10 THINGS

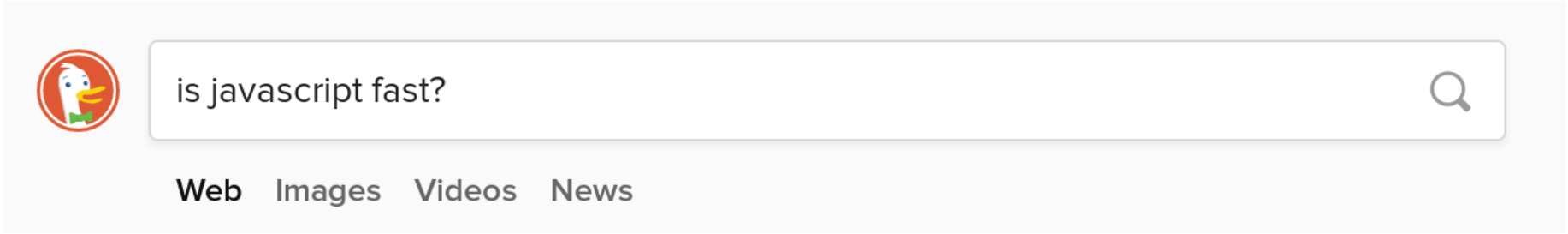
I LEARNED MAKING THE FASTEST JS SERVER RUNTIME IN THE WORLD

Paulo Lopes

Principal Software Engineer

Red Hat

@pml0pes

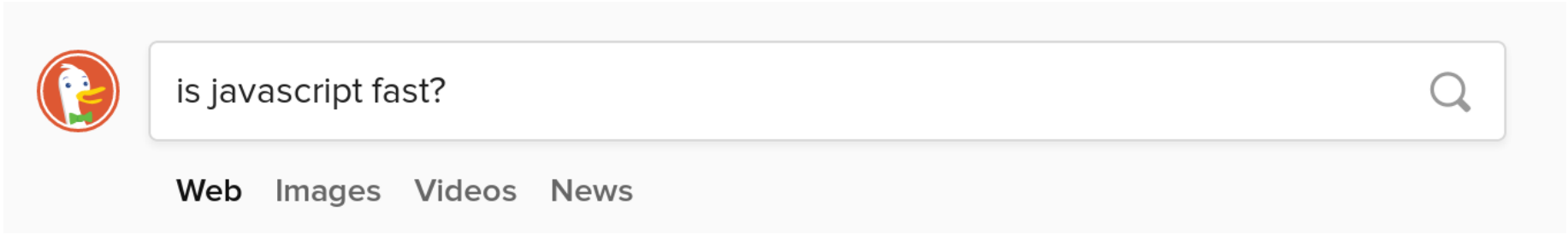


How fast is node.js? - Quora

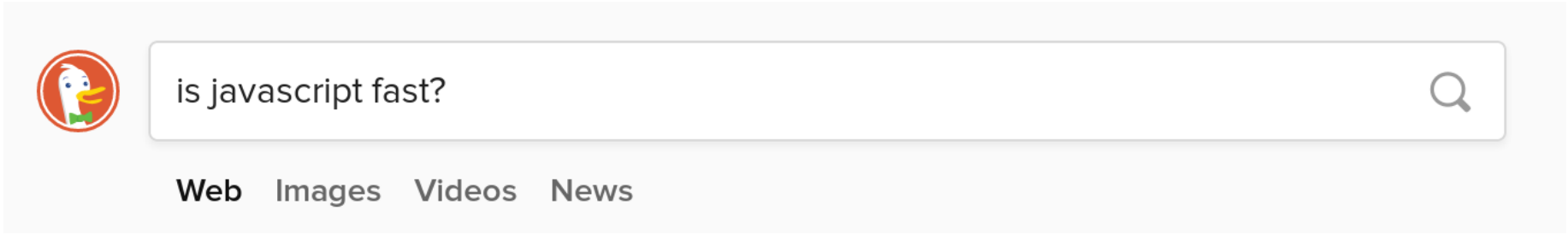
https://www.quora.com/How-fast-is-node-js

Node uses Chrome's V8 engine, which is a JIT compiler. This makes it fairly **fast**--not reliably as **fast** as a compiled language, but **fast**. Good JITs can under the right circumstances be very, very **fast**--PyPy (Python JIT) is actually as **fast** as C in some cases.

@pml0pes


is javascript fast?
Web
Images
Videos
News
Why NodeJS is so fast? - Eugene Obrezkov
https://blog.ghaiklor.com/why-nodejs-is-so-fast-a0ff67858f48
As soon as you understand what's going on in NodeJS server when it processes the requests, you will realise why it is so fast. But before talking about NodeJS, let's look at how request handling is done in other languages. PHP is the best example because it's very popular and unoptimised. PHP drawbacks

@pml0pes

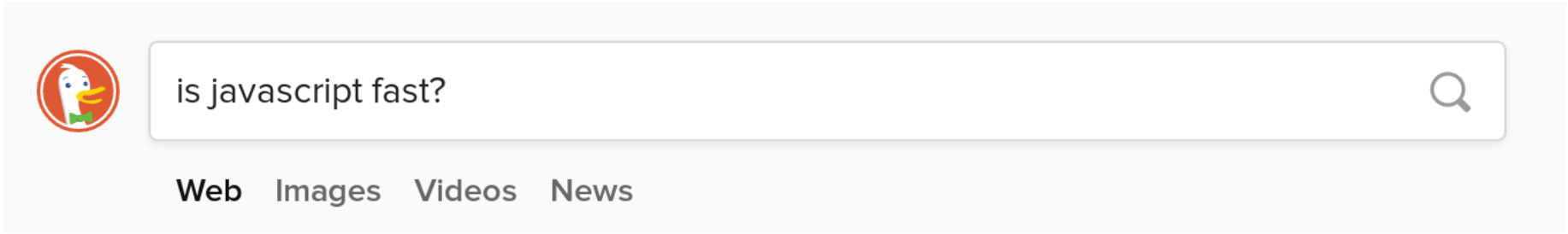


javascript - Why node.js is fast when it's single threaded ...

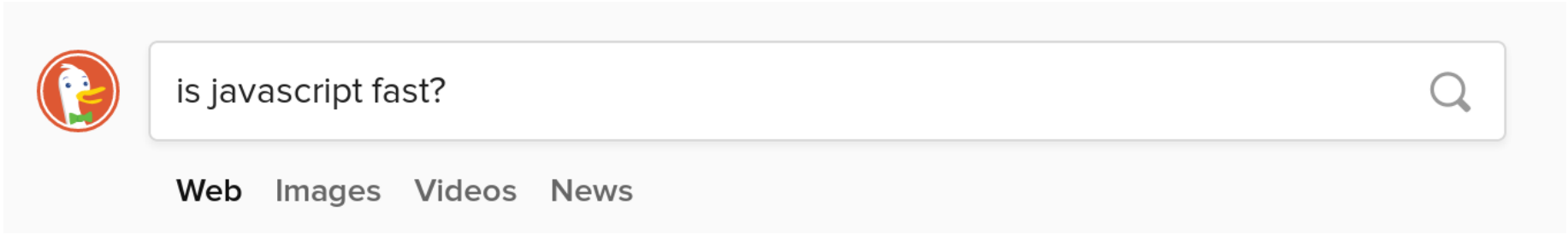
https://stackoverflow.com/questions/30638549/why-node-js-is-fast-when-its-single-th...

Despite being single threaded, how is **node.js is** faster? I haven't run any tests to find statistics, but while digging in the **node.js** forums, I find everyone says it's faster and more lightweight. But no matter how light weight it **is**, how can a single threaded server be faster than multi thread ones?

@pml0pes


is javascript fast?
Web
Images
Videos
News
How fast is Javascript compared to Java? - Stack Overflow
https://stackoverflow.com/questions/3723374/how-fast-is-javascript-compared-to-java
Node.js shines when it comes to a huge amount of short connections, so javascript is a really great fit for that sort of thing. While node.js is absolutely drenched in awesome, being the new hotness really doesn't mean it is the best at everything, no matter what the hype says.

@pml0pes


is javascript fast?
Web
Images
Videos
News
StrongLoop - What Makes Node.js Faster Than Java?
https://strongloop.com/strongblog/node-js-is-faster-than-java/
While Java or Node or something else may win a benchmark, no server has the non-blocking ecosystem of Node.js today. Over 50k modules all written in the async style, ready to use. Countless code examples strewn about the web. Lessons and tutorials all using the async style.

@pml0pes

DO WE TRUST THE INTERNET?

@pml0pes

DO WE TRUST THE INTERNET?

😯 I DON'T!

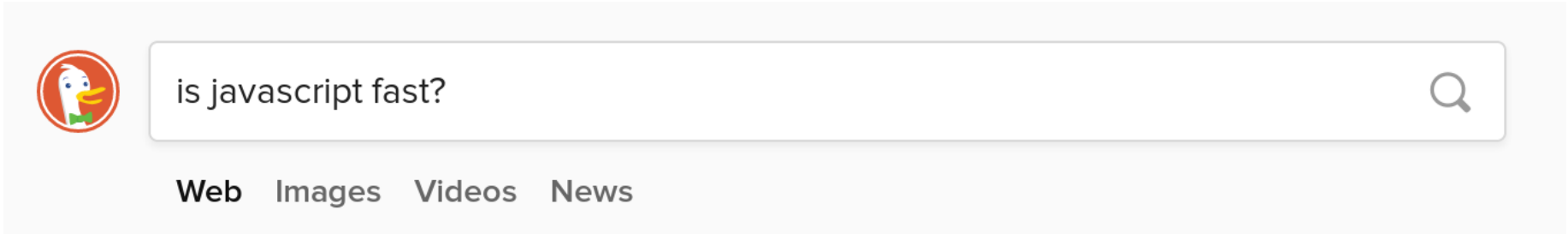
@pml0pes

engineering noun

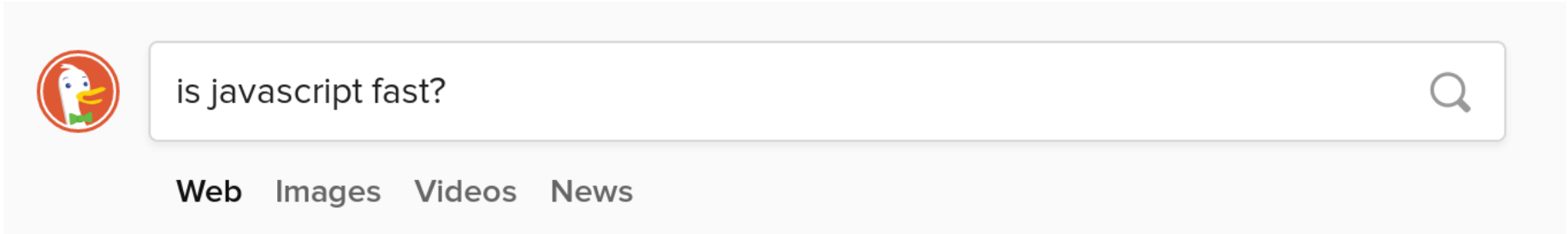
*en·gi·neer·ing / \ ˌen-jə-ˈnir-iŋ *

> *the application of science and mathematics by which the properties of matter and the sources of energy in nature are made useful to people.*
>
> *(Merriam-Webster)*

