JAVASCRIPT IS COMING TO EAT YOU

Citizen Tim

January 2004

The Dean Scream

How we build the web

1994 to 2004

The Renaissance

2004 to 2014

The Industrial Revolution

2014 to Today

The Technological Revolution

2004 to 2014

The Industrial Revolution

- Rise and domination of the CMS
- Drupal, Joomla, and WordPress
- SaaS/Commercial CMS
- LAMP stack
- The Cloud
- Ajax/Jquery (Gmail)

2014 to Today

The Technological Revolution

- React/Angular/Vue
- Decoupled CMS
- Node.js and JS libraries
- JAMstack and APIs
- Managed services
- Serverless

Most of us are still living in 2014.

And it is a pretty great place to be.

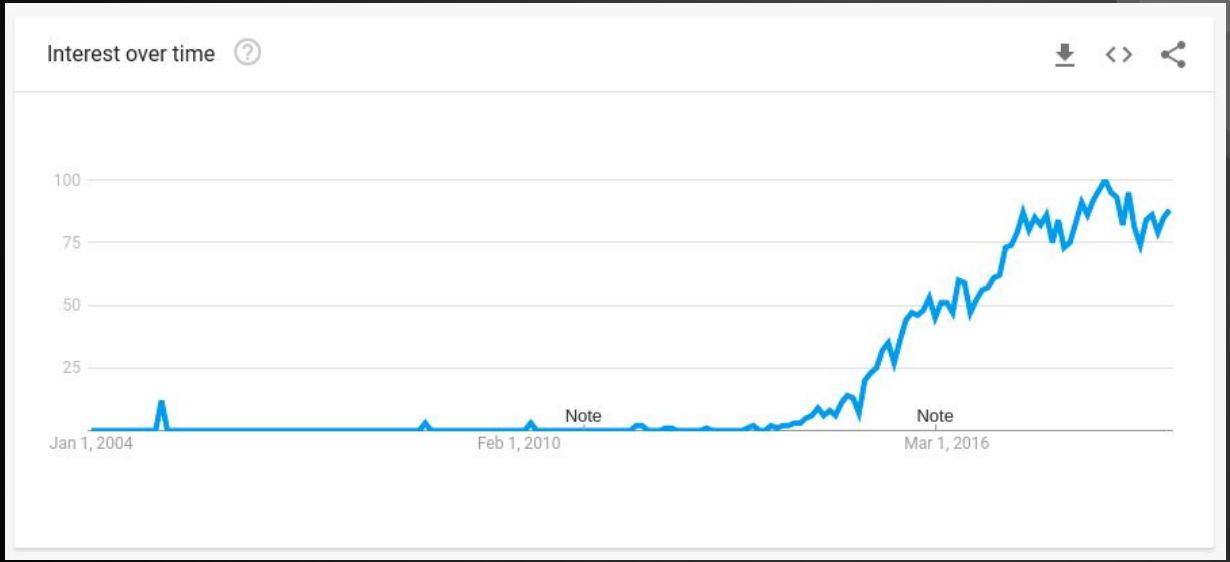
Winning all the things.

- We enjoy a multi-billion dollar global marketplace
- Innovation continues at great speed
- Drupal or WordPress are still an obvious choice for most websites

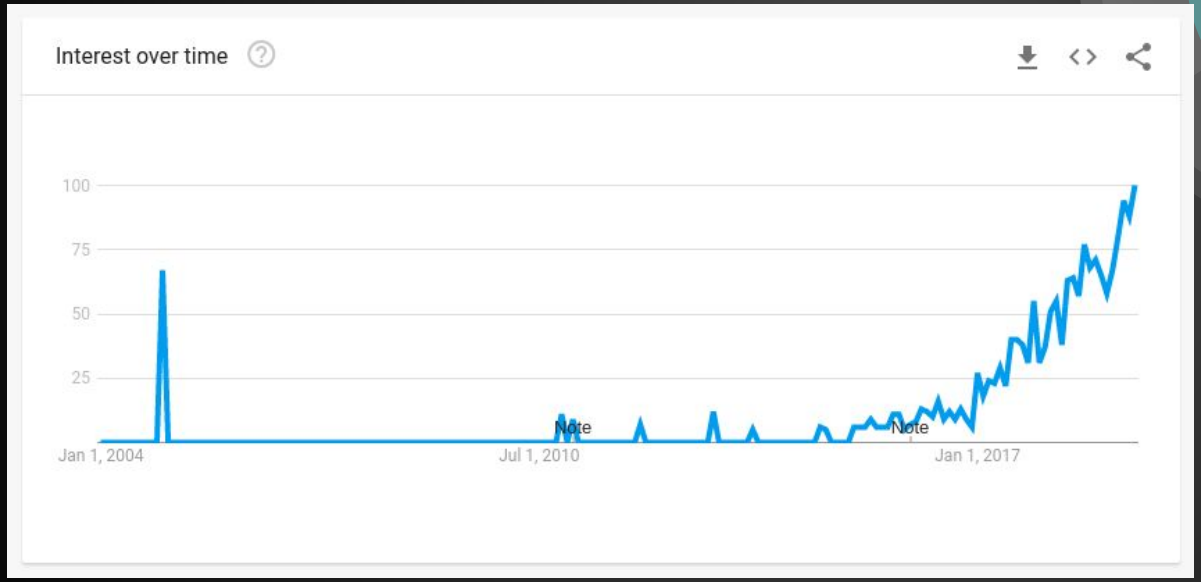
Come, Watson, come!

The game is afoot.

