Wrocław University of Science and Technology

Large Scale Data Processing

Lecture 7 – Cassandra, Kafka, CQRS, ES

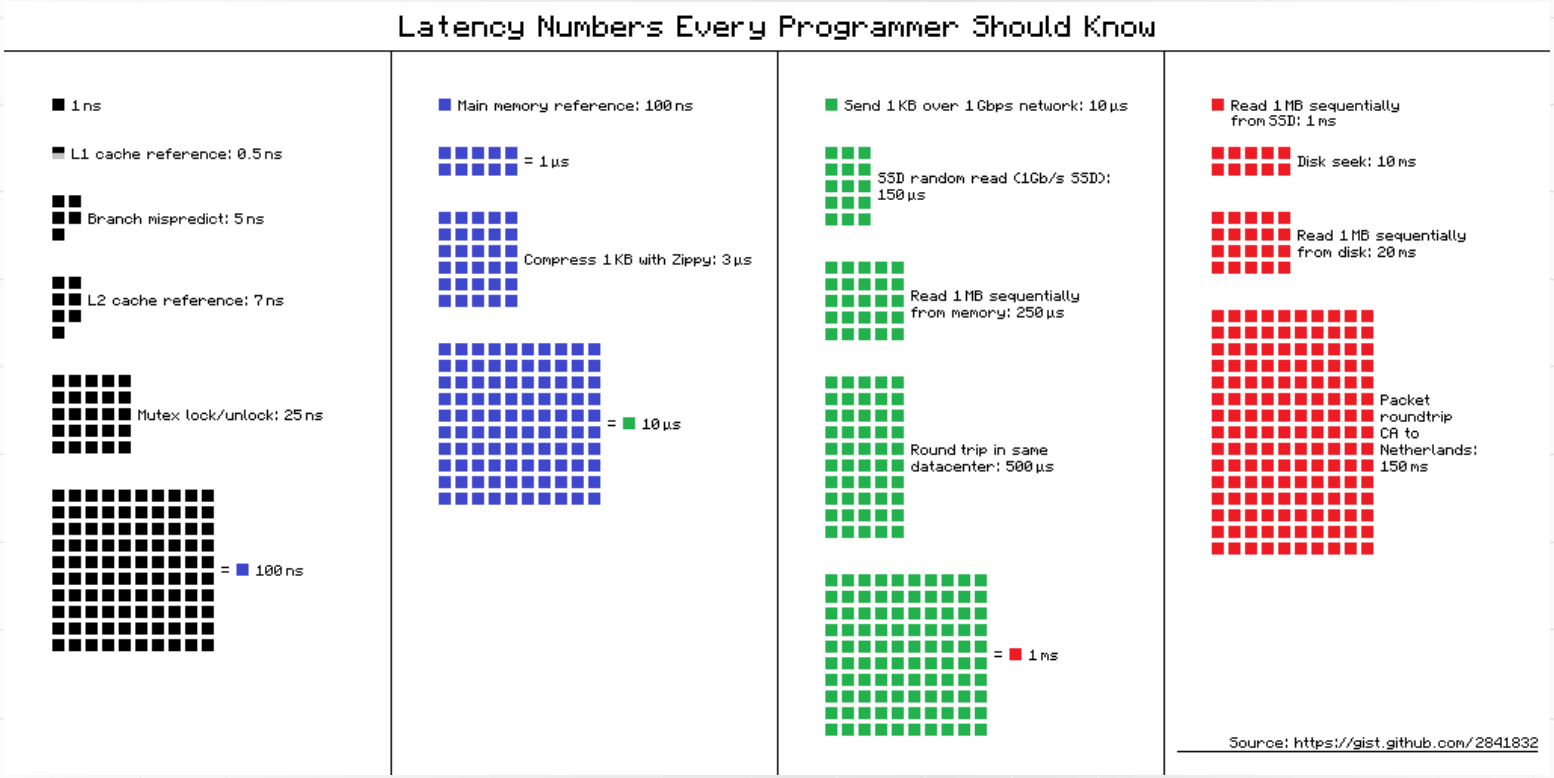
dr hab. inż. Tomasz Kajdanowicz, Piotr Bielak, Roman Bartusiak, Krzysztof Rajda

