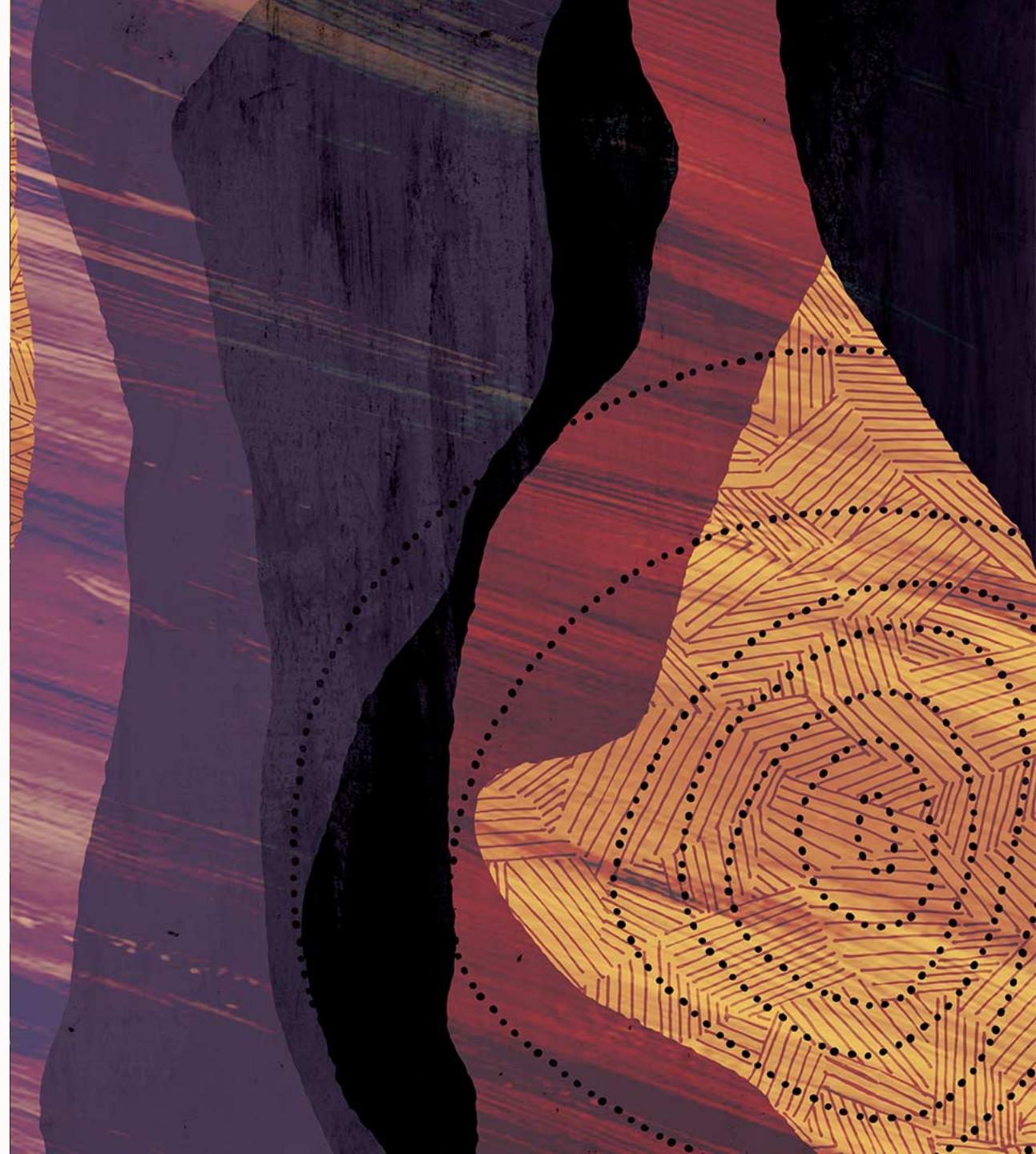
Visual Studio Code Plugins



<https://marketplace.visualstudio.com/search?term=graalvm&target=VSCode&category>All%20categories&sortBy=Relevance>

