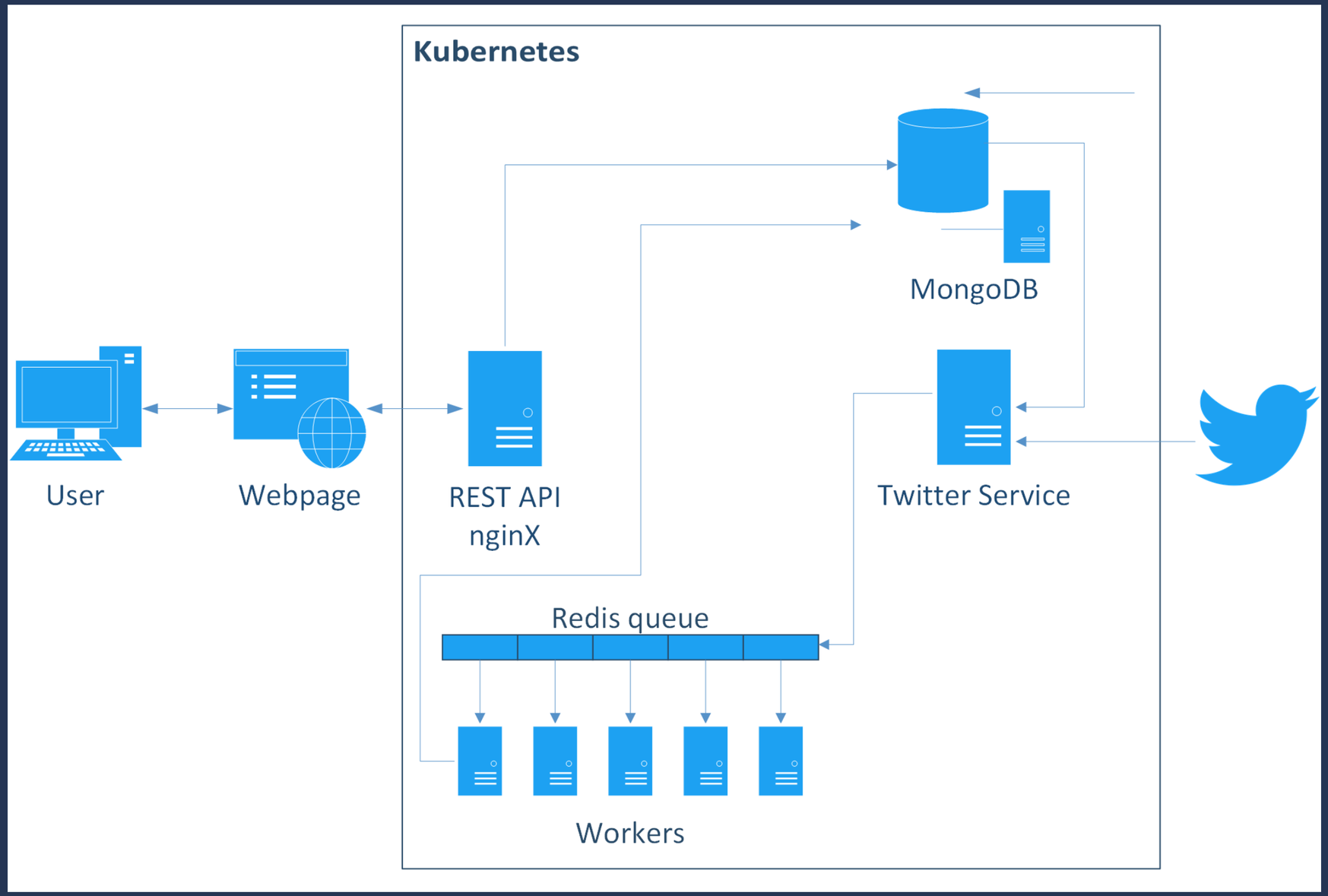
