



NODE.JS?

Monitoria - Engenharia de
Software

Me apresentando... E não porque quero...



Sidney Alex



saaa@cin.ufpe.br



@silvercent011



S1d3A

Me apresentando... E não porque quero...



André Filho



acasf@cin.ufpe.br



@mrdedede





Web 1.0

"Read Only",
Decentralized



Web 2.0

Participatory,
Centralized



Web 3

No Intermediaries,
Decentralized

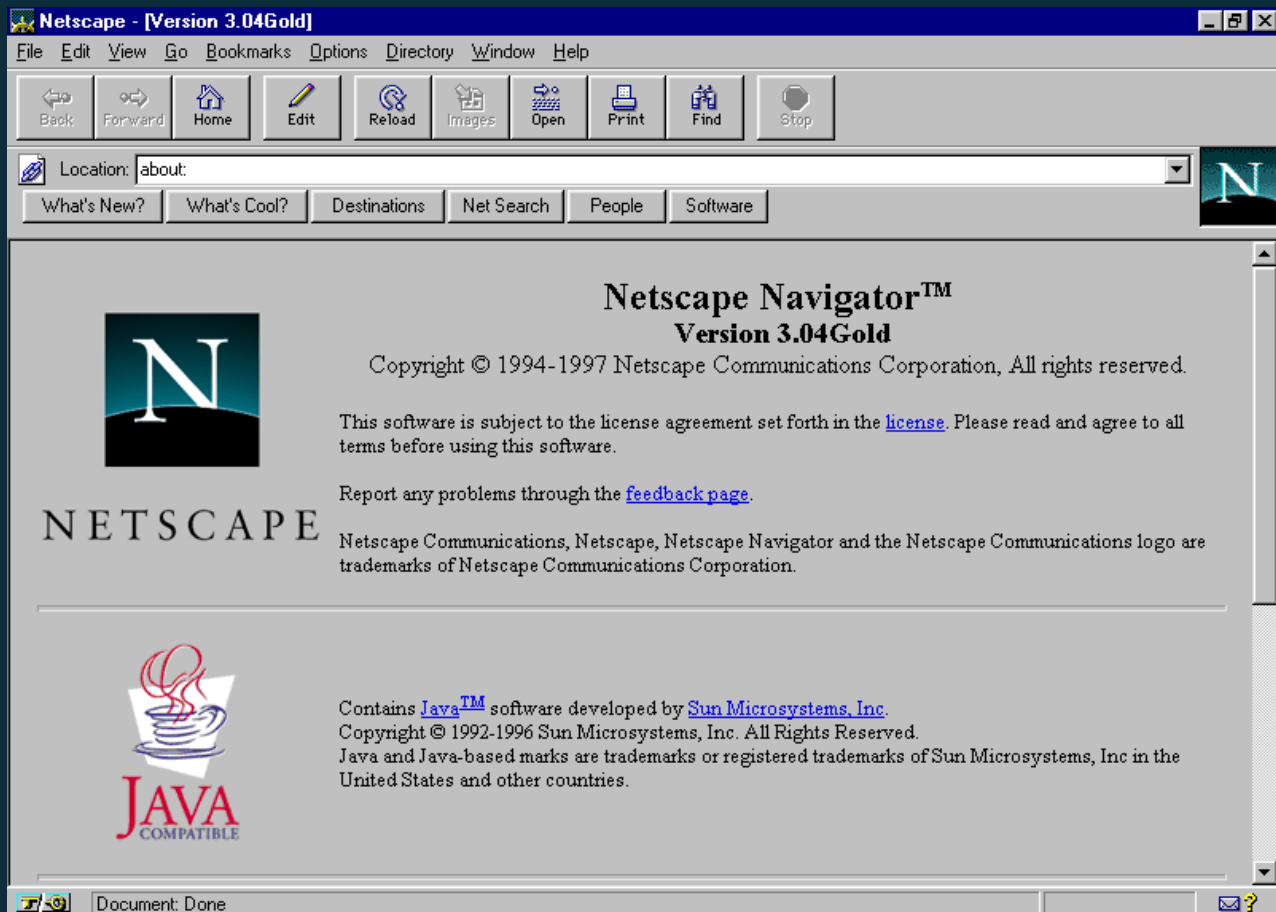


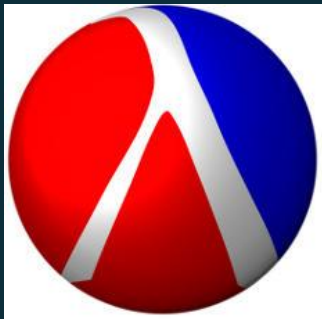


Web 1.0

"Read Only",
Decentralized







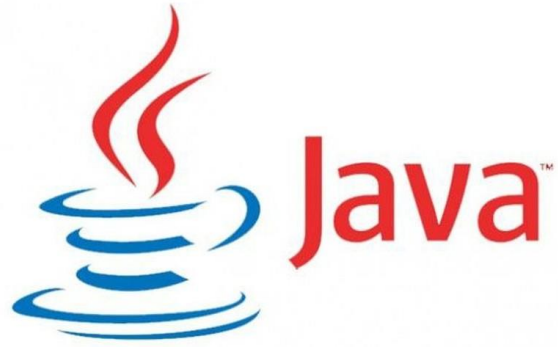
```
//Scheme Code
```

```
(import  
  (rnrs)  
  (ironscheme clr))
```

```
;Define a function write-ln  
(define (write-ln fmt . args)  
  (clr-static-call System.Console WriteLine  
(clr-cast System.String fmt)  
  (clr-cast System.Object[]  
    (list->vector args)))))
```

```
; And invoke it!
```

```
(write-ln "{0}" "Hello World!")
```





Brendan Eich
(esse bichão aí)
Criador do JavaScript



Brendan Eich
(esse bichão aí)
Criador do JavaScript



O JavaScript foi criado para funcionar apenas em client-side (browser)

PCWEEK

How to Implement

NETSCAPE LIVEWIRE PRO

**From Conception to Launch:
Gain Control of Complex,
Interactive Web Sites**

- Take charge with the latest drag-and-drop site management techniques
- Create and deploy well-organized Web sites—then easily edit and update them locally
- Harness Netscape's development advantages of JavaScript and LiveWire's built-in compiler
- Master server-side programming that lets clients browse, search, and update networked databases



