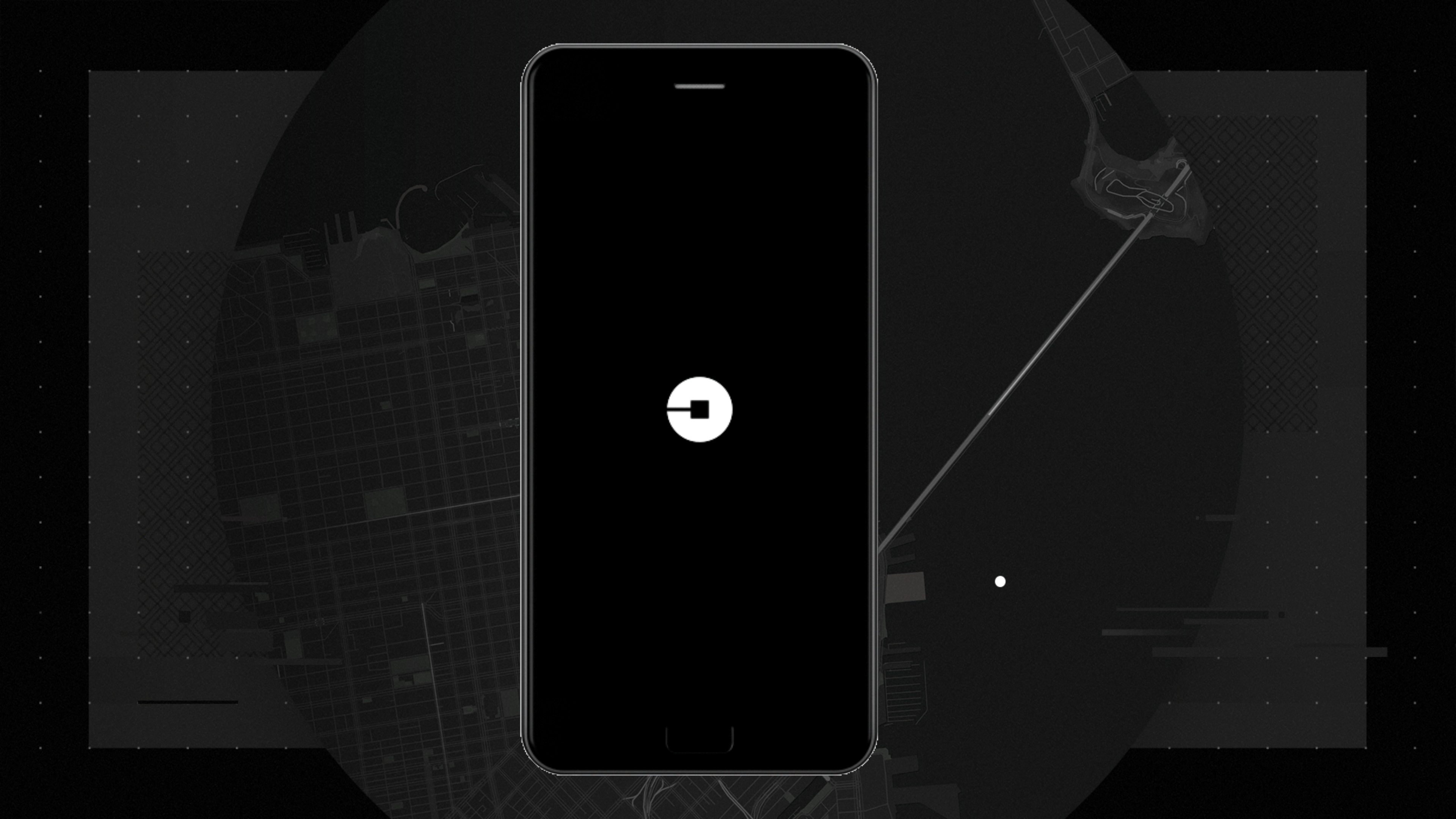
goto; conference

JavaScript @ Uber

UBER

Dustin Whittle, Developer Advocate

Presenting the work of **many** people at Uber



470+

Cities

73

Countries

1.5M+

Active Driver Partners

75%

of the U.S. population lives in a county with access to Uber

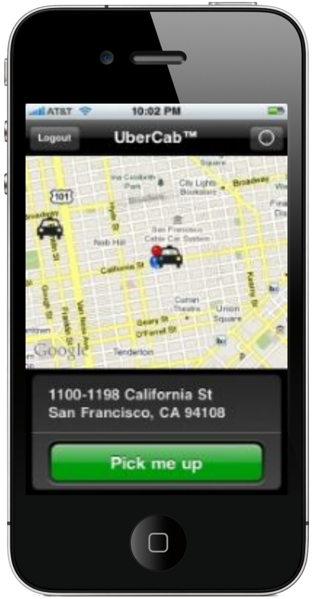
5+ Million

Trips per day

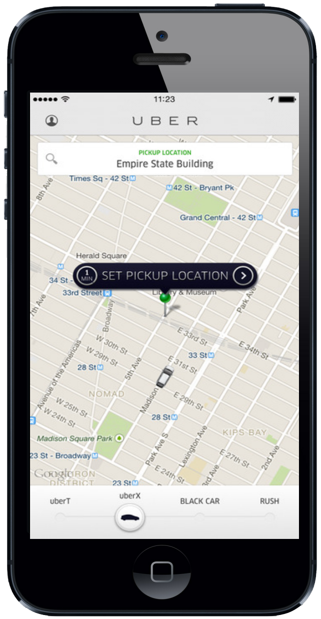
2+ Billion

Trips completed

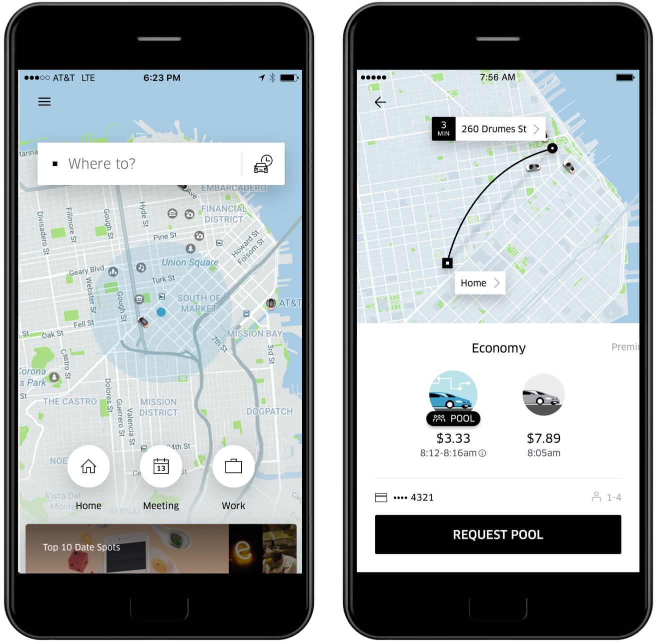
Evolution of the Uber App



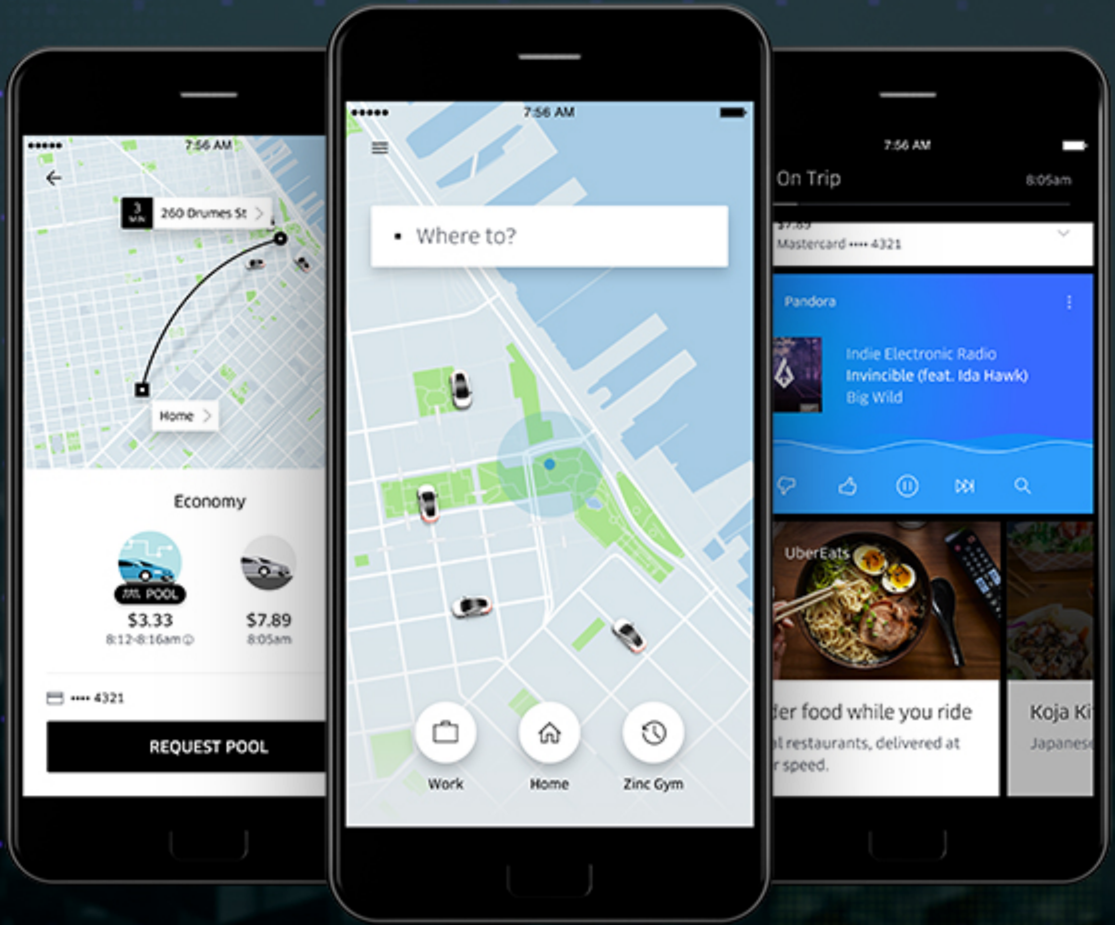
2009



2012

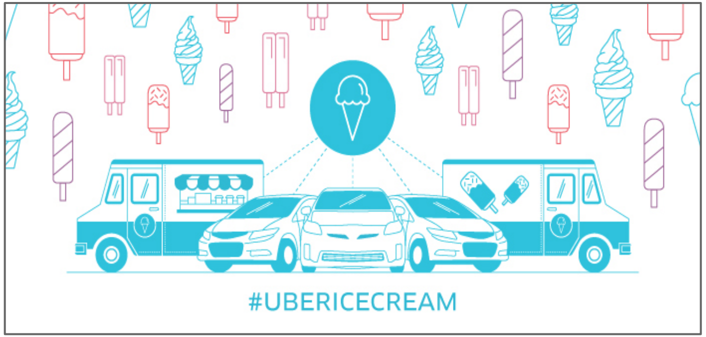


2016


7:56 AM
3 MIN
260 Drumes St
Home
Economy
POOL
$3.33
8:12-8:16am
$7.89
8:05am
•••• 4321
REQUEST POOL
7:56 AM
Where to?
Work
Home
Zinc Gym
7:56 AM
On Trip
8:05am
Mastercard •••• 4321
Pandora
Indie Electronic Radio
Invincible (feat. Ida Hawk)
Big Wild
UberEats
der food while you ride
al restaurants, delivered at
r speed.
Koja Ki
Japanes

UberCHOPPER
POWERED BY MAKANI KAI HELICOPTERS

UBER KITTENS
UBER
CHEEZ burger

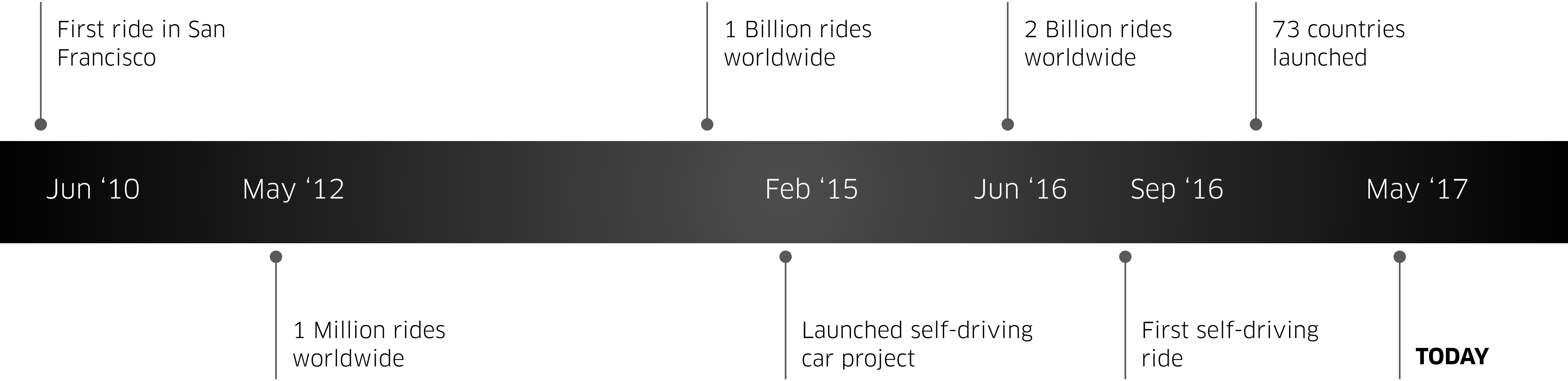

#UBERICECREAM

UBER
CHRISTMAS TREES
ON-DEMAND
#UberTREE

#UberMovers
ARRIVING AUGUST 16
UBER
Bellhops

2013

Singapore



Jun ‘10
First ride in San Francisco

May ‘12
1 Million rides worldwide

Feb ‘15
1 Billion rides worldwide

Launched self-driving car project

Jun ‘16
2 Billion rides worldwide

Sep ‘16
First self-driving ride

73 countries launched

May ‘17
TODAY

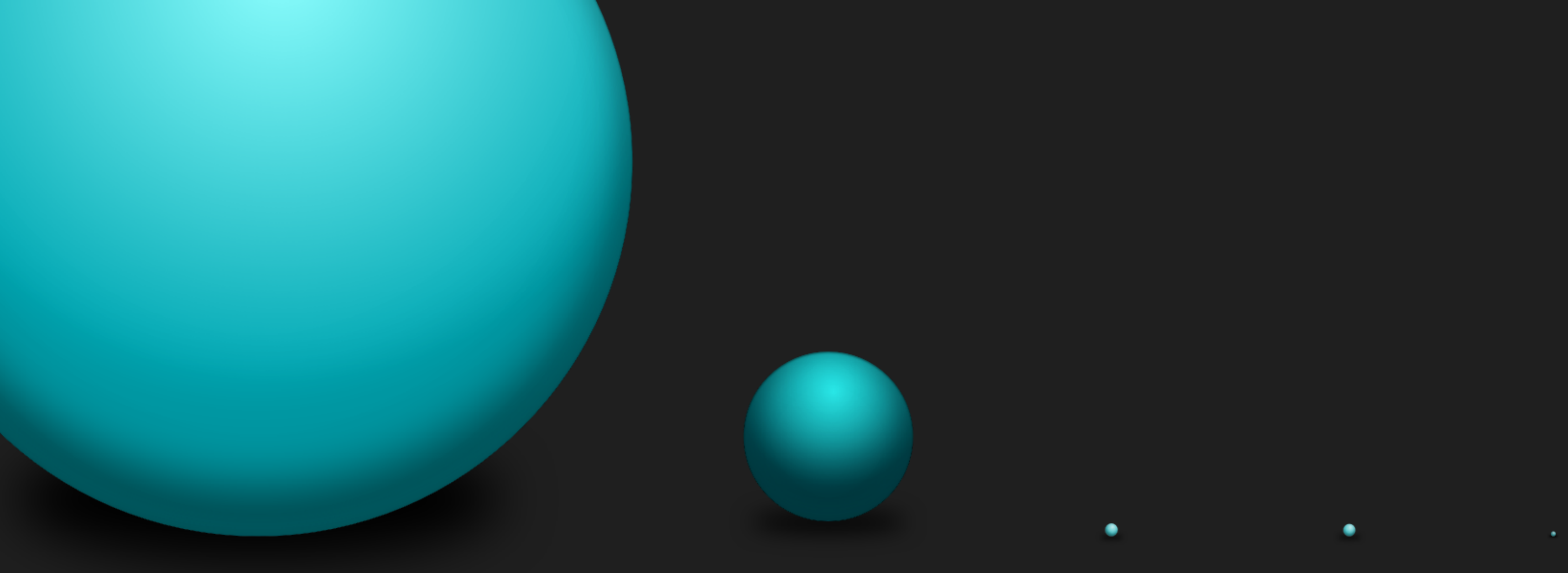
We hired lots of engineers

They wrote a lot of software

Organized as Programs and Platforms

	Program	Platform
Mission	Business Focused	Technology focused
Consumers	Non-technical + Internal/ External	Technical + Internal
Products	Feature-based products	Technology consumed by programs
Team	Cross-functional	Specialized
Direction	Product drivers	Engineering Drivers

Building a **platform** that interacts with the real world adds another level of complexity



15,000+
Repositories

3,400+
Micro Services

260+
Open Source Repositories

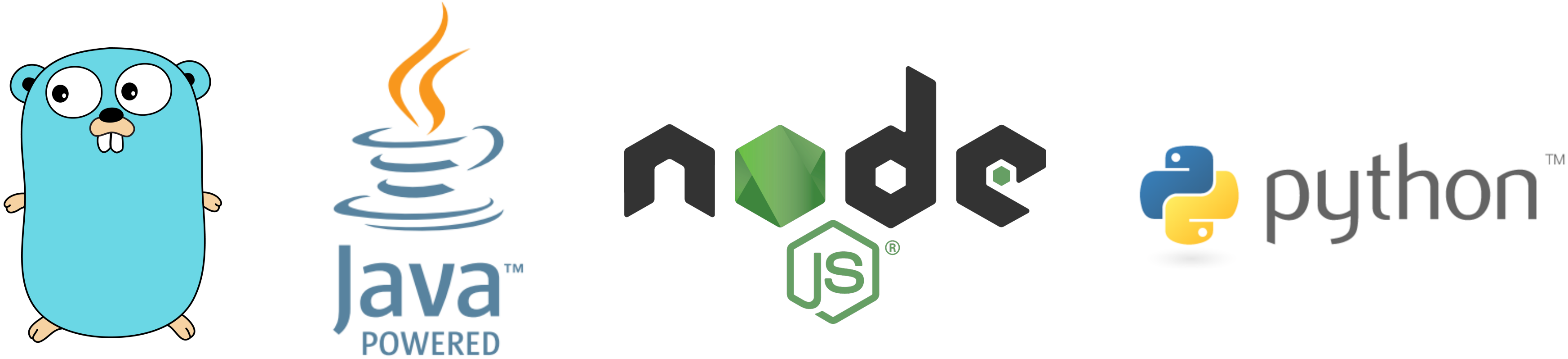
250+
Web Apps

10
Mobile Apps

Uber's **ability to scale** has come from high-velocity, hard engineering, along with autonomous teams and domains

Languages in use at Uber

We are a polyglot engineering organization, and we speak many languages.