November 30, 2021

HR EXCELLENCE IN RESEARCH

Overview

Latencies


Latency Numbers Every Programmer Should Know
1 ns
L1 cache reference: 0.5 ns
Branch mispredict: 5 ns
L2 cache reference: 7 ns
Mutex lock/unlock: 25 ns
= 100 ns
Main memory reference: 100 ns
= 1 µs
Compress 1 KB with Zippy: 3 µs
= 10 µs
Send 1 KB over 1 Gbps network: 10 µs
SSD random read (1Gb/s SSD): 150 µs
Read 1 MB sequentially from memory: 250 µs
Round trip in same datacenter: 500 µs
= 1 ms
Read 1 MB sequentially from SSD: 1 ms
Disk seek: 10 ms
Read 1 MB sequentially from disk: 20 ms
Packet roundtrip CA to Netherlands: 150 ms
Source: https://gist.github.com/2841832

Overview

Latencies

Cassandra

Kafka

CQRS

Event sourcing

Cassandra

General

Cassandra

- multi-master
- linear throughput
- online cluster growth
- partitioned queries
- schema

Data organization

Cassandra

- Keyspace
- Table
- Partition
- Row
- Column

Data organization

Cassandra

- Sparse
- Cell

Unsupported

Cassandra

- Cross partition transactions
- Distributed joins
- Foreign keys or referential integrity.

Why - Unsupported

Cassandra

- cross-partition coordination
- a lot of communication
- slow
- hard to provide in HA

Supported

Cassandra

- single partition transactions
- secondary indices
- materialized views
- collections
- user defined types, aggregates, functions

Wrocław University of Science and Technology

Dynamo like

Cassandra

- Amazon
- ring membership
- partitioning using consistent hashing
- multi-master
- gossip protocol
- incremental scale-out

