PADERBORN UNIVERSITY

Basilisk – Continuous Benchmarking for Triplestores

Fabian Rensing

Supervisor: Prof. Dr. Axel Ngonga
Paderborn University

June 15, 2022

Agenda

Contents

Continuous Benchmarking of Triplestores

Benchmark

Used to measure and compare the performance of systems with a defined set of operations and metrics

Triplestore

Specialized database for storing knowledge graphs

”Continuous”

Continuous / automatic performing of benchmarks on triplestores

Triplestores

Used for storing and accessing Knowledge graphs
knowledge graph is
TURTLE syntax
SPARQL endpoint

Benchmarks

Benchmarks for triplestores consist of dataset and query file

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?

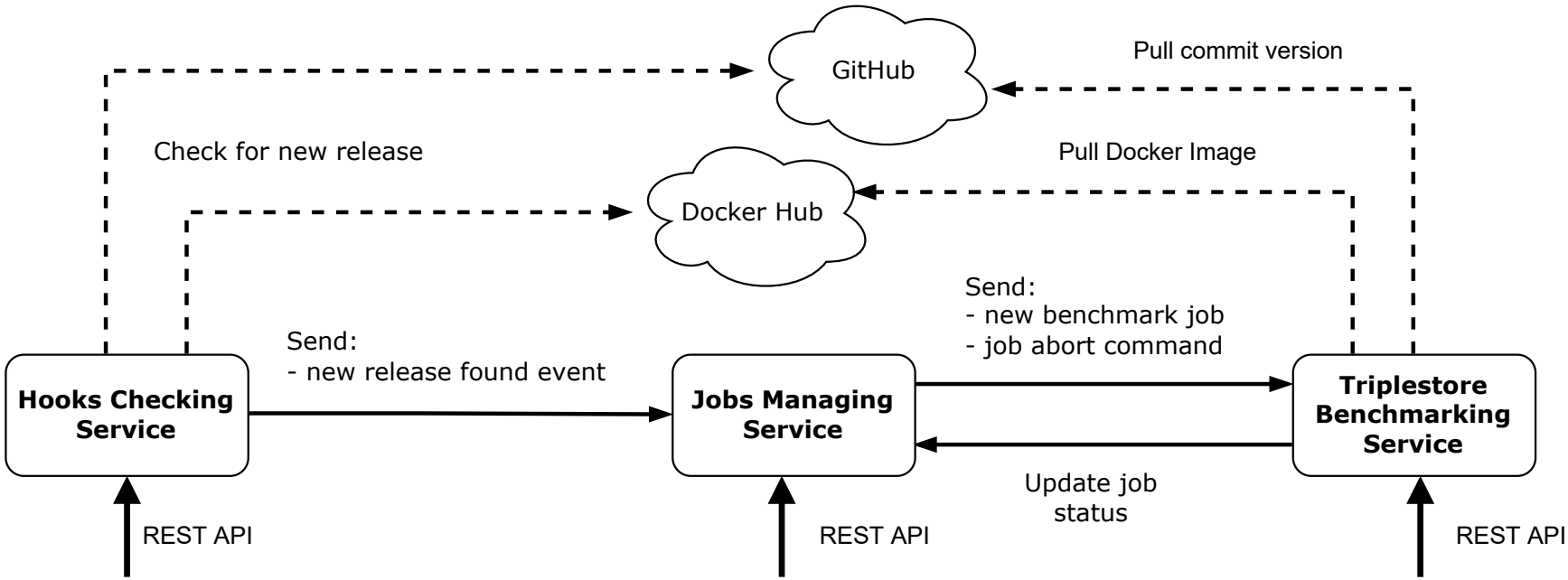
- Each benchmark requires a manual setup
  - Setting up and starting the triplestore
  - Loading the benchmark dataset
  - Configuring the benchmark framework (e.g. IGUANA)
  - Executing the Benchmark
- These steps are time-consuming, but not complicated
- Automating the benchmark process mitigates redundant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

