

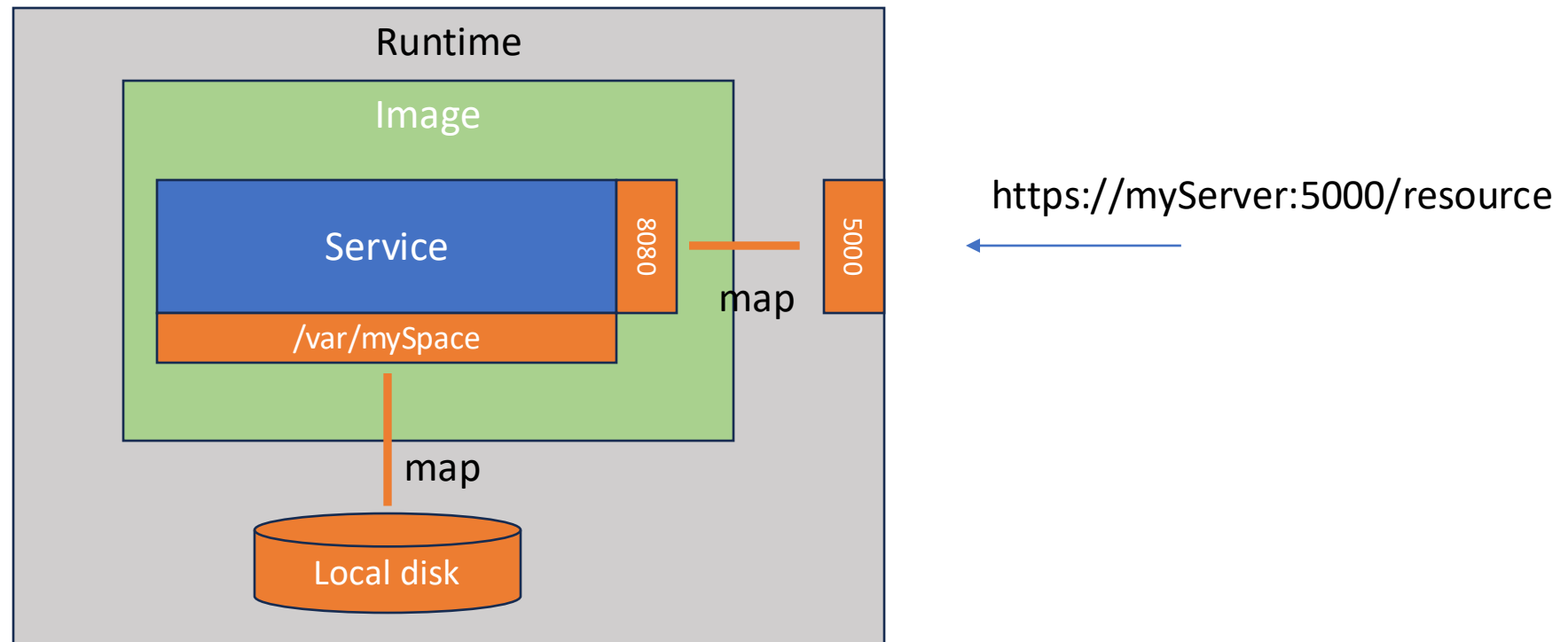
ACP / 3 + 4

Michael Glienecke, PhD

Welcome again

- A bit of recap
- Storing data

