



JavaScript the next
big... **BYTECODE**

The background image is a black and white photograph of a traditional stone wall, possibly made of dry-stacked stones, with a thick, rounded top. The wall is set against a dark, textured background. The title text is overlaid on a white rectangular area in the center-left of the slide.

About://me



**Mathieu Breton : /[mathieu] bRətɔ̃/
(n.m.)**

- Consultant Xebia, membre actif de la communauté web à Paris.
- Chevalier de l'ordre du mérite des Single Page Applications.
- Artisan web, instructeur Neo4J, docteur ès Grails et contributeur Ember.JS.



De quoi allons nous parler ?

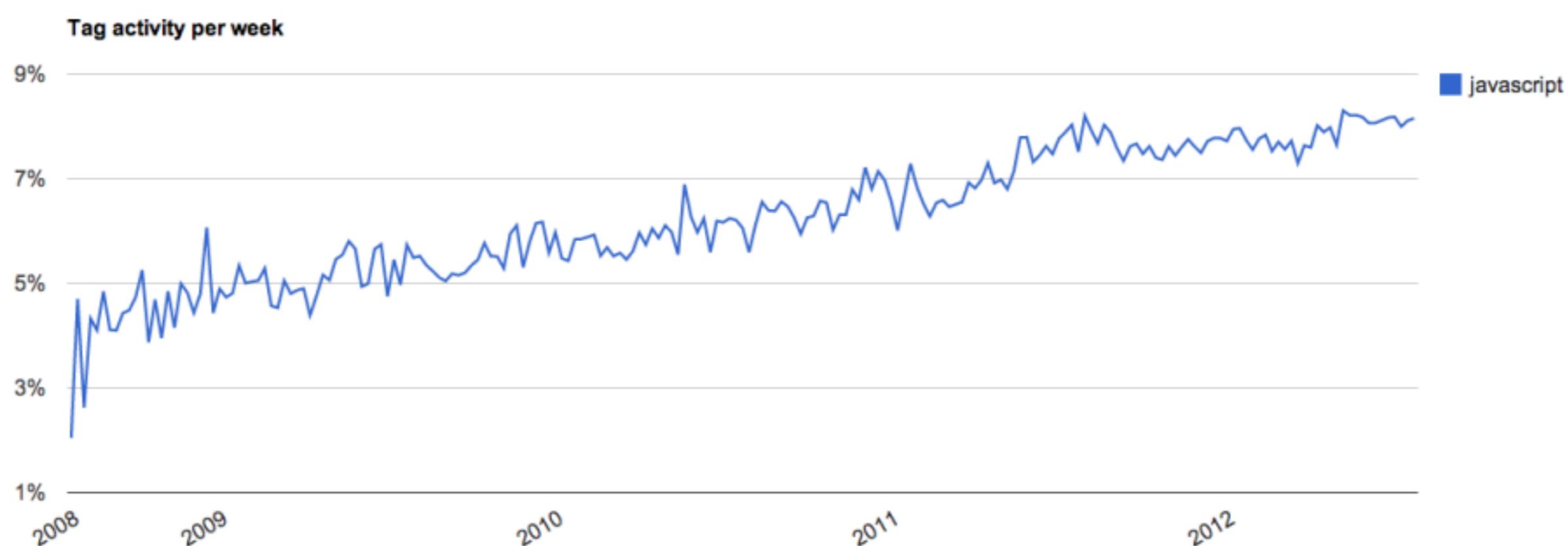
- DE JAVASCRIPT
- DE SON APOGEE
- DE LA VIE DE TOUS LES JOURS AVEC LUI
- ET DE SON AVENIR



L'apogée

L'apogée

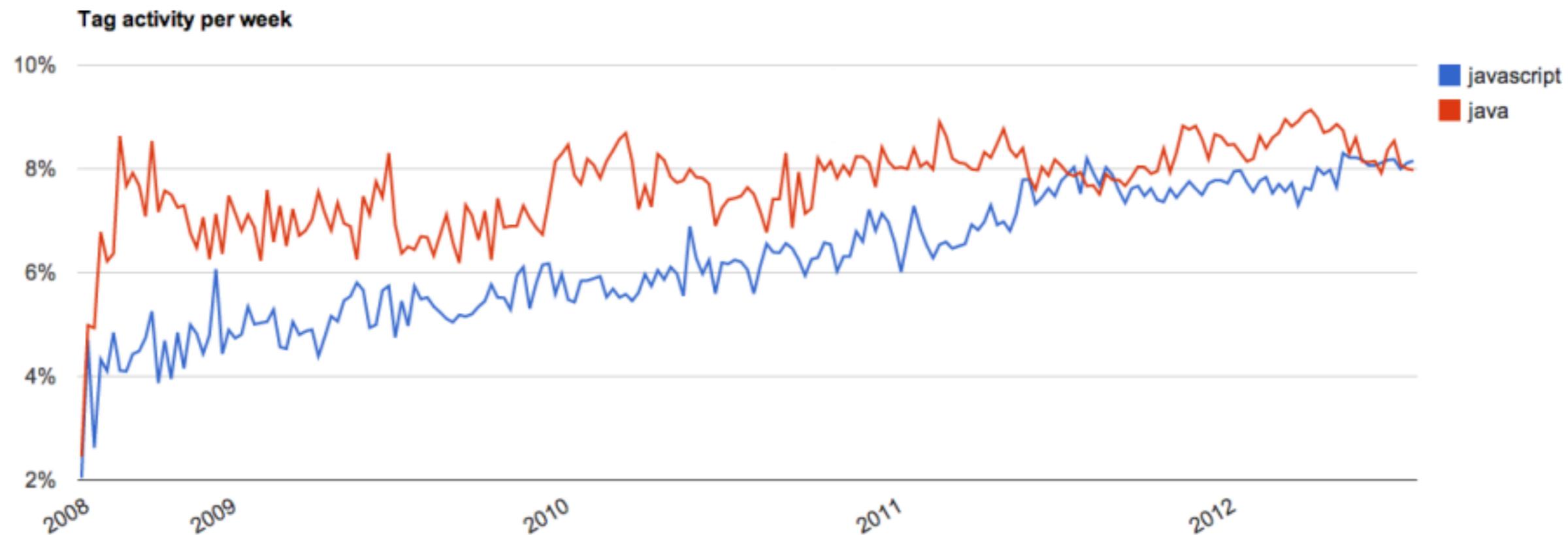
JavaScript sur Stackoverflow



TM
DEVOXX France

L'apogée

Java/JavaScript sur Stackoverflow



L'apogée



BACKBONE.JS

JavaScript est partout



Bootstrap



ANGULARJS
by Google



DEVOXX France



L'apogée

POURQUOI ?

L'apogée

Serveur Web

```
var express = require('express');
var app = express();
app.get('/', function (req, res) {
  res.send('Hello, DevoxxFR 2014');
});
app.listen(3000);
```

Programmation fonctionnelle

```
_chain([1, 2, 3, 200])
  .filter(function (num) { return num % 2 === 0; })
  .tap(alert) // alerted [2, 200]
  .map(function (num) { return num * num })
  .value(); // [4,40000]
```



L'apogée

RAPIDE*

SIMPLE

PRODUCTIF

DEVOXXTM France



L'apogée

Le JavaScript c'est bien
tout le temps !

DEVOXX France

TM



DEVOXX



Le JS SHOW

Javascript Console
js>



JavaScript a un Typage Dynamique Implicite Faible

Le typage JS génère un outillage de mauvaise qualité

Un navigateur compatible avec Adobe Flash Player ou HTML5 est requis pour la lecture de vidéo.

[Télécharger la dernière version de Flash Player](#)

[En savoir plus à propos du passage à un navigateur HTML5](#)



Le typage JS rend moins pertinents les tests unitaires

```
1 // Source
2 function A() {}
3 A.prototype.plus = function (arg1, arg2) {
4     return arg1 + arg2;
5 };
6
7 function B(a) {this.a = a;}
8 B.prototype.plusCaller = function (arg1, arg
9     return this.a.plus(arg1, arg2);
10};
11
12 // Test
13 test("Unit test", function () {
14     var dirtyMockA = {
15         plus: function (arg1, arg2) {
16             if (arg1 == 1 && arg2 == 1) {
17                 return 2;
18             }
19         }
20     };
21
22     var b = new B(dirtyMockA);
23     equal(b.plusCaller(1, 1), 2);
24});
```

Qunit test

Tests completed in 236 milliseconds.
1 assertions of 1 passed, 0 failed.

1. Unit test (0, 1, 1) Rerun

1 ms

Un JDK vous dites, humm ...

Un SDK JS ressemble à :



Lo-Dash
underscore.js





L'avenir

L'avenir

Le langage web de demain sera

- toujours aussi productif
- moins WTF !
- typé plus fortement
 - plus facile à maintenir
 - doté d'outils plus puissant



L'avenir

BYTECODE de quoi ?



L'avenir

L'avenir

Plus de 170 langages/transpileurs
génèrent du JavaScript



L'avenir

Les leaders

L'avenir

TypeScript



L'avenir

CoffeeScript



L'avenir

Dart



L'avenir

ASM. JS





TypeScript

TypeScript

Créé en février 2012 par Microsoft
Passé récemment en V1.0
Sous license Apache 2 License



TypeScript



Intégration dans la suite Visual Studio

A screenshot of the Visual Studio code editor showing a context menu for a piece of TypeScript code. The menu includes options like Refactor, Go To Definition, Go To Declaration, Find All References, Breakpoint, Run To Cursor, Run Flagged Threads To Cursor, Cut, Copy, Paste, and Outlining. The "Rename..." option is highlighted. The code itself is a snippet of Node.js/TypeScript code related to an application configuration.

```
///<reference path='..\\node\\node.d.ts' />

import http = module("http")
import url = module("url")
import routes = module("./routes/index")
import db = module("./db")
import express = module("express")

var app = express();

// Configuration
app.configure(function () {
    app.set('port', process.env.PORT || 3000);
    app.set('env', 'development');
    app.use(express.static(__dirname + '/public'));
    app.use(express.json());
    app.use(express.urlencoded({ extended: true }));
});

app.configure(function () {
    app.use(express.errorHandler());
});
```

howStack: true)));

TypeScript

Ressemble à ce qu'il y aura en **ECMA6**

TypeScript

Typage explicite **statique** faible, optionel



TypeScript

Modularisation du code

TypeScript

```
1 module Sayings {  
2     export class Greeter {  
3         greeting:string;  
4  
5         constructor(message:string) {  
6             this.greeting = message;  
7         }  
8  
9         greet() {  
10            return "Hello, " + this.greeting  
11        }  
12    }  
13 }  
14 var greeter = new Sayings.Greeter("world");  
15 var button = document.createElement('button'  
16 button.textContent = "Say Hello";  
17 button.onclick = function () {  
18     alert(greeter.greet());  
19 };  
20  
21 document.body.appendChild(button);
```