The Basilisk Platform

Main Idea for the Platform

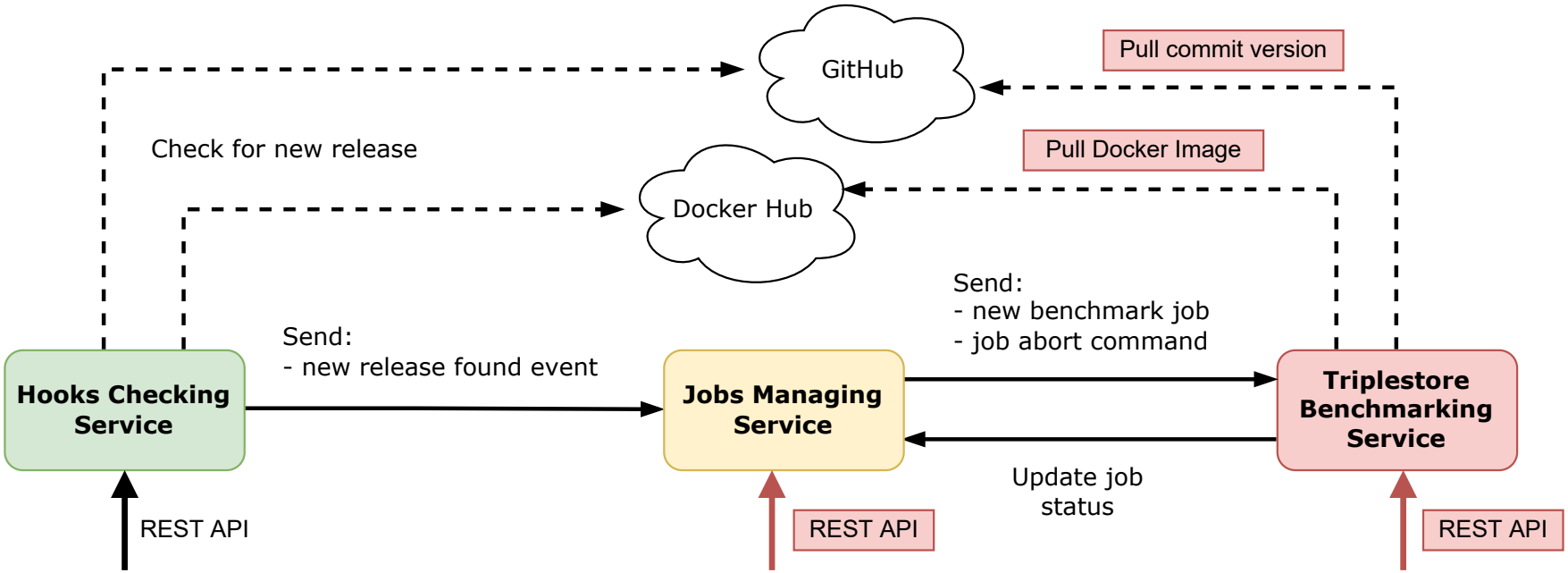
- Continuously check for new triplestore releases or Pull Request
- Automatically perform a benchmark if a new release is found
- Store the benchmark results in a triplestore

Platform Architecture



- User interaction via stateless REST endpoints
  - All configurations are done on the REST endpoints at the JMS
  - HCS and TBS have one endpoint for starting or stopping
- Services communicate via a RabbitMQ message broker

Implementation

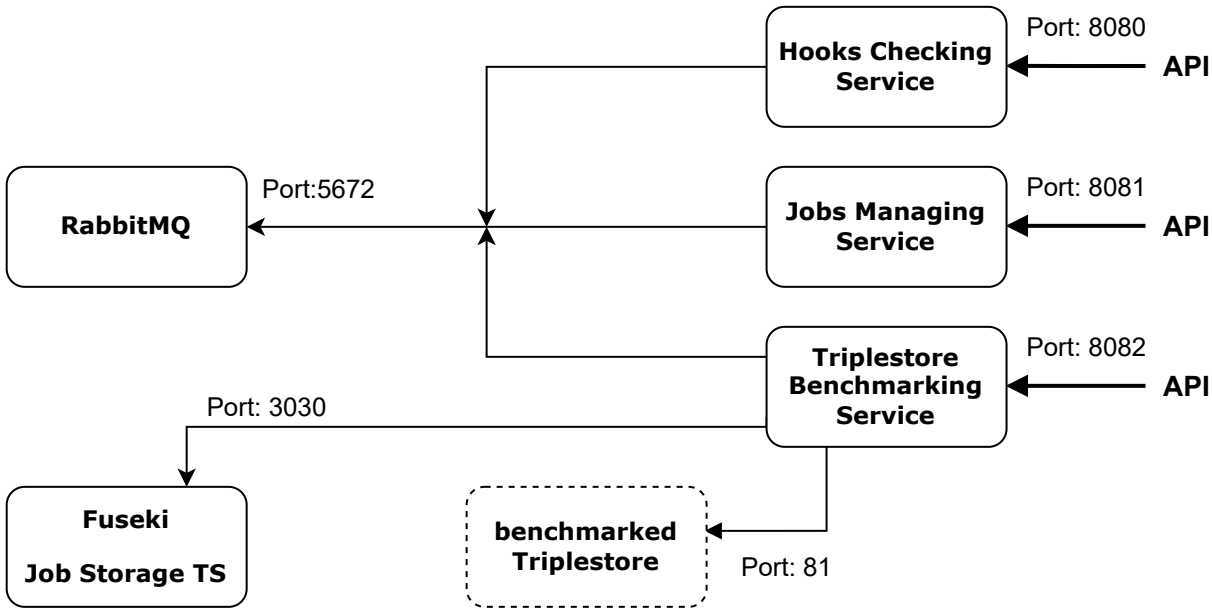


- Minor changes to data model
- Code restructure

- 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

Evaluation of Basilisk platform
Wie Evaluation
Is Basilisk platform helpful in Dev process?
How fast were benchmarks executed

Future Work

Benchmark process for GitHub repos.
User management

Summary

What has been done?
I continued the development of the Basilisk platform.
The platform now allows the user to setup a

Thank you for your attention!
References!

content...

