

App Dev

Stefano Balietti

Center for European Social Science Research at Mannheim University (MZES)
Alfred-Weber Institute of Economics at Heidelberg University

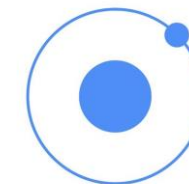
@balietti | stefanobalietti.com | @nodegameorg | nodegame.org



Building Digital Skills: 20.5 May 2022, University of Luzern



Express



Single-Page Application (SPA) Frameworks

A fairly recent trend involves web-applications that are fully loaded in the browser on the first page-load, and then every operation is carried out via AJAX or similar fetch requests.

These applications are called [Single-Page Application \(SPA\)](#)

Implementing one can be quite complex, therefore many frameworks have emerged

Single-Page Application (SPA) Frameworks

A fairly recent trend involves web-applications that are fully loaded in the browser on the first page-load, and then every operation is carried out via AJAX or similar fetch requests.

These applications are called [Single-Page Application \(SPA\)](#)

Implementing one can be quite complex, therefore many frameworks have emerged

Among the [most popular](#) are:

- **Angular**
- **React**
- **Vue**

Angular



<https://angular.io/>

[Angular](#) is a framework originally implemented by Google

Follows Model-View-Controller ([MVC](#)) pattern (now [MVVM](#)).

Amongst the first SPA frameworks implemented

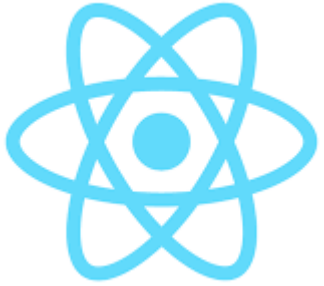
In decline?

The update from Angular 1 (AngularJS) to Angular 2 left many developers unsatisfied.

Now in **TypeScript**

Hard learning curve, appropriate for large projects with several developers.

React



<https://reactjs.org/>

```
ReactDOM.render(  
  <h1>Hello, world!</h1>,  
  document.getElementById('root')  
);
```

Originally created at Facebook

One of the first frameworks to make use of a **Virtual DOM**

Virtual DOM is a lightweight representation of the browser's DOM.

Whenever one variable is changed, a new Virtual DOM is created and compared with its previous version, the **diff** is then updated in the DOM.

Uses **JSX**: XHTML mixed within the JavaScript code, therefore requires a compiler that transform this JS and XHTML mix into HTML

Largest developer community amongst the big SPA frameworks (unlikely to change in the near future)

Easier to get started, but requires several plugins (e.g., for global state management).

Vue



<https://vuejs.org/>

Newest (of the three)

Similar to Angular, but more flexible, does not enforce a coding pattern

Uses **SFCs** (Single File Components) where JS, CSS, and HTML are combined together.

Easiest to get started, allows for faster prototyping and later extensions via components

Small size and higher performance

```
<template>
  <p>{{ greeting }} World!</p>
</template>

<script>
module.exports = {
  data: function() {
    return {
      greeting: "Hello"
    }
  }
}
</script>

<style scoped>
p {
  font-size: 2em;
  text-align: center;
}
</style>
```

[Image source](#)

Which one?



In technology, **network effects** are king. You want to build your app on a technological stack/framework that has a large and thriving community. *Which one?*

Which one?



In technology, **network effects** are king. You want to build your app on a technological stack/framework that has a large and thriving community. *Which one?*

Let's look at the data in 2021:

- State of JS Survey
- State of CSS Survey
- NPM Trends
- Stack Overflow Insights Survey
- Stack Overflow Insights Trends

State of JS Survey 2020

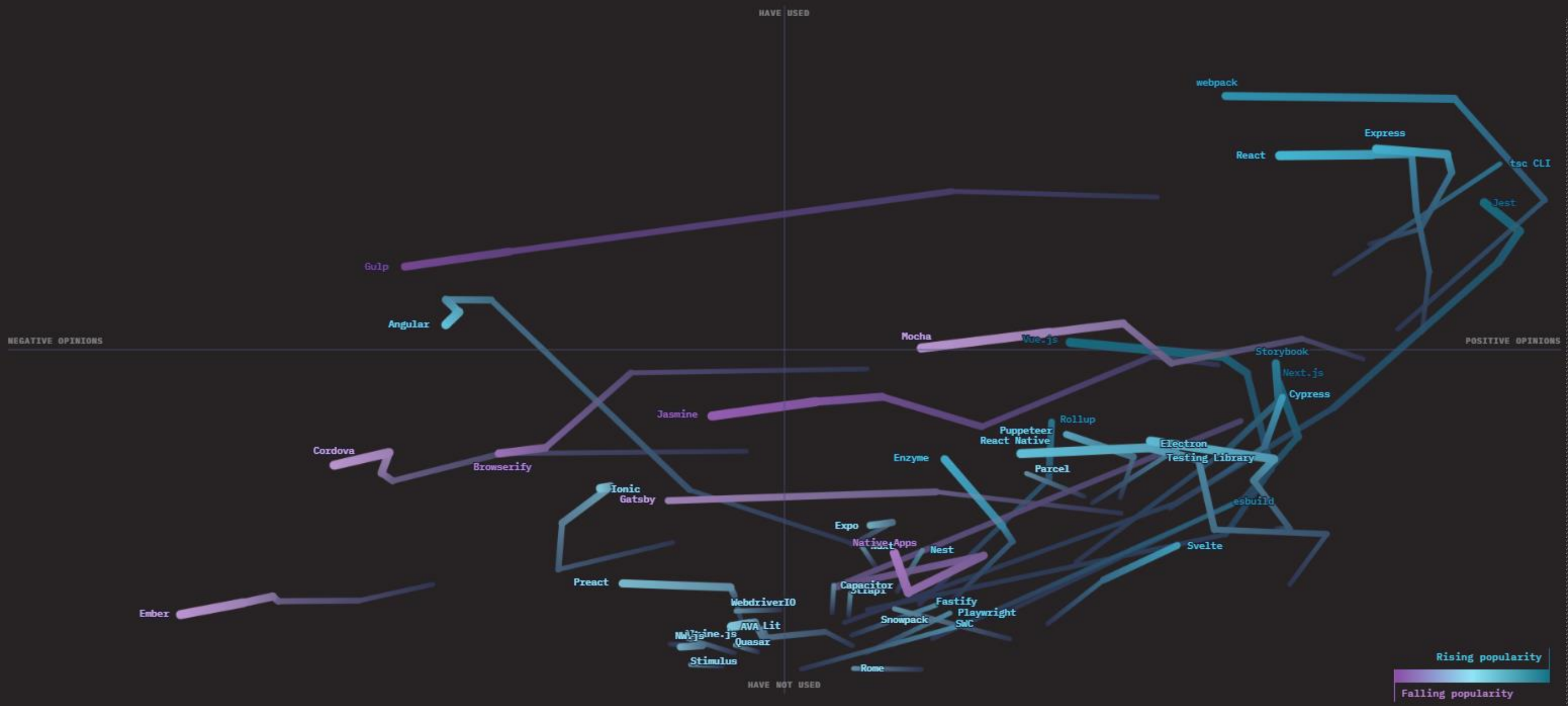


16,085 people

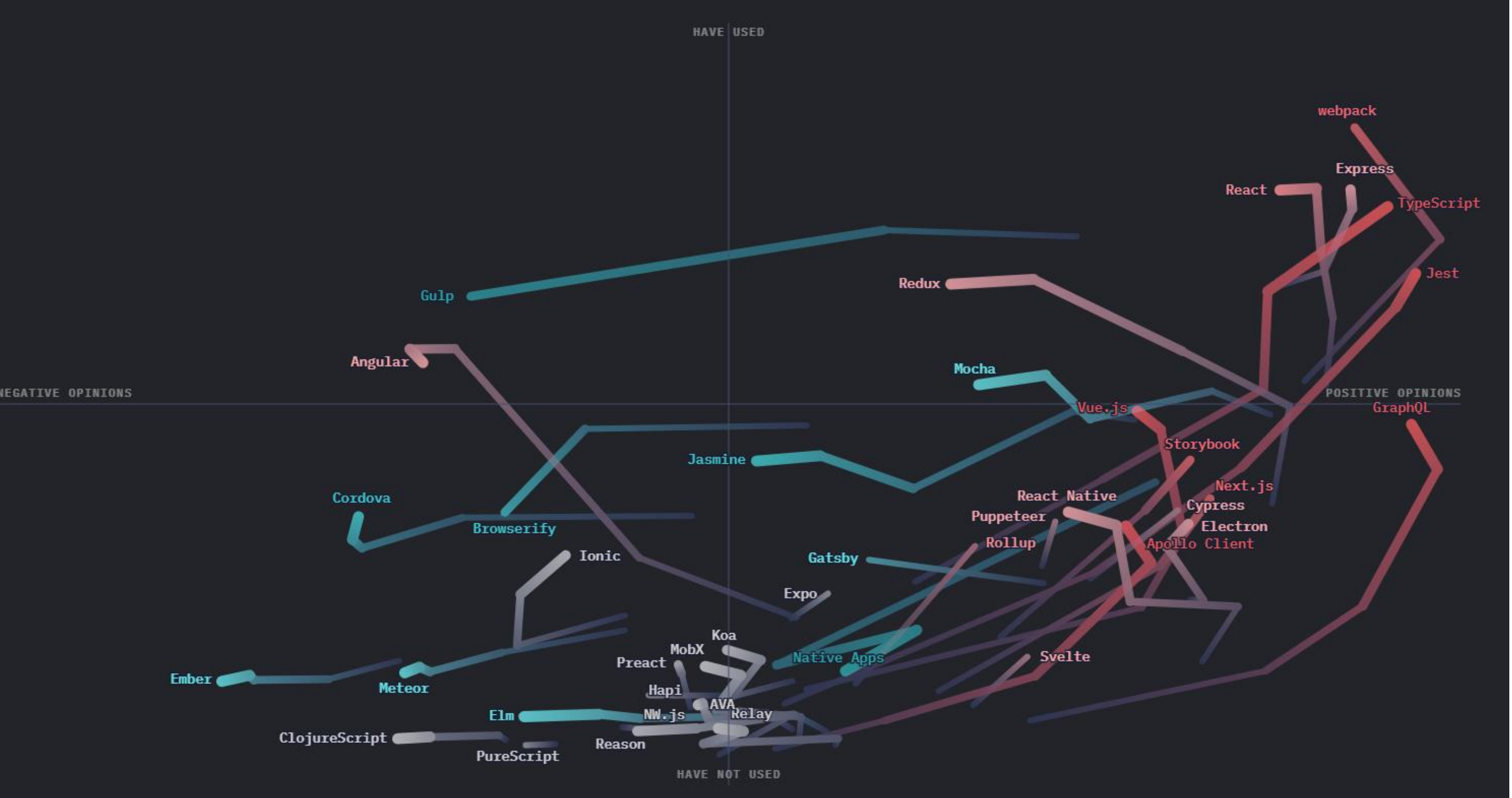
<https://2021.stateofjs.com/>

CHANGES OVER TIME

Each line goes from 2016 to 2020. A higher point means a technology has been used by more people, and a point further to the right means more users want to learn it; or have used it and would use it again.



<https://2021.stateofjs.com/en-US/libraries>

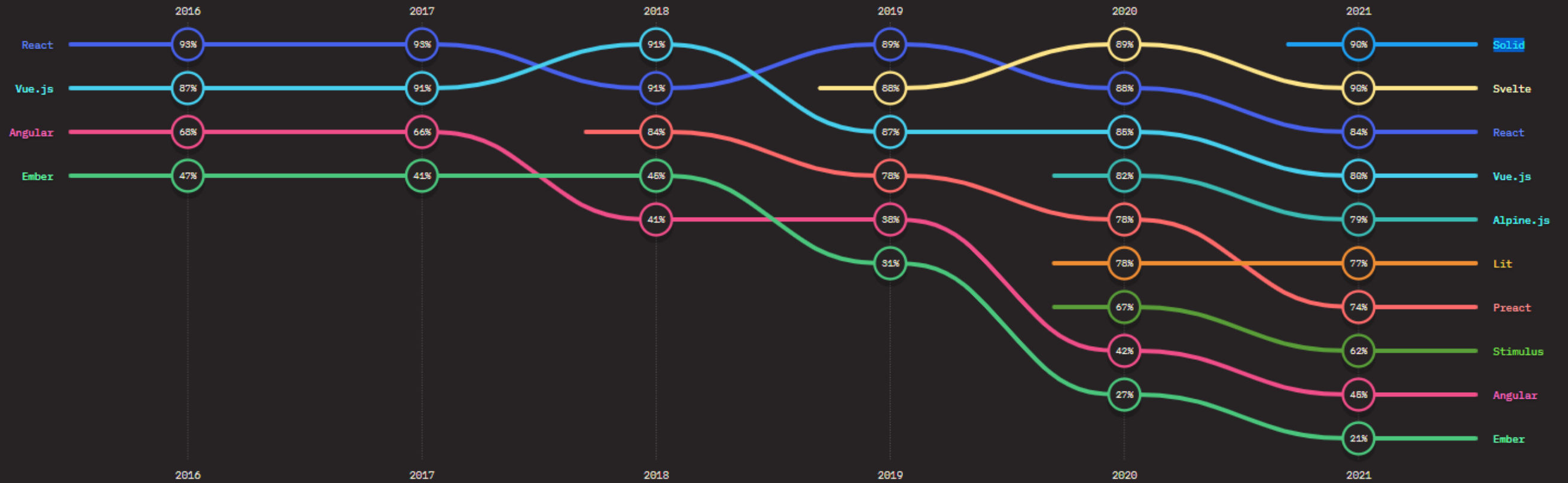


State of JS: Front End Frameworks Satisfaction

RANKINGS

Satisfaction, interest, usage, and awareness ratio rankings.

