↑

flow^up

Unlocking the power of GraphQL in Angular

Matěj Chalk

Front-end developer

@ flow^up

/ matejchalk

/ matejchalk

Structure of presentation

Question 1

What is GraphQL?

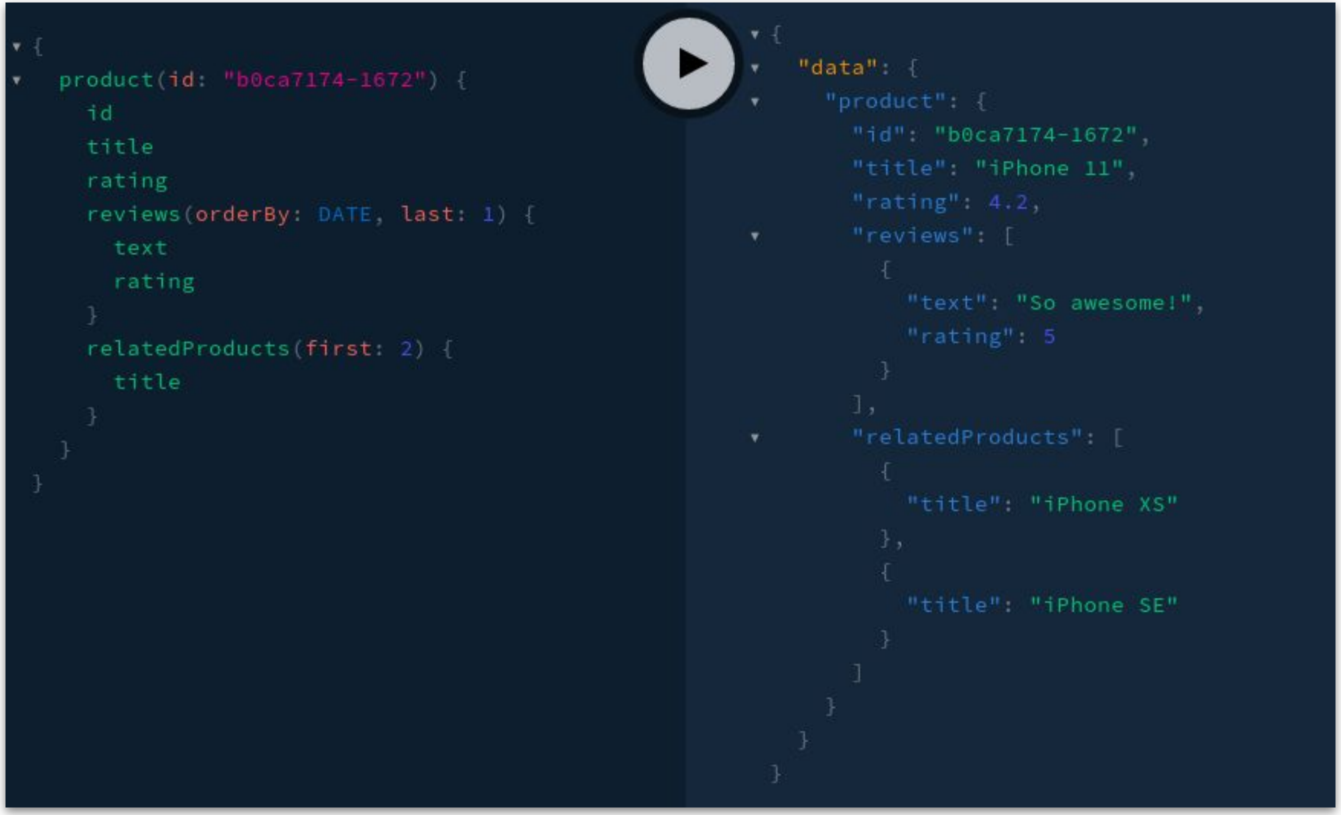
- **a query language for APIs**
- open-sourced by Facebook in 2015
- rapidly growing ecosystem
- widely adopted in production

Core principles

- single endpoint exposes all data as a graph
- **strongly-typed schema**
- clients **query exactly what they need** (nothing more, nothing less)
- **multiple resources combined in single request**