Consisten hashing

Cassandra

- hash function
- "modulo"
- replication
- last-write wins

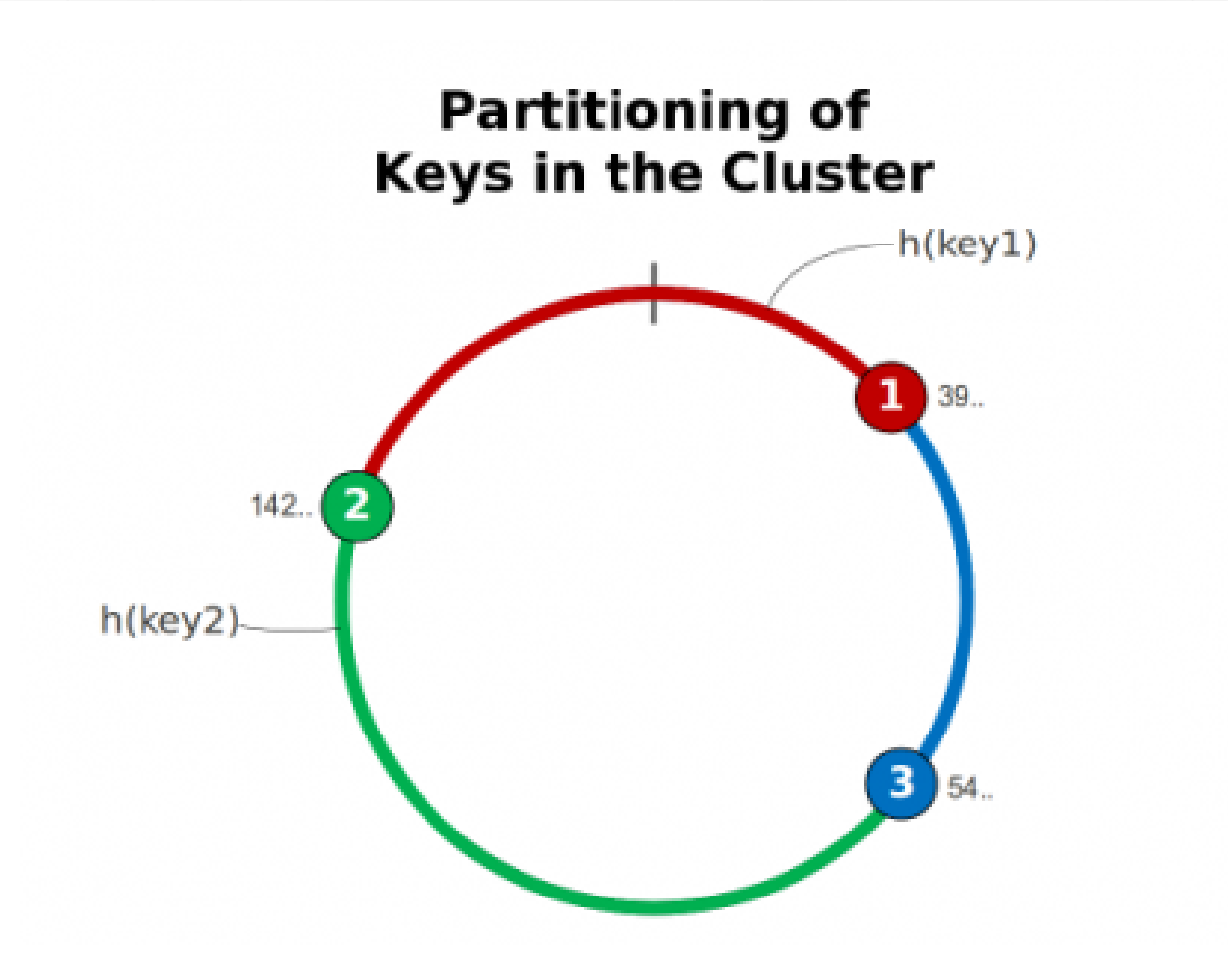
Token ring

Cassandra

- hash function values
- module -> forms ring
- no master node

Token ring

Cassandra



Consistency

Cassandra

- ONE
- QUORUM
- ALL

Wrocław University of Science and Technology

Network topology strategy

Cassandra

- multiple data-centers
- eventual-consistency
- clockwise walking the ring
- same data center

Gossip protocol

Cassandra

- peer-to-peer
- one node does not need to talk to all other nodes
- gossip data and nodes health
- avoiding communication chaos
- gossip when nodes are exchanging information

