

The background of the slide is a photograph taken from an aerial perspective, showing a vast landscape of rolling hills and fields under a dramatic sky. The sun is low on the horizon, casting a warm, golden glow over the terrain. The colors range from deep blues and purples in the upper atmosphere to bright yellows and oranges near the horizon. The foreground shows more detailed textures of the land.

从REST到GraphQL

2017 JavaScript 中国开发者大会

7月15 - 16日@上海



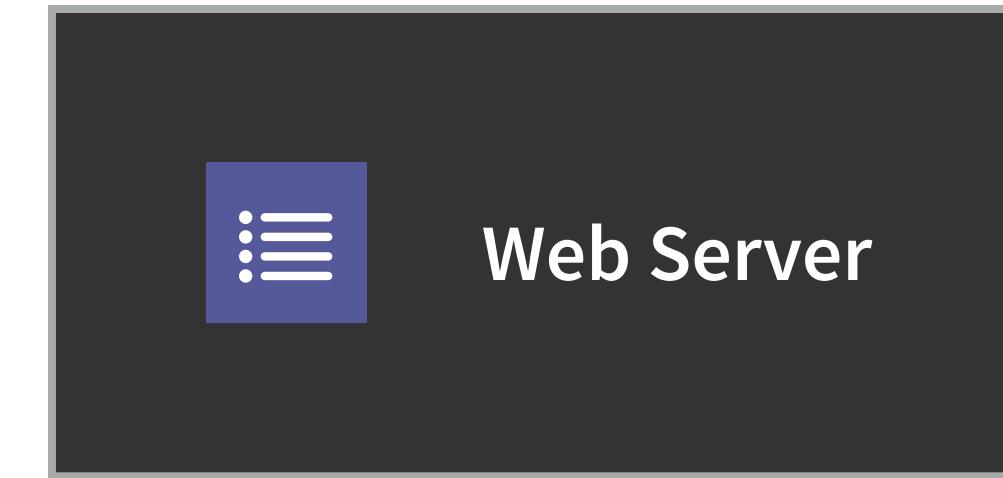
Jonas Helfer
@helferjs
Software Engineer, Databricks



