

# JAMSTACK

*THE TOTAL VICTORY OF JAVASCRIPT*



@swyx // @jscamp

# DISTRIBUTION

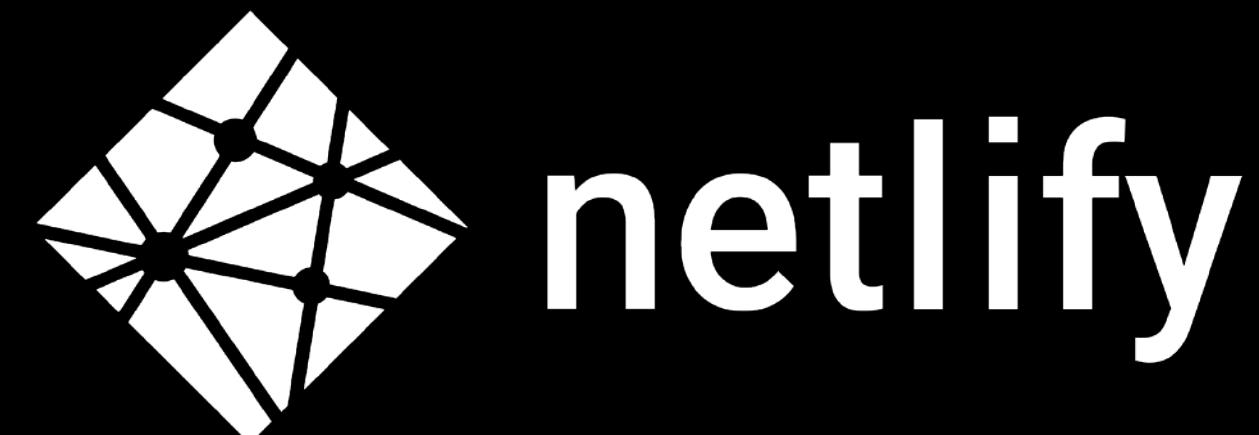






@swyx

#LearnInPublic



**ISN'T JAMSTACK JUST ANOTHER  
WORD FOR STATIC SITES?**

SOME PEOPLE, WHO HAVE BEEN AROUND A WHILE

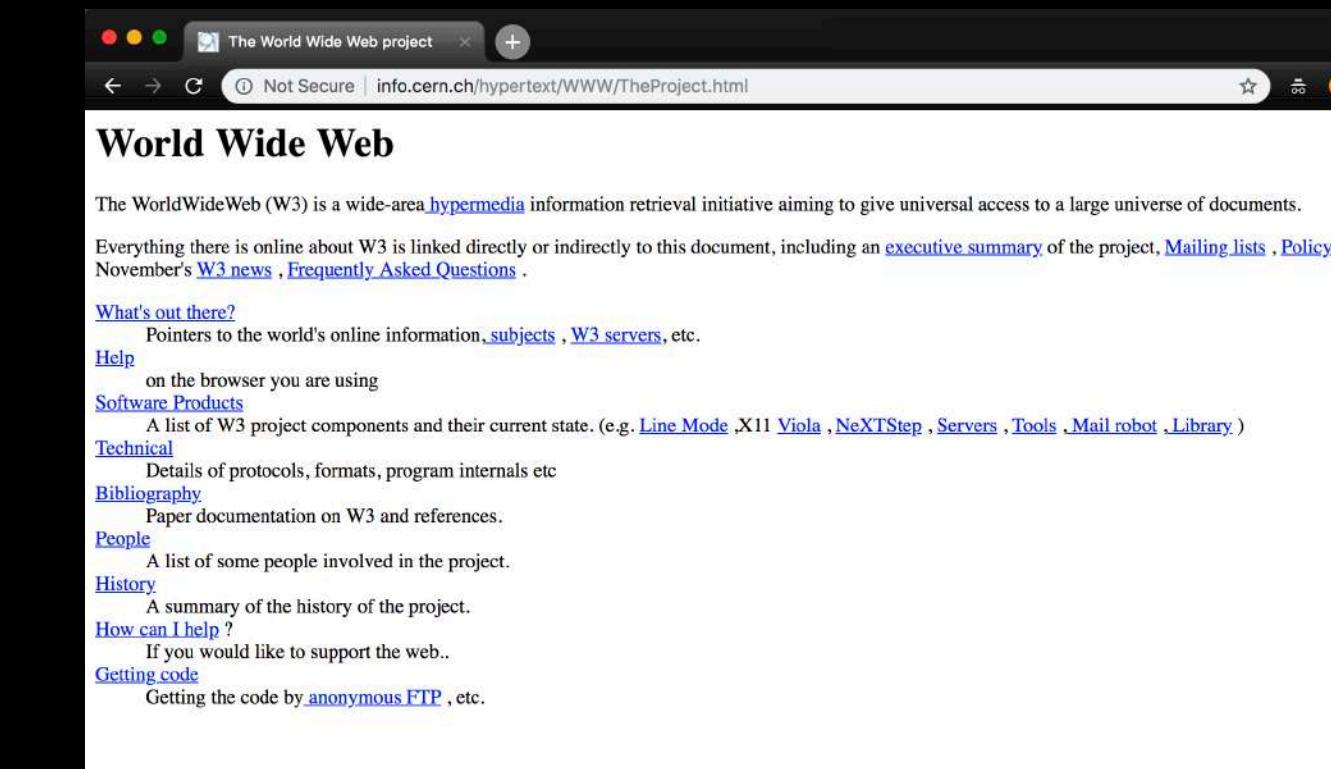
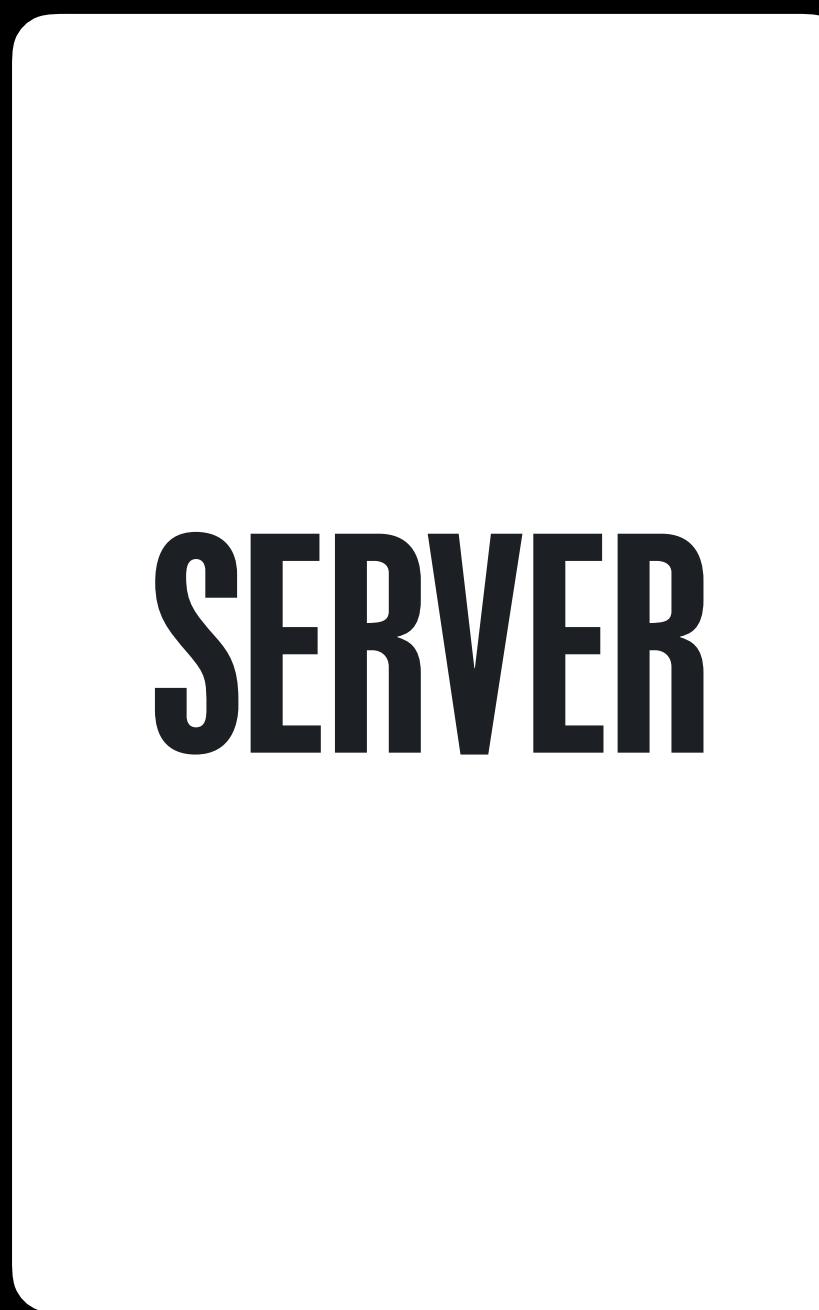
# jamstack

*A modern architecture —*

*Create fast and secure sites and dynamic apps  
with JavaScript, APIs, and pre-rendered  
Markup, served without web servers*

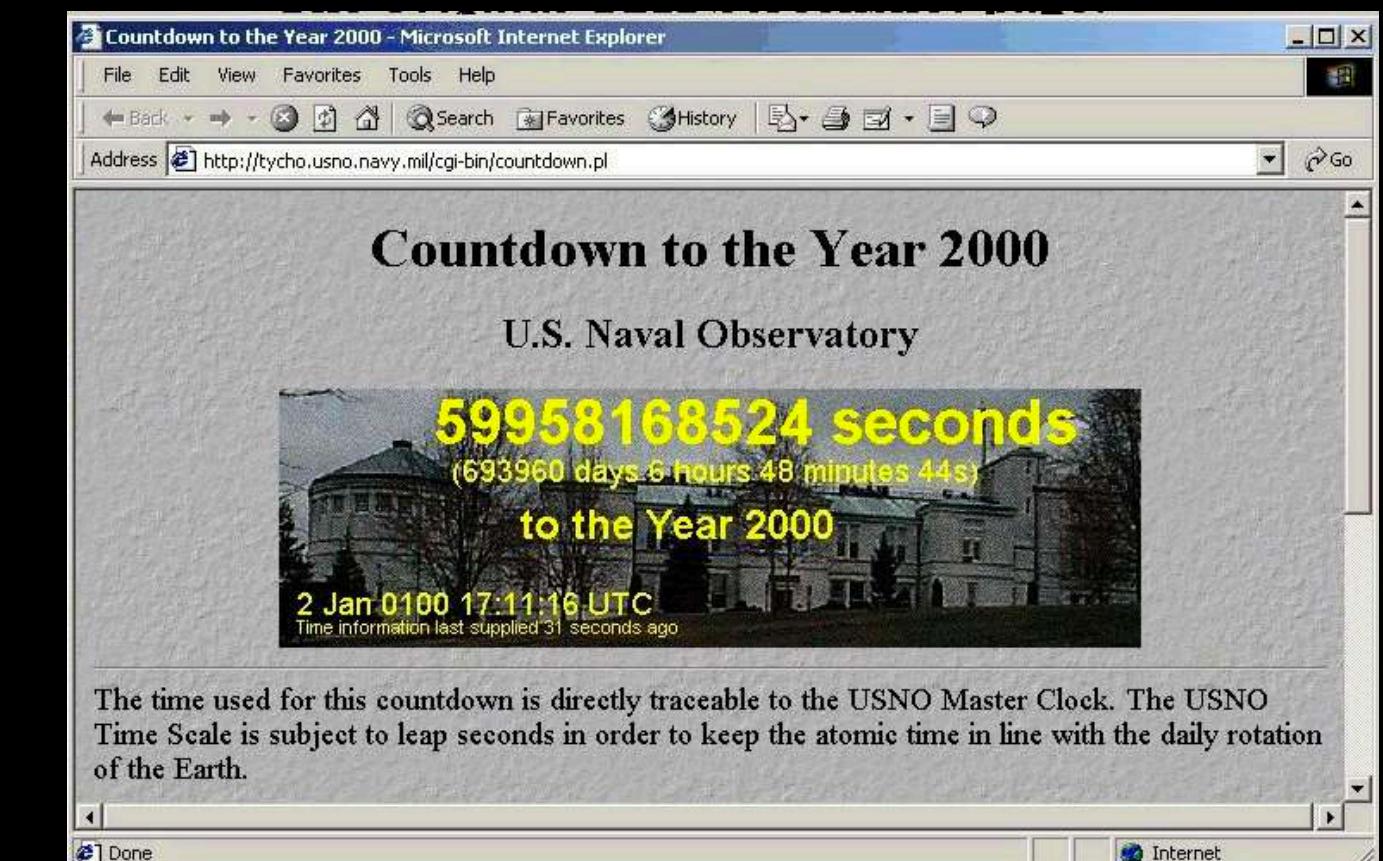
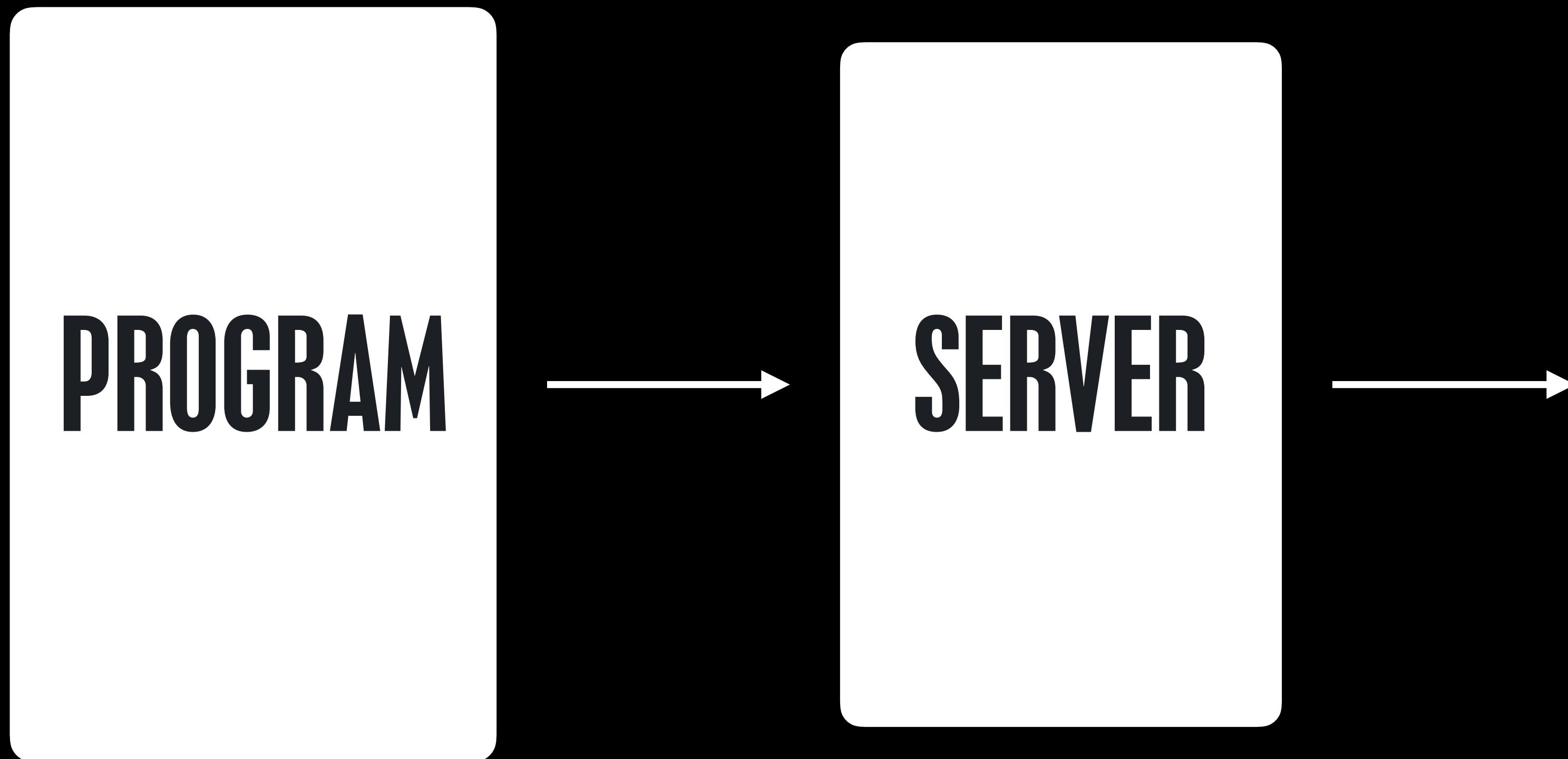
---