In order of preference - many services are moving to Go (or Java)

Why JavaScript?

- **A single stack for frontend and backend**
- Simple interface with thorough documentation
- Lends itself to fast prototyping and quick iteration
- Asynchronous, nimble, flexible
- Avoid concurrency challenges
- Increasingly mature module ecosystem
- **Today we only use Node.js for web applications**

node
JS
®

How Uber uses JavaScript

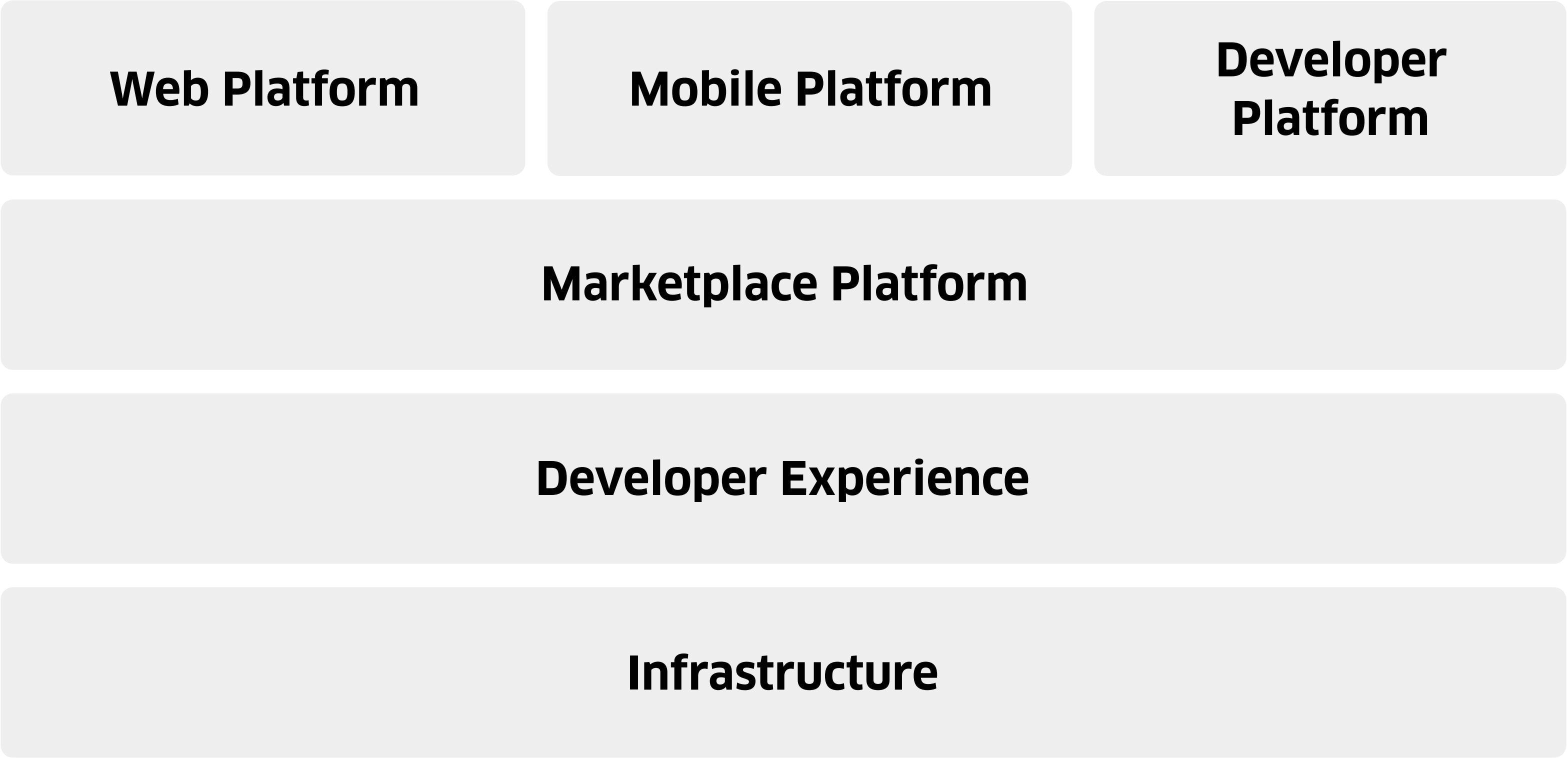
Marketplace Platform (core services)

Web Platform (web presence)

Visualizing Data (everywhere)

Developer Platform (external API)

Uber invests in developer **productivity**



Web Platform

Mobile Platform

Developer Platform

Marketplace Platform

Developer Experience

Infrastructure

Let Builders Build

12000+ full-time employees globally

- **Send proposal and request for comments to entire eng org**
  - Abstract, Architecture, UX, Ops, Security, Timeline
- **Start with scaffolding that is purpose built for our stack**
  - Skip the boilerplate while following best practices
- **Deploy with standardized global infrastructure**
  - Automate everything in a consistent way
- **Playbooks**
  - Operationalize teams at scale

Uber Service Foundation

- Asset Pipeline + CDN
- RPC (HTTP, TChannel, etc)
- Translations + Internationalization
- Data (Schemaless, Cassandra, SQL)
- Network
- Compute
- Storage
- Logging
- Performance + Failure Testing

- Instrumentation (Jaegar + OpenTracing)
- Source Control (Phabricator)
- Continuous Integration (Jenkins)
- Infrastructure + Deployments + ChatOps (u*)
- Monitoring + Alerting
- Metrics + Analytics
- Security + Compliance
- Experiments

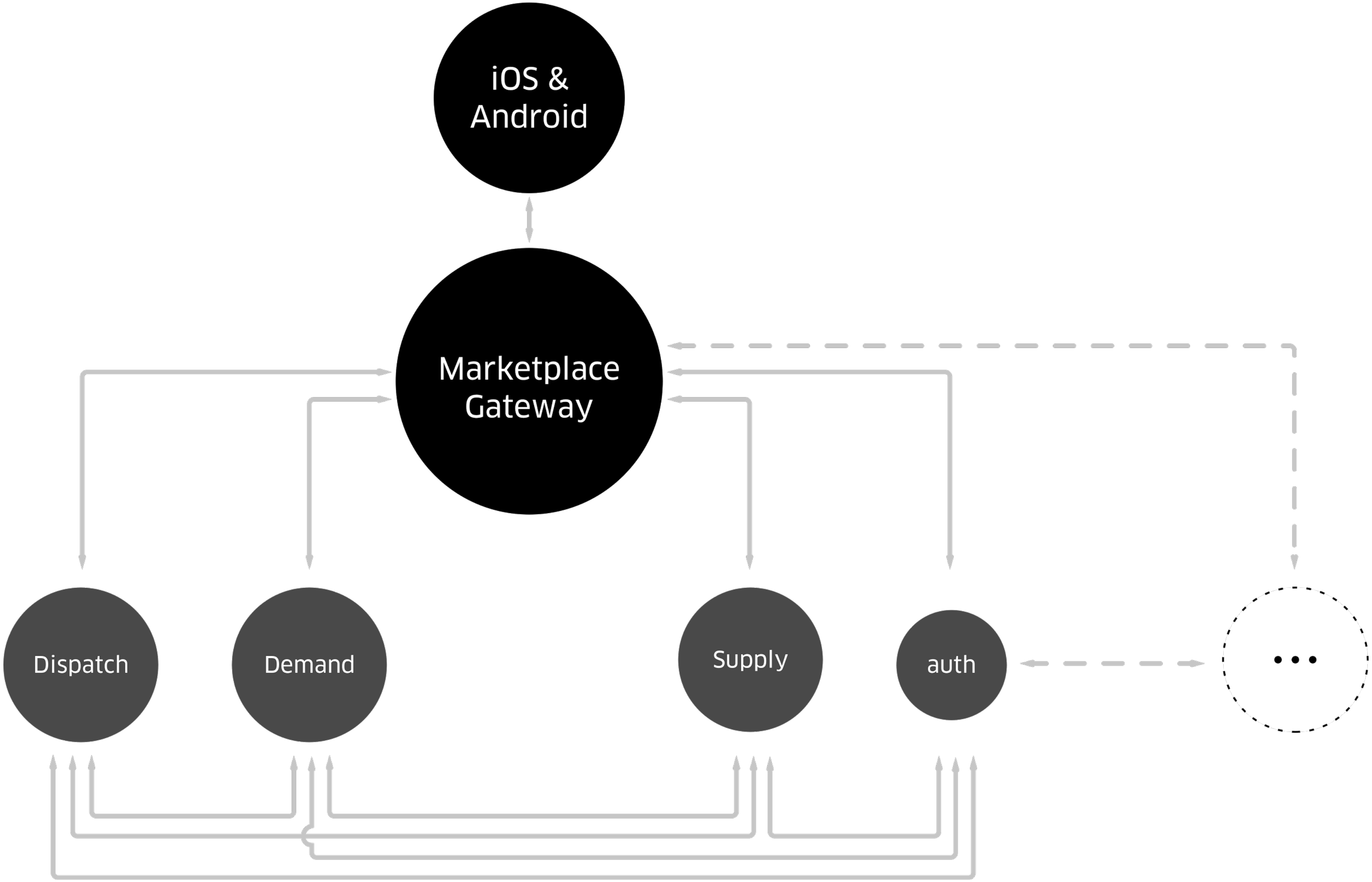
How Uber uses JavaScript

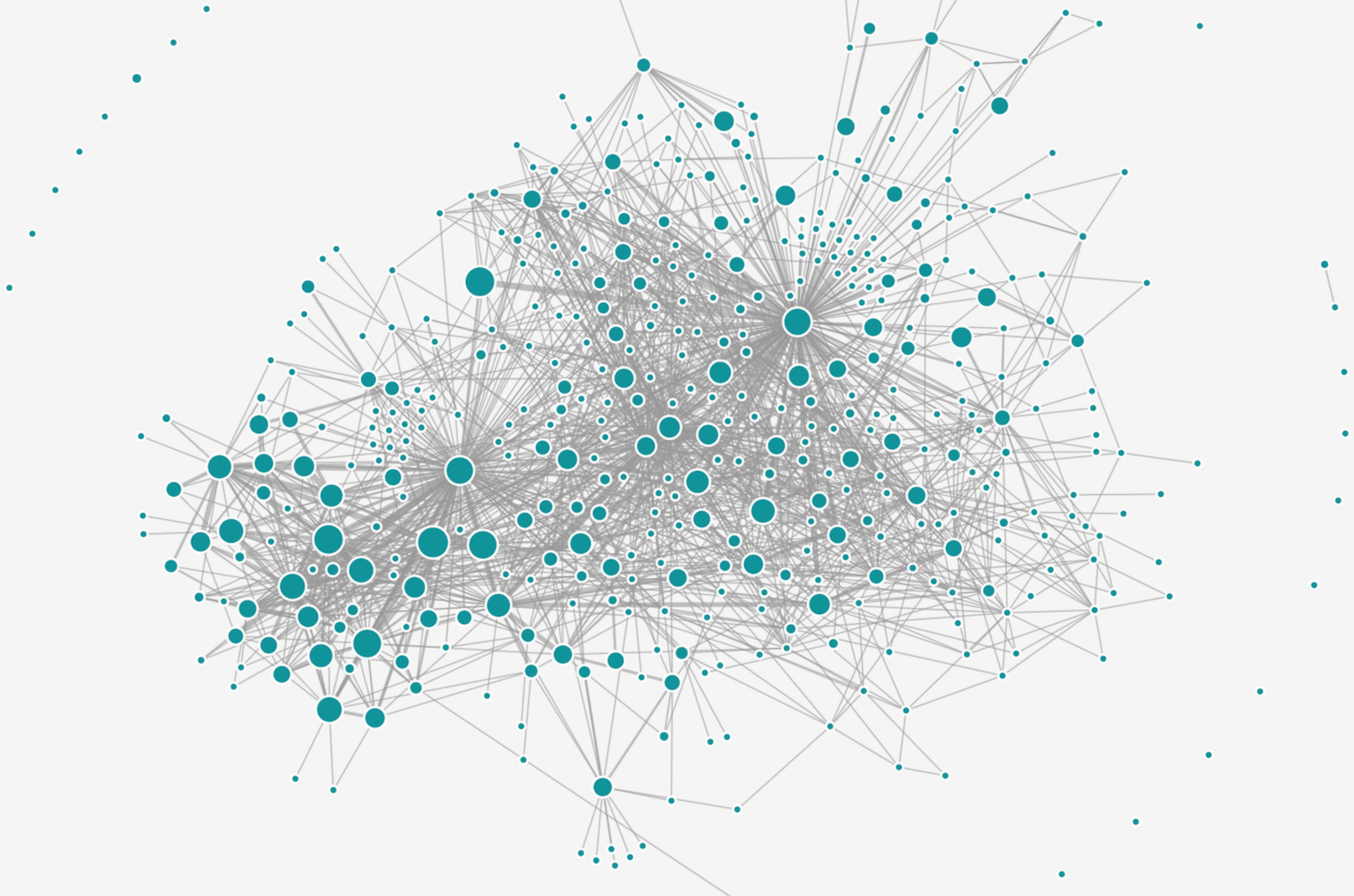
Marketplace Platform (core services)

Web Platform (web presence)

Visualizing Data (everywhere)

Developer Platform (external API)


iOS & Android
Marketplace Gateway
Dispatch
Demand
Supply
auth
•••



Marketplace Gateway

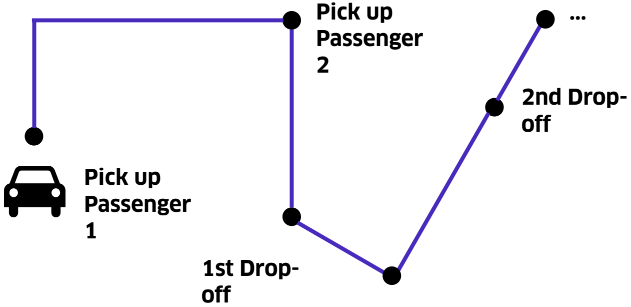
Real-time API is the only interface

Real-time API is the frontline web service of Uber serving **99.99% of all the mobile traffic with >1000 HTTP endpoints.**

- Acts as a router to 3400+ microservices
- A single place for monitoring and documentation

Logistics Challenges

- How to optimize for overall marketplace efficiency?
- How to dynamically price trips and balance efficiency?
- How to maximize pick-up efficiency (pick up zones, pick up suggestions)?
- How to minimize wait times for drivers and riders?
- How to deal with different street configurations?
- Combine different logistics products (Pool, Eats, Rush)



UberPool is a traveling salesman problem on steroids

Uber Engineering Stack Evolution

2009-2010 - Outsourced PHP + MySQL

Jan 2011 - "Dispatch" - Node.JS/MongoDB

Jan 2011 - "API" - Python/SQLAlchemy/MySQL

Feb 2012 - Dispatch swaps MongoDB for Redis

May 2012 - Dispatch adds on fallback

Jan 2013 - First non-API Python services

Feb 2013 - API switched to Postgres

Mar 2014 - New Python services use MySQL

Mar 2014 - Schemaless begins, must finish before Postgres collapse

Sep 2014 - First Schemaless - trips out of Postgres

Aug 2015 - Dispatch X.0 / Ringpop / Riak

Jan 2016 - Go, Java, More Abstractions

May 2017 - UberFx for Go, Cloud

Monolith ≠ bad

- Well, not bad at first, but can turn into a ball of mud quickly
- Optimizes development velocity of early stage projects
- Consolidates operational overhead in one place
- Monolith isn’t as bad when there is a single team that owns it

A microservice is a service that is focused on only one thing.

One Thing != One API

Pros

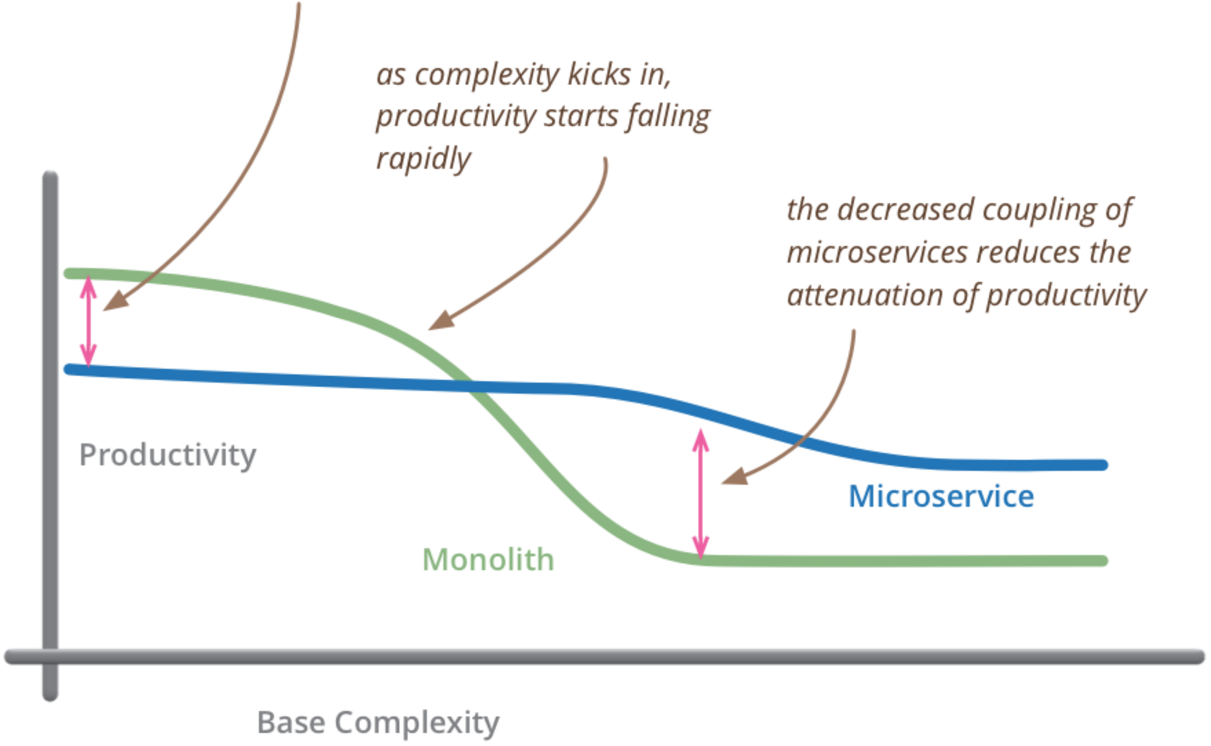
What’s good

- **Clear ownership boundaries**
    - Most of the time
- **Product-specific velocity**
    - Move the business faster
- **Able to solve scaling issues**
    - Systems as well as company
    - Training, documentation, tooling
- **Language/platform independence**
    - Best tool for the job
- **Reliability of the system**
    - Independent
    - Scalable
    - Testable