JavaScript

```
1 var Sayings;  
2 (function (Sayings) {  
3     var Greeter = (function () {  
4         function Greeter(message) {  
5             this.greeting = message;  
6         }  
7         Greeter.prototype.greet = function () {  
8             return "Hello, " + this.greeting;  
9         };  
10        return Greeter;  
11    })();  
12    Sayings.Greeter = Greeter;  
13 })(Sayings || (Sayings = {}));  
14  
15 var greeter = new Sayings.Greeter("world");  
16 var button = document.createElement('button'  
17 button.textContent = "Say Hello";  
18 button.onclick = function () {  
19     alert(greeter.greet());  
20 };  
21 document.body.appendChild(button);  
22
```



JavaScript - Orienté Objet - Les génériques - Les modules

TypeScript

Nécessite des interfaces d'interopérabilité
github.com/borisyankov/DefinitelyTyped

```
/*
 * Interface for the AJAX setting that will configure the AJAX request
 */
interface JQueryAjaxSettings {
    accepts?: any;
    async?: boolean;
    beforeSend? (jqXHR: JQueryXHR, settings: JQueryAjaxSettings): any;
    cache?: boolean;
    complete? (jqXHR: JQueryXHR, textStatus:string): any;
    contents?: { [key: string]: any; };
    contentType?: any;
    context?: any;
    converters?: { [key: string]: any; };
    crossDomain?: boolean;
    data?: any;
    dataFilter? (data:any, ty:any): any;
}
```



CoffeeScript

CoffeeScript

Créé en **Décembre 2009** par Jeremy Ashkenas

It's just JavaScript



CoffeeScript

Peut être utilisé dès à présent avec le *transpileur*

CoffeeScript

Une *syntaxe* très épurée

`(){}; , return*`

CoffeeScript

Modèle *orienté objet*

```
class Animal
  constructor: (@name)->

  class Pet extends Animal
    constructor: (@owner, name) ->
      super name

    cat = new Pet "Mathieu", "Garfield"

    if cat instanceof Pet
      console.log "#{cat.name} is a Pet"
```

CoffeeScript



Du sucre *syntaxique*

```
# for comprehension
foods = ['broccoli', 'spinach', 'chocolate']
eat food for food in foods when food isnt 'chocolate'

# for with range
countdown = (num for num in [10..1])

# Chained opeator
cholesterol = 127

healthy = 200 > cholesterol > 60
```

CoffeeScript



Encore du sucre syntaxique



```
# New operators
winner = yes if pick in [47, 92, 13]

launch() if ignition is on

volume = 10 if band isnt SpinalTap

letTheWildRumpusBegin() unless answer is no

solipsism = true if mind? and not world?
```

CoffeeScript



Beaucoup de sucre syntaxique

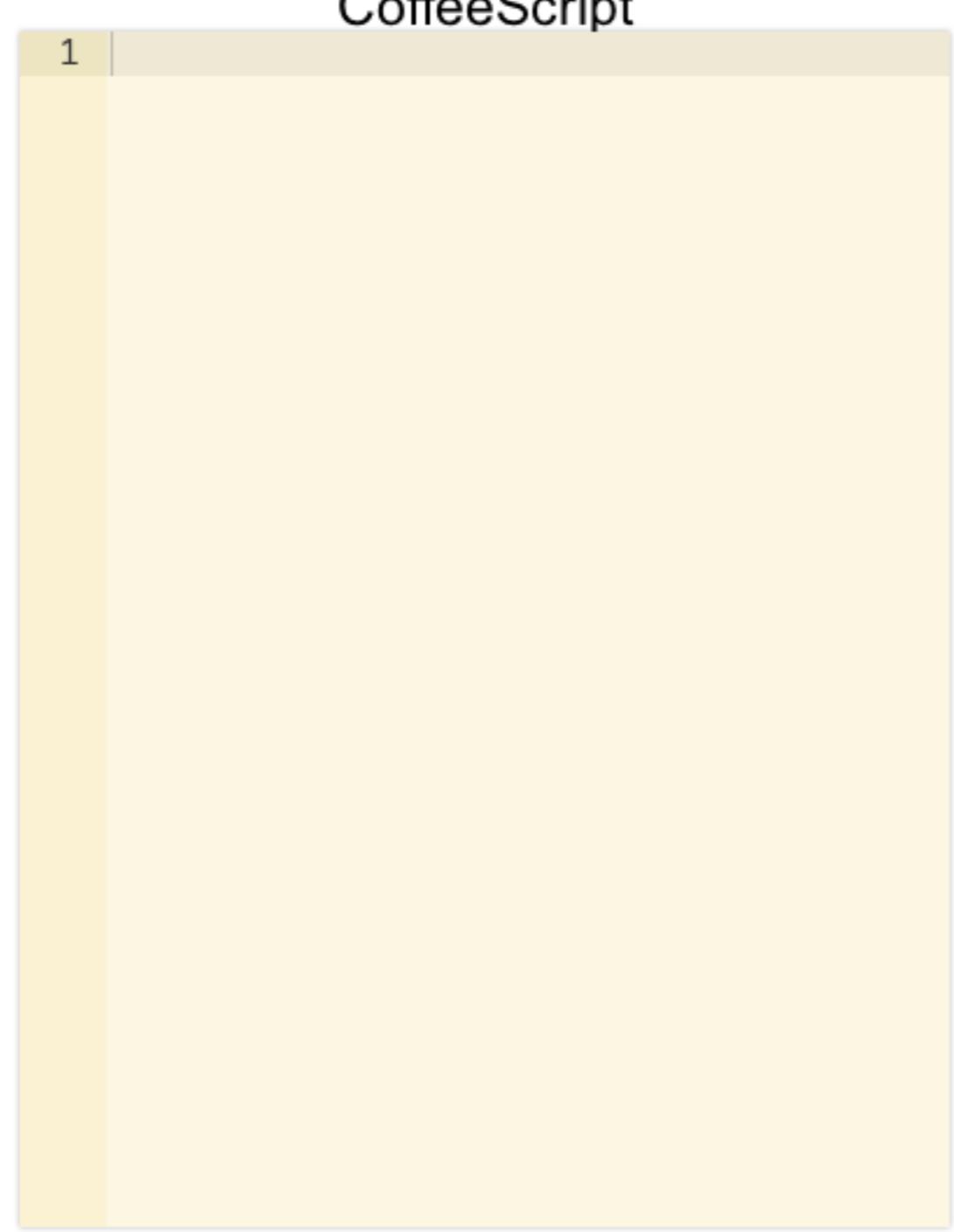


```
#destructuring assignment
weatherReport = (location) ->
  # Make an Ajax request to fetch the weather...
  [location, 72, "Mostly Sunny"]

[city, temp, forecast] = weatherReport "Berkeley, CA"

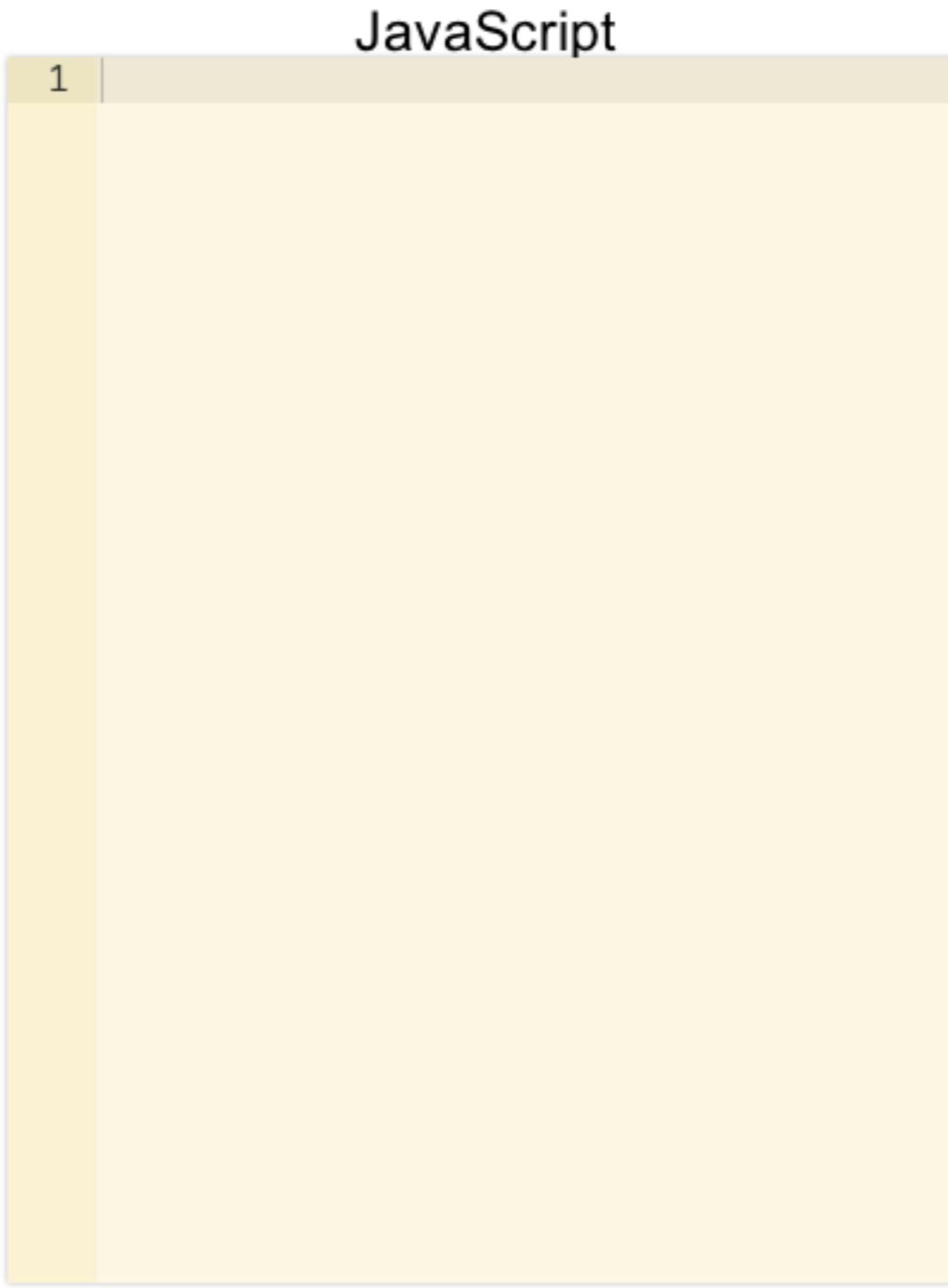
#the strings become magic
sentence = "#{ 22 / 7 } is a decent approximation of π"
html = """
<strong>
  cup of coffeescript
</strong>
"""
```

