Queryable APIs with GraphQL

Phillip Kruger

March 2021

Agenda

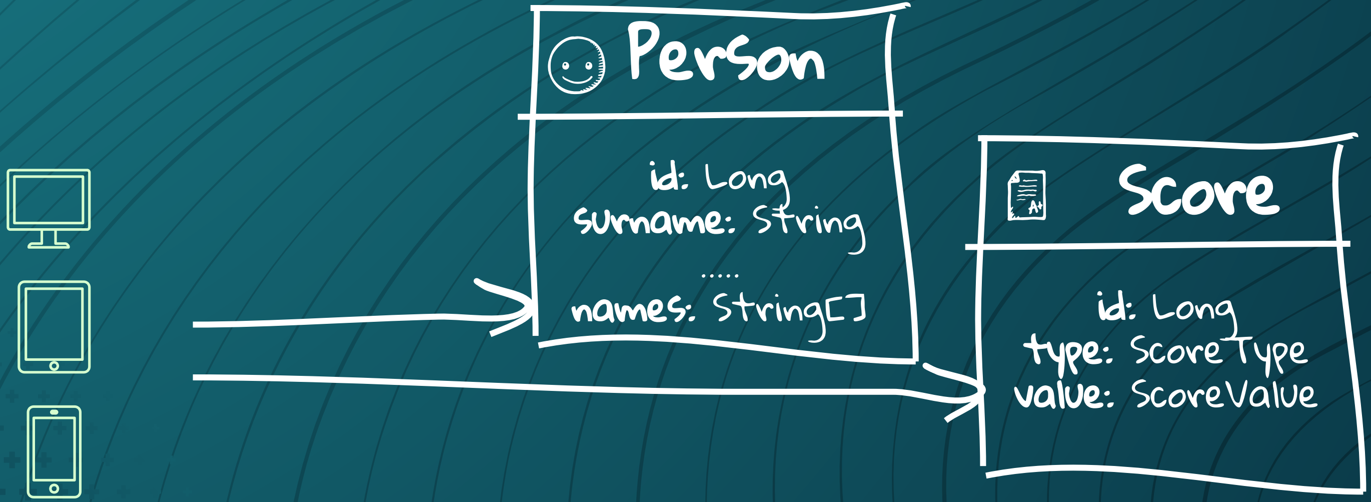
- The Use case - explain our example
- REST - build it in JAX-RS
- The Problem - over and under fetch
- GraphQL - convert this to GraphQL
- More GraphQL - what else can we do with GraphQL
- Next - what is in the pipeline