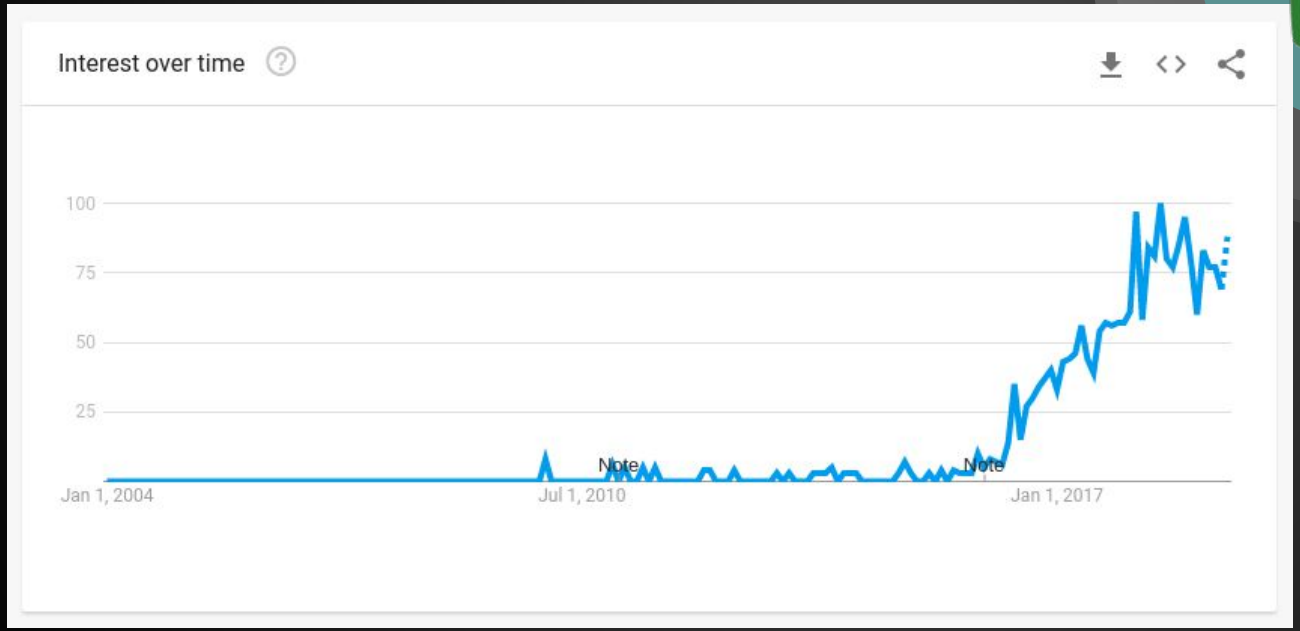
REACT & ANGULAR
2004 to present



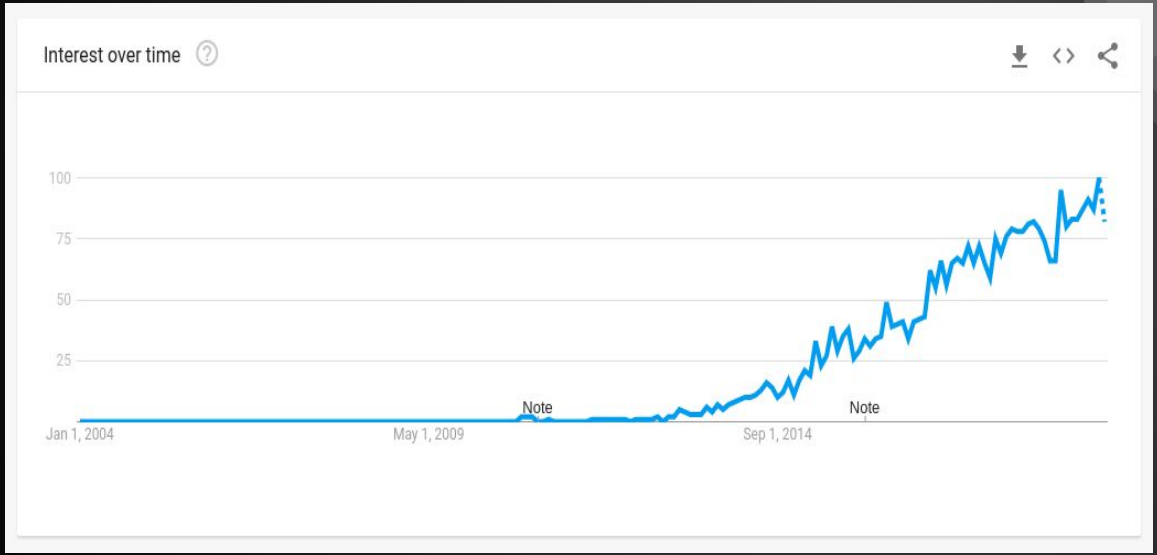
HEADLESS CMS
2004 to present



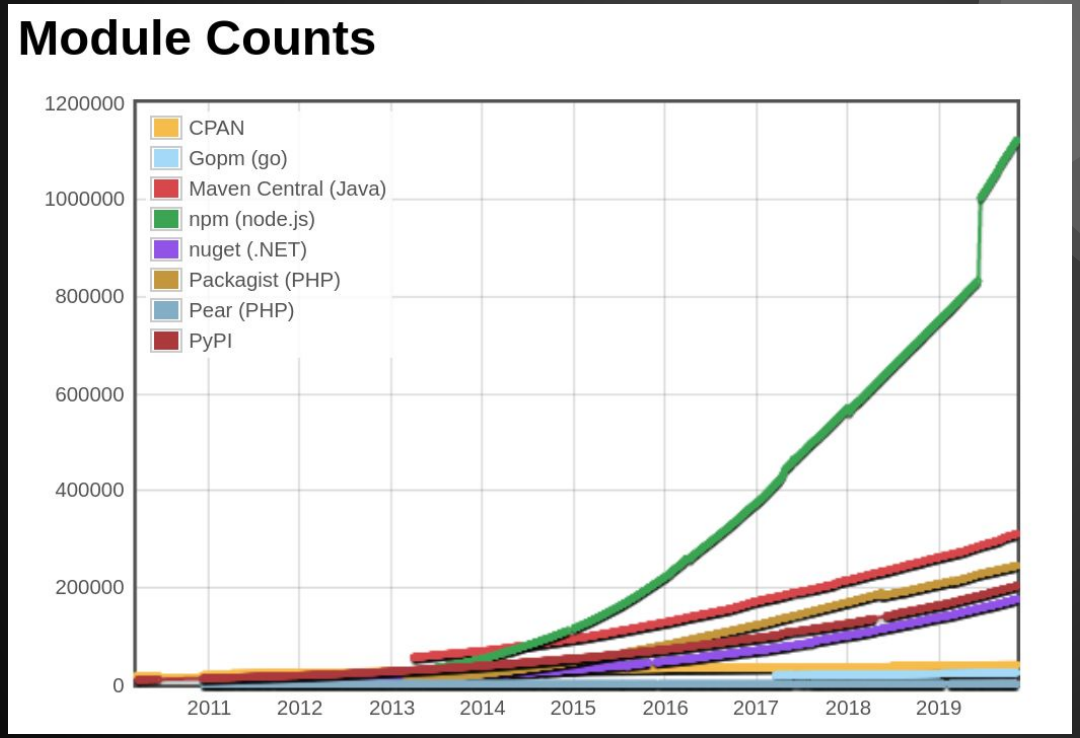
PROGRESSIVE WEB APPS
2004 to present



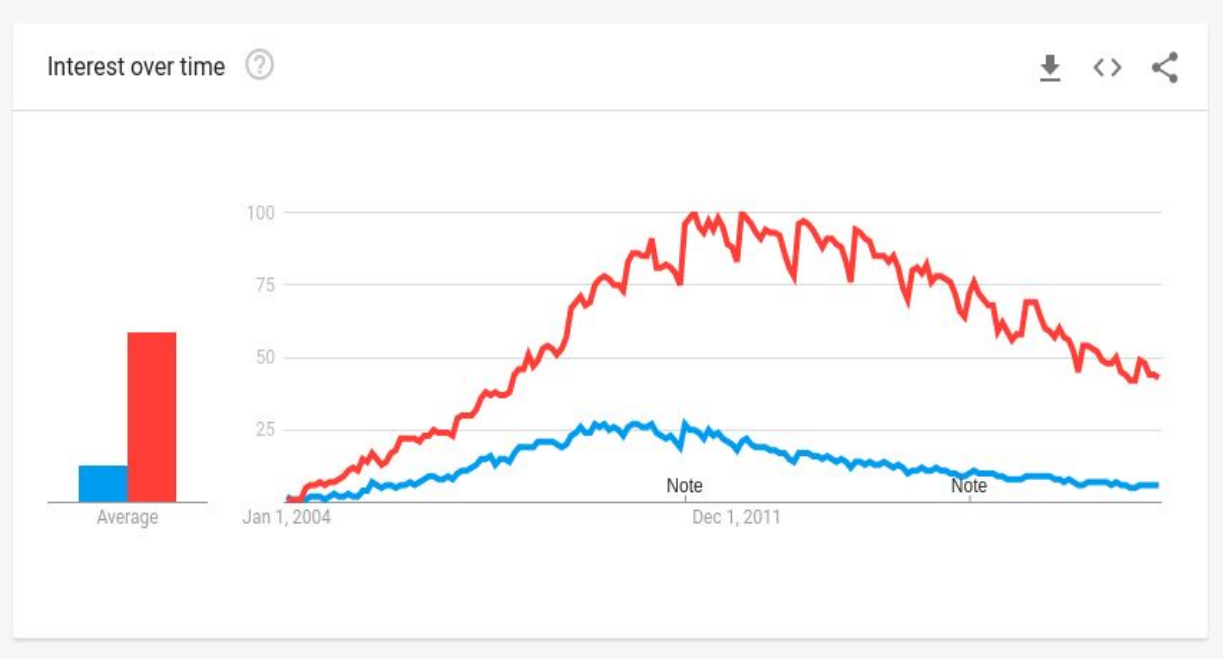
FULL STACK DEVELOPER
2004 to present



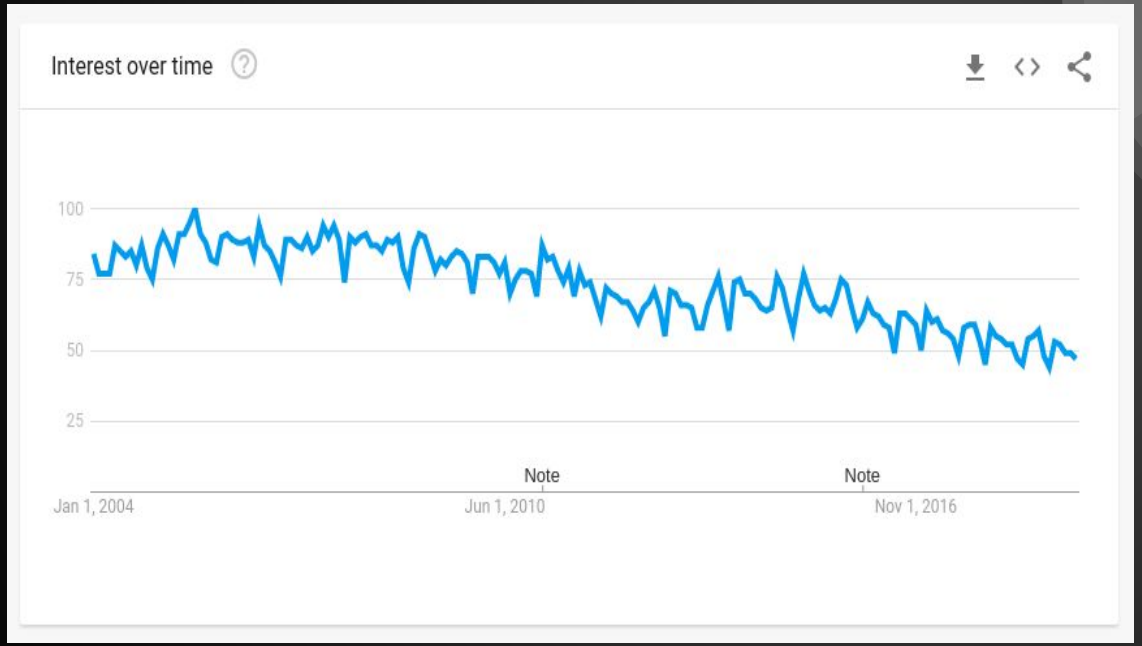
Popular Language Modules
All time



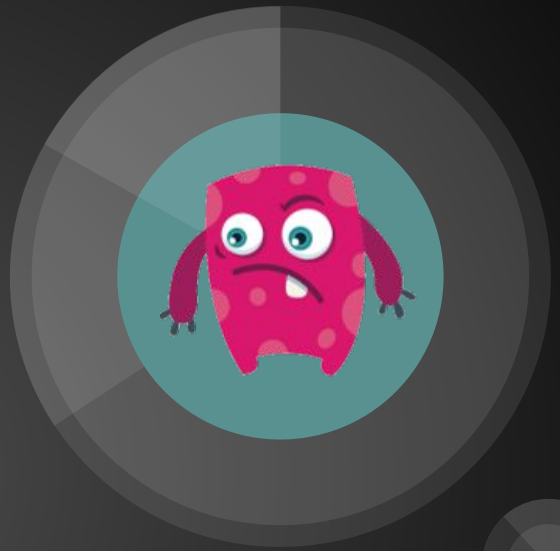
DRUPAL & WORDPRESS
2004 to present



CMS
2004 to present



“We would prefer Python, Ruby, and/or Javascript based applications. **We would steer clear of Drupal.**”



Jesus Manuel Olivas Retweeted

Nader Dabit
@dabit3

There is a massive paradigm shift happening right now. The front & back end are rapidly moving closer & closer together, assisted by the rise of managed services & serverless tech.

Those who acknowledge & take advantage of this will be the biggest winners in the coming years.

11:21 AM · 21 Apr 19 · Twitter Web Client

129 Retweets **464** Likes

Important Realization #1:

We are rapidly changing how and what we build.

Option A

Stay the course.

Option A

"Nobody ever got fired for buying IBM."

Option A

Option B

Learn and adapt.