Instrumentation and Tooling

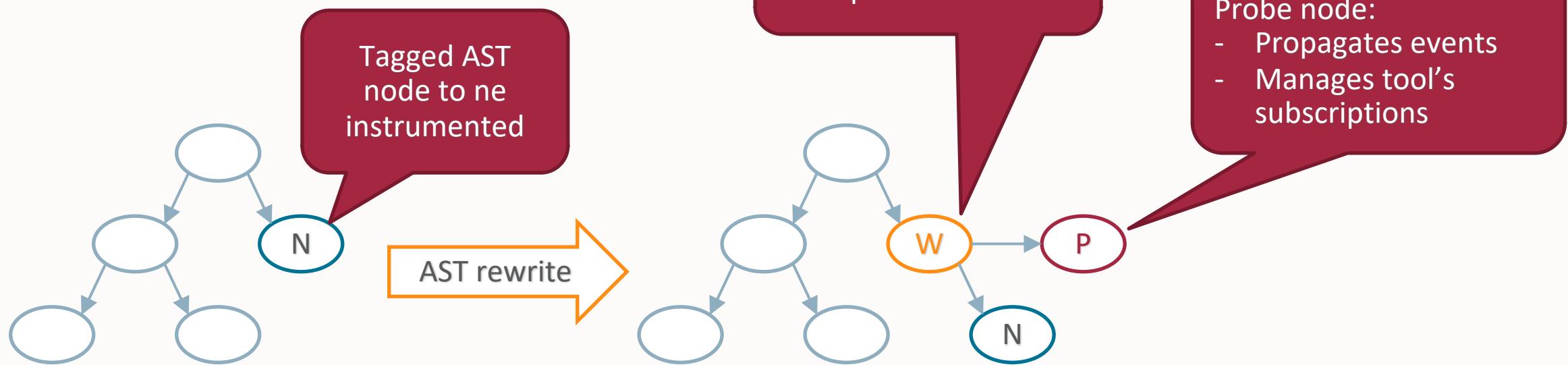
Instrumentation



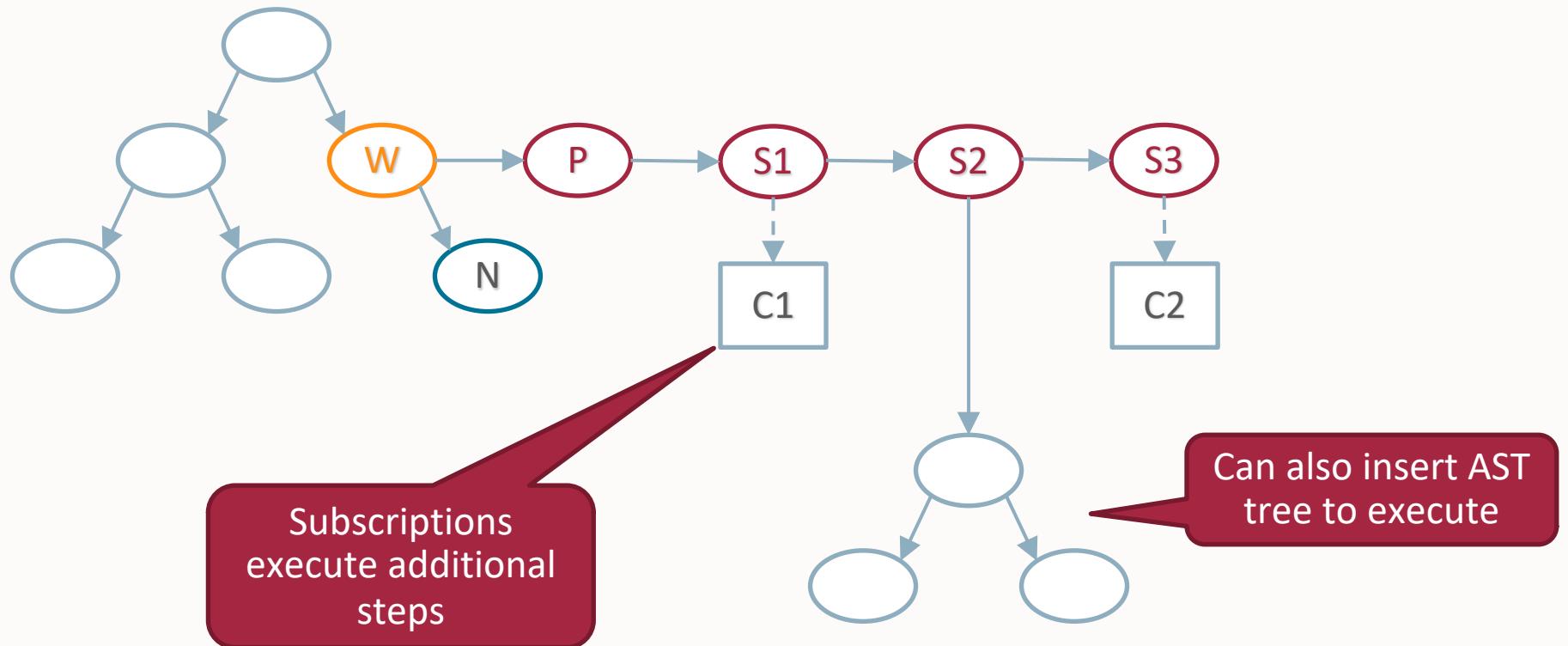
Instrumentation

- One API built on top of rewriting AST
 - Languages need to only implement the instrumentation API
 - Minimal language support requirements
 - A language gets all the tools for free
 - Tools work for all languages
- Languages tags nodes
 - Expression, Call, Root, etc.
- Tools then can request to subscribe to events on tagged nodes
- Close to zero overhead when not used