@pml0pes



@pml0pes



@pml0pes

#1:

DEFINE SERVER

@pml0pes

server noun

serv·er | \ ˈsər-vər

> *a computer in a network that is used to provide services (such as access to files or shared peripherals or the routing of e-mail) to other computers in the network.*
>
> *(Wikipedia)*

@pml0pes

#2:

DEFINE FAST

@pml0pes

Site Reliability Engineering

is a discipline that incorporates aspects of software engineering and applies them to infrastructure and operations problems. The main goals are to create ultra-scalable and highly reliable software systems

https://landing.google.com/sre/books/

Site Reliability Engineering metrics

- Rate
- Errors
- Latency
- *Saturation*
- *Utilization*

Site Reliability Engineering metrics

- Rate — *Request rate, in requests/sec*
- Errors
- Latency
- *Saturation*
- *Utilization*

Site Reliability Engineering metrics

- Rate — *Request rate, in requests/sec*
- Errors — *Error rate, in errors/sec*
- Latency
- *Saturation*
- *Utilization*

Site Reliability Engineering metrics

- Rate — *Request rate, in requests/sec*
- Errors — *Error rate, in errors/sec*
- Latency — *Response time, including queue/wait time*
- *Saturation*
- *Utilization*

#3:

UNDERSTAND THE ENVIRONMENT/LIFECYCLE

@pml0pes

Server Application

@pml0pes

Server Application

- long running process

Server Application

- long running process
- deployed on cloud or bare metal

Server Application

- long running process
- deployed on cloud or bare metal
- attached to a fast network

Server Application

- long running process
- deployed on cloud or bare metal
- attached to a fast network
- "enough" CPU/Memory

#5:

MEASURE / BENCHMARK

@pml0pes

BENCHMARKING IS HARD

- *Meaningful benchmarks are even harder*

BENCHMARKING IS HARD

- *Meaningful benchmarks are even harder*
- Techempower Framework Benchmarks

BENCHMARKING IS HARD

- *Meaningful benchmarks are even harder*
- Techempower Framework Benchmarks
  - Contributors: **512**

BENCHMARKING IS HARD

- *Meaningful benchmarks are even harder*
- Techempower Framework Benchmarks
  - Contributors: **512**
  - Pull Requests: **3824**

BENCHMARKING IS HARD

- *Meaningful benchmarks are even harder*
- Techempower Framework Benchmarks
  - Contributors: **512**
  - Pull Requests: **3824**
  - Commits: **10900**

BENCHMARKING IS HARD

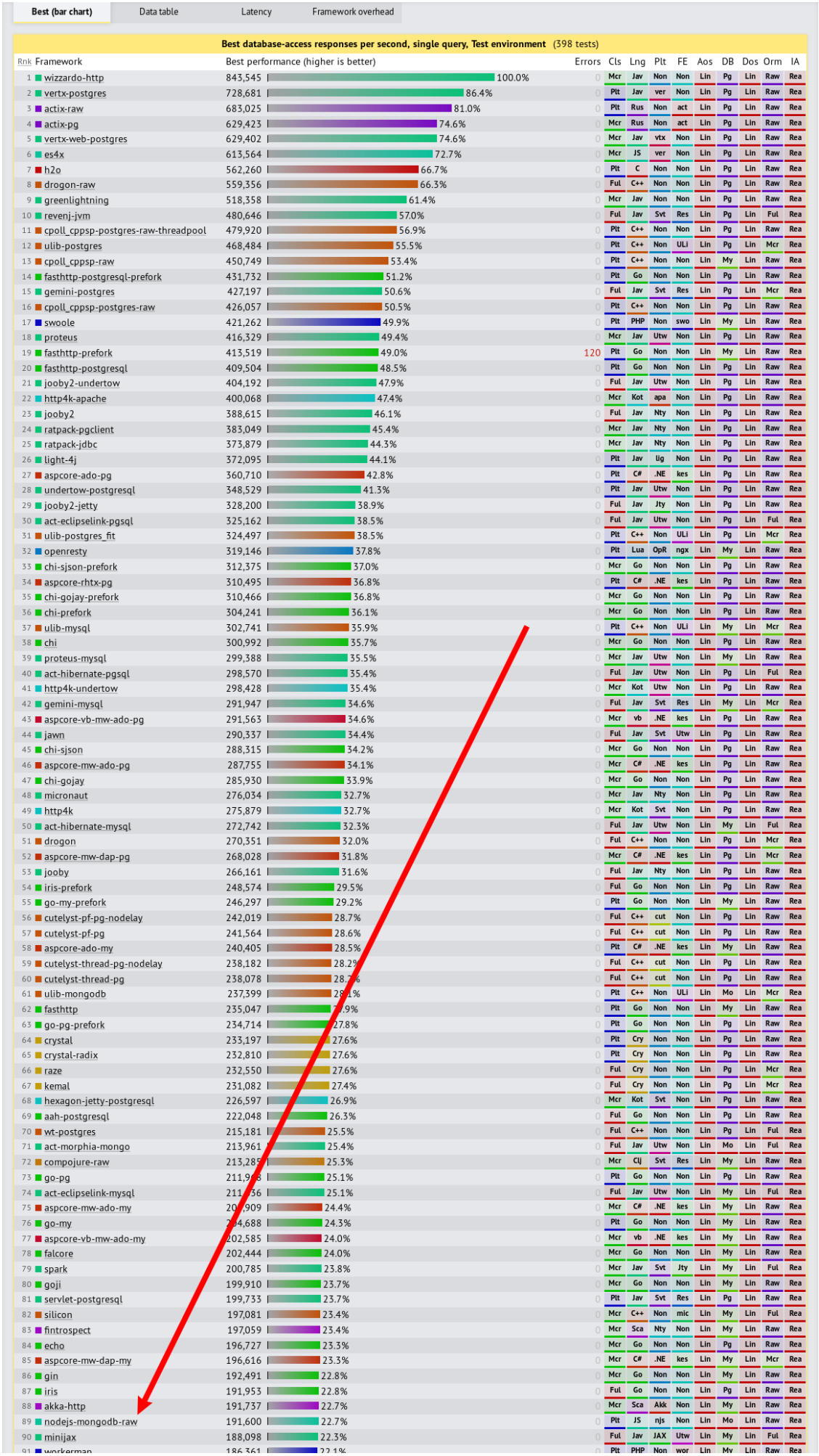
- *Meaningful benchmarks are even harder*
- Techempower Framework Benchmarks
  - Contributors: **512**
  - Pull Requests: **3824**
  - Commits: **10900**

https://github.com/TechEmpower/FrameworkBenchmarks/

Best (bar chart) | Data table | Latency | Framework overhead

Best database-access responses per second, single query, Test environment (398 tests)