Satisfaction Interest Usage Awareness

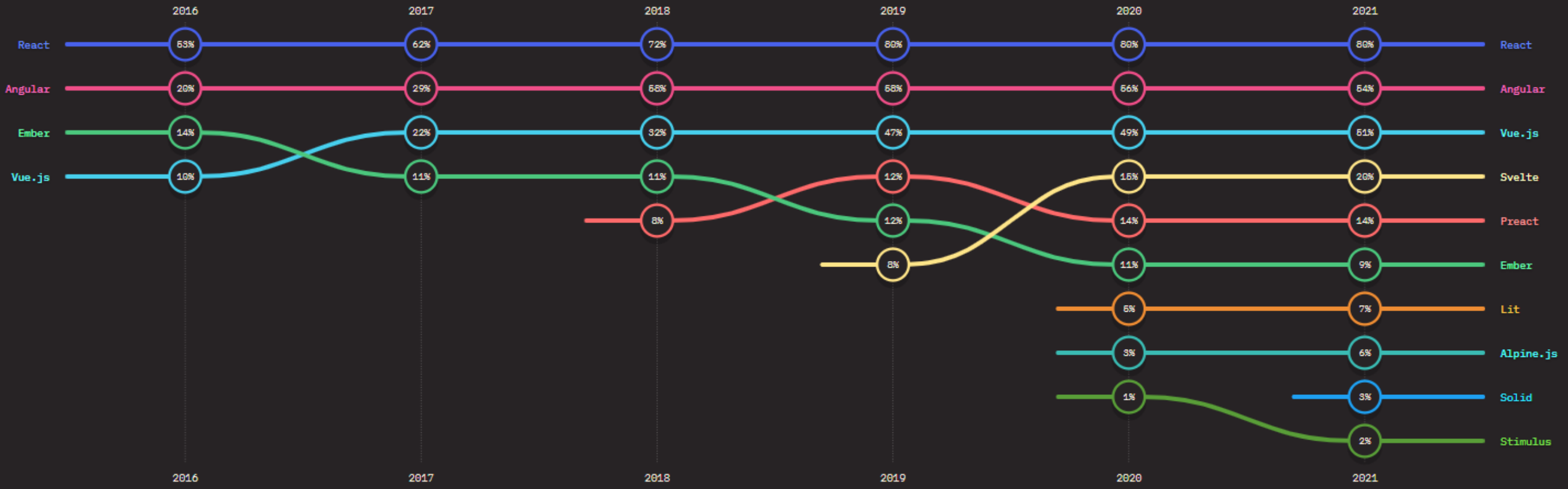


State of JS: Front End Frameworks Usage

RANKINGS

Satisfaction, interest, usage, and awareness ratio rankings.

Satisfaction Interest **Usage** Awareness

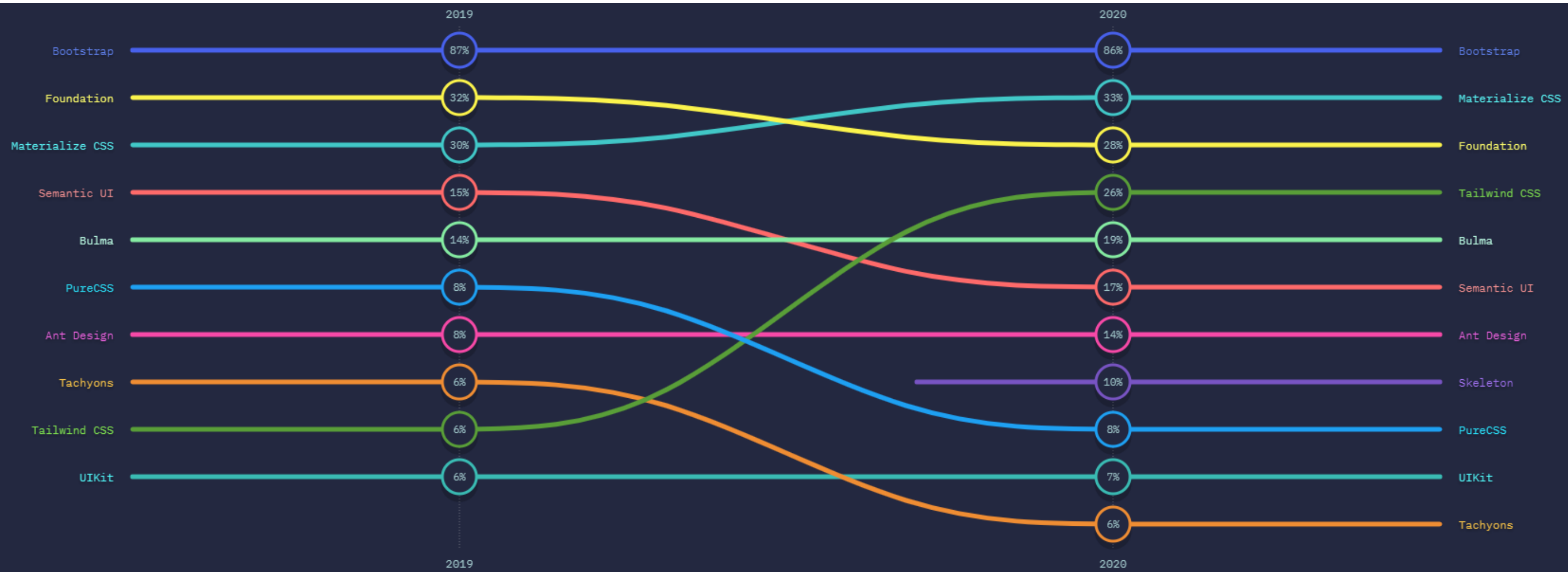




11,000 respondents from 102 countries

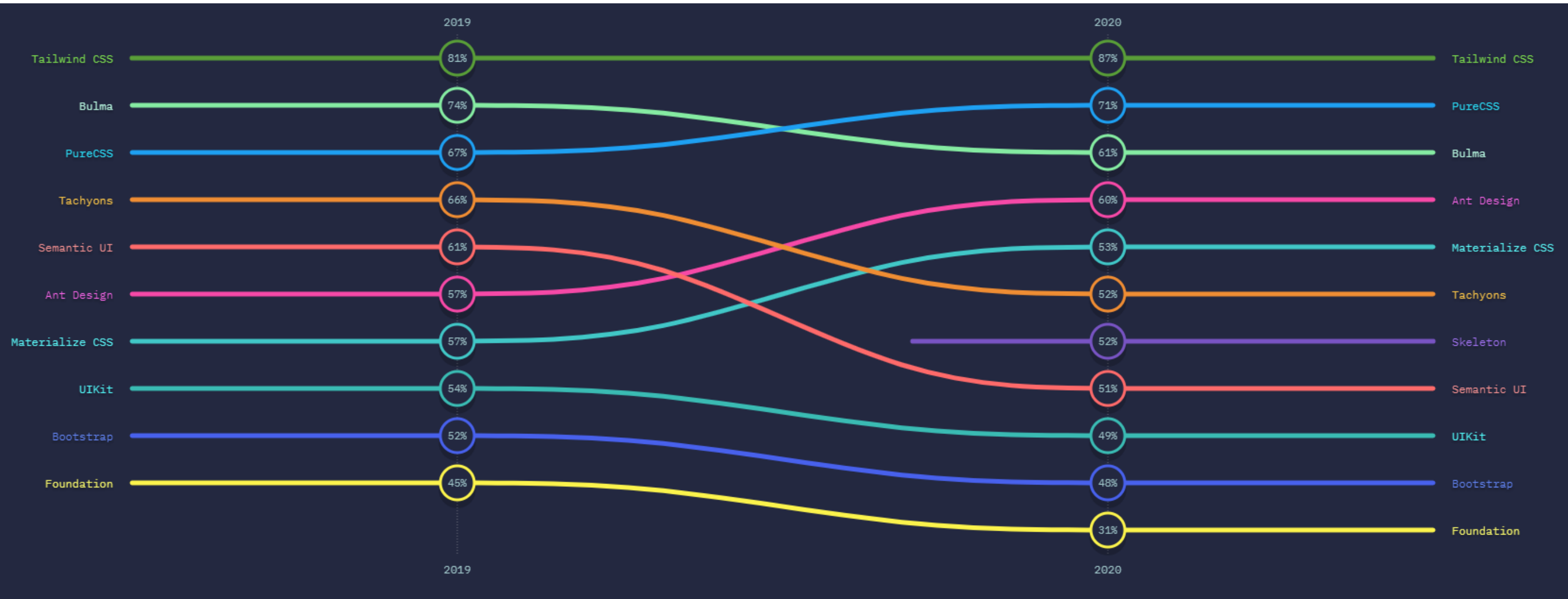
<https://2020.stateofcss.com/en-US/report/>