Part 1: Theory

REST vs GraphQL

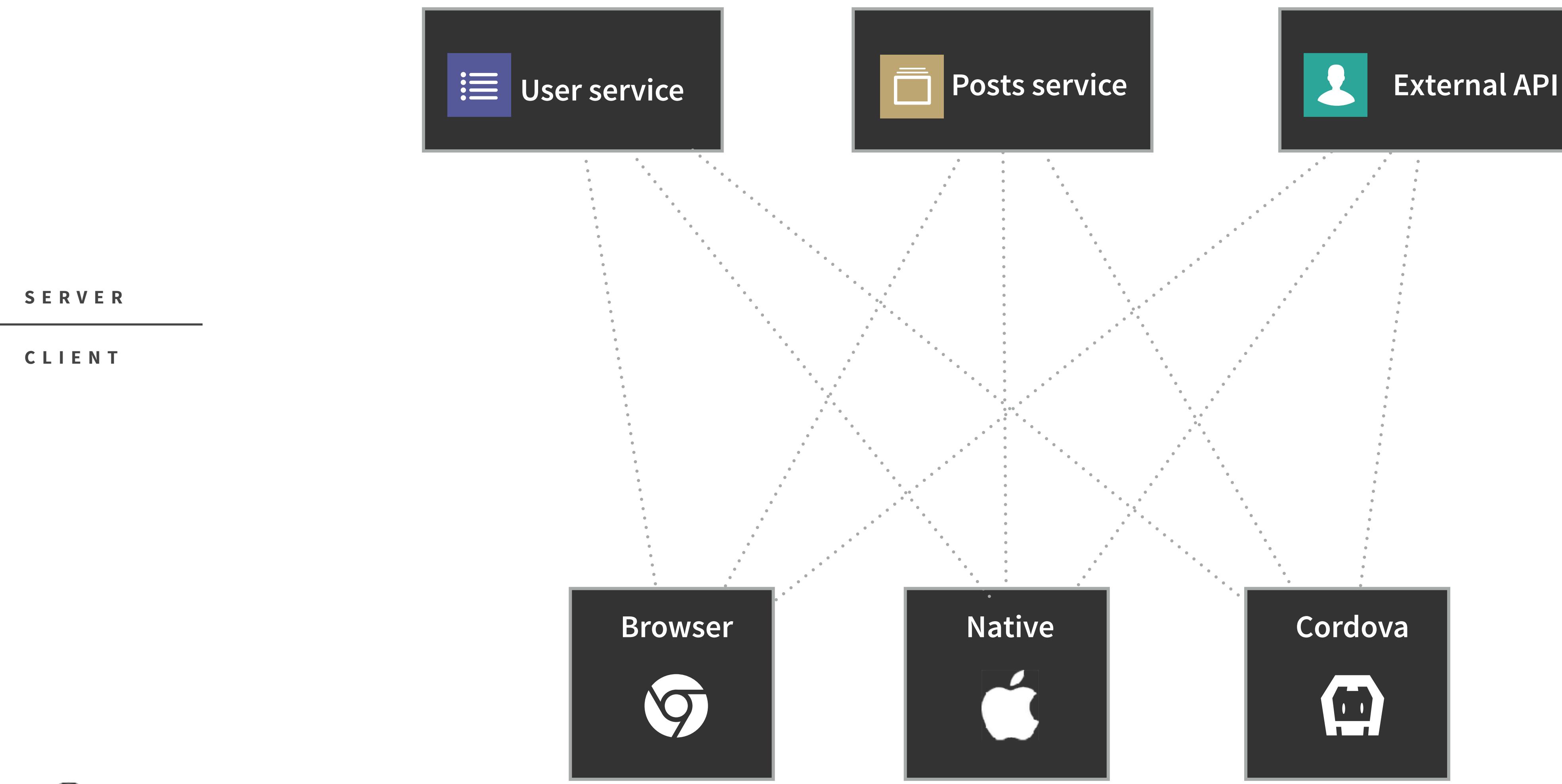
Web development in the early days



↓
`<html>`

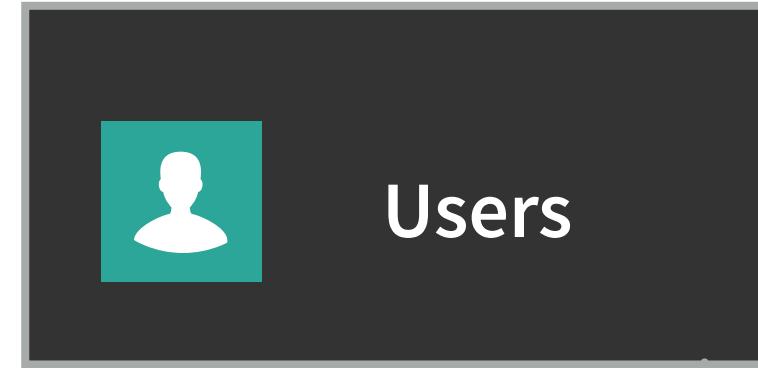


Web development in 2017