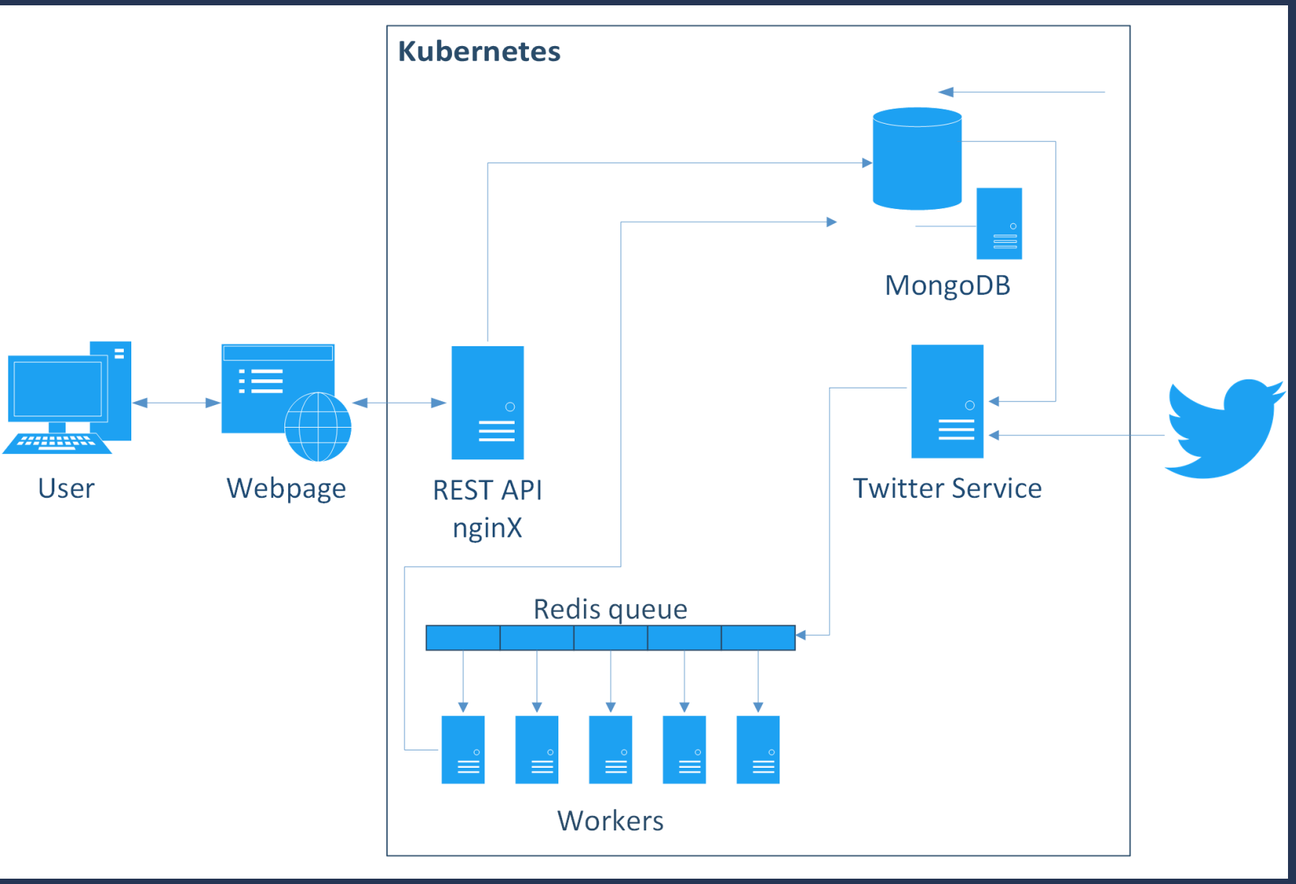
Project Presentation Stage 1

