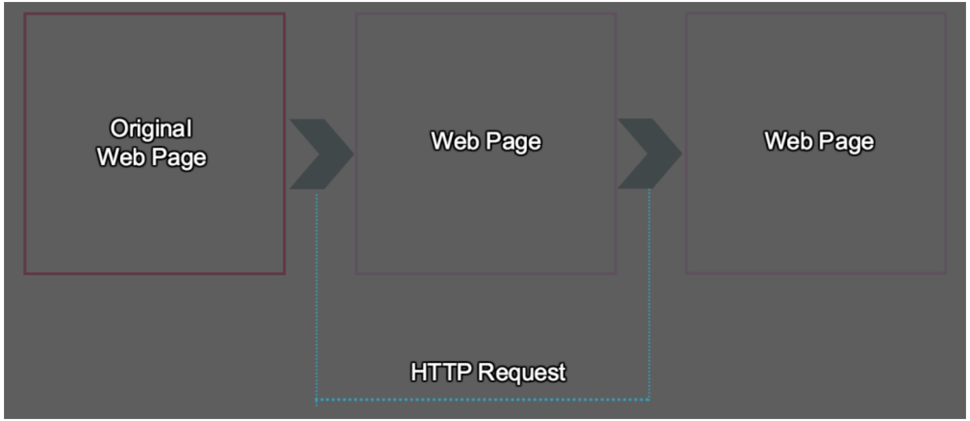
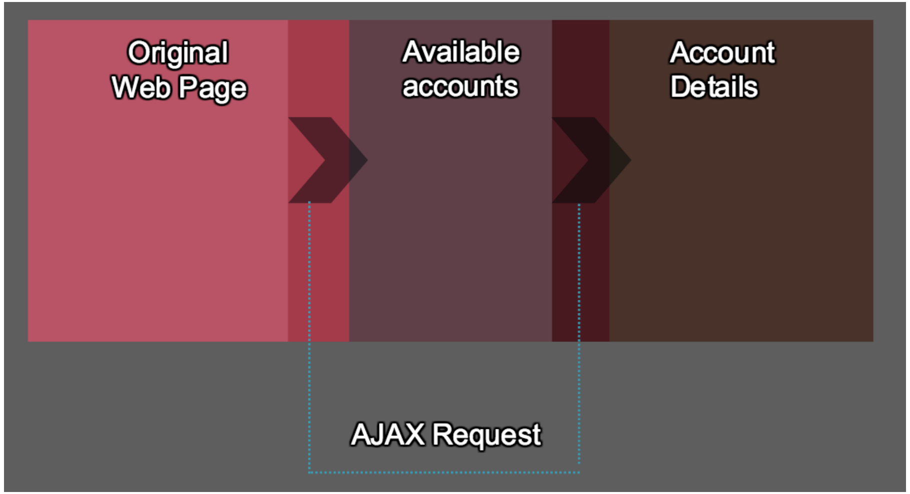
SECTION 9:
RELAY AND GRAPHQL