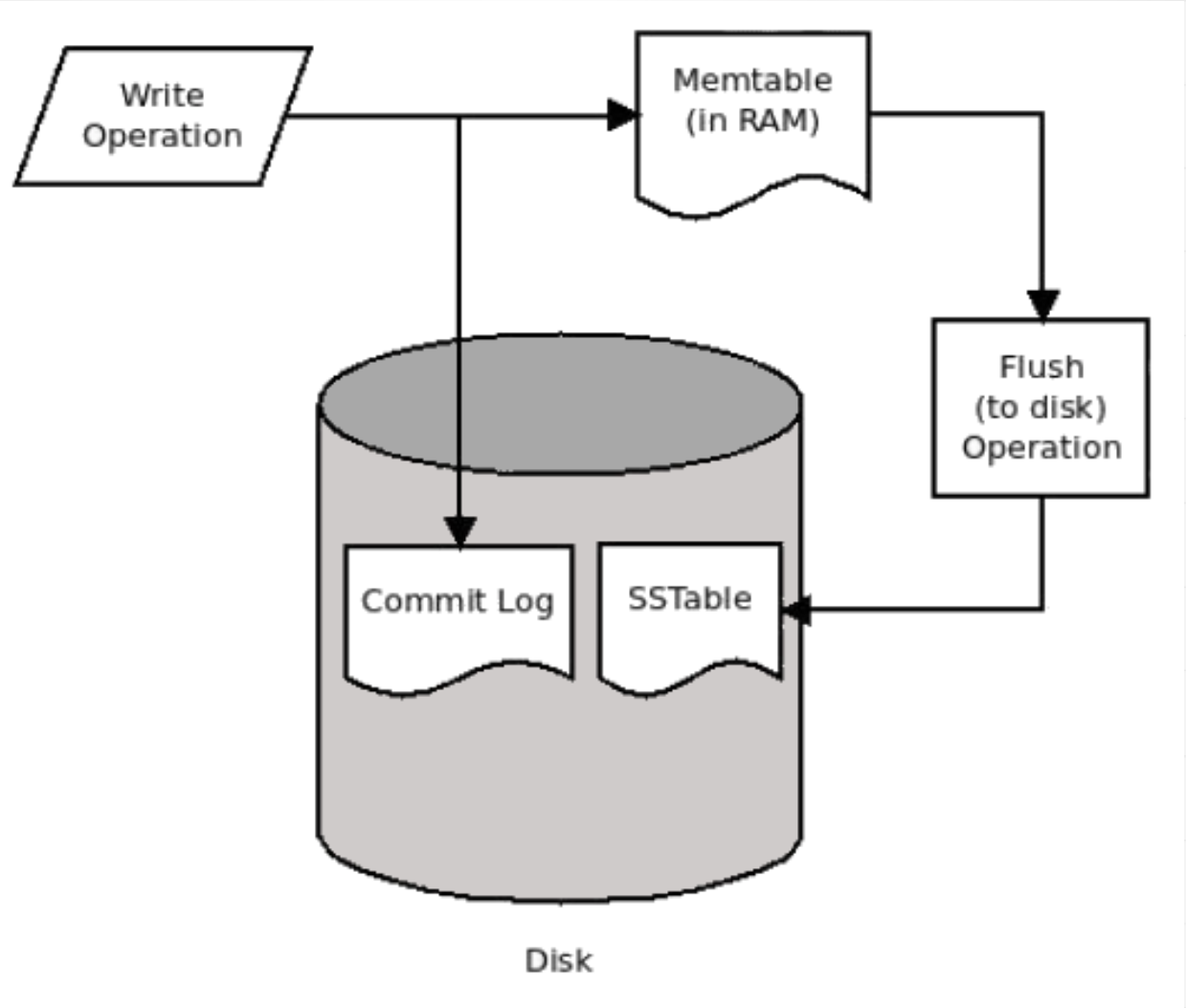
Write path

Cassandra

- simple
- fast
- commit log
- memtable
- sstable

Write path

Cassandra



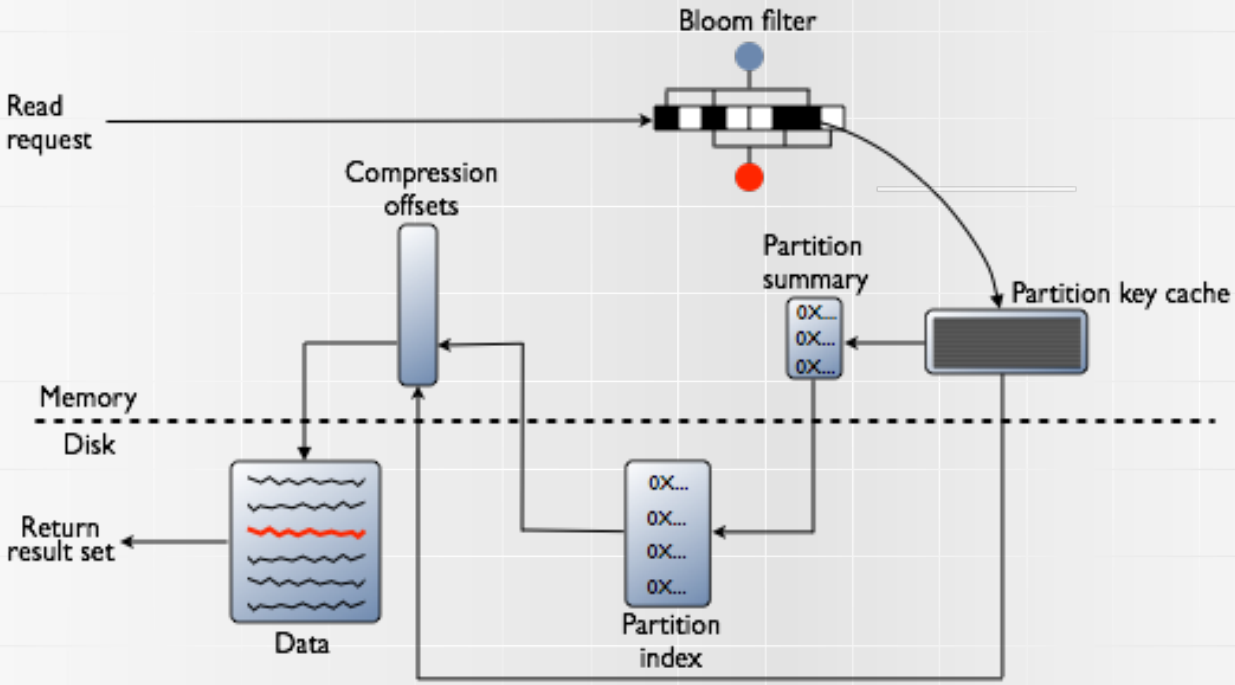
Read path

Cassandra

- complex
- count(*) - can fail
- bloom-filters
- caches

Read path

Cassandra



Bloom filter

Cassandra

- probabilistic
- false-positives
- bit vector
- k hash functions
- partition-key based