# 1. IN THE BEGINNING





# DYNAMIC



# GENERATED FROM DATA

DATA-BASE



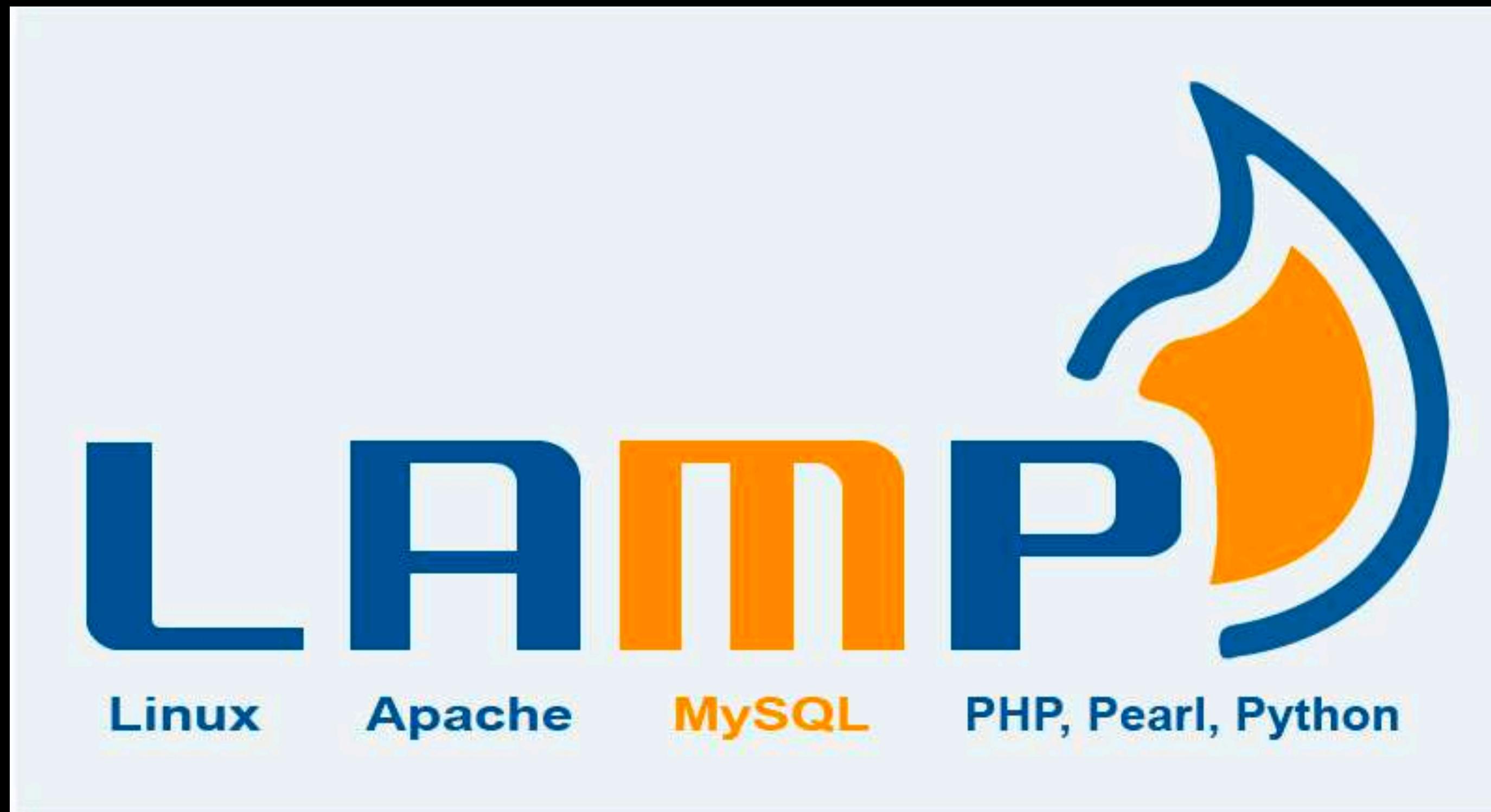
PROGRAM



SERVER



# SERVERSIDE ONLY



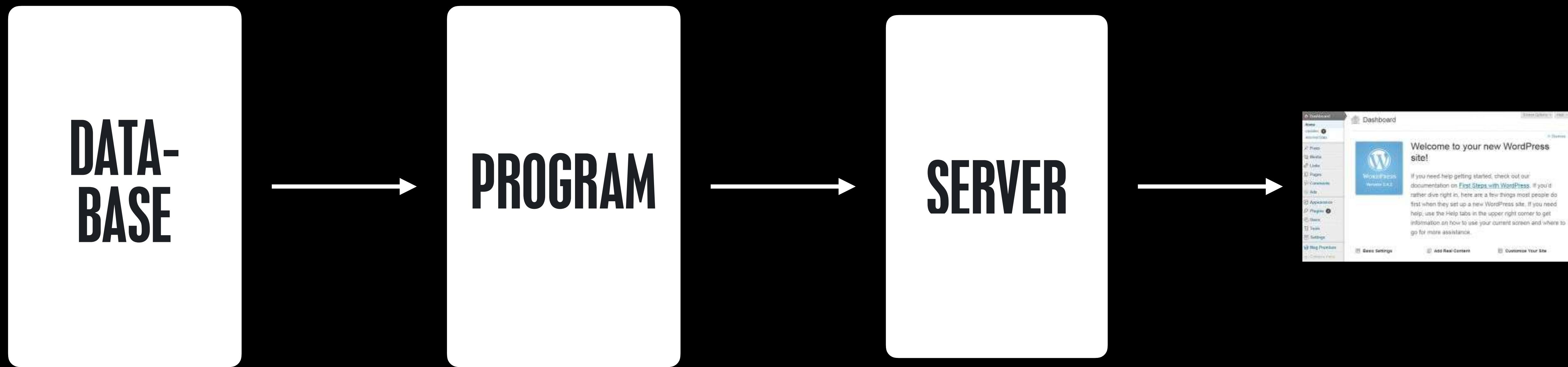
1990'S - 2000'S

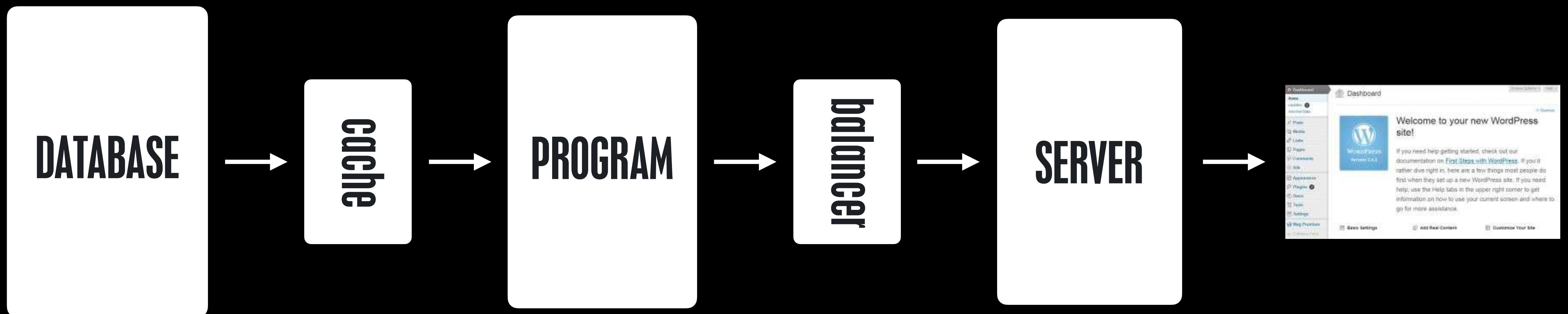


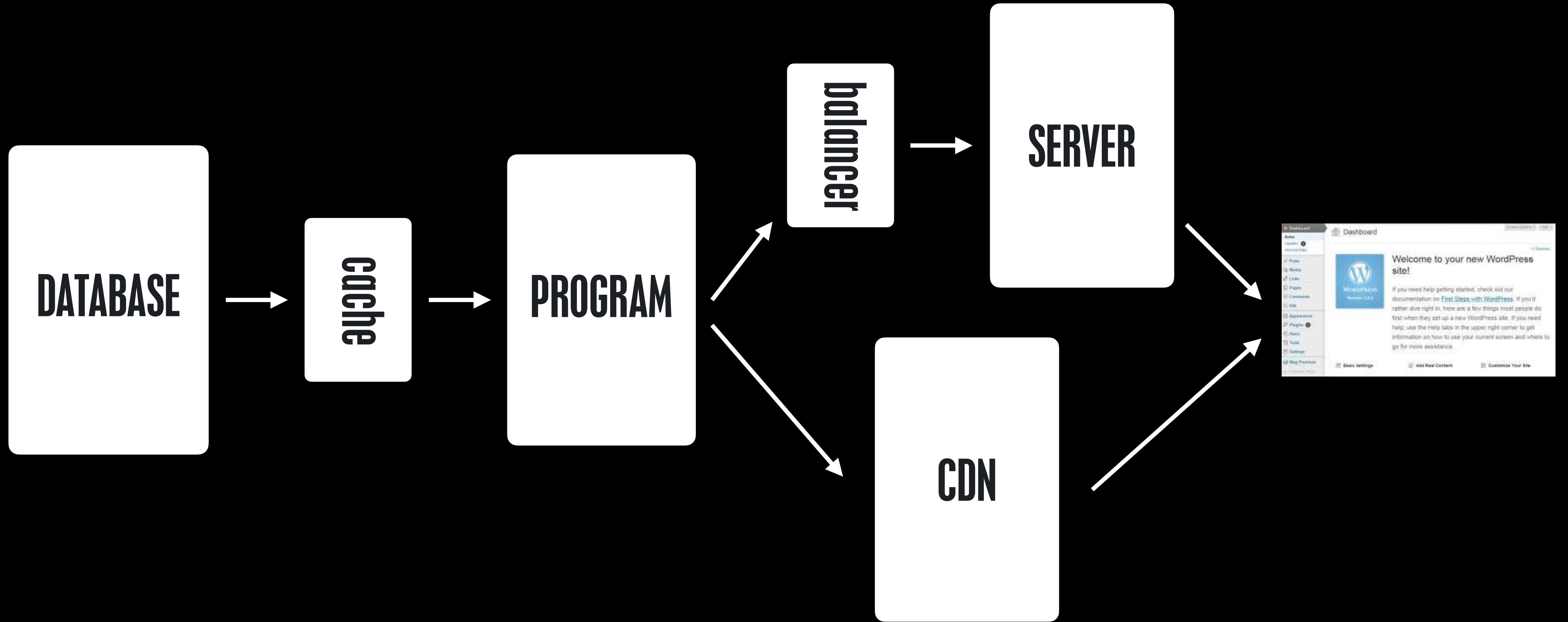


# 1. IN THE BEGINNING: LAMP STACK

# 2. LAMP PROBLEMS

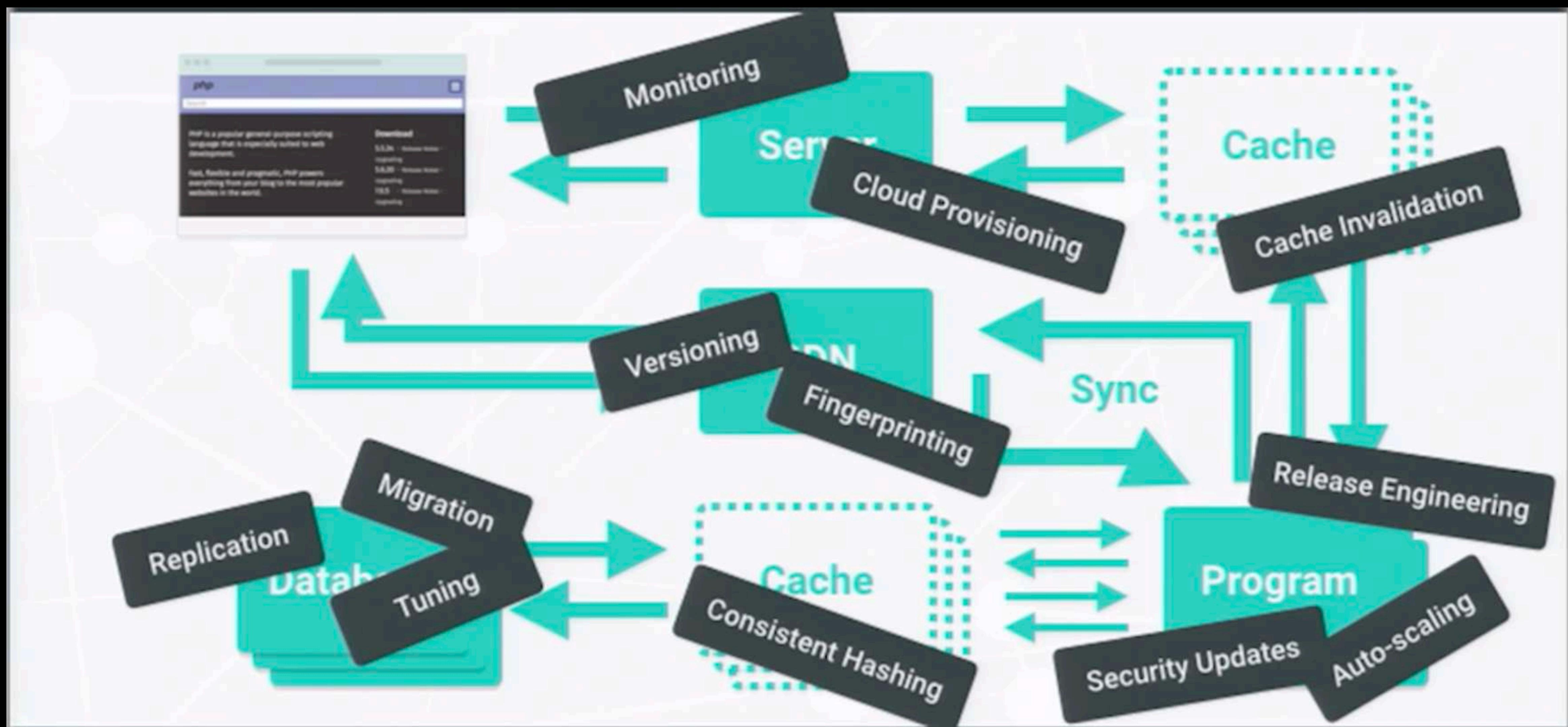






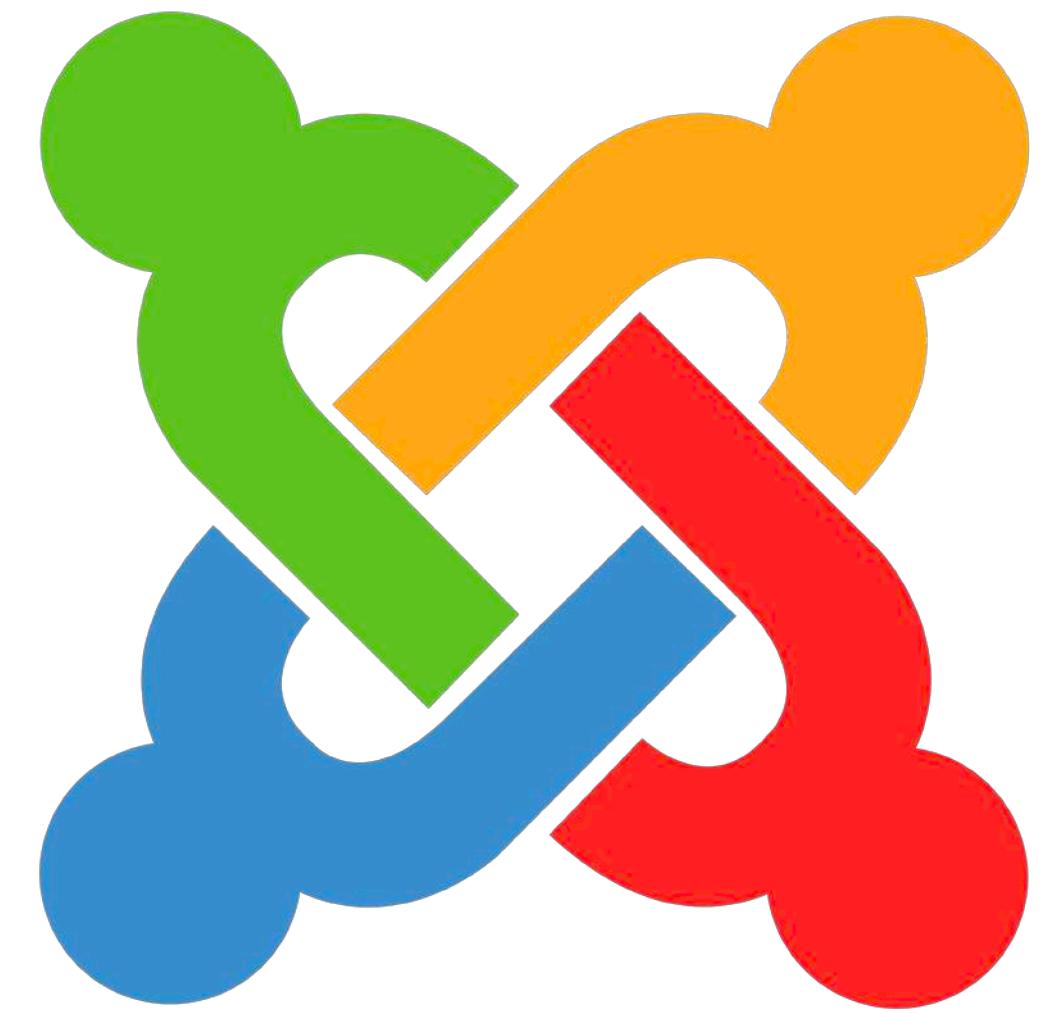
# PROBLEMS

1. MONITORING
2. VERSIONING
3. REPLICATION /  
MIGRATION
4. CONSISTENT HASHING
5. FINGERPRINTING
6. SECURITY UPDATES
7. AUTOSCALING
8. RELEASE ENGINEERING
9. CACHE INVALIDATION
10. CLOUD PROVISIONING





# NOT JAMstack



Drupal™



SQUARESPACE

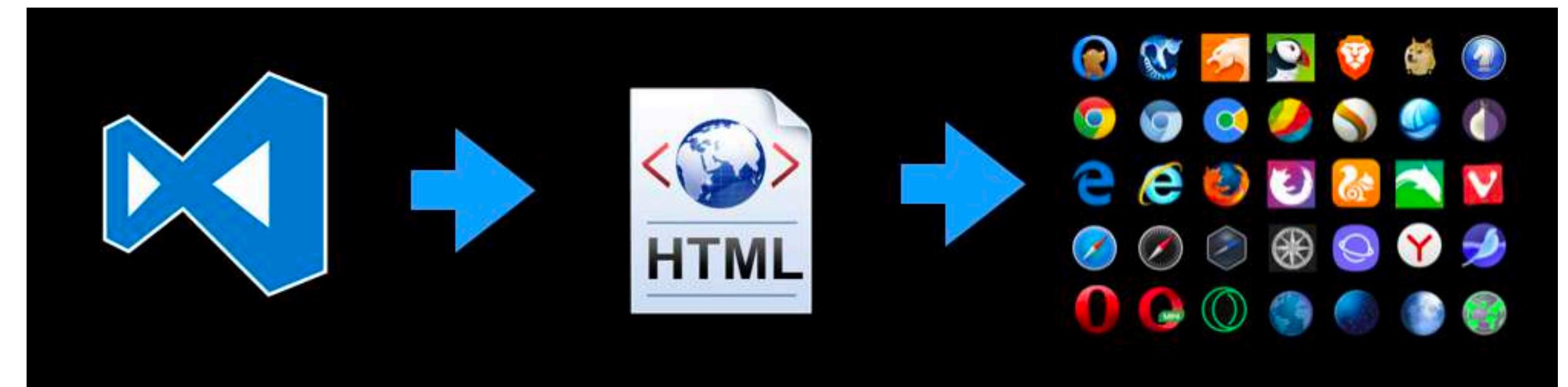
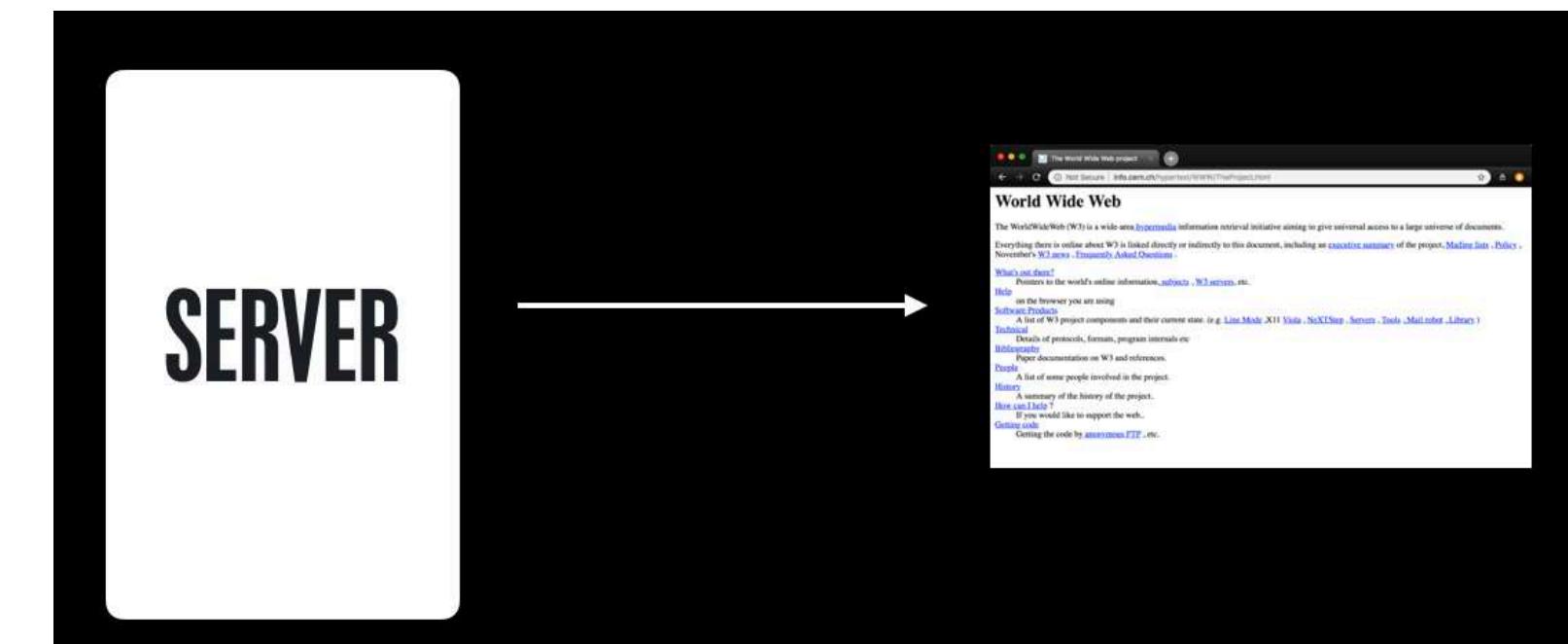
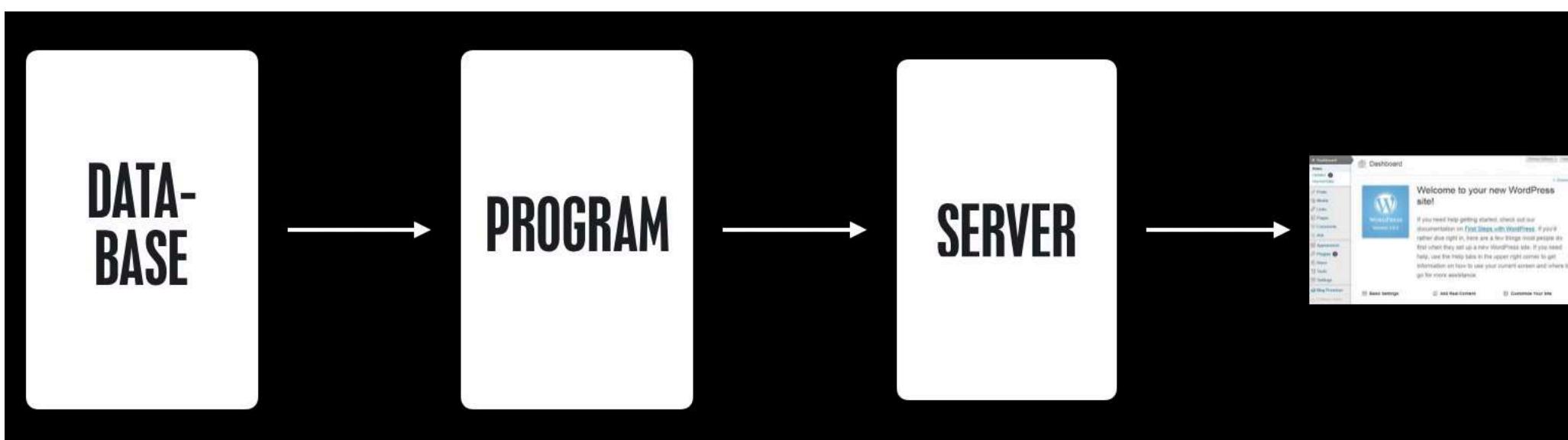
# LAMP PROBLEMS

SECURITY  
RELIABILITY  
COMPLEXITY/COST

**TRADITIONAL STACKS  
ADD STATIC LAYERS  
ADD INFRASTRUCTURE  
IN ORDER TO IMPROVE  
PERFORMANCE**



# STATIC SITE GENERATORS (V1)





# SSG's are (JA)M-stack



HUNDREDS more

*jamstack*

JAVASCRIPT & API & MARKUP

YOU DON'T HAVE TO USE ALL THREE

The screenshot shows a web browser displaying the CSS-Tricks article titled "JAMstack? More like SHAMstack.". The page has a dark background with a large, stylized letter "A" graphic. At the top, there's a navigation bar with links for Articles, Videos, Almanac, Snippets, Newsletter, Jobs, and Guides, along with a search bar containing the placeholder "e.g. flexbox". Below the header, the main content area features the article title in large white font. To the left of the title, there's a small profile picture of the author, Chris Coyier, and his bio: "Author Chris Coyier". Below the author info, there are links for "7 Comments" and "Join the Conversation". To the right of the author info, there are publication details: "Published Jun 5, 2019" and "Updated Jun 5, 2019". A large, semi-transparent "A" graphic is visible in the background of the main content area. On the right side of the page, there's a white sidebar containing a quote from the author: "To me, the most important part of JAMstack is rooted in the concept of static file hosting. Static file hosting is the foundation of all the power. It opens up a bunch of doors, like:" followed by a bulleted list of three items.