Stefanie Ziltener, Marc Heimgartner, Benjamin Bürgisser, Simon Tännler

Advanced Software Engineering FS 2017, University of Zürich

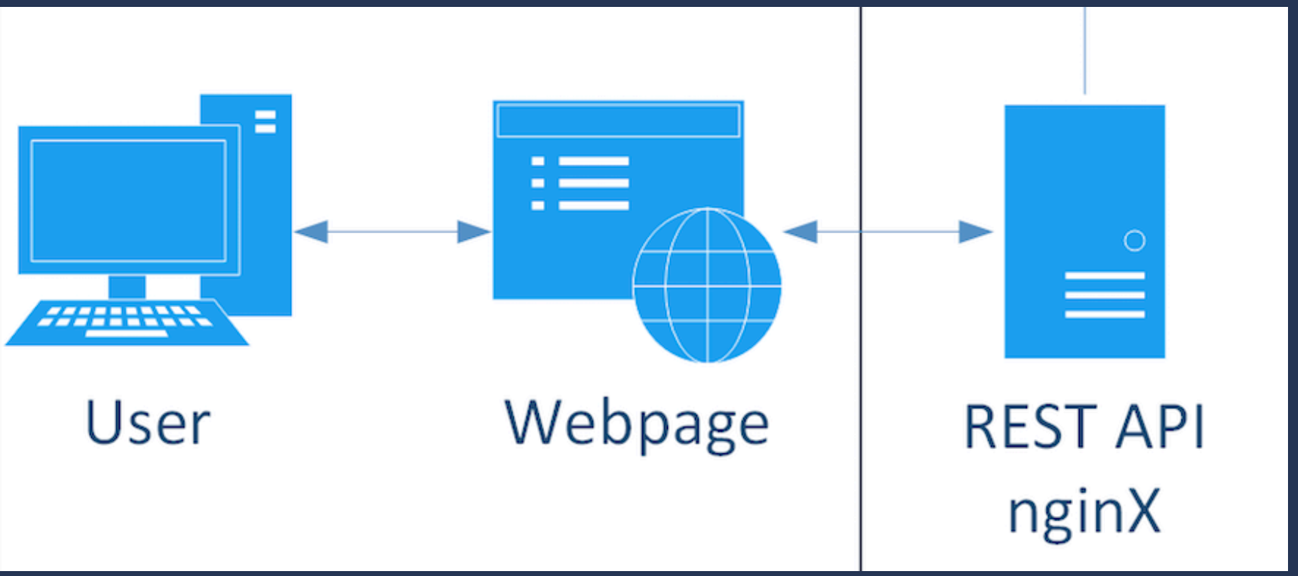

Kubernetes
MongoDB
User
Webpage
REST API
nginX
Twitter Service
Redis queue
Workers



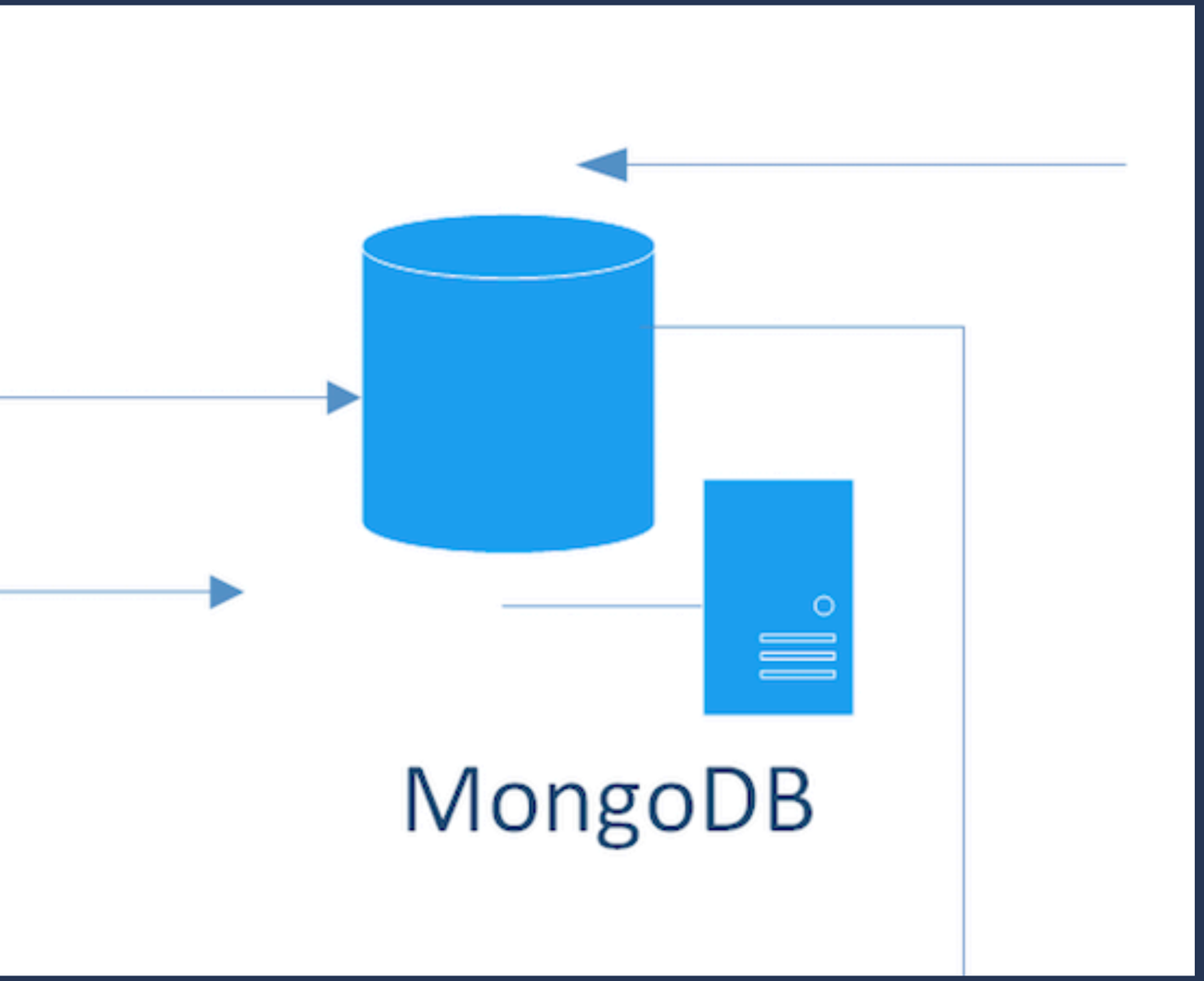
Containerized Microservices deployed through Kubernetes