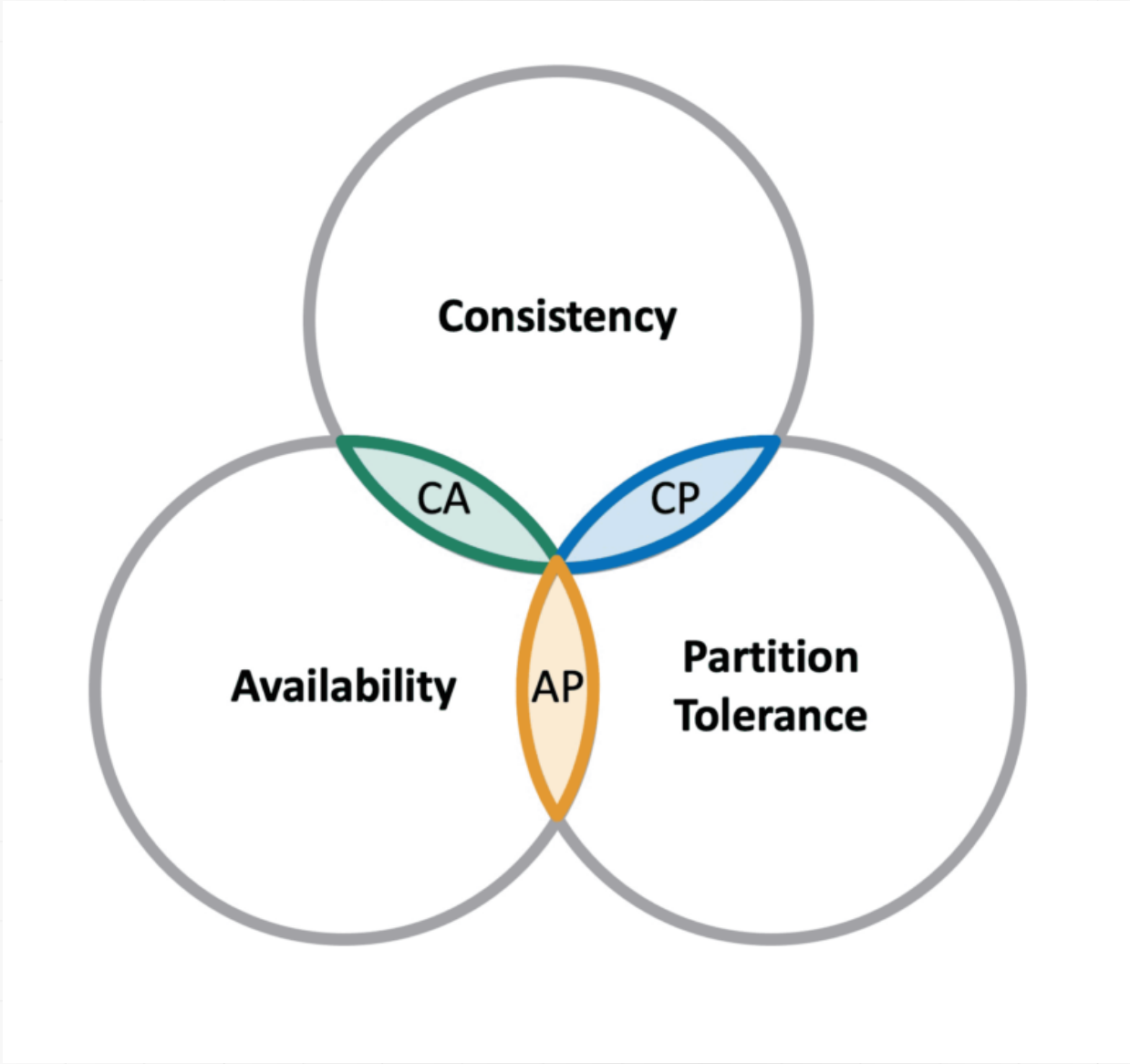
CAP

Cassandra

- Consistency
- Availability
- Partition tolerance

CAP

Cassandra



Guarantees

Cassandra

- High Scalability
- High Availability
- Durability
- Eventual Consistency of writes to a single table
- Lightweight transactions with linearizable consistency
- Batched writes across multiple tables are guaranteed to succeed completely or not at all
- Secondary indexes are guaranteed to be consistent with their local replicas data

Eventual consistency

Cassandra

- all updates will reach all nodes eventually
- divergent versions of same data
- trade-off

Tombstones

Cassandra

- deletes
- insert/update null
- internal operations (materialized views)
- compaction