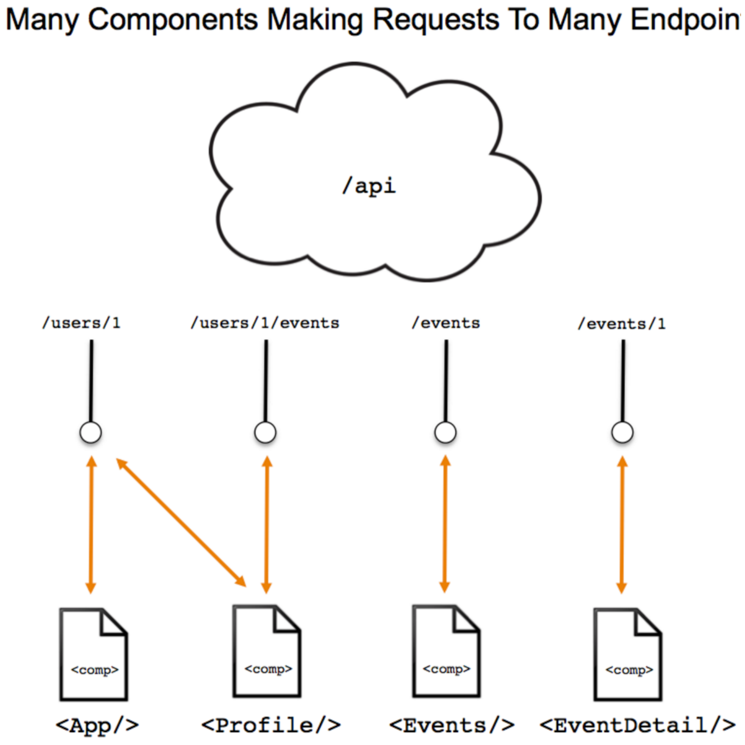
A LITTLE BIT OF HISTORY



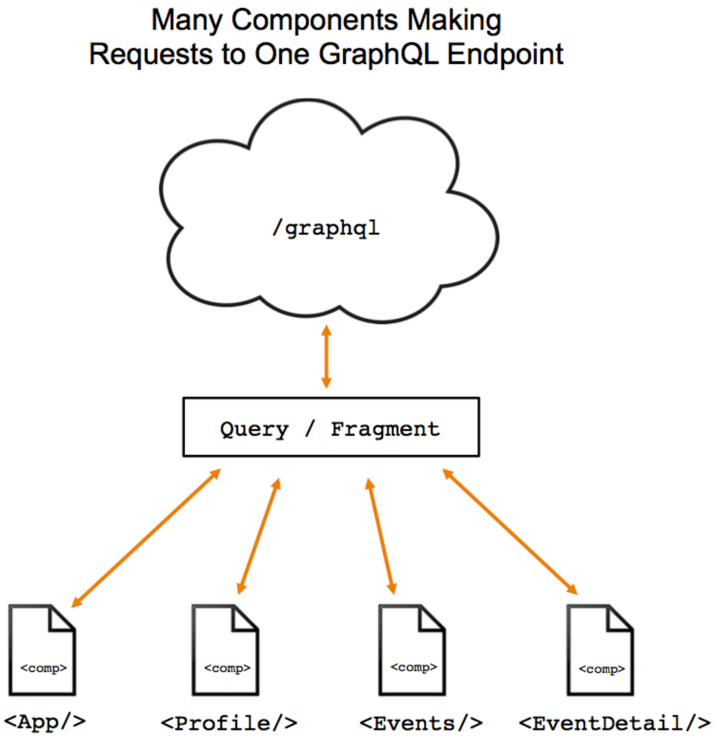


FLUX + DATA = RELAY

- Relay provides easy flexible way to fetch data for component



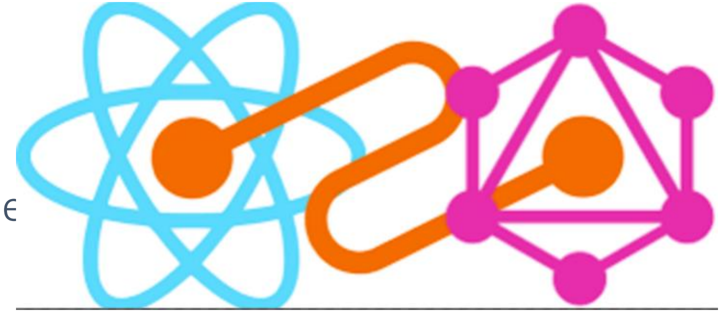
Many Components Making Requests To Many Endpoin



Many Components Making Requests to One GraphQL Endpoint

RELAY

- Relay is a JavaScript application framework that can talk to a GraphQL server.
- Declarative
- Lets define data needs for each of their React compone with GraphQL.
- Mutates data based on actions
- Provides optimistic updates (simulates mutation in the client, so the user doesn't need to wait for the server result to see the change they made)
- Provides data caching to reduce server round trips



GRAPHQL

- Get user:

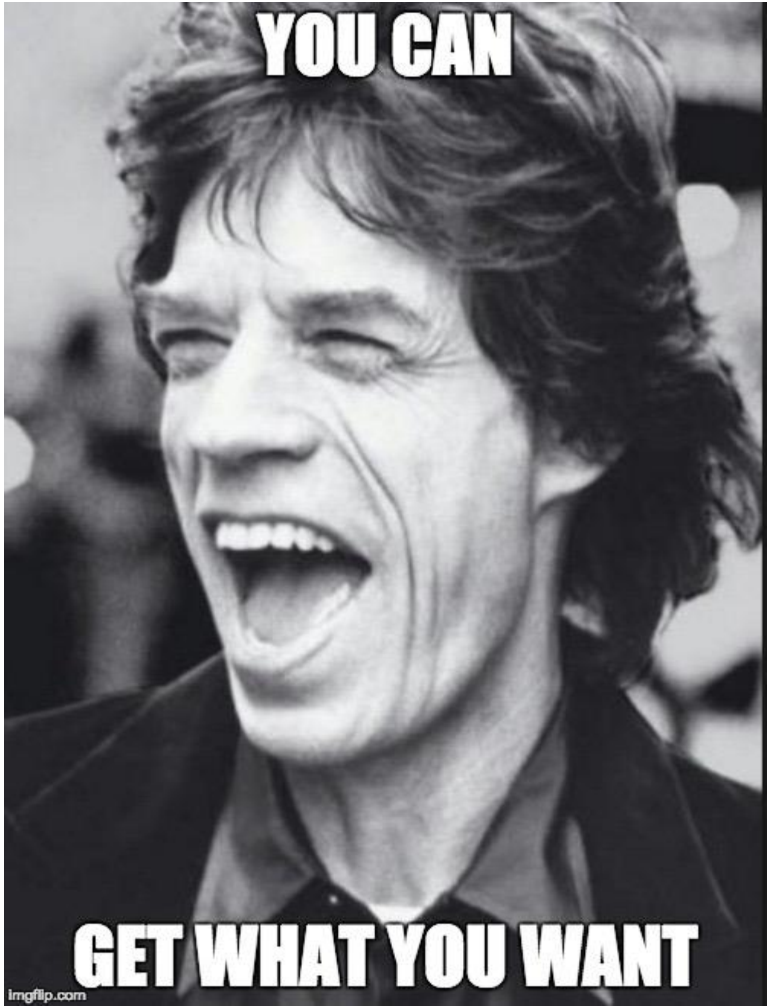
```
{
    name,
    address,
    profilePicture:{
        uri
    }
 },
```

- Get only user name:

```
{
    name
 },
```

- Get only user name:

```
{
    address
 },
```



GRAPHQL

- GraphQL is an application layer query language, which queries and mutates (or updates) data
- Solves problems:
  - Client-side developers should have a way to ask for the data they need.
  - Server-side developers should have a way to expose the data they have.
  - There should be an easy and efficient way to query data (the REST API consumes a lot of resources, especially with mobile apps).
- GraphQL sits between the client apps and the actual data sources
- You can create a GraphQL server in Node, PHP or any other platform you like. Mobile apps, web apps and anyone else should be able to connect to a GraphQL server. Then they can query and mutate data.
- But GraphQL does not comes with a transport layer. That’s the responsibility of a high-level framework, like Relay.

GRAPHQL

Query and Response

```
{
   product(id: "react-native-for-android") {
      name,
      link,
      votes,
      comments {
         text
      }
   }
}
```

```
{
   "data": {
      "product": {
         {
            "name": "React Native for Android",
            "link": "https://facebook.github.io/react-native/",
            "votes": "167"
            "comments": [
               {
                  "Huuuuge for cross-platform progress.",
               },
               {
                  "Exciting stuff.",
               }
            ]
         }
      }
   }
}
```