A microservice in docker

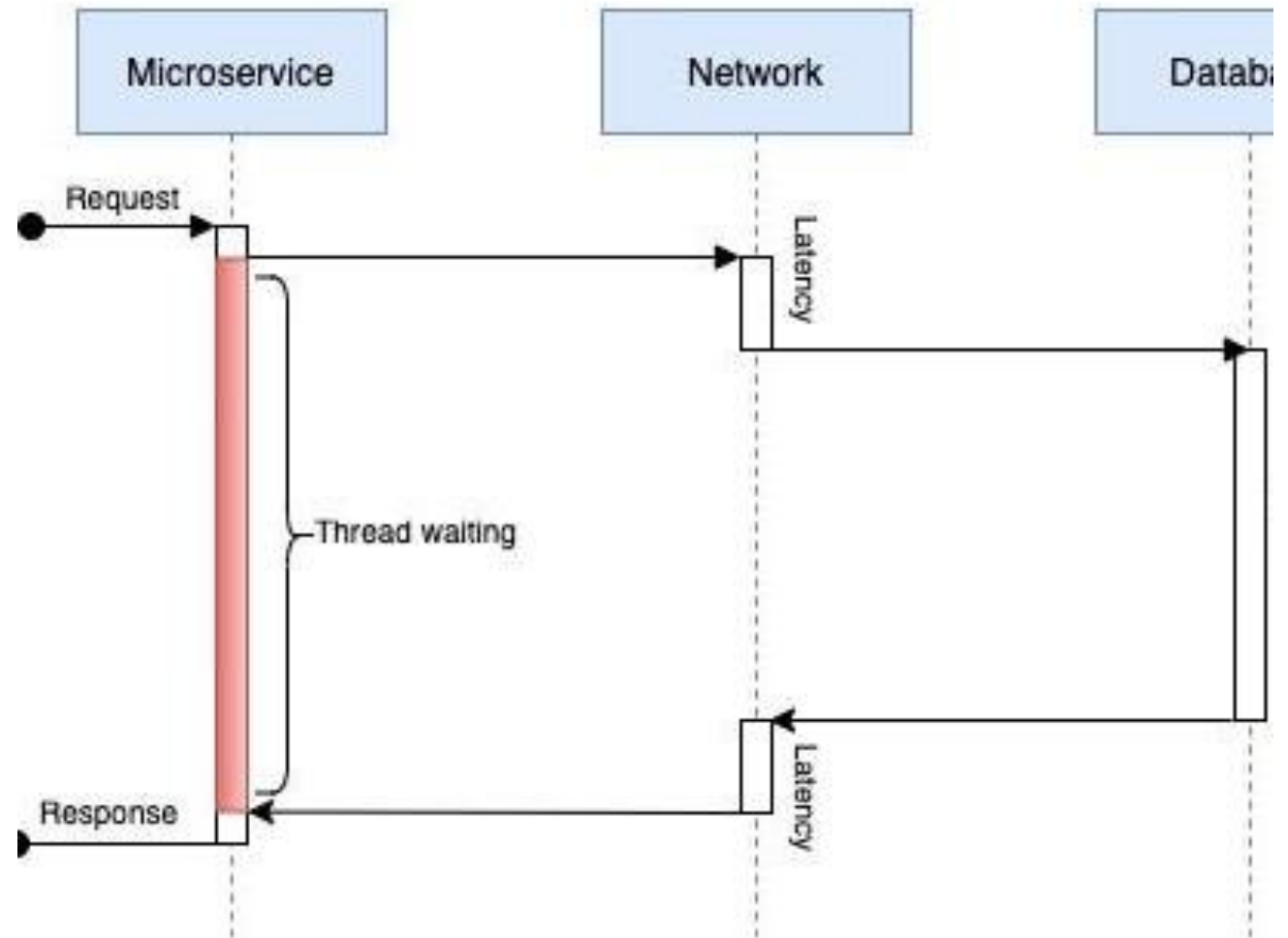


Microservices structure

Spring Boot uses Tomcat

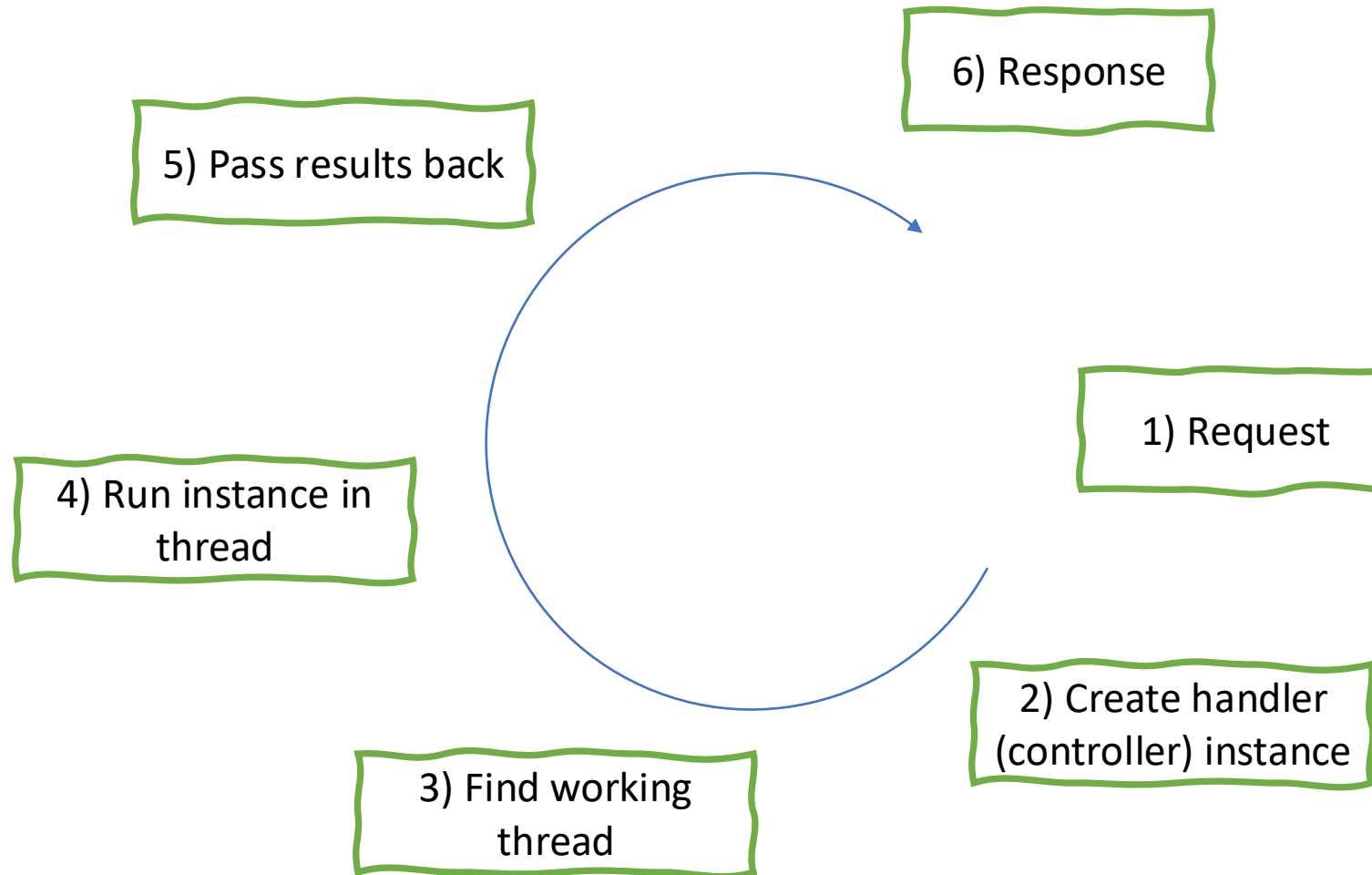
Some limitations (<https://oskar-uit-de-bos.medium.com/the-performance-challenge-in-java-microservices-e51cce3977e9>)