To me, the most important part of JAMstack is rooted in the concept of static file hosting. Static file hosting is the foundation of all the power. It opens up a bunch of doors, like:

- **Everything can be CDN-hosted.** "The edge," as they say. Even the HTML (the M in JAMStack also refers to Markup) can be CDN-hosted, which you otherwise can't do. That gives you an amazing base of speed that encourages you to keep that speed as you build.
- **The project feels easier to work with.** Git clone, npm install, build. Deployments are git pushes of a dist folder. It's so cool, for example, [Netlify](#) gives you a URL for every build, even on branches you're working on. This is made possible by deploys being kind of immutable. A set of files at a particular point in time.
- **Cloud functions are awesome.** Because you don't have a traditional server-side language to reach for, you build it with [cloud functions](#) when you do need server-side code — which is a cool way to architect things anyway and is very spiritually connected to all this.

# JAMSTACK

*means*

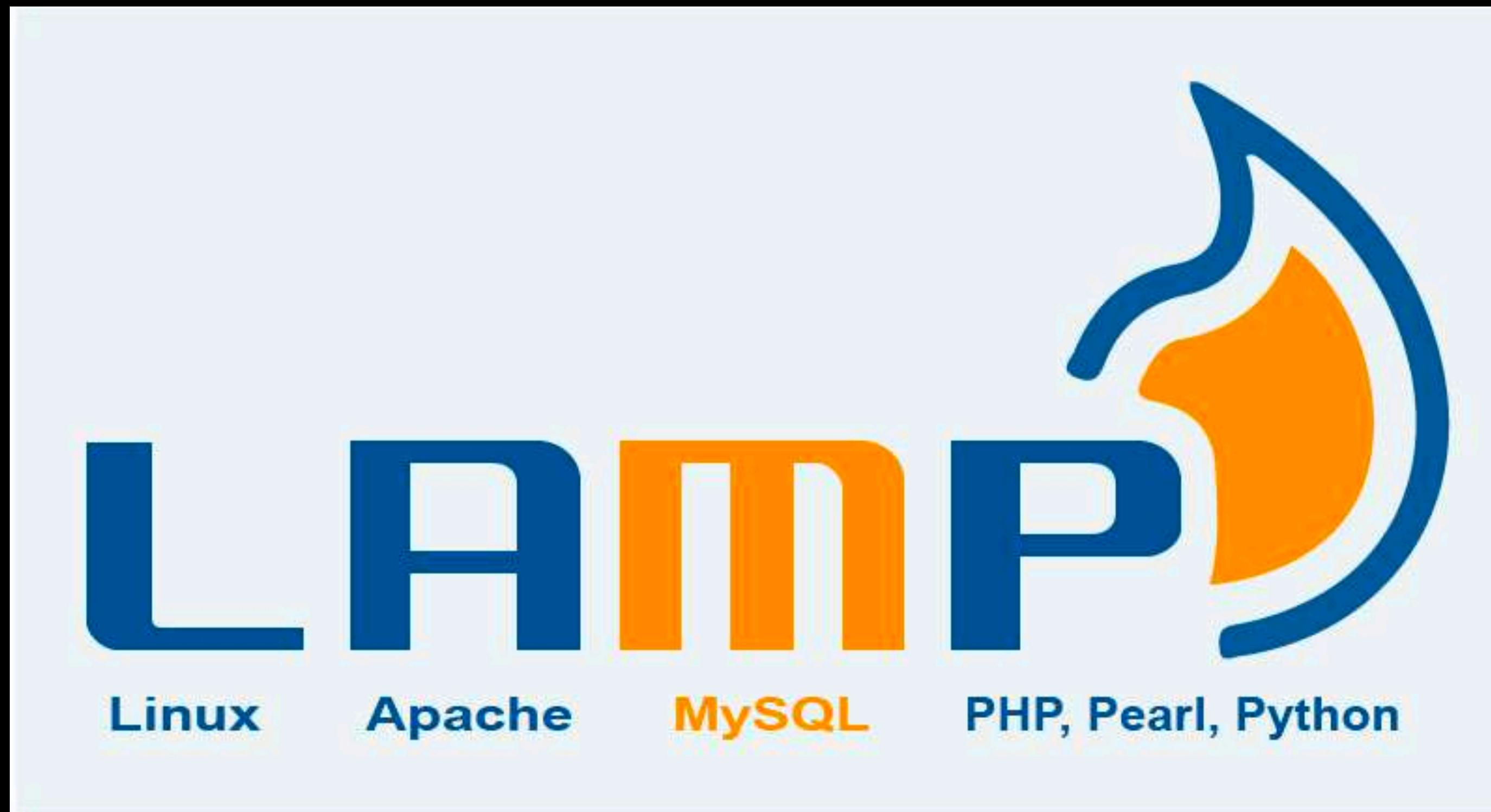
CAPABLE OF BEING SERVED  
DIRECTLY FROM A CDN

- 1. IN THE BEGINNING: LAMP STACK**
- 2. LAMP PROBLEMS**
- 3. MEAN STACK**

# JAVASCRIPT GOT SERIOUS

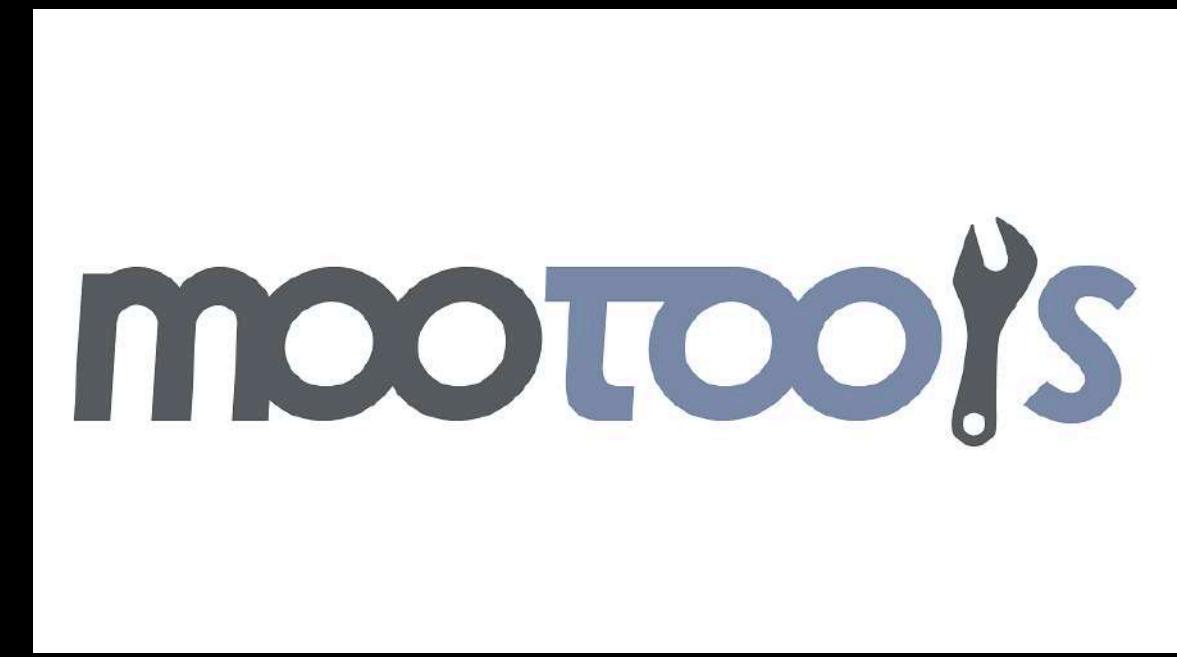
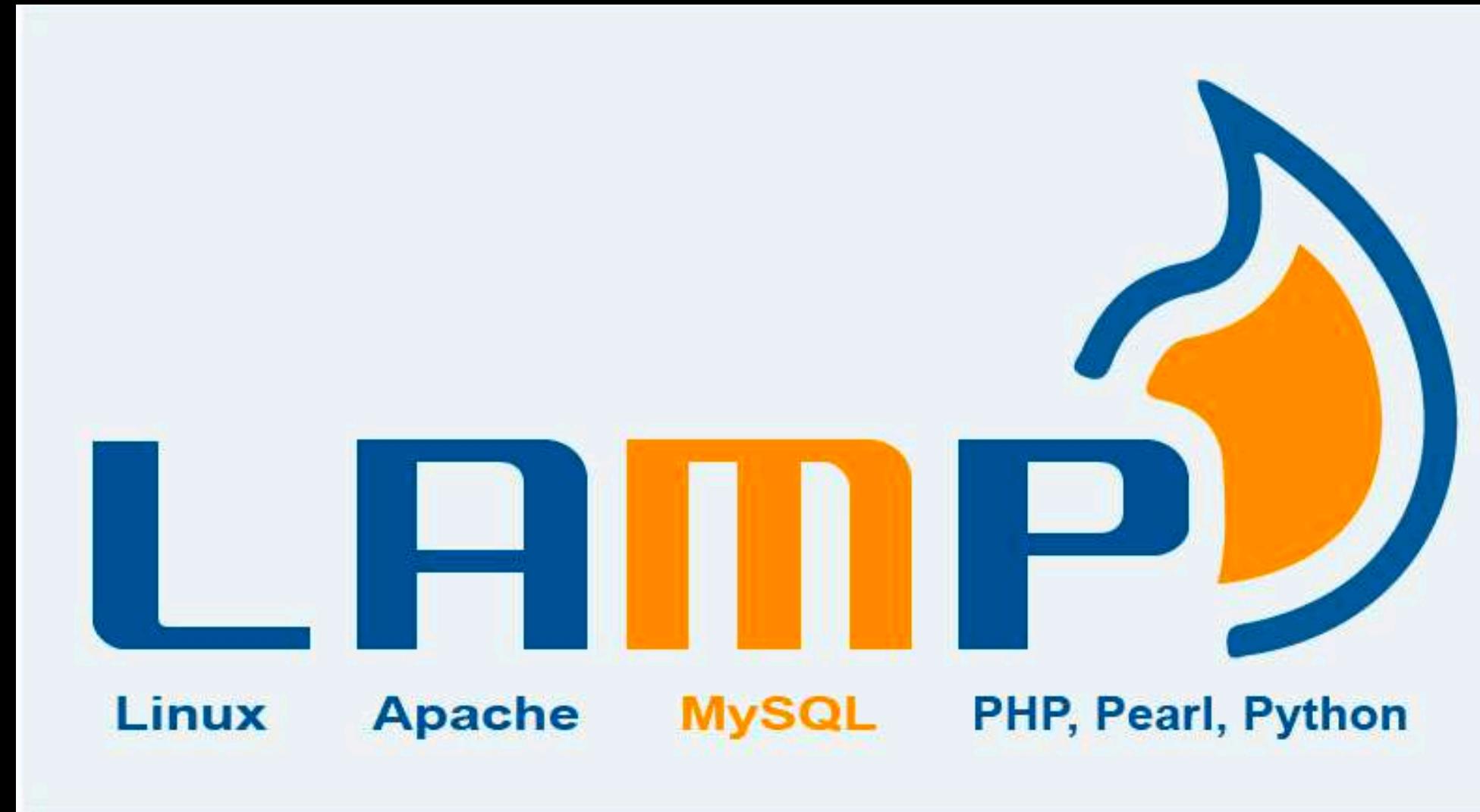


# SERVERSIDE ONLY



1990'S - 2000'S

# SERVER-AND-CLIENTSIDE



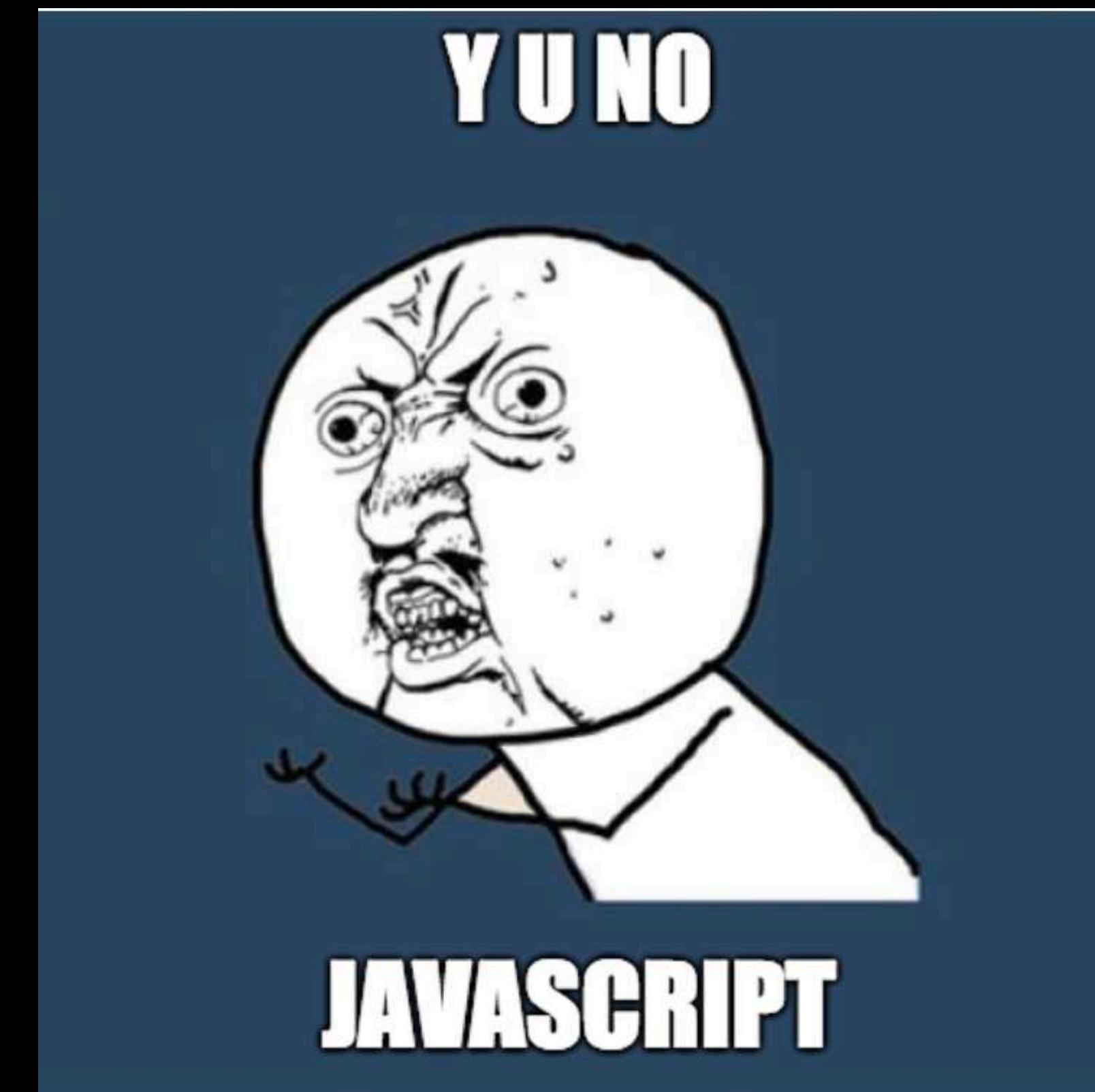
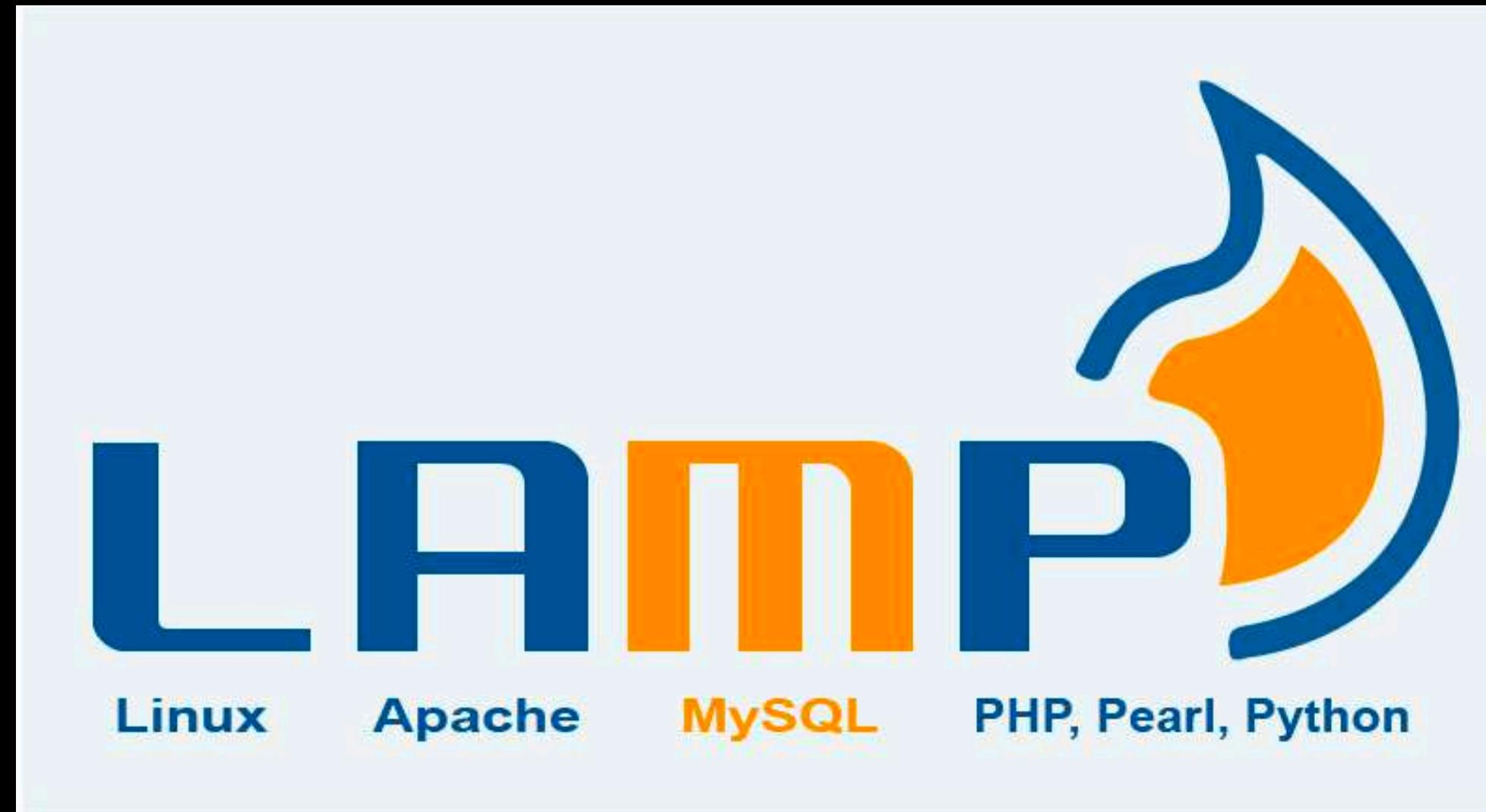
LATE - 2000'S



# ATWOOD'S LAW

Any application that can be written in **JavaScript**,  
will eventually be written in **JavaScript**.