Rnk	Framework	Best performance (higher is better)	Errors	Cls	Lng	Plt	FE	Aos	DB	Dos	Orm	IA
1	wizzardo-http	843,545 100.0%	0	Mcr	Jav	Non	Non	Lin	Pg	Lin	Raw	Rea
2	vertx-postgres	728,681 86.4%	0	Plt	Jav	ver	Non	Lin	Pg	Lin	Raw	Rea
3	actix-raw	683,025 81.0%	0	Plt	Rus	Non	act	Lin	Pg	Lin	Raw	Rea
4	actix-pg	629,423 74.6%	0	Mcr	Rus	Non	act	Lin	Pg	Lin	Raw	Rea
5	vertx-web-postgres	629,402 74.6%	0	Mcr	Jav	vtx	Non	Lin	Pg	Lin	Raw	Rea
6	es4x	613,564 72.7%	0	Mcr	JS	ver	Non	Lin	Pg	Lin	Raw	Rea
7	h2o	562,260 66.7%	0	Plt	C	Non	Non	Lin	Pg	Lin	Raw	Rea
8	drogon-raw	559,356 66.3%	0	Ful	C++	Non	Non	Lin	Pg	Lin	Raw	Rea
9	greenlightning	518,358 61.4%	0	Mcr	Jav	Non	Non	Lin	Pg	Lin	Raw	Rea
10	revenj-jvm	480,646 57.0%	0	Ful	Jav	Svt	Res	Lin	Pg	Lin	Ful	Rea
11	cpoll_cppsp-postgres-raw-threadpool	479,920 56.9%	0	Plt	C++	Non	Non	Lin	Pg	Lin	Raw	Rea
12	ulib-postgres	468,484 55.5%	0	Plt	C++	Non	ULi	Lin	Pg	Lin	Mcr	Rea
13	cpoll_cppsp-raw	450,749 53.4%	0	Plt	C++	Non	Non	Lin	My	Lin	Raw	Rea
14	fasthttp-postgresql-prefork	431,732 51.2%	0	Plt	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
15	gemini-postgres	427,197 50.6%	0	Ful	Jav	Svt	Res	Lin	Pg	Lin	Mcr	Rea
16	cpoll_cppsp-postgres-raw	426,057 50.5%	0	Plt	C++	Non	Non	Lin	Pg	Lin	Raw	Rea
17	swoole	421,262 49.9%	0	Plt	PHP	Non	swo	Lin	My	Lin	Raw	Rea
18	proteus	416,329 49.4%	0	Mcr	Jav	Utw	Non	Lin	Pg	Lin	Raw	Rea
19	fasthttp-prefork	413,519 49.0%	120	Plt	Go	Non	Non	Lin	My	Lin	Raw	Rea
20	fasthttp-postgresql	409,504 48.5%	0	Plt	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
21	jooby2-undertow	404,192 47.9%	0	Ful	Jav	Utw	Non	Lin	Pg	Lin	Raw	Rea
22	http4k-apache	400,068 47.4%	0	Mcr	Kot	apa	Non	Lin	Pg	Lin	Raw	Rea
23	jooby2	388,615 46.1%	0	Ful	Jav	Nty	Non	Lin	Pg	Lin	Raw	Rea
24	ratpack-pgclient	383,049 45.4%	0	Mcr	Jav	Nty	Non	Lin	Pg	Lin	Raw	Rea
25	ratpack-jdbc	373,879 44.3%	0	Mcr	Jav	Nty	Non	Lin	Pg	Lin	Raw	Rea
26	light-4j	372,095 44.1%	0	Plt	Jav	lig	Non	Lin	Pg	Lin	Raw	Rea
27	aspcore-ado-pg	360,710 42.8%	0	Plt	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
28	undertow-postgresql	348,529 41.3%	0	Plt	Jav	Utw	Non	Lin	Pg	Lin	Raw	Rea
29	jooby2-jetty	328,200 38.9%	0	Ful	Jav	Jty	Non	Lin	Pg	Lin	Raw	Rea
30	act-eclipselink-pgsql	325,162 38.5%	0	Ful	Jav	Utw	Non	Lin	Pg	Lin	Ful	Rea
31	ulib-postgres_fit	324,497 38.5%	0	Plt	C++	Non	ULi	Lin	Pg	Lin	Mcr	Rea
32	openresty	319,146 37.8%	0	Plt	Lua	OpR	ngx	Lin	My	Lin	Raw	Rea
33	chi-sjson-prefork	312,375 37.0%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
34	aspcore-rhtx-pg	310,495 36.8%	0	Plt	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
35	chi-gojay-prefork	310,466 36.8%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
36	chi-prefork	304,241 36.1%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
37	ulib-mysql	302,741 35.9%	0	Plt	C++	Non	ULi	Lin	My	Lin	Mcr	Rea
38	chi	300,992 35.7%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
39	proteus-mysql	299,388 35.5%	0	Mcr	Jav	Utw	Non	Lin	My	Lin	Raw	Rea
40	act-hibernate-pgsql	298,570 35.4%	0	Ful	Jav	Utw	Non	Lin	Pg	Lin	Ful	Rea
41	http4k-undertow	298,428 35.4%	0	Mcr	Kot	Utw	Non	Lin	Pg	Lin	Raw	Rea
42	gemini-mysql	291,947 34.6%	0	Ful	Jav	Svt	Res	Lin	My	Lin	Mcr	Rea
43	aspcore-vb-mw-ado-pg	291,563 34.6%	0	Mcr	vb	.NE	kes	Lin	Pg	Lin	Raw	Rea
44	jawn	290,337 34.4%	0	Ful	Jav	Svt	Utw	Lin	Pg	Lin	Raw	Rea
45	chi-sjson	288,315 34.2%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
46	aspcore-mw-ado-pg	287,755 34.1%	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
47	chi-gojay	285,930 33.9%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
48	micronaut	276,034 32.7%	0	Mcr	Jav	Nty	Non	Lin	Pg	Lin	Raw	Rea
49	http4k	275,879 32.7%	0	Mcr	Kot	Svt	Non	Lin	Pg	Lin	Raw	Rea
50	act-hibernate-mysql	272,742 32.3%	0	Ful	Jav	Utw	Non	Lin	My	Lin	Ful	Rea
51	drogon	270,351 32.0%	0	Ful	C++	Non	Non	Lin	Pg	Lin	Mcr	Rea
52	aspcore-mw-dap-pg	268,028 31.8%	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Mcr	Rea
53	jooby	266,161 31.6%	0	Ful	Jav	Nty	Non	Lin	Pg	Lin	Raw	Rea
54	iris-prefork	248,574 29.5%	0	Ful	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
55	go-my-prefork	246,297 29.2%	0	Plt	Go	Non	Non	Lin	My	Lin	Raw	Rea
56	cutelyst-pf-pg-nodelay	242,019 28.7%	0	Ful	C++	cut	Non	Lin	Pg	Lin	Raw	Rea
57	cutelyst-pf-pg	241,564 28.6%	0	Ful	C++	cut	Non	Lin	Pg	Lin	Raw	Rea
58	aspcore-ado-my	240,405 28.5%	0	Plt	C#	.NE	kes	Lin	My	Lin	Raw	Rea
59	cutelyst-thread-pg-nodelay	238,182 28.2%	0	Ful	C++	cut	Non	Lin	Pg	Lin	Raw	Rea
60	cutelyst-thread-pg	238,078 28.2%	0	Ful	C++	cut	Non	Lin	Pg	Lin	Raw	Rea
61	ulib-mongodb	237,399 28.1%	0	Plt	C++	Non	ULi	Lin	Mo	Lin	Mcr	Rea
62	fasthttp	235,047 27.9%	0	Plt	Go	Non	Non	Lin	My	Lin	Raw	Rea
63	go-pg-prefork	234,714 27.8%	0	Plt	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
64	crystal	233,197 27.6%	0	Plt	Cry	Non	Non	Lin	Pg	Lin	Raw	Rea
65	crystal-radix	232,810 27.6%	0	Plt	Cry	Non	Non	Lin	Pg	Lin	Raw	Rea
66	raze	232,550 27.6%	0	Ful	Cry	Non	Non	Lin	Pg	Lin	Mcr	Rea
67	kemal	231,082 27.4%	0	Ful	Cry	Non	Non	Lin	Pg	Lin	Mcr	Rea
68	hexagon-jetty-postgresql	226,597 26.9%	0	Mcr	Kot	Svt	Non	Lin	Pg	Lin	Raw	Rea
69	aah-postgresql	222,048 26.3%	0	Ful	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
70	wt-postgres	215,181 25.5%	0	Ful	C++	Non	Non	Lin	Pg	Lin	Ful	Rea
71	act-morphia-mongo	213,961 25.4%	0	Ful	Jav	Utw	Non	Lin	Mo	Lin	Ful	Rea
72	compojure-raw	213,285 25.3%	0	Mcr	Clj	Svt	Res	Lin	My	Lin	Raw	Rea
73	go-pg	211,968 25.1%	0	Plt	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
74	act-eclipselink-mysql	211,636 25.1%	0	Ful	Jav	Utw	Non	Lin	My	Lin	Ful	Rea
75	aspcore-mw-ado-my	205,909 24.4%	0	Mcr	C#	.NE	kes	Lin	My	Lin	Raw	Rea
76	go-my	204,688 24.3%	0	Plt	Go	Non	Non	Lin	My	Lin	Raw	Rea
77	aspcore-vb-mw-ado-my	202,585 24.0%	0	Mcr	vb	.NE	kes	Lin	My	Lin	Raw	Rea
78	falcore	202,444 24.0%	0	Mcr	Go	Non	Non	Lin	My	Lin	Raw	Rea
79	spark	200,785 23.8%	0	Mcr	Jav	Svt	Jty	Lin	My	Lin	Ful	Rea
80	goji	199,910 23.7%	0	Mcr	Go	Non	Non	Lin	My	Lin	Raw	Rea
81	servlet-postgresql	199,733 23.7%	0	Plt	Jav	Svt	Res	Lin	Pg	Lin	Raw	Rea
82	silicon	197,081 23.4%	0	Mcr	C++	Non	mic	Lin	My	Lin	Ful	Rea
83	fintrospect	197,059 23.4%	0	Mcr	Sca	Nty	Non	Lin	My	Lin	Raw	Rea
84	echo	196,727 23.3%	0	Mcr	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
85	aspcore-mw-dap-my	196,616 23.3%	0	Mcr	C#	.NE	kes	Lin	My	Lin	Mcr	Rea
86	gin	192,491 22.8%	0	Mcr	Go	Non	Non	Lin	My	Lin	Raw	Rea
87	iris	191,953 22.8%	0	Ful	Go	Non	Non	Lin	Pg	Lin	Raw	Rea
88	akka-http	191,737 22.7%	0	Mcr	Sca	Akk	Non	Lin	My	Lin	Raw	Rea
89	nodejs-mongodb-raw	191,600 22.7%	0	Plt	JS	njs	Non	Lin	Mo	Lin	Raw	Rea
90	minijax	188,098 22.3%	0	Ful	Jav	JAX	Utw	Lin	My	Lin	Ful	Rea
91	workerman	186,361 22.1%	0	Plt	PHP	Non	wor	Lin	My	Lin	Raw	Rea