State of CSS: Frameworks Usage



<https://2020.stateofcss.com/en-US/report/>

State of CSS: Frameworks Satisfaction



<https://2020.stateofcss.com/en-US/report/>



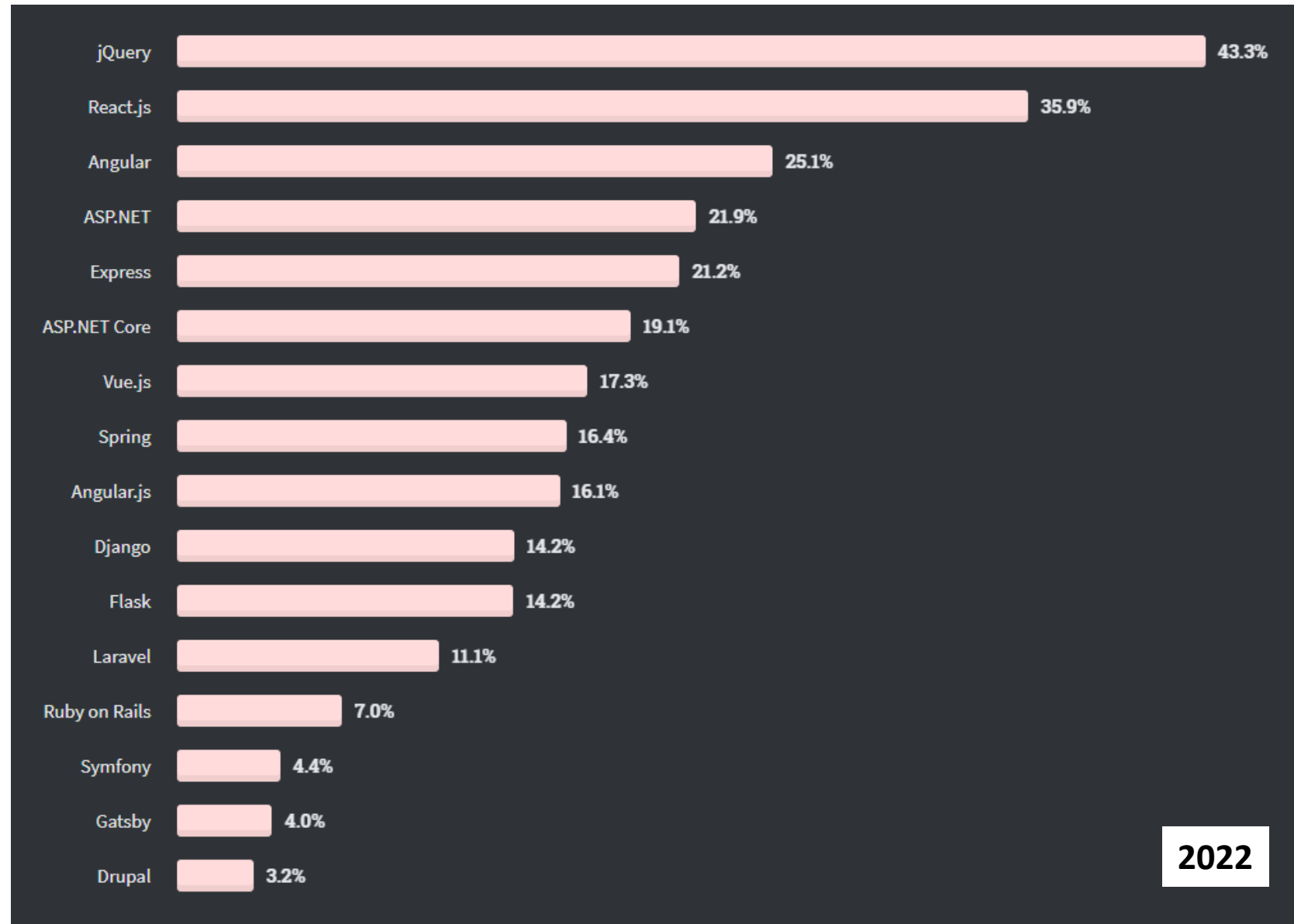
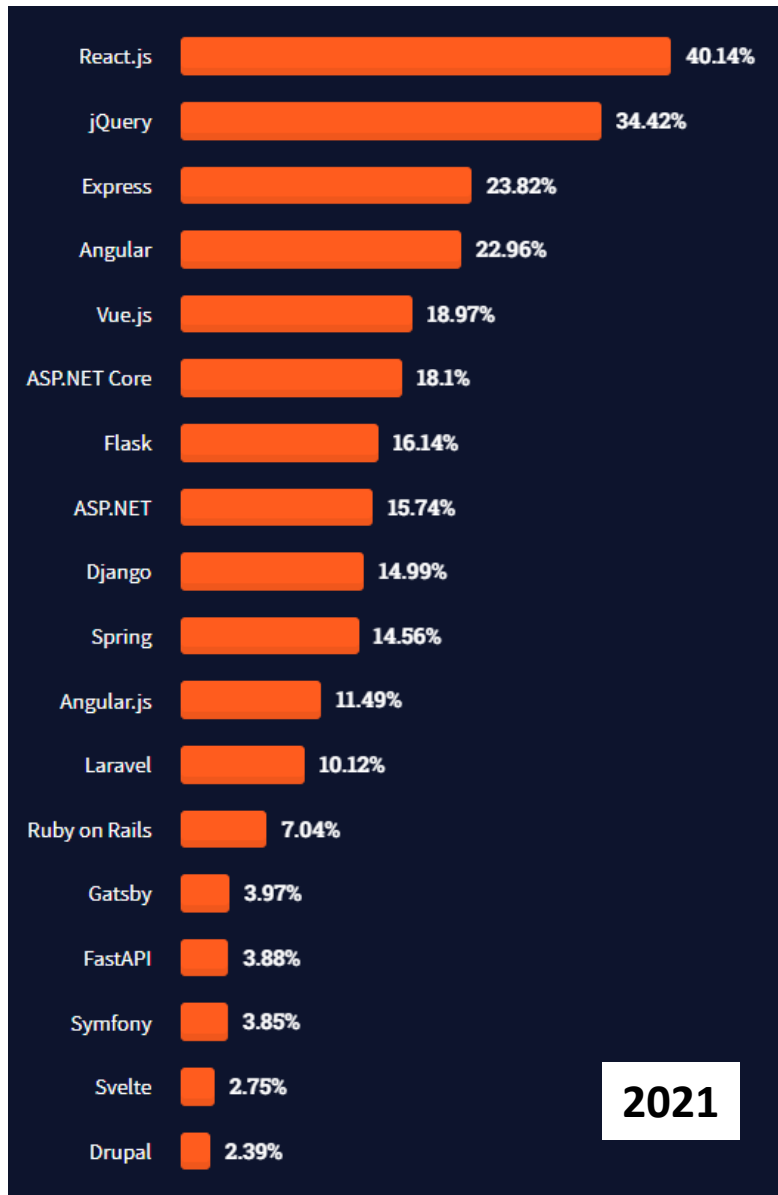
2021 Developer Survey

In May 2021 over 80,000 developers told us how they learn and level up, which tools they're using, and what they want.

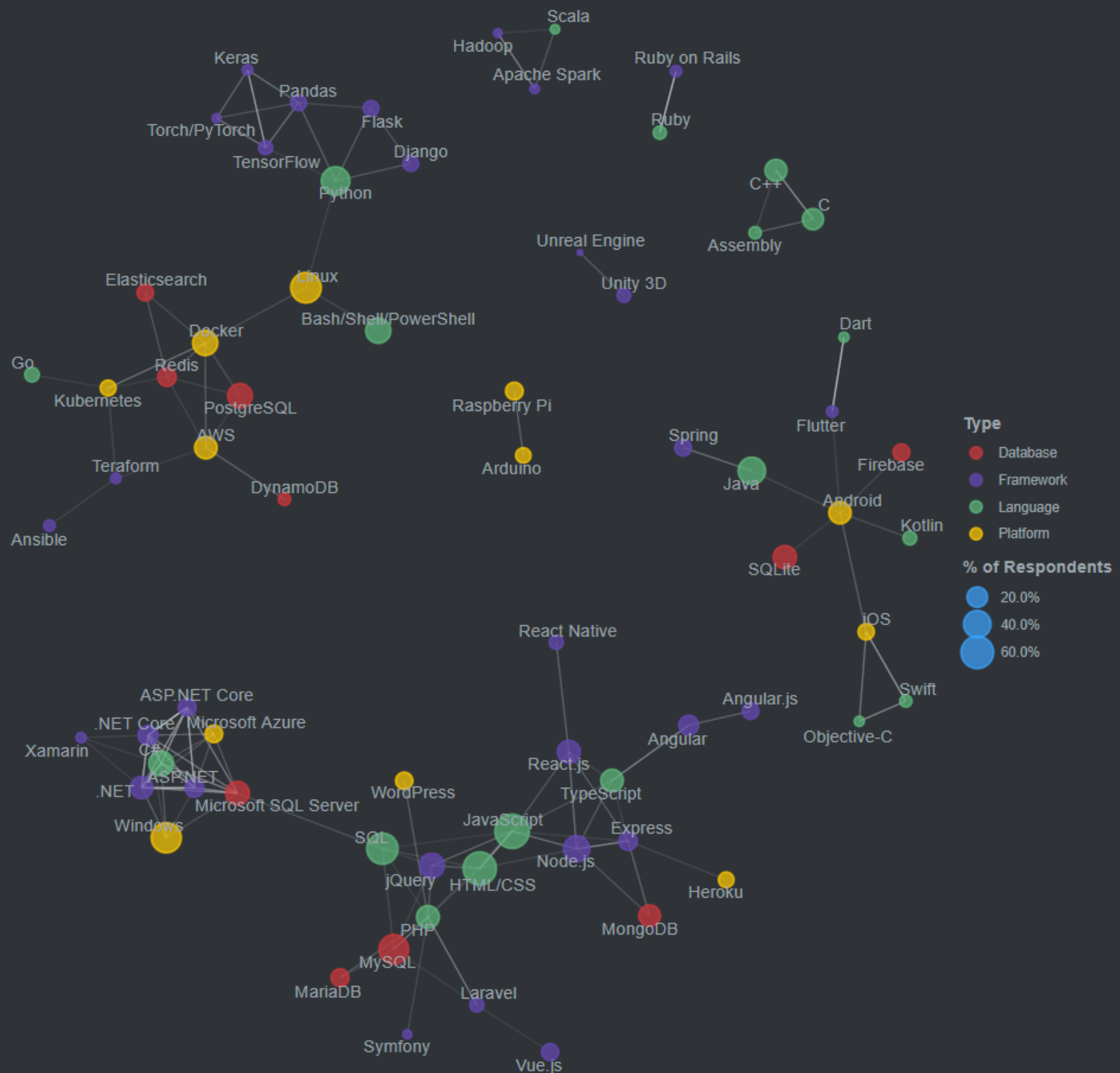
[Read the overview →](#)

[Methodology →](#)

