Readme

Architecture Analysis

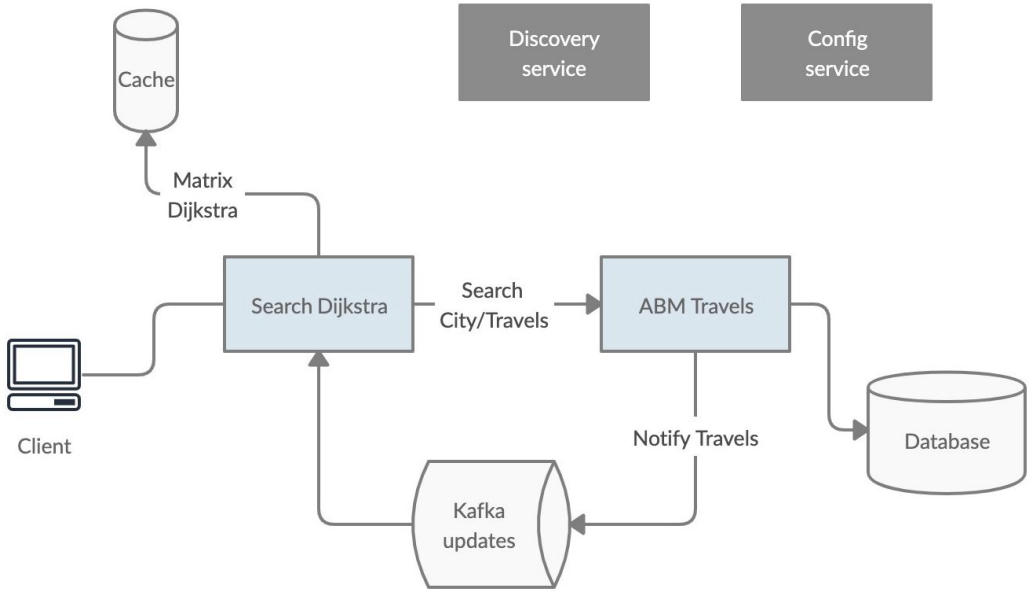

Figure Arch.1

Introduction

In order to solve this itineraries problem solution, involved in a scalable architecture i will develop all the components shown in Arch.1 using kubernetes to handle docker containers and scalability, streams with kafka to distributed and fault tolerance communication, key value store to save dijkstra structure.

//TODO: Only docker containers build, linked by docker-compose

Components:

Search: Dockerized microservice in charge of solving itinerary search through Dijkstra algorithm. Dijkstra matrix is persisted in an external cache which is updated when the consumer receives notifications from kafka.

Ideally, this will be n pods on kubernetes to scale if necessary.

//TODO: Not implemented external cache or key-value yet, structure handled on node memory (Sync issues)

ABM travels: Dockerized microservice used to persist city and travel information. When a POST/PUT/PATCH/DELETE happens over travel resource or a city is deleted, travel producer sends the appropriate notification to kafka in order to update search engine.

Ideally, this will be n pods on kubernetes to scale if necessary.

Database: Mysql docker container used to persist ABM travels information. This storage could not be a relational base due to easy scalability, but taking into account my experience and the importance of this information, a relational db is ok. Besides, mysql could be clusterized with Mysql cluster or Galera(Madiadb)

Kafka: Search microservice is subscribed to travels updates in order to change its memory structure to resolve searches. Availability service publish this new information in a travel topic to update search microservice. Kafka is built to be distributed and fault tolerant so applies to our target.

Cache: Ideally this will be persisted in a key value store or cache service, not implemented yet. To resolve search, i have in memory the matrix data to apply Dijkstra algorithm.

Config Service: Centralize configurations in microservice. NOT IMPLEMENTED