Monolithic CMS

vs

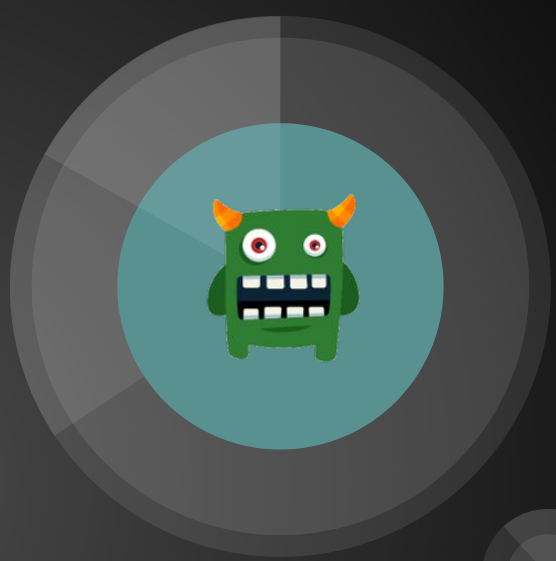
DECOUPLED FRONTEND

API INTEGRATIONS

MANAGED SERVICES

MICROSERVICES

SERVERLESS



Monolithic CMS

vs

DECOUPLED FRONTEND

API INTEGRATIONS

MANAGED SERVICES

MICROSERVICES

SERVERLESS

JavaScript!

Monolithic CMS

Content Management
Landing pages, news, events

Custom applications
Databases, tools

Integrations
APIs, migrations, importers, exporters

Membership/CiviCRM
Customizations, roles, reporting

Web server
Database server
CDN

Frontend website!
HTML, JS, Twig

Customization
Customize all the things

Commerce
Sell, sell, sell!

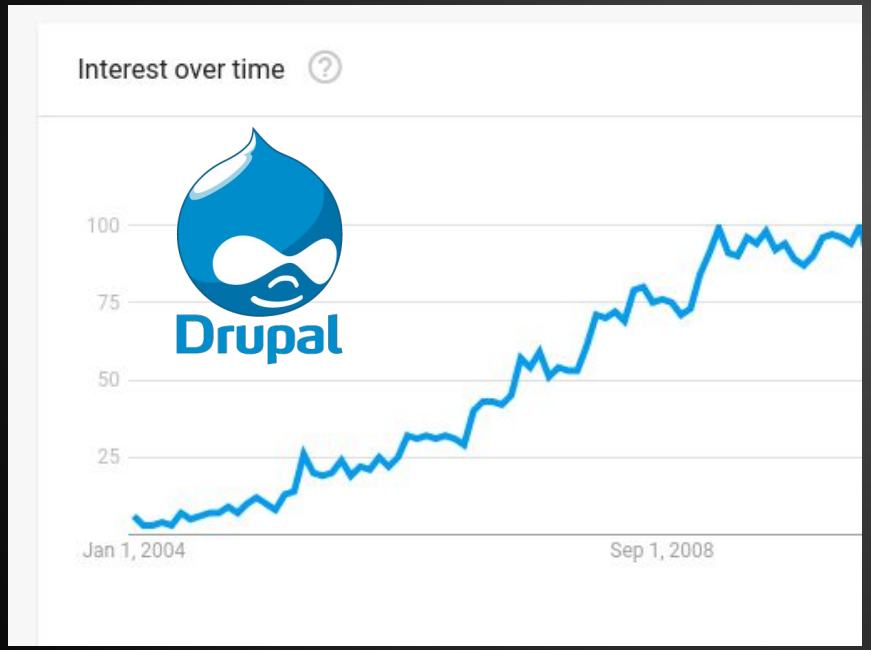
Search
Search API, Solr

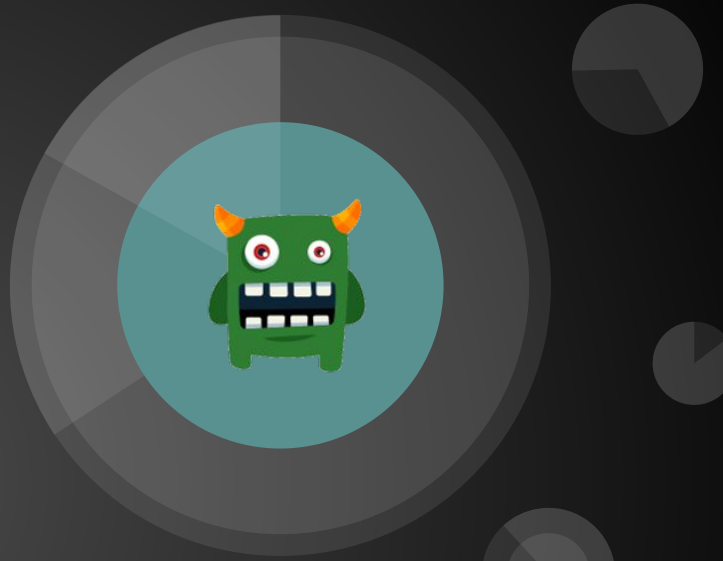
Forums and galleries oh my
All the things!

The Golden Hammer

“I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail.”

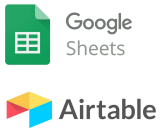
Once upon a time you dressed so fine.