Instrumentation



Instrumentation



Implementing difficult language features

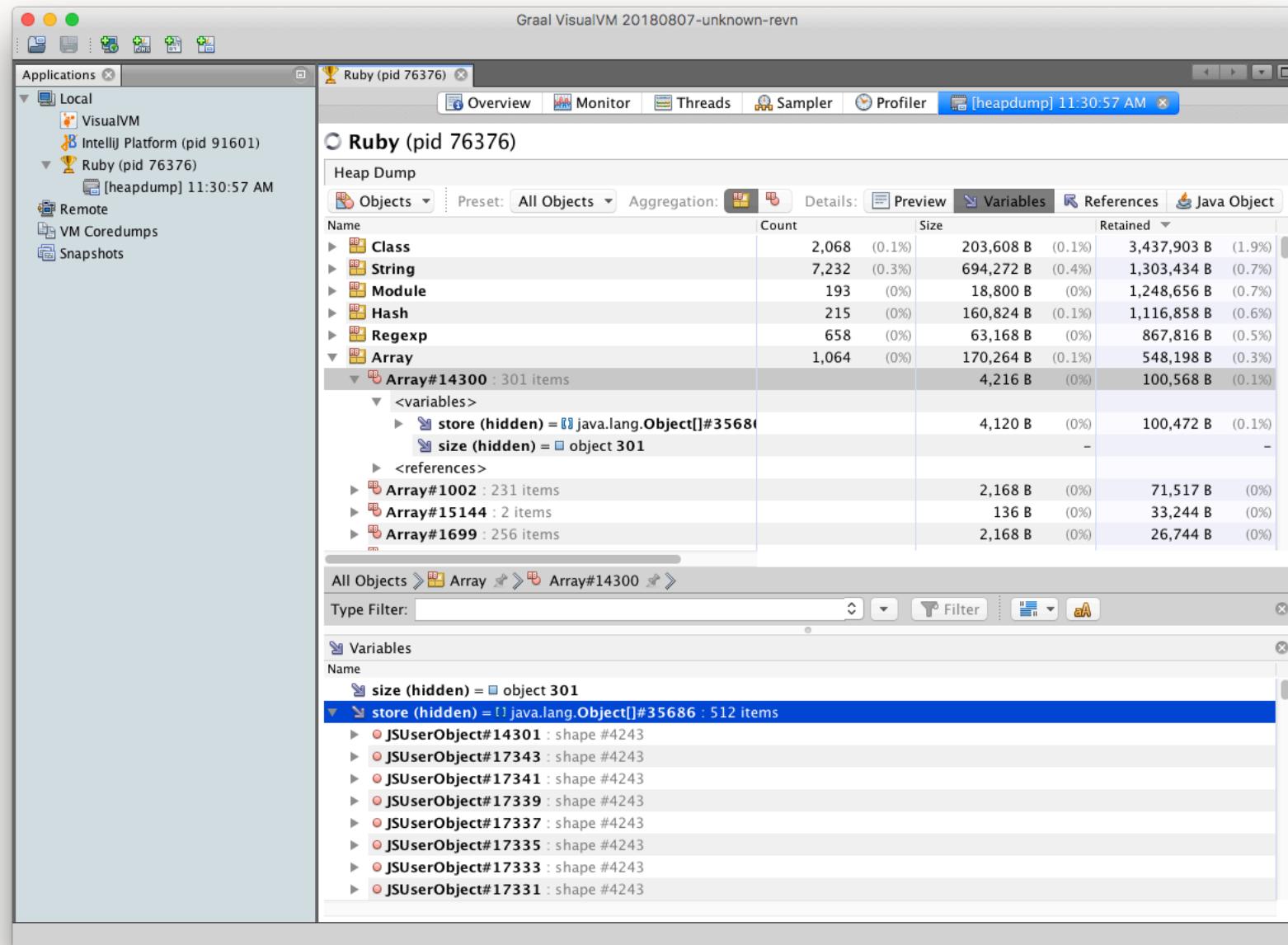
- Instrumentation is also used by the languages themselves
- R stepping
- Ruby `set_trace_func`
 - Attaches a function which gets executed on each: call, line, ...