Best (bar chart)
Data table
Latency
Framework overhead
Best database-access responses per second, single query, Test environment (398 tests)
Rnk Framework
Best performance (higher is better)
Errors
Cls Lng Plt FE Aos DB Dos Orm IA
1 wizzardo-http 843,545 100.0% 0 Mcr Jav Non Non Lin Pg Lin Raw Rea
2 vertx-postgres 728,681 86.4% 0 Plt Jav ver Non Lin Pg Lin Raw Rea
3 actix-raw 683,025 81.0% 0 Plt Rus Non act Lin Pg Lin Raw Rea
4 actix-pg 629,423 74.6% 0 Mcr Rus Non act Lin Pg Lin Raw Rea
5 vertx-web-postgres 629,402 74.6% 0 Mcr Jav vtx Non Lin Pg Lin Raw Rea
6 es4x 613,564 72.7% 0 Mcr JS ver Non Lin Pg Lin Raw Rea
7 h2o 562,260 66.7% 0 Plt C Non Non Lin Pg Lin Raw Rea
8 drogon-raw 559,356 66.3% 0 Ful C++ Non Non Lin Pg Lin Raw Rea
9 greenlightning 518,358 61.4% 0 Mcr Jav Non Non Lin Pg Lin Raw Rea
10 revenj-jvm 480,646 57.0% 0 Ful Jav Svt Res Lin Pg Lin Ful Rea
11 cpoll_cppsp-postgres-raw-threadpool 479,920 56.9% 0 Plt C++ Non Non Lin Pg Lin Raw Rea
12 ulib-postgres 468,484 55.5% 0 Plt C++ Non ULi Lin Pg Lin Mcr Rea
13 cpoll_cppsp-raw 450,749 53.4% 0 Plt C++ Non Non Lin My Lin Raw Rea
14 fasthttp-postgresql-prefork 431,732 51.2% 0 Plt Go Non Non Lin Pg Lin Raw Rea
15 gemini-postgres 427,197 50.6% 0 Ful Jav Svt Res Lin Pg Lin Mcr Rea
16 cpoll_cppsp-postgres-raw 426,057 50.5% 0 Plt C++ Non Non Lin Pg Lin Raw Rea
17 swoole 421,262 49.9% 0 Plt PHP Non swo Lin My Lin Raw Rea
18 proteus 416,329 49.4% 0 Mcr Jav Utw Non Lin Pg Lin Raw Rea
19 fasthttp-prefork 413,519 49.0% 120 Plt Go Non Non Lin My Lin Raw Rea
20 fasthttp-postgresql 409,504 48.5% 0 Plt Go Non Non Lin Pg Lin Raw Rea
21 jooby2-undertow 404,192 47.9% 0 Ful Jav Utw Non Lin Pg Lin Raw Rea
22 http4k-apache 400,068 47.4% 0 Mcr Kot apa Non Lin Pg Lin Raw Rea
23 jooby2 388,615 46.1% 0 Ful Jav Nty Non Lin Pg Lin Raw Rea
24 ratpack-pgclient 383,049 45.4% 0 Mcr Jav Nty Non Lin Pg Lin Raw Rea
25 ratpack-jdbc 373,879 44.3% 0 Mcr Jav Nty Non Lin Pg Lin Raw Rea
26 light-4j 372,095 44.1% 0 Plt Jav lig Non Lin Pg Lin Raw Rea
27 aspcore-ado-pg 360,710 42.8% 0 Plt C# .NE kes Lin Pg Lin Raw Rea
28 undertow-postgresql 348,529 41.3% 0 Plt Jav Utw Non Lin Pg Lin Raw Rea
29 jooby2-jetty 328,200 38.9% 0 Ful Jav Jty Non Lin Pg Lin Raw Rea
30 act-eclipselink-pgsql 325,162 38.5% 0 Ful Jav Utw Non Lin Pg Lin Ful Rea
31 ulib-postgres_fit 324,497 38.5% 0 Plt C++ Non ULi Lin Pg Lin Mcr Rea
32 openresty 319,146 37.8% 0 Plt Lua OpR ngx Lin My Lin Raw Rea
33 chi-sjson-prefork 312,375 37.0% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
34 aspcore-rhtx-pg 310,495 36.8% 0 Plt C# .NE kes Lin Pg Lin Raw Rea
35 chi-gojay-prefork 310,466 36.8% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
36 chi-prefork 304,241 36.1% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
37 ulib-mysql 302,741 35.9% 0 Plt C++ Non ULi Lin My Lin Mcr Rea
38 chi 300,992 35.7% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
39 proteus-mysql 299,388 35.5% 0 Mcr Jav Utw Non Lin My Lin Raw Rea
40 act-hibernate-pgsql 298,570 35.4% 0 Ful Jav Utw Non Lin Pg Lin Ful Rea
41 http4k-undertow 298,428 35.4% 0 Mcr Kot Utw Non Lin Pg Lin Raw Rea
42 gemini-mysql 291,947 34.6% 0 Ful Jav Svt Res Lin My Lin Mcr Rea
43 aspcore-vb-mw-ado-pg 291,563 34.6% 0 Mcr vb .NE kes Lin Pg Lin Raw Rea
44 jawn 290,337 34.4% 0 Ful Jav Svt Utw Lin Pg Lin Raw Rea
45 chi-sjson 288,315 34.2% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
46 aspcore-mw-ado-pg 287,755 34.1% 0 Mcr C# .NE kes Lin Pg Lin Raw Rea
47 chi-gojay 285,930 33.9% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
48 micronaut 276,034 32.7% 0 Mcr Jav Nty Non Lin Pg Lin Raw Rea
49 http4k 275,879 32.7% 0 Mcr Kot Svt Non Lin Pg Lin Raw Rea
50 act-hibernate-mysql 272,742 32.3% 0 Ful Jav Utw Non Lin My Lin Ful Rea
51 drogon 270,351 32.0% 0 Ful C++ Non Non Lin Pg Lin Mcr Rea
52 aspcore-mw-dap-pg 268,028 31.8% 0 Mcr C# .NE kes Lin Pg Lin Mcr Rea
53 jooby 266,161 31.6% 0 Ful Jav Nty Non Lin Pg Lin Raw Rea
54 iris-prefork 248,574 29.5% 0 Ful Go Non Non Lin Pg Lin Raw Rea
55 go-my-prefork 246,297 29.2% 0 Plt Go Non Non Lin My Lin Raw Rea
56 cutelyst-pf-pg-nodelay 242,019 28.7% 0 Ful C++ cut Non Lin Pg Lin Raw Rea
57 cutelyst-pf-pg 241,564 28.6% 0 Ful C++ cut Non Lin Pg Lin Raw Rea
58 aspcore-ado-my 240,405 28.5% 0 Plt C# .NE kes Lin My Lin Raw Rea
59 cutelyst-thread-pg-nodelay 238,182 28.2% 0 Ful C++ cut Non Lin Pg Lin Raw Rea
60 cutelyst-thread-pg 238,078 28.2% 0 Ful C++ cut Non Lin Pg Lin Raw Rea
61 ulib-mongodb 237,399 28.1% 0 Plt C++ Non ULi Lin Mo Lin Mcr Rea
62 fasthttp 235,047 27.9% 0 Plt Go Non Non Lin My Lin Raw Rea
63 go-pg-prefork 234,714 27.8% 0 Plt Go Non Non Lin Pg Lin Raw Rea
64 crystal 233,197 27.6% 0 Plt Cry Non Non Lin Pg Lin Raw Rea
65 crystal-radix 232,810 27.6% 0 Plt Cry Non Non Lin Pg Lin Raw Rea
66 raze 232,550 27.6% 0 Ful Cry Non Non Lin Pg Lin Mcr Rea
67 kemal 231,082 27.4% 0 Ful Cry Non Non Lin Pg Lin Mcr Rea
68 hexagon-jetty-postgresql 226,597 26.9% 0 Mcr Kot Svt Non Lin Pg Lin Raw Rea
69 aah-postgresql 222,048 26.3% 0 Ful Go Non Non Lin Pg Lin Raw Rea
70 wt-postgres 215,181 25.5% 0 Ful C++ Non Non Lin Pg Lin Ful Rea
71 act-morphia-mongo 213,961 25.4% 0 Ful Jav Utw Non Lin Mo Lin Ful Rea
72 compojure-raw 213,285 25.3% 0 Mcr Clj Svt Res Lin My Lin Raw Rea
73 go-pg 211,968 25.1% 0 Plt Go Non Non Lin Pg Lin Raw Rea
74 act-eclipselink-mysql 211,436 25.1% 0 Ful Jav Utw Non Lin My Lin Ful Rea
75 aspcore-mw-ado-my 206,909 24.4% 0 Mcr C# .NE kes Lin My Lin Raw Rea
76 go-my 204,688 24.3% 0 Plt Go Non Non Lin My Lin Raw Rea
77 aspcore-vb-mw-ado-my 202,585 24.0% 0 Mcr vb .NE kes Lin My Lin Raw Rea
78 falcore 202,444 24.0% 0 Mcr Go Non Non Lin My Lin Raw Rea
79 spark 200,785 23.8% 0 Mcr Jav Svt Jty Lin My Lin Ful Rea
80 goji 199,910 23.7% 0 Mcr Go Non Non Lin My Lin Raw Rea
81 servlet-postgresql 199,733 23.7% 0 Plt Jav Svt Res Lin Pg Lin Raw Rea
82 silicon 197,081 23.4% 0 Mcr C++ Non mic Lin My Lin Ful Rea
83 fintrospect 197,059 23.4% 0 Mcr Sca Nty Non Lin My Lin Raw Rea
84 echo 196,727 23.3% 0 Mcr Go Non Non Lin Pg Lin Raw Rea
85 aspcore-mw-dap-my 196,616 23.3% 0 Mcr C# .NE kes Lin My Lin Mcr Rea
86 gin 192,491 22.8% 0 Mcr Go Non Non Lin My Lin Raw Rea
87 iris 191,953 22.8% 0 Ful Go Non Non Lin Pg Lin Raw Rea
88 akka-http 191,737 22.7% 0 Mcr Sca Akk Non Lin My Lin Raw Rea
89 nodejs-mongodb-raw 191,600 22.7% 0 Plt JS njs Non Lin Mo Lin Raw Rea
90 minijax 188,098 22.3% 0 Ful Jav JAX Utw Lin My Lin Ful Rea
91 workerman 186,361 22.1% 0 Plt PHP Non wor Lin My Lin Raw Rea

#6:

LOOK UNDER THE HOOD

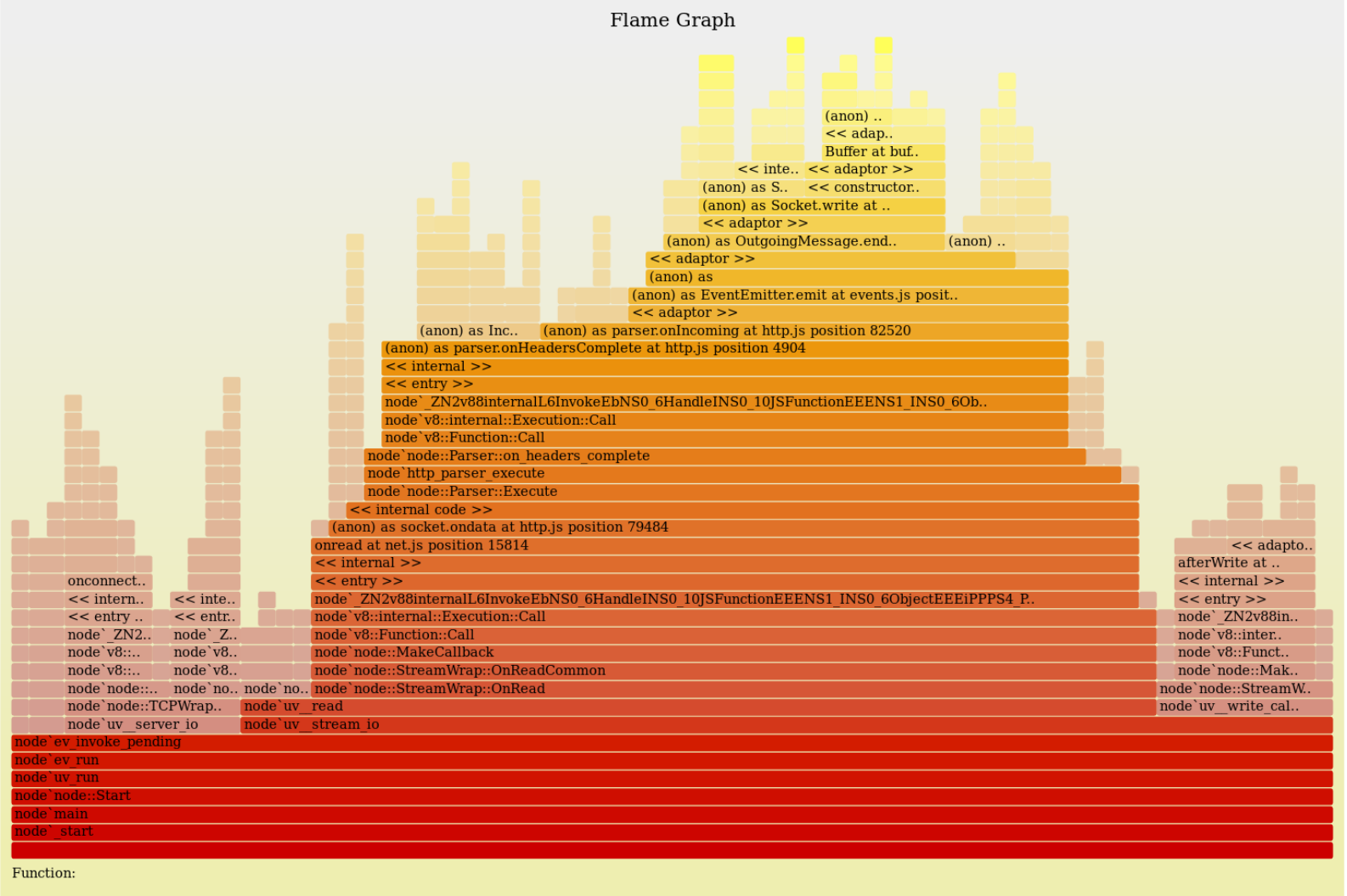
@pml0pes

```
const cluster = require('cluster'),
  numCPUs = require('os').cpus().length,
  express = require('express');

if (cluster.isMaster) {
  for (let i = 0; i < numCPUs; i++)
    cluster.fork();
} else {
  const app = module.exports = express();
  app.get('/plaintext', (req, res) =>
    res
      .header('Content-Type', 'text/plain')
      .send('Hello, World!'));
}
```