Quick Start

Create your first slide:

1. Copy all *.sty files into a directory
2. Copy packages.tex into the directory
3. Create a .tex file and add the code listed below
4. Generate your slide using LaTeX

Listing 1: Minimal Example

```
\documentclass{beamer}
\usetheme{claw}
\input{packages.tex}
\begin{document}
\begin{frame} Hello World \end{frame}
\end{document}
```

Text Formatting

Predefined Styles

- You could emphasize important parts
  (Maybe distinguish between problems and solutions)
- Use alert to display **warnings**
- Use the url command (`https://dice-research.org/`) or the href command (DICE) for links
- Highlight "*predefined terms*" like brands and `TechnicalTerms` like software components

Text Formatting

Additional Commands

Use combinations for other concepts:

- Text styles: **bold**, *italic*, underlined, SMALL CAPS
- Font families: `monospaced`, sans serif, roman
- Text colors: bluedark, gray, magenta, blue, orange, purple, red, turquoise, green
- Text sizes: tiny, scriptsize, footnotesize, small, normalsize, large, Large, LARGE, huge, Huge

Code Listings & Frame Arguments

Use these arguments to configure frames:

`fragile`	Specially interpreted contents, e.g. for listings
`plain`	No headlines, footlines, sidebars; e.g. for large images To also remove background images use: `{\usebackgroundtemplate{}[...]}`
`squeeze`	Squeezes vertical spaces, e.g. for long contents
`shrink`	Shrinks frame, e.g. for long contents

Listing 2: Frame Options

```
1 \begin{frame}[fragile]{Code Listings \& Frame
    Arguments}
2   % [...]
3 \end{frame}
```

Mathematics & Miscellaneous

- Math[1]: $5^2 = 3^2 + 4^2$
- Equations:

$$\sum_{n=1}^{\infty} \frac{1}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots \tag{1}$$

[1] This is a footnote also working in columns

Blocks

This is a Block

- This is an item

1. This is enumeration item

This is an Example Block

- This is an item

1. This is enumeration item

This is an Alert Block

- This is an item

1. This is enumeration item

Blocks

This is a Block

- This is an item

1. This is enumeration item

This is an Example Block

- This is an item

1. This is enumeration item

This is an Alert Block

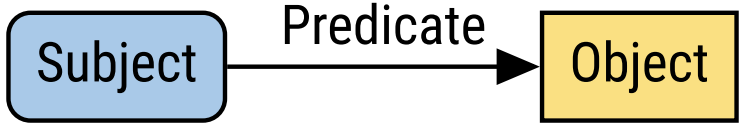
- This is an item

1. This is enumeration item

Tables

Topic	Content
Generator	Use tools like tablesgenerator.com
Large tables	Try the frame option `[shrink=.8]` (center table with `\hspace*{5cm}`)
Large tables	Combine the longtable package and the frame option `[allowframebreaks]`
Style	Try the booktabs package

Graphs



Thank you!

Questions?

Data Science Group at Paderborn University

Web: dice-research.org

Code: github.com/dice-group

Twitter: @DiceResearch

References I

[1] J. Wright, V. Miletić, and T. Tantau, “beamer – A LaTeX class for producing presentations and slides.” https://ctan.org/pkg/beamer.

[2] A. Wilke, “Claw LaTeX Beamer Template.” https://github.com/adibaba/templates.

Appendix

Predefined Base Colors

- primarybluedark
- primarybluelight
- primarygraylight
- primarygraydark
- secondarymagenta
- secondaryblue
- secondarygreen
- secondaryorange
- secondarypurple
- activeyellow
- activered
- activeturquoise
- activegreen
- specificblue

Appendix

Predefined Text Colors

- textdarkblue
- textgray
- textmagenta
- textblue
- textorange
- textpurple
- textred
- textturquoise
- textgreen
- textbluespecific

Appendix

Predefined Element Colors

- elementgray
- elementmagenta
- elementblue
- elementorange
- elementpurple
- elementyellow
- elementred
- elementturquoise
- elementgreen
- elementbluespecific

Appendix

Predefined Light Colors

- lightgray
- lightmagenta
- lightblue
- lightorange
- lightpurple
- lightyellow
- lightred
- lightturquoise
- lightgreen
- lightbluespecific

Appendix

Predefined Background Colors

- backgroundgray
- backgroundmagenta
- backgroundblue
- backgroundorange
- backgroundpurple
- backgroundyellow
- backgroundred
- backgroundturquoise
- backgroundgreen
- backgroundbluespecific