Graal VisualVM

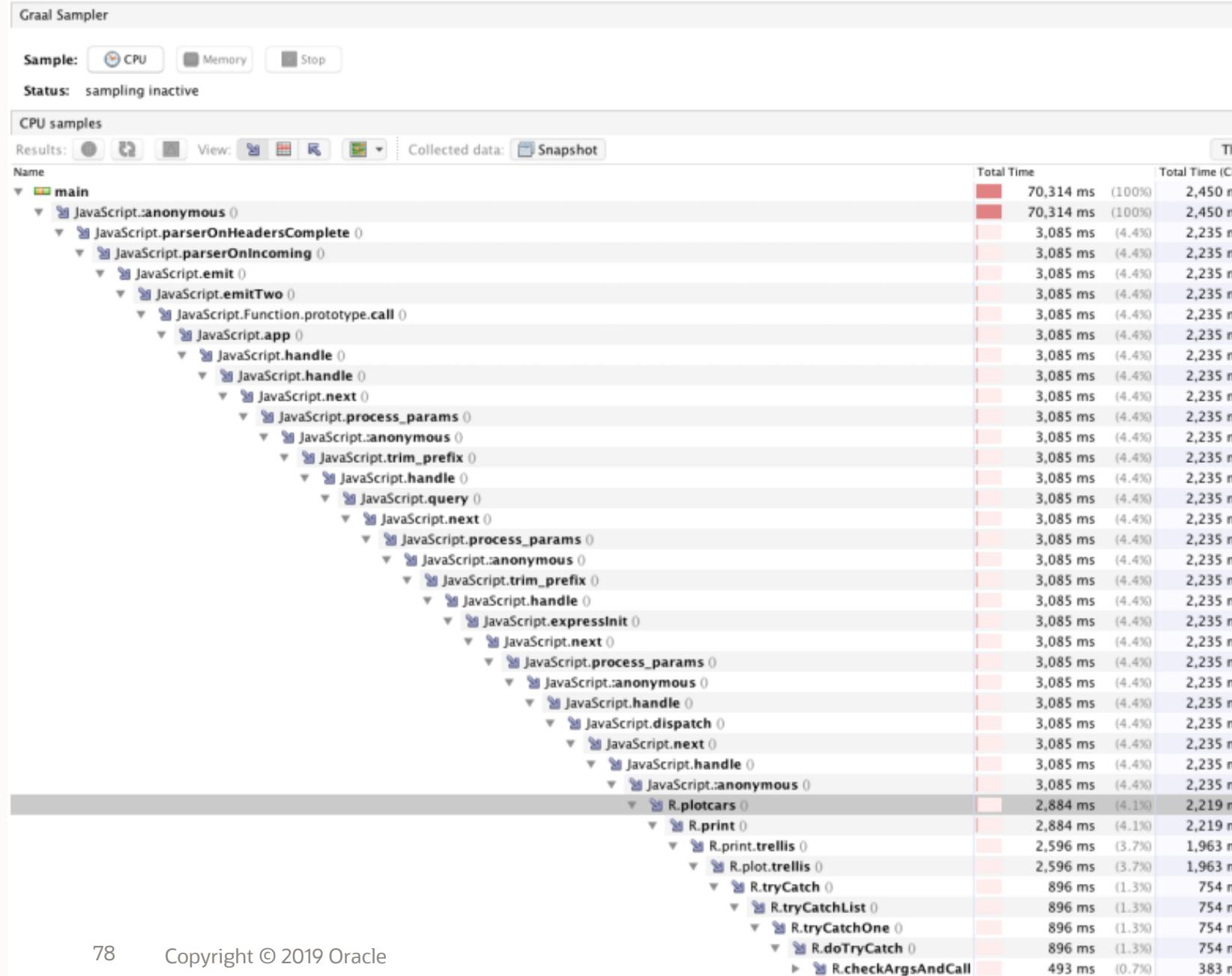


Graal VisualVM

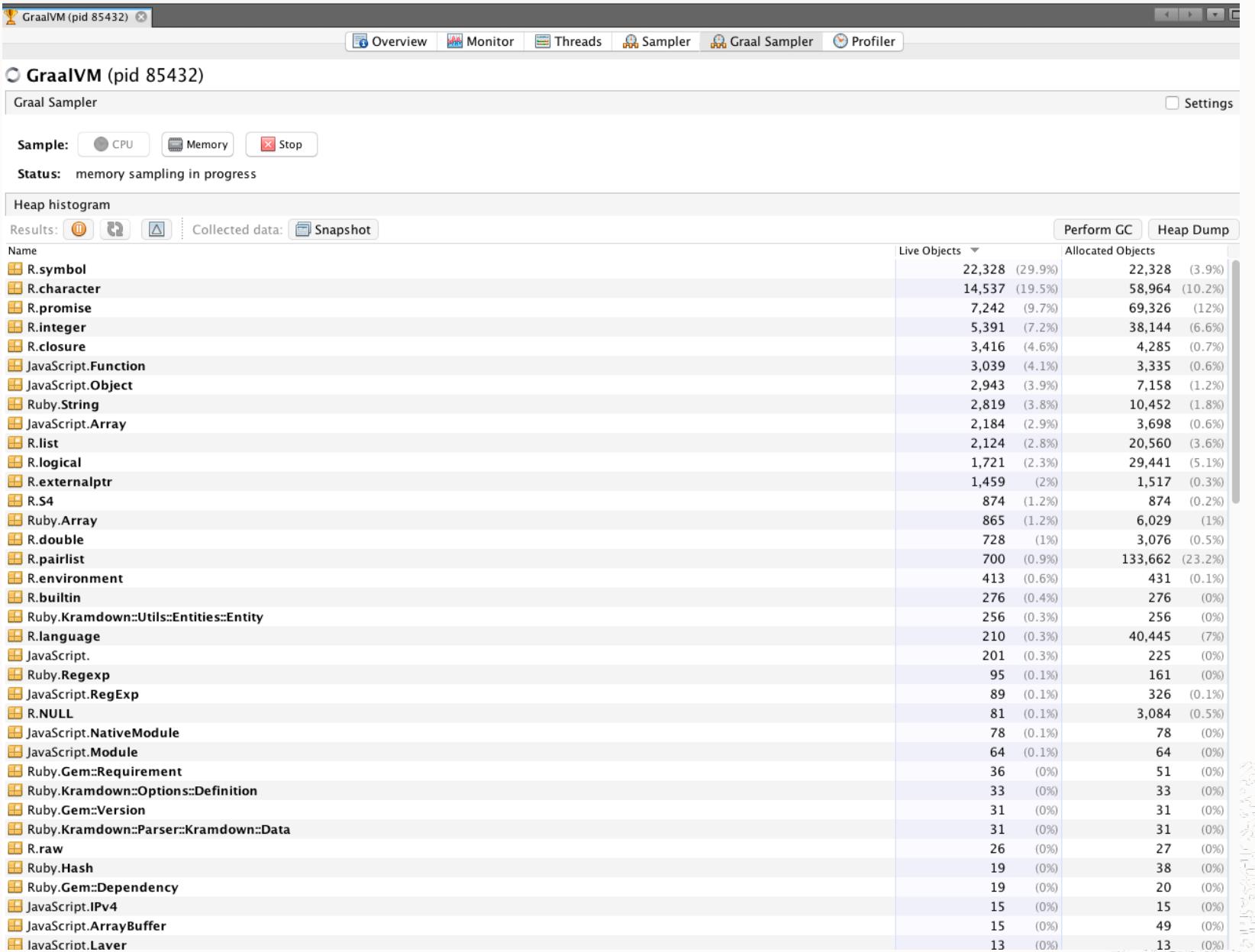
- Graal VisualVM has special support to understand Ruby, JS heap
- Compute retained sizes
 - The size including internal structures
 - String -> Rope -> byte[]
- Let's create 300 extra people in the in-memory database
 - <http://localhost:4567/add/300>
 - We can lookup the array and its size (0.1%)