- Components embedded in (Docker) Containers
- Containers have (ideally) one single responsibility

Frontend and API



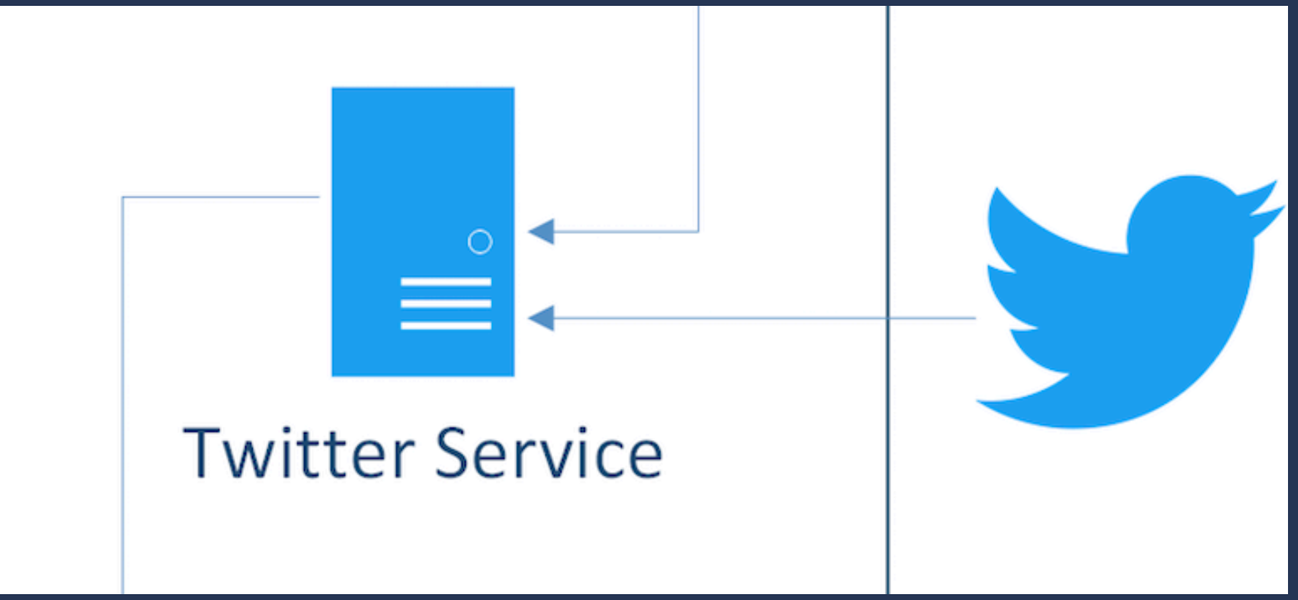
- User inputs term
- Click on register
- Send request to API



Timeseries: MongoDB (I)