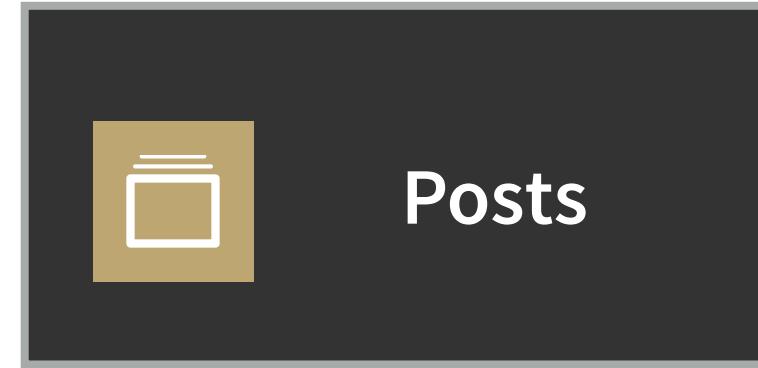


Pure REST - one endpoint per resource

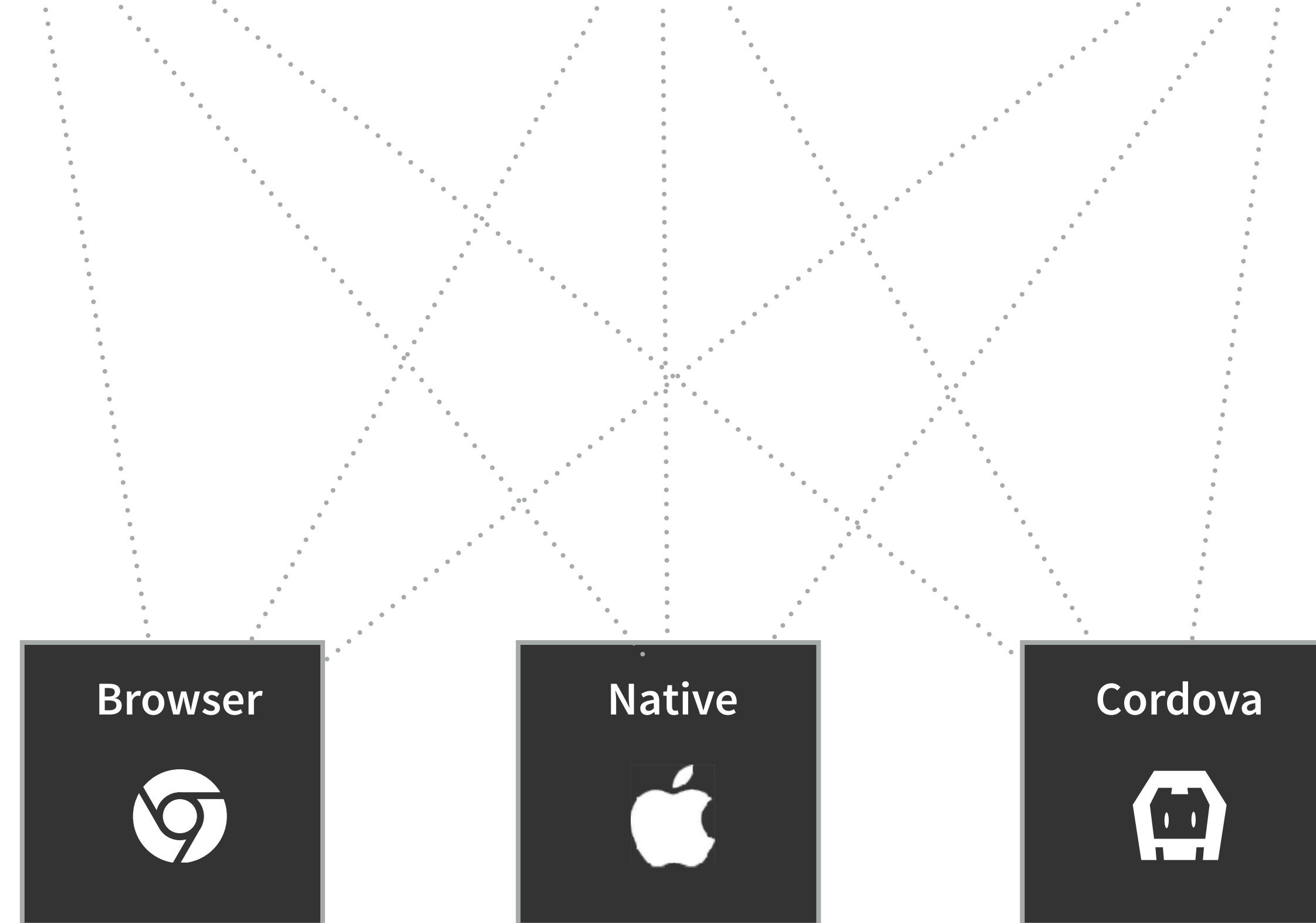
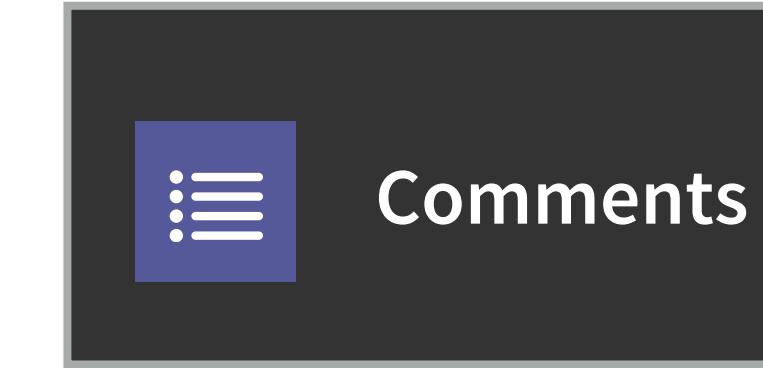
Endpoint 1