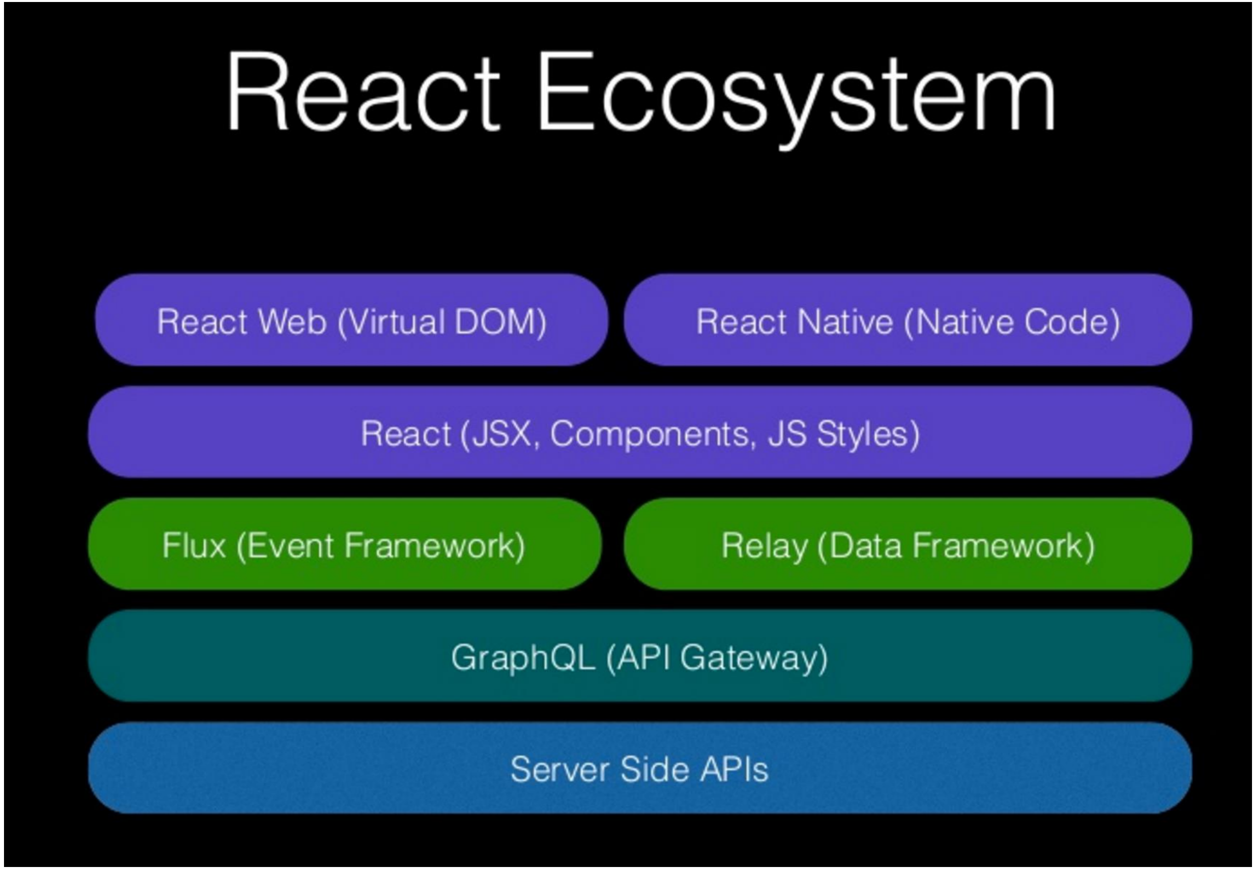
WHAT IS THE CATCH?

- No catches =)

HOW IT LOOKS?

```
class User extends React.Component {
  render() {
    var {name, address} = this.props.tea;
    return (
      <div>
        <span>{name}</span>
        <span>{address}</span>
      </div>
    );
  }
}
User = Relay.createContainer(User, {
  fragments: {
    user: () => Relay.QL`
   fragment on User {
    name,
    address,
   }
  `,
  },
})
```

ECOSYSTEM



React Ecosystem

React Web (Virtual DOM)

React Native (Native Code)

React (JSX, Components, JS Styles)

Flux (Event Framework)

Relay (Data Framework)

GraphQL (API Gateway)

Server Side APIs