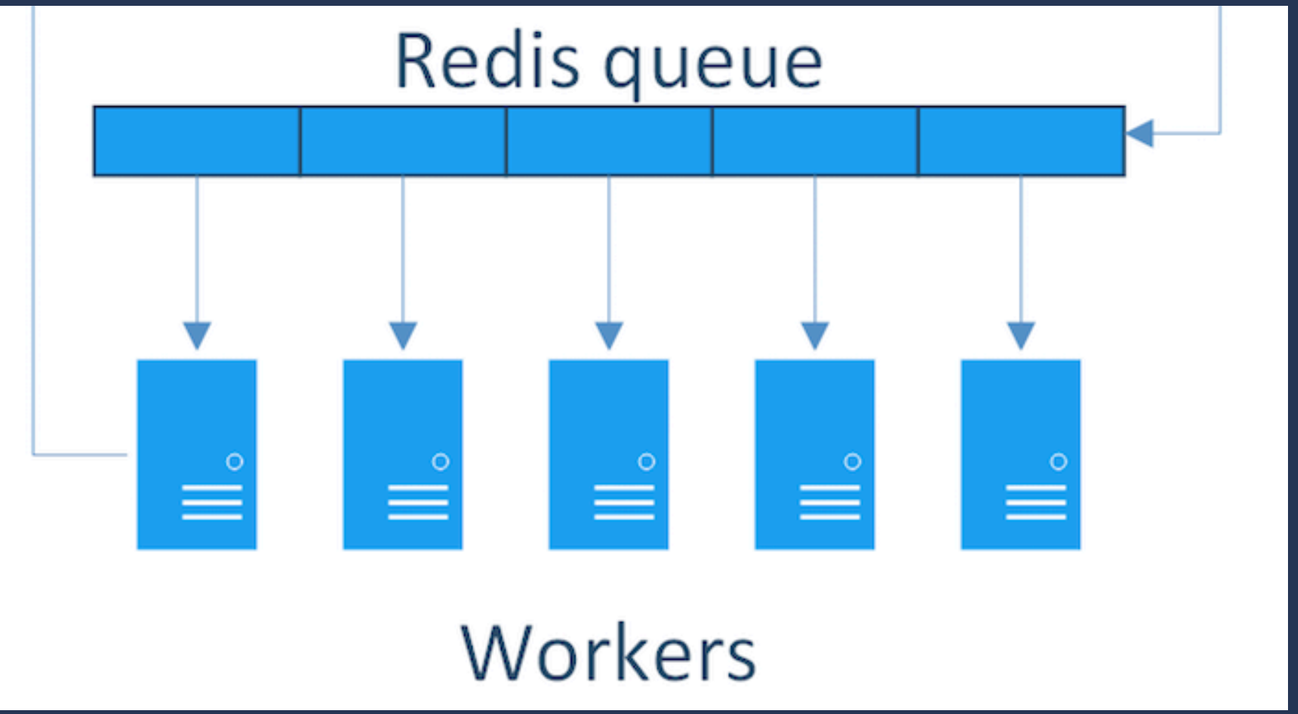
- Request Handler stores Term in MongoDB
- Persistency guaranteed by GCE Persistence Disk

Twitter Service



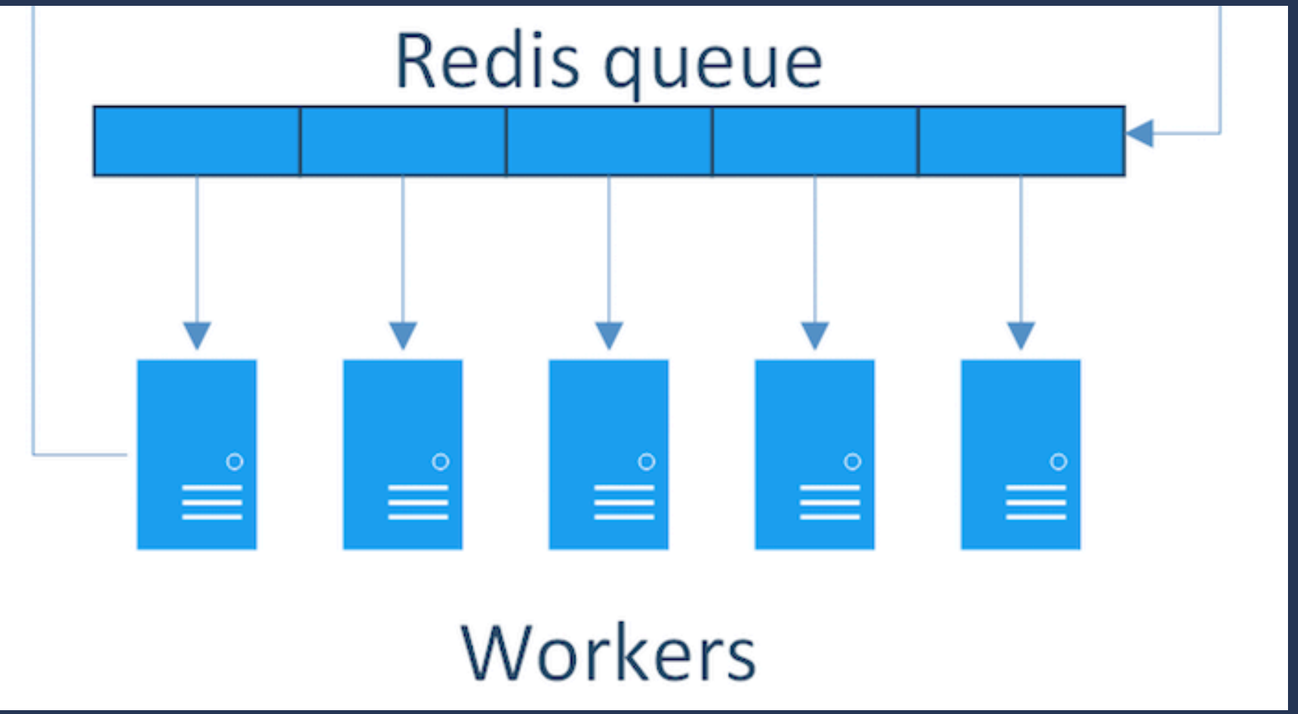
- Gets notified of newly created Terms
- Streaming stops and restarts with the new Term added for tracking
- Arriving Tweets are immediately stored into the Queue