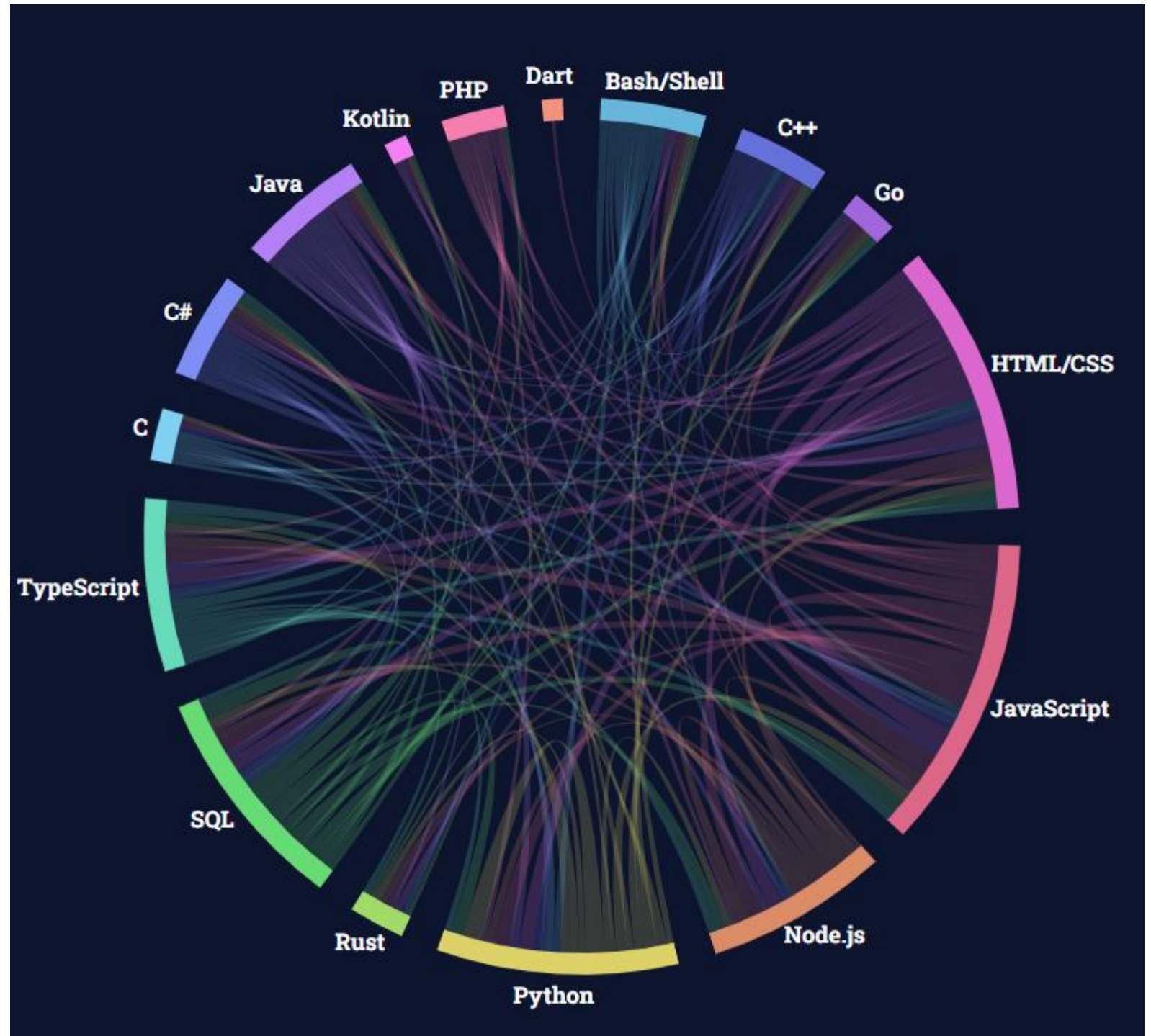
Stack Overflow Survey: Web Frameworks



Stack Overflow Survey: Platforms

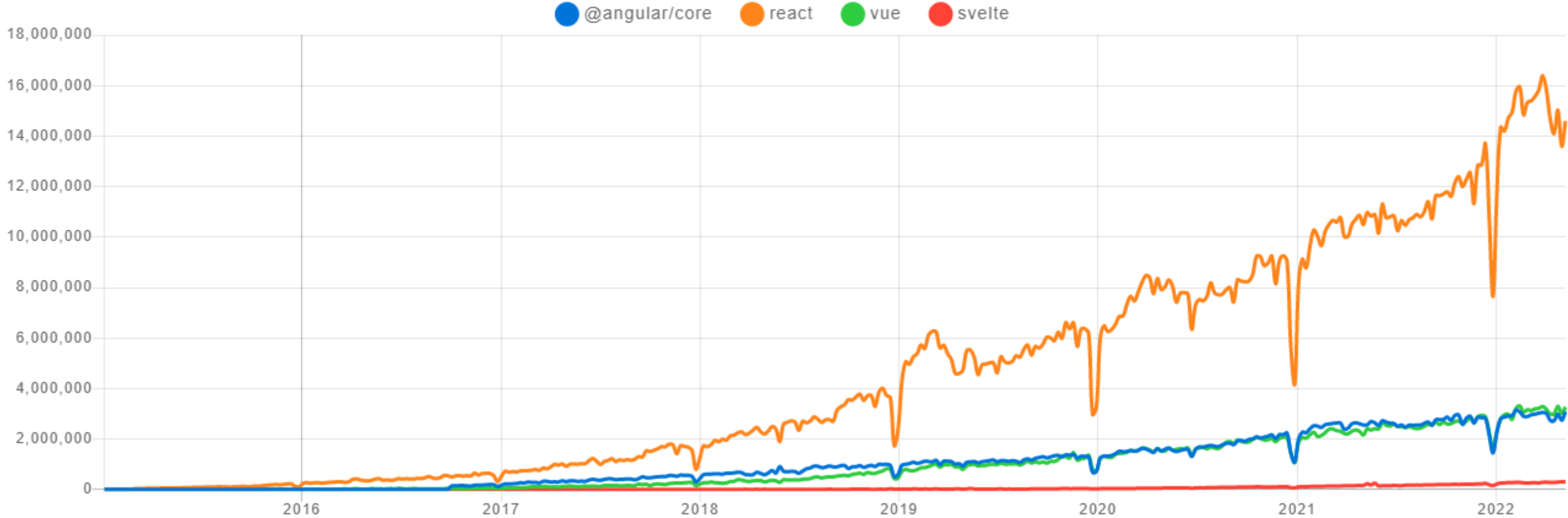


Stack Overflow Survey: Connections



NPM Trends

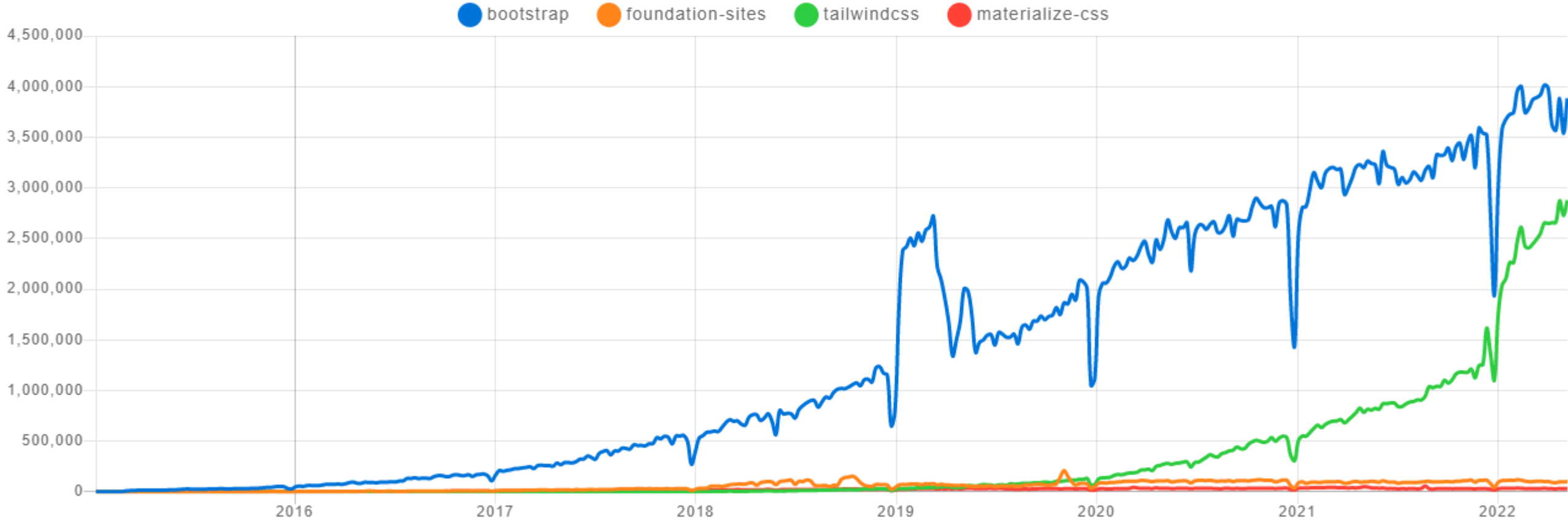
Downloads in past All time ▾



<https://www.npmtrends.com/@angular/core-vs-react-vs-vue-vs-svelte>

NPM Trends

Downloads in past All time ▾



<https://www.npmtrends.com/bootstrap-vs-foundation-sites-vs-tailwindcss-vs-materialize-css>

Stack Overflow Trends

See how technologies have trended over time based on use of their tags since 2008, when Stack Overflow was founded. Enter up to 15 tags to compare growth and decline.

Tags:

reactjs ×

angular ×

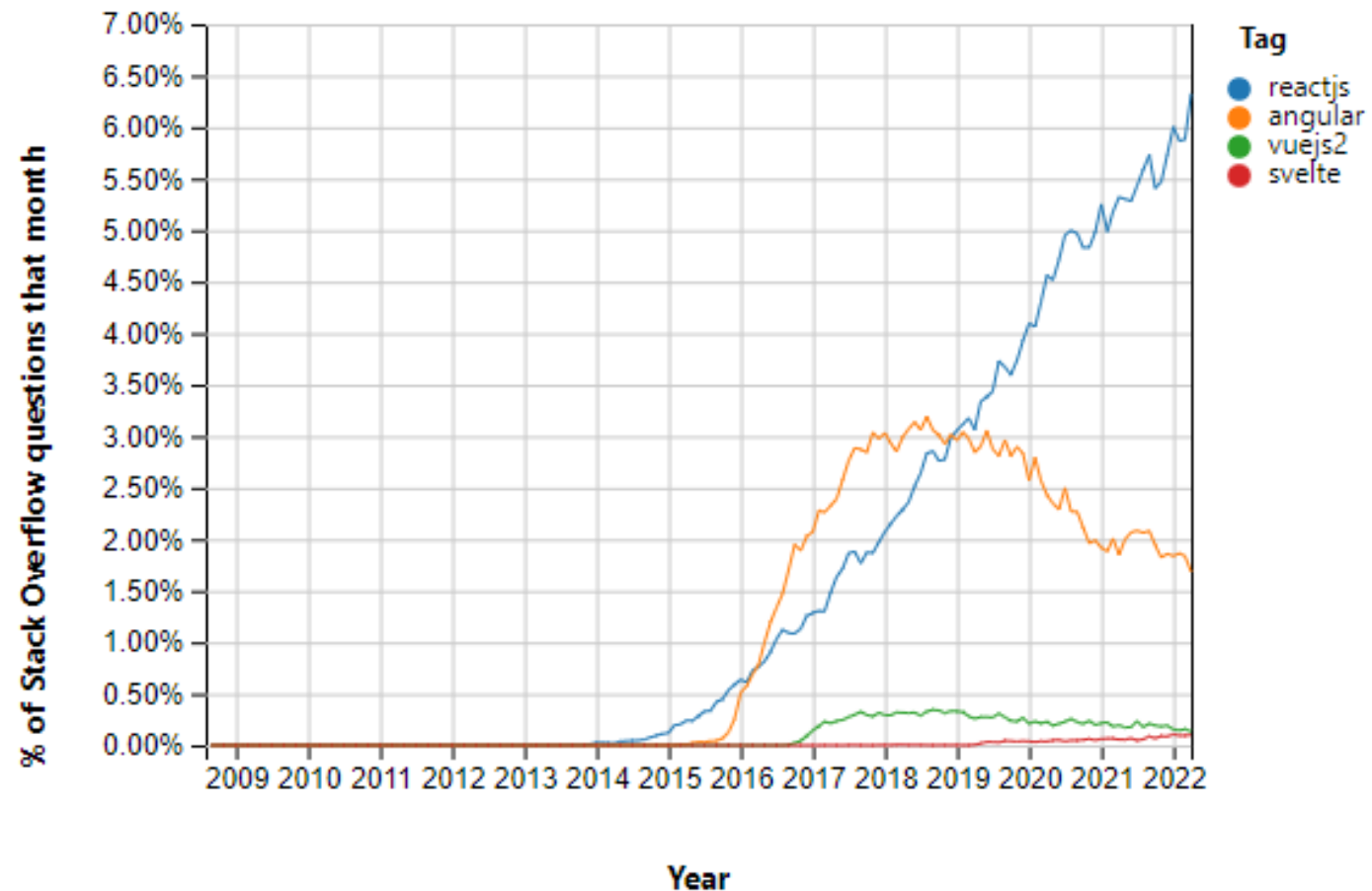
vuejs2 ×

svelte ×

Don't know what tags to look at? Try one of our presets:

- [Most Popular Languages \(TIOBE Index for May 2017\)](#)
- [Operating Systems](#)
- [Mobile Operating Systems](#)
- [Javascript Frameworks](#)
- [Smaller Javascript Frameworks](#)
- [Closed-source Browser Plugins](#)
- [Data Science and Big Data](#)
- [Apache Open-source Projects](#)

For more on this tool and what you can learn from it, see our [blog post](#).



<https://insights.stackoverflow.com/trends?tags=reactjs%2Cangular%2Cvuejs2>

SVELTE: The "Disappearing Framework" compiles to vanilla JS

SVELTE

Cybernetically enhanced web apps



Write less code

Build boilerplate-free components using languages you already know – HTML, CSS and JavaScript

[learn more →](#)

No virtual DOM

Svelte compiles your code to tiny, framework-less vanilla JS – your app starts fast and stays fast

[learn more →](#)

Truly reactive

No more complex state management libraries – Svelte brings reactivity to JavaScript itself

[learn more →](#)

Performance Benchmark

<https://krausest.github.io/js-framework-benchmark/current.html>

<https://javascript.plainenglish.io/javascript-frameworks-performance-comparison-2020-cd881ac21fce>