Endpoint 2



Endpoint 3



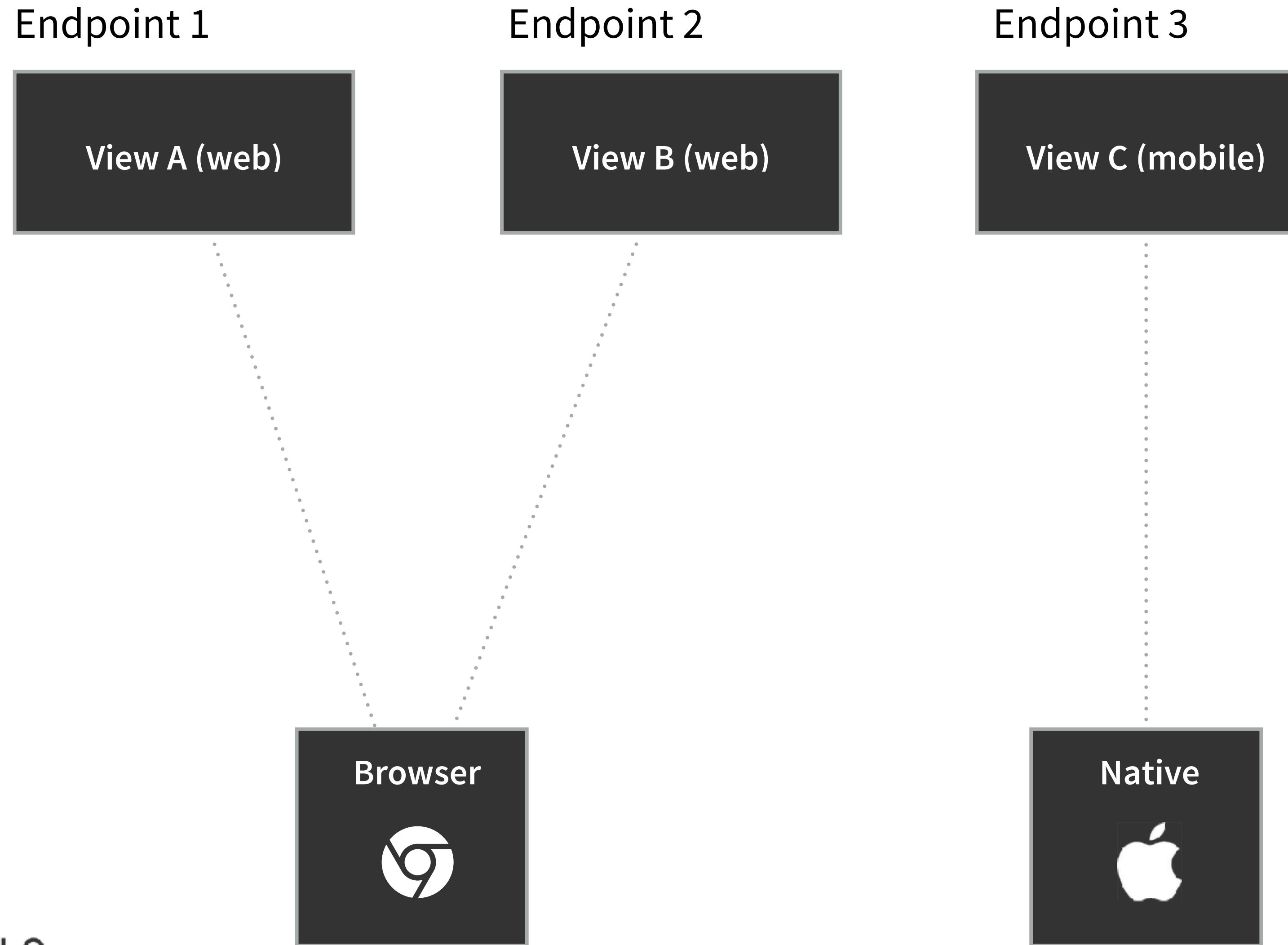
Pros

- flexible
- decoupled

Cons

- lots of roundtrips
- overfetching
- complex clients

REST-ish - one endpoint per view



Pros

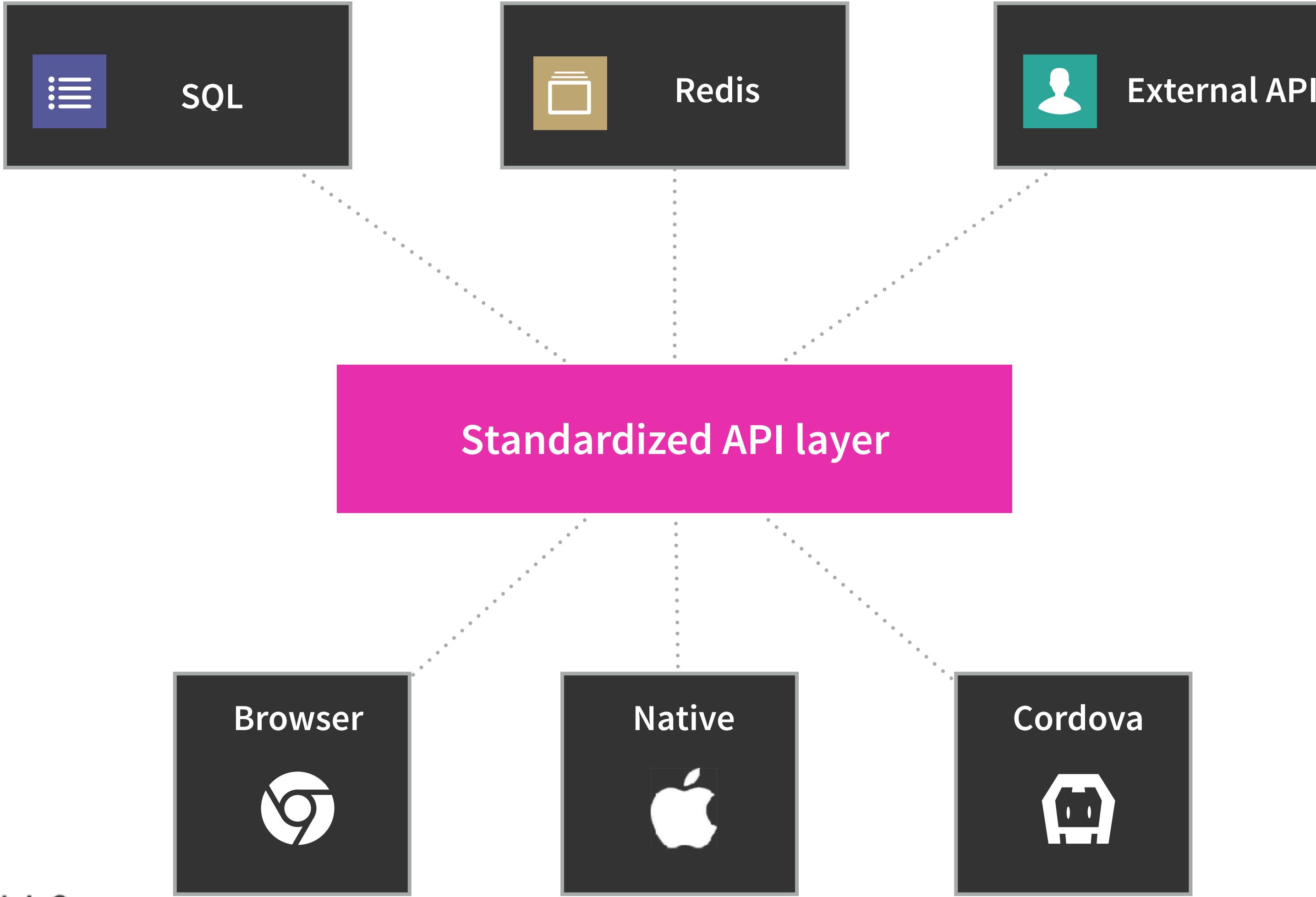
- one roundtrip
- exactly what you need

Cons

- inflexible
- high initial cost
- high maintenance cost
- slow to iterate



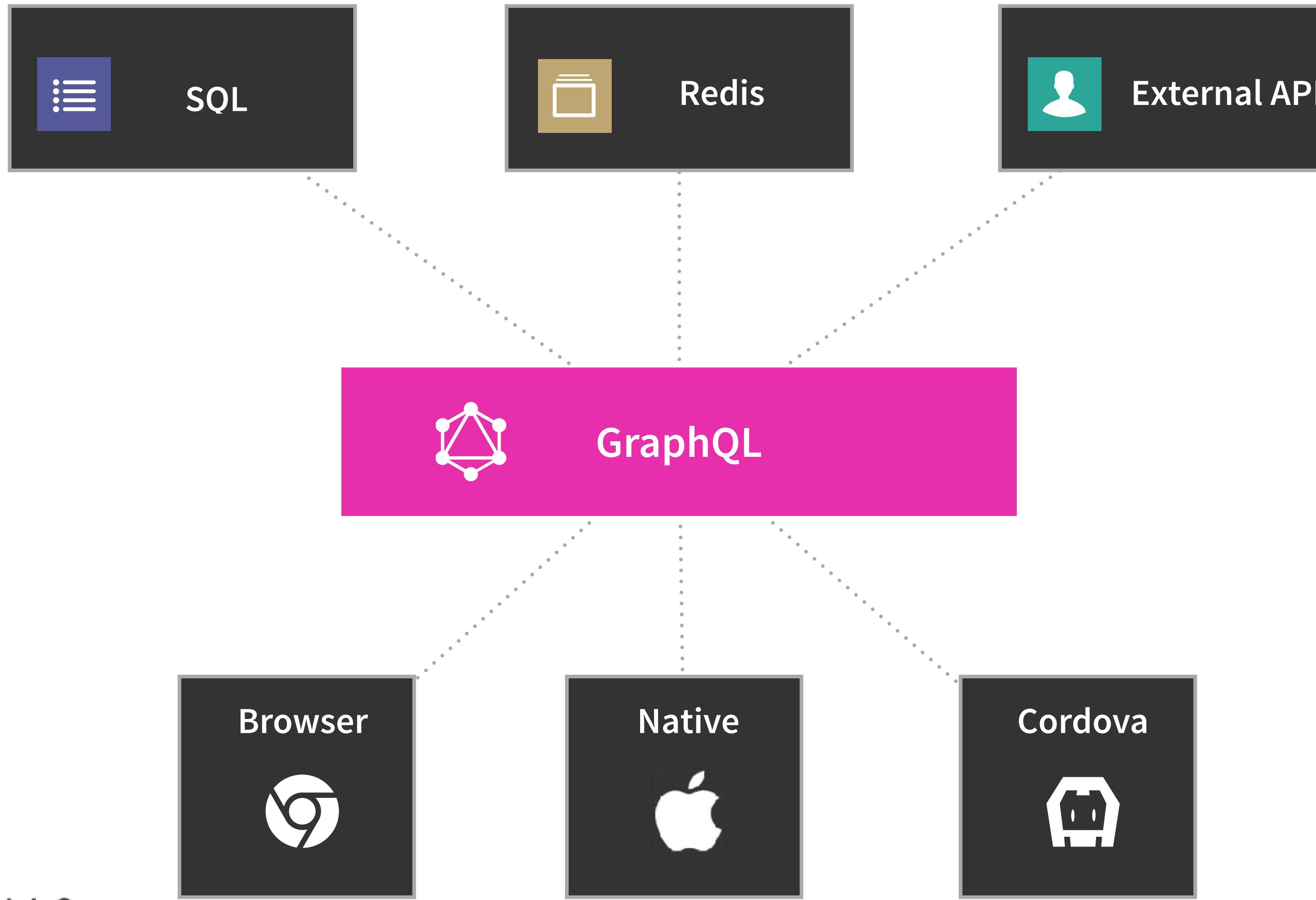
We need a standardized API layer



Requirements

- one roundtrip
- get just what you need
- flexible
- decoupled

GraphQL could be that standardized API layer



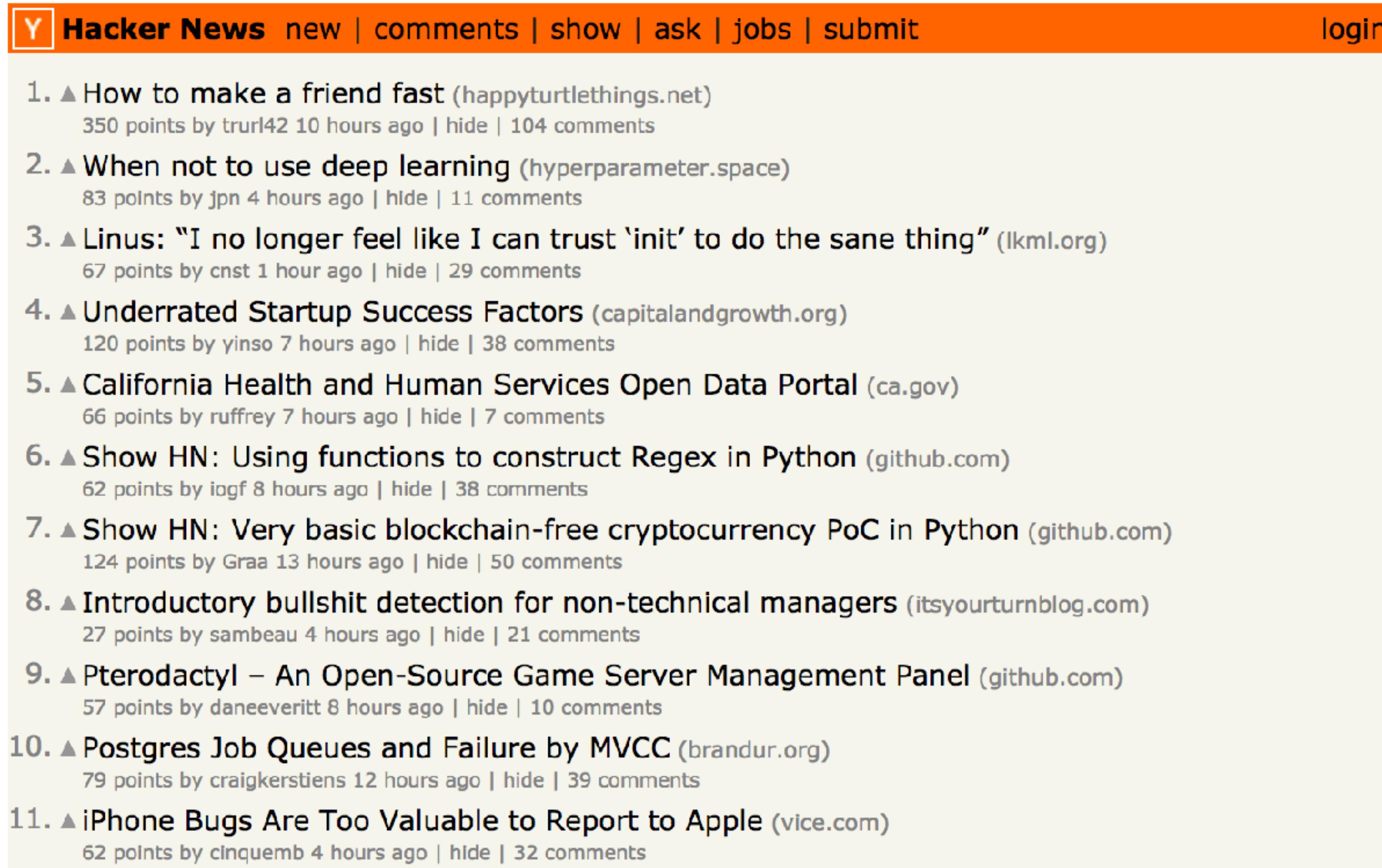
What GraphQL gives us

- one roundtrip
- get just what you need
- decoupling

GraphQL provides three main things

1. an API definition language to describe types and their relationships
2. a query language to specify exactly what data you need