Cons

What’s not so good

- **Increased complexity**
    - Naming + Discovery
    - Eventual consistency
    - Performance Impact
    - Distributed tracing
    - Testing
- **Operational overhead**
    - Monitoring
    - Continuous Integration
    - Deployment
    - Documentation
    - Instrumentation
    - Security
- **Repeated code/effort**



for less-complex systems, the extra baggage required to manage microservices reduces productivity

as complexity kicks in, productivity starts falling rapidly

the decreased coupling of microservices reduces the attenuation of productivity

but remember the skill of the team will outweigh any monolith/microservice choice

https://www.martinfowler.com/bliki/MicroservicePremium.html

Ringpop

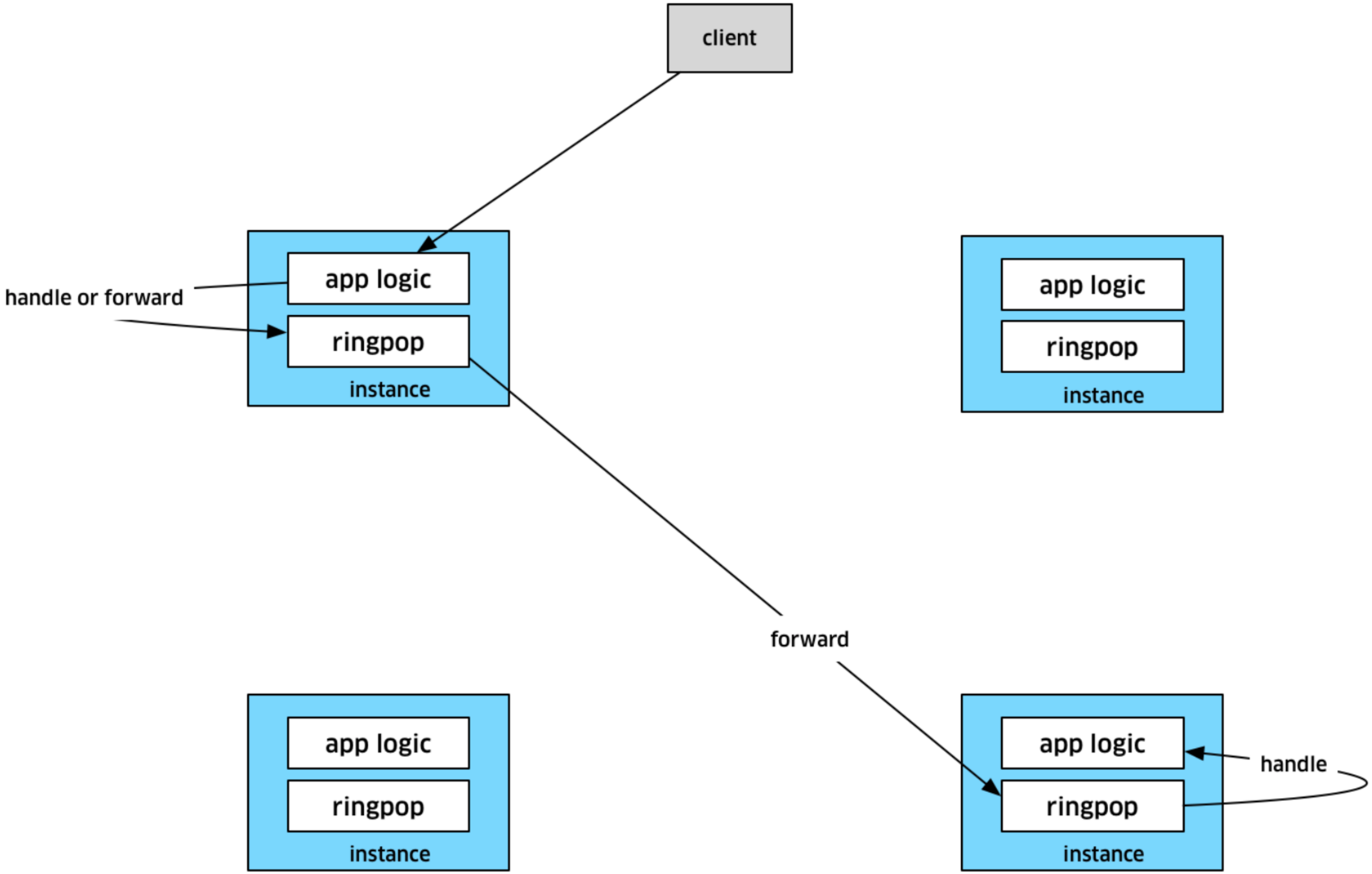
Ringpop is a library that brings cooperation and coordination to distributed applications. It maintains a consistent hash ring on top of a membership protocol and provides request forwarding as a routing convenience.

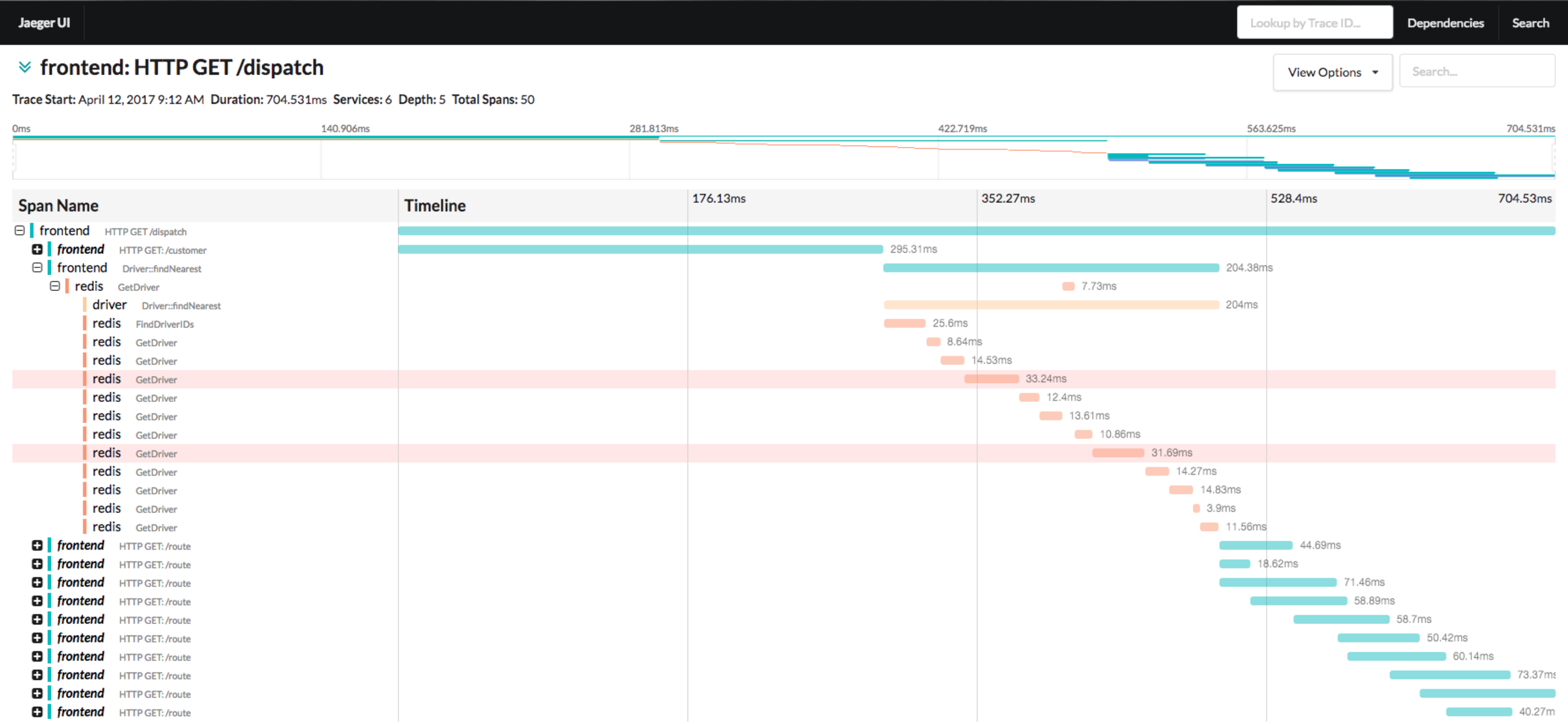
TChannel

TChannel is a networking framing protocol used for general RPC, supporting out-of-order responses at extremely high performance where intermediaries can make a forwarding decision quickly. Client libraries available in Go, Java, Node and Python.

Jaeger

Jaeger, inspired by Dapper and OpenZipkin, is a distributed tracing system. It can be used for monitoring microservice-based architectures. Client libraries available in Go, Java, Node and Python using the OpenTracing standard.


client
handle or forward
app logic
ringpop
instance
app logic
ringpop
instance
forward
app logic
ringpop
instance
app logic
ringpop
instance
handle



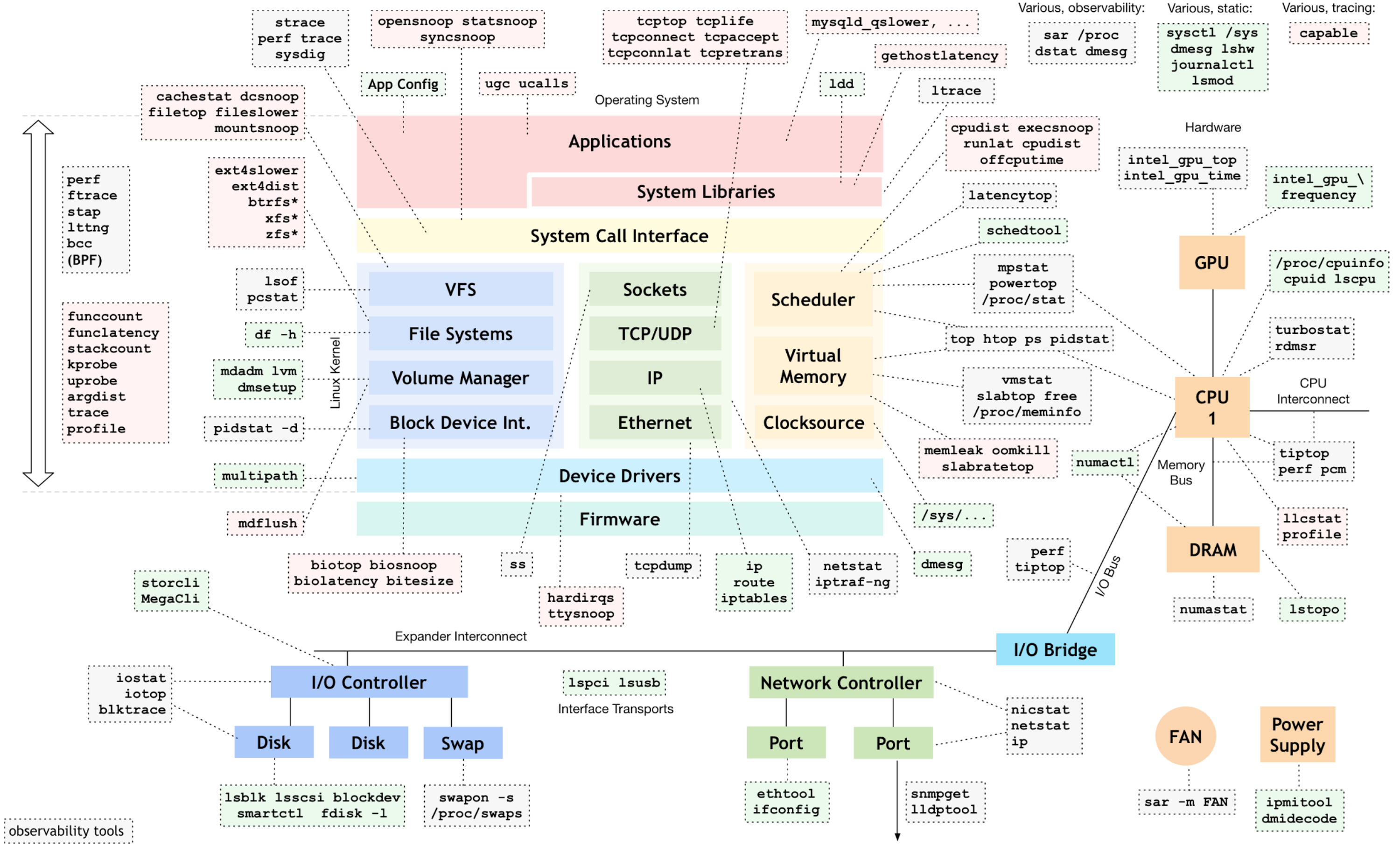
frontend: HTTP GET /dispatch

Trace Start: April 12, 2017 9:12 AM **Duration:** 704.531ms **Services:** 6 **Depth:** 5 **Total Spans:** 50

Span Name | **Timeline** 176.13ms 352.27ms 528.4ms 704.53ms

```
frontend  HTTP GET /dispatch
  frontend  HTTP GET: /customer          295.31ms
  frontend  Driver::findNearest          204.38ms
    redis  GetDriver                     7.73ms
      driver  Driver::findNearest        204ms
      redis  FindDriverIDs               25.6ms
      redis  GetDriver                   8.64ms
      redis  GetDriver                   14.53ms
      redis  GetDriver                   33.24ms
      redis  GetDriver                   12.4ms
      redis  GetDriver                   13.61ms
      redis  GetDriver                   10.86ms
      redis  GetDriver                   31.69ms
      redis  GetDriver                   14.27ms
      redis  GetDriver                   14.83ms
      redis  GetDriver                   3.9ms
      redis  GetDriver                   11.56ms
  frontend  HTTP GET: /route             44.69ms
  frontend  HTTP GET: /route             18.62ms
  frontend  HTTP GET: /route             71.46ms
  frontend  HTTP GET: /route             58.89ms
  frontend  HTTP GET: /route             58.7ms
  frontend  HTTP GET: /route             50.42ms
  frontend  HTTP GET: /route             60.14ms
  frontend  HTTP GET: /route             73.37ms
  frontend  HTTP GET: /route
  frontend  HTTP GET: /route             40.27m
```

Linux Performance Tools



Operating System

Applications

System Libraries

System Call Interface

Linux Kernel

VFS

File Systems

Volume Manager

Block Device Int.

Sockets

TCP/UDP

IP

Ethernet

Scheduler

Virtual Memory

Clocksource

Device Drivers

Firmware

Hardware

GPU

CPU 1

CPU Interconnect

Memory Bus

DRAM

I/O Bus

I/O Bridge

Expander Interconnect

I/O Controller

Disk

Disk

Swap

Interface Transports

Network Controller

Port

Port

FAN

Power Supply

Various, observability:

sar /proc
dstat dmesg

Various, static:

sysctl /sys
dmesg lshw
journalctl
lsmod

Various, tracing:

capable

strace
perf trace
sysdig

opensnoop statsnoop
syncsnoop

tcptop tcplife
tcpconnect tcpaccept
tcpconnlat tcpretrans

mysqld_qslower, ...

gethostlatency

App Config

ugc ucalls

ldd

ltrace

cachestat dcsnoop
filetop fileslower
mountsnoop

perf
ftrace
stap
lttng
bcc
(BPF)

ext4slower
ext4dist
btrfs*
xfs*
zfs*

cpudist execsnoop
runlat cpudist
offcputime

intel_gpu_top
intel_gpu_time

intel_gpu_\
frequency

latencytop

schedtool

mpstat
powertop
/proc/stat

/proc/cpuinfo
cpuid lscpu

turbostat
rdmsr

lsof
pcstat

funccount
funclatency
stackcount
kprobe
uprobe
argdist
trace
profile

df -h

mdadm lvm
dmsetup

pidstat -d

multipath

mdflush

top htop ps pidstat

vmstat
slabtop free
/proc/meminfo

memleak oomkill
slabratetop

numactl

tiptop
perf pcm

/sys/...

llcstat
profile

perf
tiptop

dmesg

numastat

lstopo

biotop biosnoop
biolatency bitesize

ss

hardirqs
ttysnoop

tcpdump

ip
route
iptables

netstat
iptraf-ng

storcli
MegaCli

iostat
iotop
blktrace

lspci lsusb

nicstat
netstat
ip

lsblk lsscsi blockdev
smartctl fdisk -l

swapon -s
/proc/swaps

ethtool
ifconfig

snmpget
lldptool

sar -m FAN

ipmitool
dmidecode

observability tools

static performance tools

perf-tools/bcc tracing tools

these can observe the state of the system at rest, without load

https://github.com/brendangregg/perf-tools https://github.com/iovisor/bcc

style inspired by reddit.com/u/redct

http://www.brendangregg.com/linuxperf.html 2017

Lessons learned scaling Node.js

Growing pains learned the hard way

- Latency is too high for ultra performant backend systems (p99 for max latency)
- Early on it made it quick to iterate, but as the size of the team scaled the developer velocity started to slow down
    - Microservices enforce a tight interface so having static typing enables large teams to catch issues earlier. It has an impact with 100+ devs.
- Quick to learn, but easy to write poor quality code
    - Enables you to move fast, but allows for sloppy code
- Great ecosystem of small libraries, but many were immature compared to Java/Go
- Don't be afraid of writing c++

Lessons learned scaling Uber

Growing pains learned the hard way

- HTTP and JSON was designed for browsers; using RPC is better for computer-to-computer requests
- After a certain age, microservices should become immutable
- Having multiple languages allows for team preferences, but segregates developers based on language and prevents easy re-use of code across services
    - Company > Team > Self
- Monorepos allow for changes to be made across multiple services atomically, but prevent future open-sourcing and subset checkouts
- Performance problems are difficult to debug cross-language without standardised service dashboards and observability tools
- Logging should never slow production down; in a failure storm, the logging system should drop rather than delay
- **Everything is a tradeoff: Be intentional**