Query example

flowup

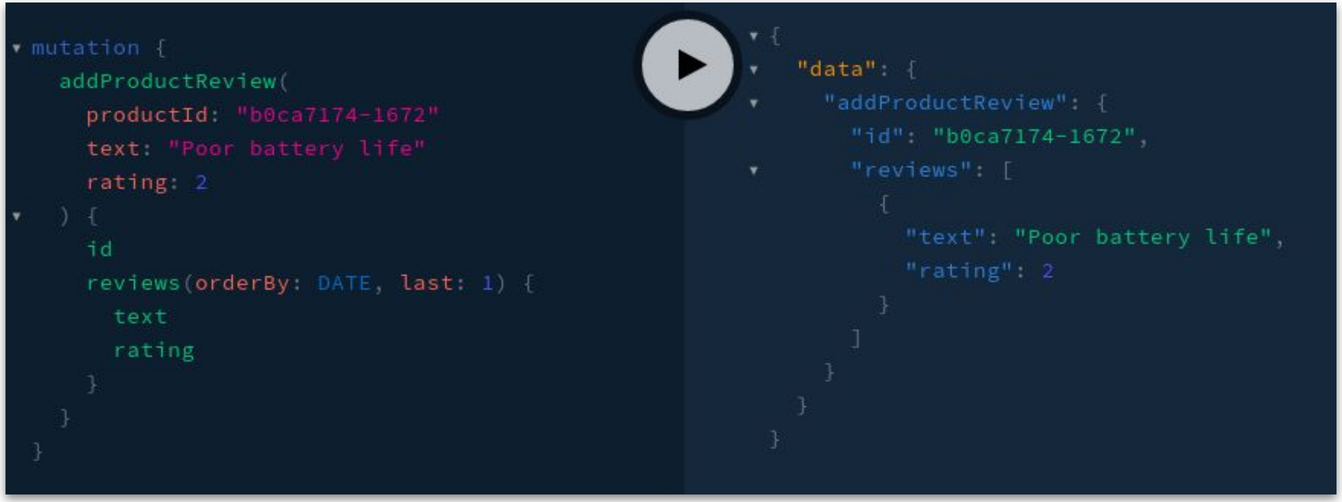


```
{
  product(id: "b0ca7174-1672") {
    id
    title
    rating
    reviews(orderBy: DATE, last: 1) {
      text
      rating
    }
    relatedProducts(first: 2) {
      title
    }
  }
}
```

```
{
  "data": {
    "product": {
      "id": "b0ca7174-1672",
      "title": "iPhone 11",
      "rating": 4.2,
      "reviews": [
        {
          "text": "So awesome!",
          "rating": 5
        }
      ],
      "relatedProducts": [
        {
          "title": "iPhone XS"
        },
        {
          "title": "iPhone SE"
        }
      ]
    }
  }
}
```

Mutation example

flowup



```
mutation {
  addProductReview(
    productId: "b0ca7174-1672"
    text: "Poor battery life"
    rating: 2
  ) {
    id
    reviews(orderBy: DATE, last: 1) {
      text
      rating
    }
  }
}
```

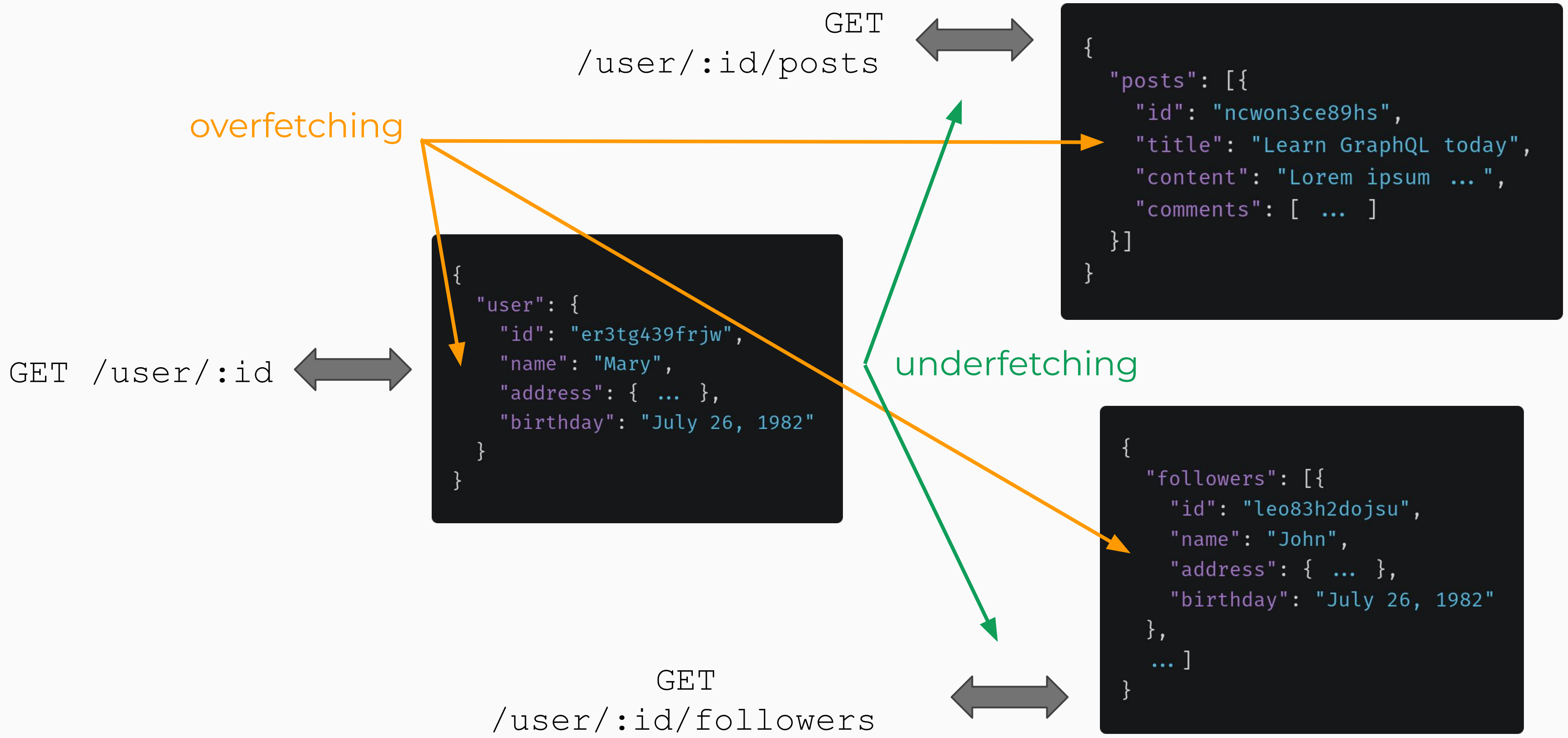
```
{
  "data": {
    "addProductReview": {
      "id": "b0ca7174-1672",
      "reviews": [
        {
          "text": "Poor battery life",
          "rating": 2
        }
      ]
    }
  }
}
```