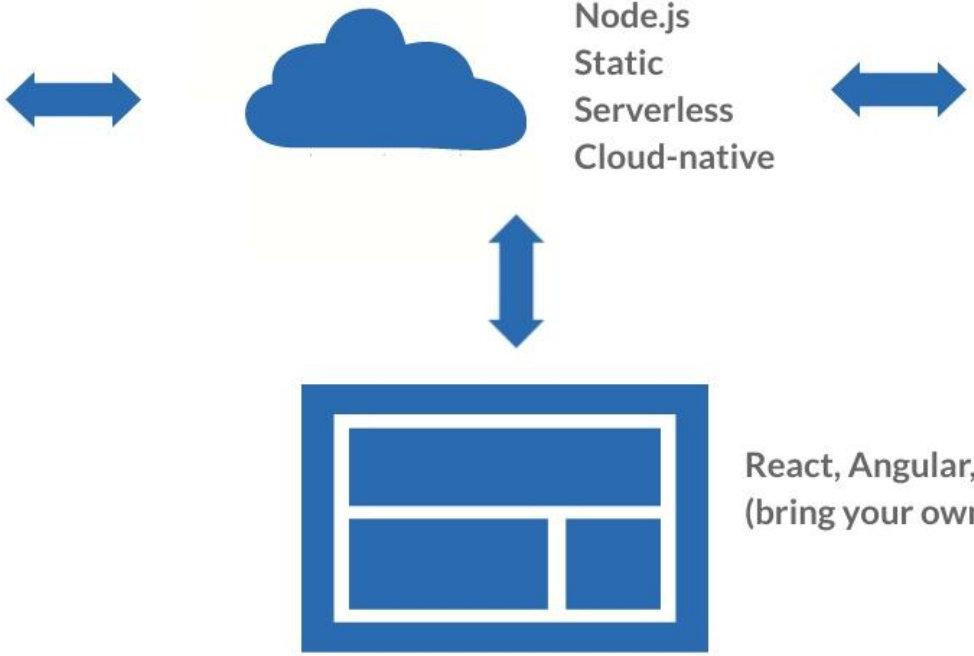


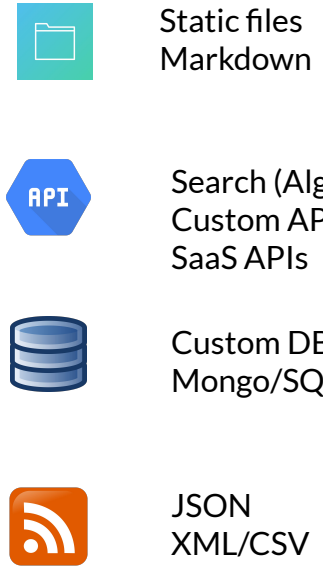
Modern Web


contentful


Google
Sheets
Airtable


shopify
salesforce
Auth0

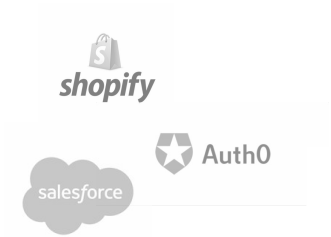

Node.js
Static
Serverless
Cloud-native
React, Angular, Vue
(bring your own!)


Static files
Markdown
API
Search (Algolia)
Custom APIs
SaaS APIs
Custom DBs
Mongo/SQLite
JSON
XML/CSV

Important Realization #2:

We are no longer building websites.

Everything else has been solved and commoditized.



So what then is our job?

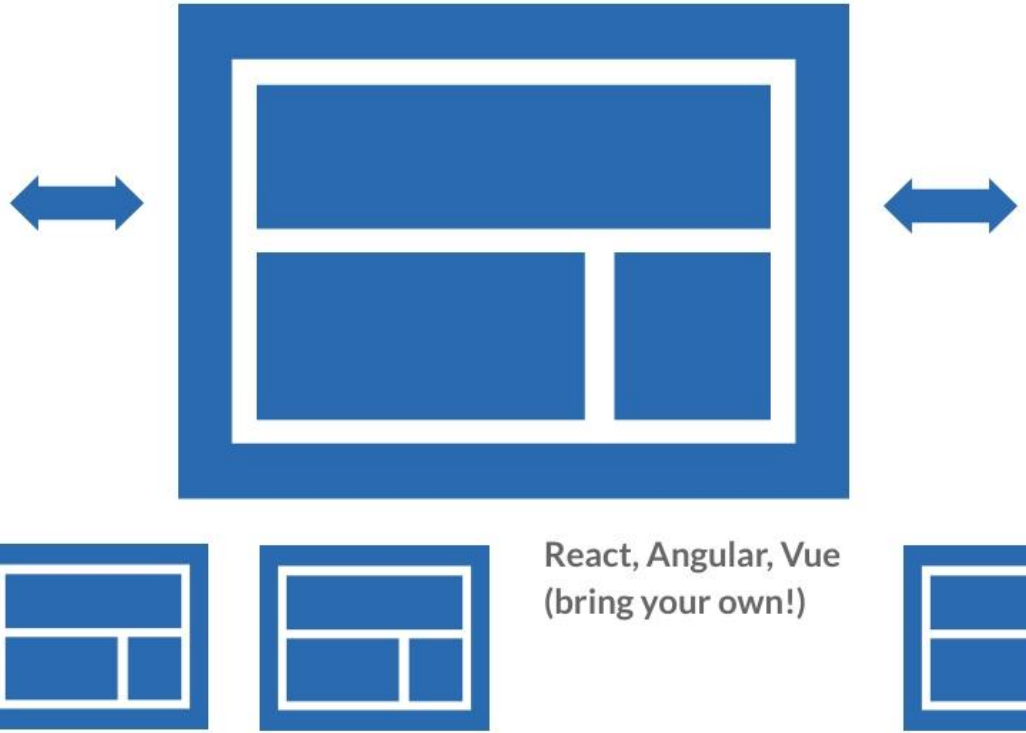
Static files
Markdown

Custom APIs
SaaS APIs

Custom DBs
Mongo/SQLite

JSON
XML/CSV

This is what really matters, right?



Static files
Markdown

Custom APIs
SaaS APIs

Custom DBs
Mongo/SQLite

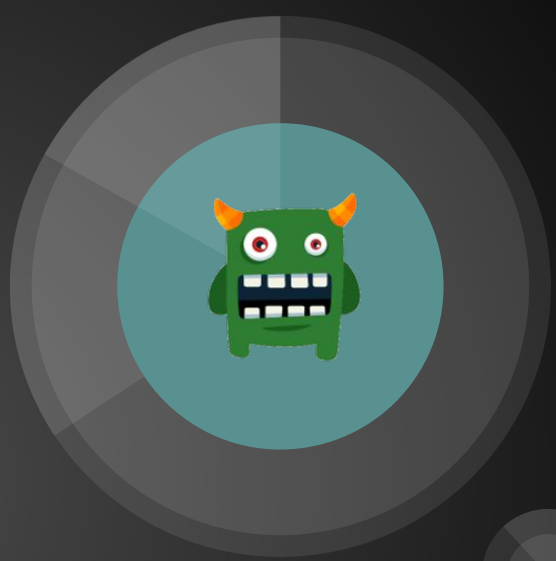
JSON
XML/CSV


STILL SKEPTICAL
imgflip.com

Can’t Drupal also provide a good user experience?

Why would I give up all of its out-of-box features?

Why add another layer of complication and expense?



From the dawn of time, web developers hath sought to build better websites, and to build them faster.

What if there is a better way to build the web?