Blocking threads and consuming resources

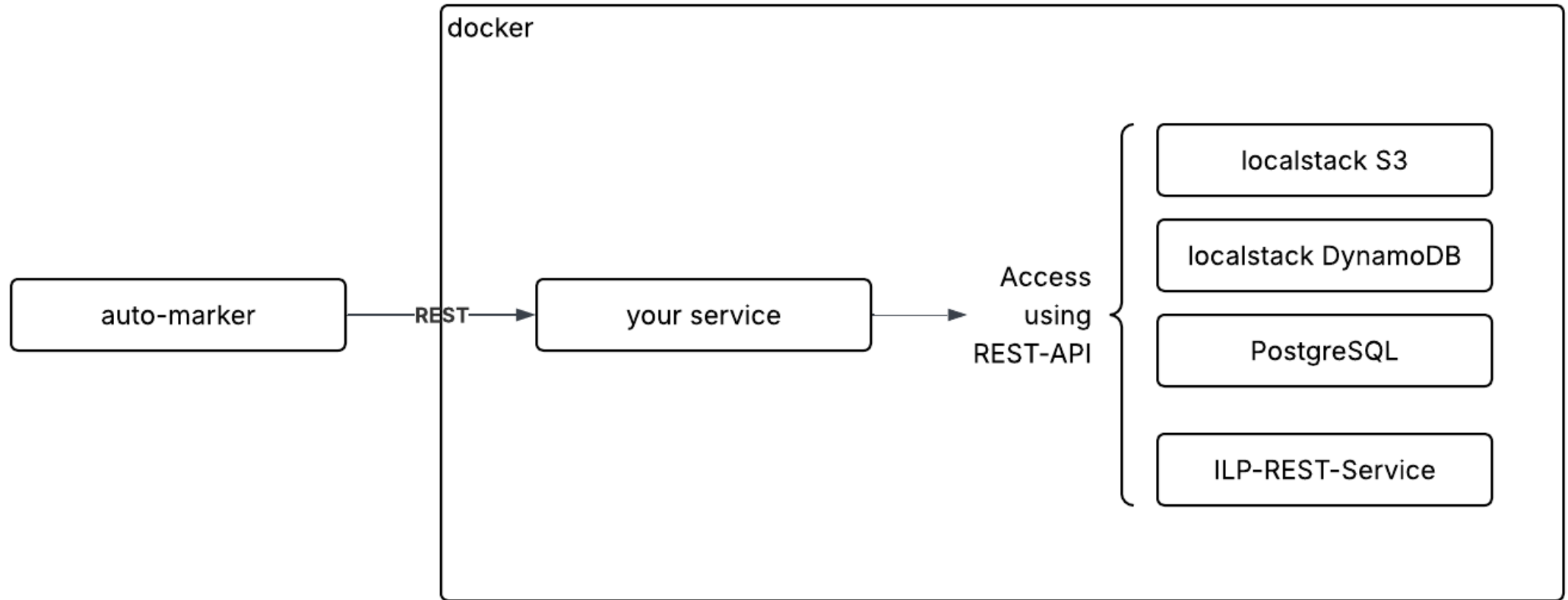


Inspection of a Java Spring Boot based service

The structure of a microservice



Service landscape



Looking at other languages for services



Pure Java (no Spring Boot)



Rust



Go

Why store data?

- Temporary data
- State
- General Persistence / long term
- Microservice structure / data
- Alternatives? In memory / queues, ...

Storage options

- Files
- SQL / No-SQL databases
 - Postgres, SQL Server, DB/2, ORACLE, ... (SQL)
 - DynamoDB (Key-Value database)
 - S3, Azure Blobs (Blob-Storage)
 - MongoDB / CosmosDB / MariaDB (No-SQL)

When do you use what?

- Files
 - Temporary, volatile things
- SQL
 - Transactional business Domain Data
 - Data Warehousing
- Key-Value database
 - Quick paced data with single key
 - Often huge volumes
 - Serverless Apps

When do you use what /2?

- Blob-Storage
 - Mass storage of data items (images, ...)
 - Big Data analysis
- No-SQL
 - Documents
 - JSON / XML data

What we will be using

- DynamoDB + S3 using localstack
 - UI in localstack UI
- PostgreSQL
 - dbeaver and pgAdmin as UI tools