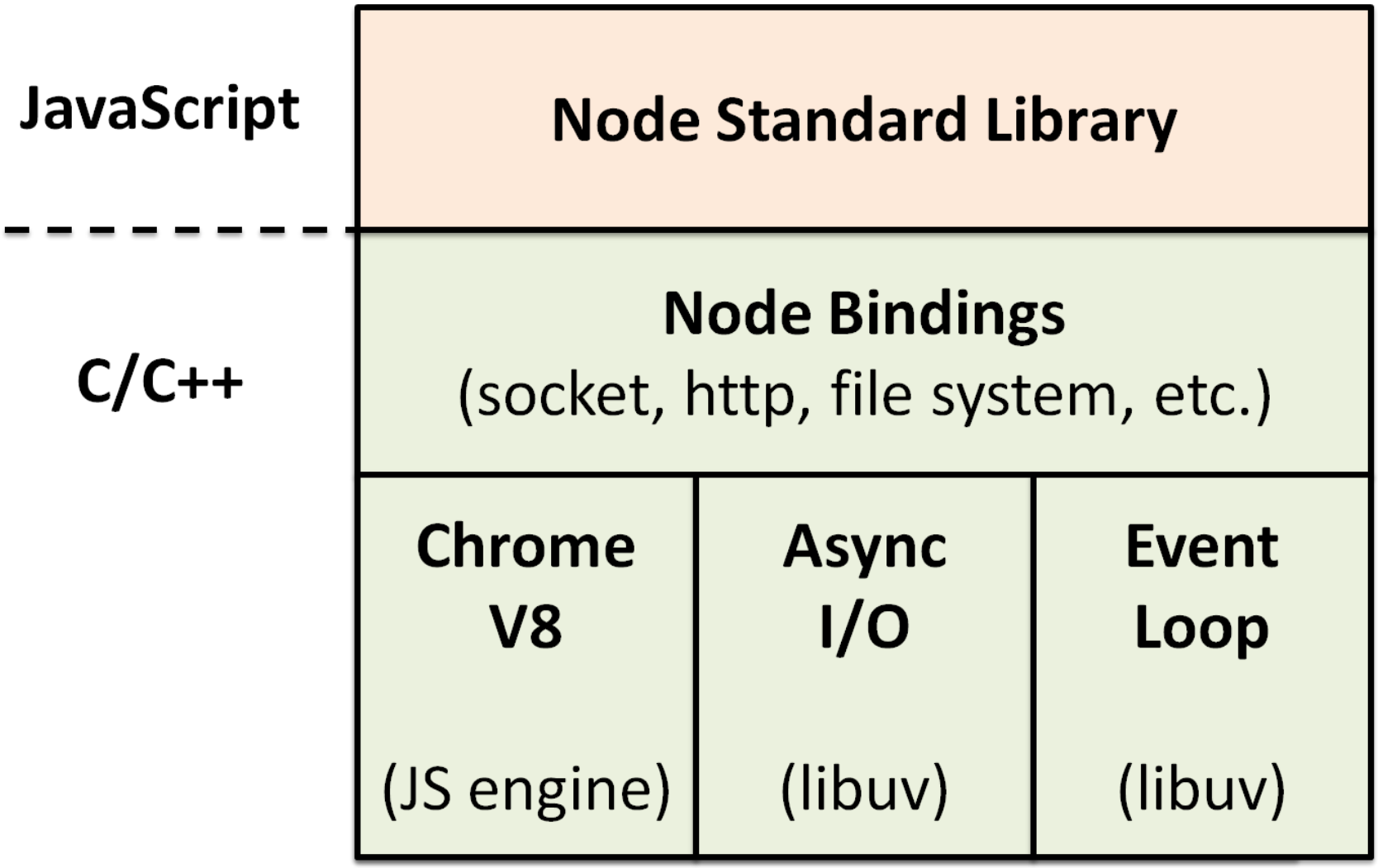
@pml0pes

ANALIZE



@pml0pes

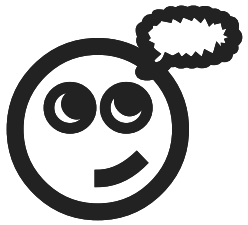
OBSERVE



@pml0pes

YOU CAN ONLY OPTIMIZE

@pml0pes



@pml0pes

#7:

JAVASCRIPT RUNTIMES

@pml0pes

V8 JavaScript engine

> *Speed up real-world performance for modern JavaScript, and enable developers to build a faster future web.*

MORE ENGINES

@pml0pes

MORE ENGINES

- ChakraCore

MORE ENGINES

- ChakraCore
- SpiderMonkey

MORE ENGINES

- ChakraCore
- SpiderMonkey
- ScriptCore

MORE ENGINES

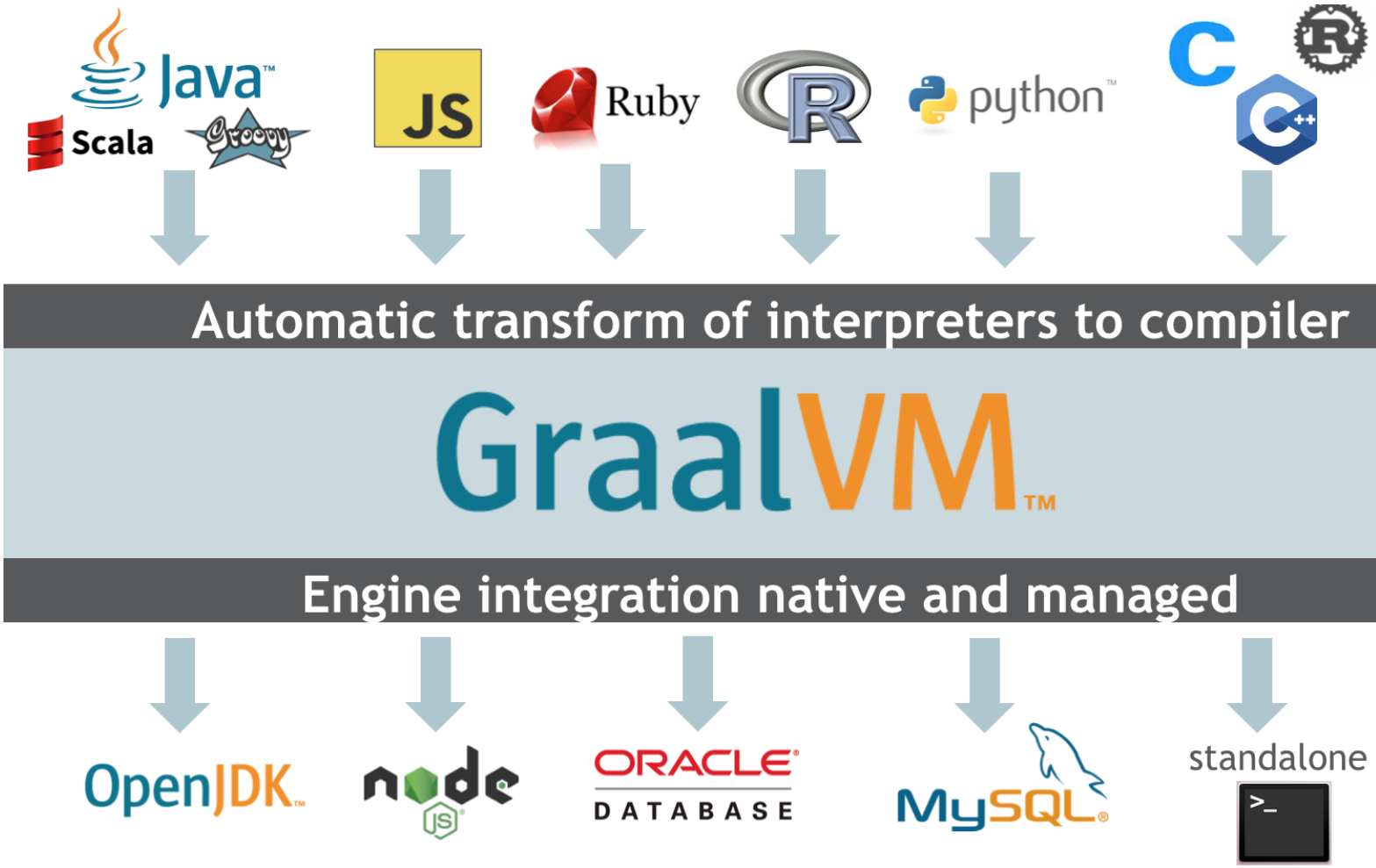
- ChakraCore
- SpiderMonkey
- ScriptCore
- **Graaljs**

#8:

TRY OTHER ENGINES

@pml0pes

Graal



https://graalvm.org

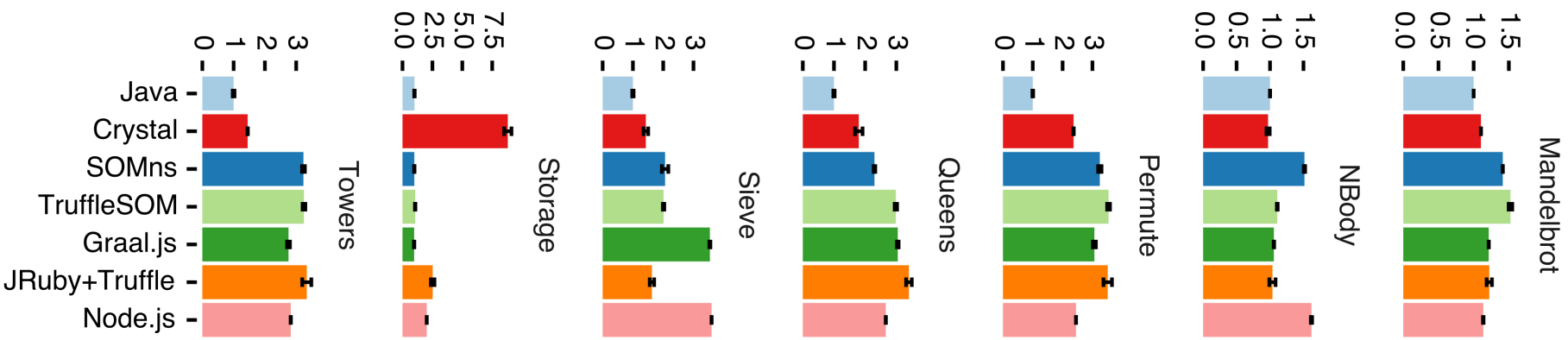
@pml0pes

GRAALJS

- *JavaScript with best possible performance*
- *Full latest ECMAScript specification*
- *Fast interoperability*
  - *Java, Scala, or Kotlin, …*
  - *or GraalVM Ruby, Python, or R …*

https://github.com/graalvm/graaljs

GRAAL VS GRAAL.JS VS NODE.JS



https://stefan-marr.de/papers/dls-marr-et-al-cross-language-compiler-benchmarking-are-we-fast-yet/

@pml0pes

HYPOTHESIS

@pml0pes

ES4X

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)
- ~~libUV~~ ⇒ Eclipse Vert.x (https://vertx.io)