Question 2

Advantages over REST

- GraphQL schema as **unfailing contract specifying API capabilities**
  - self-documenting and self-validating
- **no more underfetching or overfetching**
  - client dictates shape of response object
- rapid product development
  - **decoupled frontend and backend** (schema-driven development)
  - **client libraries** with built-in caching, optimistic updates, ...
- **powerful developer tooling**
  - based on GraphQL’s introspection capabilities
  - GraphQL Playground, compile-time checks, IDE autocomplete, API mocking, ...

GQL vs REST - REST approach



```
{
  "posts": [{
    "id": "ncwon3ce89hs",
    "title": "Learn GraphQL today",
    "content": "Lorem ipsum ... ",
    "comments": [ ... ]
  }]
}
```

```
{
  "user": {
    "id": "er3tg439frjw",
    "name": "Mary",
    "address": { ... },
    "birthday": "July 26, 1982"
  }
}
```

```
{
  "followers": [{
    "id": "leo83h2dojsu",
    "name": "John",
    "address": { ... },
    "birthday": "July 26, 1982"
  },
  ... ]
}
```

GQL vs REST - GraphQL approach

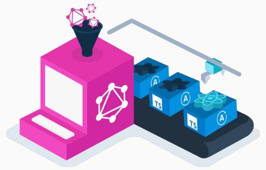
```
POST /graphql
{
  user(id: "...") {
    name
    posts {
      title
    }
    followers(last: 3) {
      name
    }
  }
}
```

```
{
  "data": {
    "user": {
      "name": "Mary",
      "posts": [
        { "title": "Learn GraphQL today" }
      ],
      "followers": [
        { "name": "John" },
        { "name": "Alice" },
        { "name": "Sarah" }
      ]
    }
  }
}
```

Question 3

GraphQL ecosystem

- frontend
  - **Apollo Client**
  - Relay
- backend
  - GraphQL.js, Express GraphQL
  - Apollo Server
  - AWS Amplify, AWS AppSync
- other
  - **GraphQL Code Generator**
  - **GraphQL Playground**



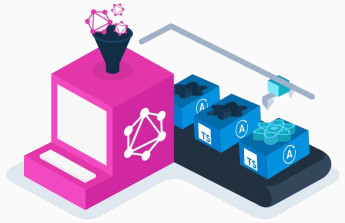
Apollo Client

- customizable **open-source GraphQL client** for JavaScript
- powerful **caching** and **state management** features
- functions as a sophisticated **data layer**
  - abstracts away the network from the developer
- integrates well with TypeScript and Angular

```
$ ng add apollo-angular
```

GraphQL Code Generator

- CLI tool
- **generates code from GraphQL schema and operations**
- customizable via configuration file and plugins
  - e.g. for TypeScript, Apollo Angular



```
$ graphql-codegen init
```

Demo

github.com/matejchalk/apollo-angular-example

Q & A

Matěj Chalk

Front-end developer

@ flowup

/ matejchalk

/ matejchalk