Worker Queue

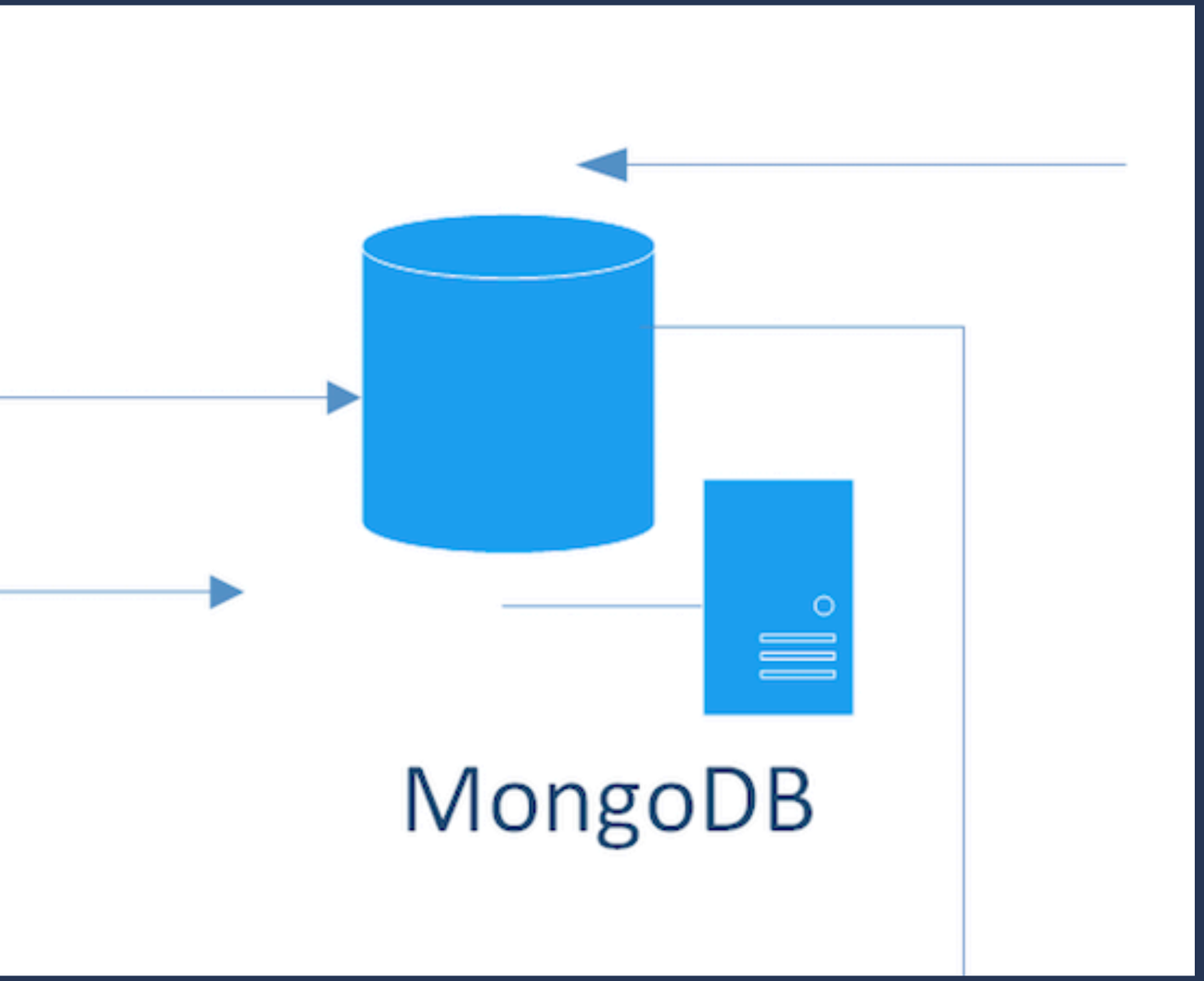


- Redis: in-memory data structure store
- A FIFO queue of Strings (Tweets)
- Load generator
  - API endpoint to add Strings to Queue directly