Amazon DynamoDB Overview



Amazon S3 Overview



PostgreSQL Overview



localstack

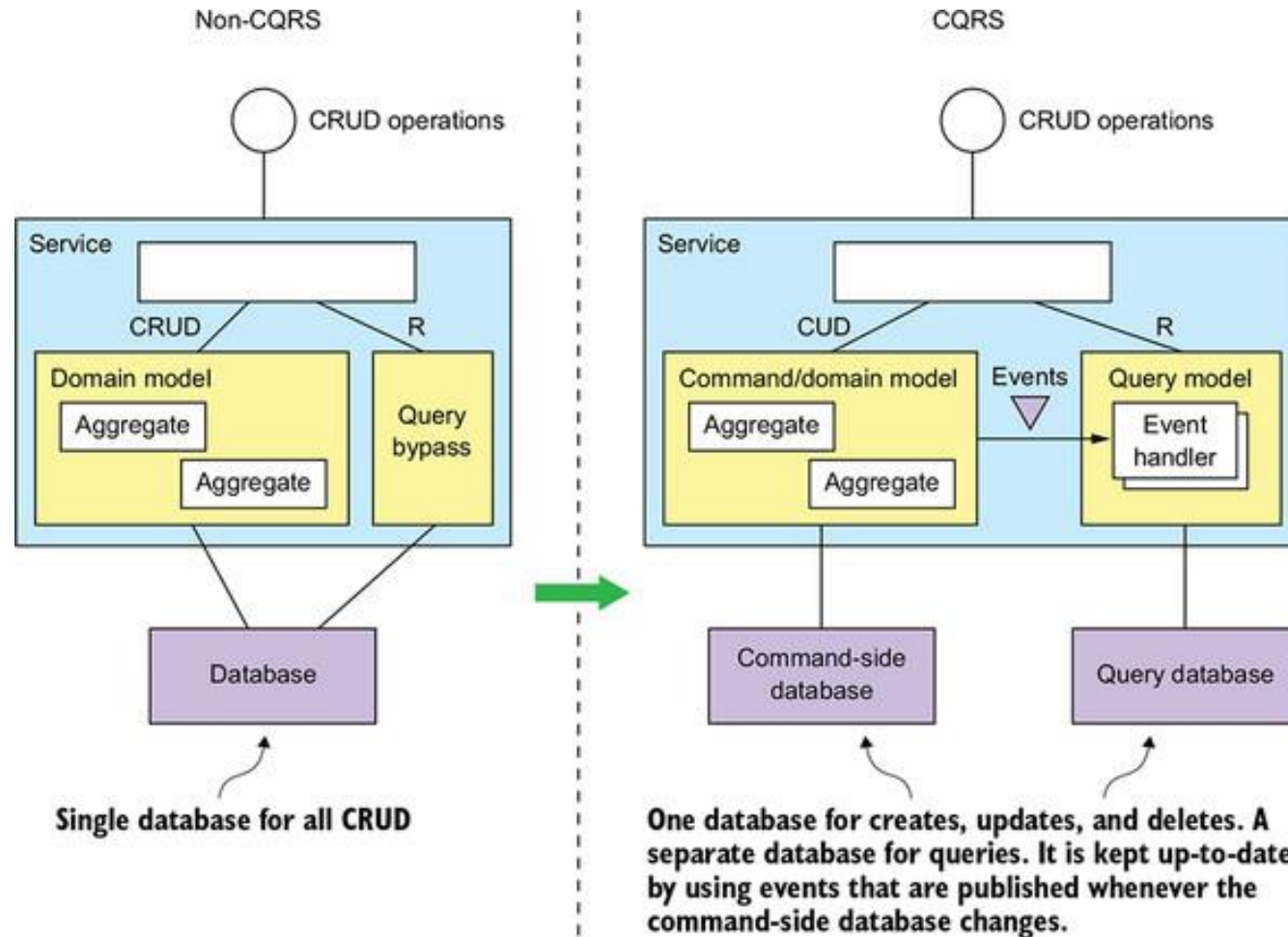
- <https://www.localstack.cloud/>
- Runs in your local system
- Emulates many (depending on level) aws-Services
- Access using the aws SDK (preferably v2)
<https://aws.amazon.com/sdk-for-java/>
- “test” / “test” as credentials for local usage

Localstack /2

- DynamoDB
 - https://docs.aws.amazon.com/code-library/latest/ug/dynamodb_example_dynamodb_Scenario_GettingStartedMovies_section.html
 - https://docs.aws.amazon.com/code-library/latest/ug/dynamodb_code_examples_actions.html
- S3:
 - https://docs.aws.amazon.com/code-library/latest/ug/s3_example_s3_Scenario_GettingStarted_section.html
- Localstack:
 - <https://docs.localstack.cloud/aws/integrations/aws-sdks/java/>

CQRS

Command Query Responsibility Segregation



CQRS Pros / Cons

- Enables the efficient implementation of queries in a microservice architecture
- Enables the efficient implementation of diverse queries
- Makes querying possible in an event sourcing-based application
- Improves separation of concerns
- More complex architecture
- Dealing with the replication lag