CoffeeScript



CoffeeScript

Ca fait des *carries*



JavaScript





Propulsé par Google en **2011**
Récemment sortie en **version 1.3**
Sous license **BSD**
standardisation ECMA par le comité **TC52**



Conçu pour :

- bâtir de larges applications web
- créer des jeux vidéos
- être plus rapide que JavaScript

Peut être exécuté dans une **machine virtuelle**
Ou sur **Dartium**, une version de Chromium



Dart

Est donc nativement côté **serveur** & côté **client**



Dart

Ou peut être compilé **en JavaScript**
Transpileur **Dart2Js**

DEVOXX France



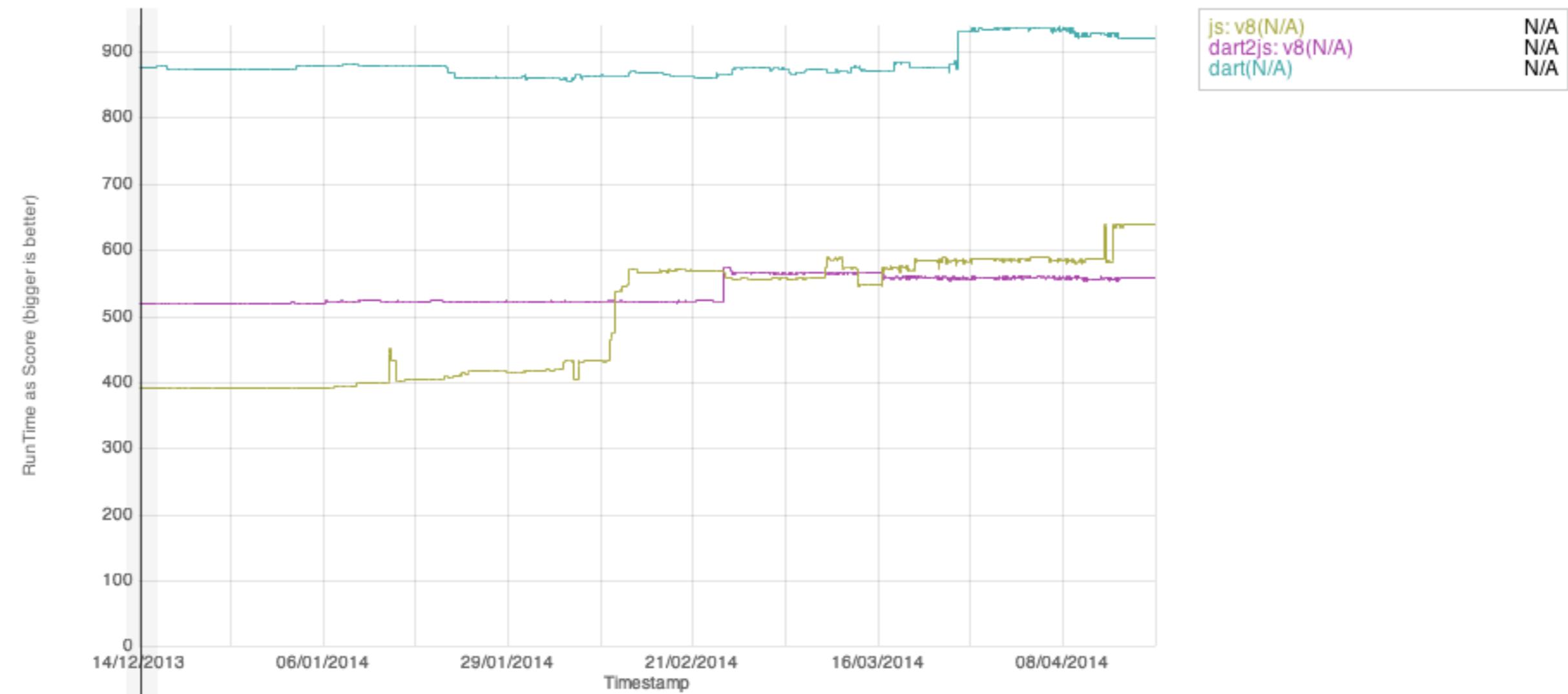


Nativement, Dart est plus rapide **rapide** que JavaScript
Et...