1. The use case

Let’s start with explaining our example

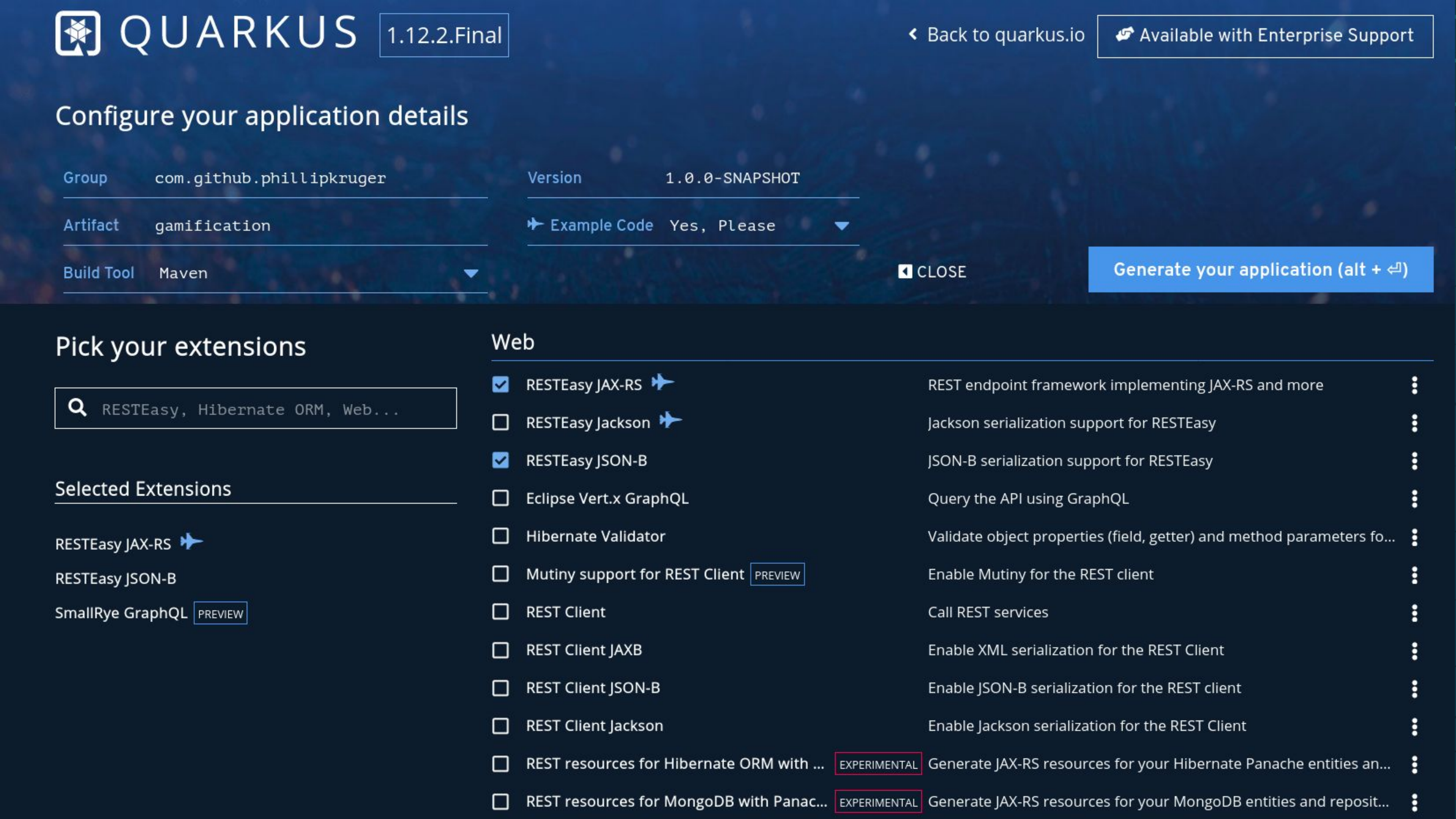
Gamification

- Score and reward users for certain actions



2. REST

Let’s build a JAX-RS Application



QUARKUS 1.12.2.Final

Back to quarkus.io | Available with Enterprise Support

Configure your application details

Group	com.github.phillipkruger	Version	1.0.0-SNAPSHOT
Artifact	gamification	Example Code	Yes, Please
Build Tool	Maven		

CLOSE | Generate your application (alt + ⏎)

Pick your extensions

RESTEasy, Hibernate ORM, Web...