Fast Group

Svelte

Avg. Group

Vue

Slowest Group

React

Angular

Name Duration for...	vanillajs-1	vanillajs	mikado-v0.7.64	wasm-bindgen-v0.2.47	vanillajs-vc	fullweb-helpers-v0.1.0	solid-v0.20.0	1more-v0.1.5	fullweb-template-v0.1.0	solid-state-v0.20.0	domdiff-v2.2.2	sinuous-v0.27.12	sifrr-v0.0.5	frtags-v0.4.7	ko-jsx-v0.16.1	stdweb-v0.4.17	vuex-jsx-v0.2.0	atfodom-v0.12.0	dohtml	ivi-v0.27.1	mimbl-v0.6.5	scarlets-frame-v0.34.6	petit-dom-v0.2.0	s2-v1.0.0	inferno-v7.4.8	san-v3.10.1	mobx-jsx-v0.14.0	dominator-v0.5.0	uhtml-v1.8.1	
Implementation notes	772	772		772	772	772			772		772	800 801	800 801			772		772	772					800			800			801
create rows creating 1,000 rows	77.9 ± 0.8 (1.00)	78.7 ± 1.3 (1.01)	84.0 ± 0.8 (1.08)	85.7 ± 1.0 (1.10)	88.9 ± 0.7 (1.14)	89.7 ± 1.0 (1.15)	83.5 ± 0.7 (1.07)	84.7 ± 1.5 (1.09)	93.4 ± 1.4 (1.20)	85.6 ± 1.1 (1.10)	86.3 ± 1.1 (1.11)	97.7 ± 1.5 (1.25)	85.4 ± 0.9 (1.10)	98.8 ± 1.1 (1.27)	93.4 ± 1.5 (1.20)	98.4 ± 1.2 (1.28)	91.8 ± 1.5 (1.18)	90.7 ± 1.0 (1.16)	86.3 ± 0.4 (1.11)	94.0 ± 2.4 (1.21)	110.4 ± 2.8 (1.42)	99.2 ± 1.5 (1.27)	100.5 ± 1.5 (1.29)	104.3 ± 2.1 (1.34)	94.0 ± 1.9 (1.21)	92.7 ± 1.6 (1.19)	103.7 ± 1.2 (1.33)	109.6 ± 0.8 (1.41)	93.5 ± 1.1 (1.20)	
replace all rows updating all 1,000 rows (5 warmup runs)	83.6 ± 0.7 (1.00)	83.7 ± 1.4 (1.00)	85.4 ± 1.1 (1.02)	88.5 ± 0.8 (1.06)	90.7 ± 1.0 (1.09)	90.1 ± 0.5 (1.06)	89.3 ± 1.6 (1.07)	84.3 ± 0.7 (1.01)	93.9 ± 1.3 (1.12)	88.9 ± 0.9 (1.08)	92.9 ± 0.5 (1.11)	97.2 ± 1.6 (1.16)	86.2 ± 1.0 (1.03)	101.6 ± 1.0 (1.22)	89.1 ± 0.9 (1.07)	94.9 ± 0.8 (1.14)	95.1 ± 2.1 (1.14)	92.4 ± 0.8 (1.11)	131.1 ± 0.6 (1.57)	88.7 ± 1.0 (1.06)	97.0 ± 1.0 (1.16)	94.4 ± 1.0 (1.13)	95.9 ± 1.0 (1.15)	101.7 ± 1.2 (1.22)	88.0 ± 1.6 (1.05)	89.5 ± 0.7 (1.07)	94.5 ± 1.2 (1.13)	103.8 ± 0.8 (1.24)	92.8 ± 1.4 (1.11)	
partial update updating every 10th row for 1,000 rows (3 warmup runs), 16x CPU slowdown	121.6 ± 2.4 (1.04)	123.5 ± 2.0 (1.05)	117.3 ± 2.1 (1.00)	123.8 ± 1.3 (1.06)	124.6 ± 2.4 (1.06)	125.1 ± 2.0 (1.07)	119.8 ± 2.3 (1.02)	124.7 ± 3.5 (1.06)	118.6 ± 1.4 (1.01)	125.4 ± 2.3 (1.07)	126.7 ± 1.4 (1.08)	122.5 ± 1.0 (1.04)	127.5 ± 1.7 (1.09)	126.6 ± 2.4 (1.08)	138.6 ± 1.8 (1.18)	124.3 ± 1.5 (1.08)	134.2 ± 2.1 (1.14)	133.9 ± 1.4 (1.14)	139.7 ± 1.3 (1.19)	159.4 ± 2.8 (1.36)	121.7 ± 1.3 (1.04)	127.3 ± 1.5 (1.09)	132.7 ± 2.0 (1.13)	128.1 ± 1.7 (1.09)	133.0 ± 1.4 (1.13)	158.6 ± 5.9 (1.35)	156.2 ± 2.1 (1.33)	125.0 ± 2.4 (1.07)	128.2 ± 3.1 (1.09)	
select row highlighting a selected row, (no warmup runs), 16x CPU slowdown	18.7 ± 1.3 (1.12)	19.7 ± 1.4 (1.18)	20.3 ± 1.4 (1.21)	21.6 ± 0.8 (1.29)	19.3 ± 1.5 (1.15)	18.3 ± 1.5 (1.09)	21.2 ± 1.5 (1.28)	25.4 ± 1.8 (1.51)	19.7 ± 1.6 (1.18)	23.4 ± 1.6 (1.39)	24.7 ± 1.2 (1.47)	18.0 ± 1.5 (1.07)	35.8 ± 1.1 (2.13)	18.8 ± 1.4 (1.12)	20.5 ± 1.9 (1.22)	21.9 ± 1.4 (1.30)	24.2 ± 1.3 (1.44)	36.7 ± 1.1 (2.19)	18.1 ± 1.2 (1.08)	28.4 ± 1.7 (1.68)	19.8 ± 1.2 (1.18)	33.1 ± 1.8 (1.97)	34.9 ± 2.4 (2.08)	24.9 ± 1.4 (1.48)	45.5 ± 1.9 (2.71)	33.8 ± 0.7 (2.01)	21.4 ± 1.9 (1.27)	21.7 ± 1.6 (1.29)	41.3 ± 1.7 (2.48)	
swap rows swap 2 rows for table with 1,000 rows, (5 warmup runs), 4x CPU slowdown	40.7 ± 0.4 (1.01)	41.2 ± 0.5 (1.02)	40.5 ± 0.4 (1.00)	40.8 ± 0.4 (1.01)	41.7 ± 0.4 (1.03)	41.9 ± 0.5 (1.04)	42.6 ± 0.6 (1.05)	42.1 ± 0.3 (1.04)	42.4 ± 0.6 (1.05)	43.9 ± 0.4 (1.08)	43.8 ± 0.5 (1.08)	43.1 ± 0.4 (1.07)	44.0 ± 0.6 (1.09)	43.2 ± 0.4 (1.07)	42.1 ± 0.3 (1.04)	41.3 ± 0.4 (1.02)	48.2 ± 0.8 (1.19)	43.2 ± 0.4 (1.07)	42.7 ± 0.8 (1.05)	44.1 ± 0.4 (1.09)	42.1 ± 0.4 (1.04)	42.5 ± 0.4 (1.05)	42.9 ± 0.5 (1.06)	42.3 ± 0.9 (1.04)	41.7 ± 0.4 (1.03)	41.9 ± 0.2 (1.04)	46.3 ± 0.4 (1.14)	41.9 ± 0.3 (1.03)	43.5 ± 0.5 (1.08)	
remove row removing one row, (5 warmup runs)	19.0 ± 0.2 (1.00)	19.1 ± 0.3 (1.00)	19.1 ± 0.3 (1.00)	19.0 ± 0.2 (1.00)	19.5 ± 0.4 (1.03)	19.5 ± 0.7 (1.03)	20.6 ± 0.8 (1.08)	19.1 ± 0.6 (1.01)	19.6 ± 0.2 (1.03)	20.5 ± 0.1 (1.08)	19.8 ± 0.4 (1.04)	19.8 ± 0.2 (1.04)	19.8 ± 0.5 (1.05)	20.6 ± 1.0 (1.09)	20.0 ± 0.5 (1.05)	20.2 ± 0.5 (1.11)	21.1 ± 0.2 (1.02)	19.4 ± 0.3 (1.02)	22.5 ± 0.2 (1.02)	19.3 ± 0.5 (1.09)	19.9 ± 0.3 (1.05)	19.6 ± 0.8 (1.04)	19.7 ± 0.5 (1.04)	19.4 ± 0.6 (1.02)	19.6 ± 0.6 (1.04)	21.2 ± 0.5 (1.12)	19.3 ± 0.6 (1.03)	20.3 ± 0.3 (1.07)		
create many rows creating 10,000 rows	767.0 ± 6.4 (1.00)	773.1 ± 7.6 (1.01)	795.2 ± 3.1 (1.04)	811.6 ± 7.2 (1.06)	847.7 ± 17.1 (1.11)	885.9 ± 3.6 (1.16)	816.3 ± 4.6 (1.08)	791.2 ± 6.0 (1.03)	937.4 ± 4.0 (1.22)	830.5 ± 6.1 (1.08)	838.8 ± 16.9 (1.09)	956.1 ± 6.4 (1.25)	790.9 ± 13.0 (1.03)	985.8 ± 3.8 (1.29)	870.2 ± 3.5 (1.13)	961.9 ± 6.7 (1.25)	877.3 ± 9.0 (1.14)	892.9 ± 6.4 (1.16)	1,425.1 ± 10.1 (1.86)	848.1 ± 4.3 (1.11)	1,051.2 ± 10.0 (1.37)	947.0 ± 15.2 (1.22)	927.6 ± 4.8 (1.21)	999.5 ± 3.2 (1.30)	837.8 ± 13.0 (1.09)	860.9 ± 15.1 (1.12)	1,059.9 ± 28.7 (1.38)	1,244.7 ± 50.4 (1.62)	929.9 ± 2.4 (1.21)	
append rows to large table appending 1,000 to a table of 10,000 rows, 2x CPU slowdown	169.6 ± 1.4 (1.00)	171.0 ± 1.7 (1.01)	177.8 ± 0.9 (1.05)	184.5 ± 1.2 (1.09)	188.5 ± 1.2 (1.11)	188.6 ± 1.8 (1.11)	179.0 ± 1.2 (1.06)	175.2 ± 1.2 (1.03)	190.5 ± 1.5 (1.12)	185.5 ± 1.6 (1.09)	184.5 ± 0.9 (1.09)	199.3 ± 2.3 (1.17)	180.3 ± 1.4 (1.08)	206.8 ± 2.1 (1.22)	194.6 ± 3.2 (1.15)	204.4 ± 2.0 (1.21)	197.7 ± 1.3 (1.17)	188.3 ± 0.8 (1.11)	177.4 ± 1.0 (1.05)	195.7 ± 1.8 (1.15)	217.3 ± 2.9 (1.28)	191.1 ± 3.0 (1.13)	196.6 ± 1.7 (1.18)	207.6 ± 1.0 (1.22)	184.6 ± 1.5 (1.09)	189.2 ± 1.3 (1.12)	210.1 ± 1.8 (1.24)	215.1 ± 0.6 (1.27)	207.3 ± 1.9 (1.22)	
clear rows clearing a table with 1,000 rows, 8x CPU slowdown	81.8 ± 0.7 (1.00)	83.9 ± 0.5 (1.03)	82.3 ± 0.7 (1.01)	88.7 ± 0.7 (1.09)	83.6 ± 0.6 (1.02)	84.9 ± 0.5 (1.04)	89.0 ± 0.4 (1.09)	85.9 ± 0.8 (1.05)	83.2 ± 0.5 (1.02)	90.1 ± 1.1 (1.10)	92.8 ± 0.6 (1.14)	102.3 ± 0.8 (1.25)	84.5 ± 1.1 (1.04)	85.4 ± 0.8 (1.05)	112.5 ± 1.1 (1.38)	93.0 ± 0.7 (1.14)	91.9 ± 0.8 (1.13)	85.7 ± 0.7 (1.05)	87.3 ± 0.8 (1.07)	84.4 ± 0.5 (1.03)	111.7 ± 0.7 (1.37)	91.1 ± 0.7 (1.12)	85.6 ± 0.7 (1.05)	107.0 ± 0.7 (1.31)	101.4 ± 0.9 (1.24)	107.4 ± 0.6 (1.32)	96.2 ± 0.8 (1.18)	105.1 ± 1.1 (1.29)	93.6 ± 0.5 (1.15)	
geometric mean of all factors in the table	1.02	1.03	1.04	1.08	1.08	1.08	1.08	1.08	1.10	1.12	1.13	1.14	1.14	1.15	1.15	1.16	1.18	1.19	1.20	1.20	1.20	1.21	1.22	1.22	1.22	1.22	1.23	1.24	1.24	

Check Them Out



Helping you **select** an MV* framework

Download

View on GitHub

Blog



<https://todomvc.com/>

Some References



<https://www.imaginarycloud.com/blog/a-javascript-ecosystem-overview/>

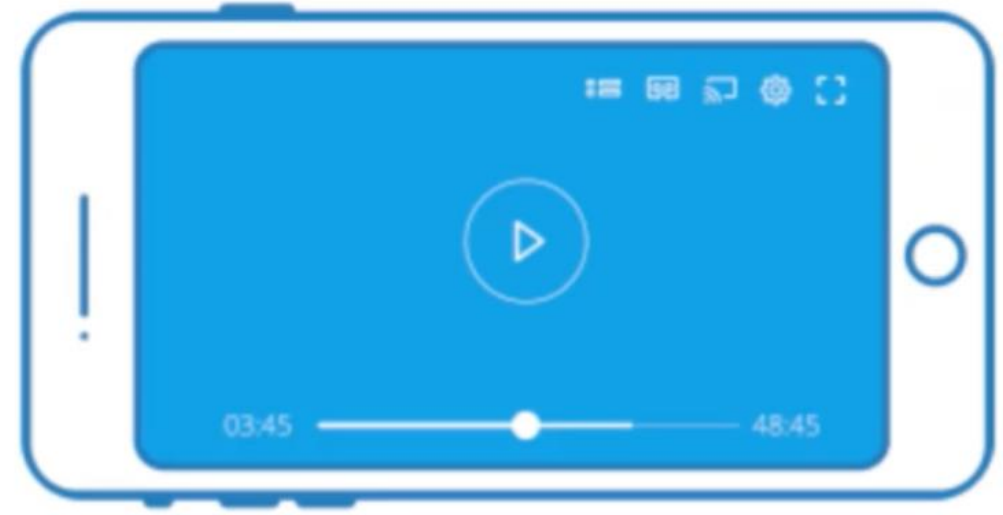
<https://codeburst.io/stack-choices-react-vs-vue-vs-angular-vs-svelte-49aa0170c634>