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)
- ~~libUV~~ ⇒ Eclipse Vert.x (https://vertx.io)
- ~~V8 JIT~~ ⇒ GraalVM JVMCI (https://graalvm.org)

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)
- ~~libUV~~ ⇒ Eclipse Vert.x (https://vertx.io)
- ~~V8 JIT~~ ⇒ GraalVM JVMCI (https://graalvm.org)
- ~~node bindings`~~ ⇒ `.d.ts` of Java APIs

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)
- ~~libUV~~ ⇒ Eclipse Vert.x (https://vertx.io)
- ~~V8 JIT~~ ⇒ GraalVM JVMCI (https://graalvm.org)
- ~~node bindings`~~ ⇒ **`.d.ts`** of Java APIs
- **`commonjs and ESM`** loader

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)
- ~~libUV~~ ⇒ Eclipse Vert.x (https://vertx.io)
- ~~V8 JIT~~ ⇒ GraalVM JVMCI (https://graalvm.org)
- ~~node bindings`~~ ⇒ **`.d.ts`** of Java APIs
- **`commonjs and ESM`** loader
- **`npm`** compatibility (no native modules for now)

https://reactiverse.io/es4x

@pml0pes

ES4X

- ~~V8~~ ⇒ GraalJS (https://graalvm.org)
- ~~libUV~~ ⇒ Eclipse Vert.x (https://vertx.io)
- ~~V8 JIT~~ ⇒ GraalVM JVMCI (https://graalvm.org)
- ~~node bindings`~~ ⇒ **`.d.ts`** of Java APIs
- **`commonjs and ESM`** loader
- **`npm`** compatibility (no native modules for now)
- debug/profile using **chrome-devtools**

https://reactiverse.io/es4x

```
import { Router } from '@vertx/web';

const app = Router.router(vertx);

app.get("/plaintext").handler(ctx => {
  ctx.response()
    .putHeader("Content-Type", "text/plain")
    .end('Hello, World!');
});
```

@pml0pes

EXPECTATION:

@pml0pes

EXPECTATION:

- User code (JavaScript)

EXPECTATION:

- User code (JavaScript)
- + Runtime (ES4X)

EXPECTATION:

- User code (JavaScript)
- + Runtime (ES4X)
- + Interop JS ⟷ Java

EXPECTATION:

- User code (JavaScript)
- + Runtime (ES4X)
- + Interop JS ⟷ Java
- + JS Engine (GraalJS)

EXPECTATION:

- User code (JavaScript)
- + Runtime (ES4X)
- + Interop JS ⟷ Java
- + JS Engine (GraalJS)
- + IO library (vert.x)

EXPECTATION:

- User code (JavaScript)
- + Runtime (ES4X)
- + Interop JS ⟷ Java
- + JS Engine (GraalJS)
- + IO library (vert.x)
- + JDK

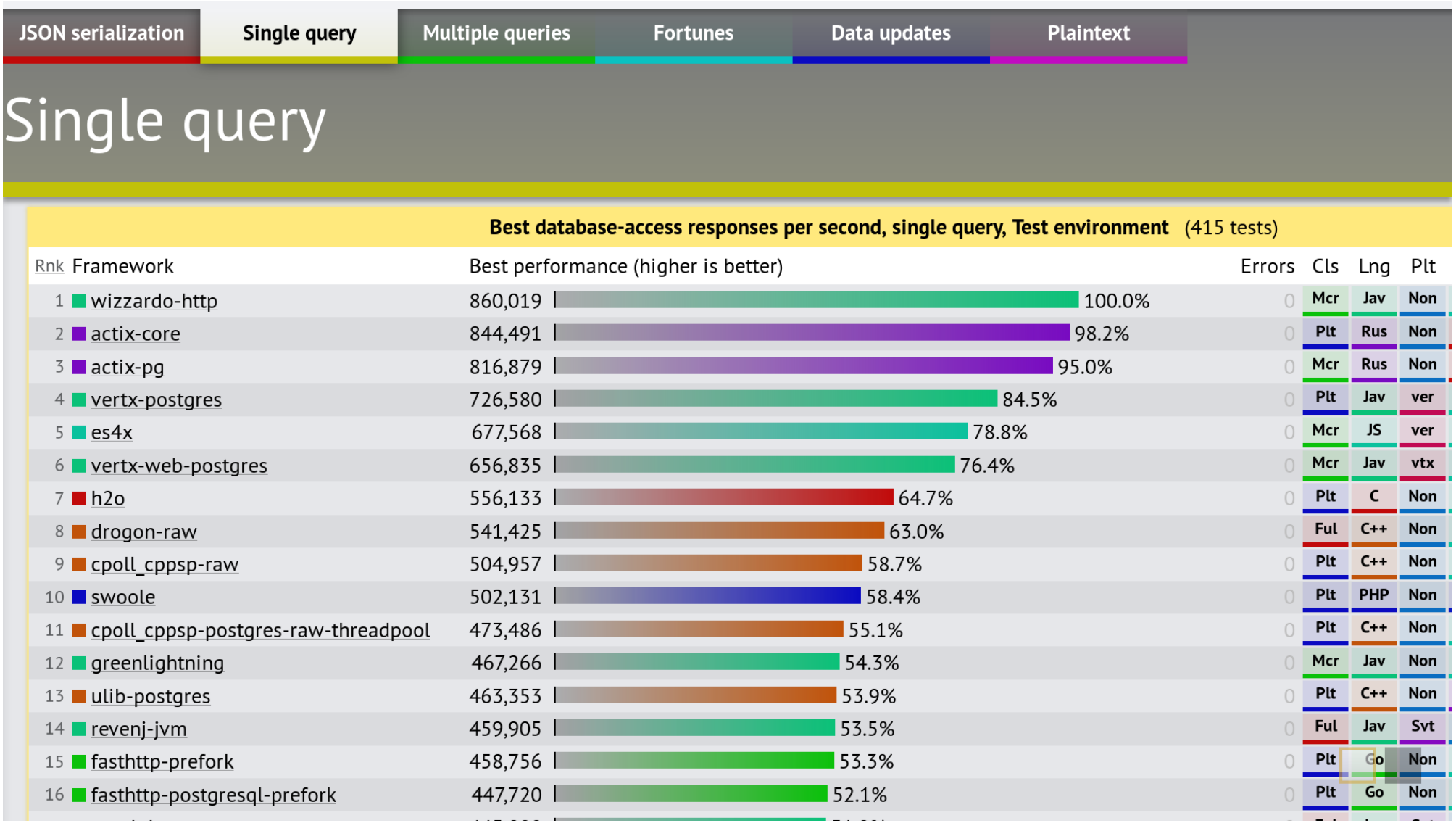
EXPECTATION:

- User code (JavaScript)
- **+** Runtime (ES4X)
- **+** Interop JS ⟷ Java
- **+** JS Engine (GraalJS)
- **+** IO library (vert.x)
- **+** JDK
- **= Full application optimization** 

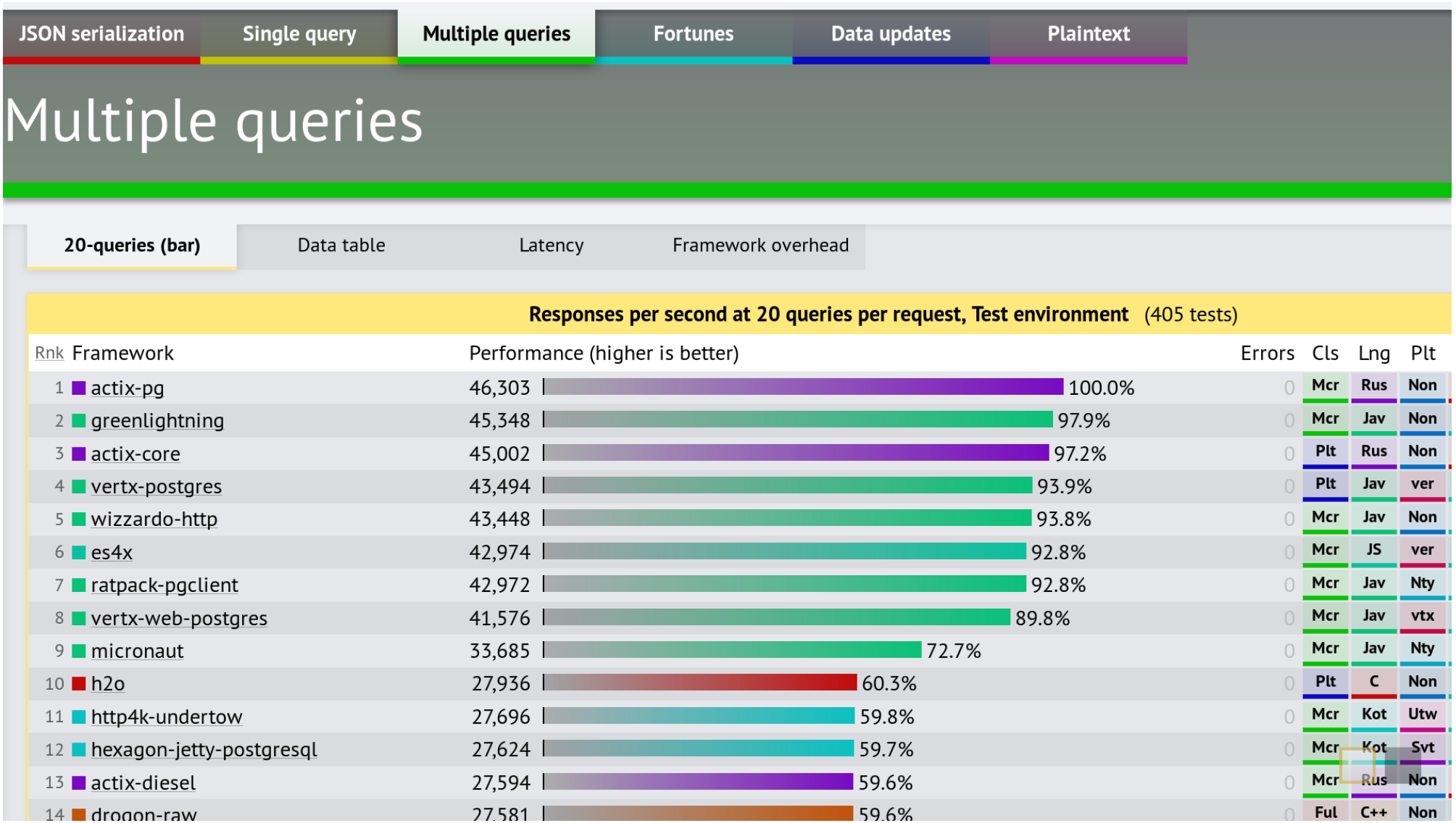
#9:

TEST EXPERIMENT

@pml0pes


JSON serialization
Single query
Multiple queries
Fortunes
Data updates
Plaintext
Single query
Best database-access responses per second, single query, Test environment (415 tests)
Rnk Framework Best performance (higher is better) Errors Cls Lng Plt
1 wizzardo-http 860,019 100.0% 0 Mcr Jav Non
2 actix-core 844,491 98.2% 0 Plt Rus Non
3 actix-pg 816,879 95.0% 0 Mcr Rus Non
4 vertx-postgres 726,580 84.5% 0 Plt Jav ver
5 es4x 677,568 78.8% 0 Mcr JS ver
6 vertx-web-postgres 656,835 76.4% 0 Mcr Jav vtx
7 h2o 556,133 64.7% 0 Plt C Non
8 drogon-raw 541,425 63.0% 0 Ful C++ Non
9 cpoll_cppsp-raw 504,957 58.7% 0 Plt C++ Non
10 swoole 502,131 58.4% 0 Plt PHP Non
11 cpoll_cppsp-postgres-raw-threadpool 473,486 55.1% 0 Plt C++ Non
12 greenlightning 467,266 54.3% 0 Mcr Jav Non
13 ulib-postgres 463,353 53.9% 0 Plt C++ Non
14 revenj-jvm 459,905 53.5% 0 Ful Jav Svt
15 fasthttp-prefork 458,756 53.3% 0 Plt Go Non
16 fasthttp-postgresql-prefork 447,720 52.1% 0 Plt Go Non

@pml0pes

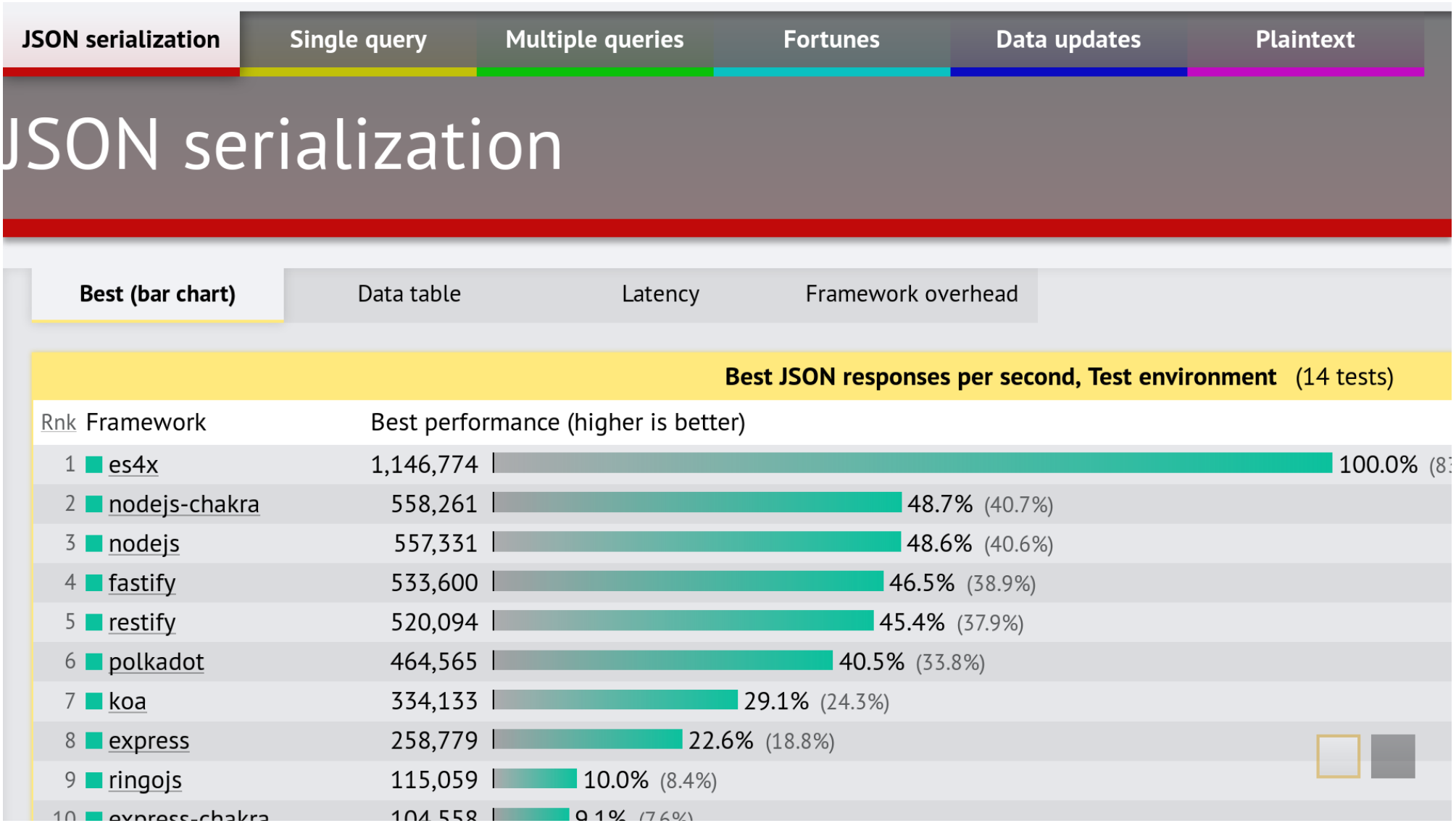

JSON serialization
Single query
Multiple queries
Fortunes
Data updates
Plaintext
Multiple queries
20-queries (bar)
Data table
Latency
Framework overhead
Responses per second at 20 queries per request, Test environment (405 tests)
Rnk Framework Performance (higher is better) Errors Cls Lng Plt
1 actix-pg 46,303 100.0% 0 Mcr Rus Non
2 greenlightning 45,348 97.9% 0 Mcr Jav Non
3 actix-core 45,002 97.2% 0 Plt Rus Non
4 vertx-postgres 43,494 93.9% 0 Plt Jav ver
5 wizzardo-http 43,448 93.8% 0 Mcr Jav Non
6 es4x 42,974 92.8% 0 Mcr JS ver
7 ratpack-pgclient 42,972 92.8% 0 Mcr Jav Nty
8 vertx-web-postgres 41,576 89.8% 0 Mcr Jav vtx
9 micronaut 33,685 72.7% 0 Mcr Jav Nty
10 h2o 27,936 60.3% 0 Plt C Non
11 http4k-undertow 27,696 59.8% 0 Mcr Kot Utw
12 hexagon-jetty-postgresql 27,624 59.7% 0 Mcr Kot Svt
13 actix-diesel 27,594 59.6% 0 Mcr Rus Non
14 drogon-raw 27,581 59.6% 0 Ful C++ Non

@pml0pes

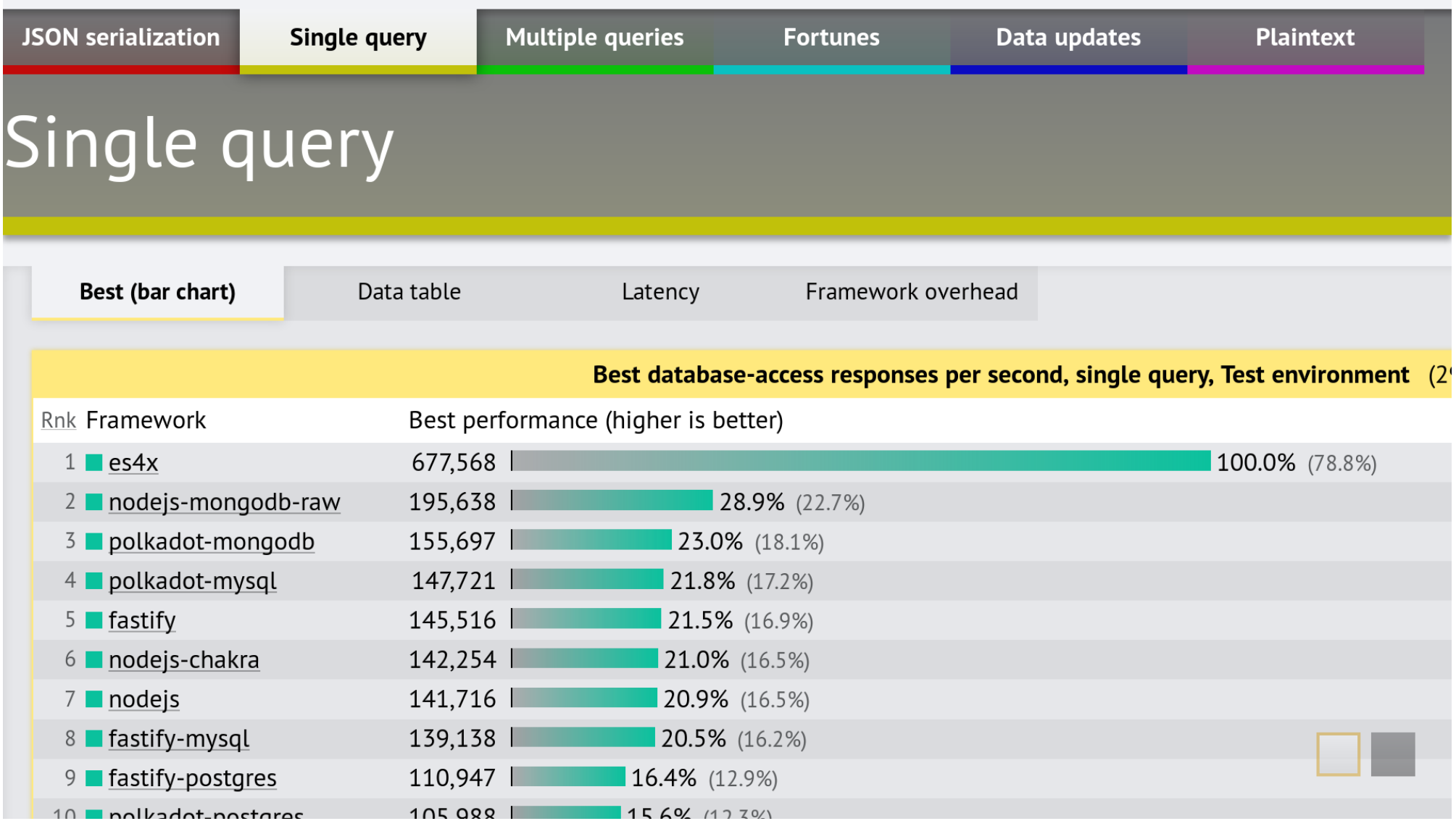
#10:

HOW DOES IT COMPARE?