<https://blog.codinghorror.com/the-principle-of-least-power/>



# EARLY 2010'S

The MongoDB homepage features a prominent banner at the top with the text "The MEAN Stack: MongoDB, ExpressJS, AngularJS and Node.js" and a "Try Free" button. Below the banner, there's a search bar and a sidebar with various navigation links. The main content area contains a blog post titled "Introducing the MEAN and MERN stacks" by Valeri Karpov, dated April 30, 2013. The post discusses the benefits of using a uniform language (JavaScript) throughout the stack. To the right, there's a newsletter sign-up form and a link to "The MongoDB Engineering Journal".

**Products Solutions Customers Resources Learn What is MongoDB? Contact Search Sign In Try Free**

# The MEAN Stack: MongoDB, ExpressJS, AngularJS and Node.js

[Explore the MEAN and MERN stacks](#)

Search all blog posts

MongoDB April 30, 2013 Releases

**January 2017** An updated post is available: [Introducing the MEAN and MERN stacks](#).

By Valeri Karpov, Kernel Tools engineer at MongoDB and co-founder of the [Ascot Project](#).

A few weeks ago, a friend of mine asked me for help with PostgreSQL. As someone who's been blissfully SQL-free for a year, I was quite curious to find out why he wasn't just using MongoDB instead. It turns out that he thinks MongoDB is too difficult to use for a quick weekend hack, and this couldn't be farther from the truth. I just finished my second 24 hour hackathon using Mongo and NodeJS ([the FinTech Hackathon](#) cosponsored by [10gen](#)) and can confidently say that there is no reason to use anything else for your next hackathon or REST API hack.

First of all, there are huge advantages to using a uniform language throughout your stack. My team uses a set of tools that we affectionately call the MFAN stack: MongoDB,

Sign up for our monthly newsletter

Name

Email Address

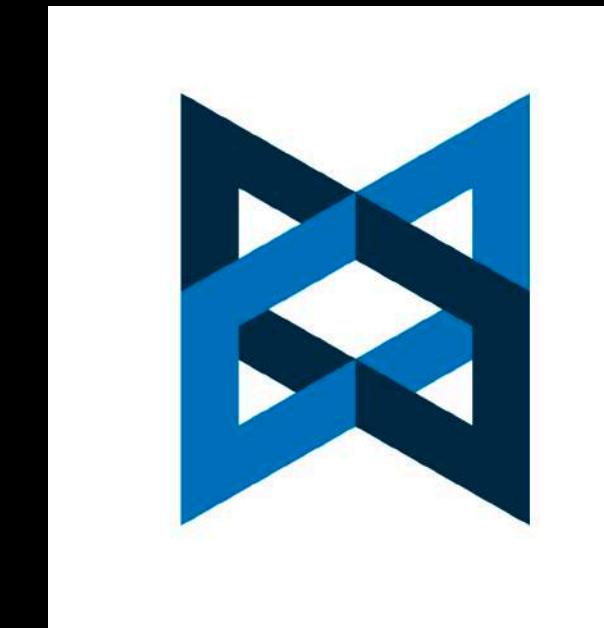
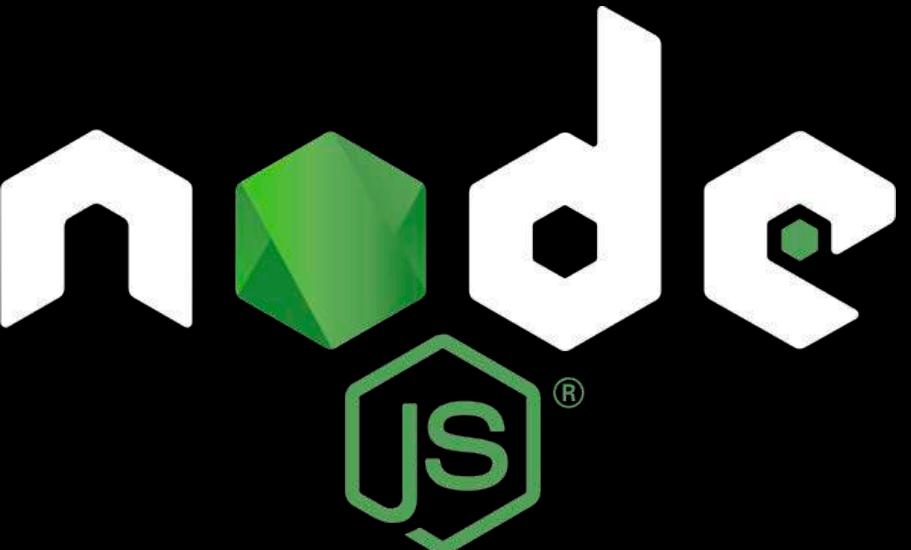
Submit

**The MongoDB Engineering Journal**

A tech blog for builders, by builders

[View Now](#)

# JAVASCRIPT WON! (YAY?)



*THE PYRRHIC VICTORY OF JAVASCRIPT*

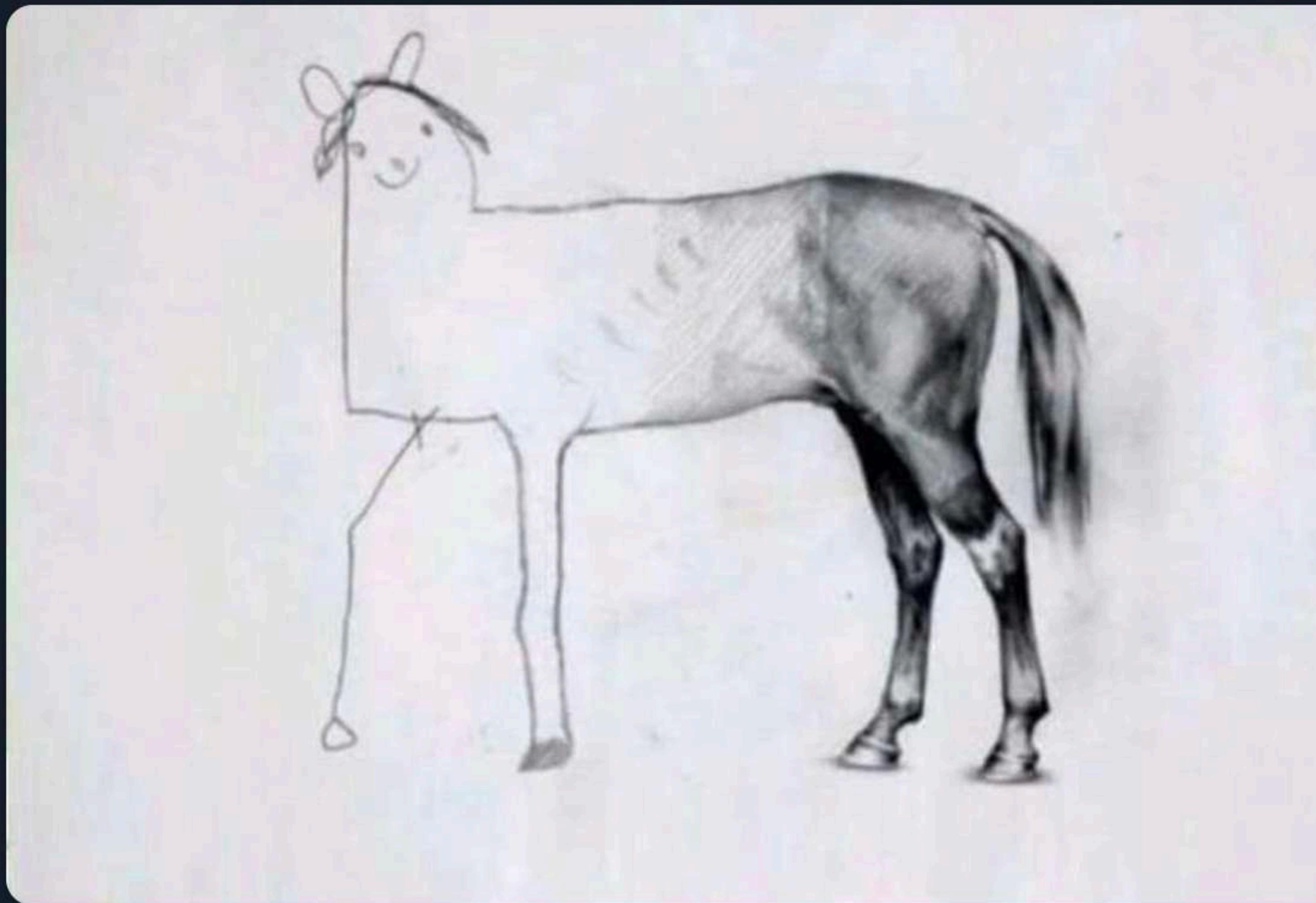
**Brian Holt**

@holtbt

Following

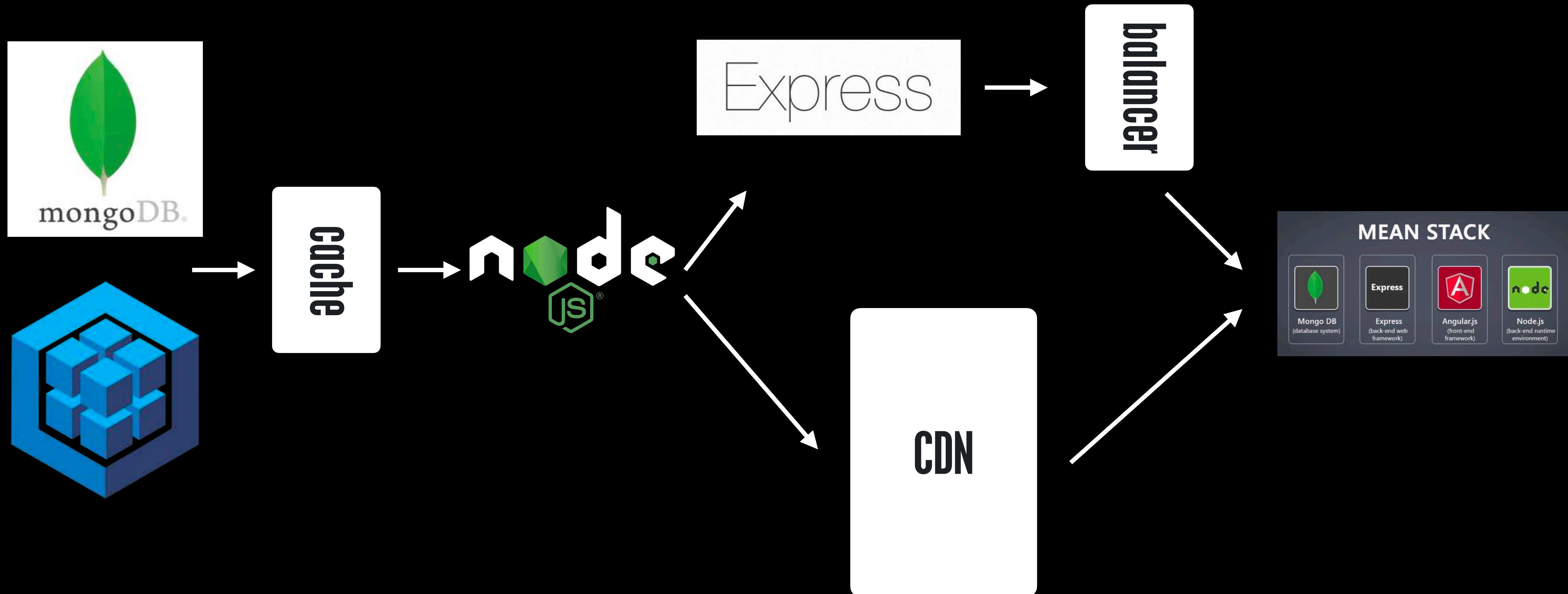


"Fullstack" developer.



1:38 AM - 24 Mar 2018 from Bucharest, Romania

- 1. IN THE BEGINNING: LAMP STACK**
- 2. LAMP PROBLEMS**
- 3. MEAN STACK**
- 4. MEAN PROBLEMS**



# PERFORMANCE!!!! JS SIZE VERSION

The New York Times

SUBSCRIBE NOW   LOG IN   ⚙

TECHNOLOGY

# For Impatient Web Users, an Eye Blink Is Just Too Long to Wait

By STEVE LOHR FEB. 29, 2012

f   t   e-mail   share   bookmark



Arvind Jain, a Google engineer, pointed out the loading speed of individual elements of a website on a test application used to check efficiency, at Google offices in Mountain View, Calif. Peter DaSilva for The New York Times

[« Web Components: The Long Game](#)

## What's All This Then?

I'm Alex Russell, a web developer working on Chrome, Blink, and the Web Platform at Google. I'm guilty of many JavaScript transgressions.

I help lead the team building a new application model for the web, and serve on ECMA TC39 (the standards body for JavaScript). I'm an elected member of the W3C Technical Architecture Group and am Tech Lead for Standards inside the Chrome team. I design and advocate for extensible, layered, data-driven evolution of the web platform.

My professional aim is to make the web a better platform and to the extent that I can keep politics and economics from creeping in, that's what this blog is about.

Other facets available upon HTTP request: twitter, facebook, flickr, linkedin, lanyrd, and quora. Expect less self-restraint there.

I'm proud husband to Frances Berriman. We live

# Can You Afford It?: Real-world Web Performance Budgets

*TL;DR: performance budgets are an essential but under-appreciated part of product success and team health. Most partners we work with are not aware of the real-world operating environment and make inappropriate technology choices as a result. We set a budget in time of <= 5 seconds first-load Time-to-Interactive and <= 2s for subsequent loads. We constrain ourselves to a real-world baseline device + network configuration to measure progress. The default global baseline is a ~\$200 Android device on a 400Kbps link with a 400ms round-trip-time ("RTT"). This translates into a budget of ~130-170KB of critical-path resources, depending on composition — the more JS you include, the smaller the bundle must be.*

We've had the pleasure of working with dozens of teams over the past few years. This work has been illuminating, sometimes in very unexpected ways. One of the most surprising results has been the frequent occurrence of "ambush by JavaScript":



Alex Russell  
@slightlylate



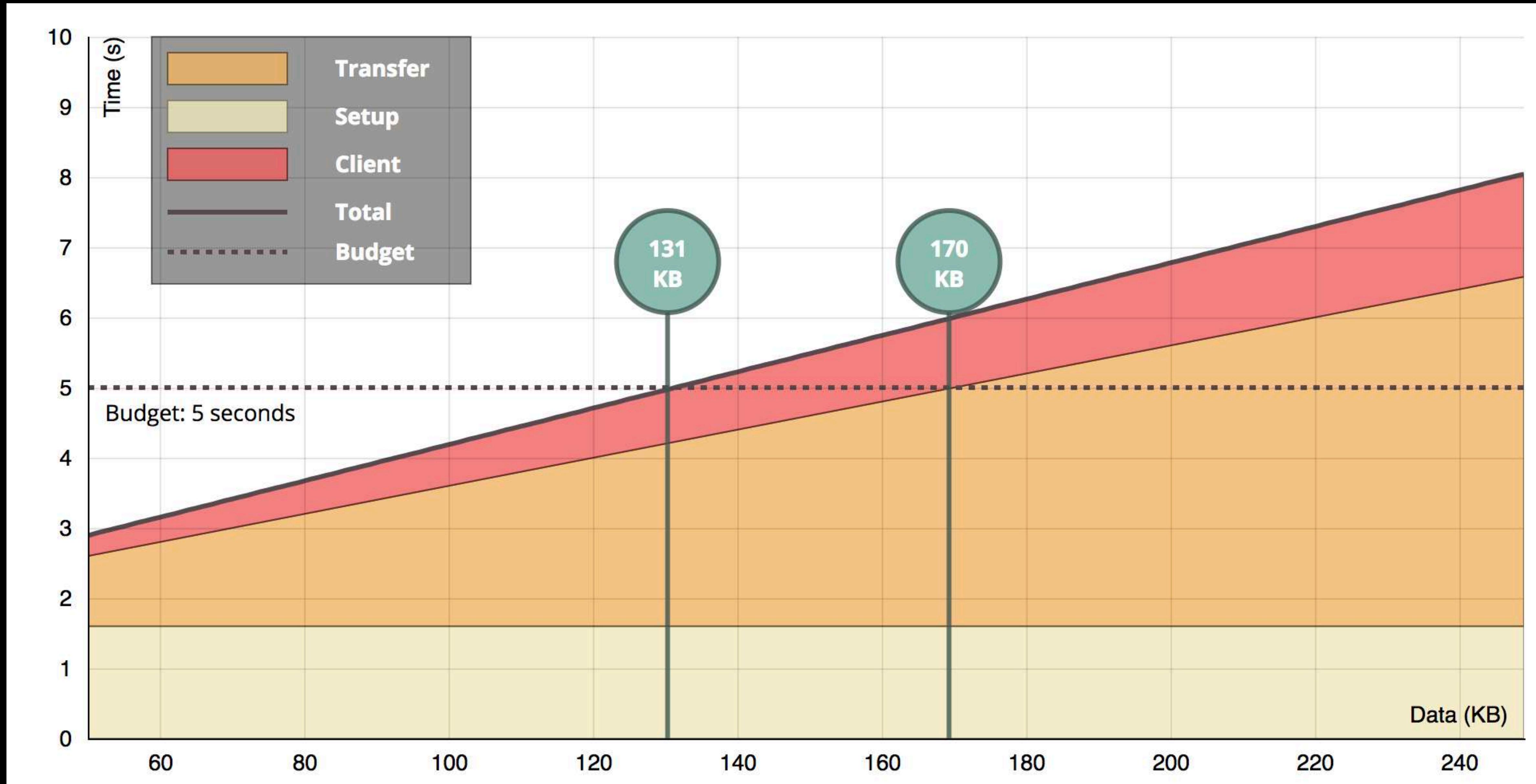
We need a new term for the business-opportunity wastage that modern front-end development has created.

Maybe "ambush by JS"?