<https://dzone.com/articles/angular-vs-react-vs-vue-which-framework-is-best-to>

Mobile Dev Frameworks



Android SDK



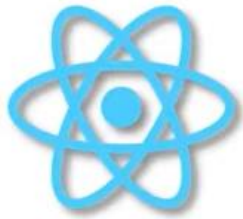
iOS SDK

Overview of Technologies

	Native iOS, Android	React Native	Xamarin	Flutter	Ionic	Capacitor	Cordova
Language(s)	Swift, Kotlin Objective C, Java, C++	JavaScript	C#	Dart	HTML, CSS, JavaScript	HTML, CSS, JavaScript	HTML, CSS, JavaScript
UI	Native UI	Native UI	Native UI	Custom	Web UI	Web UI	Web UI
Mobile UI toolkit	Yes	Yes	Yes	Yes	Yes		
Access to native SDK	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Framework(s) supported		React	.NET		Angular, React, Vue	Any JS Framework	Any JS Framework
Use custom UI libraries or Web Components					Yes	Yes	Yes
Convert web app to mobile						Yes	Yes
Access to dedicated support team					Yes	Yes	
Premium mobile security, storage, and integrations					Yes	Yes	
Cloud app publishing and CI/CD			Yes		Yes	Yes	

[Table source](#)

React Native



Flutter



NativeScript

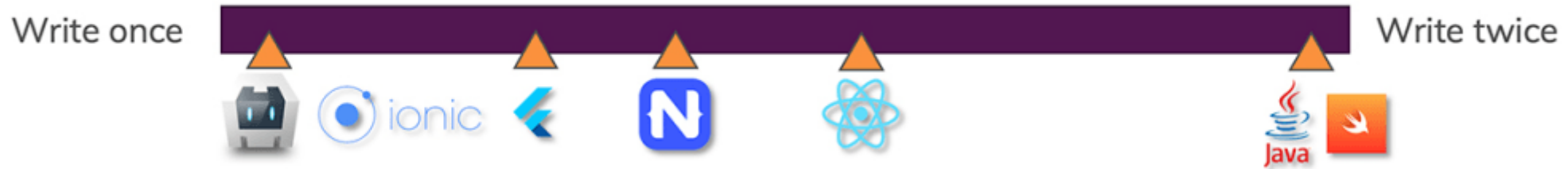


Ionic Framework



Uses JavaScript	Uses Dart	Uses JavaScript	Uses JavaScript
Partly Compiled to Native Code	Compiled to ARM C/C++ Library	Partly Compiled to Native Code	Not Compiled to Native Code (Wrapped Web App)
JS was not invented for this (but turns out to work really well...)	Dart was also not invented for this BUT is developed by the same company (Google)	JS was not invented for this (but turns out to work really well...)	JS was not invented for this (but turns out to work really well...)
Ships with some pre-built, partly adaptive components	Ships with a lot of pre-built, partly adaptive components	Ships with some pre-built, mostly adaptive components	Ships with loads of pre-built, fully adaptive components

“Write once, use everywhere”



Rich pre-styled Component Library



Performance



Accessing Native Device Features



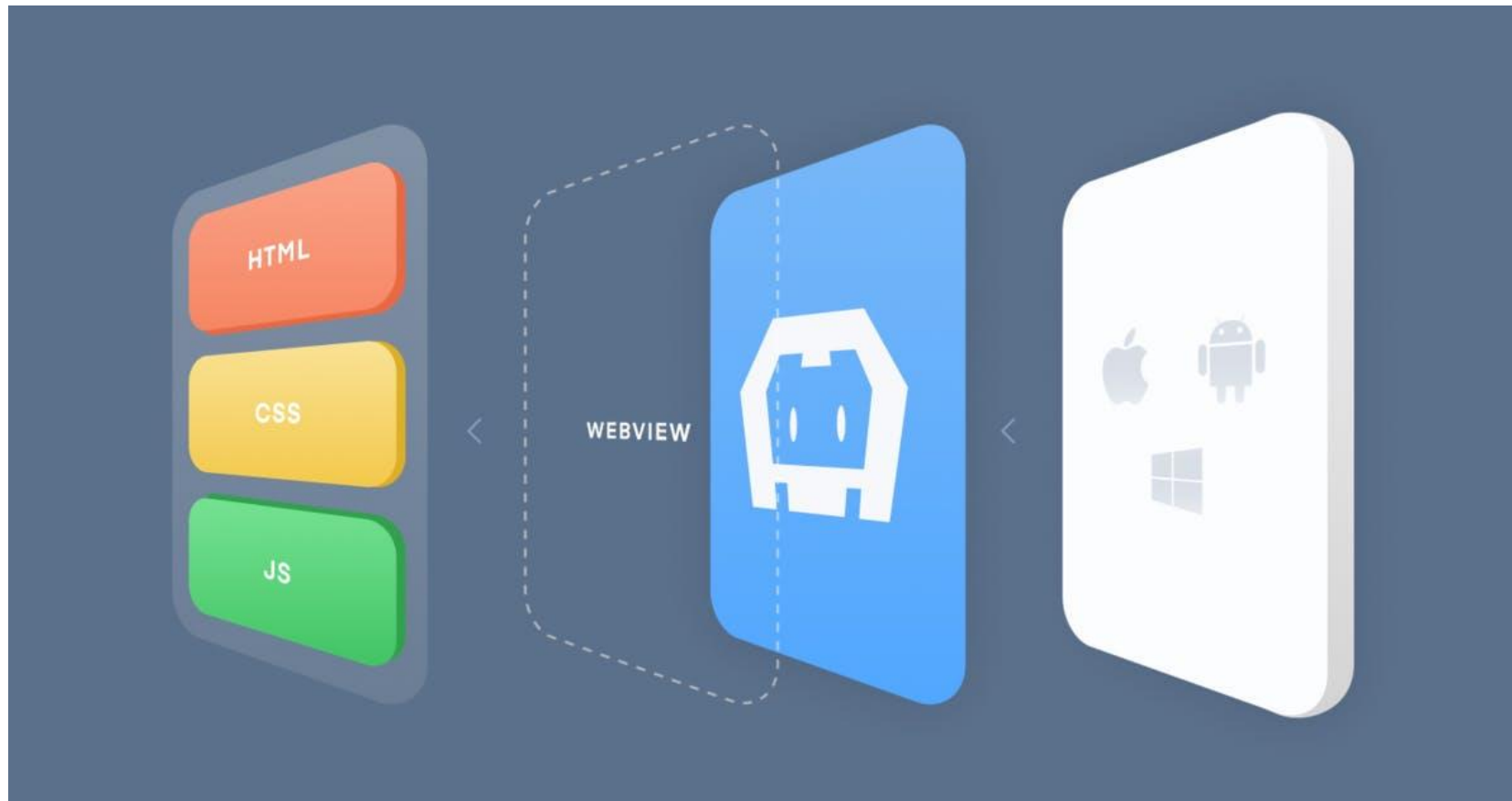
Apache Cordova



<https://cordova.apache.org/>

Cordova

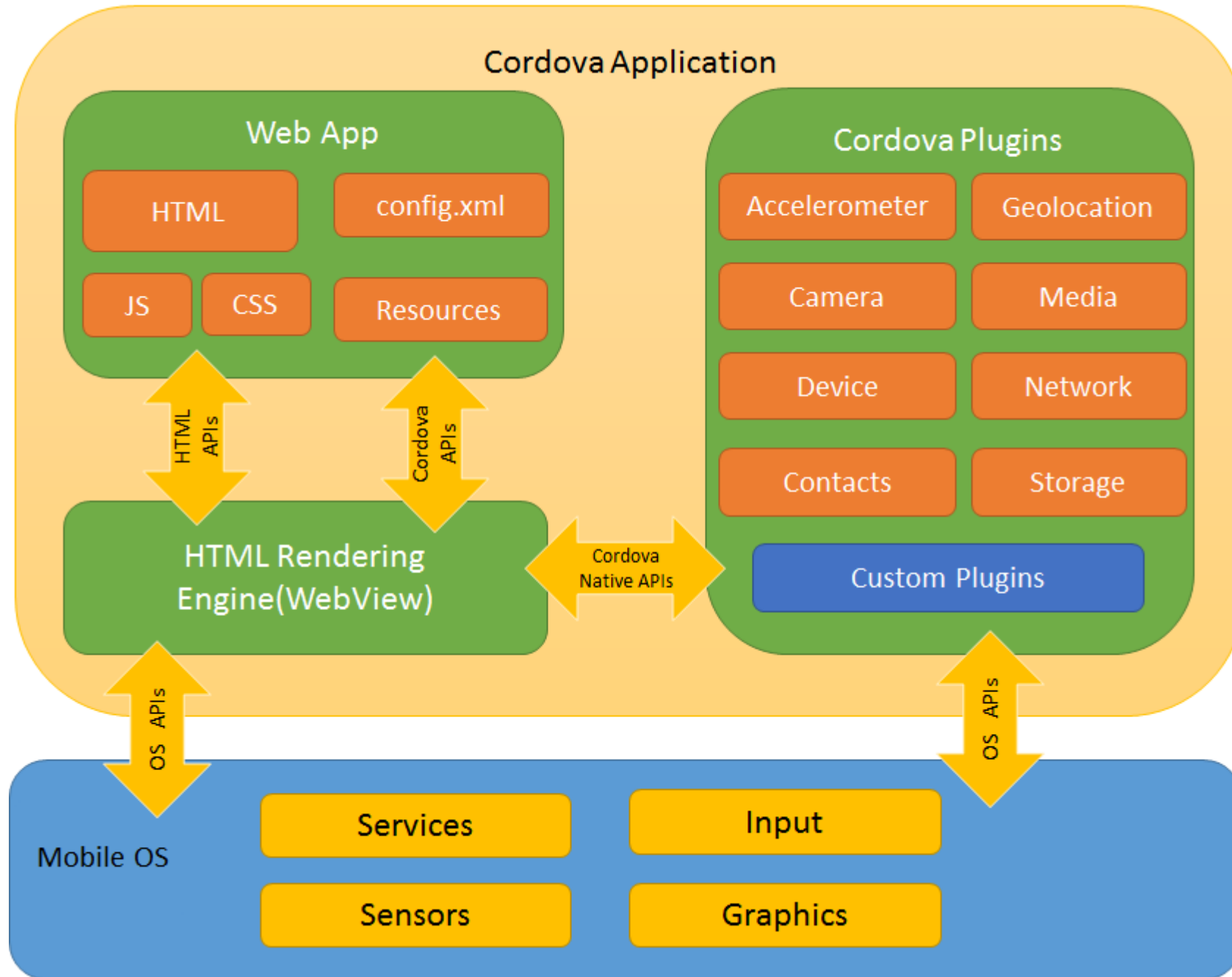
- Allows developers to create web pages that are run inside a device's browser instance called **WebView**



Cordova

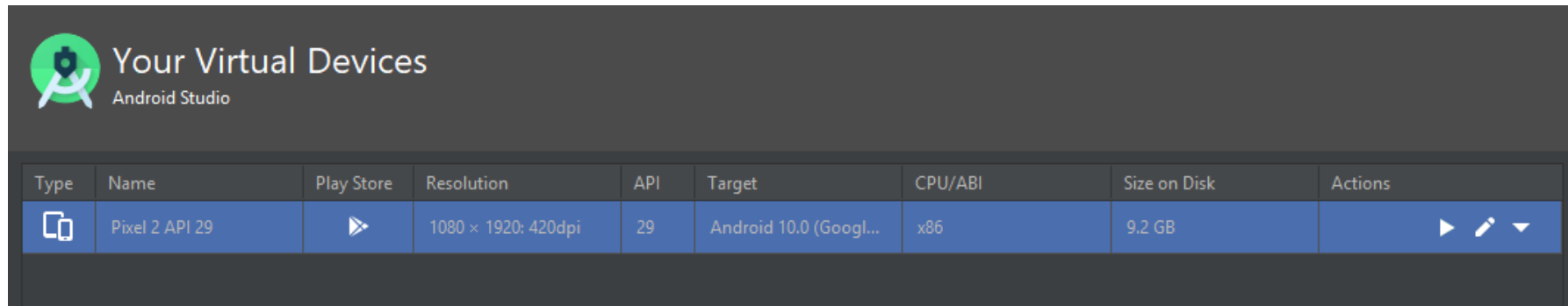







- 2008 Nitobi, a web development company from Canada created the PhoneGap framework to run web application in a mobile phone inside WebView
- 2011, Adobe bought Nitobi, and the PhoneGap framework was donated to the Apache Foundation
- Project was renamed Cordova (after the street name of Nitobi's office in Vancouver, Canada).
- Adobe ~~has~~ had some paid services such as the PhoneGap Build service
- **Adobe discontinued PhoneGap Build Services and investments in Apache Cordova on Oct 2020**