GraalVM (pid 85381)



Polyglot Stack Trace

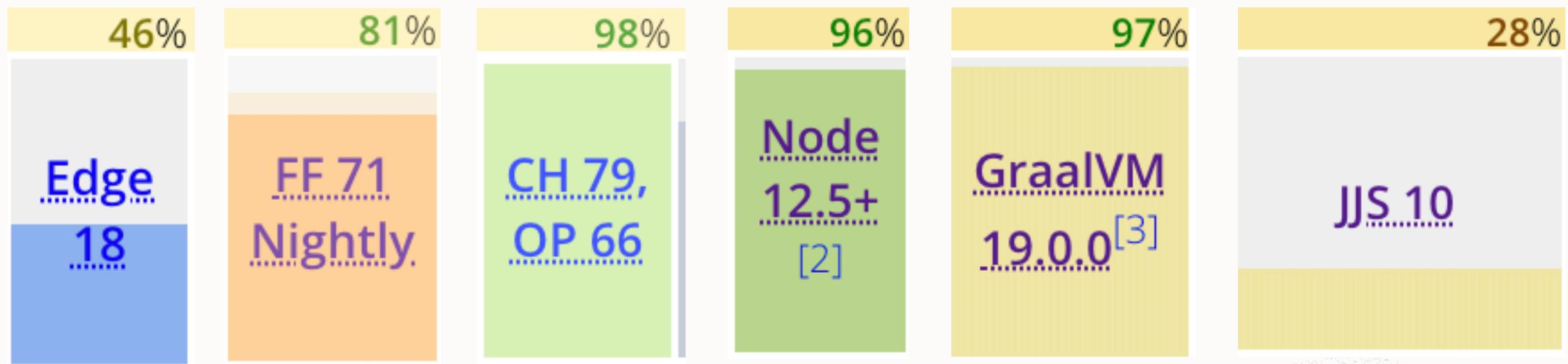


Polyglot Heap Dump

Language status



JS compatibility



<https://www.graalvm.org/docs/reference-manual/compatibility/>

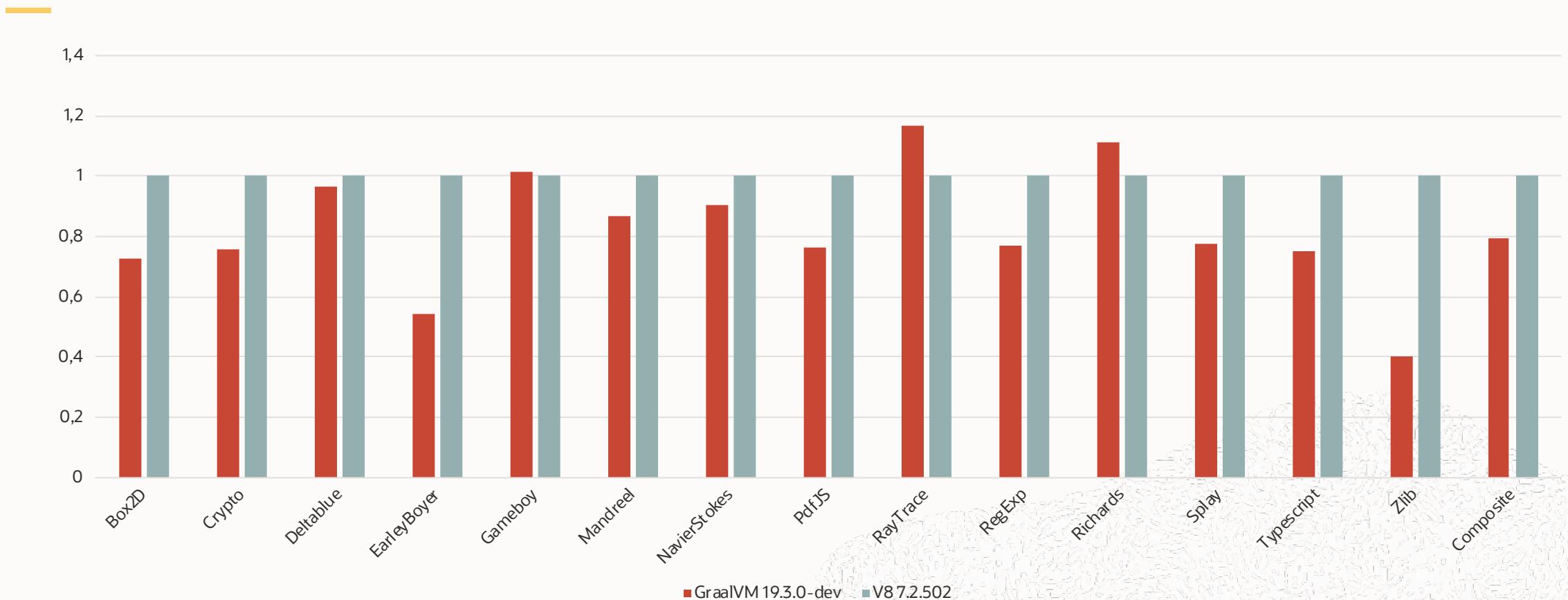
Quickly check if an NPM module, Ruby gem, or R package is compatible with GraalVM.

x CHECK!