3. an execution model that allows “resolving” single fields

Let's use an example: Hacker News



The screenshot shows the Hacker News homepage with the following 11 stories:

1. ▲ How to make a friend fast (happyturtlenthings.net)
350 points by trurl42 10 hours ago | hide | 104 comments
2. ▲ When not to use deep learning (hyperparameter.space)
83 points by jpn 4 hours ago | hide | 11 comments
3. ▲ Linus: "I no longer feel like I can trust 'init' to do the sane thing" (lkml.org)
67 points by cnst 1 hour ago | hide | 29 comments
4. ▲ Underrated Startup Success Factors (capitalandgrowth.org)
120 points by yinso 7 hours ago | hide | 38 comments
5. ▲ California Health and Human Services Open Data Portal (ca.gov)
66 points by ruffrey 7 hours ago | hide | 7 comments
6. ▲ Show HN: Using functions to construct Regex in Python (github.com)
62 points by iogf 8 hours ago | hide | 38 comments
7. ▲ Show HN: Very basic blockchain-free cryptocurrency PoC in Python (github.com)
124 points by Graa 13 hours ago | hide | 50 comments
8. ▲ Introductory bullshit detection for non-technical managers (itsyourturnblog.com)
27 points by sambeau 4 hours ago | hide | 21 comments
9. ▲ Pterodactyl – An Open-Source Game Server Management Panel (github.com)
57 points by daneeveritt 8 hours ago | hide | 10 comments
10. ▲ Postgres Job Queues and Failure by MVCC (brandur.org)
79 points by craigkerstiens 12 hours ago | hide | 39 comments
11. ▲ iPhone Bugs Are Too Valuable to Report to Apple (vice.com)
62 points by cinquemb 4 hours ago | hide | 32 comments

A simple example: schema

```
type Story {  
    title: String  
    author: Author  
    comments: [Comment]  
    score: Int  
}
```

```
type Author {  
    name: String  
}
```

...