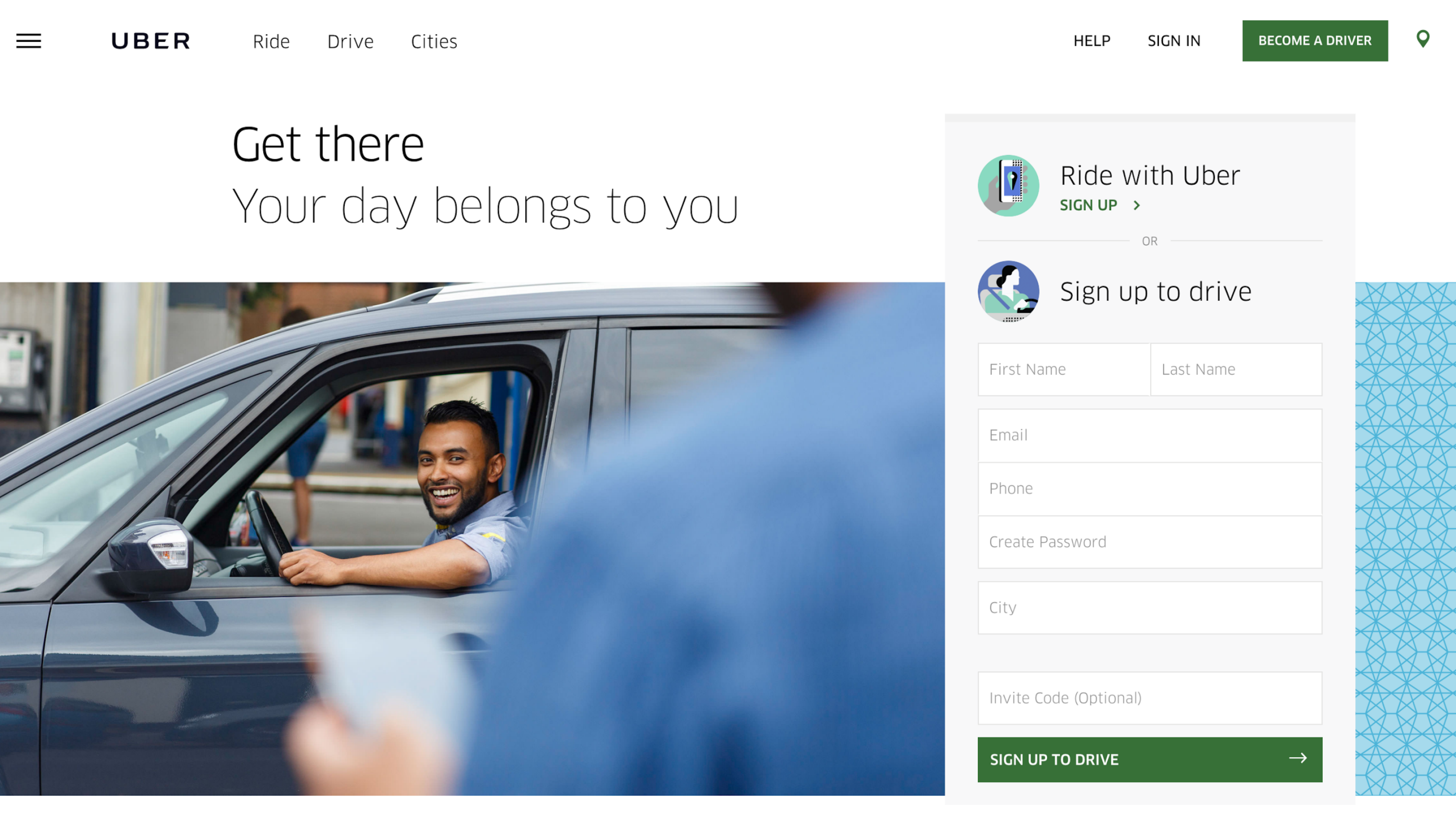
How Uber uses JavaScript

Marketplace Platform (core services)

Web Platform (web presence)

Visualizing Data (everywhere)

Developer Platform (external API)



Get there

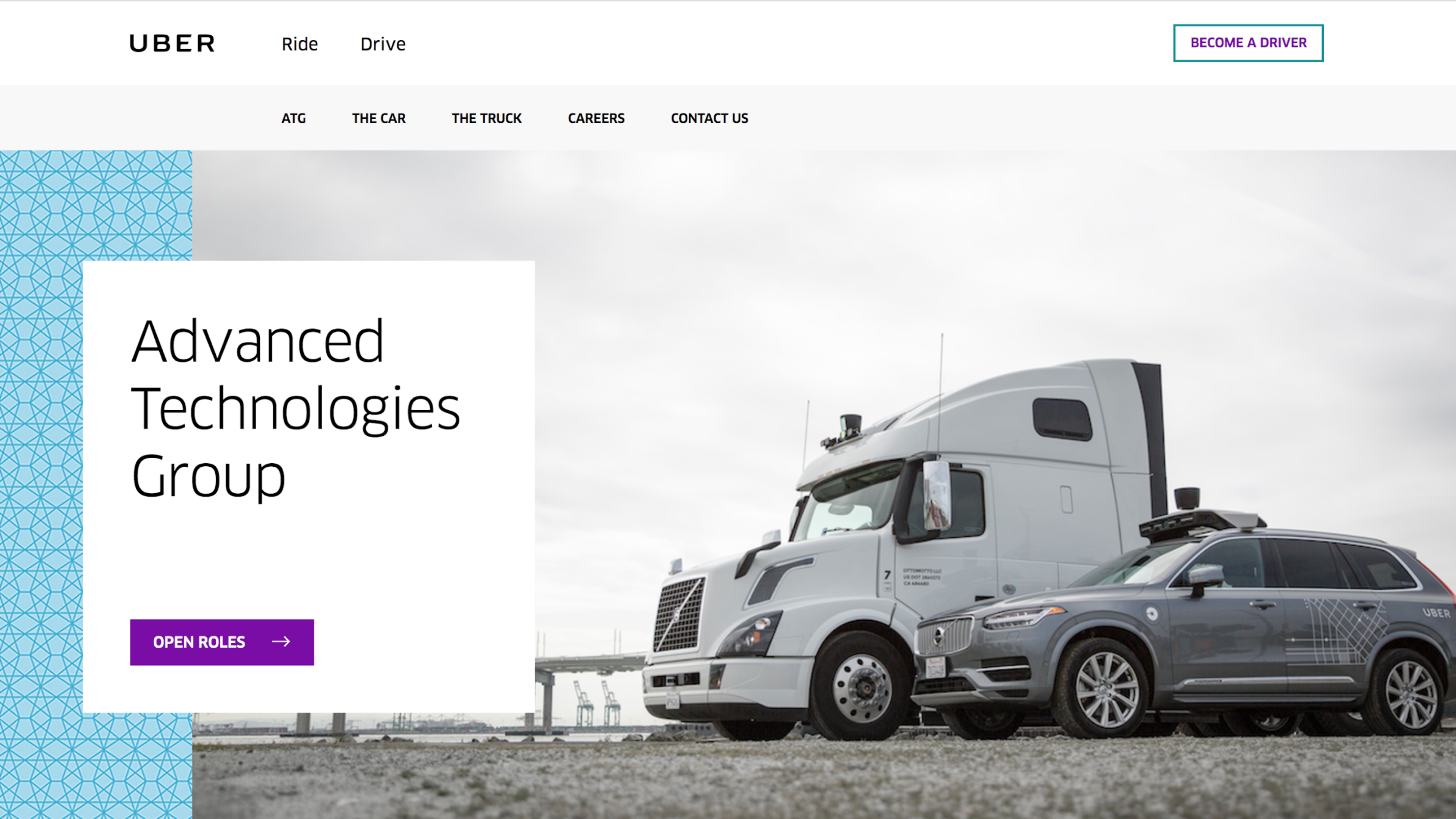
Your day belongs to you

Ride with Uber

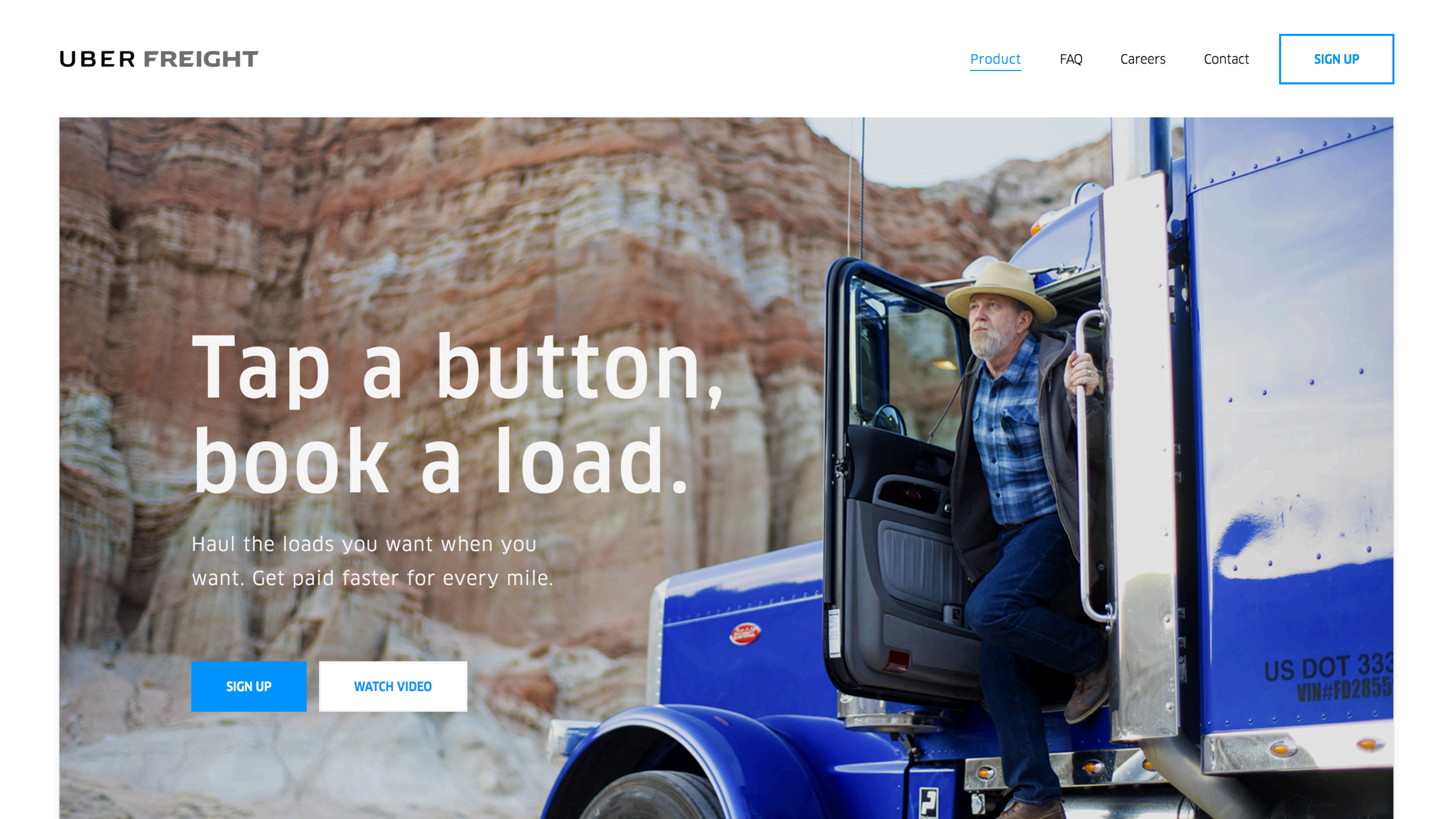
OR

Sign up to drive

First Name	Last Name
Email	
Phone	
Create Password	
City	
Invite Code (Optional)	

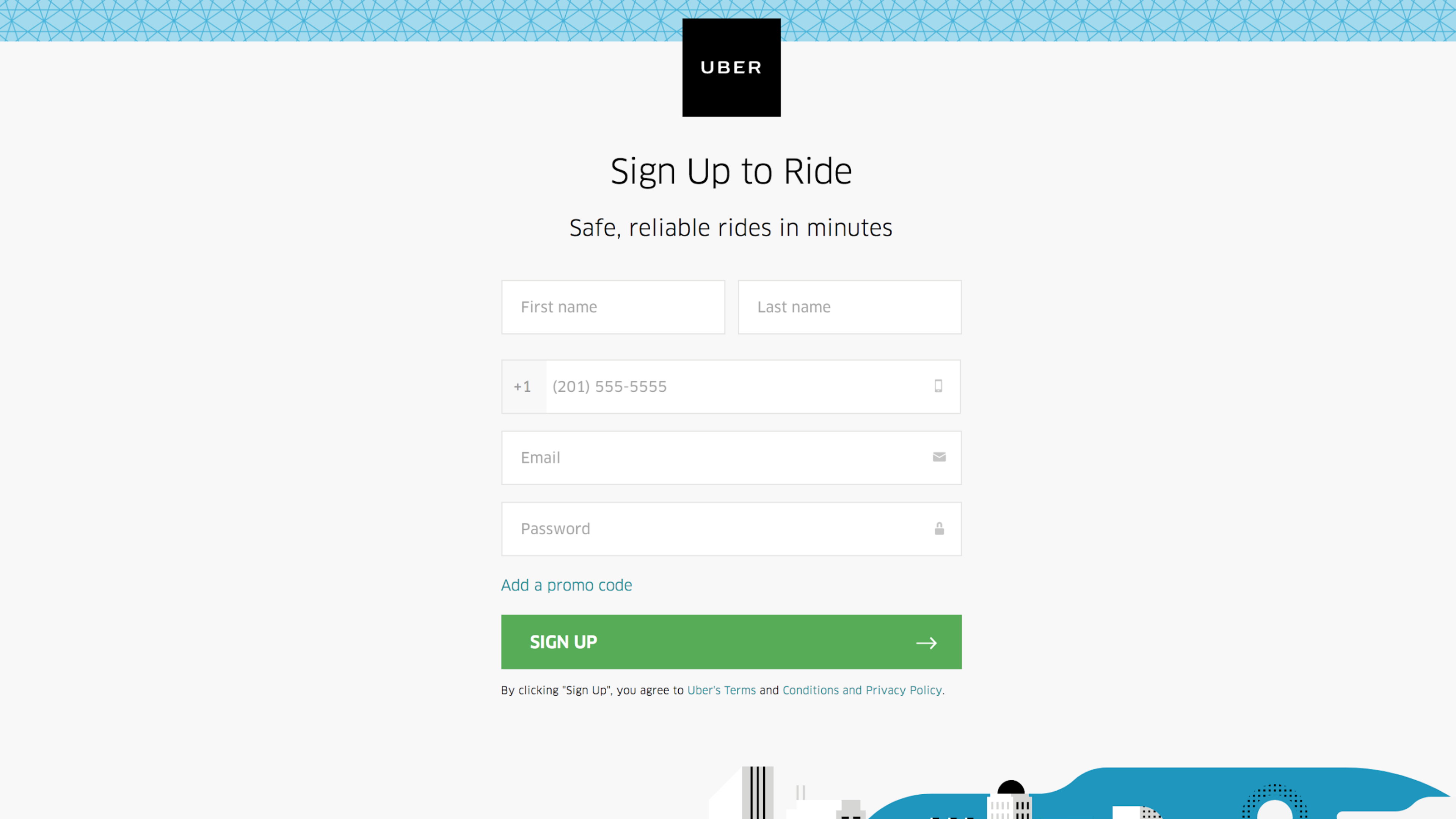


Advanced Technologies Group



Tap a button, book a load.

Haul the loads you want when you want. Get paid faster for every mile.

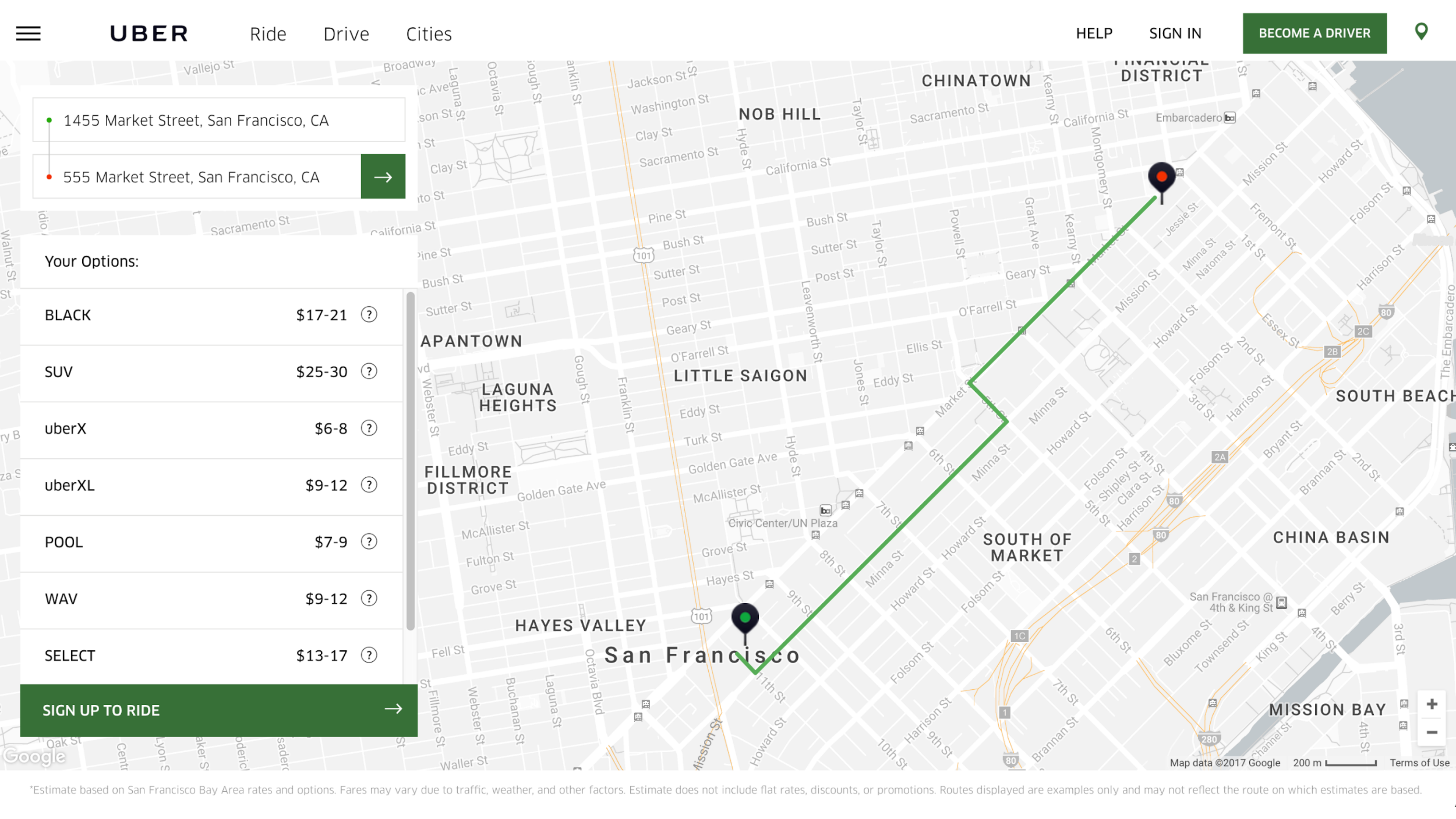


Sign Up to Ride

Safe, reliable rides in minutes

Add a promo code

By clicking "Sign Up", you agree to Uber's Terms and Conditions and Privacy Policy.