Amy Helen Johnson and Warren Ernst



Web 1.0

“Read Only”,
Decentralized



Web 2.0

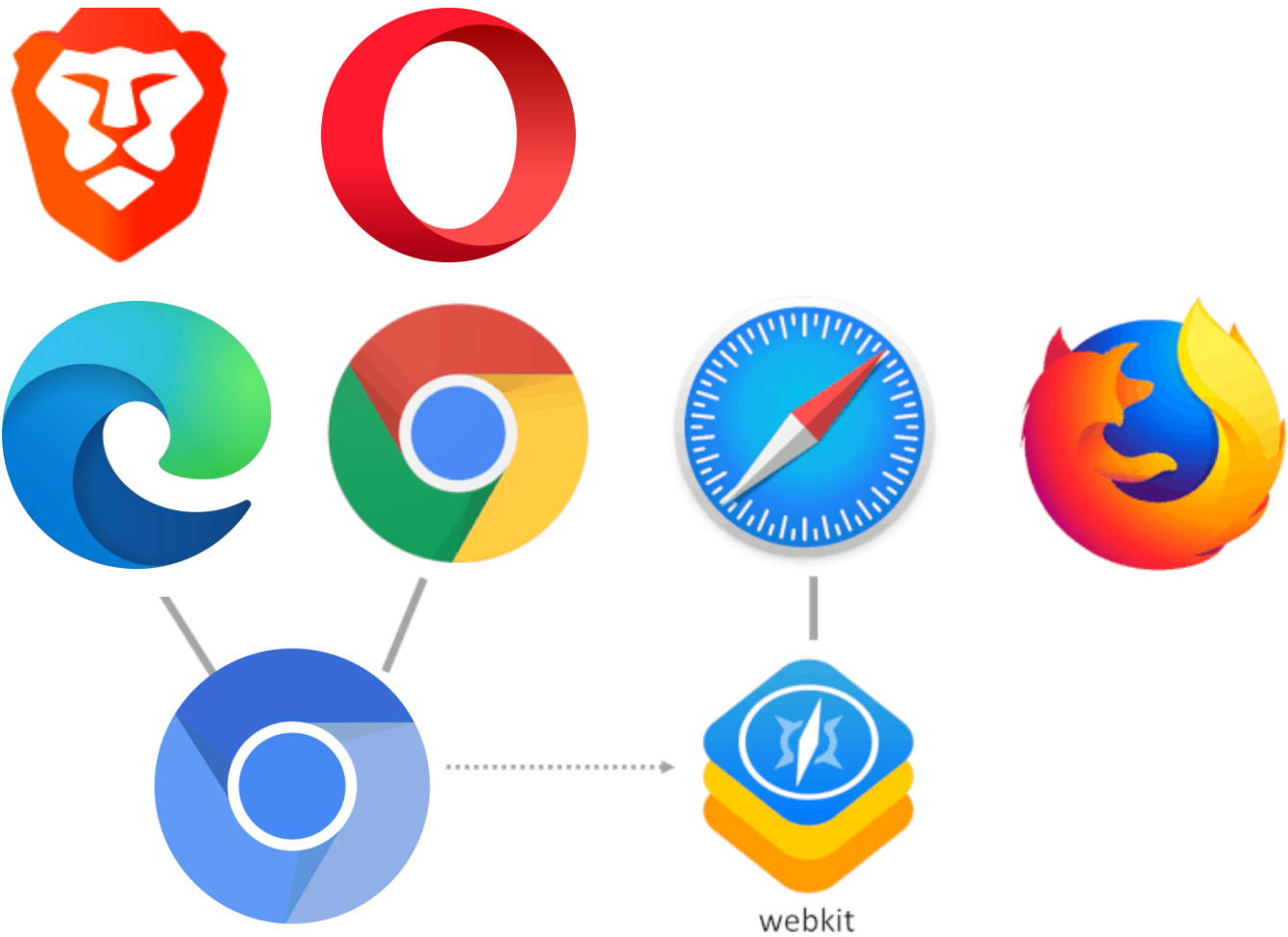
Participatory,
Centralized







webkit



Browser

Chrome

Browser Engine

Chromium

Runtime

JavaScript

V8

Renderer

HTML & CSS

Blink





Ryan Dahl
(achou que eu ia repetir
a piada aqui?)
Criador do Node.js



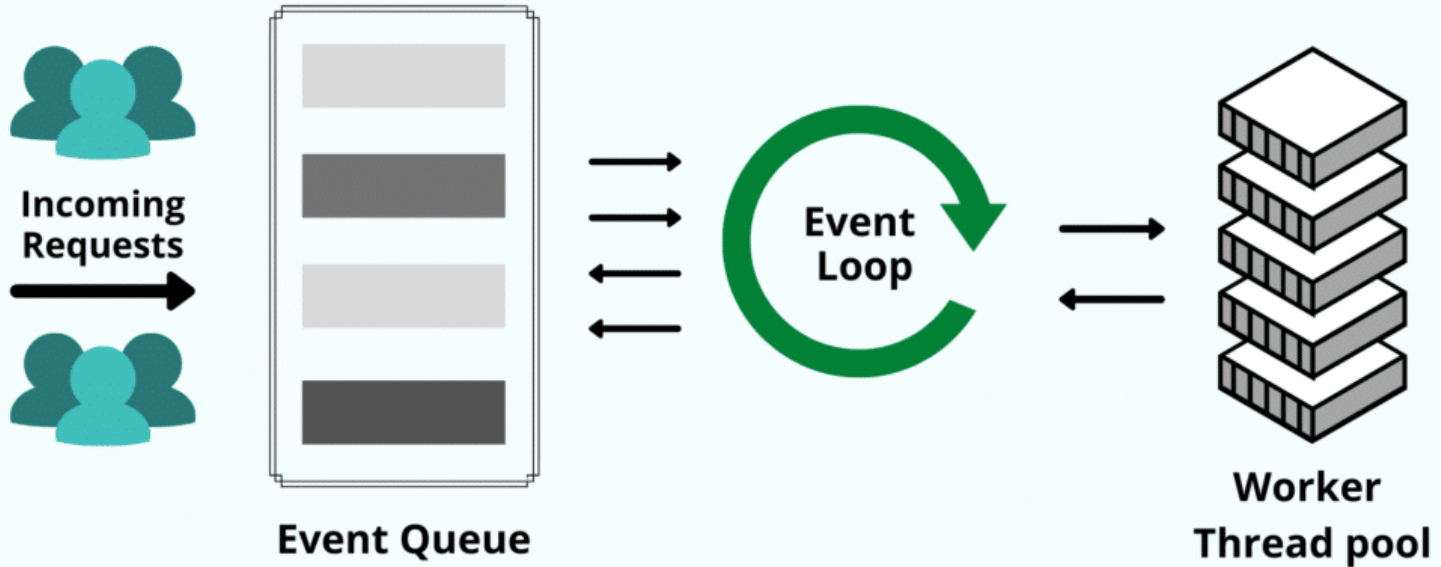


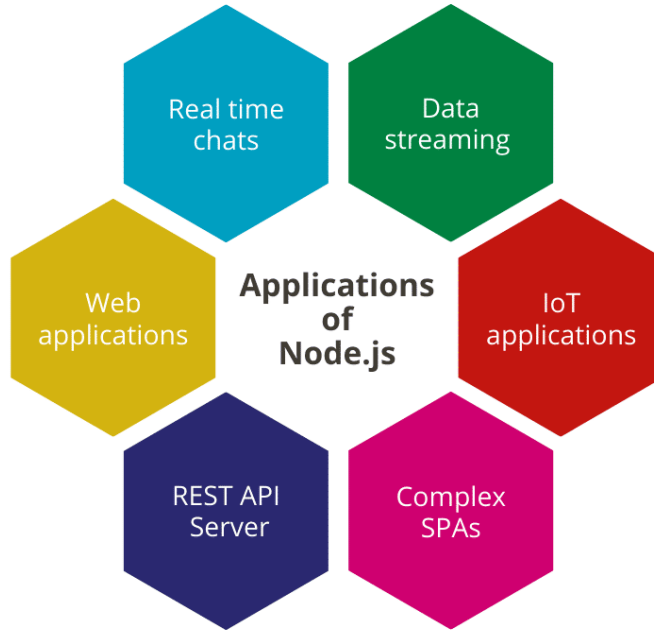
○ Node.js permite utilizar o
JavaScript no server-side
(Servidor)

O Node.js permite utilizar o
JavaScript no ~~server-side~~
(Servidor)

FORA DO BROWSER

Node.js Architecture

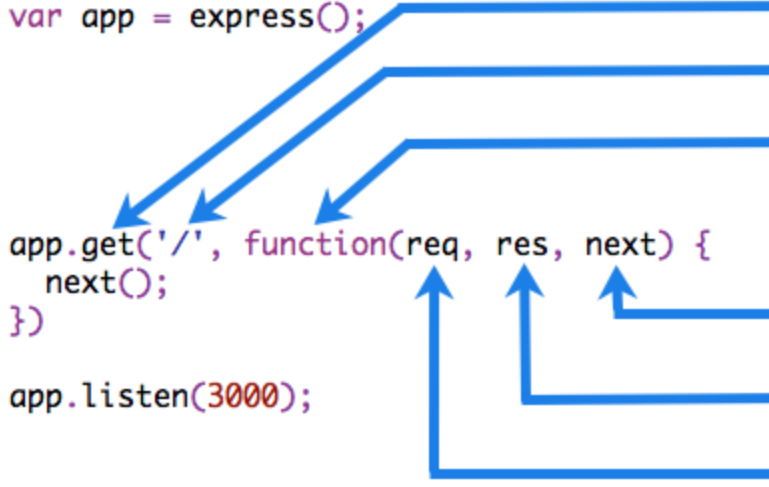




Express

Framework web
rápido, flexível e
minimalista para
Node.js

```
var express = require('express');  
var app = express();
```



The diagram consists of blue arrows. One arrow originates from the `express` variable in the first line of code and points to the `app` variable in the second line. Another arrow originates from the `express` variable and points to the `function` parameter in the `app.get` method call. A third arrow originates from the `app` variable and points to the `app.get` method call. A fourth arrow originates from the `next()` call inside the `function` and points to the `next` parameter in the `function` signature. A fifth arrow originates from the `app.listen` call and points to the `app` variable in the `app.get` call. A sixth arrow originates from the `app` variable and points to the `app.get` call. A seventh arrow originates from the `app` variable and points to the `app` variable in the `app.listen` call. An eighth arrow originates from the `app` variable and points to the `app` variable in the `app.get` call. A ninth arrow originates from the `app` variable and points to the `app` variable in the `app.get` call. A tenth arrow originates from the `app` variable and points to the `app` variable in the `app.get` call.

```
app.get('/', function(req, res, next) {  
  next();  
});
```

```
app.listen(3000);
```



Material Complementar + Listas (GitHub)

DEMO