	TRADITIONAL CMS	MODERN WEB
SPEED AND PERFORMANCE	✗	✓
FLEXIBILITY	✗	✓
USER EXPERIENCE	✗	✓
DEVELOPER EXPERIENCE	✗	✓
OVERALL COST	✗	✓

	TRADITIONAL CMS	MODERN WEB
SPEED AND PERFORMANCE		
FLEXIBILITY		
USER EXPERIENCE		
DEVELOPER EXPERIENCE	✗	✓
OVERALL COST	✗	✓

- Critical Rendering
- Code splitting and pre-renders
- Server Side Rendering (SSR)
- Node (Chrome V8)
- Static Rendering

	TRADITIONAL CMS	MODERN WEB
SPEED AND PERFORMANCE	✗	✓
FLEXIBILITY		
USER EXPERIENCE		
DEVELOPER EXPERIENCE		
OVERALL COST	✗	✓

- Data agnostic
- Frontend agnostic
- Right tool for the right job

	TRADITIONAL CMS	MODERN WEB
SPEED AND PERFORMANCE	✗	✓
FLEXIBILITY	✗	✓
USER EXPERIENCE		
DEVELOPER EXPERIENCE		
OVERALL COST		

- Faster
- More effective
- More delightful

	TRADITIONAL CMS	MODERN WEB
SPEED AND PERFORMANCE	✗	✓
FLEXIBILITY		
USER EXPERIENCE		
DEVELOPER EXPERIENCE		
OVERALL COST	✗	✓

- One stack = good feelings
- Faster and more fun
- Components/NPM/hot reloading

	TRADITIONAL CMS	MODERN WEB
SPEED AND PERFORMANCE	✗	✓
FLEXIBILITY	✗	✓
USER EXPERIENCE		
DEVELOPER EXPERIENCE		
OVERALL COST		

- Development
- Support and maintenance
- Hosting and infrastructure


WINNING
ALL THE THINGS!
imgflip.com

Important Realization #3:

You probably still need a CMS.

24 Headless CMS That Should Be On Your Radar in 2019

CMSWire.com

- Butter CMS
- Contentstack
- Contentful
- dotCMS
- Mura
- Cloud CMS
- Cockpit CMS
- Core dna
- Craft CMS
- Zesty.io
- Directus
- StoryblokWe
- GraphQL CMS
- Gentics Mesh
- Cosmic JS
- Kentico Cloud
- Prismic.io
- Quintype
- Sanity.io
- Scrivito
- Squidex
- DNN Evoq Content
- Strapi
- Superdesk

12:23 Headless CMS company ...
From techcrunch.com – delivered b

Headless CMS company Strapi raises $4 million

Romain Dillet

@romaindillet / 2 weeks ago

12:24 Contentstack raises $31....
From techcrunch.com – delivered b

Contentstack raises $31.5M Series A round for its headless CMS platform

12:34 Contentful raises $33.5M...
From techcrunch.com – delivered b

Contentful raises $33.5M for its headless CMS platform

Frederic Lardinois

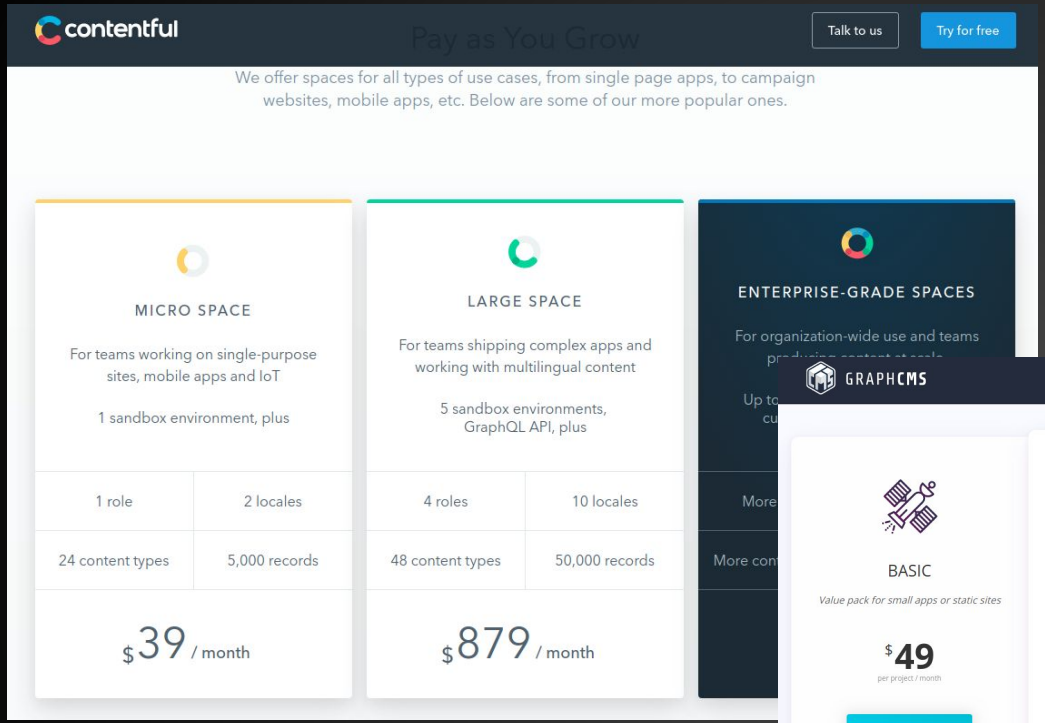
@fredericl / 11 months ago

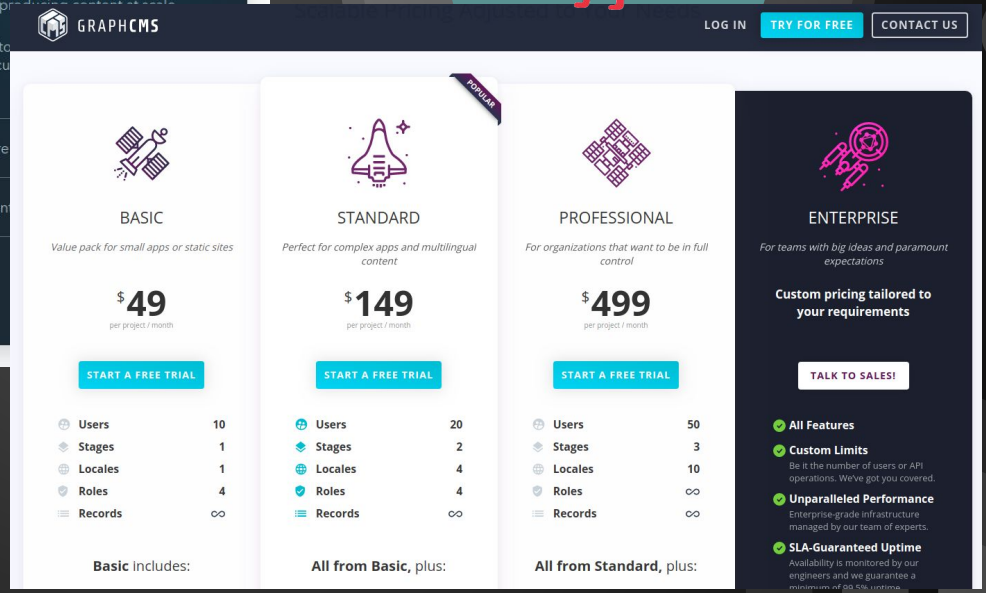
12:27 Webiny announces $348...
From techcrunch.com – delivered b

Webiny announces $348K seed to build open-source serverless CMS

Ron Miller

@ron_miller / 3 days ago





- Plus a million free tiers
- And plenty of open source

Drupal

State of Drupal presentation (May 2016)

May 12, 2016

51 sec read time

Permalink

Drupal

State of Drupal

DrupalCon

New Orleans

DrupalCon New Orleans comes at an important time in the history of Drupal. Now that Drupal 8 has launched, we have a lot of work to do to accelerate Drupal 8's adoption as well as plan what is next.