1455 Market Street, San Francisco, CA

555 Market Street, San Francisco, CA

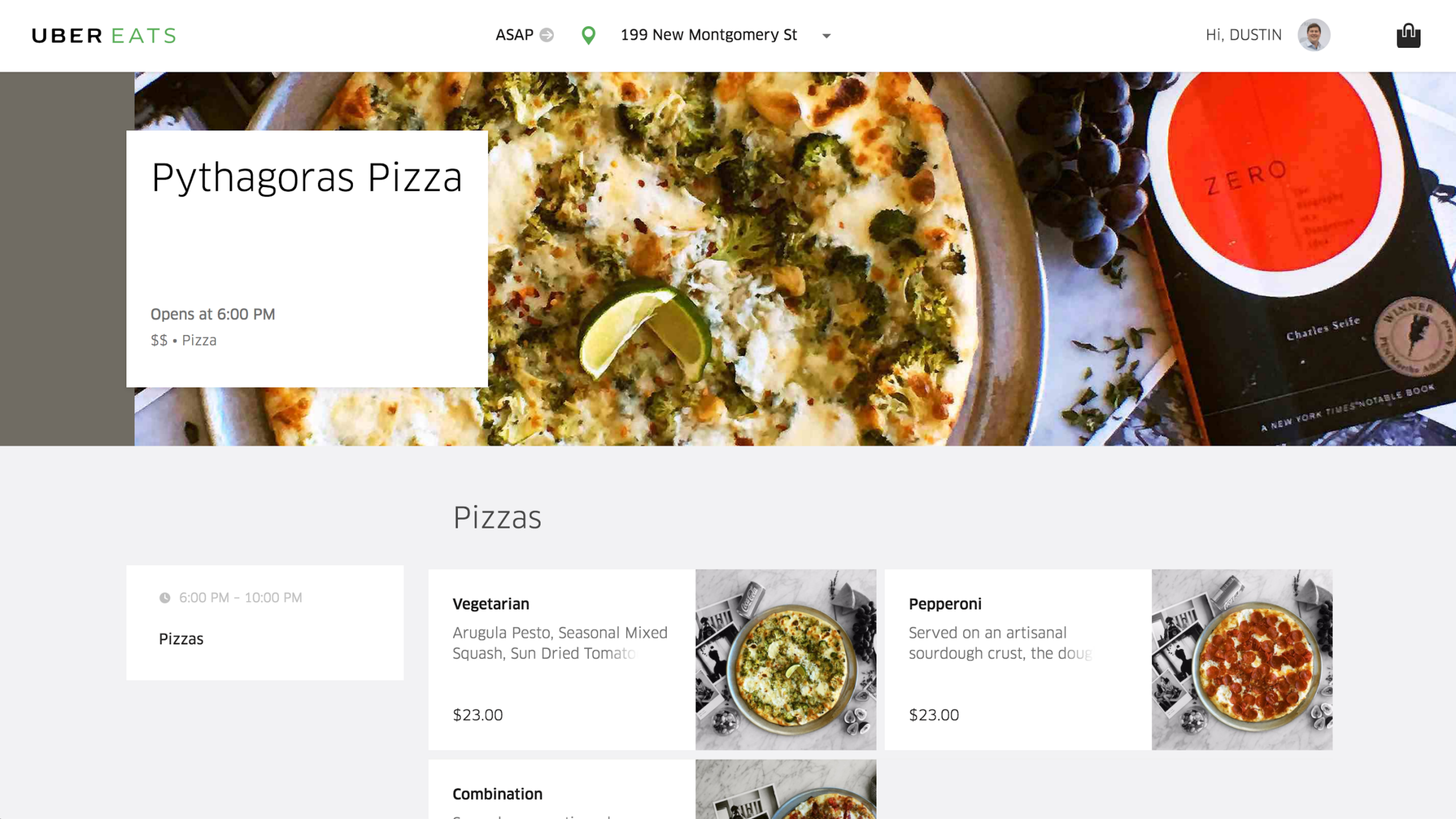
→

Your Options:

BLACK	$17-21 ?
SUV	$25-30 ?
uberX	$6-8 ?
uberXL	$9-12 ?
POOL	$7-9 ?
WAV	$9-12 ?
SELECT	$13-17 ?

SIGN UP TO RIDE →

*Estimate based on San Francisco Bay Area rates and options. Fares may vary due to traffic, weather, and other factors. Estimate does not include flat rates, discounts, or promotions. Routes displayed are examples only and may not reflect the route on which estimates are based.



Pythagoras Pizza

Opens at 6:00 PM

$$ • Pizza

Pizzas

6:00 PM – 10:00 PM

Pizzas

Vegetarian

Arugula Pesto, Seasonal Mixed Squash, Sun Dried Tomato

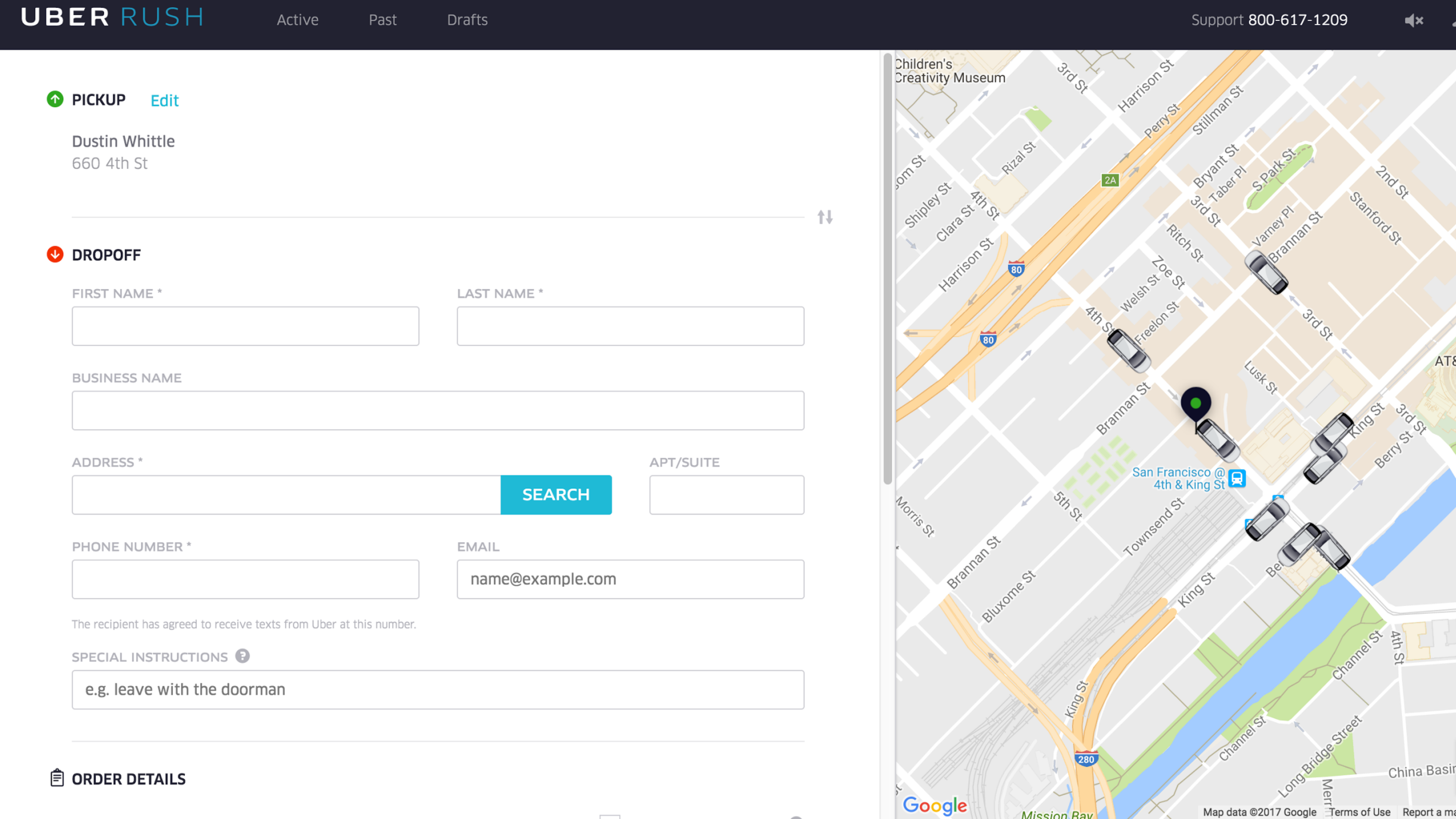
$23.00

Pepperoni

Served on an artisanal sourdough crust, the doug

$23.00

Combination



PICKUP Edit

Dustin Whittle
660 4th St

DROPOFF

FIRST NAME *

LAST NAME *

BUSINESS NAME

ADDRESS *

APT/SUITE

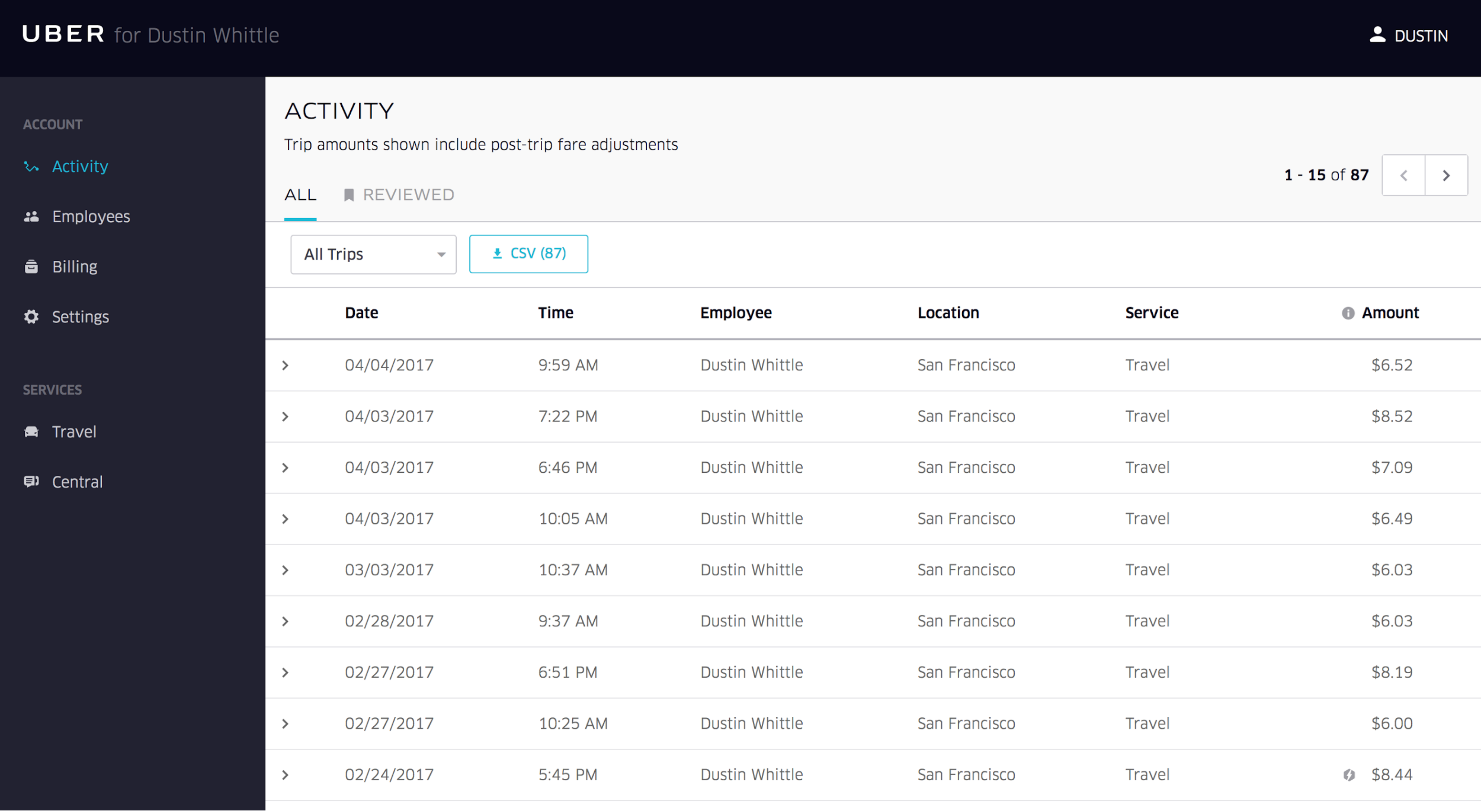
PHONE NUMBER *

EMAIL
name@example.com

The recipient has agreed to receive texts from Uber at this number.

SPECIAL INSTRUCTIONS ?
e.g. leave with the doorman

ORDER DETAILS



UBER for Dustin Whittle

DUSTIN

ACCOUNT

- Activity
- Employees
- Billing
- Settings

SERVICES

- Travel
- Central

ACTIVITY

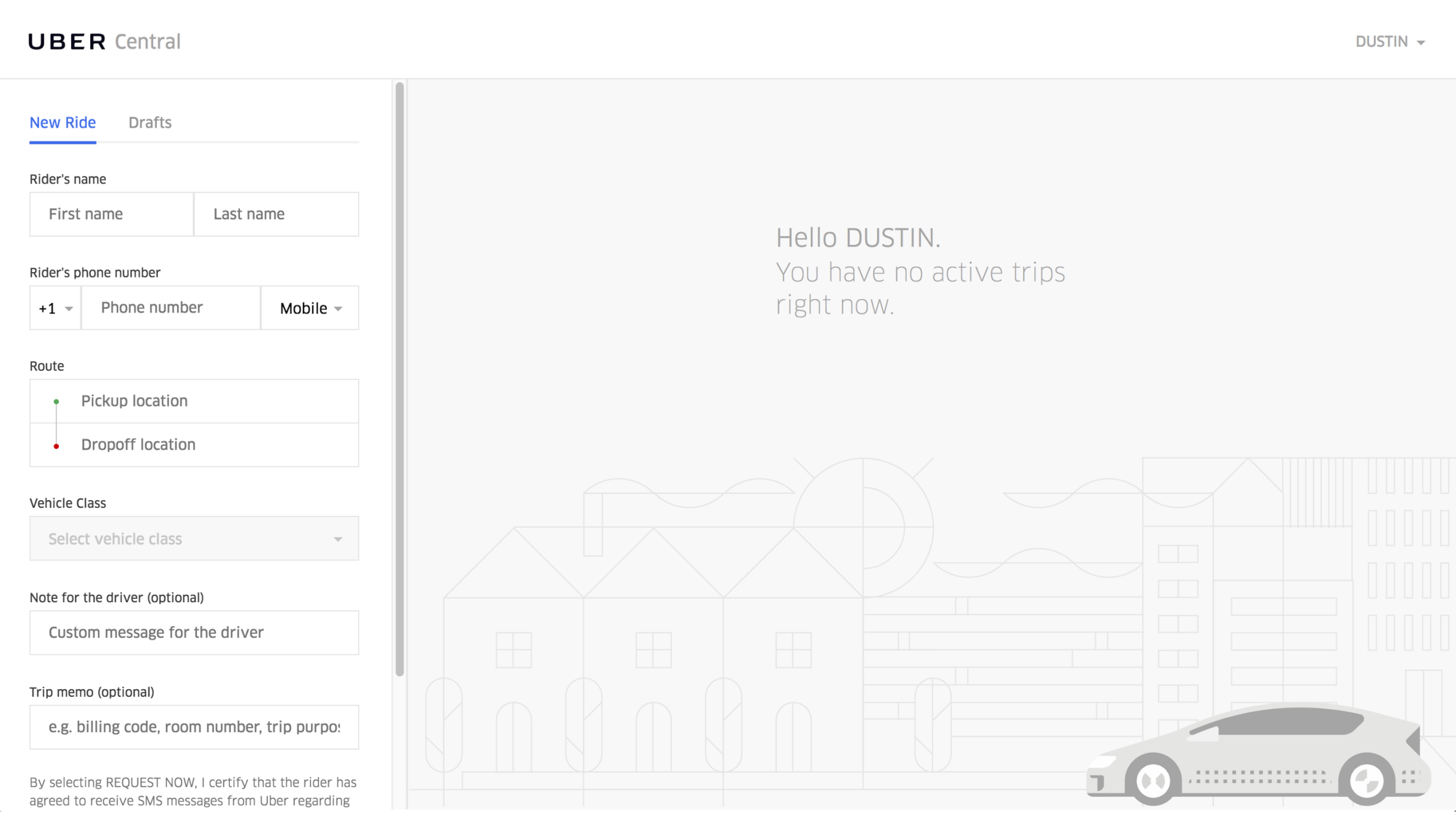
Trip amounts shown include post-trip fare adjustments

ALL REVIEWED

1 - 15 of 87 ‹ ›

All Trips ▾ CSV (87)

	Date	Time	Employee	Location	Service	Amount
›	04/04/2017	9:59 AM	Dustin Whittle	San Francisco	Travel	$6.52
›	04/03/2017	7:22 PM	Dustin Whittle	San Francisco	Travel	$8.52
›	04/03/2017	6:46 PM	Dustin Whittle	San Francisco	Travel	$7.09
›	04/03/2017	10:05 AM	Dustin Whittle	San Francisco	Travel	$6.49
›	03/03/2017	10:37 AM	Dustin Whittle	San Francisco	Travel	$6.03
›	02/28/2017	9:37 AM	Dustin Whittle	San Francisco	Travel	$6.03
›	02/27/2017	6:51 PM	Dustin Whittle	San Francisco	Travel	$8.19
›	02/27/2017	10:25 AM	Dustin Whittle	San Francisco	Travel	$6.00
›	02/24/2017	5:45 PM	Dustin Whittle	San Francisco	Travel	$8.44



Rider's name

First name | Last name

Rider's phone number

+1 | Phone number | Mobile

Route

Pickup location

Dropoff location

Vehicle Class

Select vehicle class

Note for the driver (optional)

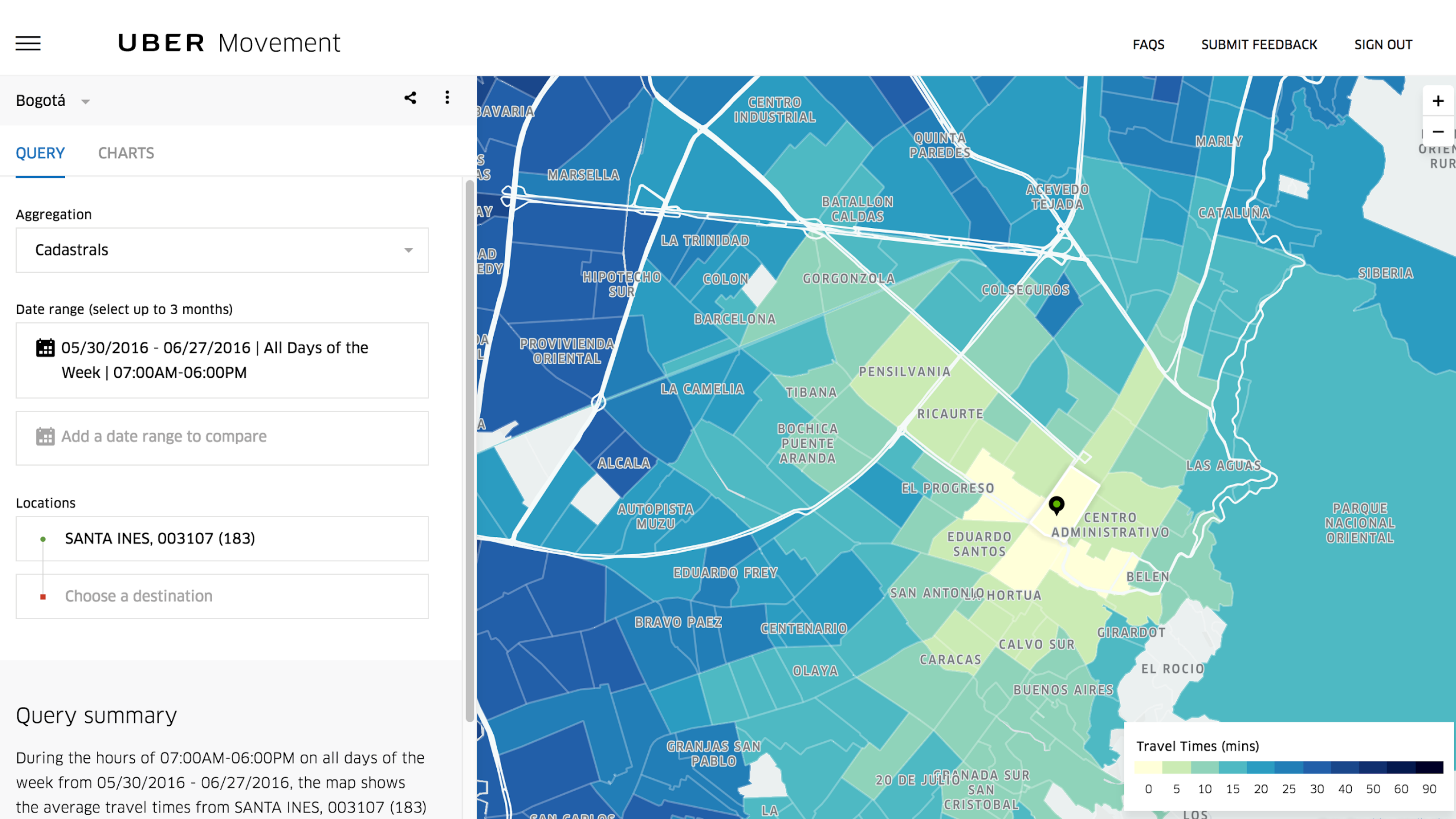
Custom message for the driver

Trip memo (optional)

e.g. billing code, room number, trip purpo:

By selecting REQUEST NOW, I certify that the rider has agreed to receive SMS messages from Uber regarding

Hello DUSTIN.
You have no active trips right now.