Le code JS généré peut être plus rapide que celui **écrit à la main**

Dart

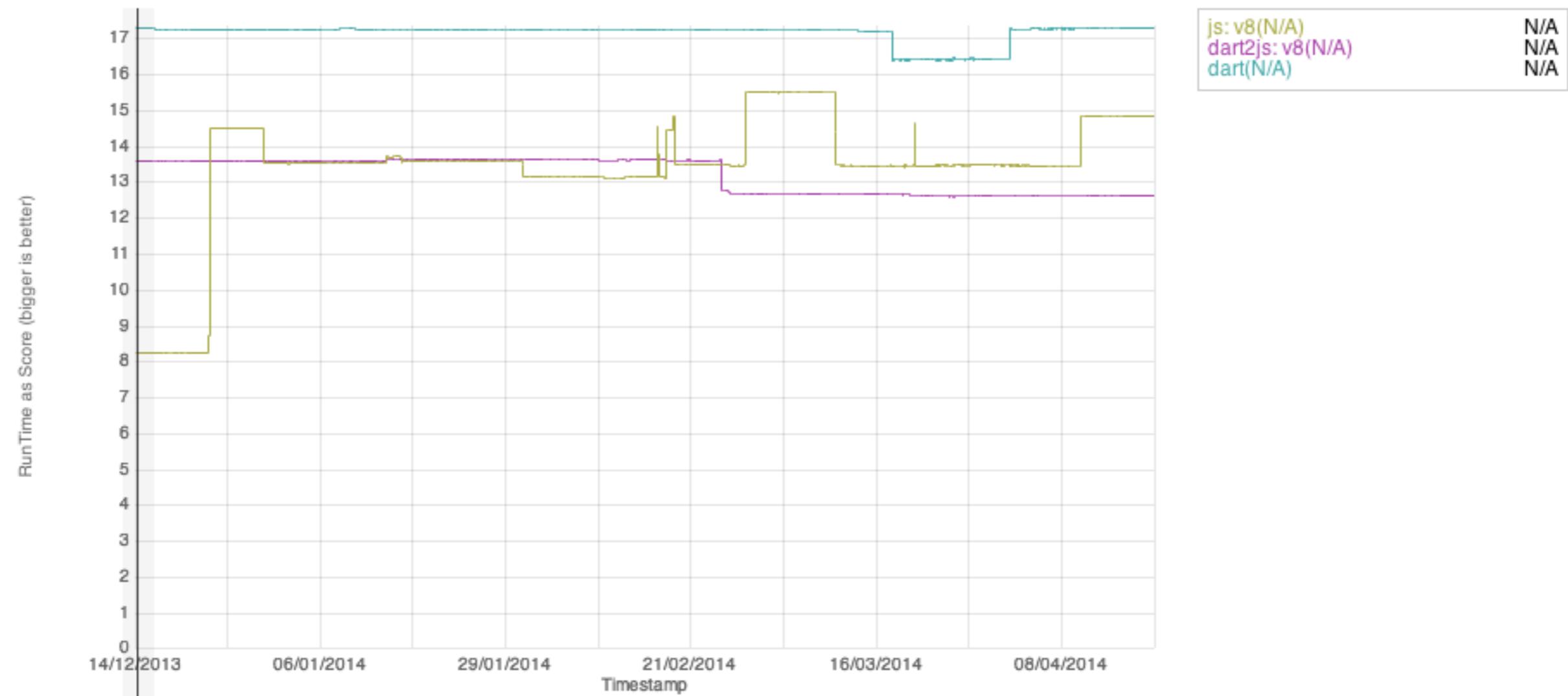
Performance - DeltaBlue



DEVOXX France

Dart

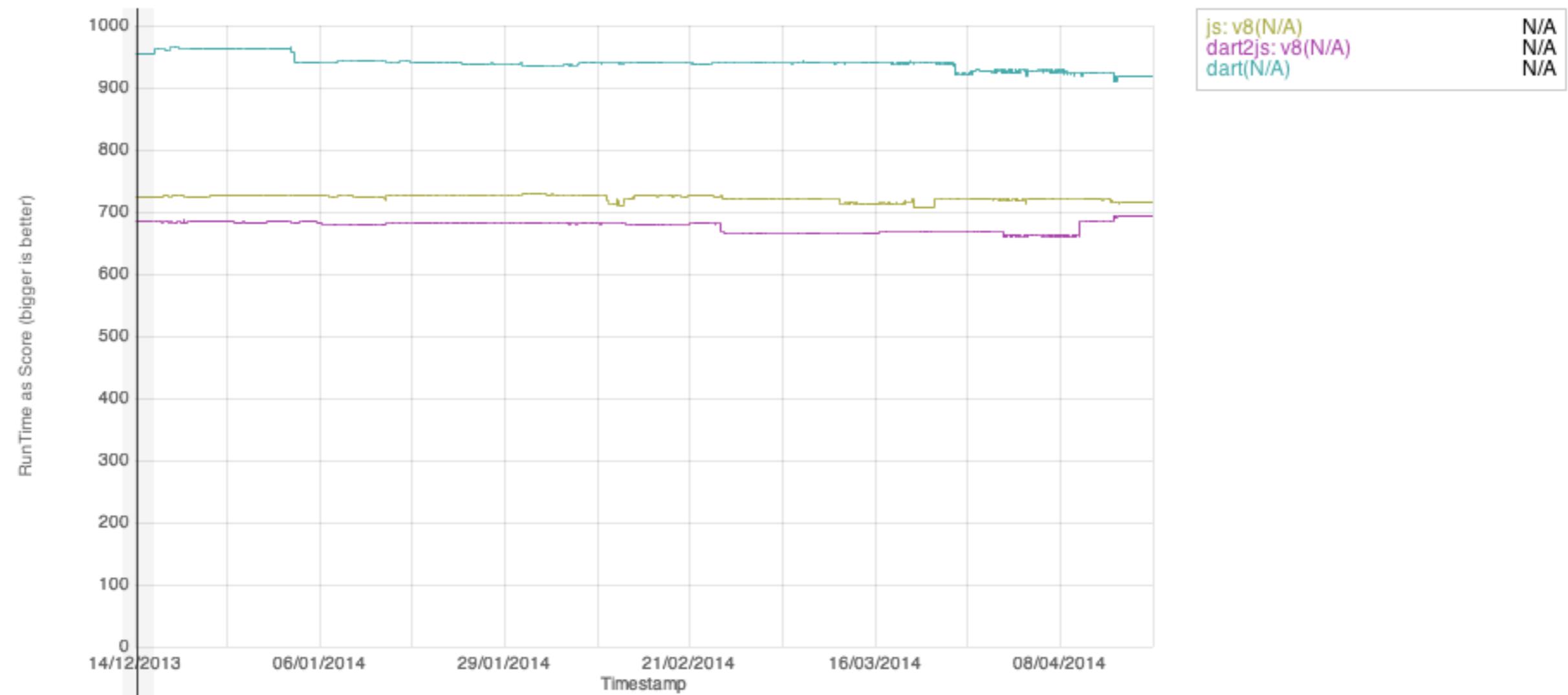
Performance - FluidMotion



DEVOXX France

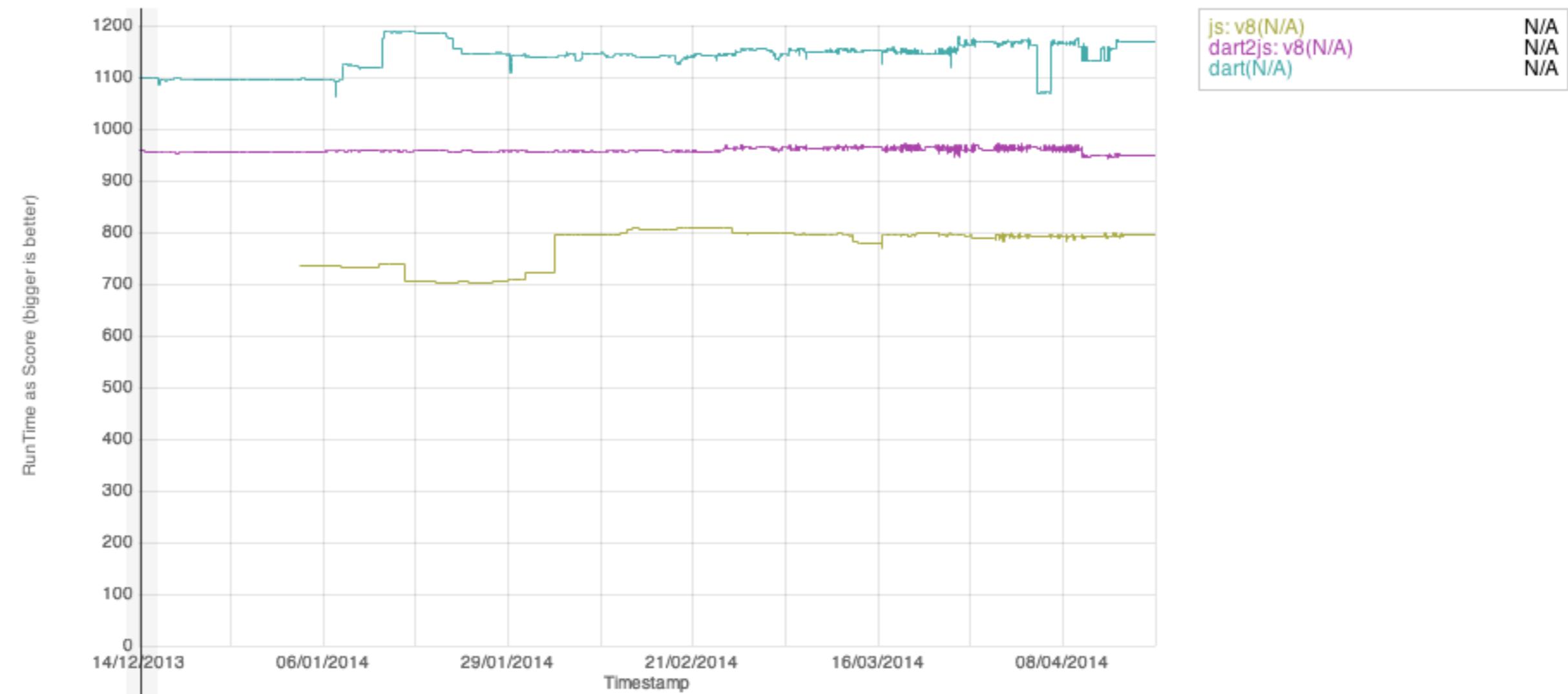
Dart

Performance - Richards



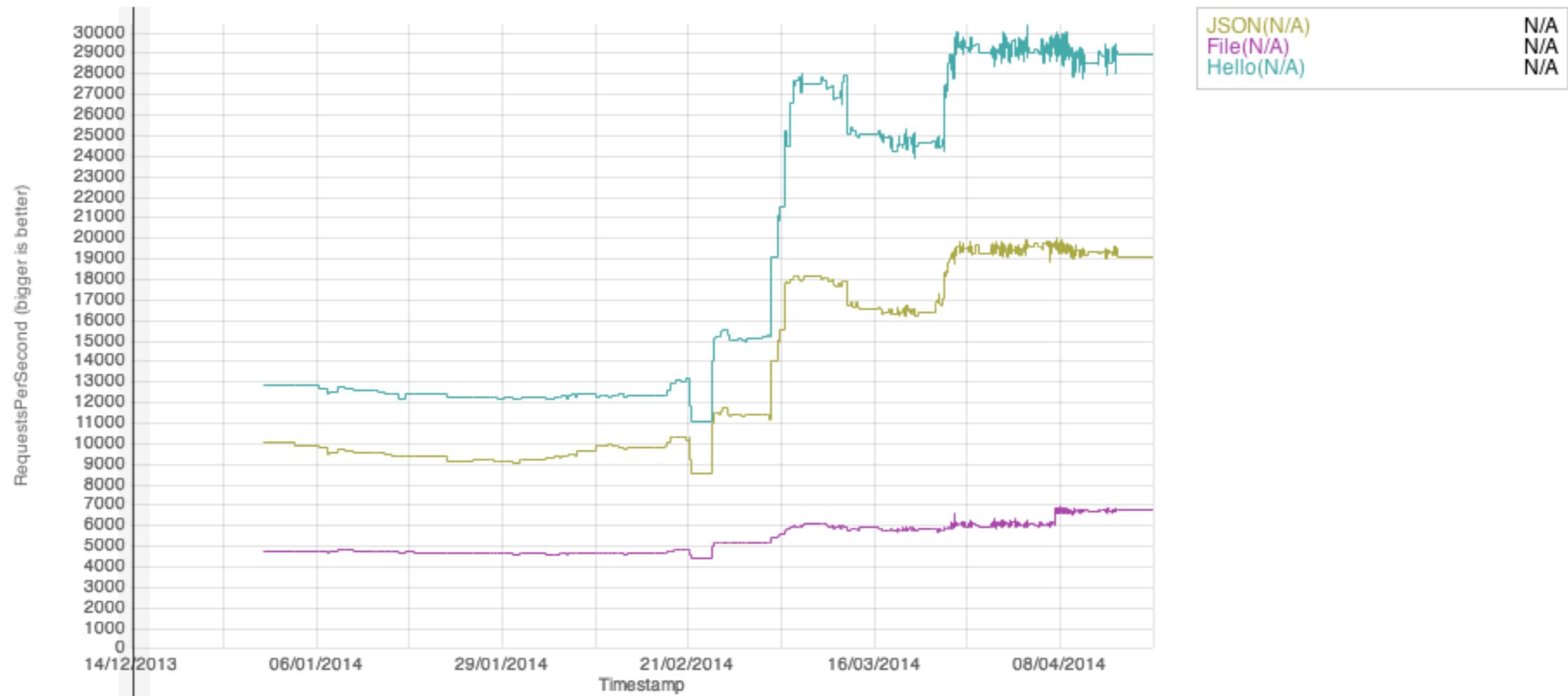
Dart

Performance - Tracer



Dart

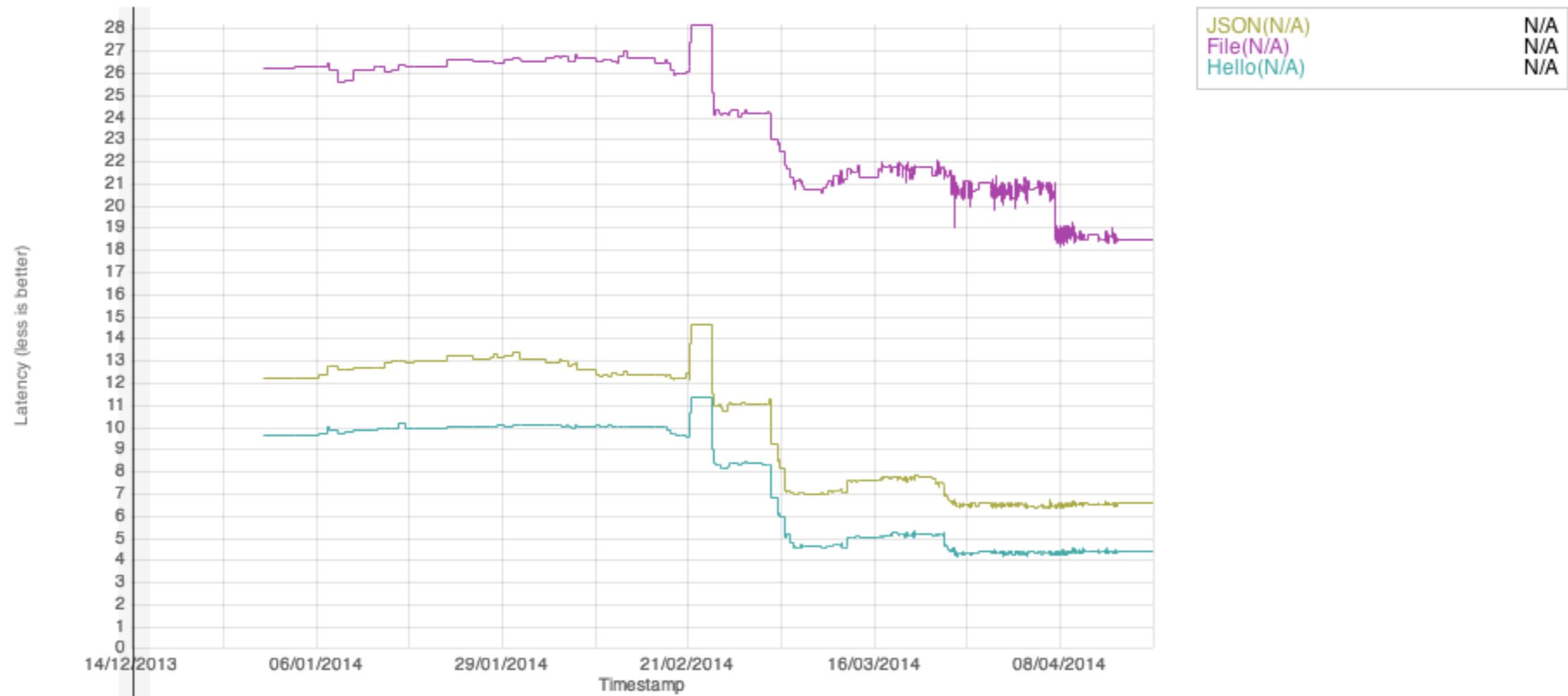
Performance - RequestPerSecond



DEVOXX France

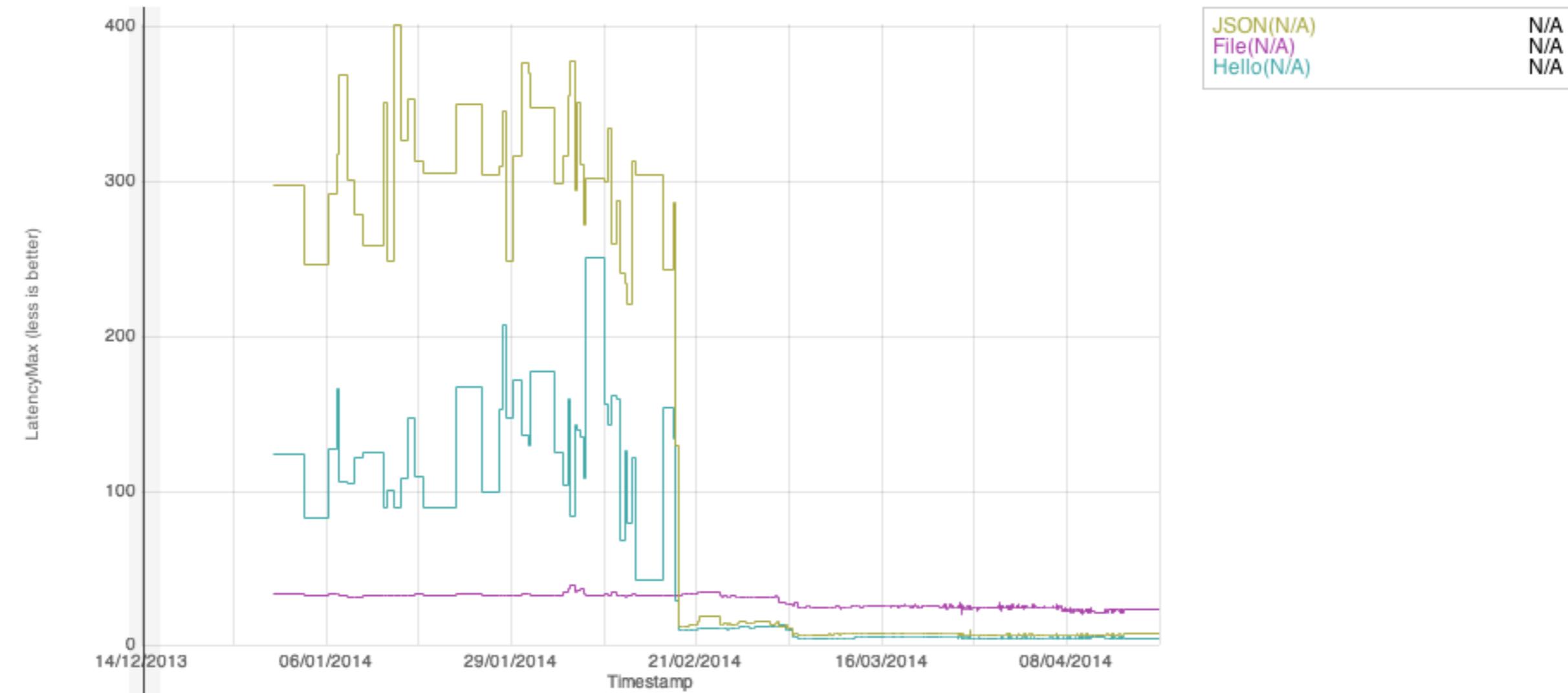
Dart

Performance - Avg Latency



Dart

Performance - Max Latency



DEVOXX France™



Typage "fort" explicite statique, mais optionel

```
var name = "Mathieu";
// ou
String name = "Mathieu";

void main() {
    var foo = "Salut numéro ";
    var bar = 5;