Graal.js

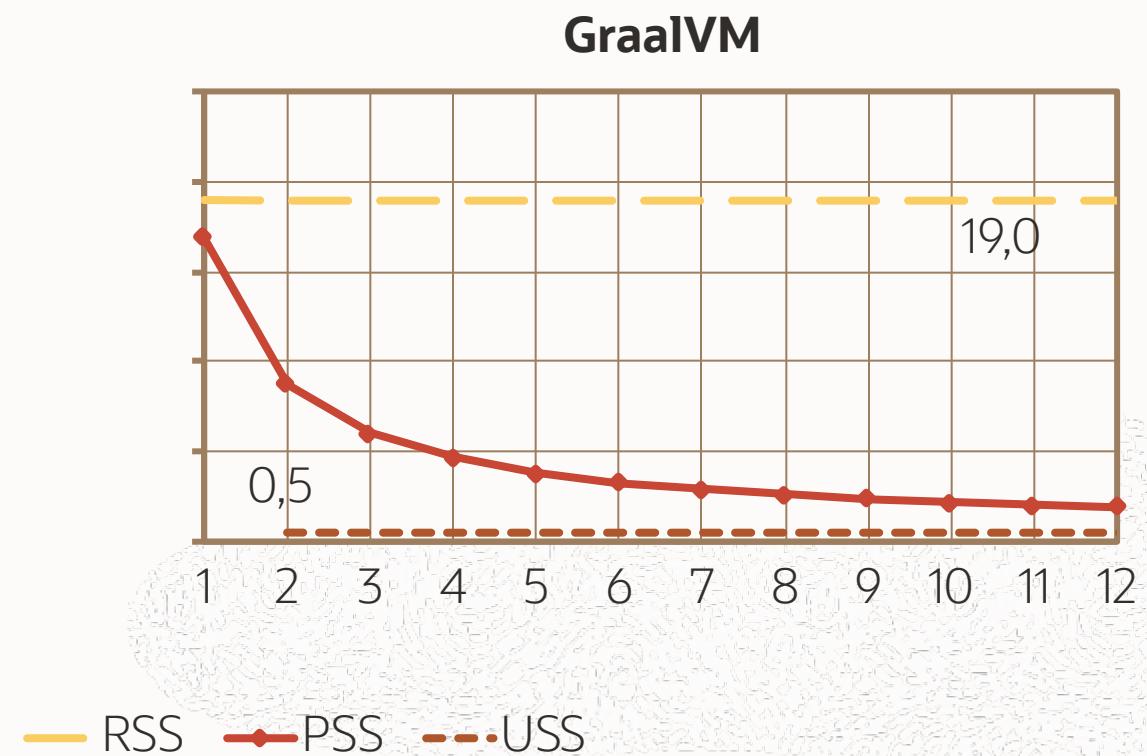
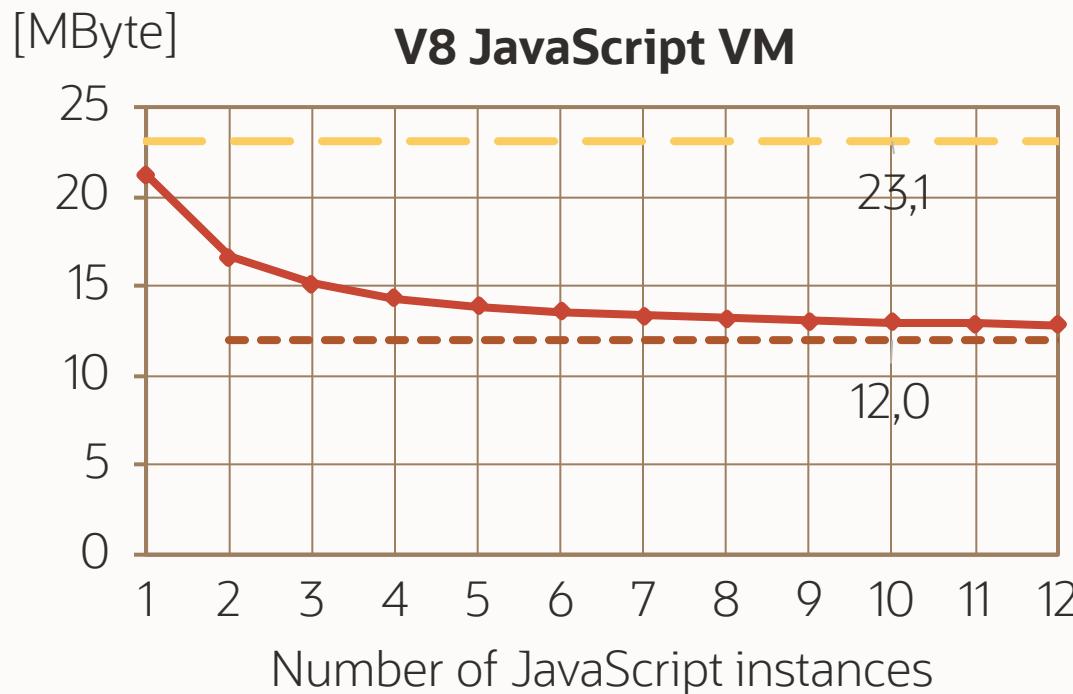
NAME	VERSION	STATUS
json-url	~> 1.0	100.00% tests pass

Graal.js Performance (versus V8)