A simple example: query

```
query {  
  stories(limit: 3) {  
    title  
    comments {  
      totalCount  
    }  
    author {  
      name  
    }  
  }  
}
```

A simple example: resolver function

```
function stories(root, args) {  
  return StoriesCollection.find()  
    .sort({ score: -1 })  
    .limit(args.limit);  
}
```

GraphQL ~= REST

GraphQL

Schema

Resolvers

Query / Mutation

REST

-

Endpoints

GET / POST request (URL)

GraphQL: a specification, not an implementation

S E R V E R S

- Node.js
- Ruby
- Python
- Scala
- Java
- Erlang

C L I E N T S

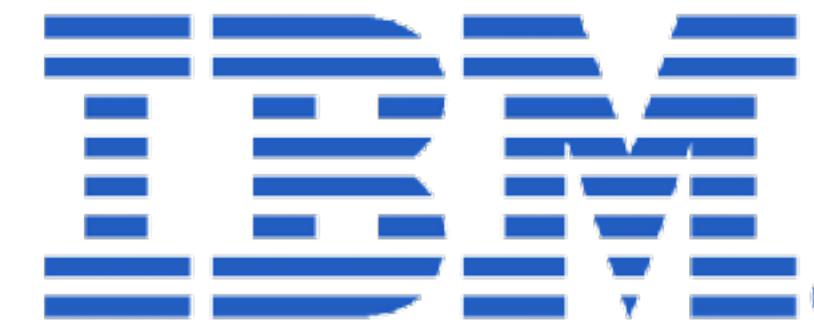
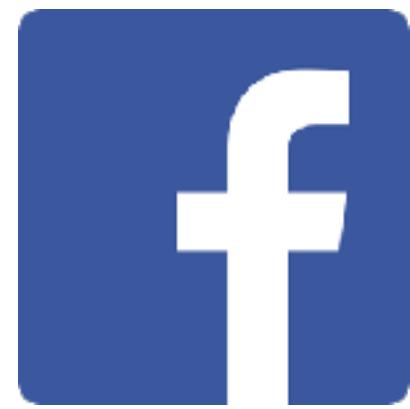
- Plain HTTP request
- React
- Angular
- Vue
- Polymer
- Native iOS/Android

T O O L S

- API explorer (GraphiQL)
- Editor autocomplete
- Query validation
- Mocking
- Performance analysis
- Code generation

...and more

Who is using GraphQL?



intuit.



Product Hunt

ticketmaster



credit karma™



@helferjs



Part 2 : Practice
Let's try a real example!

Hacker News - news.ycombinator.com

Y **Hacker News** new | comments | show | ask | jobs | submit [login](#)

1. ▲ **How to make a friend fast** (happyturtlenthings.net)
350 points by [trurl42](#) 10 hours ago | hide | 104 comments
2. ▲ **When not to use deep learning** (hyperparameter.space)
83 points by [jpn](#) 4 hours ago | hide | 11 comments
3. ▲ **Linus: "I no longer feel like I can trust 'init' to do the sane thing"** (lkml.org)
67 points by [cnst](#) 1 hour ago | hide | 29 comments
4. ▲ **Underrated Startup Success Factors** (capitalandgrowth.org)
120 points by [yinso](#) 7 hours ago | hide | 38 comments
5. ▲ **California Health and Human Services Open Data Portal** (ca.gov)
66 points by [ruffrey](#) 7 hours ago | hide | 7 comments
6. ▲ **Show HN: Using functions to construct Regex in Python** (github.com)
62 points by [iogf](#) 8 hours ago | hide | 38 comments
7. ▲ **Show HN: Very basic blockchain-free cryptocurrency PoC in Python** (github.com)
124 points by [Graa](#) 13 hours ago | hide | 50 comments
8. ▲ **Introductory bullshit detection for non-technical managers** (itsyourturnblog.com)
27 points by [sambeau](#) 4 hours ago | hide | 21 comments
9. ▲ **Pterodactyl – An Open-Source Game Server Management Panel** (github.com)
57 points by [daneeveritt](#) 8 hours ago | hide | 10 comments
10. ▲ **Postgres Job Queues and Failure by MVCC** (brandur.org)
79 points by [craigkerstiens](#) 12 hours ago | hide | 39 comments
11. ▲ **iPhone Bugs Are Too Valuable to Report to Apple** (vice.com)
62 points by [cinquemb](#) 4 hours ago | hide | 32 comments

Live exercise goals

Current API - HN REST API [<https://github.com/HackerNews/API>]

Final API - GraphQL API (on top of HN REST API)