    print(foo + bar); // <= Illegal argument(s): 5
}
```

Modèle orienté objet

```
abstract class Animal {  
    var name = "";  
  
    Animal();  
    Animal.fromName(this.name);  
}  
  
class Pet extends Animal {  
    var owner = "";  
}
```



Asynchronisme basé sur des **futurs**

```
import 'dart:io';
import 'dart:async';

void printDailyNewsDigest() {
    File file = new File("dailyNewsDigest.txt");
    file.readAsString().then((content) => print(content)).catchError((erro:
}
```

Bien intégré à polymer

```
import 'package:polymer/polymer.dart';
import 'dart:html';

@CustomTag('click-counter')
class ClickCounterElement extends PolymerElement {
  @observable
  int count = 0;

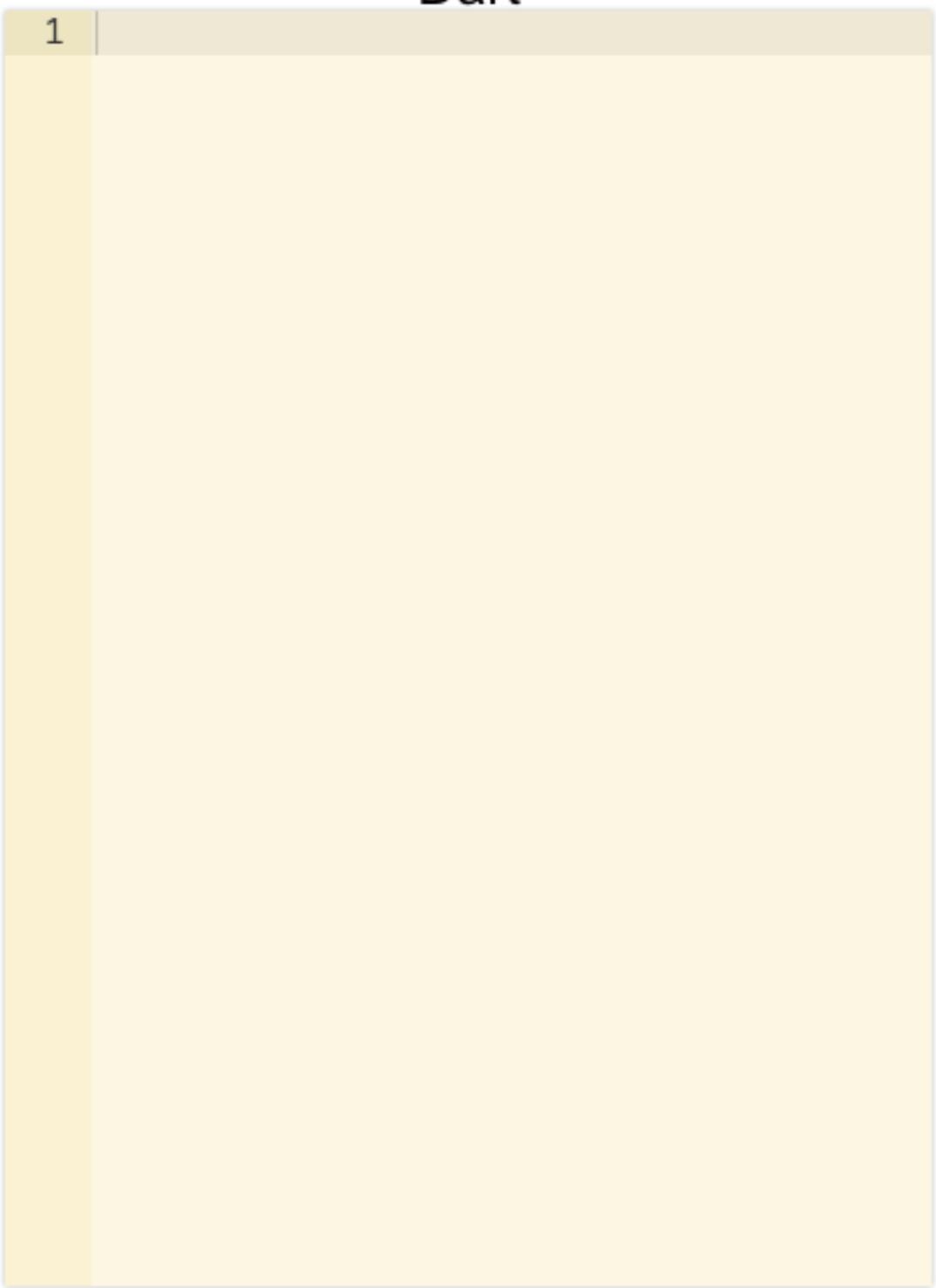
  void increment(Event e, var detail, Node target) {
    count += 1;
  }
}
```