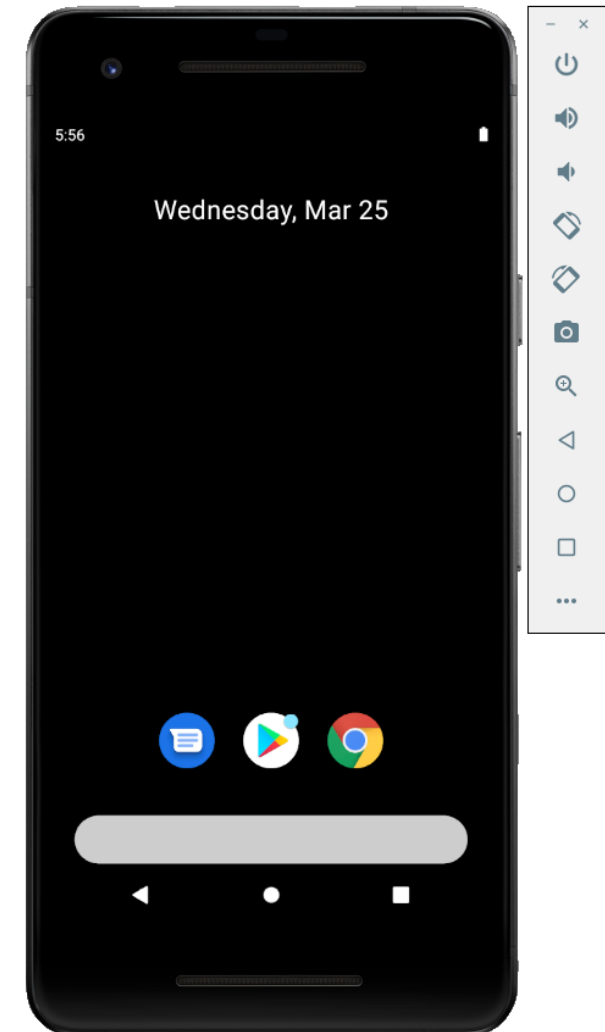


Cordova Installation and Configuration

- Follow steps in separate slide deck (include installing Android Studio)



Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Pixel 2 API 29		1080 x 1920: 420dpi	29	Android 10.0 (Googl...	x86	9.2 GB	  



Running and Debugging (Android)

- After creating a Cordova project

To test in emulator

```
cordova run android
```

To test on your phone:

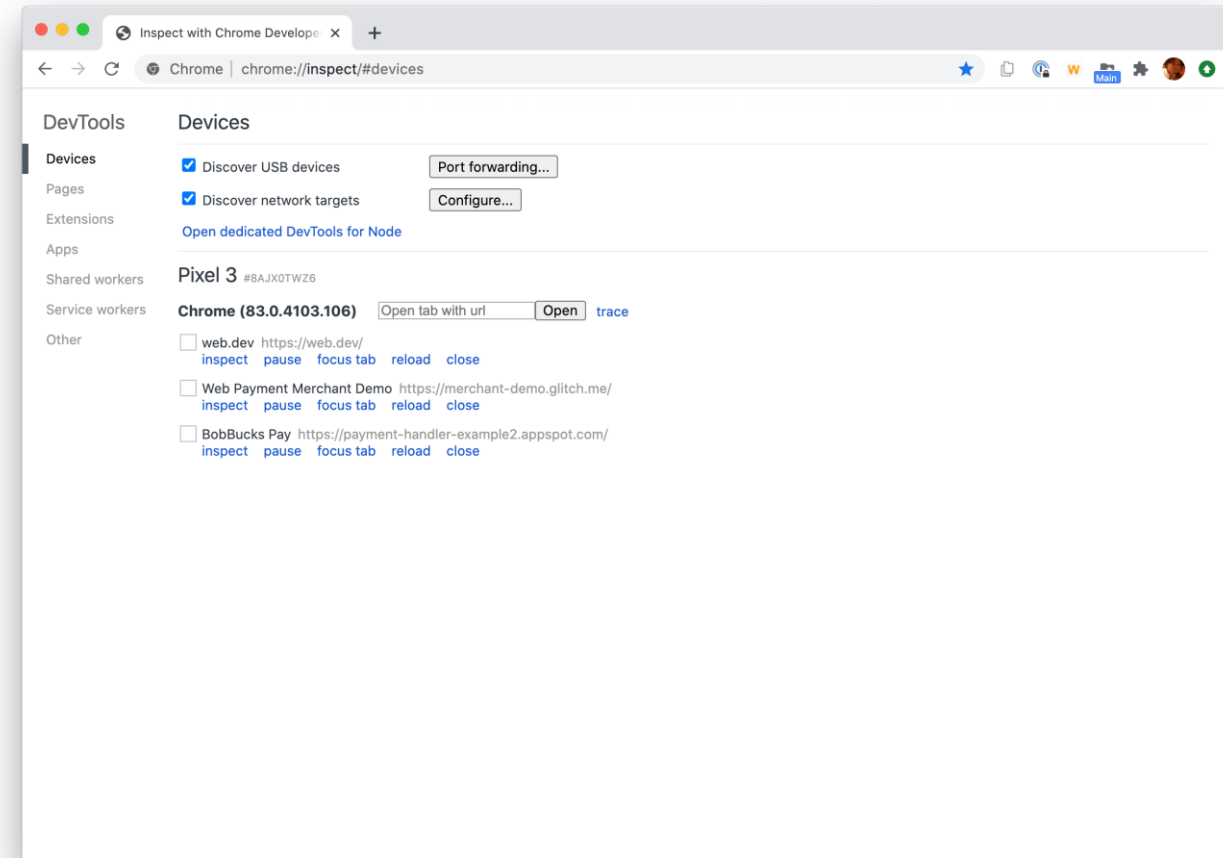
```
cordova run android --device
```

(device must be connected, [developer mode](#) enabled, [USB debugging](#) enabled)

Debugging Android Phone

- Follow this guide:
- <https://developer.chrome.com/docs/devtools/remote-debugging/>
- But remember to enable **Port Forwarding**. For instance:
- If you are running an Express server on your machine on localhost:3000, you need to add port forwarding to localhost:3000
- Then your phone can access your server and vice versa

chrome://inspect#devices



Ionic Framework

Quick intro

Ionic is...

The leading cross-platform dev solution

Powering >15% of apps in the app stores

5 Million developers worldwide

A web-first approach



<https://ionicframework.com/>

Ionic



- Open source for hybrid mobile app development.
- Originally released in 2013 and built on top of [AngularJS](#) and [Apache Cordova](#).
- Since 2019 introduced [Web Components](#), allowing the user to choose any user interface framework, such as [Angular](#), [React](#) or [Vue.js](#), as well as vanilla JS.
- Web Components make use of (i) **custom elements**, (ii) rendered by the browser in the **shadow DOM**, (iii) using **HTML templates**.
- Uses native functionality based on [Apache Cordova](#) plugins
- It is introducing an alternative to Cordova, named Capacitor

Design Guidelines



To design a user interface design you need to keep up to users expectations, which are different across different platforms.

For example, an Android application that has an iOS-style UI will probably not go over well.

It is important to respect the conventions of each platform and the various Human Interface guidelines:

- <https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/>
- <https://developer.android.com/design/>

Ionic has got you covered: it automatically adapts the UI to the platform in which it runs.

Ionic Components Overview

Output

```
<ion-img>  
<ion-badge>  
<ion-loading>  
<ion-label>  
<ion-title>  
<ion-thumbnail>  
<ion-toolbar>  
<ion-alert>  
<ion-toast>  
<ion-modal>  
...
```

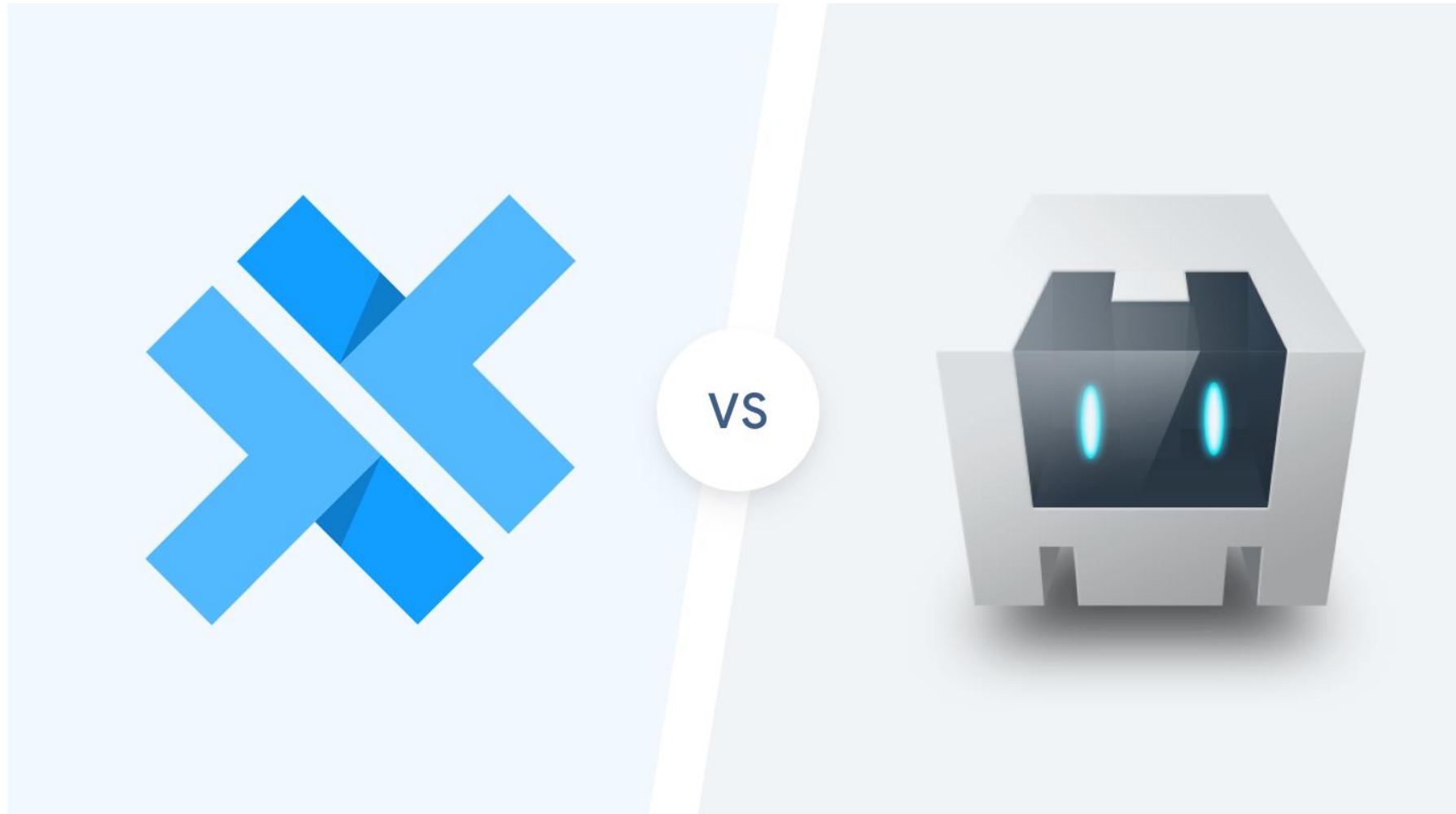
Layout

```
<ion-grid>  
<ion-row>  
<ion-col>  
<ion-list>  
<ion-card>  
<ion-infinite-scroll>  
<ion-tabs>  
...
```

Input

```
<ion-button>  
<ion-input>  
<ion-textarea>  
<ion-menu>  
<ion-select>  
<ion-datetime>  
<ion-fab>  
<ion-toggle>  
...
```

Ionic Developed Capacitor



<https://ionic.io/resources/articles/capacitor-vs-cordova-modern-hybrid-app-development>

MAY 25, 2022 • 100% ONLINE

The #1 event for web developers building for mobile

It's never been easier—or more fun—to build mobile experiences with the web. Join us for a full day of talks from experts and leaders in the web community, showing how the web is pushing the boundaries of mobile app development. Get insights on the latest web libraries, frameworks, and tools that are empowering web developers to build stunning mobile and cross-platform apps using the power of the web.

