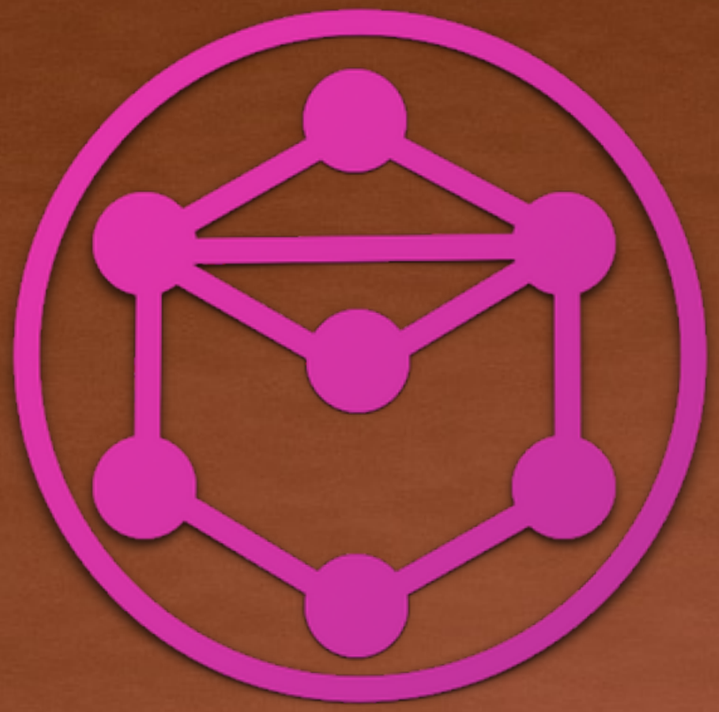

INTRODUCTION TO GRAPHQL

JASDEEP SINGH

AGENDA

- HISTORY
- BACKGROUND
- WHAT IS GRAPHQL?
- GRAPHQL & REST
- GRAPHQL OPERATIONS
- TERMINOLOGY
- QUESTIONS

HISTORY

- Conceived & Developed at Facebook in 2012.
- During the era when Facebook was transitioning from HTML5 based mobile apps to Native apps.
- Heavily used at Facebook in their Native mobile apps.
- Today, it powers hundreds of billions of API calls a day at Facebook and other companies too.

BACKGROUND

- GraphQL is a declarative query language.
- Using GraphQL, a developer can write/define function calls (called Queries) to fetch data rather than actual queries to a database or remote endpoints.
- A GraphQL server interprets these function calls and is able to resolve with the requested data.

WHAT IS GRAPHQL?

A GRAPHQL QUERY IS A STRING THAT IS SENT TO A SERVER TO BE INTERPRETED AND FULFILLED, WHICH THEN RETURNS JSON BACK TO THE CLIENT.

```
{
  user(id: 4802170) {
    id
    name
    isViewerFriend
    profilePicture(size: 50)  {
      uri
      width
      height
    }
    friendConnection(first: 5) {
      totalCount
      friends {
        id
        name
      }
    }
  }
}
```

```
{
  "data": {
    "user": {
      "id": "4802170",
      "name": "Lee Byron",
      "isViewerFriend": true,
      "profilePicture": {
        "uri": "cdn://pic/4802170/50",
        "width": 50,
        "height": 50
      },
      "friendConnection": {
        "totalCount": 13,
        "friends": [
          {
            "id": "305249",
            "name": "Stephen Schwink"
          },
          {
            "id": "3108935",
            "name": "Nathaniel Roman"
```

GRAPHQL & REST

LIMITATIONS OF REST

- Many Endpoints
- Bloated Responses
- Overfetching or Underfetching of data
- Multiple roundtrips
- Versioning can be a pain sometimes
- Not Introspective

HOW GRAPHQL RESOLVES REST LIMITATIONS

- Single endpoint: A shared endpoint can resolve GraphQL queries into root calls and send back a single, unified response.
- Tailored responses: Client-driven queries mean that the response is catered to the demands of the client rather than the limitations of the API.
- Fewer round trips
- Backwards compatibility: GraphQL endpoints can be developed independently of the versioning of endpoints, as the requests and responses are very dynamic by nature.
- Introspective: GraphQL has a native and highly extensible schema and type system.

GRAPHQL OPERATIONS

```
query {
  # Read-only fetch
}
mutation {
  # Write then fetch
}
```

DEMO & GRAPHQL TERMINOLOGY

- Queries
- Fields
- Aliases
- Fragments
- Mutations [We will not cover mutations]

QUESTIONS?

THANK YOU!