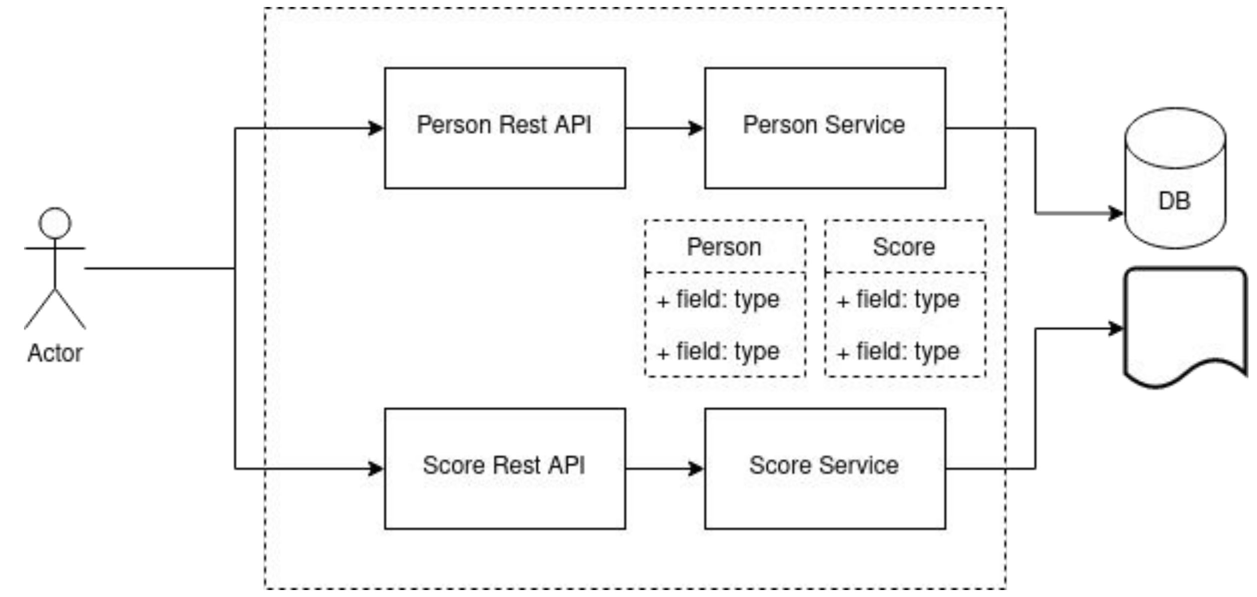
Selected Extensions

- RESTEasy JAX-RS
- RESTEasy JSON-B
- SmallRye GraphQL PREVIEW

Web

✓	RESTEasy JAX-RS	REST endpoint framework implementing JAX-RS and more
☐	RESTEasy Jackson	Jackson serialization support for RESTEasy
✓	RESTEasy JSON-B	JSON-B serialization support for RESTEasy
☐	Eclipse Vert.x GraphQL	Query the API using GraphQL
☐	Hibernate Validator	Validate object properties (field, getter) and method parameters fo...
☐	Mutiny support for REST Client PREVIEW	Enable Mutiny for the REST client
☐	REST Client	Call REST services
☐	REST Client JAXB	Enable XML serialization for the REST Client
☐	REST Client JSON-B	Enable JSON-B serialization for the REST client
☐	REST Client Jackson	Enable Jackson serialization for the REST Client
☐	REST resources for Hibernate ORM with ... EXPERIMENTAL	Generate JAX-RS resources for your Hibernate Panache entities an...
☐	REST resources for MongoDB with Panac... EXPERIMENTAL	Generate JAX-RS resources for your MongoDB entities and reposit...

High level design



3. The Problem

Over and under fetching

“

Over-fetching is fetching too much data, aka there is data in the response you don't use.

Under-fetching is not having enough data with a call to an endpoint, leading you to call a second endpoint.