<https://ionic.io/ionicconf>

Progressive Web Apps (PWAs)



Progressive Web Apps (PWAs)



- **Reliable:** Through caching it is available offline
- **Engaging:** Has access to native features (some of them)
- **Installable:** create a link/shortcut that looks like a native app/program on devices

PWAs vs Native App vs Traditional Web Apps

	Features	Reach
Native Apps	Full access to OS and device	Limited , Top 3 apps win
Traditional Web Apps	Highly limited access to OS	High reach
Progressive Web Apps	Good access to OS and device	High reach

Avg. mobile user spend on 80% of time on top 3 mobile apps and install 0 apps.

PWAs Stats: A New Trend


- At the end of 2020, approximately **1% of websites included a Service Worker, and 2.2%** had an installable Web App Manifest file.
- In 2021, [18% of page loads](#) have a Service Worker controlling them.
- It's also good to compare the platform's growth: **Service Workers' usage has increased 38% in the last year (2021).**
- Number of origins with PWAs has grown **170%** in 2020

Can I Use PWAs?

Add to home screen (A2HS) - WD


The ability for a user to "install" a website and use it as if it was a natively installed app. To enable this behaviour, a website must serve a valid Web App Manifest and load its assets through a [Service Worker](#).

Usage

% of **all users**  ?

Global

73.01% + 14.71% = 87.72%

Current aligned Usage relative Date relative Filtered All 

IE	Edge [*]	Firefox	Chrome	Safari	Opera	Safari on iOS [*]	Opera Mini [*]	Android Browser [*]	Opera Mobile [*]	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser	Baidu Browser	KaiOS Browser
	12-16	2-75														
	¹ 17-18	³ 76-85	4-38			3.2-11.2										
6-10	79-100	86-99	39-100	3.1-15.3	10-85	² 11.3-15.3		2.1-4.4.4	12-12.1				4-15.0			
11	101	100	101	15.4	86	² 15.4	all	101	64	101	100	12.12	16.0	10.4	7.12	2.5
		101-102	102-104	TP	87											

<https://caniuse.com/web-app-manifest>

Can I Use PWAs? (2021)

Add to home screen (A2HS) - WD

Usage

% of all users ?

Global

73.48% + 13.29% = 86.77%

The ability for a user to "install" a website and use it as if it was a natively installed app. To enable this behaviour, a website must serve a valid Web App Manifest and load it's assets through a [Service Worker](#).

Current aligned Usage relative Date relative Filtered All

IE	Edge *	Firefox	Chrome	Safari	Opera	Safari on iOS *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser	Baidu Browser	KaiOS Browser
	12-16	2-75														
	¹ 17-18	³ 76-85	4-38			3.2-11.2										
6-10	79-89	86-87	39-89	3.1-14	10-74	² 11.3-14.4		2.1-4.4.4	12-12.1				4-13.0			
11	90	88	90	14.1	75	² 14.5	all	90	62	90	87	12.12	14.0	10.4	7.12	2.5
		89-90	91-93	TP												

<https://caniuse.com/web-app-manifest>

PWAs Building Blocks



Manifest

Makes App installable

Service Worker

Offline caching, Web Push

Other Principles and API

Geolocation, Responsive
Design, Media API

Service Workers

Service workers are JS processes ~~running~~ lurking in the background waiting for browser events

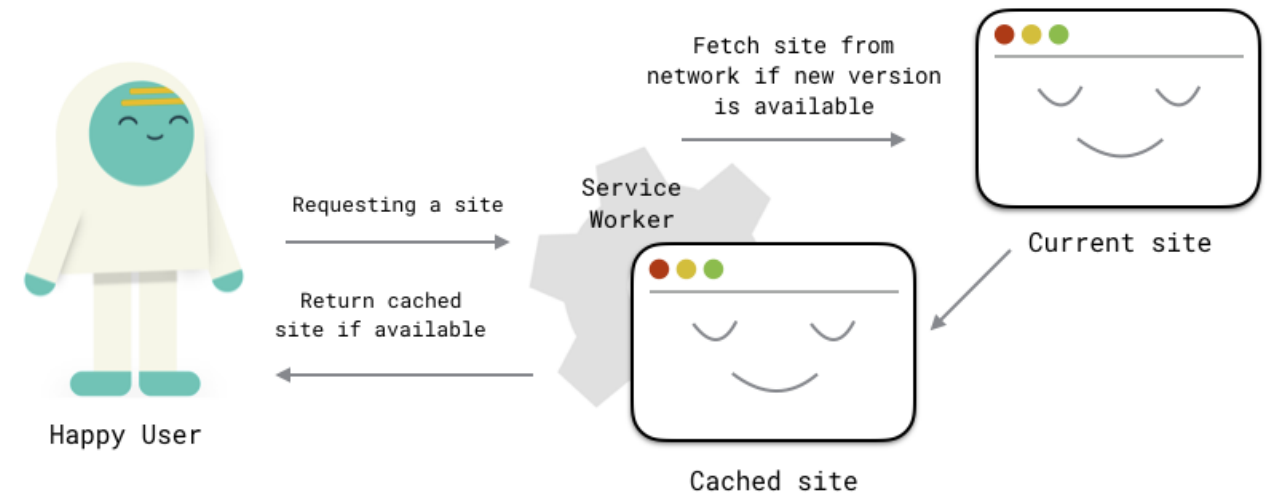
They are NOT associated with any tab

They require **HTTPS** to be executed

They are the foundation of **Progressive Web Apps (PWA)**, which can be installed on any device providing mobile-like experience

You may have some installed on your browser:

`chrome://inspect/#service-workers`



[Image source](#)

- <https://jakearchibald.github.io/isserviceworkerready/>
- <https://w3c.github.io/ServiceWorker/>

Events for Service Workers

Event

Source

Fetch

Browser or Page-related JavaScript initiates a Fetch (Http request)

Push Notifications

Service Worker receives Web Push Notification (from Server)

Notification Interaction

User interacts with displayed Notification

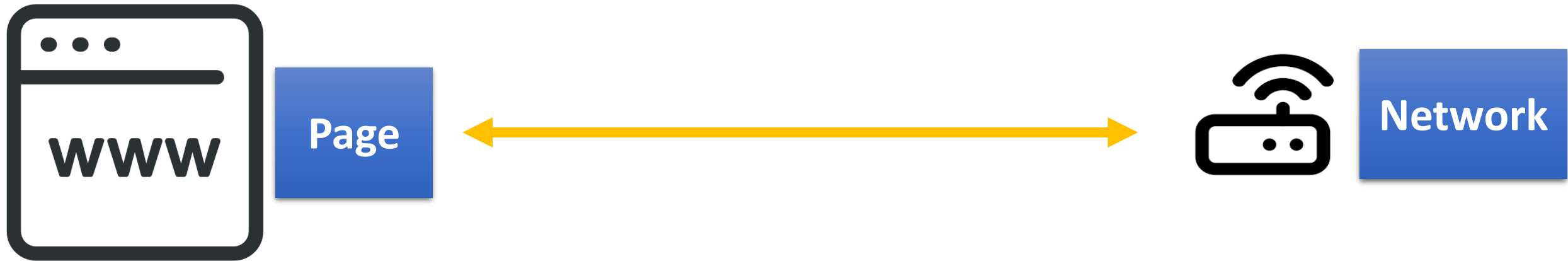
Background Sync

Service Worker receives Background Sync Event (e.g. Internet Connection was restored)

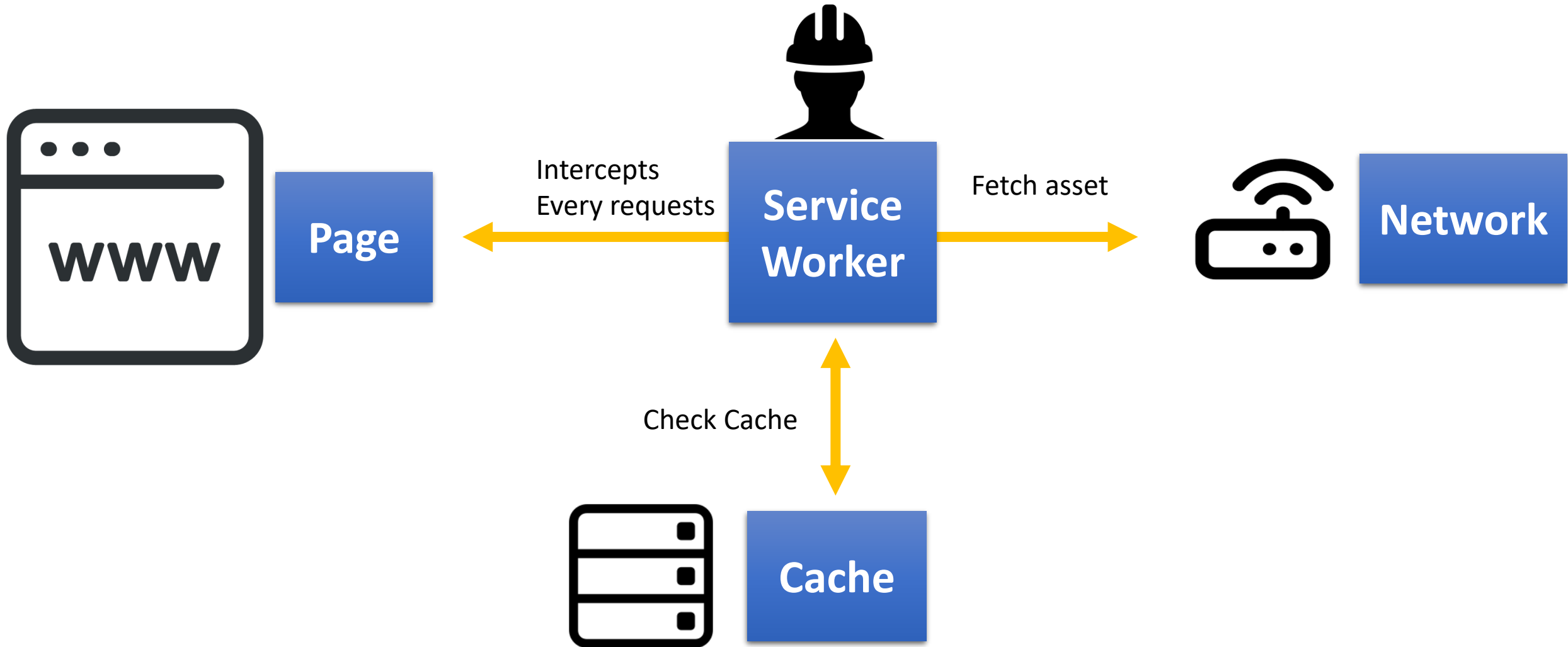
Service Worker Lifecycle

Service Worker Phase changes

Idea behind PWA Caching



Idea behind PWA Caching



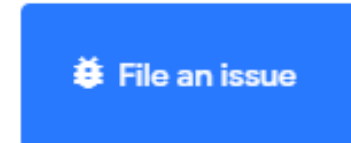
Check How Much PWA Your Site Is

Lighthouse  <https://developers.google.com/web/tools/lighthouse>

Lighthouse is an [open-source](#), automated tool for improving the quality of web pages. You can run it against any web page, public or requiring authentication. It has audits for performance, accessibility, progressive web apps, SEO and more.

You can run Lighthouse in Chrome DevTools, from the command line, or as a Node module. You give Lighthouse a URL to audit, it runs a series of audits against the page, and then it generates a report on how well the page did. From there, use the failing audits as indicators on how to improve the page. Each audit has a reference doc explaining why the audit is important, as well as how to fix it.

You can also use [Lighthouse CI](#) to prevent regressions on your sites.



For instance, you may check: <https://app.ft.com/stream/home>

Some References



<https://medium.com/@firt/progressive-web-apps-in-2020-c15018c9931c>

<https://firt.dev/pwa-2021/>

<https://brainhub.eu/library/is-pwa-the-future/>

<https://web.dev/customize-install/>

<https://web.dev/add-manifest/>

<chrome://apps/>

Thank You Slide 😊



HTML, JS, CSS

Bootstrap

jQuery

Node.JS

Express

Heroku

Chrome Extensions

~~nodeGame~~

Cordova

Ionic Framework

PWAs

...