Compute Workers



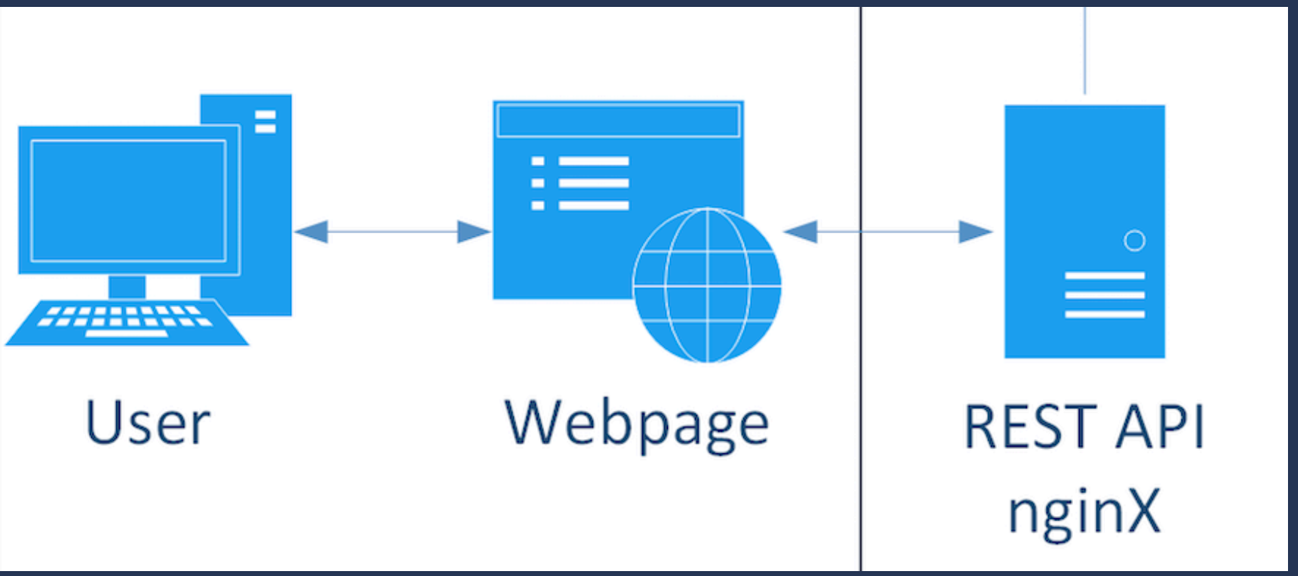
- Running Workers process the Queue:
  - Assign Tweet to Term (filtering)
  - Calculate Sentiment
- Length of the Queue defined the number of Workers
  - Scaled through Kubernetes.



Timeseries: MongoDB (II)

- Workers store the calculated Sentiment into MongoDB.

Displaying Results



- Rest API gets request
- Collect relevant data from MongoDB
- Browser renders data
- Socket gets opened for continues pushes