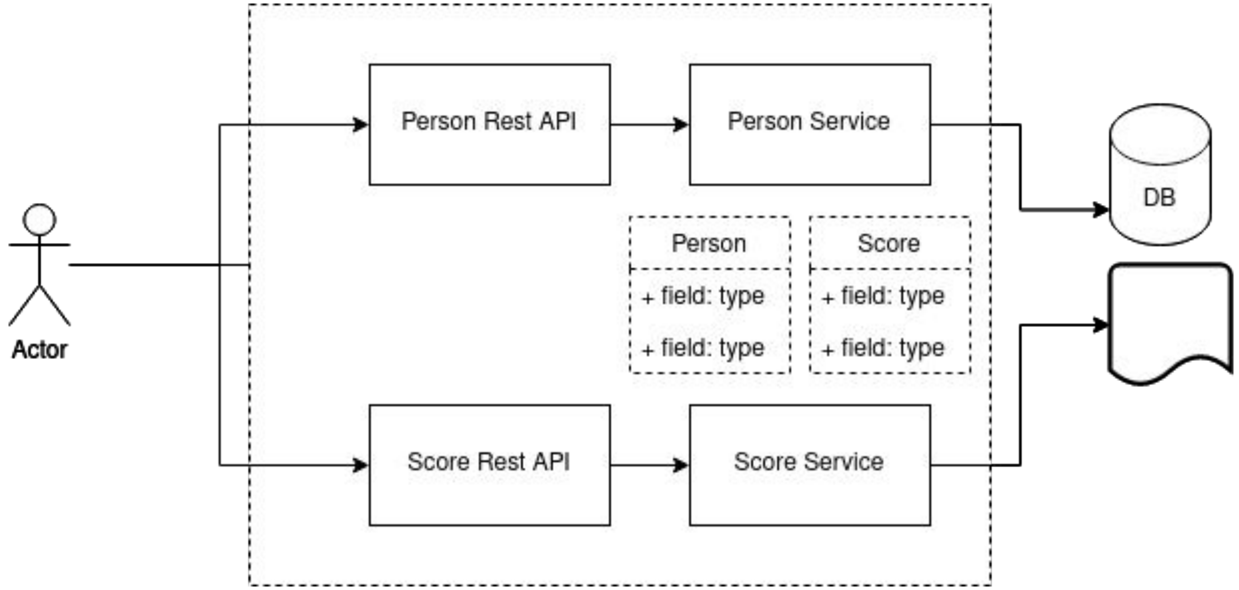
4. GraphQL

Let’s convert it to a GraphQL application

History of GraphQL

- Developed and open sourced by Facebook
- Specification
  http://facebook.github.io/graphql
- Alternative to REST
- Declarative data fetching
- Increased mobile usage
- Variety of different frontend frameworks
- Rapid feature development
- Since 2012. Publically 2015