11:40 AM - Oct 4, 2017

49 36 people are talking about this





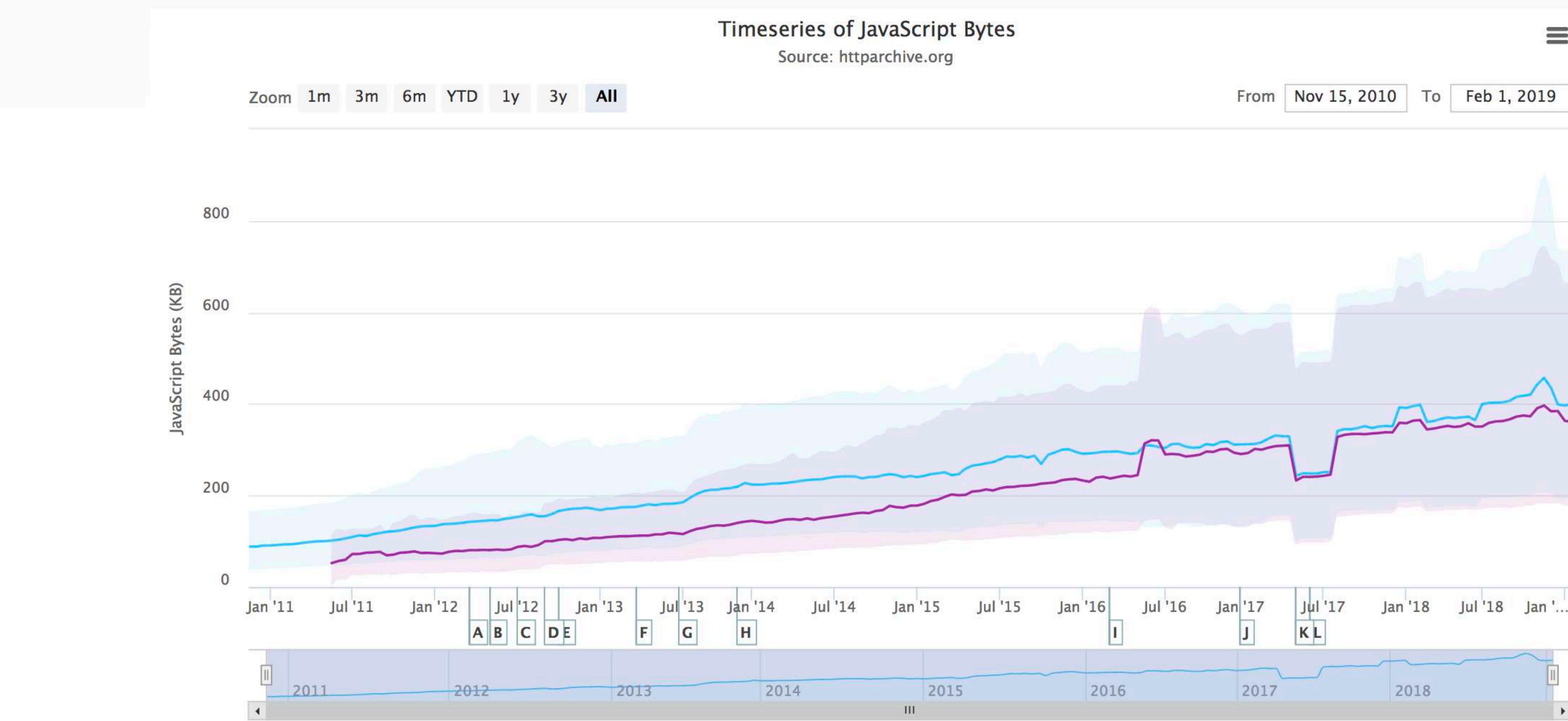
## JavaScript Bytes

The sum of transfer size kilobytes of all external scripts requested by the page. An external script is identified as a resource with the `js` or `json` file extensions or a MIME type containing `script` or `json`.

See also: [Page Weight](#)

MEDIAN DESKTOP  
**399.9 KB**  
▲351.4%

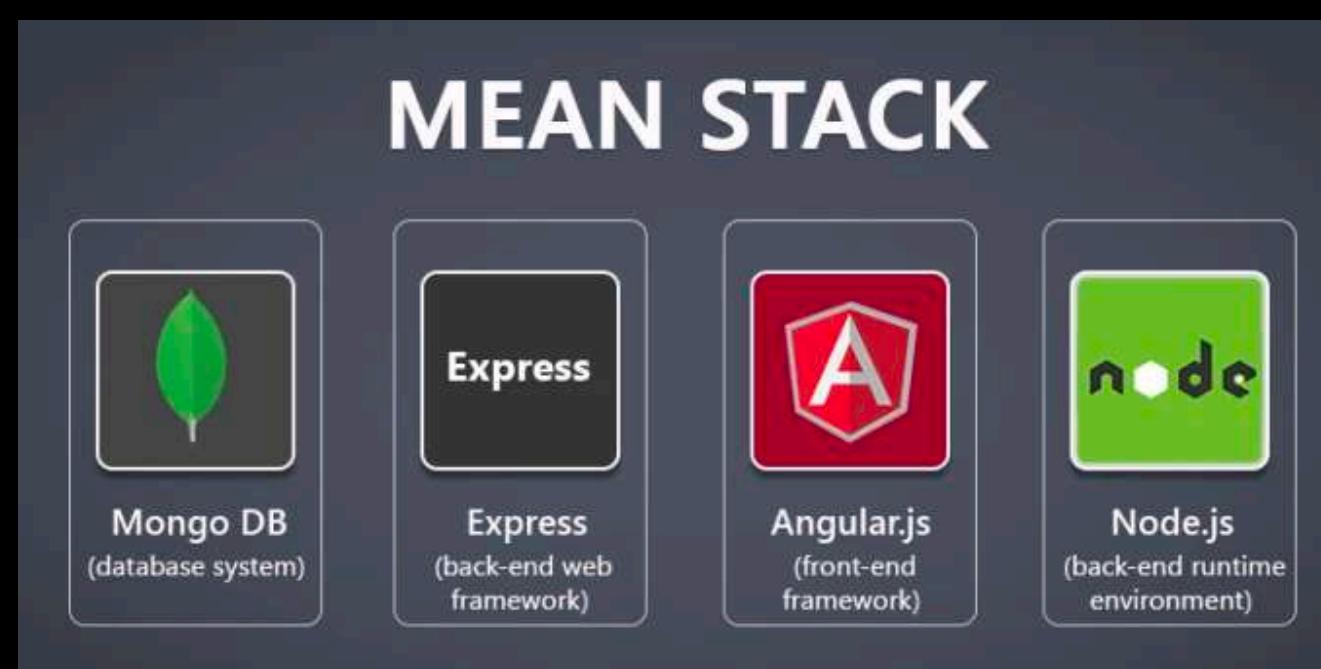
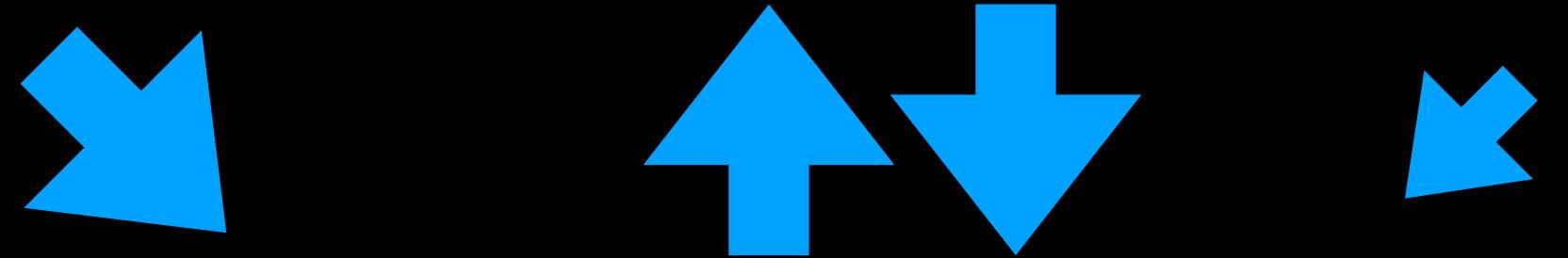
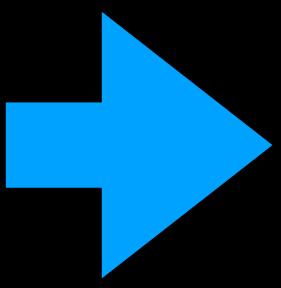
MEDIAN MOBILE  
**360.7 KB**  
▲588.4%



# CHEAT: CLIENTSIDE ROUTING

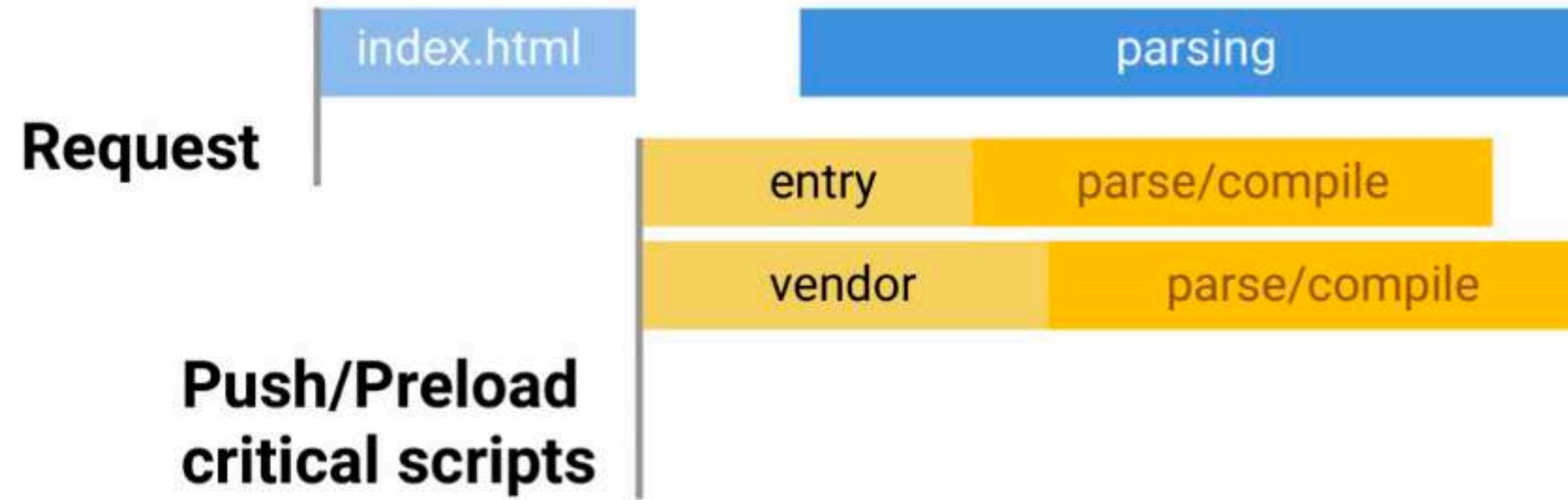


ES6 →



# PRPL Pattern

**Push the minimal code for the initial route**

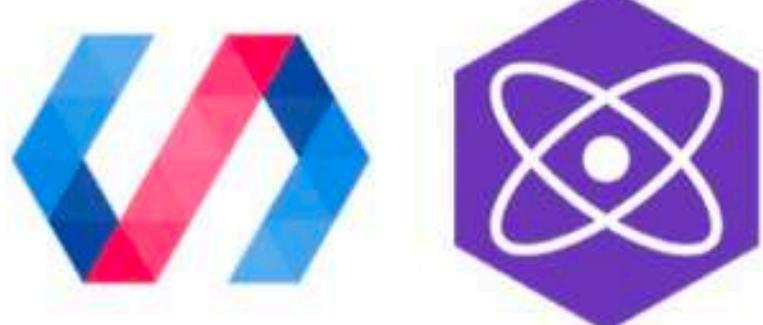


**Push/Preload critical scripts**

**Render route & get interactive**

**Navigate next route**

SUPPORTED BY CLIs



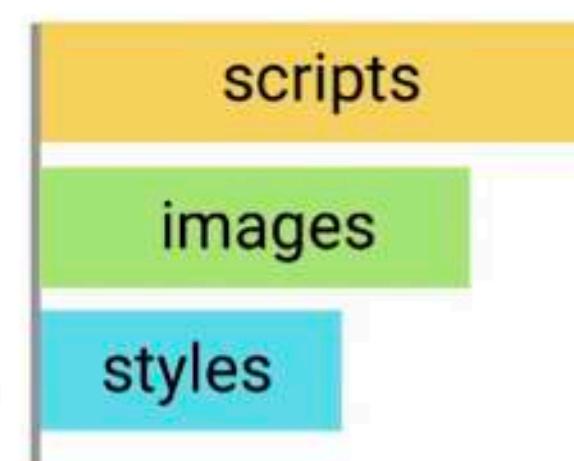
**Lazy-load async (split) routes**



**Navigate to another route**

**Pre-cache using Service Workers**

**Cache remaining resources**



USED BY SITES LIKE



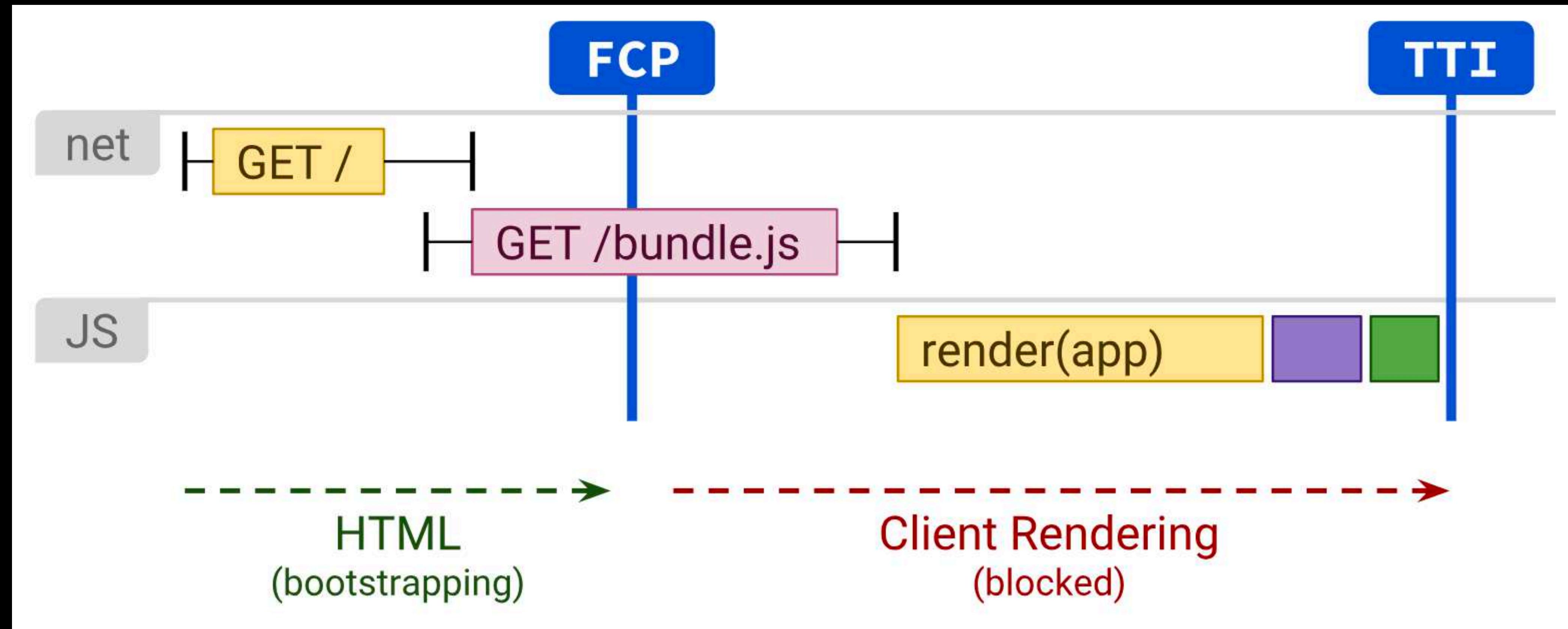
# PERFORMANCE!!!!!! TTI VERSION

<> index.html x

```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="utf-8" />
5      <meta
6        name="viewport"
7        content="width=device-width,initial-scale=1,shrink-to-fit=no"
8      />
9      <meta name="theme-color" content="#000000" />
10     <link rel="manifest" href="/manifest.json" />
11     <link rel="shortcut icon" href="/favicon.ico" />
12     <title>React App</title>
13     <link href="/static/css/main.1efe295c.chunk.css" rel="stylesheet" />
14   </head>
15   <body>
16     <noscript>You need to enable JavaScript to run this app.</noscript>
17     <div id="root"></div> ←
18 +   <script>...
103    ><script src="/static/js/1.5b265051.chunk.js"></script>
104    ><script src="/static/js/main.bf592139.chunk.js"></script>
105   </body>
106 </html>
107
```

