APOLLO

{ REST }

REBBIS

GraphQL
REST

Andranik Barseghyan

https://github.com/Rebiss/Presentation

CONTENTS

GraphQL

REST

Conclusion

REST

Representational state transfer (REST) is a software architectural style that defines a set of constraints to be used for creating Web services. Web services that conform to the REST architectural style, called RESTful Web services, provide interoperability between computer systems on the internet.

REST

Roy Fielding

Difficulty Distribution

URIs.

HTTP Header.

JSON Body.

Well suited for CRUD.

RESTful Web Service

Contract First.

Code First

HATEOAS.

REST problems

Multiple Endpoints

REST APIs are usually a collection of endpoints, where each endpoint represents a resource. So when a client needs data from multiple resources, it needs to perform multiple round-trips to a REST API to put together the data it needs.

Not Client-Friendly

In a REST API, there is no client request language. Clients do not have control over what data the server will return. There is no language through which they can do so. More accurately, the language available for clients is very limited.

Network Delays

Long Response Times.
TCP 3-Way Handshake.
HTTP Handshake.
Over-fetching
end
Under-fetching.

Multiple Versions

Big problem with REST APIs is versioning. If you need to support multiple versions, that usually means new endpoints. This leads to more problems while using and maintaining those endpoints and it might be the cause of code duplication on the server.

GraphQL

GraphQL is a query language for your API, and a server-side runtime for executing queries by using a type system you define for your data.

GraphQL problems

N + 1 Reqest

Namespace

Growing Query Complexity

Caching

GraphQL Core Concept

Queries and Mutations

Introspection Queries.

GET request, POST request, Response.

Schemas and Types

The GraphQL schema provides a clear contract for client-server communication.

chema Definition Language (SDL)

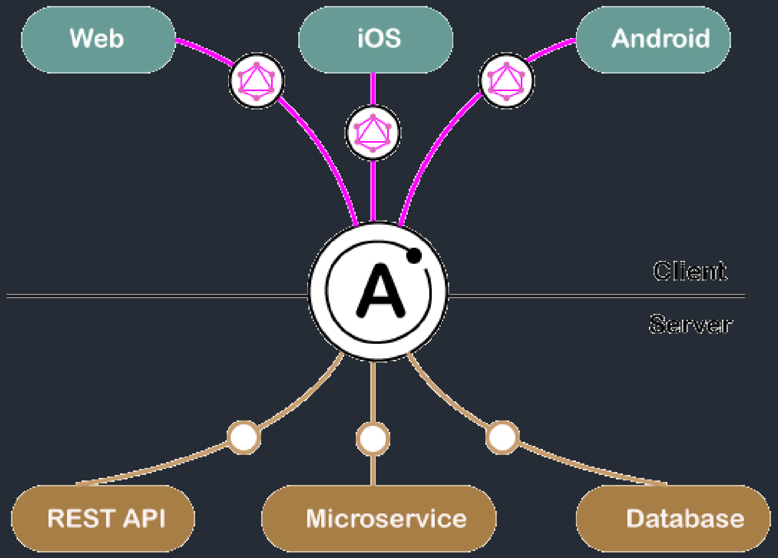
Subscriptions

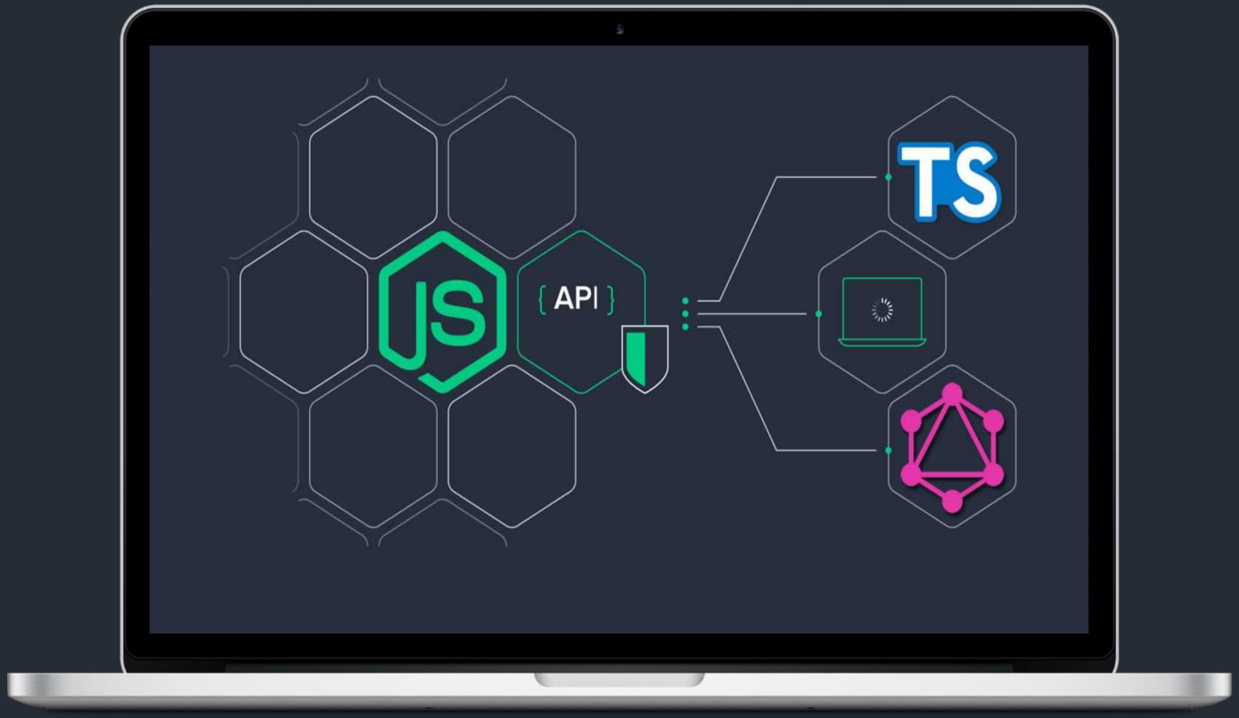
The GraphQL specification allows for something called subscriptions that are like GraphQL queries but instead of returning data in one read, you get data pushed from the server. Without having to deal with websocket code.

Resolvers

In its most basic form, a GraphQL server will have one resolver function per field in its schema. Each resolver knows how to fetch the data for its field. Since a GraphQL query at its essence is just a collection of fields, all a GraphQL server actually needs to do in order to gather the requested data is invoke all the resolver functions for the fields specified in the query.

Example





https://github.com/Rebiss/gq-rc

Conclusion

Representational state transfer

2015

GraphQ

GraphQL is a data query language and specification developed internally by Facebook in 2012 before being publicly open-sourced in 2015.

gRPC

gRPC (Remote Procedure Calls) is an open-source remote procedure call (RPC) system initially developed at Google in 2015.

2015

2019

GraphQL Foundation

The GraphQL Foundation is a neutral foundation founded by global technology and application development companies.

Why Not ?

2021

2020

THANK YOU