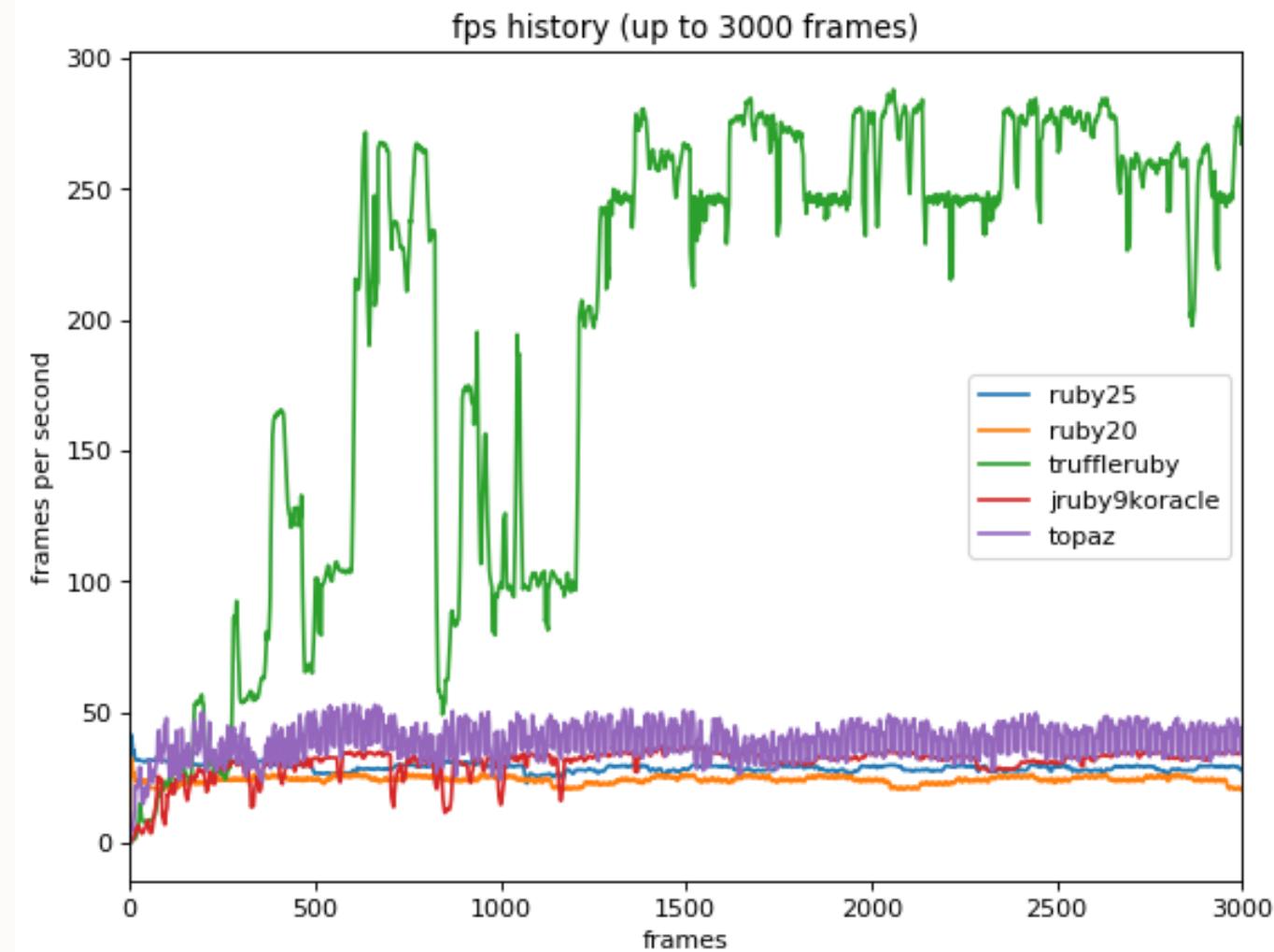


JavaScript Memory Footprint: V8 vs. GraalVM

- Memory for the first JavaScript instance: **23 MByte vs. 19 MByte**
- Memory for each additional JavaScript instance: **12 MByte vs. 0.5 MByte**



