PADERBORN UNIVERSITY

Basilisk – Continuous Benchmarking for Triplestores

Fabian Rensing

Supervisor: Prof. Dr. Axel Ngonga
Paderborn University

June 20, 2022

Agenda

Contents

Triplestores

- Specialized databases for storing knowledge graphs

- Knowledge graphs are imported using a RDF syntax

```
@prefix dbr: <http://dbpedia.org/resource/> .
@prefix dbo: <http://dbpedia.org/ontology/> .
dbr:Germany dbo:capital dbr:Berlin .
```

- Requests are send using the query language SPARQL

Benchmarks

- Benchmarks are used to measure the performance of triplestores
- Benchmark metrics:
  - QMPH - Query Mixes per Hour
  - avgQPS - Average Queries per second
- Triplestore benchmark consists of a dataset and query file

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?capital
WHERE {
  dbr:Germany dbo:capital ?capital .
}
```

Motivation

Why are Benchmarks Needed?

- Measure and compare the performance of different triplestores
- Triplestores might handle some scenarios better than others

- Compare different versions of one triplestore

Motivation

When are Benchmarks Needed?

- Triplestores are developed in teams
- During development, benchmarks help to evaluate new implemented features
- Use Cases during the development process:
  - Performing a benchmark on a new Pull Request in GitHub
  - Performing a benchmark on a new triplestore release on GitHub or Docker Hub

Motivation

Why are Continuous/Automatic Benchmarks Needed?

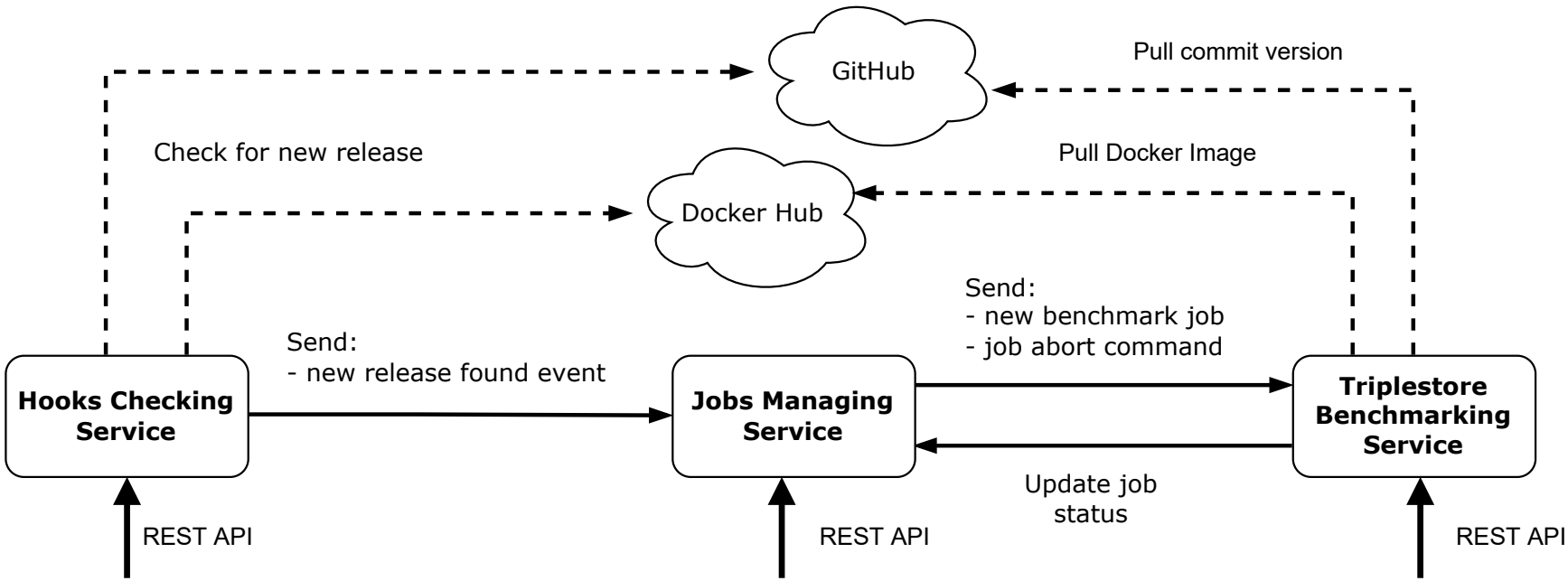
- Each benchmark requires a manual setup
  - Setting up and starting the triplestore
  - Loading the benchmark dataset
  - Configuring the benchmark framework (e.g. IGUANA)
  - Executing the Benchmark
- These steps are time-consuming, but not complicated
- Automating the benchmark process mitigates redundant manual configuration

The Basilisk Platform

Main Idea for the Platform

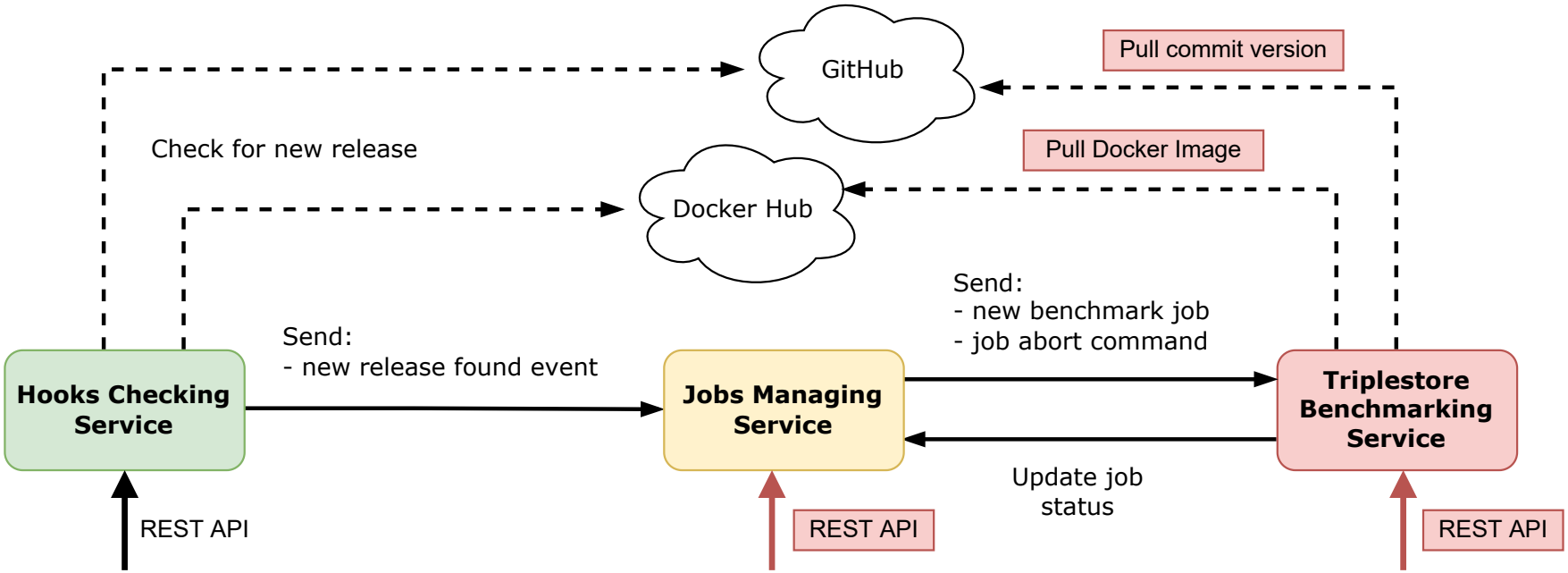
- Continuously check for new triplestore releases or Pull Request
- Automatically perform a benchmark if a new release is found
- Store the benchmark results