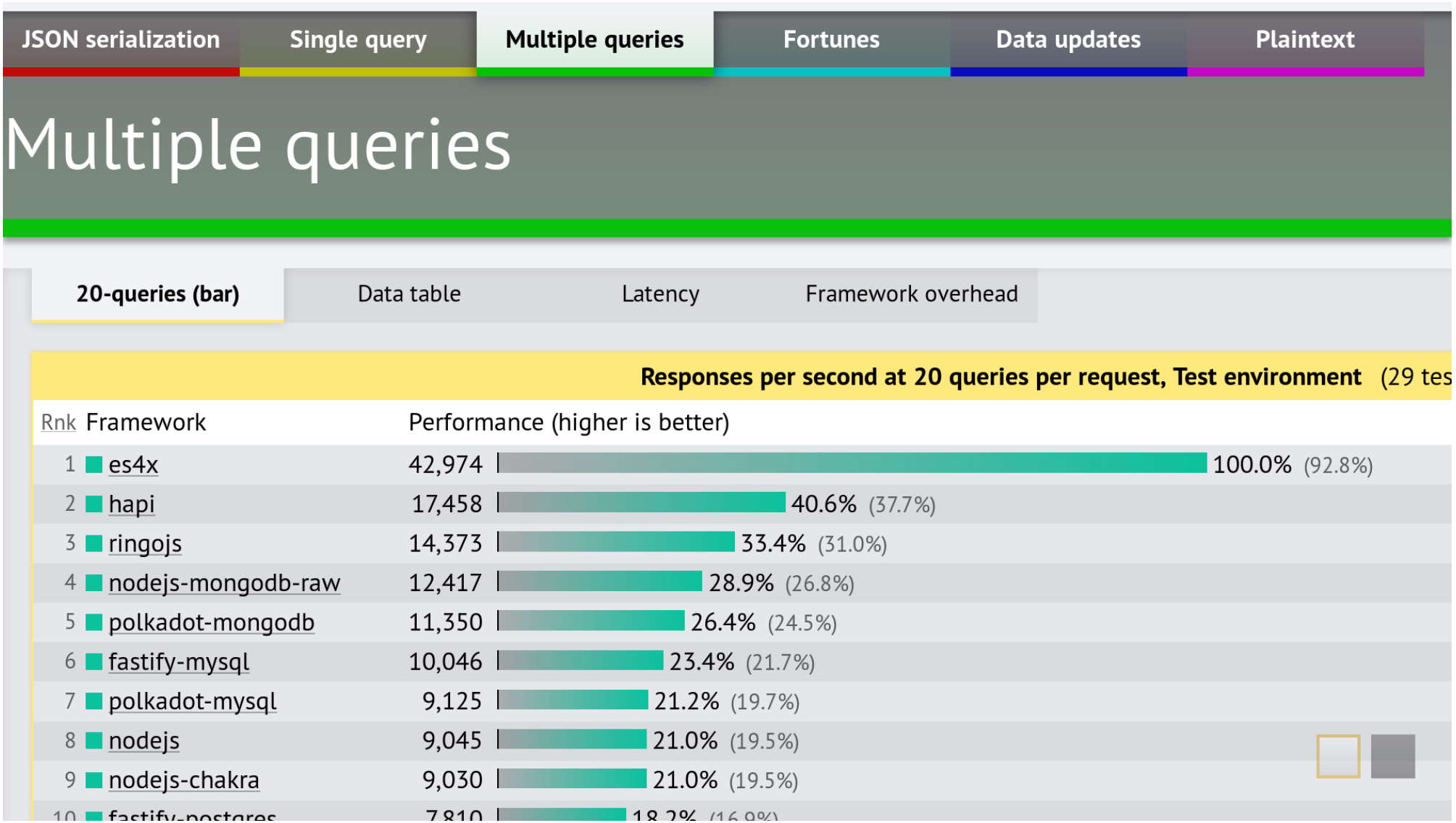
@pml0pes


JSON serialization
Single query
Multiple queries
Fortunes
Data updates
Plaintext
JSON serialization
Best (bar chart)
Data table
Latency
Framework overhead
Best JSON responses per second, Test environment (14 tests)
Rnk Framework
Best performance (higher is better)
1 es4x 1,146,774 100.0%
2 nodejs-chakra 558,261 48.7% (40.7%)
3 nodejs 557,331 48.6% (40.6%)
4 fastify 533,600 46.5% (38.9%)
5 restify 520,094 45.4% (37.9%)
6 polkadot 464,565 40.5% (33.8%)
7 koa 334,133 29.1% (24.3%)
8 express 258,779 22.6% (18.8%)
9 ringojs 115,059 10.0% (8.4%)

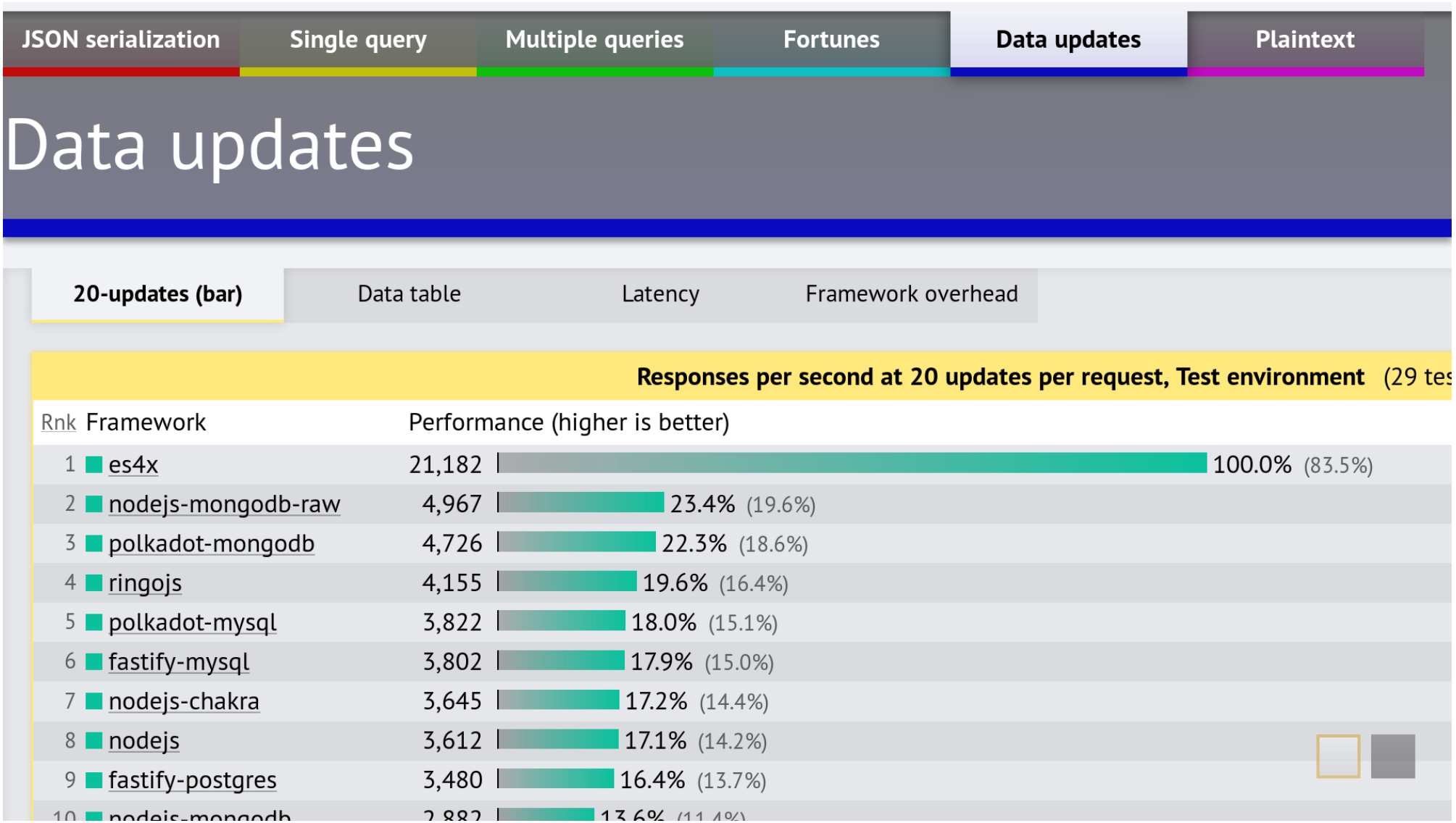
@pml0pes


JSON serialization
Single query
Multiple queries
Fortunes
Data updates
Plaintext
Single query
Best (bar chart)
Data table
Latency
Framework overhead
Best database-access responses per second, single query, Test environment
Rnk Framework
Best performance (higher is better)
1 es4x 677,568 100.0% (78.8%)
2 nodejs-mongodb-raw 195,638 28.9% (22.7%)
3 polkadot-mongodb 155,697 23.0% (18.1%)
4 polkadot-mysql 147,721 21.8% (17.2%)
5 fastify 145,516 21.5% (16.9%)
6 nodejs-chakra 142,254 21.0% (16.5%)
7 nodejs 141,716 20.9% (16.5%)
8 fastify-mysql 139,138 20.5% (16.2%)
9 fastify-postgres 110,947 16.4% (12.9%)

@pml0pes


JSON serialization
Single query
Multiple queries
Fortunes
Data updates
Plaintext
Multiple queries
20-queries (bar)
Data table
Latency
Framework overhead
Responses per second at 20 queries per request, Test environment (29 tes
Rnk Framework
Performance (higher is better)
1 es4x 42,974 100.0% (92.8%)
2 hapi 17,458 40.6% (37.7%)
3 ringojs 14,373 33.4% (31.0%)
4 nodejs-mongodb-raw 12,417 28.9% (26.8%)
5 polkadot-mongodb 11,350 26.4% (24.5%)
6 fastify-mysql 10,046 23.4% (21.7%)
7 polkadot-mysql 9,125 21.2% (19.7%)
8 nodejs 9,045 21.0% (19.5%)
9 nodejs-chakra 9,030 21.0% (19.5%)

@pml0pes


JSON serialization
Single query
Multiple queries
Fortunes
Data updates
Plaintext
Data updates
20-updates (bar)
Data table
Latency
Framework overhead
Responses per second at 20 updates per request, Test environment (29 tes
Rnk Framework
Performance (higher is better)
1 es4x 21,182 100.0% (83.5%)
2 nodejs-mongodb-raw 4,967 23.4% (19.6%)
3 polkadot-mongodb 4,726 22.3% (18.6%)
4 ringojs 4,155 19.6% (16.4%)
5 polkadot-mysql 3,822 18.0% (15.1%)
6 fastify-mysql 3,802 17.9% (15.0%)
7 nodejs-chakra 3,645 17.2% (14.4%)
8 nodejs 3,612 17.1% (14.2%)
9 fastify-postgres 3,480 16.4% (13.7%)

@pml0pes

REQ/SEC

Node*	ES4X**	Test
558,261	**1,146,774**	JSON
195,638	**677,568**	Single Query
17,458	**42,974**	Multiple Query
4,967	**21,182**	Data Updates
881,321	**2,268,524**	Plaintext

Higher is better

CONCLUSION

- *There's nothing wrong with **JavaScript**.*

CONCLUSION

- *There's nothing wrong with **JavaScript**.*
- Yes **JavaScript** is **fast**

CONCLUSION

- *There's nothing wrong with **JavaScript***.
- Yes **JavaScript** is **fast**
- You don't need to switch to Go/Rust/etc...

CONCLUSION

- *There's nothing wrong with **JavaScript***.
- Yes **JavaScript** is **fast**
- You don't need to switch to Go/Rust/etc...
- *Dare to **Experiment***

THANK YOU!

- @pml0pes
- pmlopes
- https://reactiverse.io/es4x
- https://vertx.io
- https://graalvm.org