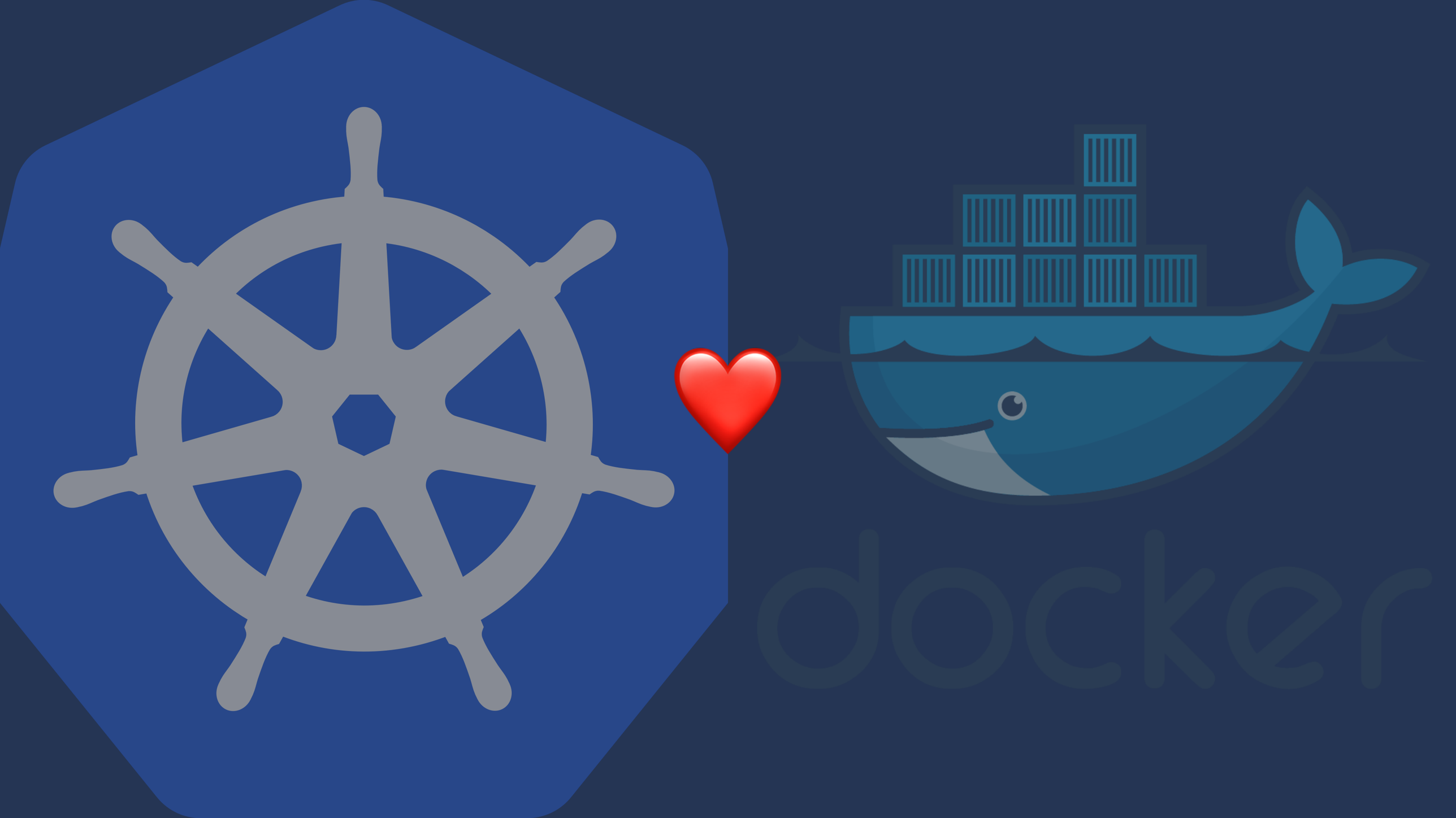
Architectural Styles

- Client / Server through Rest API
- Event-Driven notification of Term updates
- Blackboard: Redis Queue
  - Factory: Twitter Service
  - Worker: Compute Workers
- Highly decoupled Microservices

Do you even scale?

Initial Idea

- Digital Ocean
- Docker Swarm


docker

What does Kubernetes do for us

- Every component is potentially scalable through Kubernetes
  - Even MongoDB!
- Fault Tolerance:
  - Container recovery through Kubernetes
  - Decoupled design and Microservice
- Elasticity
  - Container scaling through Kubernetes

Concerns

- High lock-in to Kubernetes
- Twitter: 400 keyword tracking per stream
  - "default access level allows up to 400 track keywords"
- Matching: may not match to terms perfectly
- Redis: may become a bottleneck
  - but we highly doubt it

Technology Zoo - Platform

- Cloud Platform: Google Container Engine (GCE)
  - Easy support of Kubernetes
- Containerization: Docker
  - Popular Container engine
- Container orchestration: Kubernetes
  - Popular Container orchestration

Technology Zoo - Backend

- Programming Language: Google Go
  - New Programming language 🎉
  - Uniquely suited for Web development
  - Have I mentioned it's fast?
- Terms Storage: MongoDB
  - Easy data schema
- Queue Storage: Redis

Technology Zoo - Frontend

- Frontend: Vuejs
  - Similar to Angular and React
- Webserver: nginx
  - Battle-proven Webserver

Dev Environment

Docker containers using Docker Compose

Demo

Questions?