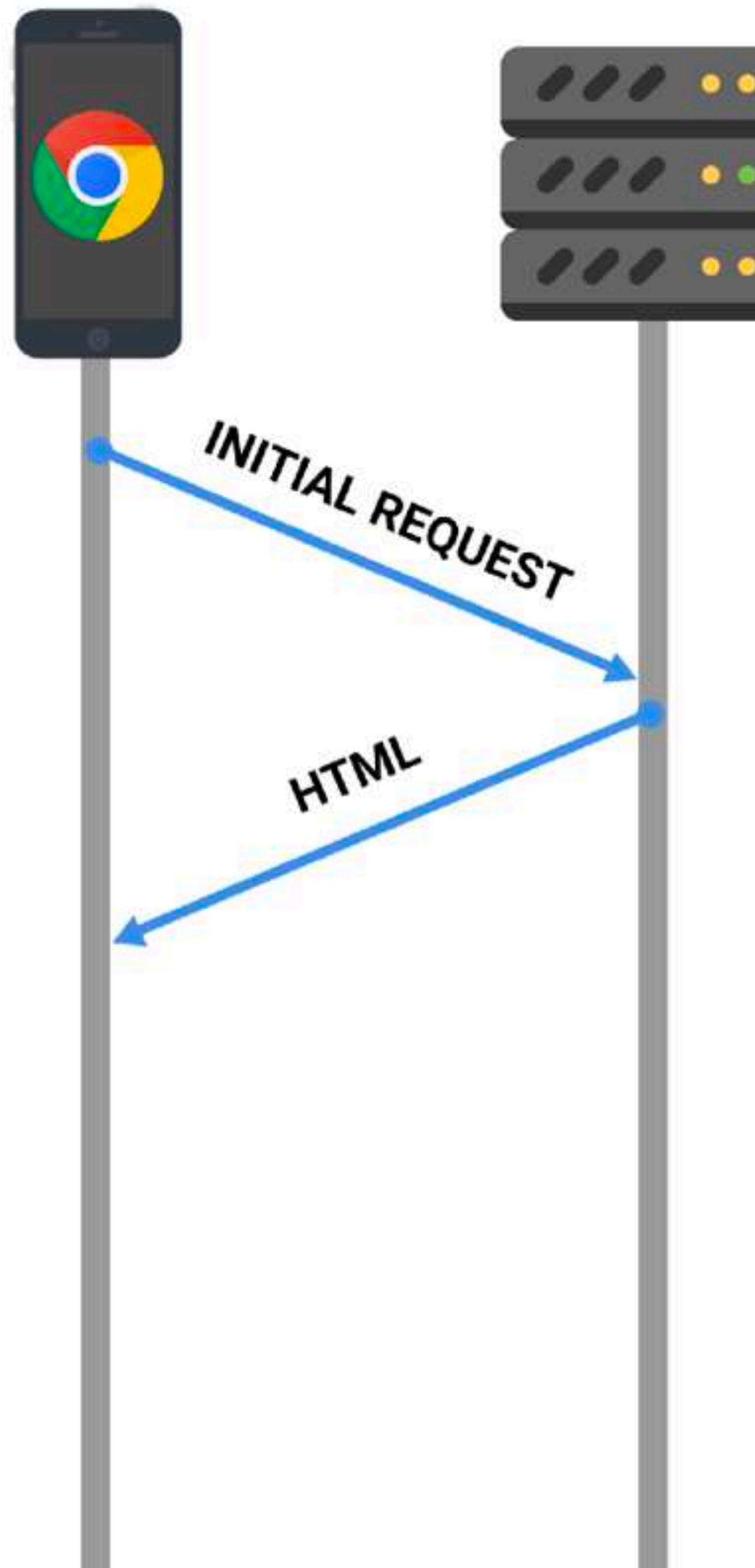
# SINGLE PAGE APPS

# CLIENT-SIDE RENDERING

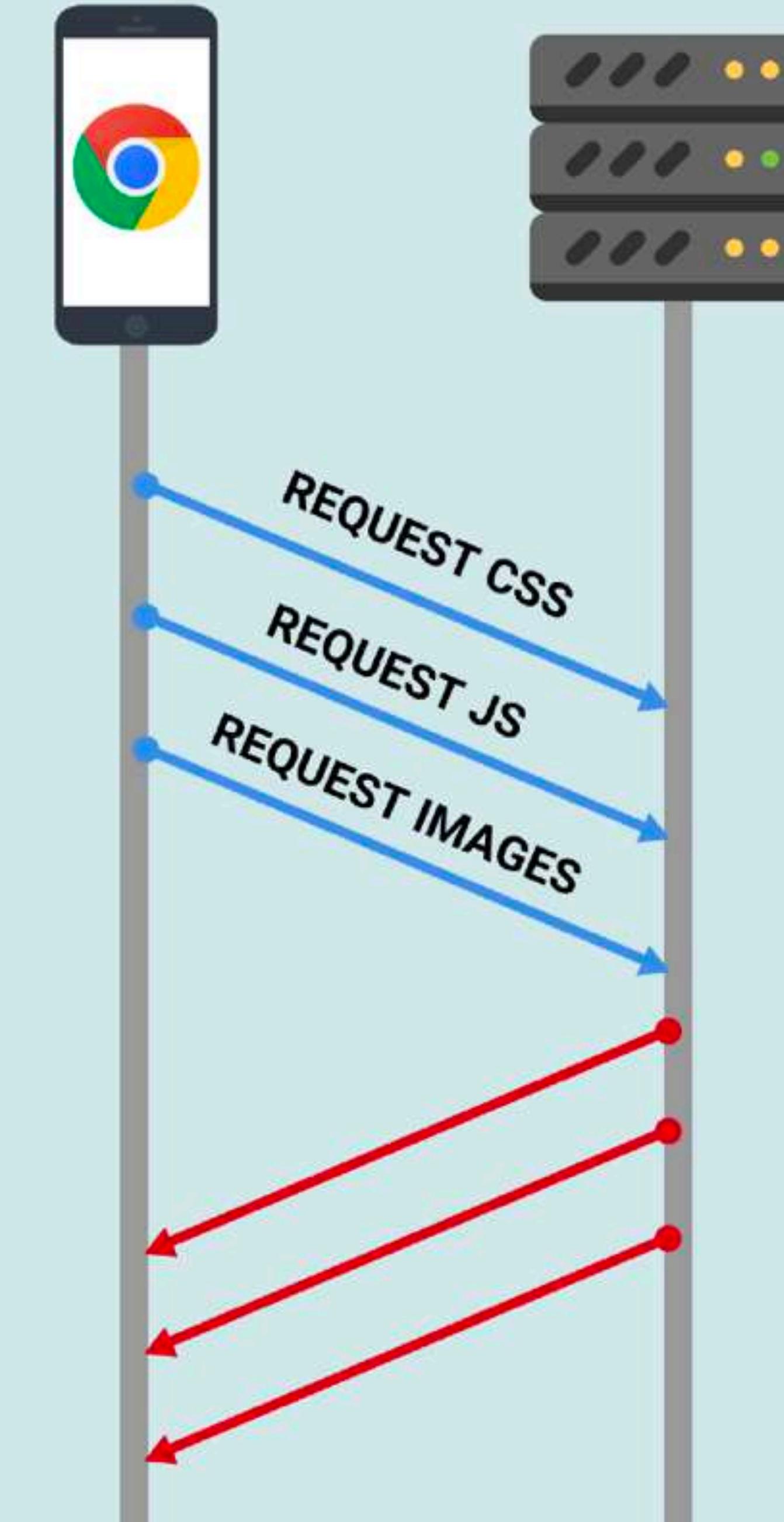


<https://developers.google.com/web/updates/2019/02/rendering-on-the-web>

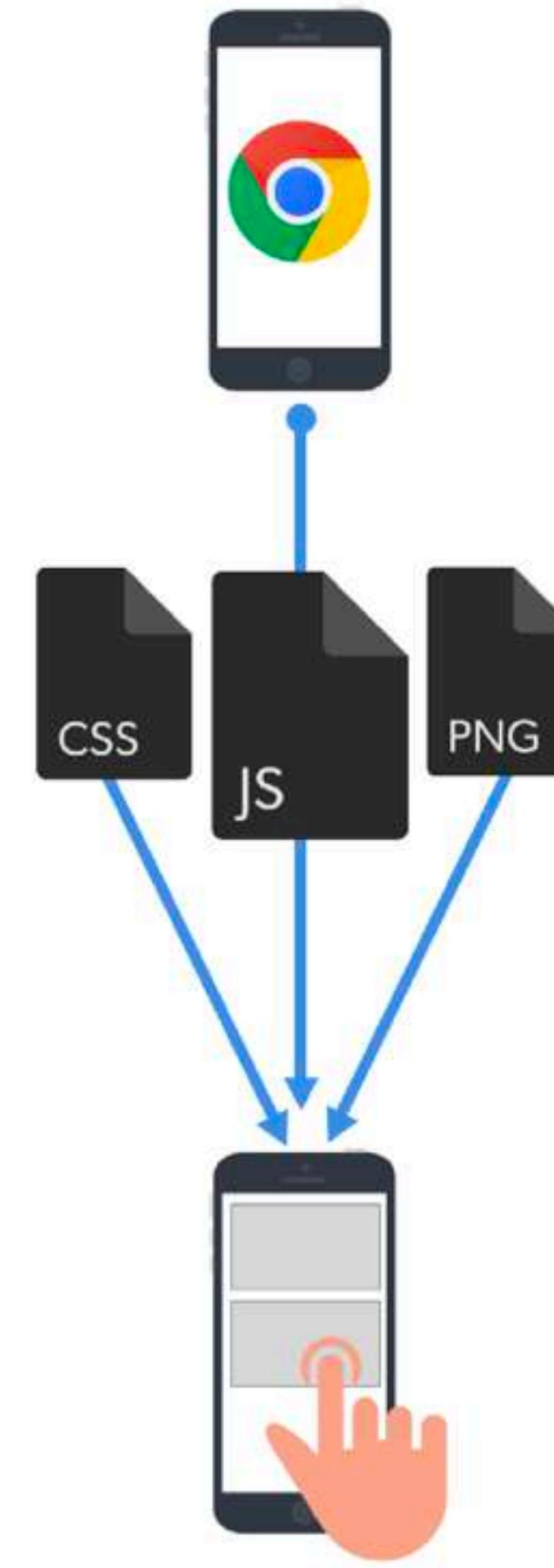
First request



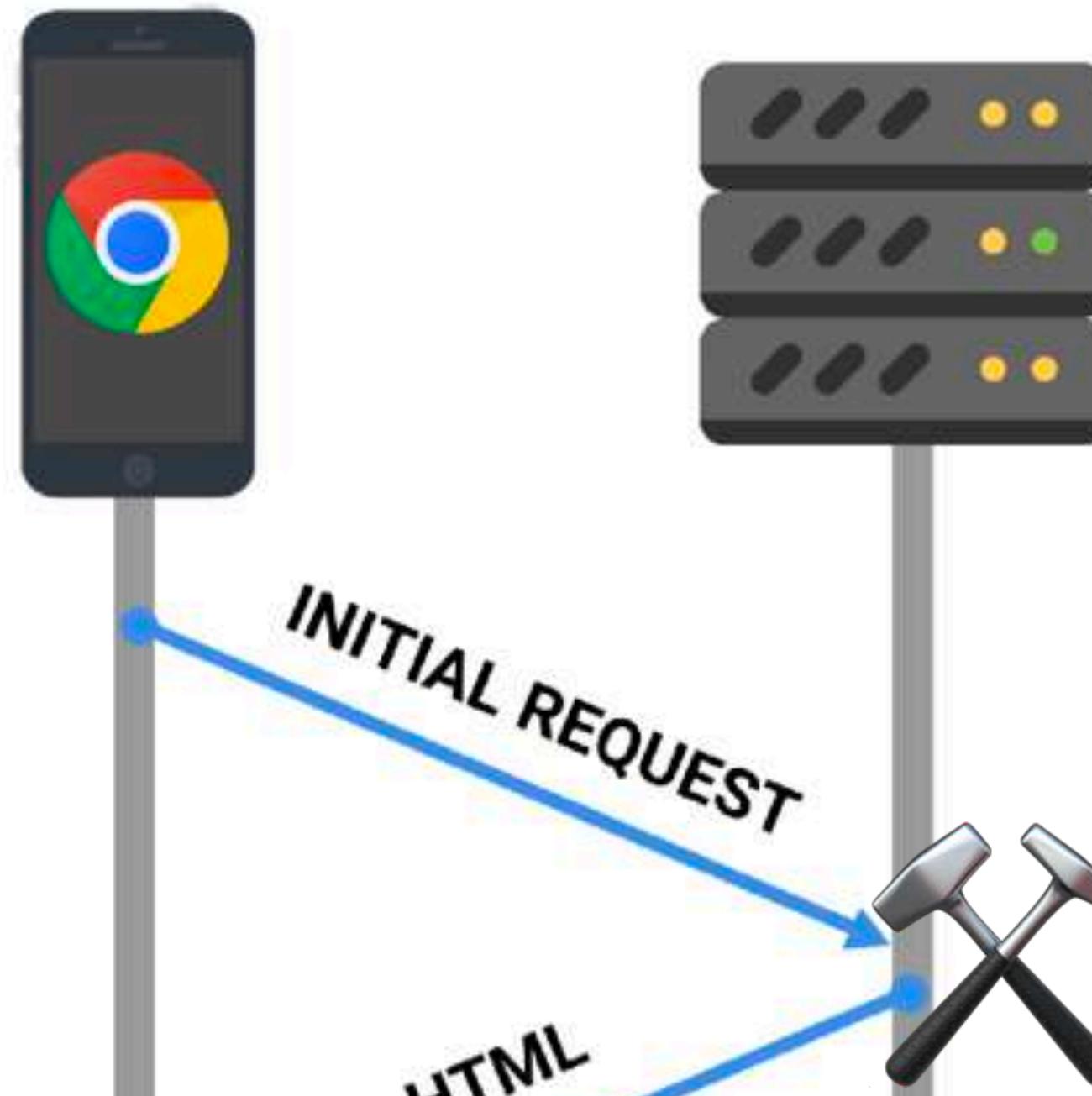
Fetch resources



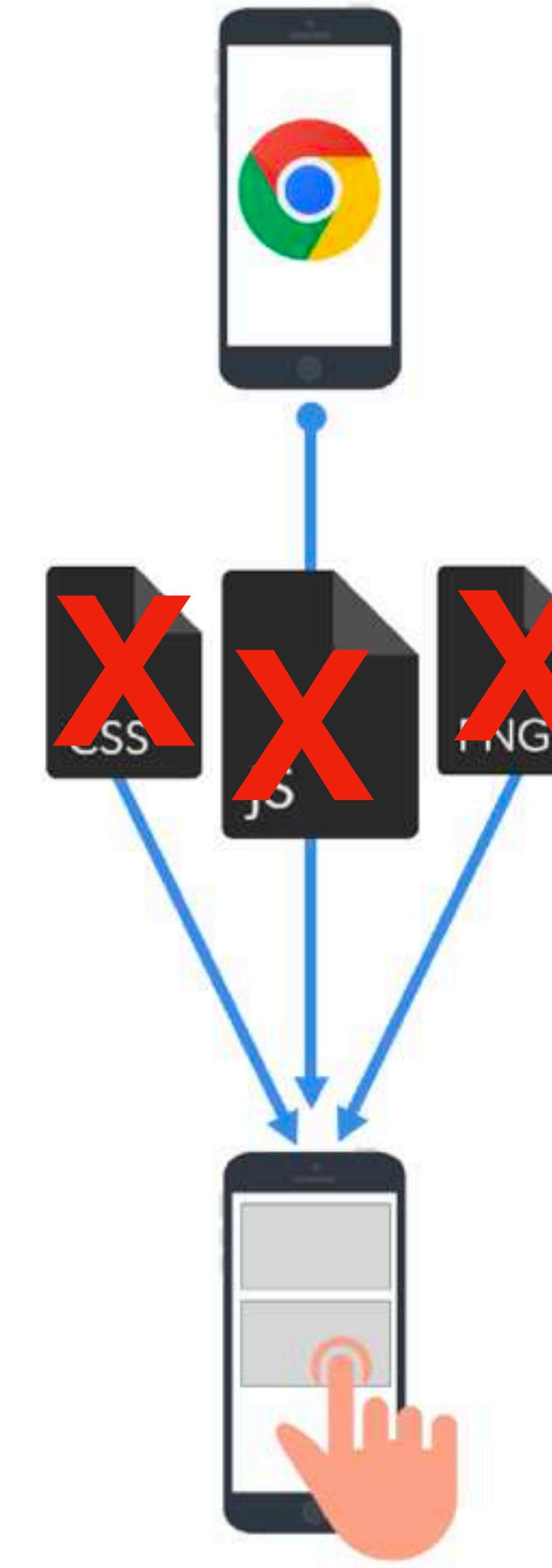
Decompress, Parse/Compile, Render



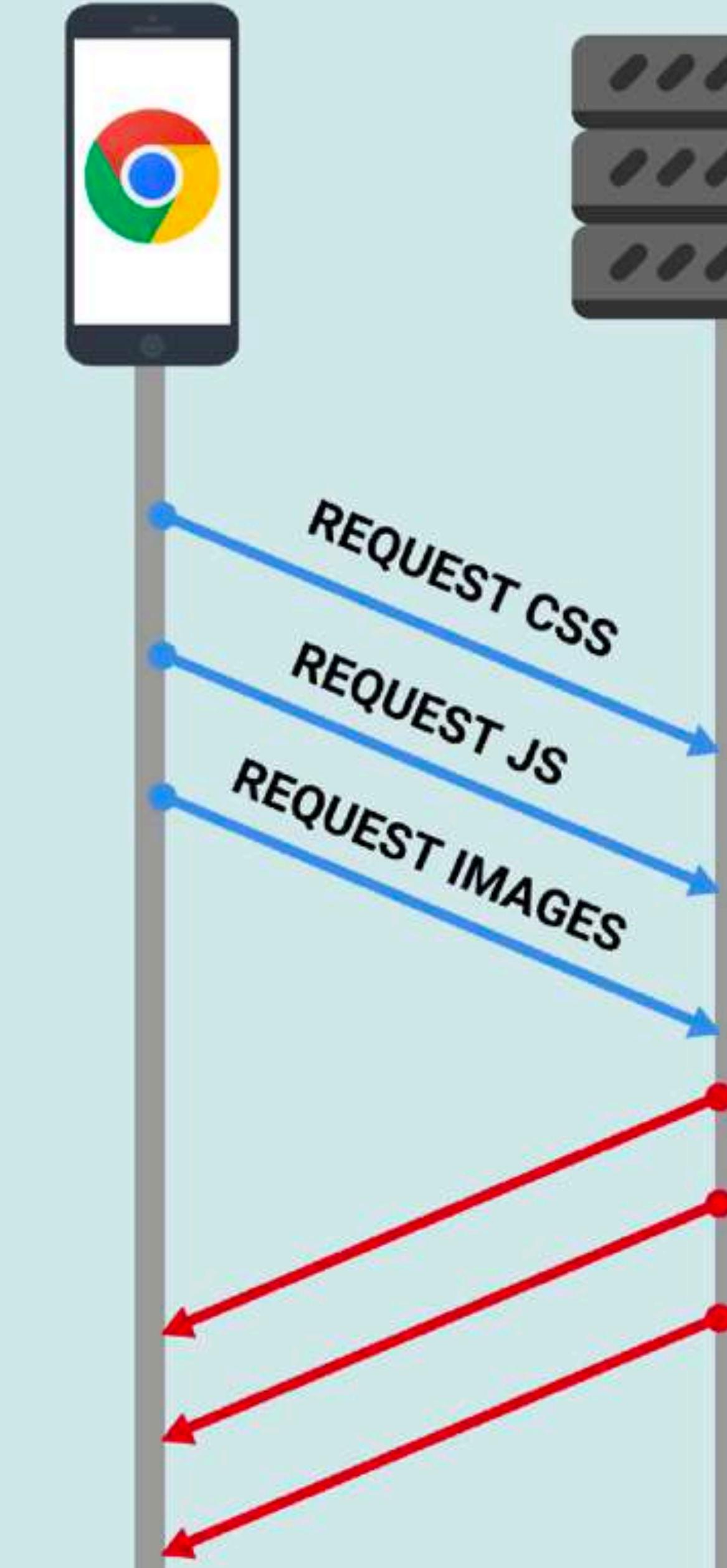
## First request

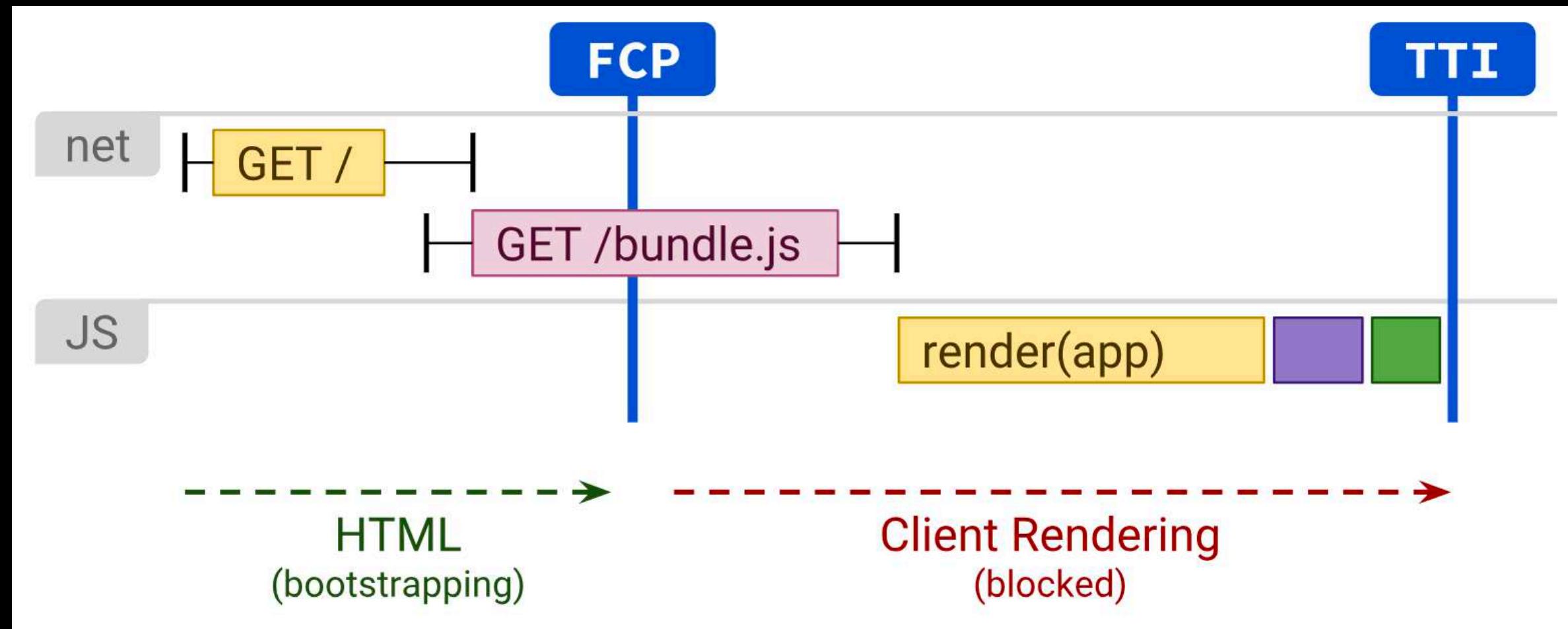


## Decompress, Parse/Compile, Render

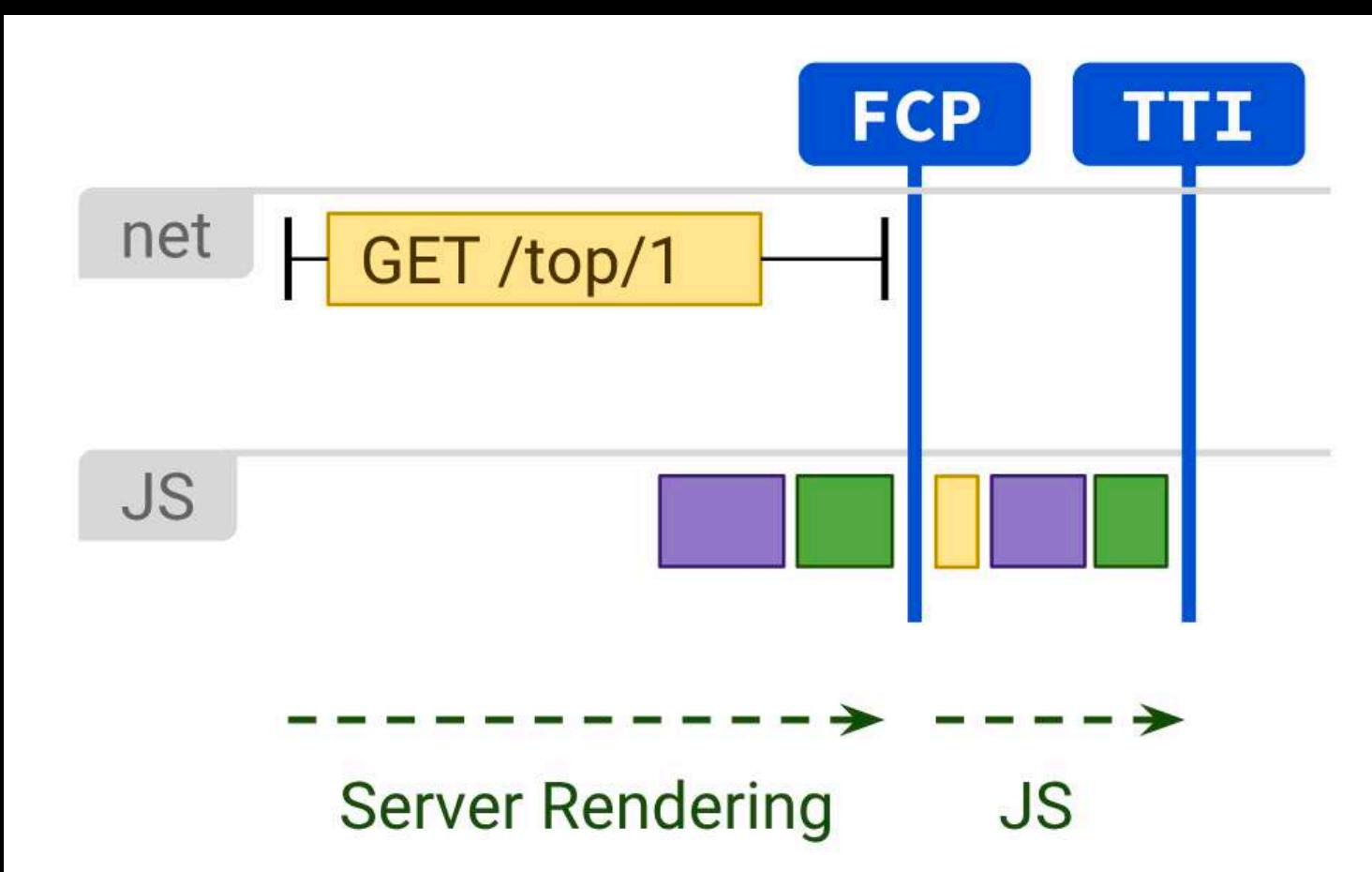


## Fetch resources





# CLIENT-SIDE RENDERING



# SERVER-SIDE RENDERING

<https://developers.google.com/web/updates/2019/02/rendering-on-the-web>

# MEAN PROBLEMS

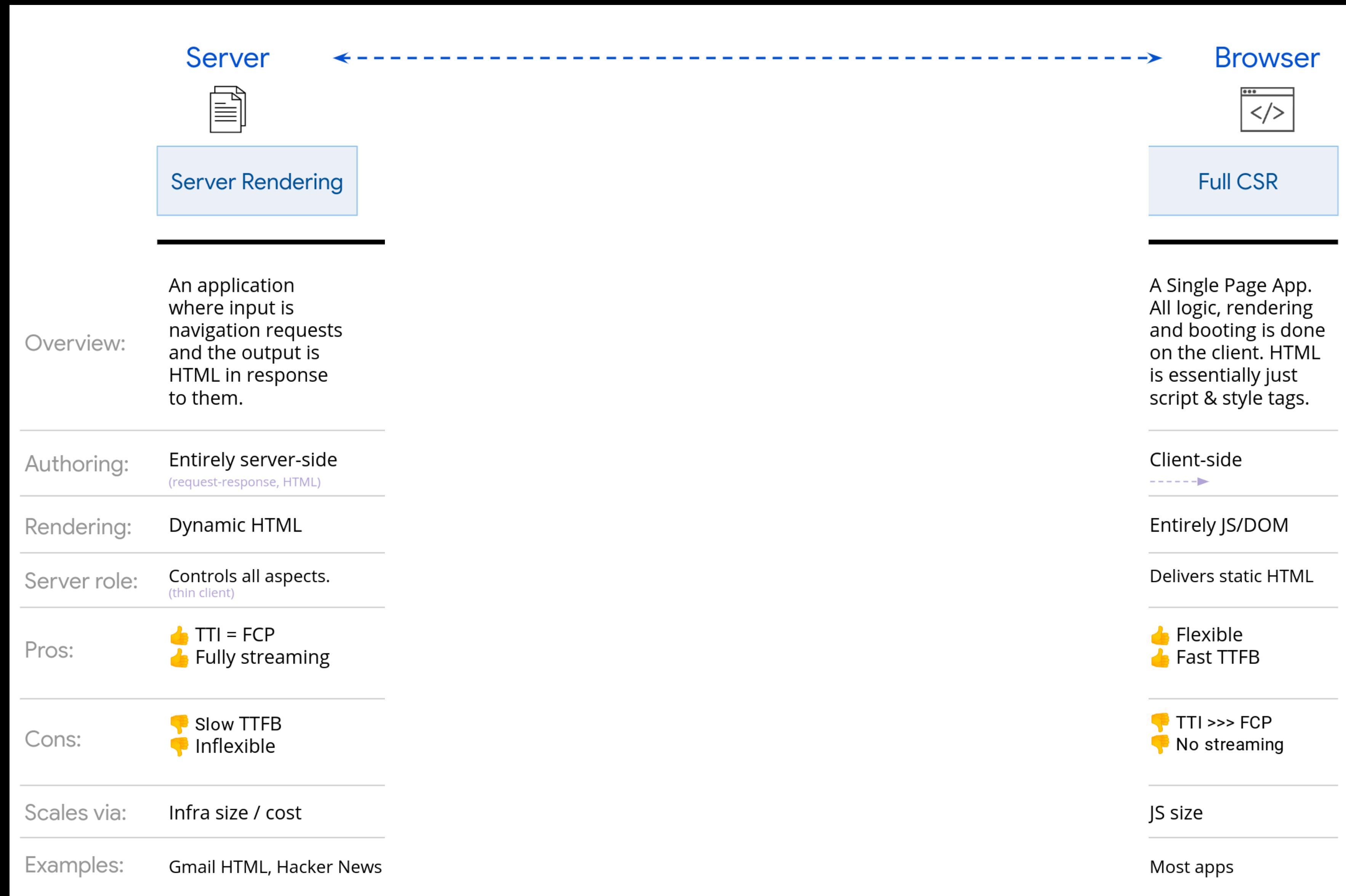
SECURITY

RELIABILITY

COMPLEXITY/COST

+

PERFORMANCE?



<https://developers.google.com/web/updates/2019/02/rendering-on-the-web>

- 1. IN THE BEGINNING: LAMP STACK**
- 2. LAMP PROBLEMS**
- 3. MEAN STACK**
- 4. MEAN PROBLEMS**
- 5. THE CHANGING ENVIRONMENT**

# GIT BASED, CI/CD WORKFLOW



# THE API ECONOMY

Mapping



Payments



Banking



Email



SSO



AI



Storage



Communication  
(Messaging/Voice)



Files



Background Check



Collaboration



Security/Fraud



IaaS



User Management



Identification



Social



Customer Engagement



Location



Healthcare



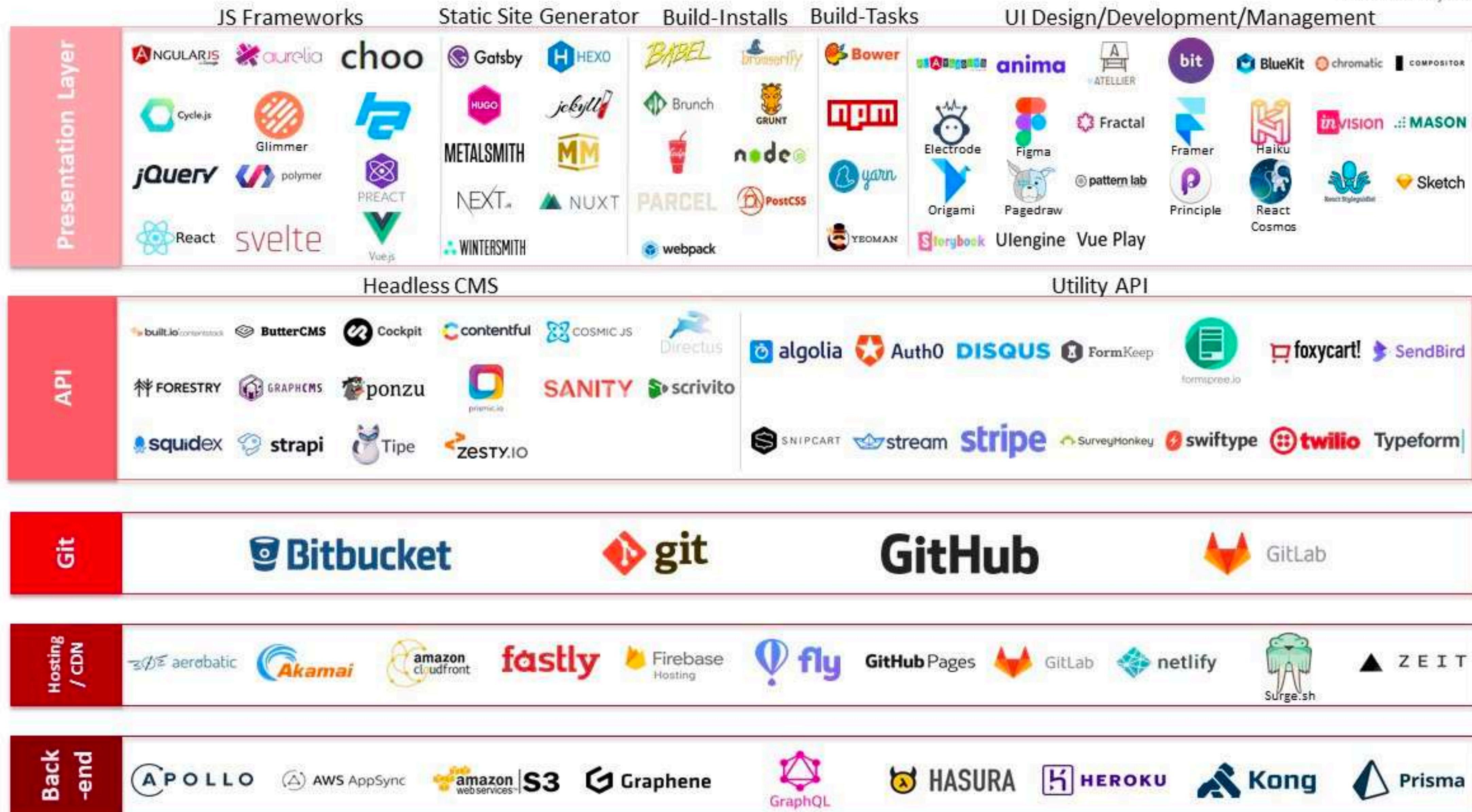
Enterprise Search



Test/Monitor



# JAMstack Ecosystem



# SERVERLESS

<https://techcrunch.com/2014/11/13/amazon-launches-lambda-an-event-driven-compute-service/>

## Amazon Launches Lambda, An Event-Driven Compute Service

Ron Miller 4 years



The image is a screenshot of a TechCrunch article. At the top, there's a navigation bar with a lock icon and the URL. Below it is the main title 'Amazon Launches Lambda, An Event-Driven Compute Service' in large, bold, black font. Underneath the title is the author's name 'Ron Miller' and the publication date '4 years'. The main content area features a dark background with white and yellow text. It says 'Introducing AWS Lambda' with a red 'new' badge above the 'o' in 'Lambda'. Below that, it describes it as 'An event-driven computing service for dynamic applications'. The bottom of the image shows a dark, blurred background of what appears to be a presentation slide.