Platform Architecture



- User interaction via stateless REST endpoints
- Services communicate via a RabbitMQ message broker

Implementation

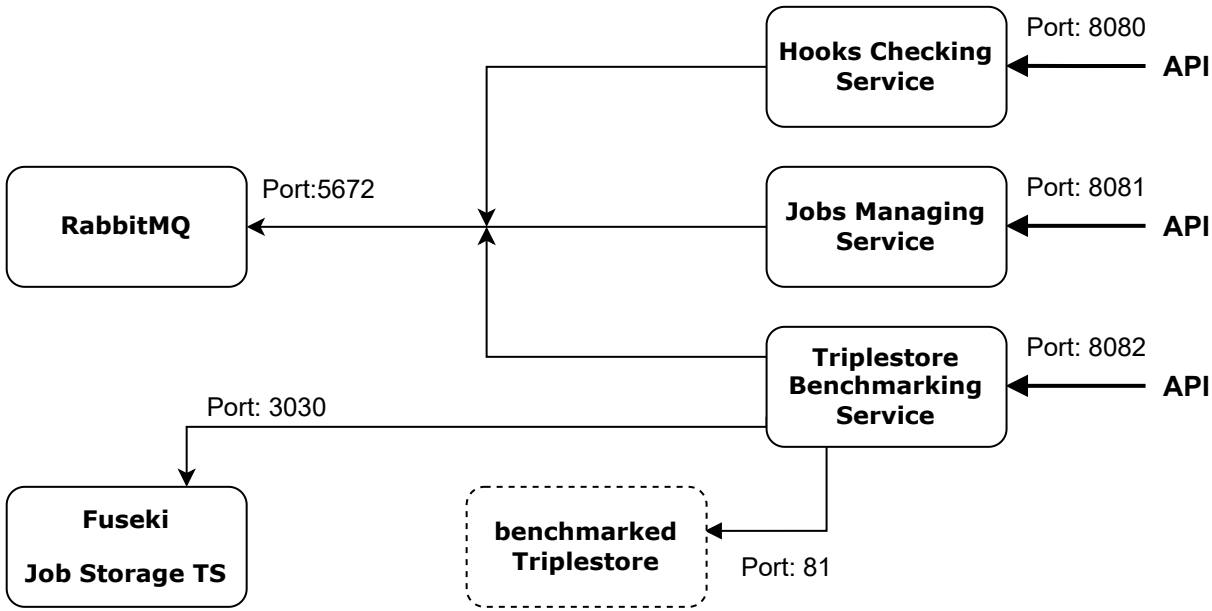


- Minor changes to data model
- Code restructure

- Adding REST endpoints
- Data model restructure
- Small fixes in functionality

- Implementing main functionality
- Docker container management
- IGUANA configuration

Deployment



- Docker Compose deployment
- 5 containers + 1 container running the benchmarked triplestore

Evaluation

- Main goal of the platform is to simplify and automize the benchmark process of triplestores
- Evaluation of the added value to the benchmark process
  - Using TENTRIS and Oxigraph triplestores
  - Pulling official images from Docker Hub
  - Comparing the manual benchmark process to the Basilisk process

Comparison

Manual Process

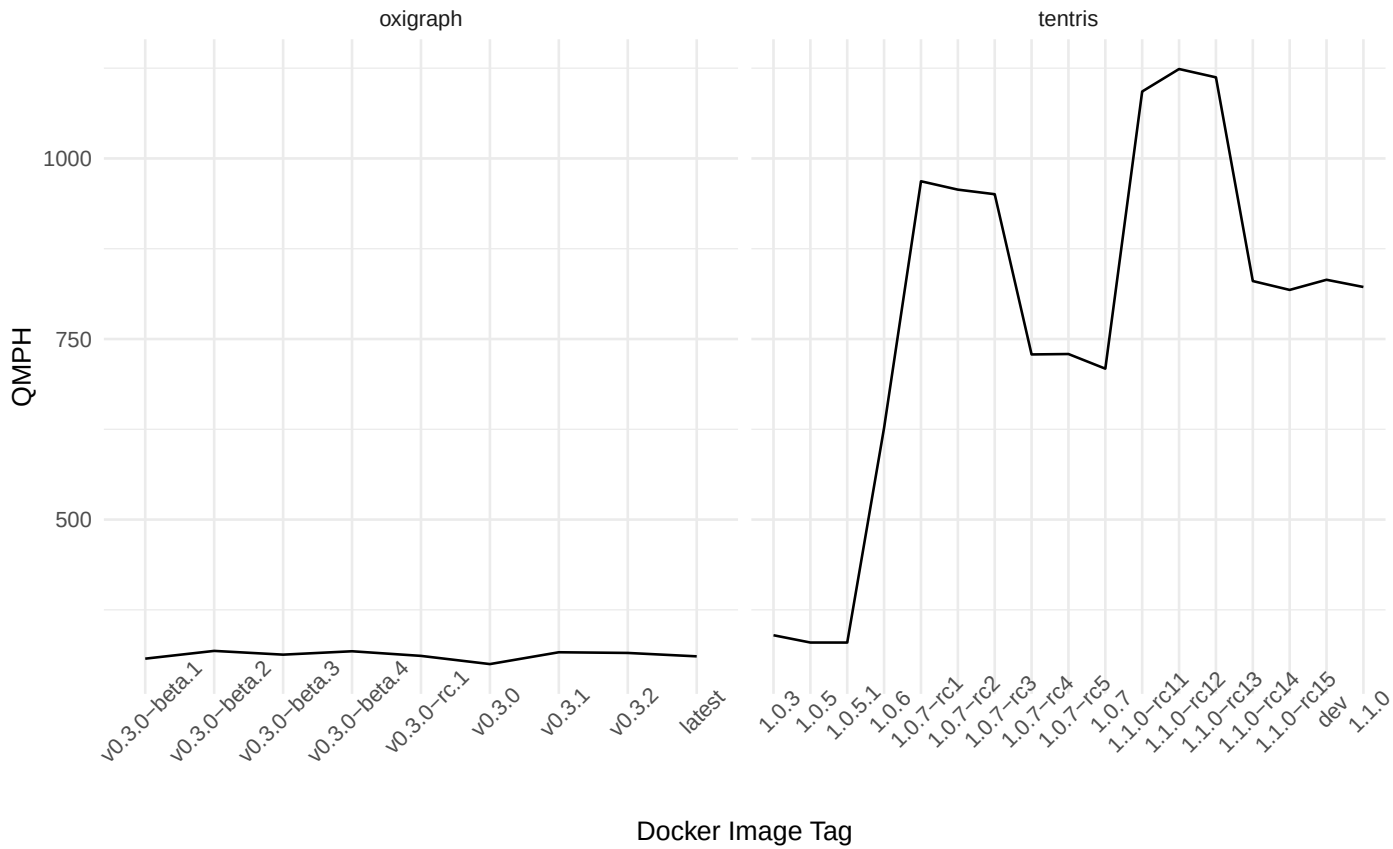
1. Finding initial benchmark setup
   - Setting up triplestore in Docker container
   - Loading dataset
   - Configuring IGUANA framework
   - Executing benchmark
2. Executing further benchmarks
   - Repeat same manual steps

Basilisk Process

1. Finding initial benchmark setup
   - Setting up triplestore in Docker container
   - Loading dataset
   - Configuring IGUANA framework
   - Executing benchmark
   - Transfer configuration to Basilisk platform
2. Executing further benchmarks
   - Automated through platform
   - Starting a manual job

Evaluation

- Using the platform multiple benchmarks were performed using the Semantic Web Dog Food (SWDF) benchmark
  - Benchmarked 16 versions of TENTRIS
  - 9 versions of Oxigraph



Future Work

- Implement benchmark process for GitHub repositories
- Development of an user management and rights system
- Development of the Basilisk frontend

Summary

- Continued the work on the Basilisk platform
- Implemented the whole benchmark process for Docker Hub repositories
- Deployed the platform in the University network
- Evaluated the platform and compared it to the manual benchmark process

Thank you for your attention!

Questions?