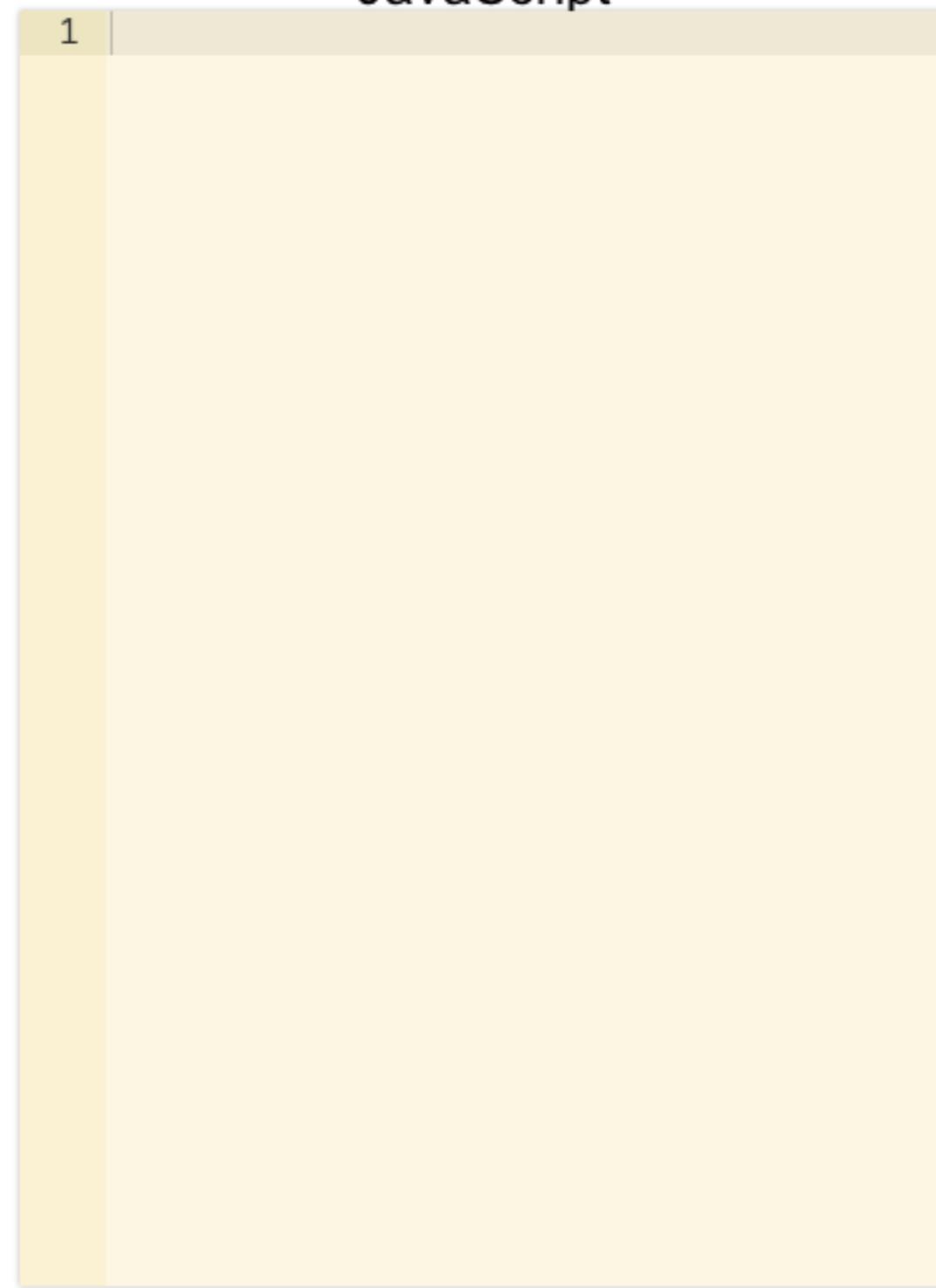
Fourni avec un écosystème riche

- pub.dartlang.org
- Dart Editor
- Une vraie API (Dart:HTML, Dart:IO, Dart:collection)

Dart



Dart



JavaScript





ASM.js

ASM.js

Ca vient de chez Mozilla
Sous license MIT
N'est pas pas un langage



ASM.js

Est un SOUS-ENSEMBLE de JavaScript très restrictif
"USE ASM"



ASM.js

Fait pour être le plus rapide possible, avec DES
PERFORMANCES PROCHE DU CODE NATIF

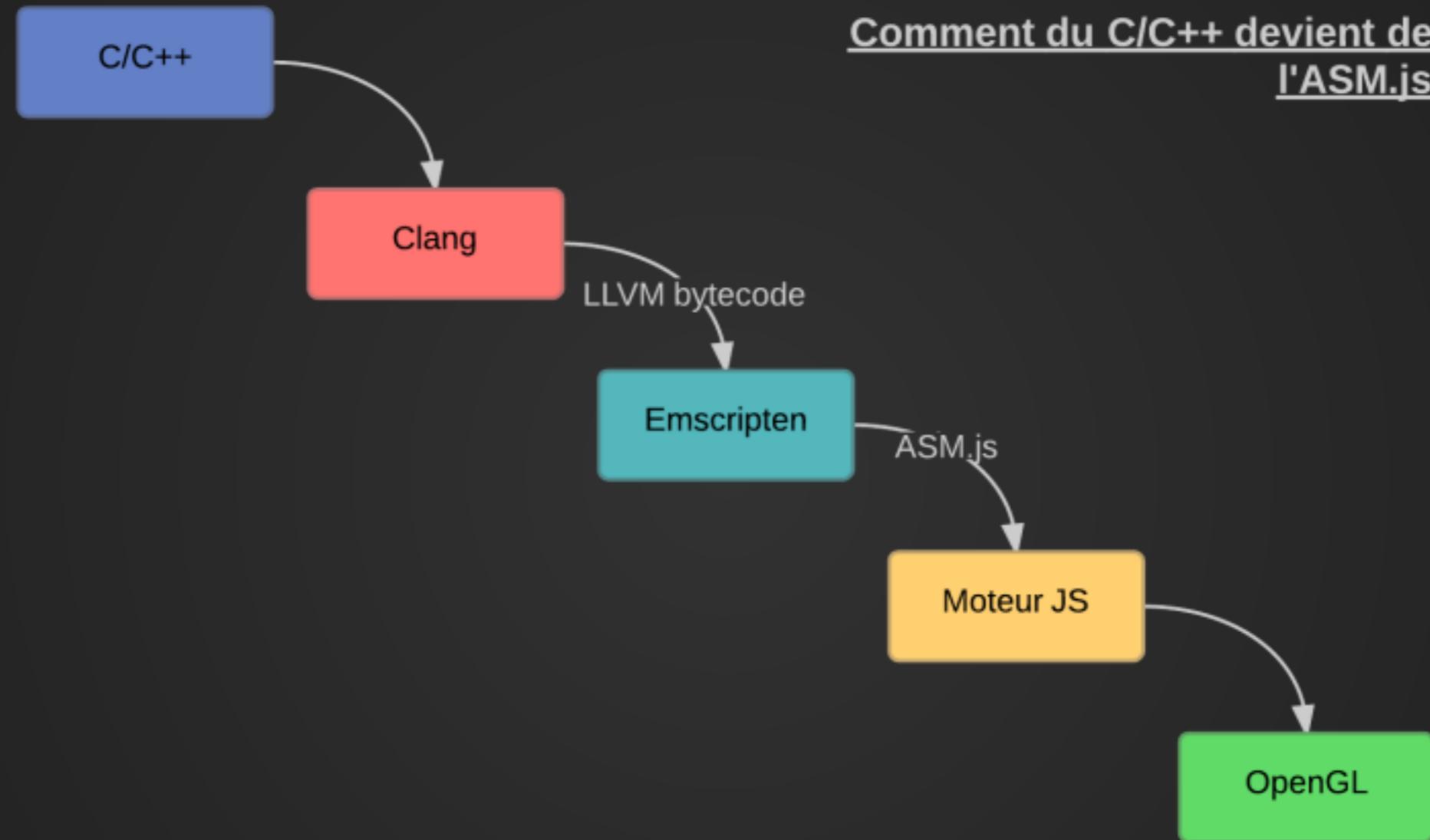


ASM.js

Compatible tous butineurs
Accéléré depuis la version 22 de FIREFOX



ASM.js



ASM.js

Vraiment plus rapide ?

Bananabread : goo.gl/gFN5jp

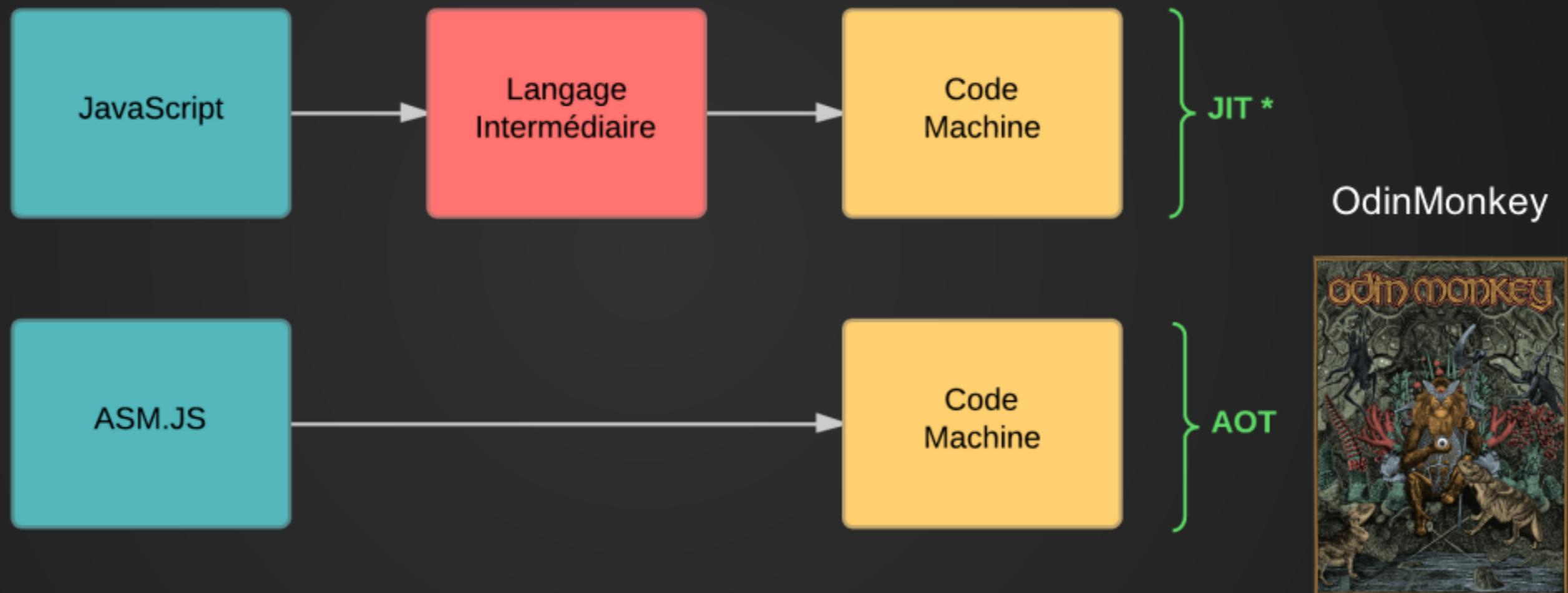


Ok, comment c'est possible ?