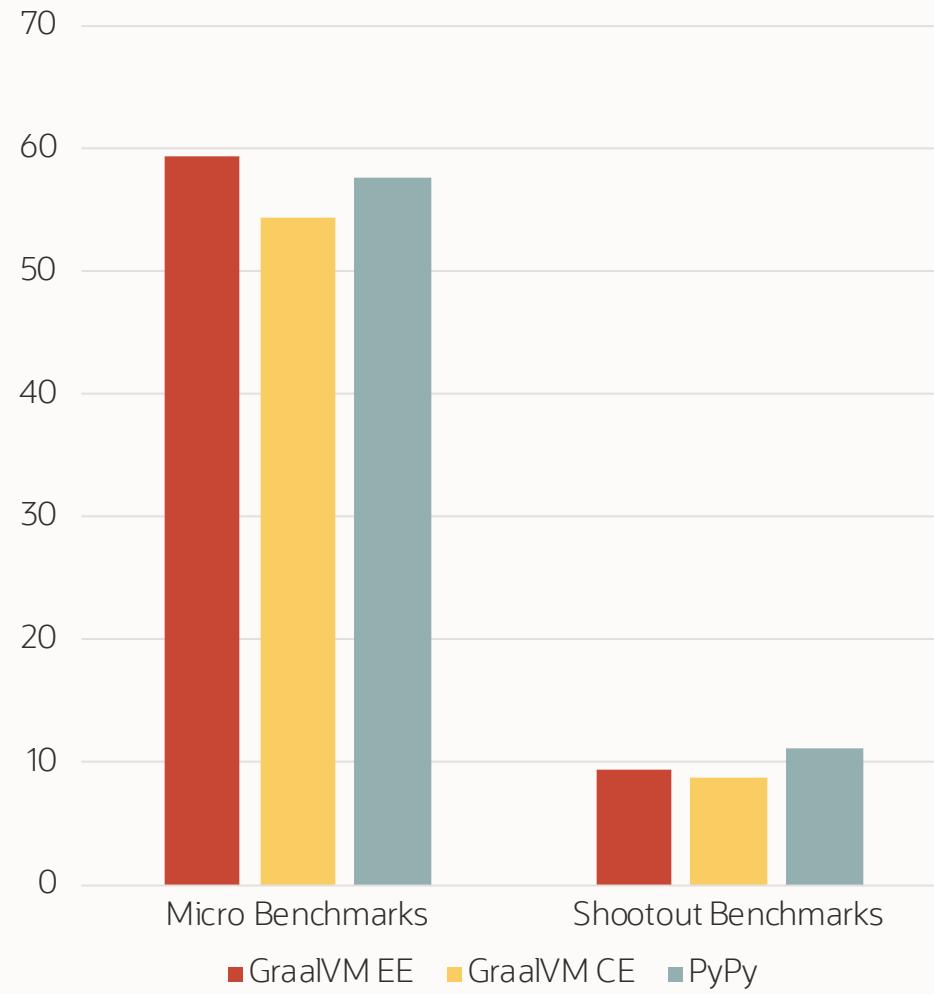
Optcarrot – TruffleRuby performance



Python Performance

Comparable to PyPy, the fastest alternative

Geomean Speedup over CPython
(more is better)



Conclusion



Conclusion

For User

- Polyglot
- Tooling
- Performance
- Compatibility

For language implementer

- AST interpreter – simple implementation
- All the tooling from the start
 - Debugging
 - Profiling
- Performance from the beginning
- AOT compilation
 - Executable
 - Embedding

Production-Ready

Java
Scala, Groovy, Kotlin
JavaScript
Node.js
Native Image
VisualVM

Experimental

Ruby
R
LLVM Toolchain

Visionary

Python
VSCode Plugin
GPU Integration
WebAssembly
LLVM Backend

The GraalVM team has created a universal virtual machine with an ambitious goal to make software engineers all over the world more productive. After years of research, GraalVM is now a production-ready technology, bringing performance improvements and additional capabilities to a wide range of languages and platforms.

Now we are expanding our GraalVM internship program with several openings, available for all our research centers. This is a great opportunity to work and learn within an international team of professionals and contribute to the project success. We are proud for our ongoing collaboration with several acknowledged universities, such as Johannes Kepler University in Linz, Charles University of Prague, Technical University of Berlin, University of Edinburgh, LaBRI, University of California (Irvine), Purdue University, Technical University of Dortmund, University of California (Davis), and the University of Lugano. Keep reading to find out about research centers, possible research topics and application process.

Opportunities for You

- Get a chance to apply your skills and knowledge to solve complex computer problems
- Contribute to an open-source technology with contributors and users all over the world
- Work in a distributed self-driven international team
- Choose one of our research centers across the globe
- Gain invaluable experience in what it is like to work at a leading global hardware and software systems innovator
- Learn from the colleagues who are industry experts and scientists

Our Research Center Locations



Possible Research Areas

- Explore new just-in-time compiler optimization phases
- Research the application of machine learning for optimizing compiler configurations
- Investigate techniques for more efficient memory usage
- Add a new language to the ecosystem of GraalVM
- Join the efforts to develop a fully meta-circular Java runtime written in Java
- Create better tooling for polyglot programming and other GraalVM features
- Discover and close attack vectors available to malicious code
- Embed GraalVM in other data storage engines
- Discover and close attack vectors available to malicious code
- Build security testing frameworks to automate assessment of a guest language attack surface

Your Skills

Given the broad range of opportunities, specific skills will depend on the specific topic. In general, if you can tick several of the following skills, we probably have a place for you:

- Strong Java programming knowledge (required)
- Fluent English communication (required)
- Experience with compiler technology
- Node.js developer experience
- Programming experience in one or more of the following languages:
 - JavaScript
 - Python
 - Ruby
 - R
 - C/C++

How to Apply

In order to apply, please send an email to graalvm-internships_wv.grp@oracle.com including the following:

- Your CV
- Description of your motivation and area of interest
- Your preferred location
- Link to your GitHub profile (optional)

The deadline for application is **November 30, 2019**.

The length of the internship can vary based on the candidate's constraints. The usual duration is between 3 and 6 months. We pay a competitive salary depending on the location of choice.

We look forward to welcoming you as a part of our team!

Internship

<https://www.graalvm.org/community/internship/>

The deadline for application is
November 30, 2019

Thank you.

Questions?

<https://www.graalvm.org/>



Safe harbor statement

The preceding is intended to outline our general product direction. 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.

The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

GraalVM Native Image technology (including SubstrateVM) is early adopter technology. It is available only under an early adopter license and remains subject to potentially significant further changes, compatibility testing and certification.





ORACLE