UBER Movement

FAQS SUBMIT FEEDBACK SIGN OUT

Bogotá

QUERY CHARTS

Aggregation

Cadastrals

Date range (select up to 3 months)

05/30/2016 - 06/27/2016 | All Days of the Week | 07:00AM-06:00PM

Add a date range to compare

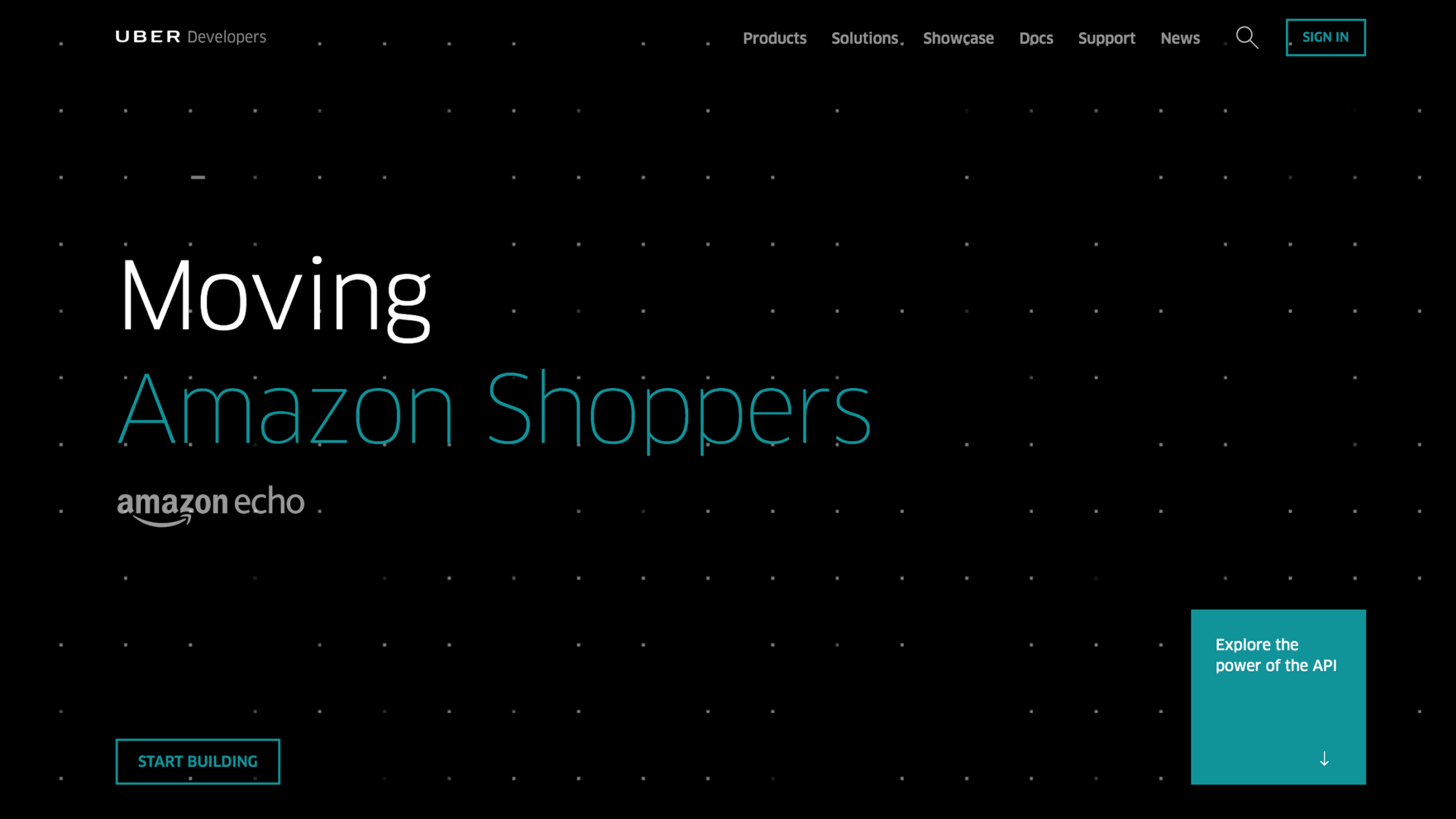
Locations

SANTA INES, 003107 (183)

Choose a destination

Query summary

During the hours of 07:00AM-06:00PM on all days of the week from 05/30/2016 - 06/27/2016, the map shows the average travel times from SANTA INES, 003107 (183)



Moving
Amazon Shoppers

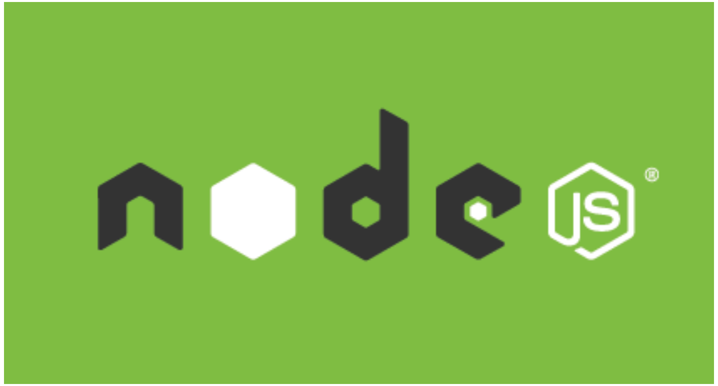
Explore the power of the API ↓

The Web Platform

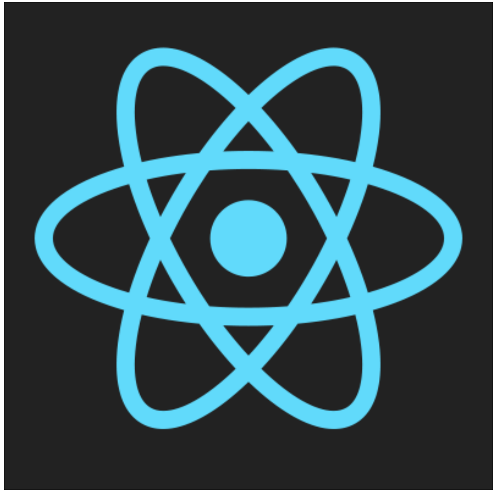
Serving 250+ web properties

Without a consistent foundation, all of these applications could have been built off of entirely different tools and architectures, creating a higher likelihood of security vulnerabilities, duplicated effort across teams, technical debt, minimal UI consistency, etc.

Express


node
JS

npm



Redux

Uber Web Platform built from Open Source

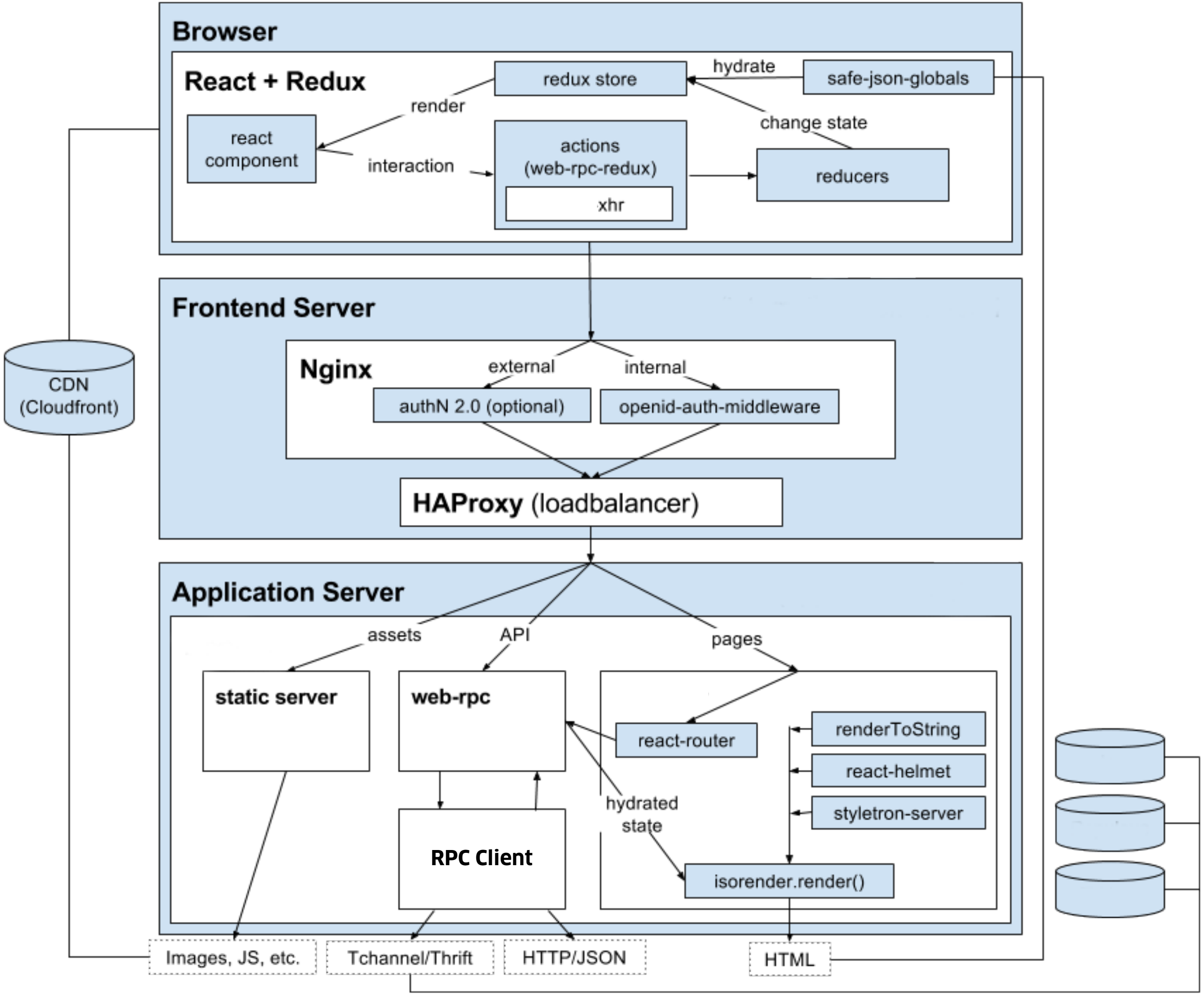
- **Nginx + HAProxy**: Web frontends + load balancers
- **NPM**: NPM registry
- **Babel**: For ES6 and ES7 transpilation
- **Express**: Fast, unopinionated, minimalist web framework for Node.js
- **React**: UI rendering library
- **Redux**: Predictable state application framework
- **Styletron**: Universal, high-performance JavaScript styles
- **ESLint**: Maintain a consistent code style across many teams
- **Browserify**: For bundling client-side code
- **Tape + Enzyme + Sinon**: Testing frameworks
- **Unitest + Istanbul**: Seamless Node.js and Browser testing with coverage
- **Gulp + Dev Tools**: Standardized build tooling

Uber Web Platform added special sauce

- Internal npm registry with caching
- Yeoman like scaffolding for bootstrap new apps with best practices
- Customized middlewares for our *Express* based app framework
    - Authentication, Security (XSS, CSRF, CSP), Metrics, Logging, I18n, Errors, Instrumentation, Analytics
- Bootstrap like UI component library build on top of React.js
- RPC library for intelligently using http/tchannel
- Shared Build + Test Gulp tasks with hot reloading, ES6/JSX, asset versioning and deployment pipelines, testing + linting
- Universal rendering for server/client

Frontend vs Backend

- We separate our **frontend** services (those that serve web pages) from our **backend** services (those that get data from some database)
- Services can take advantage of universal rendering + tooling
- You can utilize languages better suited for your backend services (Java, Go)
- You can deploy and scale your services separately
- You can reuse the backend service API for other clients


Browser
React + Redux
redux store
hydrate
safe-json-globals
render
change state
react component
interaction
actions (web-rpc-redux)
xhr
reducers
Frontend Server
Nginx
external
internal
authN 2.0 (optional)
openid-auth-middleware
HAProxy (loadbalancer)
CDN (Cloudfront)
Application Server
assets
API
pages
static server
web-rpc
react-router
renderToString
react-helmet
styletron-server
hydrated state
RPC Client
isorender.render()
Images, JS, etc.
Tchannel/Thrift
HTTP/JSON
HTML

React components for flexibility

Uber’s take on bootstrap

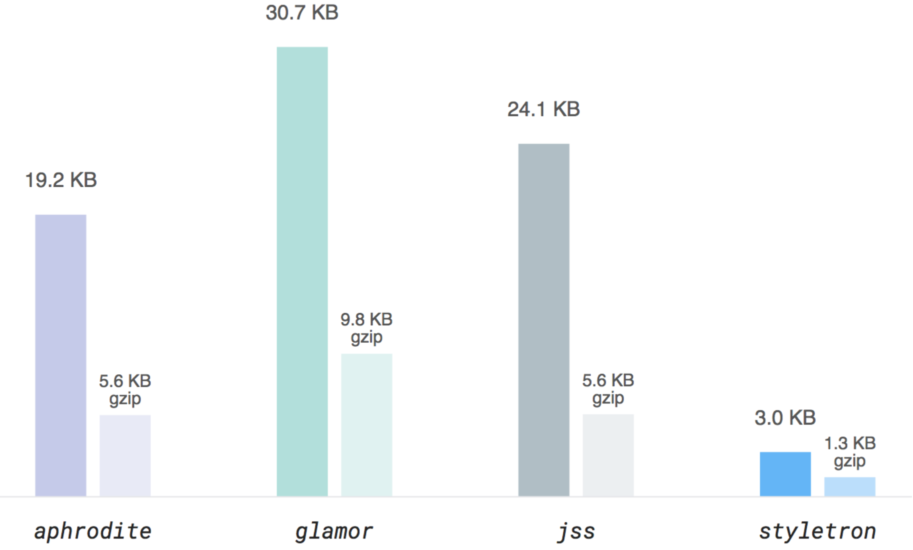
- Beautiful sites with flexible components you don’t have to maintain yourself
  - Standard UI components
  - Optimized for performance
  - Encapsulated style
- Consistent look and feel throughout all components that fit brand
- Analytics and instrumentation baked-in
- Seamless upgrades to new versions

CSS in JS with styletron

```
import Styletron from 'styletron';
import { injectStyle } from 'styletron-utils';

// Create a Styletron instance
const styletron = new Styletron();

const className = injectStyle(styletron, {
  color: 'red',
  display: 'inline-block',
  fontSize: '1.6em'
});
// Css is injected into the page
// and a class name is returned
```



How Uber uses JavaScript

Marketplace Platform (core services)

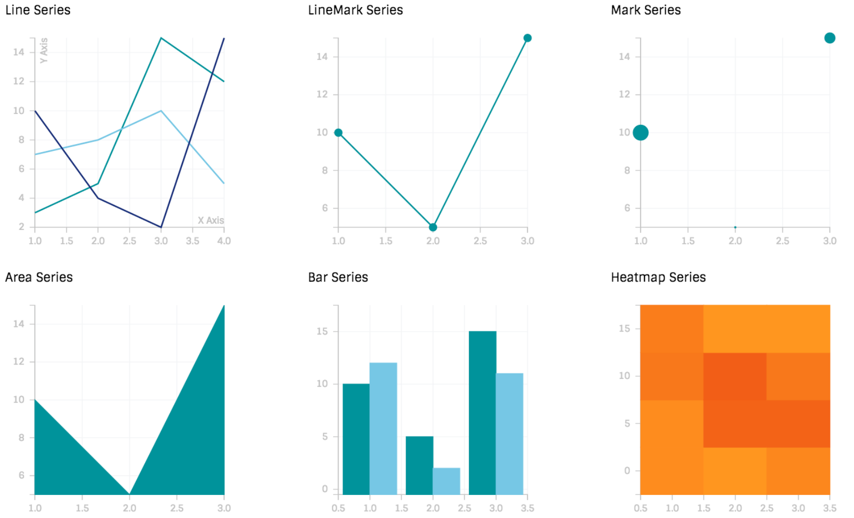
Web Platform (web presence)

Visualizing Data (everywhere)