- Suppression des types complexes (String, Object, Boolean)
- Manipulation uniquement de type numérique sur 32 bits
- Utilisation massive des opérateurs bitwise
- Unboxing en type primitif
- Statiquement typé
- Désactivation du garbage collector
- Disparition des vérifications runtime



Un AOT compiler plutôt qu'un JIT compiler



Show me the code

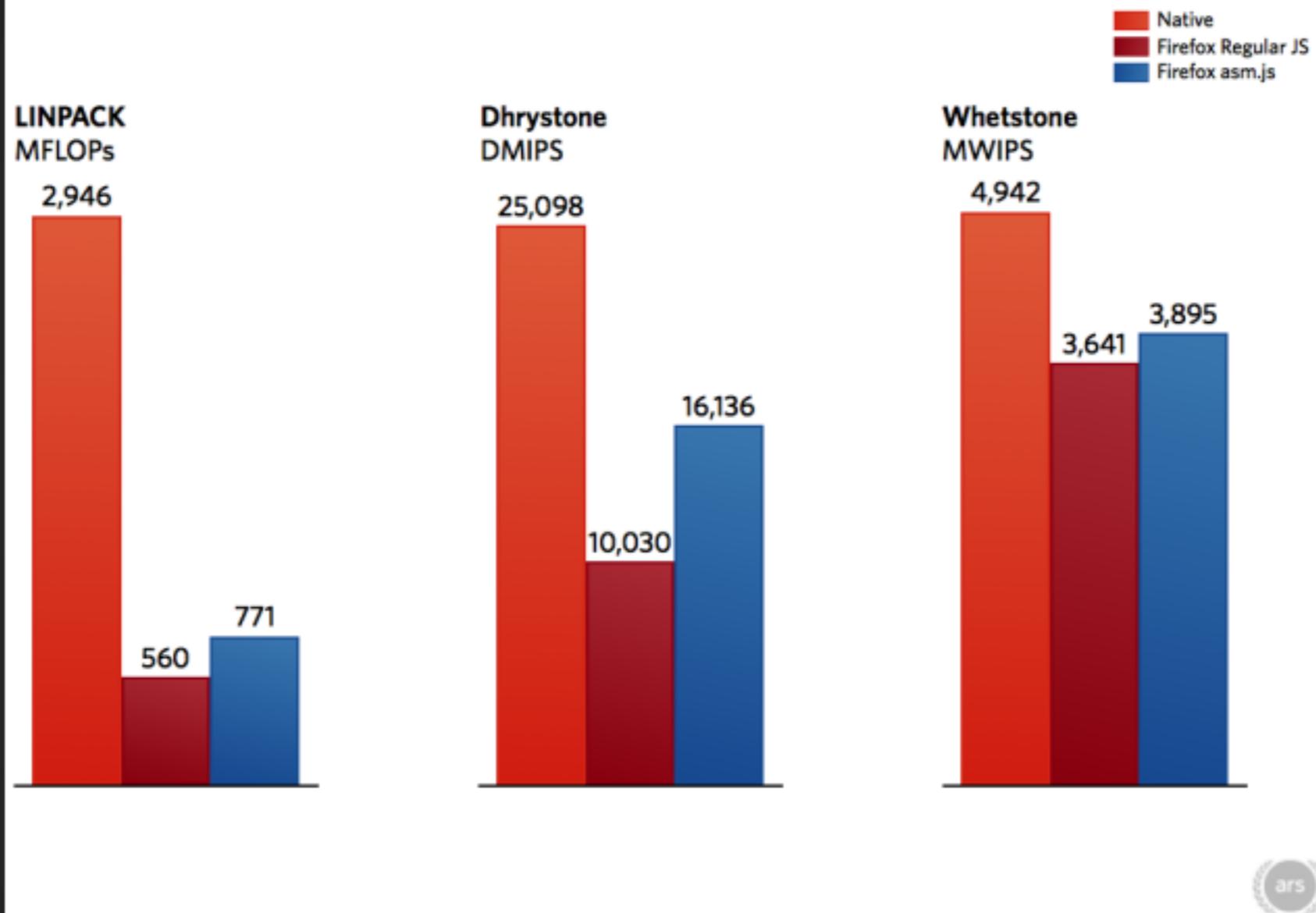
```
function MyAsmModule(stdlib, foreign, heap) {
    "use asm";

    // module body...

    return {
        export1: f1,
        export2: f2
        // ...
    };
}
```

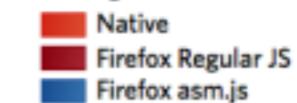


Classic benchmarks: Native code vs. Emscripten vs. asm.js

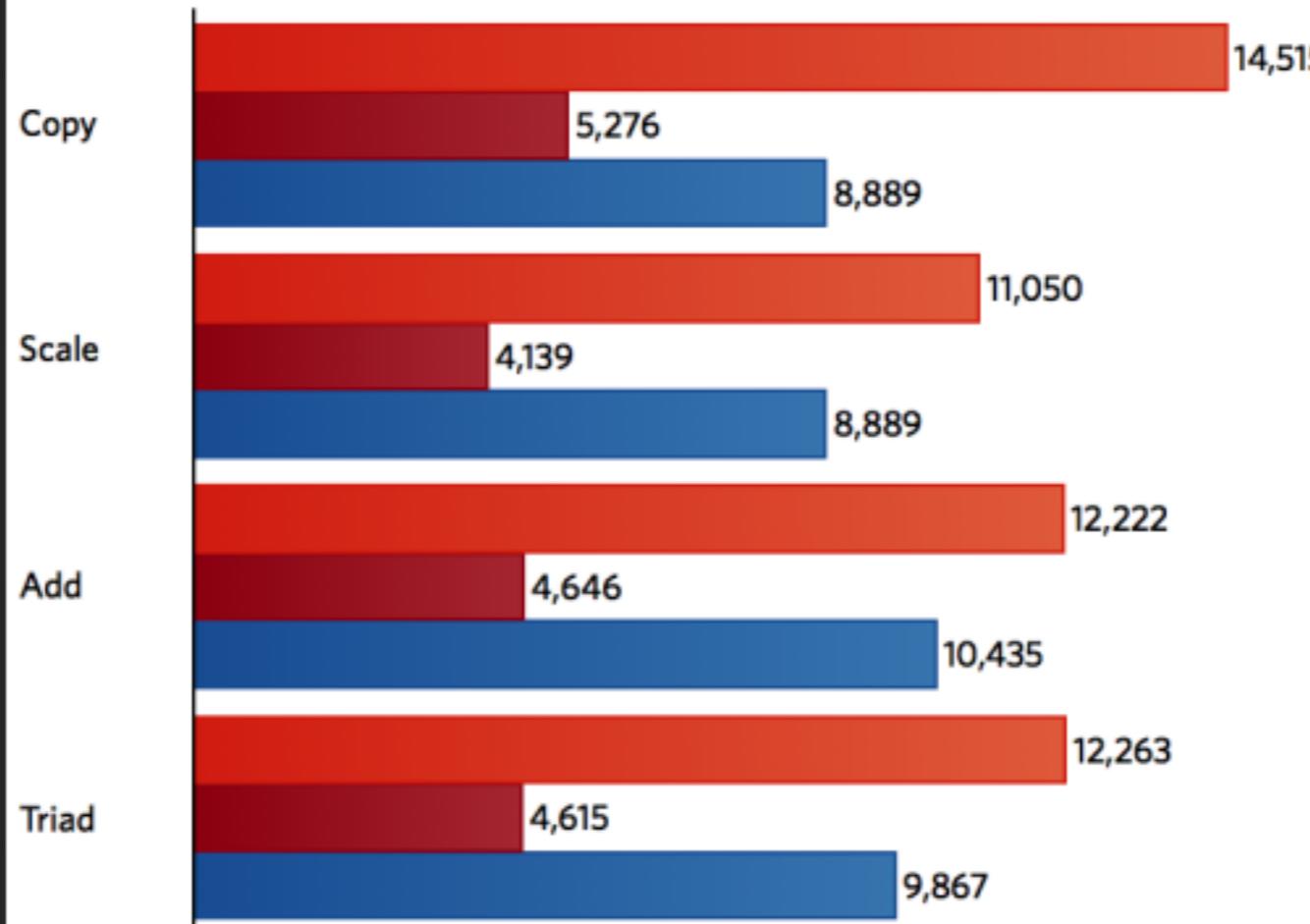


STREAM benchmarks: Native code vs. Emscripten vs. asm.js

MBps



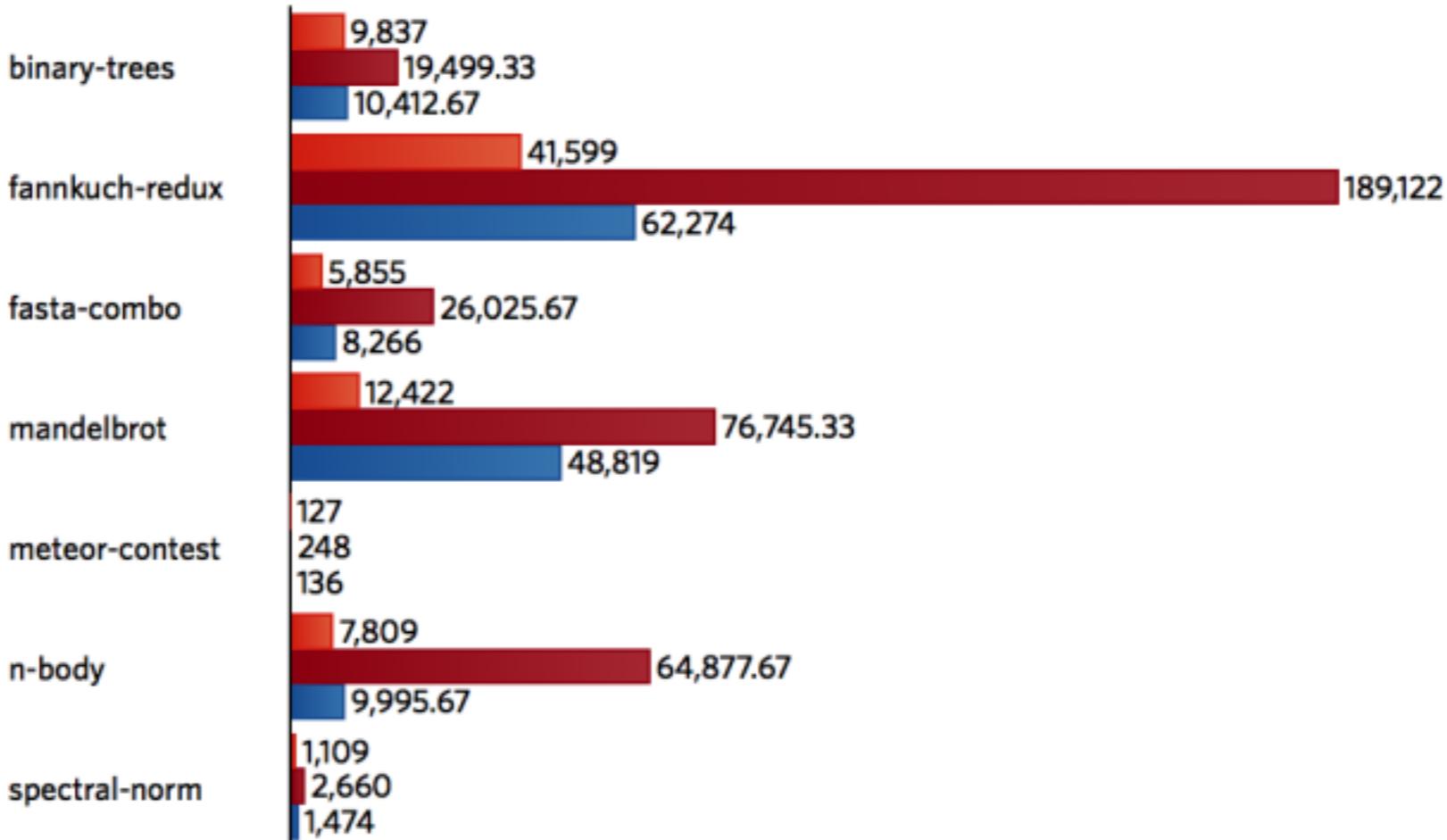
Benchmark	Native	Firefox Regular JS	Firefox asm.js
Copy	14,515	5,276	8,889
Scale	11,050	4,139	8,889
Add	12,222	4,646	10,435
Triad	12,263	4,615	9,867