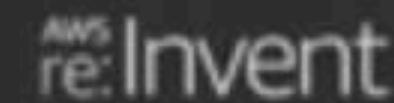
# SERVERLESS AT SCALE IS THE NEW NORM



THOMSON REUTERS  
processes **4,000 requests per second**



ingests, analyzes and stores **17+ petabytes of data** per season



processes **half a trillion validations of stock trades daily**



API traffic to register and license more than **47 million driver records** in Great Britain,



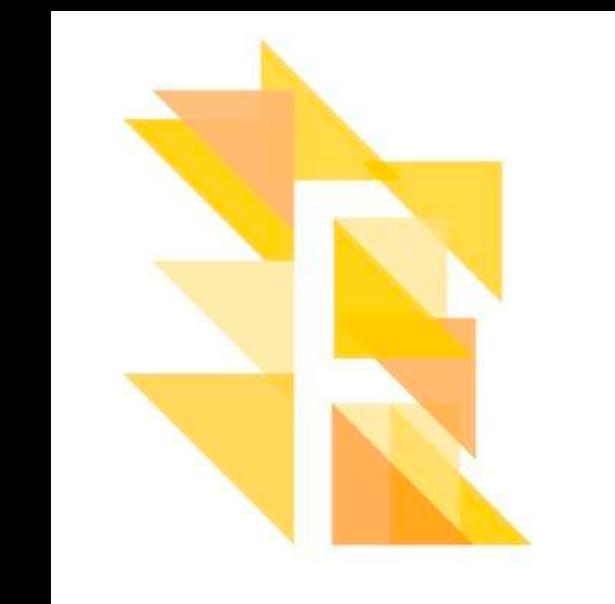
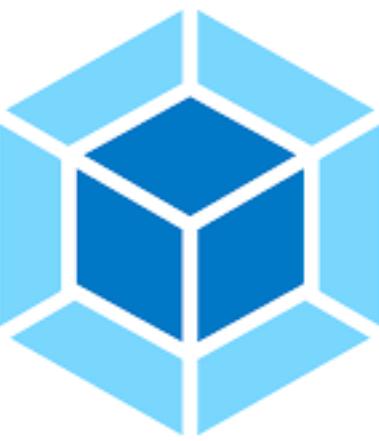
executes **16 million requests** a month



processes **tens of billions** of data points monthly



# BUILD TOOLS



# StaticGen

A List of Static Site Generators for JAMstack Sites



[About](#) [Contribute](#) [About JAMstack](#) [Need a Static CMS?](#)

## Filter

Any Language

Any Template

Any License

## Sort

GitHub stars

## Jekyll

★ 36973  
156  
8063  
6735  
+2784 +36 +528 +412

A simple, blog-aware, static site generator.

## Next

★ 35237  
235  
3949  
N/A  
+10213 -16 +1578

A framework for statically-exported React apps

## Hugo

★ 33052  
346  
3760  
6016  
+7597 +90 +666 +2374

A Fast and Flexible Static Site Generator.

**Get started with one click!**

For generators with the "Deploy to Netlify" button, you can deploy a new site from a template with one click. Get HTTPS, continuous delivery, and bring a custom domain, free of

*Nest and MailChimp now use static website generators for their primary websites. Vox Media has built a whole publishing system*



# STATIC SITE GENERATORS (v2)

JS (Templating, Components, Styling)  
Rehydration (SSR + SPA)





**Mark Dalgleish**  
@markdalgleish

Following



2:23 PM - 20 Mar 2019

2,038 Retweets 6,282 Likes

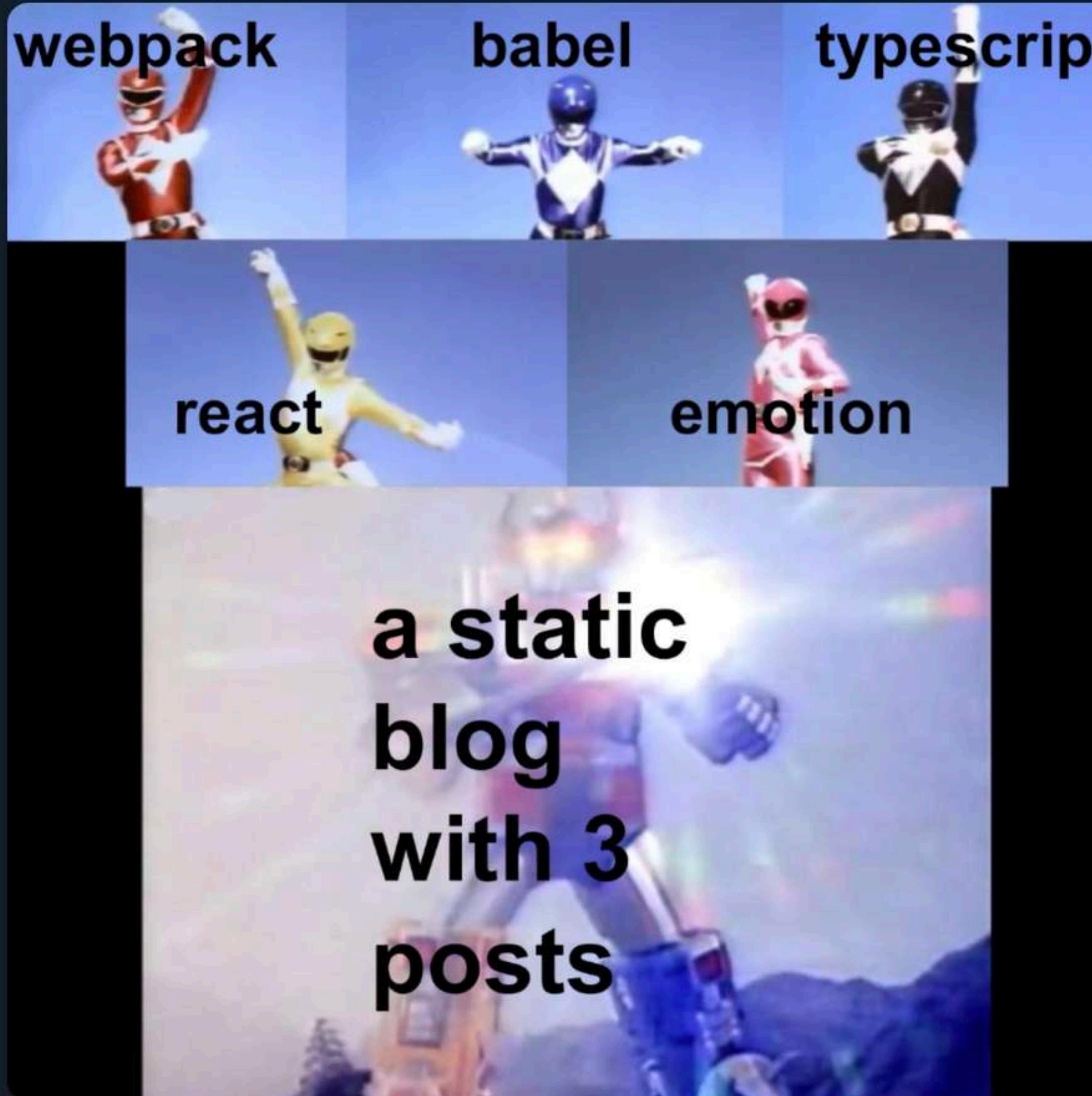




Mark Dalgleish  
@markdalgleish

Following

@swyx



2:23 PM - 20 Mar 2019

2,038 Retweets 6,282 Likes



Sean Larkin (廖肖恩) @TheLarkInn · Mar 20

Replying to @markdalgleish

Promised next @webpack features



1



10



112



I

1 more reply

HMM.... 