Current App - A simple HN clone [<https://codesandbox.io/s/9n4kMx8x>]

Final App - Simple HN clone with GraphQL (replacing REST requests)

```
1 // Welcome to Launchpad!
2 // Log in to edit and save pads, and run queries in GraphQL on the right.
3
4
5 import {
6   makeExecutableSchema,
7   addMockFunctionsToSchema,
8 } from 'graphql-tools';
9
10 // Construct a schema the usual way, using GraphQL schema language
11 // This already contains all the information we need to create mocks
12 // of the correct types for any query.
13 const typeDefs = `
14   type User {
15     id: Int
16     name: String
17     posts(limit: Int): [Post]
18   }
19
20   type Post {
21     id: Int
22     title: String
23     views: Int
24     author: User
25   }
26
27   type Query {
28     aString: String
29     aBoolean: Boolean
30     anInt: Int
31     author(id: Int): User
32     topPosts(limit: Int): [Post]
33   }
34 `;
35
36 export const schema = makeExecutableSchema({ typeDefs });
37
38 const mocks = {
```

GraphQL ➤ Prettify

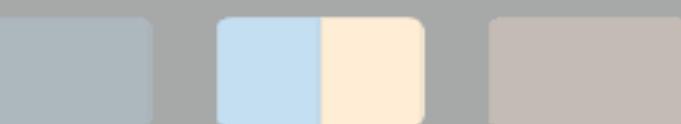
Save Reset Headers

```
1 # Welcome to GraphQL
2
3 # Click the play button above to run
4 # one of the example queries below.
5
6 query scalars {
7   anInt
8   aString
9   aBoolean
10}
11
12 query author {
13   author(id: 2){
14     id
15     name
16     posts(limit: 3) {
17       id
18       title
19     }
20   }
21 }
22
23 query topPosts {
24   topPosts(limit: 5) {
25     id
26     title
27     author {
28       id
29       name
30     }
31   }
32 }
```

QUERY VARIABLES

1

{
 "data": [
 "topPosts": [
 {
 "id": -55,
 "title": "Hello World",
 "author": {
 "id": 48,
 "name": "Hello World"
 }
 },
 {
 "id": 54,
 "title": "Hello World",
 "author": {
 "id": 51,
 "name": "Hello World"
 }
 }
]
]



Project

Title

HN REST example

Description

Example of loading data from the HN REST API for a React component

Author

[helper](#)

Statistics

0 likes

1 fork

1 view

Files

HN REST example

StoriesList.js

StoriesListWithData.js

index.html

index.js

/index.js

Prettify

Save



```
1 import React from 'react';
2 import { render } from 'react-dom';
3 import 'isomorphic-fetch';
4 import StoriesListWithData from './StoriesListWithData';
5
6 const styles = {
7   fontFamily: 'sans-serif',
8   textAlign: 'center',
9 };
10
11 const App = () =>
12   <div style={styles}>
13     <StoriesListWithData />
14   </div>;
15
16 render(<App />, document.getElementById('root'))
```

codesandbox.io

1. [How to make a friend fast](#) by trurl42
2. [Linus: "I no longer feel like I can trust 'init' to do the sane thing"](#) by cnst
3. [When not to use deep learning](#) by jpn
4. [Underrated Startup Success Factors](#) by yinso
5. [California Health and Human Services Open Data Portal](#) by ruffrey
6. [Show HN: Using functions to construct Regex in Python](#) by iogf
7. [Show HN: Very basic blockchain-free cryptocurrency PoC in Python](#) by Graa
8. [Pterodactyl – An Open-Source Game Server Management Panel](#) by daneeveritt
9. [Introductory bullshit detection for non-technical managers](#) by sambeau
10. [Categories, Job Durations and Failure by MVCC](#) by craigkerstiens
11. [iPhone Bugs Are Too Valuable to Report to Apple](#) by cinquemb
12. [Using chip memory more efficiently](#) by redtuesday
13. [Ten years after first shipping Openmoko Neo1973](#) by richardboegli
14. [Binary co-founder blames "corrupted" media for his resignation offer](#) by craigkerstiens
15. [Counterintuitive problem: People in a room keep giving dollars to random others](#) by aqsalose
16. [Porting NetBSD to Allwinner H3 SoCs](#) by zdw
17. [Sea Peoples](#) by bane
18. [A \(Detailed\) Elliptic Curve Cryptography Tutorial](#) by 68c12c16
19. [How startups such as Dropbox, Airbnb, Groupon acquired their first users](#) by catherinezng
20. [Show HN: Console based email client with Lua scripting](#) by stevekemp

Get the links on my GitHub



<https://github.com/helper/js-conf-china>



Part 3: The future

Advantages of GraphQL

1. No overfetching / underfetching
2. Really easy documentation
3. Strong typing

Open questions

1. No standard caching solutions (REST has HTTP caching)
2. No standard for error handling (REST has HTTP has status codes)
3. GraphQL is still evolving

Where to go from here

graphql.com

graphql.org

dev-blog.apollodata.com



www.apollodata.com



Jonas Helfer
@helperjs



We're hiring! Visit **meteor.io** for details.

@helperjs