Benchmark game: Native code vs. Emscripten vs. asm.js

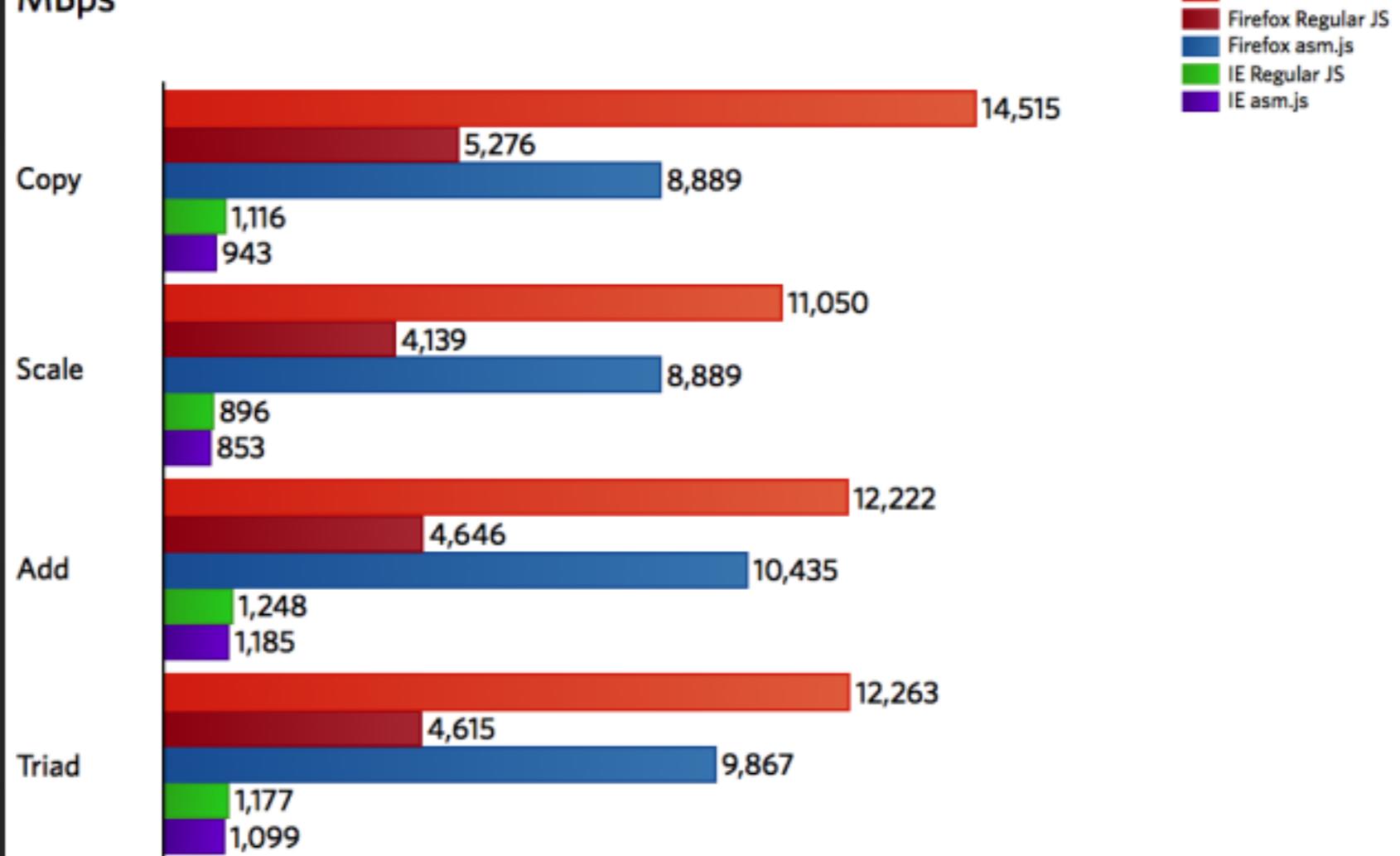
Milliseconds

Native
Firefox Regular JS
Firefox asm.js

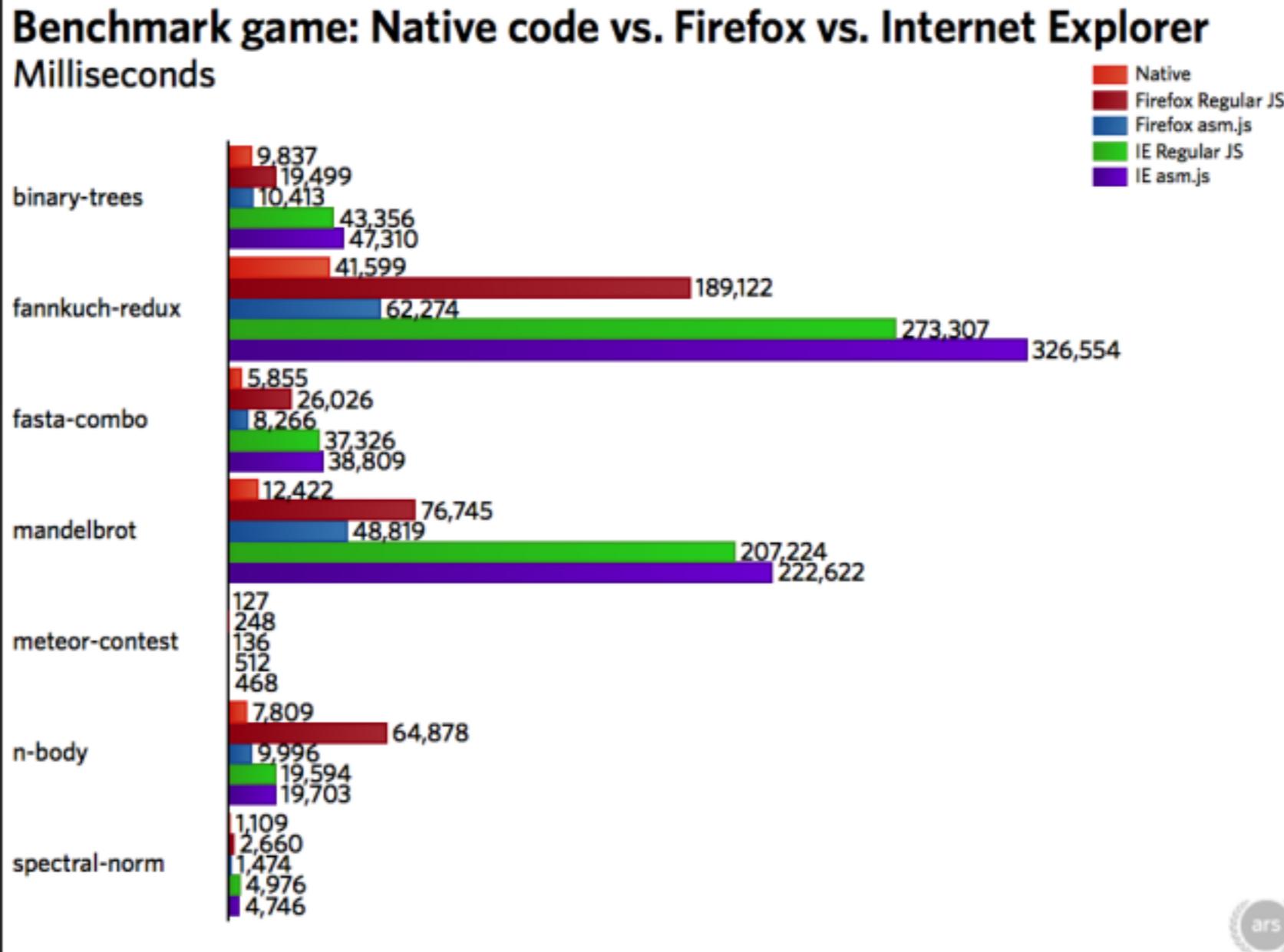


STREAM benchmarks: Native code vs. Firefox vs. Internet Explorer

MBps



ASM.js



ASM.js

Déjà plusieurs PORTAGES en C/C++

Moteur de jeux (Unreal Engine)

Base de données (SQLite)



ASM.js

Demo

C/C++



JavaScript



So...

So..



So...

Un projet évolue, au début le typage est superflux à terme
il est vital



So...

Les grands du web on compris que JavaScript ne suffisait pas.



So...

CoffeeScript et TypeScript utilisables à court terme
Mais ne corrigent pas tous (`this`, `perf`)



So...

Dart est le plus complet et de loin
Stratégique pour Google (Chrome & Android ?)



So...

ASM.js est très prometteur
Gestion mémoire manuel, pour l'instant
Le JNI de JavaScript ?



Merci



Question ??