Git Workflow  
+ Build Tools  
+ Serverless  
+ API Economy  
+ Static Sites

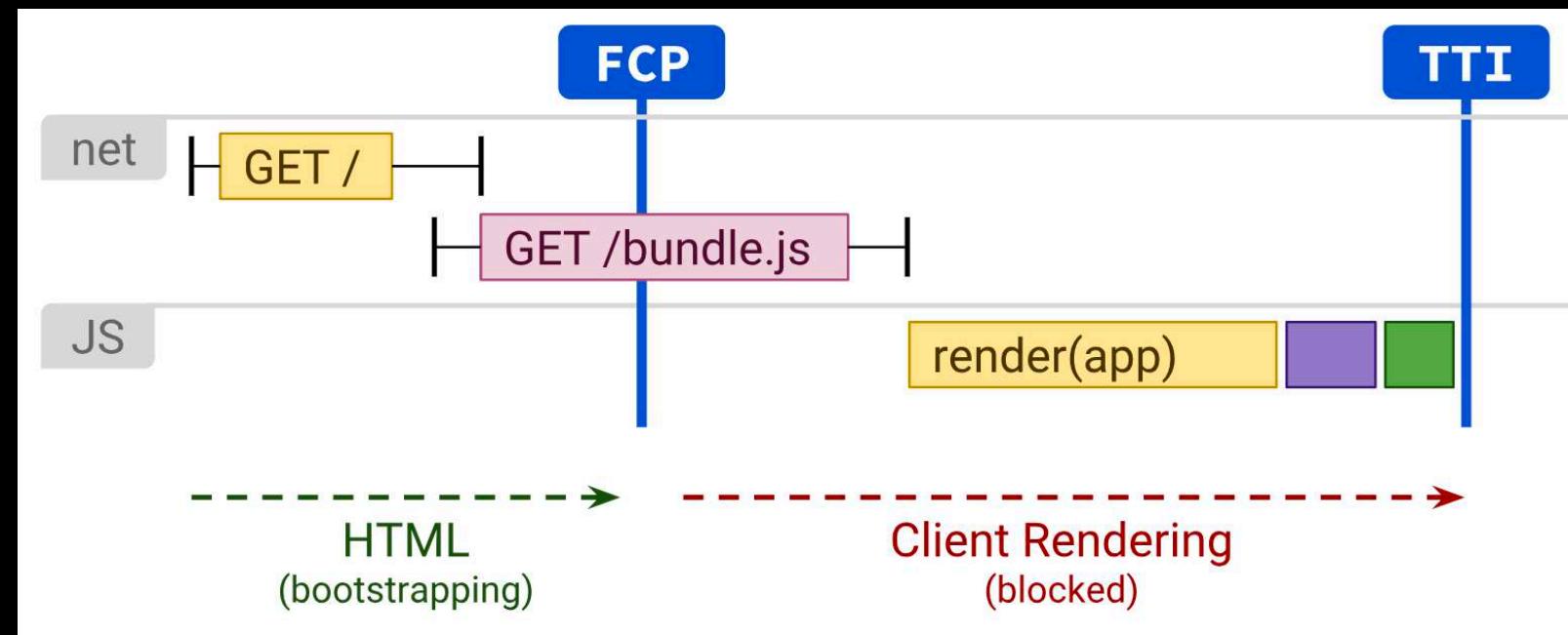
- 1. IN THE BEGINNING: LAMP STACK**
- 2. LAMP PROBLEMS**
- 3. MEAN STACK**
- 4. MEAN PROBLEMS**
- 5. THE CHANGING ENVIRONMENT**
- 6. JAM STACK**

# jamstack

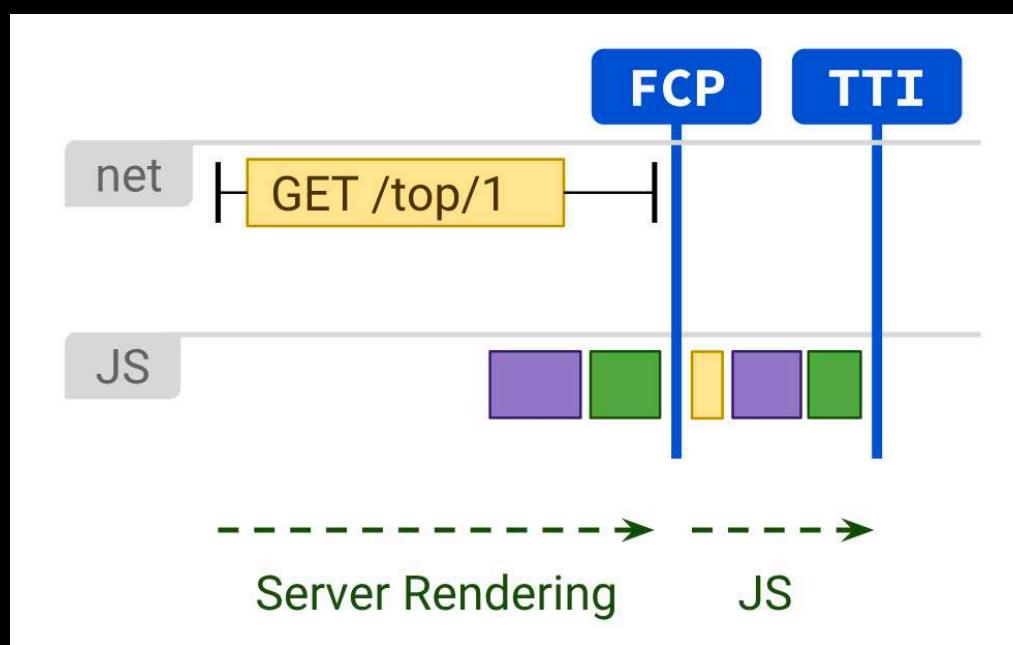
*A modern architecture —*

*Create fast and secure sites and dynamic apps  
with JavaScript, APIs, and pre-rendered  
Markup, served without web servers*

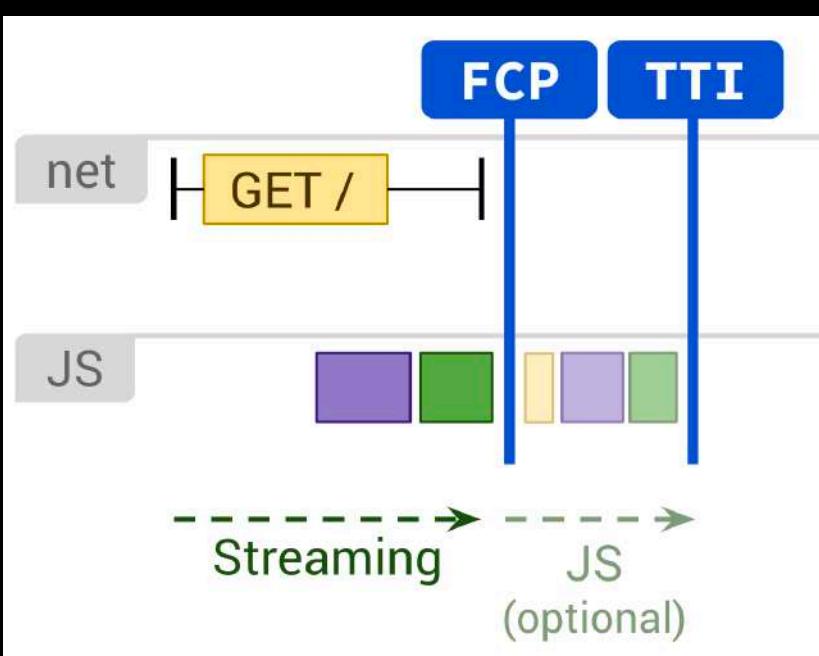
---



# CLIENT-SIDE RENDERING



# SERVER-SIDE RENDERING



# STATIC RENDERING

<https://developers.google.com/web/updates/2019/02/rendering-on-the-web>

Server		Browser			
					
	Server Rendering	“Static SSR”	SSR with (Re)hydration	CSR with Prerendering	Full CSR
Overview:	An application where input is navigation requests and the output is HTML in response to them.	Built as a Single Page App, but all pages prerendered to static HTML as a build step, and the JS is <b>removed</b> .	Built as a Single Page App. The server prerenders pages, but the full app is also booted on the client.	A Single Page App, where the initial shell/skeleton is prerendered to static HTML at build time.	A Single Page App. All logic, rendering and booting is done on the client. HTML is essentially just script & style tags.
Authoring:	Entirely server-side <small>(request-response, HTML)</small>	Built as if client-side <small>(components, DOM*, fetch)</small>	Built as client-side	Client-side	Client-side
Rendering:	Dynamic HTML	Static HTML	Dynamic HTML <b>and</b> JS/DOM	Partial static HTML, then JS/DOM	Entirely JS/DOM
Server role:	Controls all aspects. <small>(thin client)</small>	Delivers static HTML	Renders pages <small>(navigation requests)</small>	Delivers static HTML	Delivers static HTML
Pros:	<ul style="list-style-type: none"> <li>👍 TTI = FCP</li> <li>👍 Fully streaming</li> </ul>	<ul style="list-style-type: none"> <li>👍 Fast TTFB</li> <li>👍 TTI = FCP</li> <li>👍 Fully streaming</li> </ul>	<ul style="list-style-type: none"> <li>👍 Flexible</li> </ul>	<ul style="list-style-type: none"> <li>👍 Flexible</li> <li>👍 Fast TTFB</li> </ul>	<ul style="list-style-type: none"> <li>👍 Flexible</li> <li>👍 Fast TTFB</li> </ul>
Cons:	<ul style="list-style-type: none"> <li>👎 Slow TTFB</li> <li>👎 Inflexible</li> </ul>	<ul style="list-style-type: none"> <li>👎 Inflexible</li> <li>👎 Leads to hydration</li> </ul>	<ul style="list-style-type: none"> <li>👎 Slow TTFB</li> <li>👎 TTI &gt;&gt;&gt; FCP</li> <li>👎 Usually buffered</li> </ul>	<ul style="list-style-type: none"> <li>👎 TTI &gt; FCP</li> <li>👎 Limited streaming</li> </ul>	<ul style="list-style-type: none"> <li>👎 TTI &gt;&gt;&gt; FCP</li> <li>👎 No streaming</li> </ul>
Scales via:	Infra size / cost	build/deploy size	Infra size & JS size	JS size	JS size
Examples:	Gmail HTML, Hacker News	Docusaurus, Netflix*	<a href="#">Next.js</a> , <a href="#">Razzle</a> , etc	Gatsby, Vuepress, etc	Most apps

<https://developers.google.com/web/updates/2019/02/rendering-on-the-web>

# THESE ARE JAMSTACK

GitHub Pages

S3 Bucket / Azure / GCP

Zeit Now v2

Netlify

# THESE ARE JAMSTACK

The Progressive JavaScript Framework

WHY VUE.JS? GET STARTED GITHUB

Special Sponsor Standard Library

Approachable Versatile Performant

We help the daring build legendary companies.

SMASHING MAGAZINE

Articles Books Events Jobs Membership

Don't Miss These Latest Articles

Noemi Stauffer wrote 7 Gorgeous Free And Open-Source Typefaces And When To Use Them

Daniel Ni wrote The Ultimate Guide To Building Scalable Web Scrapers With Scrapy

Introducing a brilliantly simple way to work

Discover the Citrix Workspace intelligent experience

Start working smarter

Our Services Your Industries Trends & Insights Our Customers About Us 888.874.6388 Login Lead

WORKFORCE ANALYTICS How SMBs Can Harness the Power of Their HR... INCREDIBLE NOW Why Workplace Culture is Having a Moment PEOPLE MATTER What's this plant-based food co's key ingredient?

HR SOLUTIONS FOR THE PEOPLE WHO MATTER

Chat with TriNet

PayPal About PayPal.Me Terms My PayPal.Me

Help others know it's you they're paying

Create your link, share it, and start receiving money. Split a bill with friends or accept payments from customers. Anyone can use it.

Create Your PayPal.Me Link



## shawn wang's team ▾

Sites Domains Members Audit log Team settings

Search site



**nervous-morse-5627ad**

Deploys from [GitHub](#)

Owned by shawn wang's team

Last published on Dec 20, 2017 (2 years ago)



shawn wang

User settings

Sign out



**infallible-morse-1a2b8a**

Deploys from [GitHub](#)

Owned by shawn wang's team

Last published on Jul 9 (8 days ago)



**stupefied-lalande-9c1670**

Deploys from [GitHub](#)

Owned by shawn wang's team

Last published on Jul 3 (14 days ago)



**wonderful-goldwasser-65ff62**

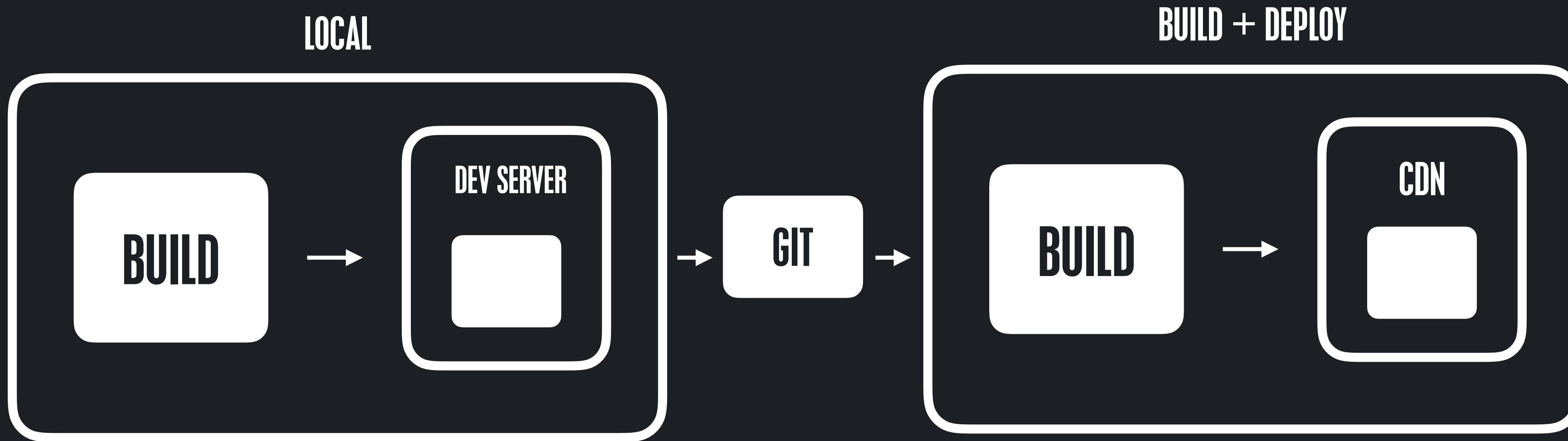
Deploys from [GitHub](#)

Owned by shawn wang's team

Last published on Jul 3 (14 days ago)



# DEVELOPER EXPERIENCE





Instant roll backs  
Deploy previews  
Branch deploys  
Functions  
Forms



Instant roll backs  
Deploy previews  
Branch deploys  
Functions  
Forms

Identity  
Analytics  
Redirects  
A/B Testing  
Managed DNS

- 1. IN THE BEGINNING: LAMP STACK**
- 2. LAMP PROBLEMS**
- 3. MEAN STACK**
- 4. MEAN PROBLEMS**
- 5. THE CHANGING ENVIRONMENT**
- 6. JAM STACK**
- 7. RECAP**

# THE OPEN WEB

## Apps Account for Over 80% of Mobile Time Share of Total Mobile Minutes by Browser / App



When considering mobile (smartphone and tablet) minutes in isolation, they are overwhelmingly dominated by app consumption – over 80% of all mobile time in the markets considered for this report.

Levels appear marginally higher in Latin America, with Argentina, Brazil and Mexico all falling within the top 5 of the markets shown here.





# WANT MORE?

**JAMstack\_conf**

JAMstack is growing, and this year we will be in two brand new locations. New York City, and London—both together with our flagship conference in San Francisco.

New York      April 9-10

London      July 9-10

San Francisco      October 16-18



**QCon LONDON by InfoQ**  
CELEBRATING 14 YEARS  
Conference: March 2-4, 2020  
Workshops: March 5-6, 2020

**Presentation: Bringing JAMStack to the Enterprise**

Track: JavaScript: Powering the Modern Web  
Location: Whittle, 3rd flr.  
Duration: 1:40pm - 2:30pm  
Day of week: Wednesday

More talks on:  
[JavaScript](#) [Interview Available](#)

Share this on:

**What You'll Learn**

- Find out what is JAMStack, what are the benefits of using it.
- Learn about static web apps, the benefits in performance, security, deployment, etc.
- Hear about some of the challenges PayPal has faced deploying static apps on their own infrastructure.

**netlifycommunity**

all categories ▾ all tags ▾ all ▾ Latest New (17) Unread (4) Top My Posts Bookmarks Categories + New Topic

Topic

Netlify Identity causing unexplained layout bugs in React components  
Access Control / Identity

Form not showing in dashboard, duplicate form names? Netlify Forms

Functions - Failed to upload file Deploying and Building

🔒 [Common Issue] How to debug your form Netlify Forms

Not receiving email on GoDaddy email address DNS

Combining SPA redirect with Language condition Redirects

**Showcase**

**tND** the New Dynamic build better faster websites

- Database
- Events
- Articles
- Videos
- Community
- Showcase
- Find an Expert
- Jobs

**Coffee Collective** Gatsby Snipcart

**Datica.com** Middleman Contentful

**Prima** Gatsby

**Buoyant** Hugo Forestry

**Storybook** Gatsby

**Algolia** Gatsby Algolia

**Repower Australia** Jekyll Forestry

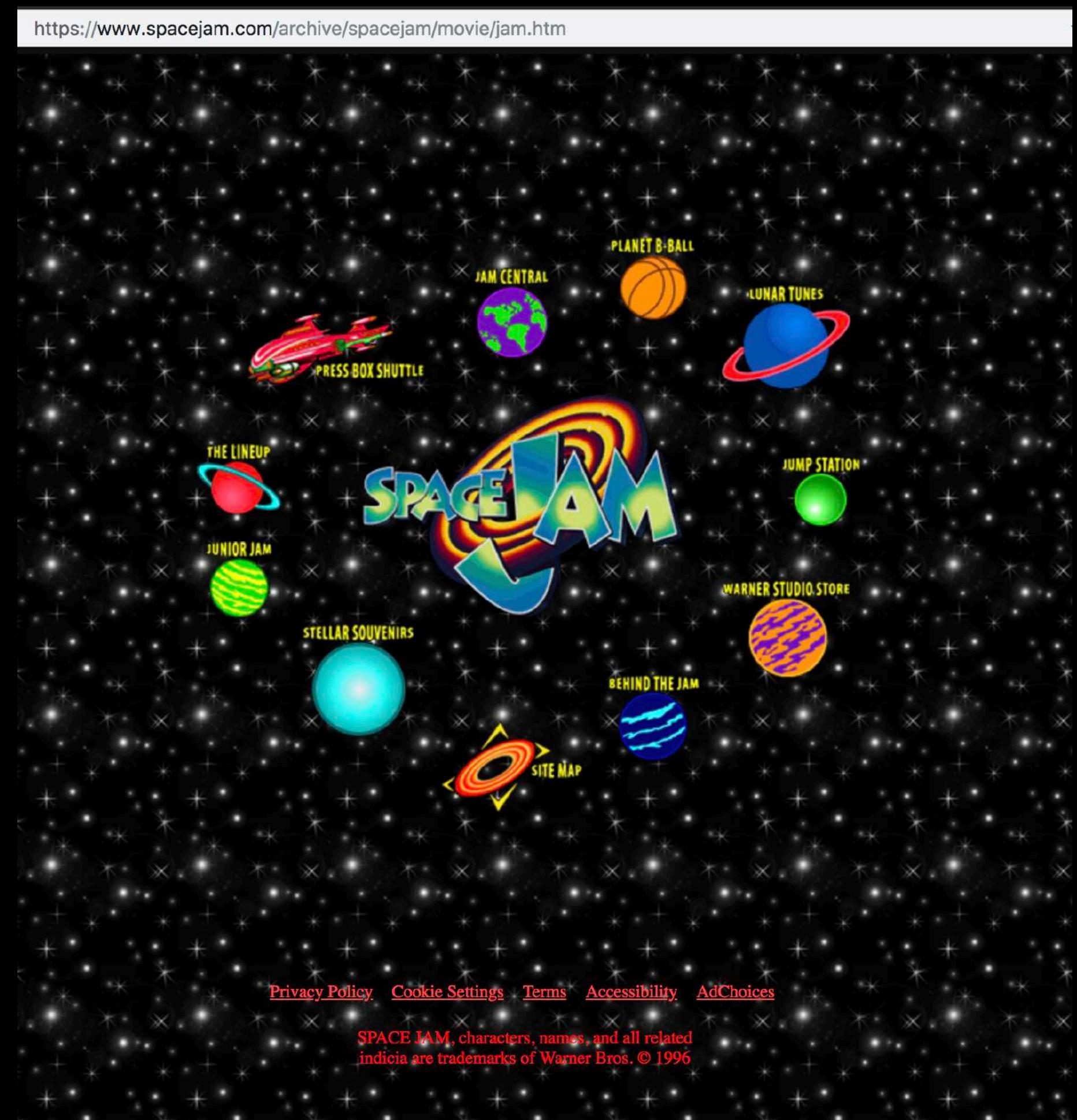
**Howard Schultz** Gatsby Netlify

**vimeo** Join Log in Pricing Features Watch Stock New

**MATHIAS BIILMANN**  
THE NEW FRONT-END STACK: JAVASCRIPT, APIs AND MARKUP

**S SMASHING CONFERENCE SAN FRANCISCO**

# THE UNBREAKABLE WEB



# THANKS JSCAMP! QUESTIONS WELCOME



[SWYX.IO/TALKS](https://swyx.io/talks) | [SWYX@NETLIFY.COM](mailto:swyx@netlify.com) | [@SWYX](https://twitter.com/swyx)