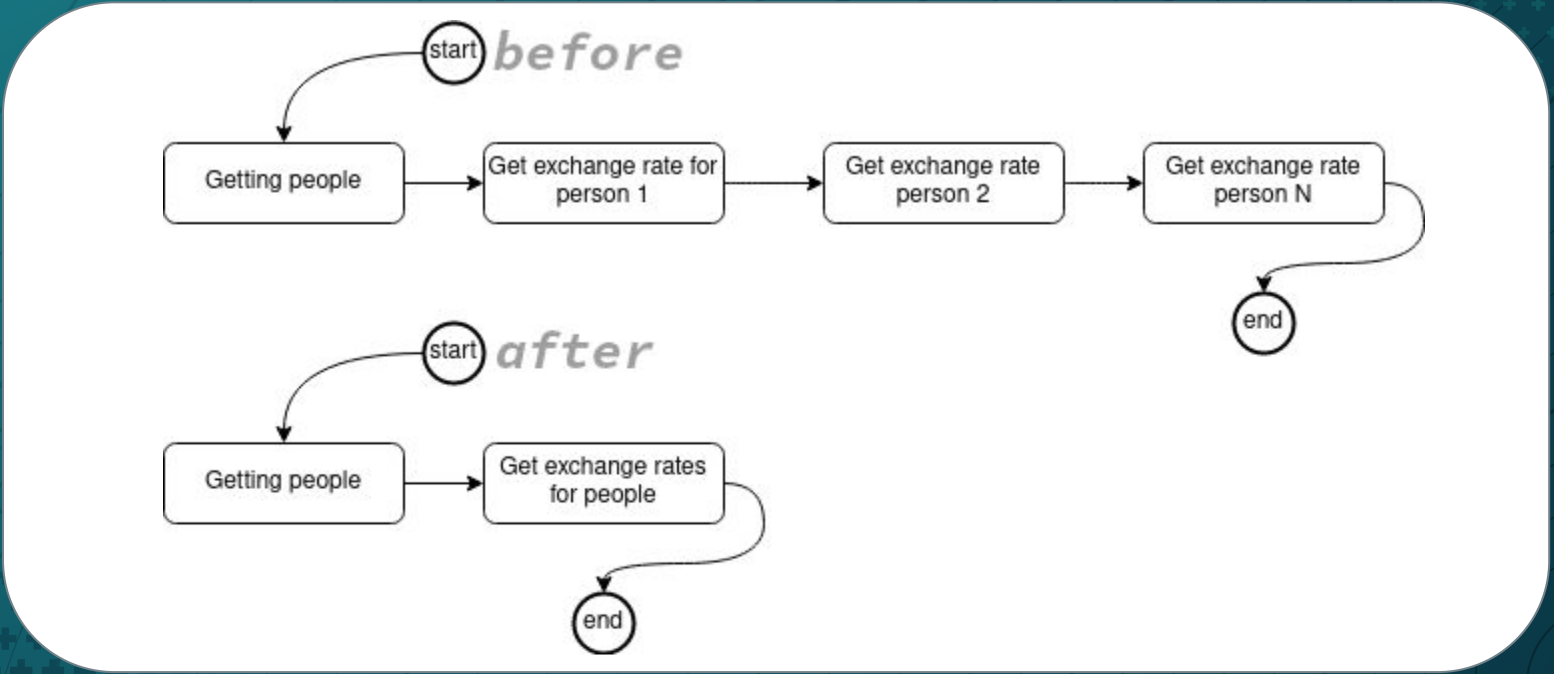
High level design



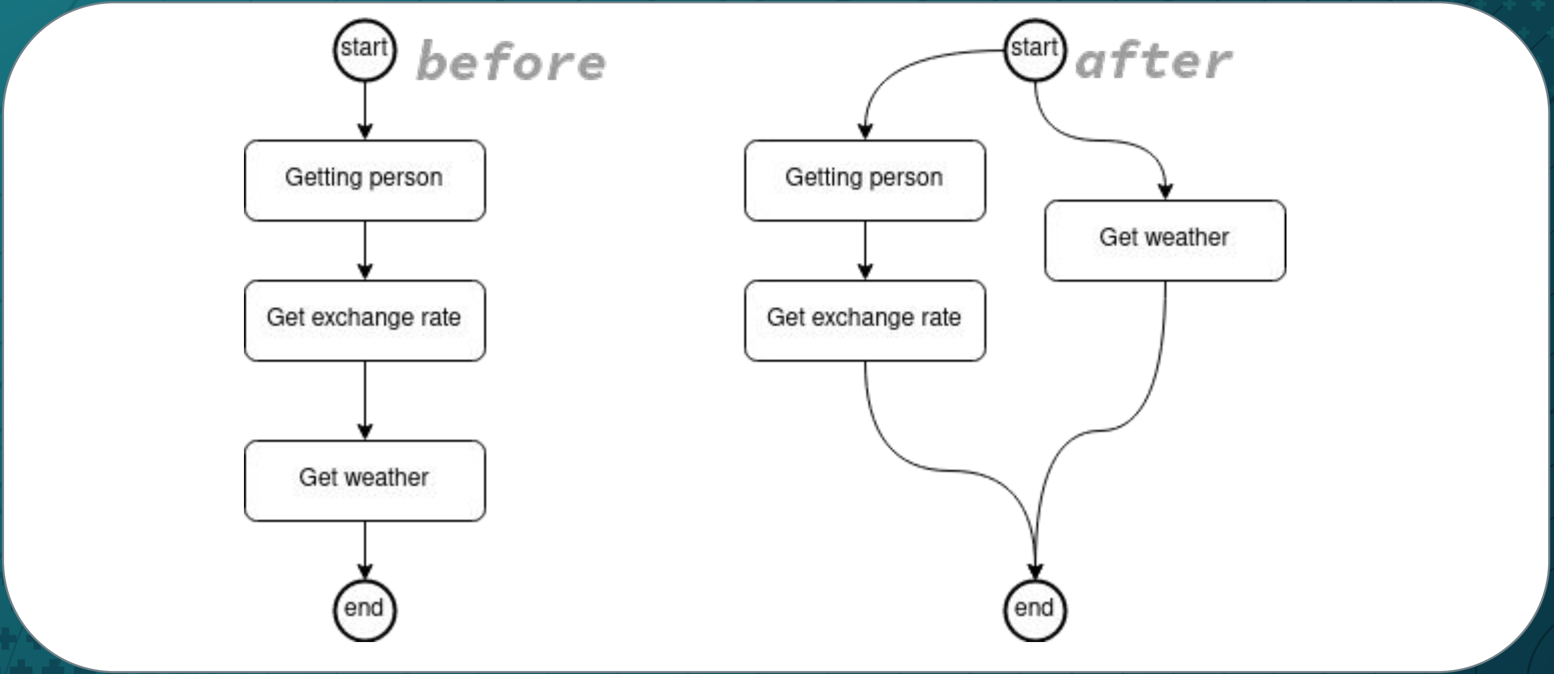
GraphQL solved over and under fetching

- Query
- Source
- Batch
- Multiple requests
- Asynchronous

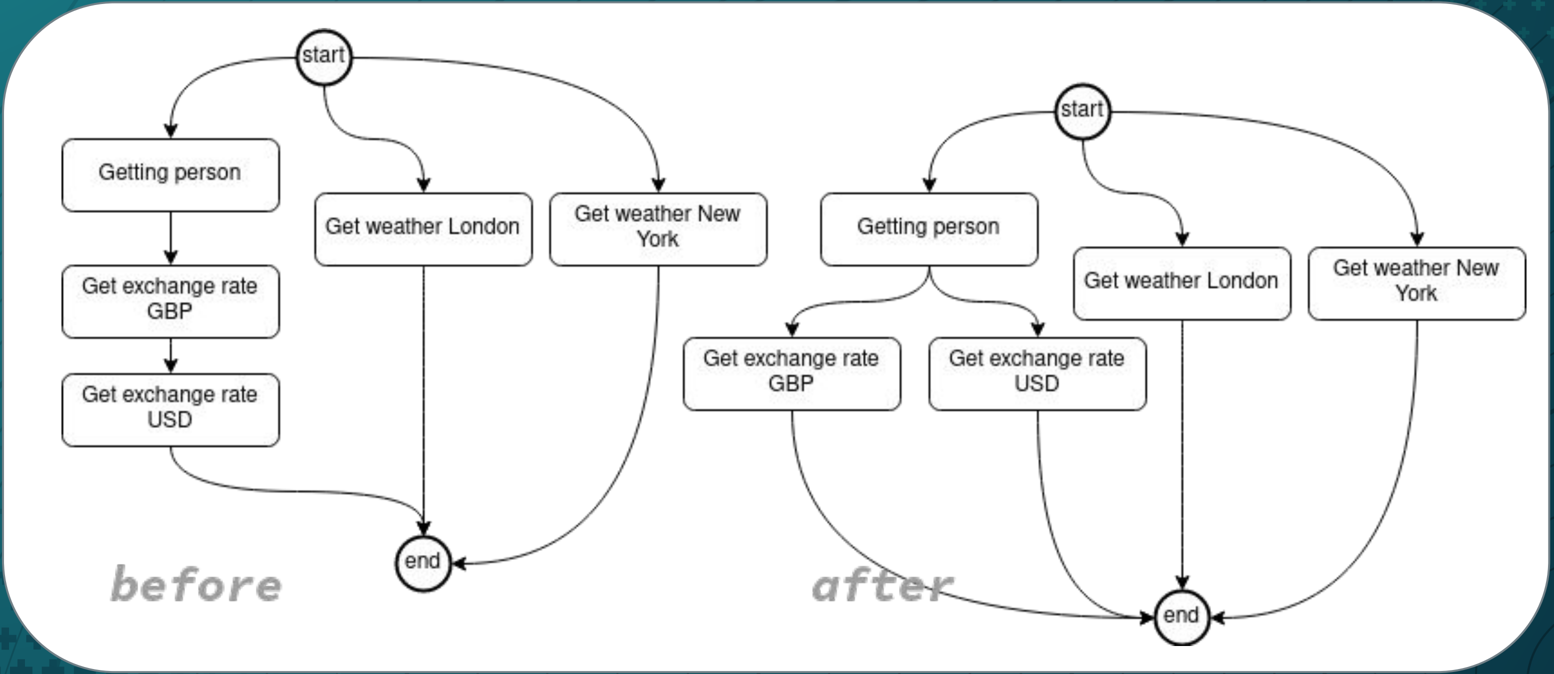
Batch


start
before
Getting people
Get exchange rate for person 1
Get exchange rate person 2
Get exchange rate person N
end
start
after
Getting people
Get exchange rates for people
end

Asynchronous



Asynchronous



5. More GraphQL

What else can we do with GraphQL

What else can we do

- Errors and partial response
- Transformation and mapping
- Mutation
- Introspection
- Security
- Operational Context
- Events
- Custom execution

Integrations

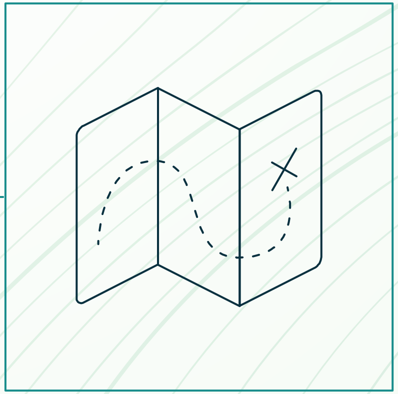
- JsonB
- Security
- Context Propagation
- Bean validation
- Metrics
- Tracing
- Generics

6. Next

What is in the pipeline

What we are working on

- Client(s)
- Subscriptions
- Paging and filtering



Thanks!

Any questions?

You can find me at

@phillipkruger

www.phillip-kruger.com

https://github.com/phillip-kruger/graphql-example
https://github.com/phillip-kruger/graphql-experimental