In my keynote presentation, I shared my thoughts on where we should focus our efforts in order for Drupal to continue its path to become the leading platform for assembling the world's best digital experiences.

Based on recent survey data, I proposed key initiatives for Drupal, as well as shared my vision for building cross-channel customer experiences that span various devices, including conversational technologies like Amazon Echo.

DRIES BUYTAERT ARCHIVE

The best JSON:API implementation in existence

The JSON:API module for Drupal is almost certainly the most feature-complete and easiest-to-use JSON:API implementation in existence.

The Drupal JSON:API implementation supports *every* feature of the JSON:API 1.0 specification out-of-the-box. Every Drupal entity (a *resource object* in JSON:API terminology) is automatically made available through JSON:API. Existing access controls for both reading and writing are respected. Both translations and revisions of entities are also made available. Furthermore, querying entities (*filtering resource collections* in JSON:API terminology) is possible without any configuration (e.g. setting up a "Drupal View"), which means front-end developers can get started on their work right away.

/jsonapi

Not secure | dev-mtcv.pantheonsite.io/jsonapi/node/page

Apps Office Gmail Github

```
// 20190606144854
// http://dev-mtcv.pantheonsite.io/jsonapi/node/page

{
  "jsonapi": {
    "version": "1.0",
    "meta": {
      "links": {
        "self": {
          "href": "http://jsonapi.org/format/1.0/"
        }
      }
    }
  },
  "data": [
    {
      "type": "node--page",
      "id": "4e4028d5-8b4e-458b-8320-22fb2a01771a",
      "attributes": {
        "drupal_internal__nid": 2,
        "drupal_internal__vid": 52,
        "langcode": "en",
        "revision_timestamp": "2019-05-28T20:27:28+00:00",
        "revision_log": null,
        "status": true,
        "title": "Safe Browsing",
        "created": "2019-05-12T17:41:59+00:00",
        "changed": "2019-05-28T20:27:28+00:00",
        "promote": false,
        "sticky": false,
        "default_langcode": true,
        "revision_translation_affected": null,
        "path": {
          "alias": "/safe-browsing",
          "pid": 1,
          "langcode": "en"
        }
      },
```

/jsonapi

Important Realization #4:

Drupal solved most of the hard problems 10 years ago or more.

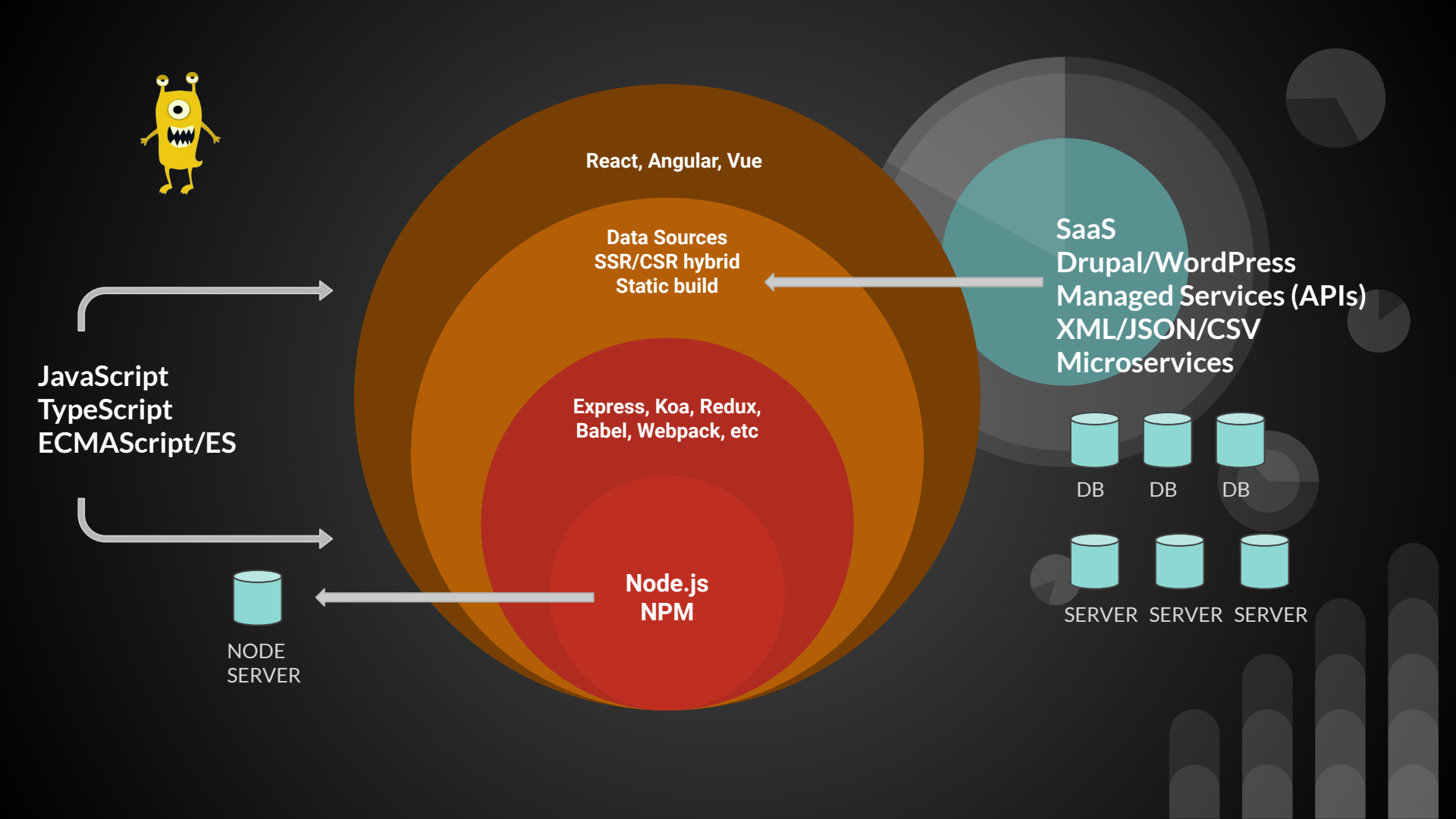
- Content architecture and entities
- Nodes and taxonomies
- Entity References
- Workflow and Permissions
- Content scheduling
- Media and File Handling
- Battle tested goodness

Important Realization #5:

Drupal is still the bomb.