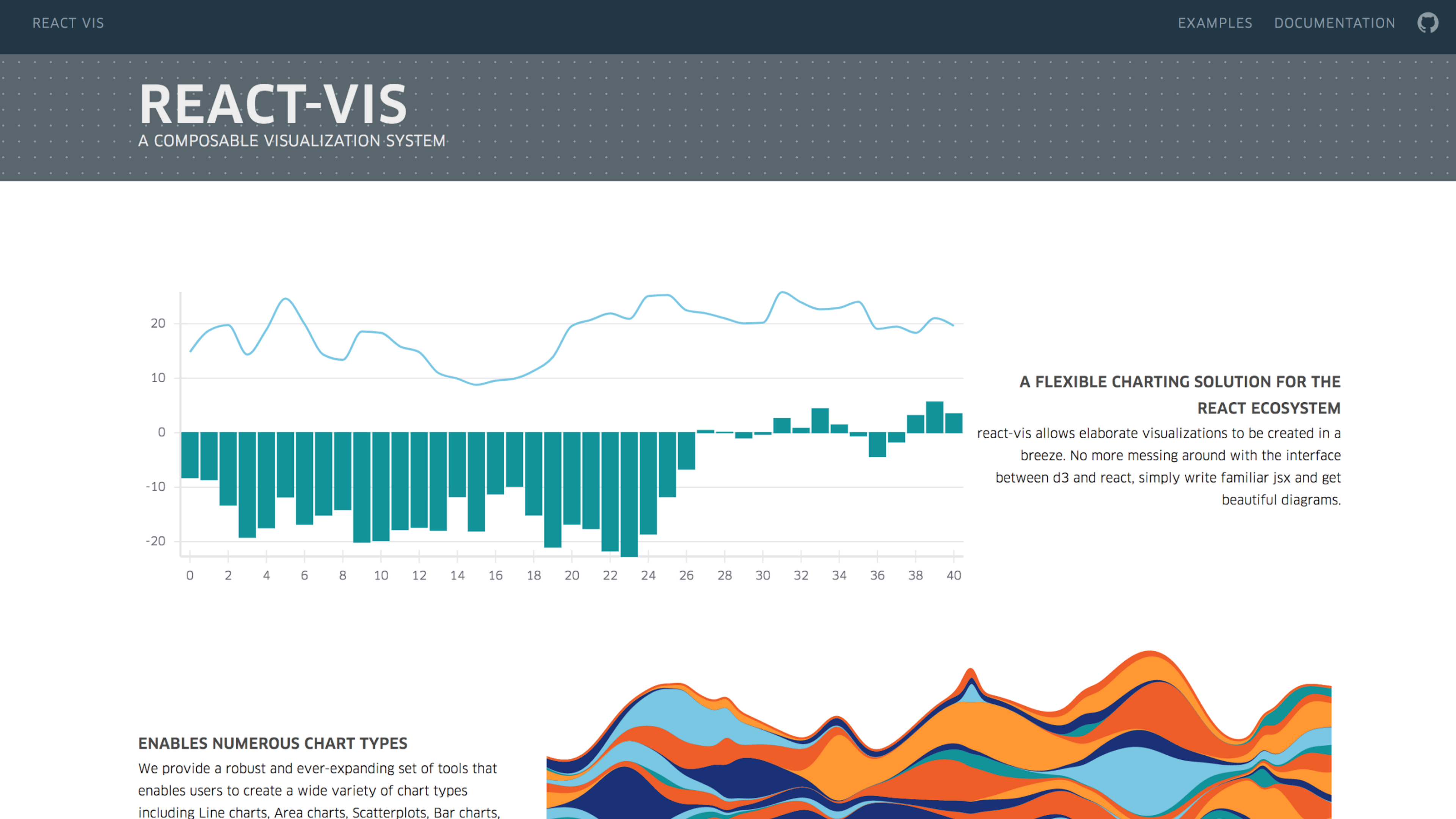
Developer Platform (external API)

Visualization Frameworks

- **react-vis:** Charts and Networks using D3 + React
- **react-map-gl:** A React interface to MapboxGL-js
- **deck.gl:** Layered WebGL approach system for visualization
- **luma.gl:** A JavaScript WebGL Framework for Data Visualization



SAN FRANCISCO
CHINATOWN
PACIFIC AVE
PINE ST
POST ST
HAIGHT-ASHBURY
16TH ST
MISSION DISTRICT
POTRERO HILL
NOE VALLEY
26TH ST
Presidio
Hunters Point
NORTH BEACH
NOB HILL
SACRAMENTO ST
SOUTH OF MARKET
SOUTH BEACH
WEST SOMA



REACT-VIS

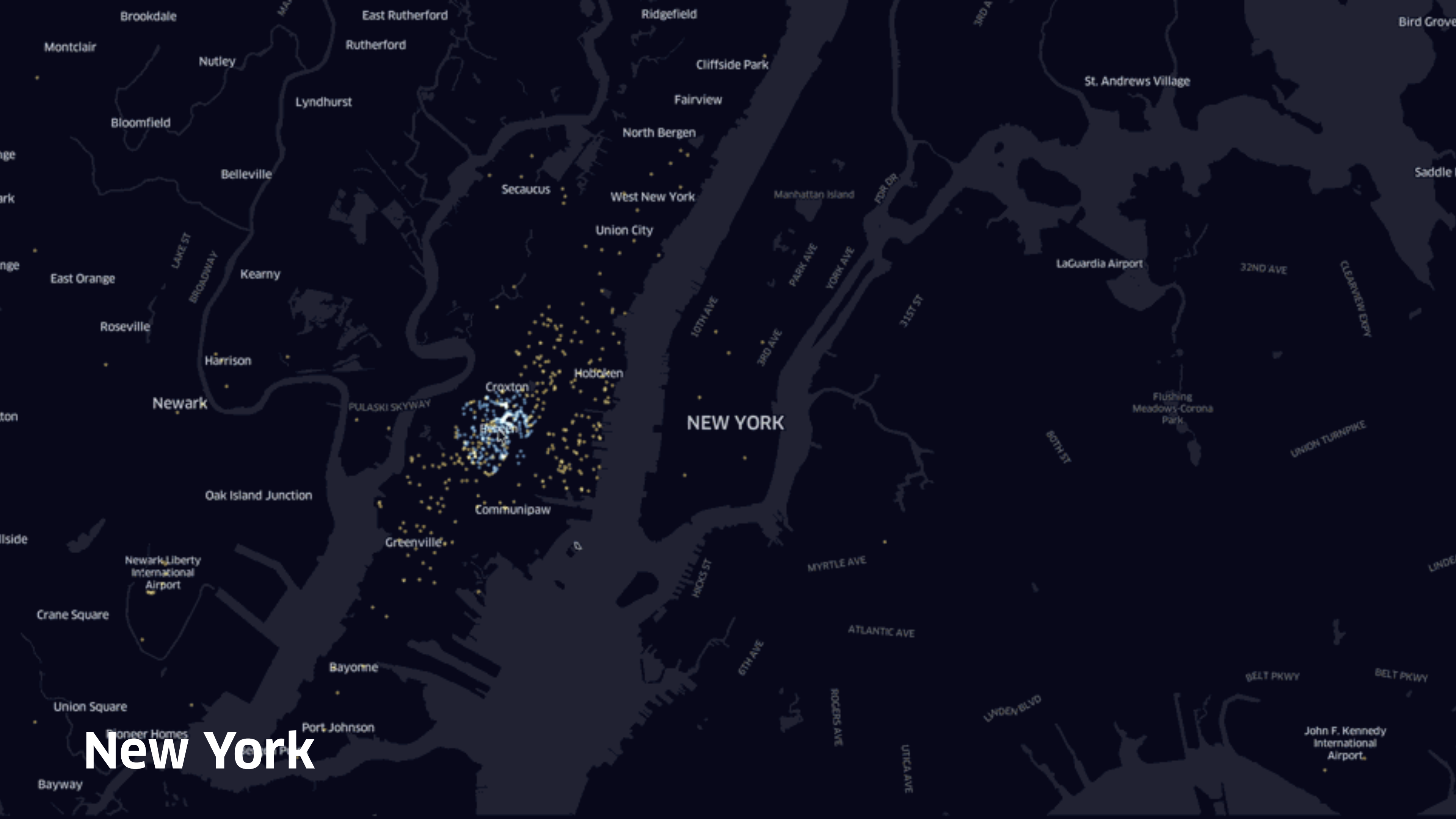
A COMPOSABLE VISUALIZATION SYSTEM

A FLEXIBLE CHARTING SOLUTION FOR THE REACT ECOSYSTEM

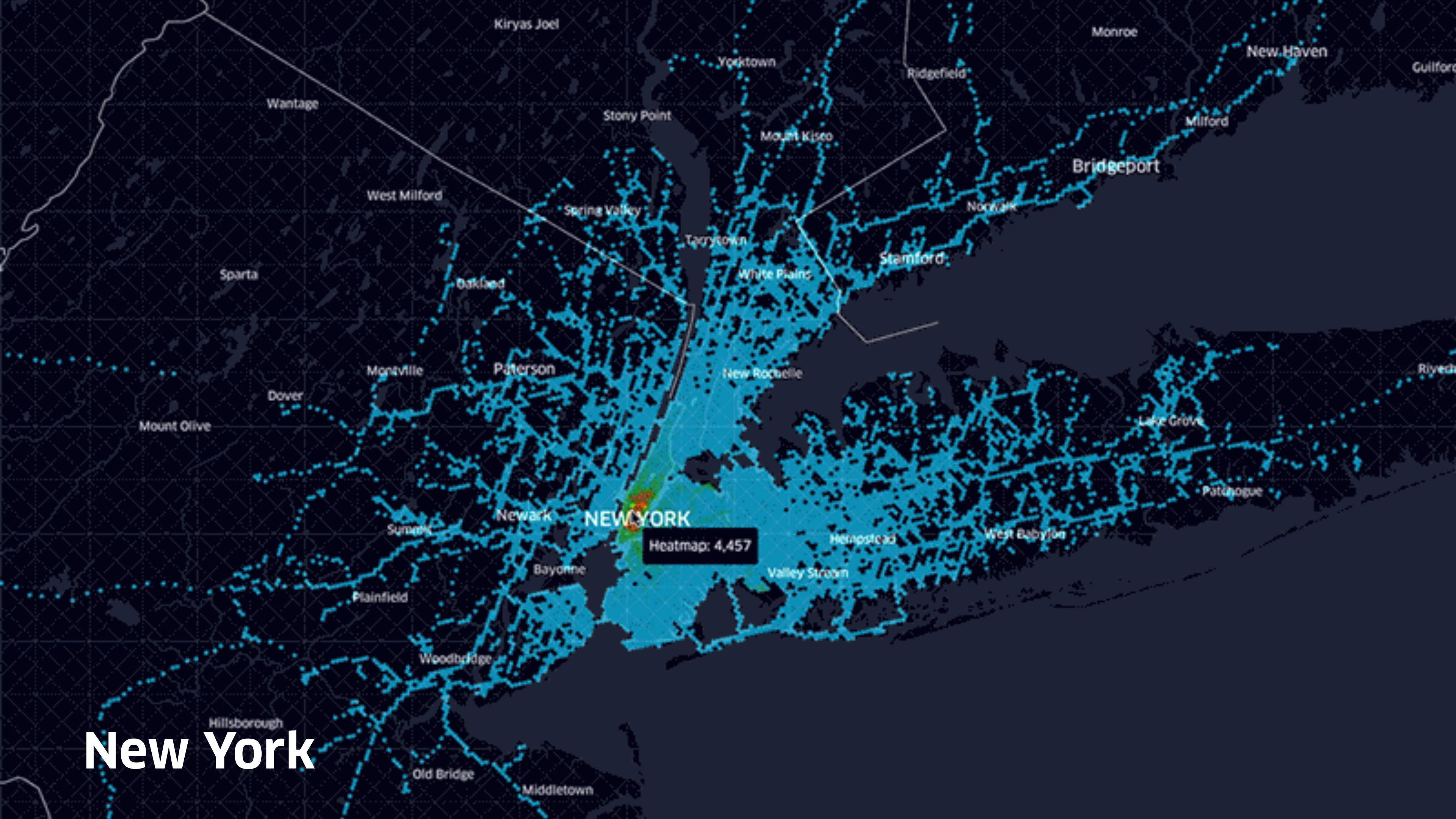
react-vis allows elaborate visualizations to be created in a breeze. No more messing around with the interface between d3 and react, simply write familiar jsx and get beautiful diagrams.

ENABLES NUMEROUS CHART TYPES

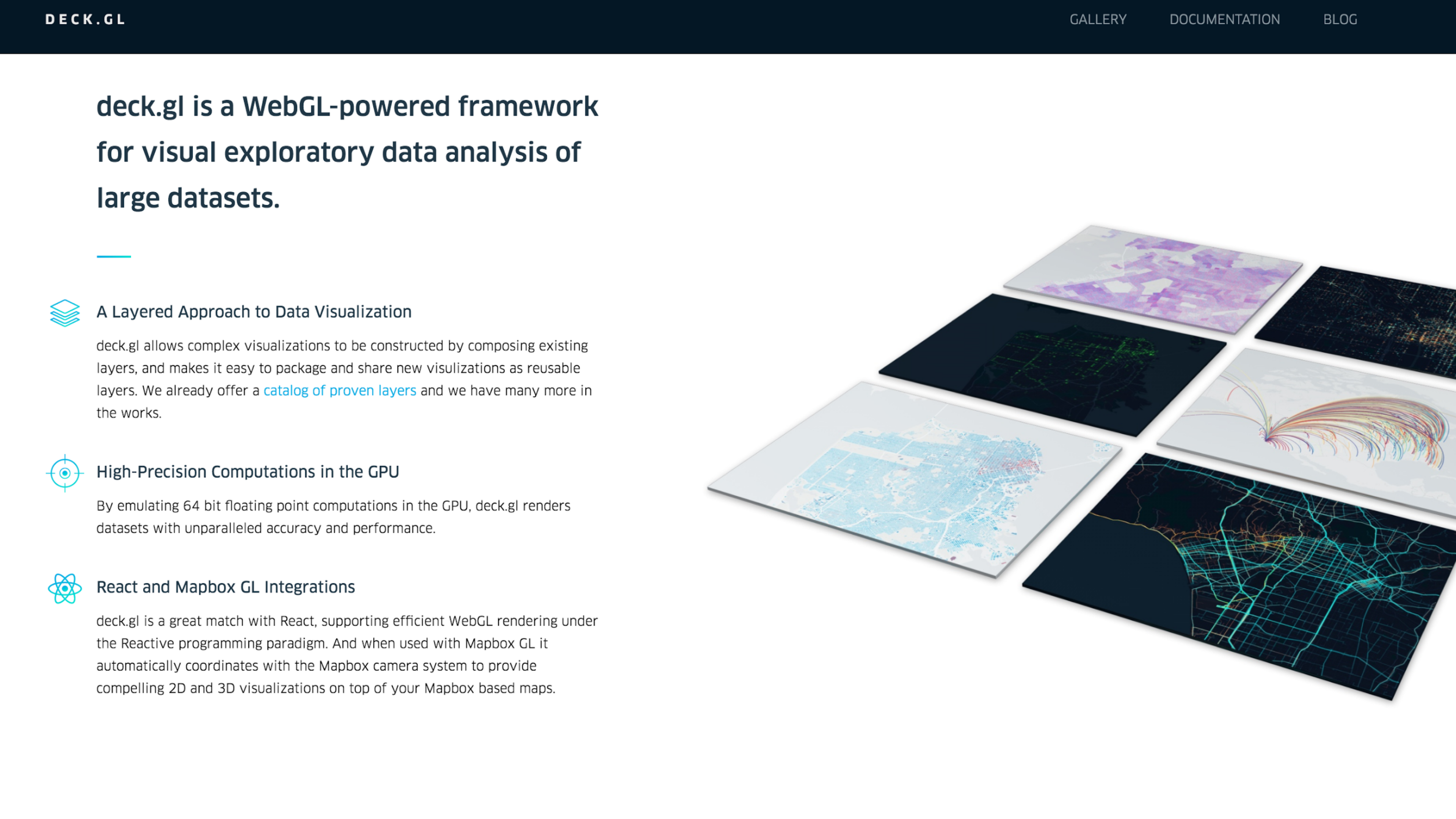
We provide a robust and ever-expanding set of tools that enables users to create a wide variety of chart types including Line charts, Area charts, Scatterplots, Bar charts,


Brookdale
East Rutherford
Ridgefield
Bird Grove
Montclair
Rutherford
Nutley
Cliffside Park
St. Andrews Village
Lyndhurst
Fairview
Bloomfield
North Bergen
Saddle
Belleville
Secaucus
West New York
Manhattan Island
Union City
LaGuardia Airport
East Orange
Kearny
Roseville
Harrison
Hoboken
Croxton
Newark
PULASKI SKYWAY
NEW YORK
Flushing Meadows-Corona Park
UNION TURNPIKE
Oak Island Junction
Communipaw
Greenville
Newark Liberty International Airport
MYRTLE AVE
Crane Square
ATLANTIC AVE
Bayonne
BELT PKWY
Union Square
Port Johnson
John F. Kennedy International Airport
Bayway

New York


Kiryas Joel
Yorktown
Monroe
New Haven
Guilford
Ridgefield
Wantage
Stony Point
Mount Kisco
Milford
Bridgeport
West Milford
Spring Valley
Norwalk
Tarrytown
Stamford
Sparta
White Plains
Oakland
Monroville
Paterson
New Rochelle
Riverhead
Dover
Mount Olive
Lake Grove
Patchogue
Newark
NEW YORK
Summit
Heatmap: 4,457
Hempstead
West Babylon
Bayonne
Valley Stream
Plainfield
Woodbridge
Hillsborough
Old Bridge
Middletown

New York



deck.gl is a WebGL-powered framework for visual exploratory data analysis of large datasets.

A Layered Approach to Data Visualization

deck.gl allows complex visualizations to be constructed by composing existing layers, and makes it easy to package and share new visulizations as reusable layers. We already offer a catalog of proven layers and we have many more in the works.

High-Precision Computations in the GPU

By emulating 64 bit floating point computations in the GPU, deck.gl renders datasets with unparalleled accuracy and performance.

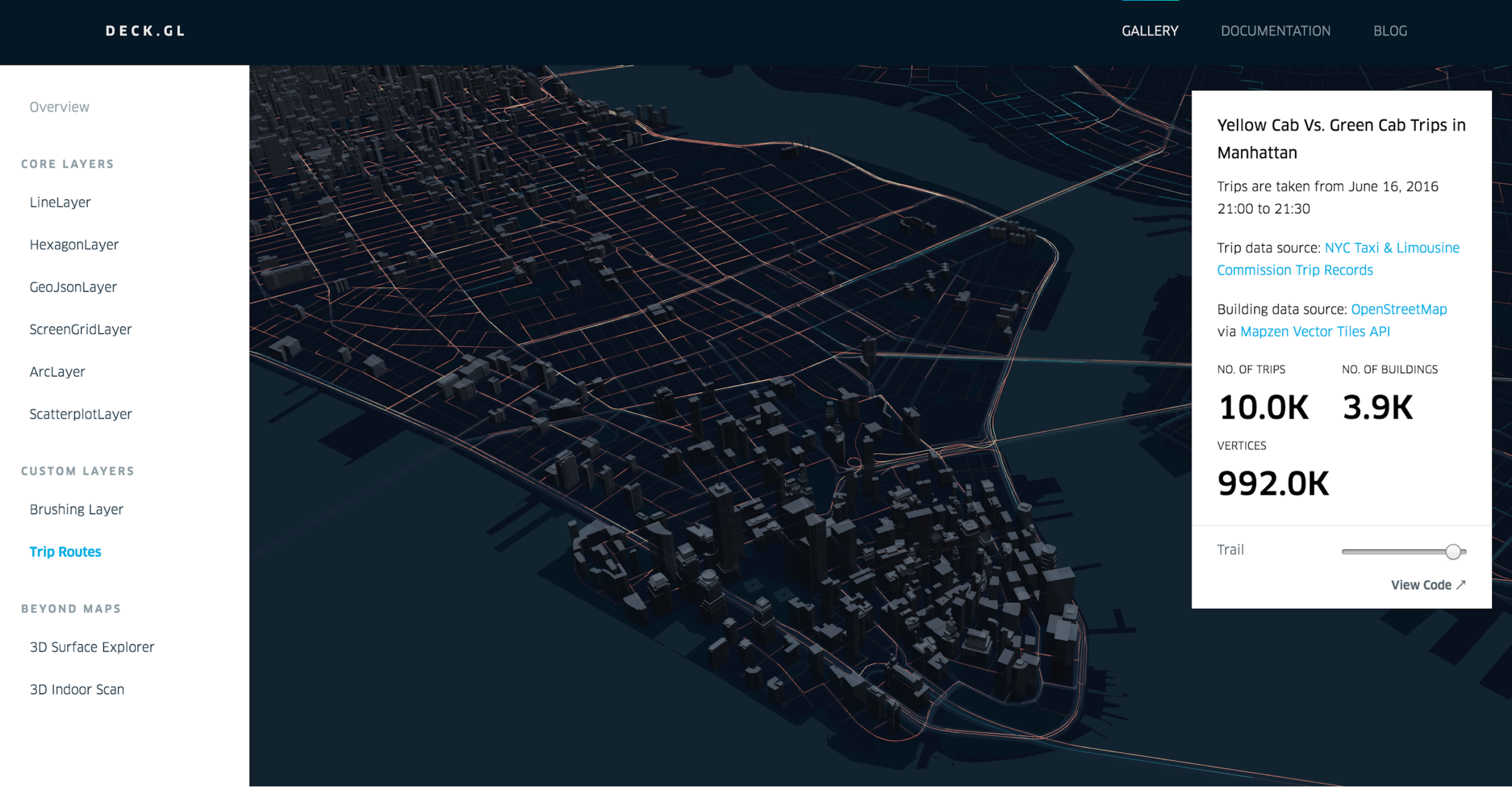
React and Mapbox GL Integrations

deck.gl is a great match with React, supporting efficient WebGL rendering under the Reactive programming paradigm. And when used with Mapbox GL it automatically coordinates with the Mapbox camera system to provide compelling 2D and 3D visualizations on top of your Mapbox based maps.

luma.gl

A JavaScript WebGL Framework for Data Visualization

- ES6, WebGL 2.0, component-based platform
- Interoperable with other popular libraries like stack.gl
- Shader library with a 64-bit floating point emulation package
- Advanced debugging, tracing, error checking for WebGL



Yellow Cab Vs. Green Cab Trips in Manhattan

Trips are taken from June 16, 2016 21:00 to 21:30

Trip data source: NYC Taxi & Limousine Commission Trip Records

Building data source: OpenStreetMap via Mapzen Vector Tiles API

NO. OF TRIPS	NO. OF BUILDINGS
10.0K	3.9K

VERTICES

992.0K

Trail

San Francisco

How Uber uses JavaScript

Marketplace Platform (core services)

Web Platform (web presence)

Visualizing Data (everywhere)

Developer Platform (external API)

Uber Developer Platform

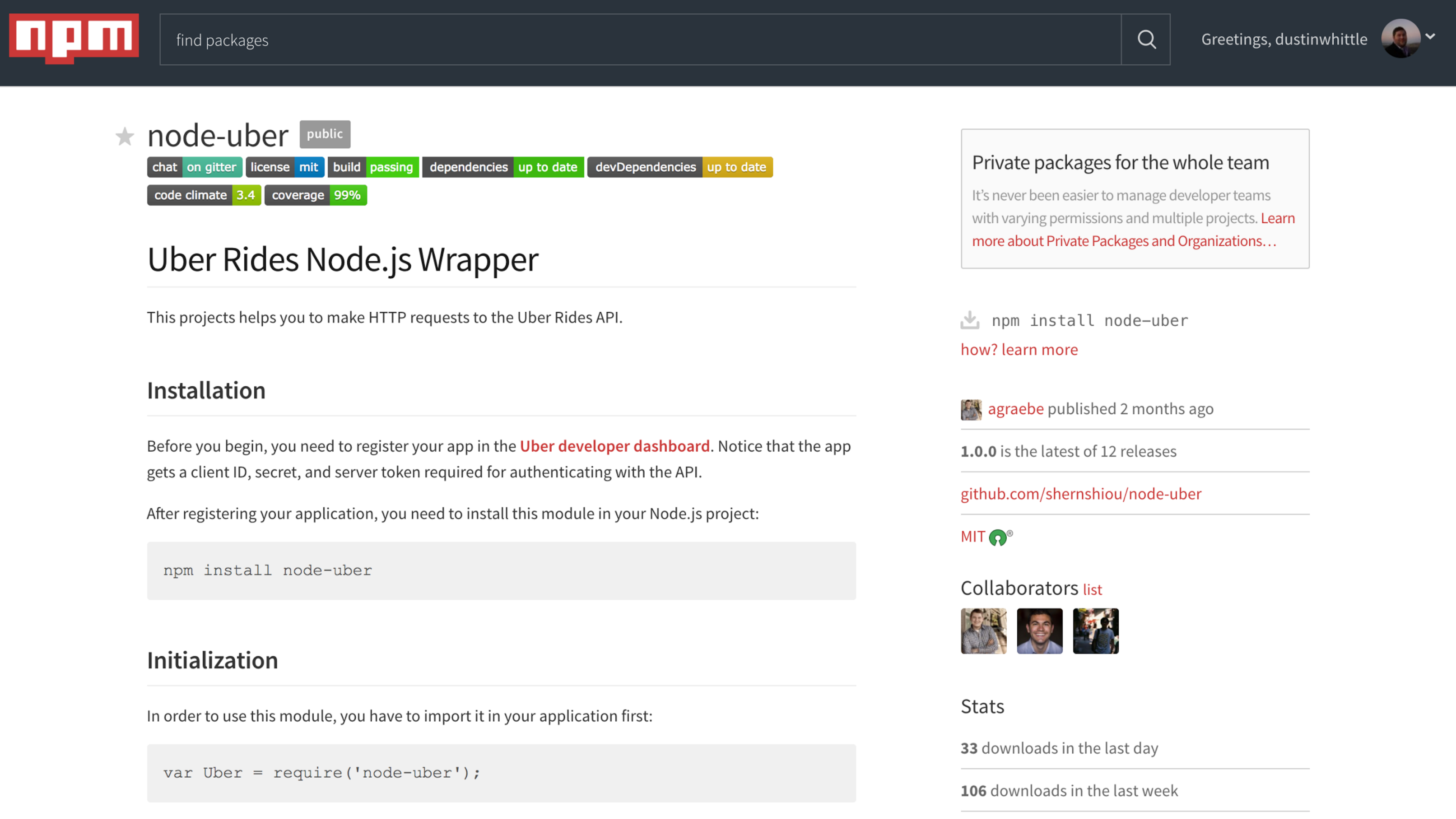
Enabling the world to build moving experiences with Uber

Ride Requests

Trip Experiences

Drivers LIMITED ACCESS

Deliveries



node-uber public

chat on gitter | license mit | build passing | dependencies up to date | devDependencies up to date | code climate 3.4 | coverage 99%

Uber Rides Node.js Wrapper

This projects helps you to make HTTP requests to the Uber Rides API.

Installation

Before you begin, you need to register your app in the **Uber developer dashboard**. Notice that the app gets a client ID, secret, and server token required for authenticating with the API.

After registering your application, you need to install this module in your Node.js project:

```
npm install node-uber
```

Initialization

In order to use this module, you have to import it in your application first:

```
var Uber = require('node-uber');
```

```
npm install node-uber
```

how? learn more

agraebe published 2 months ago

1.0.0 is the latest of 12 releases

github.com/shernshiou/node-uber

MIT

Collaborators list

Stats

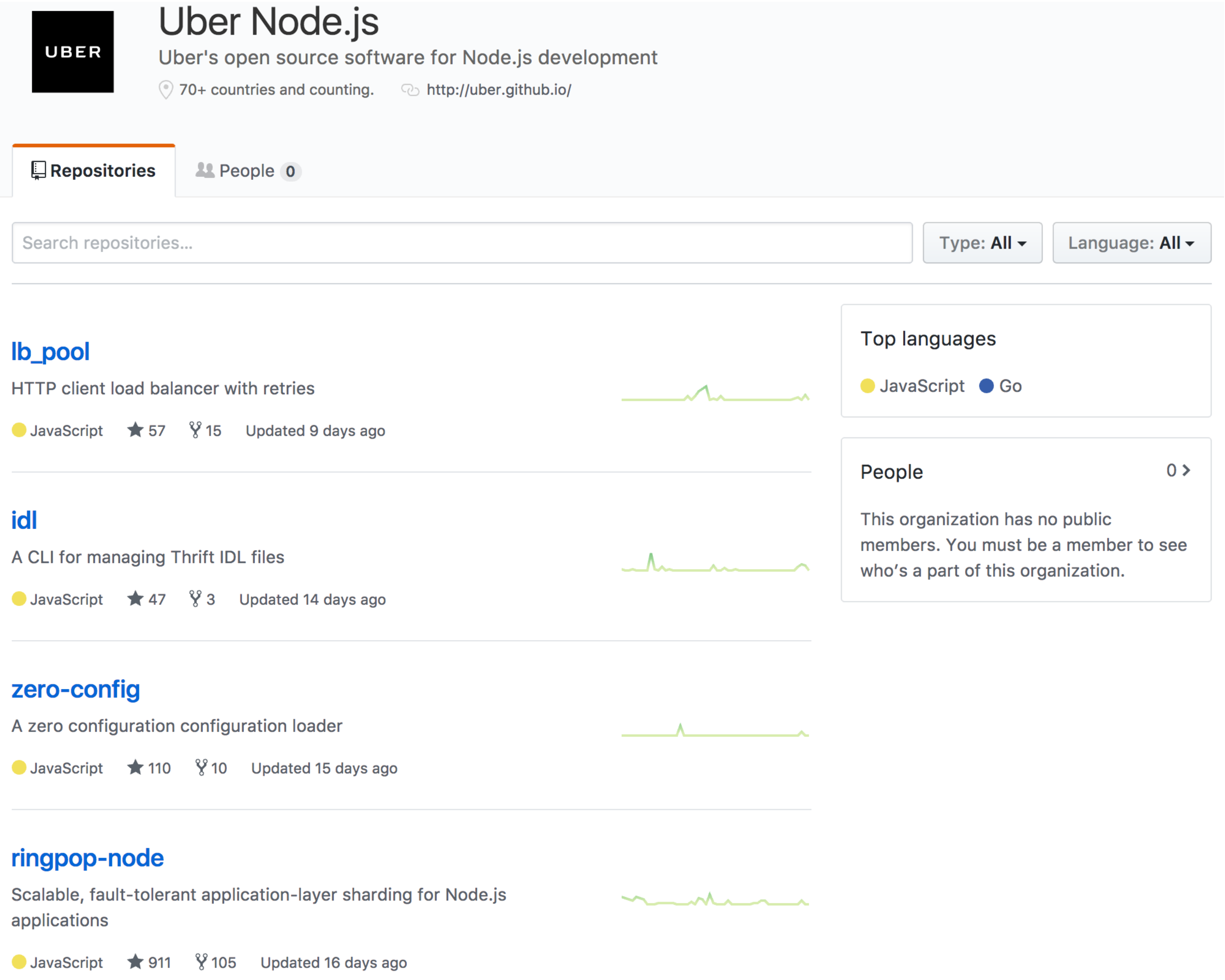
33 downloads in the last day

106 downloads in the last week

UBER Engineering

Open Source at Uber

Uber loves open source and contributing to the open source community.



UBER

Uber Node.js

Uber's open source software for Node.js development

70+ countries and counting. http://uber.github.io/

Search repositories...

lb_pool

HTTP client load balancer with retries

● JavaScript ★ 57 ⑂ 15 Updated 9 days ago

idl

A CLI for managing Thrift IDL files

● JavaScript ★ 47 ⑂ 3 Updated 14 days ago

zero-config

A zero configuration configuration loader

● JavaScript ★ 110 ⑂ 10 Updated 15 days ago

ringpop-node

Scalable, fault-tolerant application-layer sharding for Node.js applications

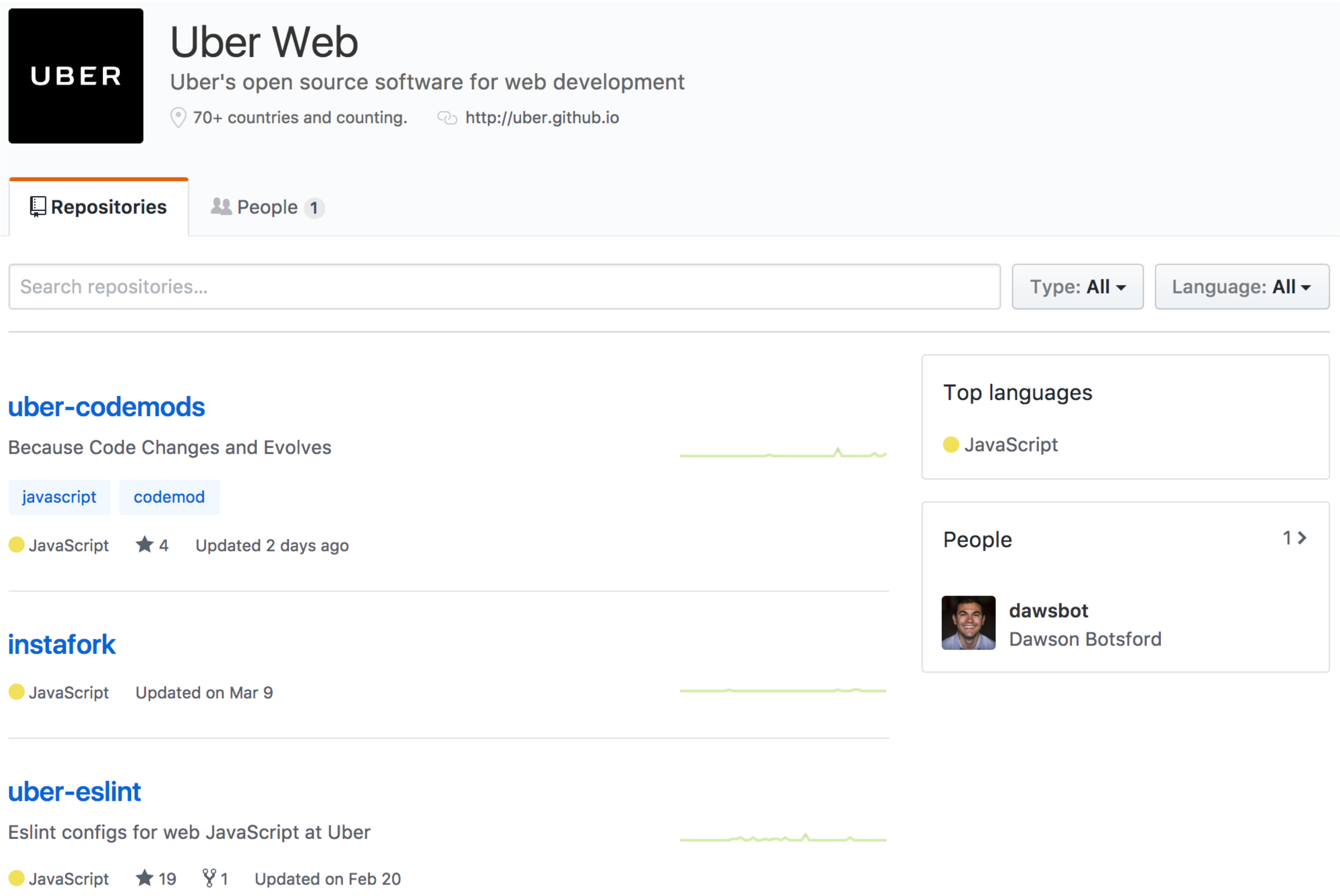
● JavaScript ★ 911 ⑂ 105 Updated 16 days ago

Top languages

● JavaScript ● Go

People 0 ›

This organization has no public members. You must be a member to see who’s a part of this organization.



Uber Web

Uber's open source software for web development

70+ countries and counting. http://uber.github.io

uber-codemods

Because Code Changes and Evolves

javascript codemod

● JavaScript ★ 4 Updated 2 days ago

instafork

● JavaScript Updated on Mar 9

uber-eslint

Eslint configs for web JavaScript at Uber

● JavaScript ★ 19 1 Updated on Feb 20

Top languages

● JavaScript

People 1 ›

dawsbot
Dawson Botsford

UBER Engineering

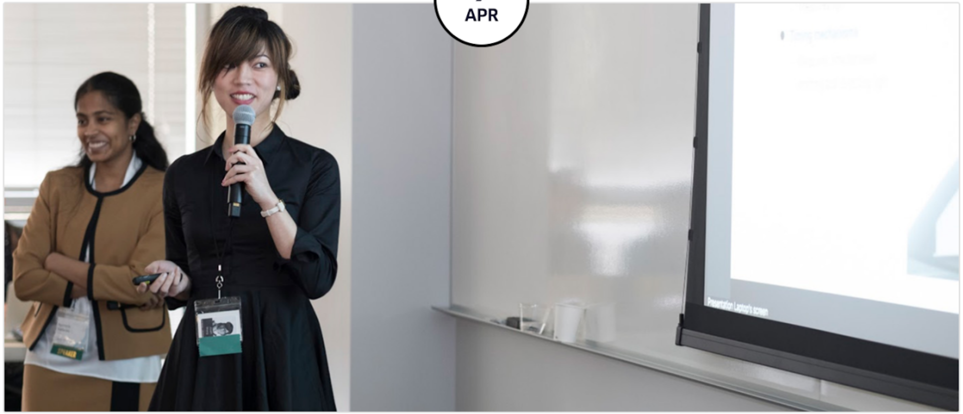
JOIN THE TEAM

MEET THE PEOPLE

SEARCH

Uber Engineering Blog

7
APR



PRESENTING THE ENGINEERING BEHIND UBER AT OUR TECHNOLOGY DAY

An open platform for Building Moving Experiences

What future will **you** build?

https://developers.uber.com

Credits

All of the content from this presentation comes from other talks by engineers far smarter than myself

- **Lessons learned from scaling to 2000 engineers and 1000+ services** - Matt Ranney
- **Uber Architecture: Moving Bits and Atoms at Scale** - Andrii Iasynetskyi
- **Thanks to the many amazing people of the Marketplace Platform, Web Platform, Developer Platform, and Visualization Teams at Uber!**
- **Come join us and work with these amazing people. Hiring globally (San Francisco, Amsterdam, Sofia, ...)**

Questions?

Thanks!