- Free and Open source
- API-first
- No “record” limitations
- No “rate” limitations
- Arguably better at the fundamentals of CMS
- Right tool for this job


LET'S
DO THIS!
quickmeme.com


React, Angular, Vue
Data Sources
SSR/CSR hybrid
Static build
Express, Koa, Redux,
Babel, Webpack, etc
Node.js
NPM
JavaScript
TypeScript
ECMAScript/ES
NODE
SERVER
SaaS
Drupal/WordPress
Managed Services (APIs)
XML/JSON/CSV
Microservices
DB
DB
DB
SERVER
SERVER
SERVER

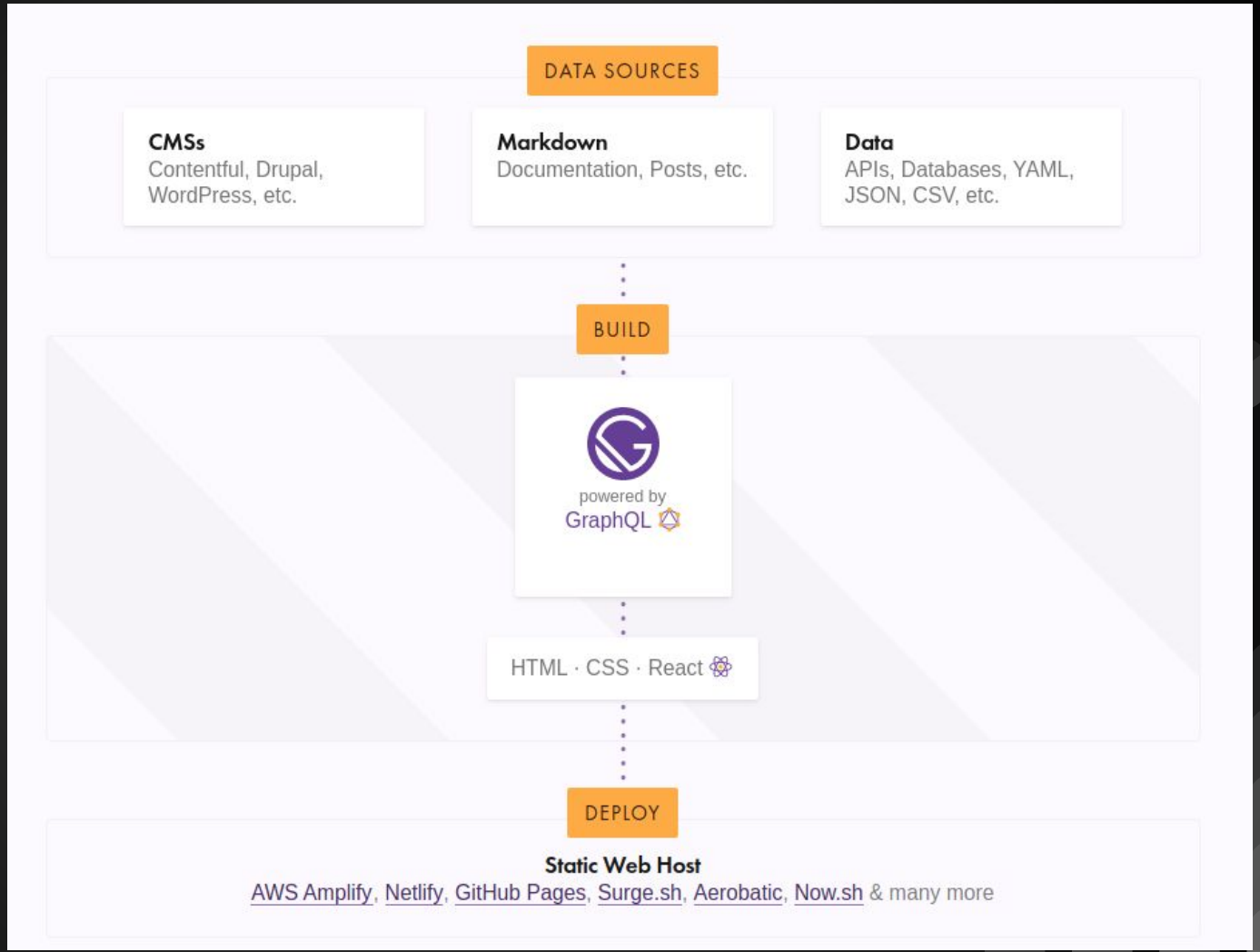
React

A JavaScript library for building user interfaces

Get Started

Take the Tutorial >

Meet Gatsby!


DATA SOURCES
CMSs
Contentful, Drupal, WordPress, etc.
Markdown
Documentation, Posts, etc.
Data
APIs, Databases, YAML, JSON, CSV, etc.
BUILD
powered by
GraphQL
HTML · CSS · React
DEPLOY
Static Web Host
AWS Amplify, Netlify, GitHub Pages, Surge.sh, Aerobatic, Now.sh & many more

NEXT.js 8

The React Framework for

Server-Rendered Apps

See Showcase

License: MIT View Docs GitHub

“I would never build anything in React **without choosing Next or Gatsby.**”

Wes Bos, Feb. 2019

JAMSTACK

Best Practices Examples Resources Community Diversity and I

JAMstack: noun \'jam-stak'\

Modern web development architecture based on client-side JavaScript, reusable APIs, and prebuilt Markup.

The World is your Playground:

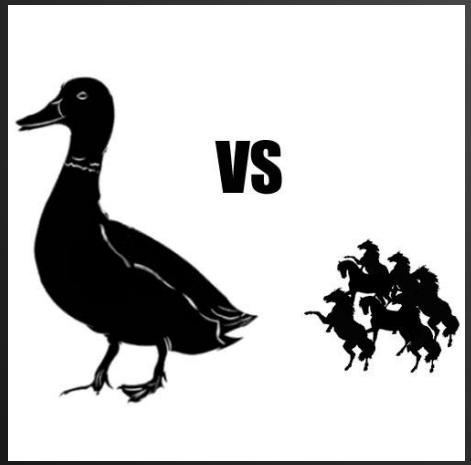
- Netlify, Now, AWS Amplify
- Codepen/Sandbox/Glitch
- One click demos
- Free tiers for miles
- Excellent documentation

Important Realization #6:

Things that were super hard 5 years ago are now easy (and even fun!)

Important Realization #7:

Would you rather fight 100 duck sized horses, or one horse sized duck?



8:32 45%

Tweet

Kyle Mathews
@kylemathews

Websites are just derived data y'all

7:29 PM · 04 May 19 · Twitter Web Client

14 Likes

Thank you!

Citizen Tim

Electric Citizen | October 2019