Overview

Latencies

Cassandra

Kafka

CQRS

Event sourcing

Kafka

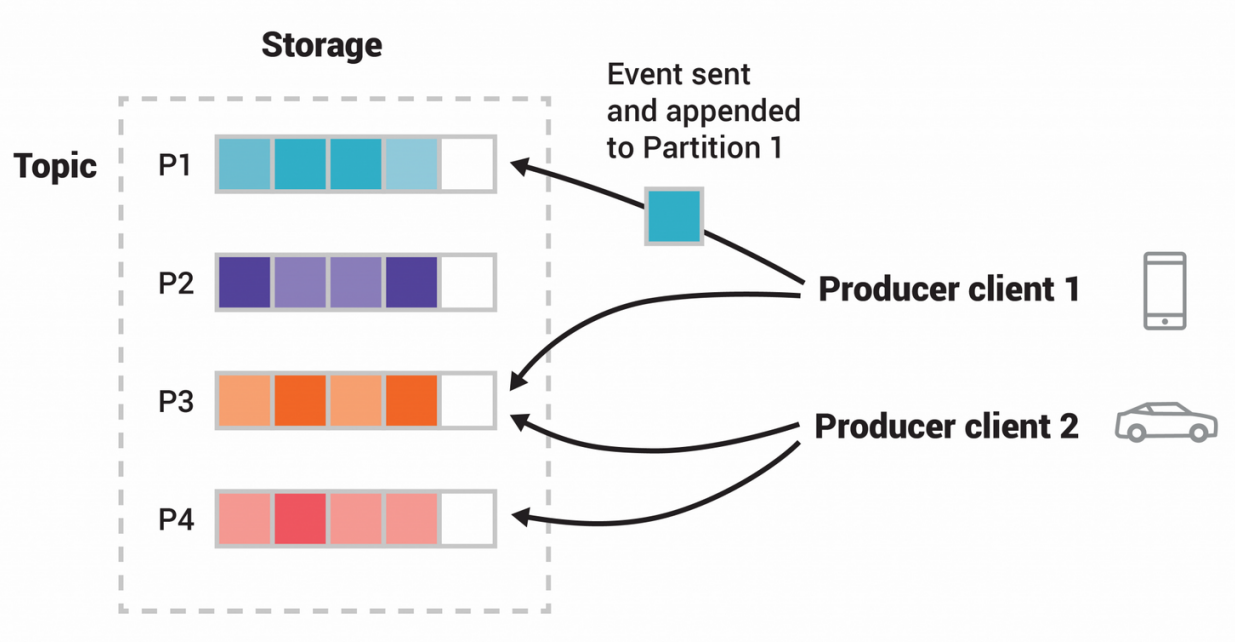
Concepts

Kafka

- event
- producer
- consumer
- topic
- partitioning
- replication

Concepts

Kafka



Event streaming platform

Kafka

- end-to-end
- publish and subscribe to
- store
- process

How it work

Kafka

- servers
- clients
- TCP
- storage layer + Kafka Connect
- fault-tolerant
- scalable

Storage

Kafka

- filesystem
- linear vs random
- pagecache
- filesystem
- OS-optimization
- JVM
- os-cache -> always warm
- all data saved to disk without flush

Efficiency

Kafka

- persistent queue -> linear operations
- constant time
- binary message format
- sendfile

Wrocław University of Science and Technology

Load balancing

Kafka

- client-routing
- data directly to partition leader
- client decides how

Batching

Kafka

- message set
- network optimization
- storage optimization
- compression

Push vs pull

Kafka

- push
  - server needs to manage semantics
  - diverse consumers
  - how to make consumer get at max pace
- pull - Kafka
  - consumer decides
  - more aggressive batching (!!!)
  - busy waiting -> long polling

Position

Kafka

- metadata
- agreement
- ACK -> slow
- consumer groups
- single integer - offset

Semantics

Kafka

- at-most once
- at-least once
- exactly-once

Semantics

Kafka

- each producer has ID
- each message in partition has ID

Replication

Kafka

- failure protection
- one broker is a leader of partition
- responsible for replication
- followers consume from leader
-

Leader election

Kafka

- in-sync replicas
- controller - batching ellections

Overview

Latencies

Cassandra

Kafka

CQRS

Event sourcing

Wrocław University of Science and Technology

CQRS

- Command Query Responsibility Segregation
- microservices
- database per service
- how to join?
- separation of concerns
- eventual consistency

Overview

Latencies

Cassandra

Kafka

CQRS

Event sourcing

Event sourcing

- update DB and send domain event
- how to do it to have guarantees?
- model entity changes as set of events
- services can subscribe to those events

Wrocław University of Science and Technology

Next week

Event sourcing

- Produkcyjne aspekty utrzymywania i wdrażania aplikacji

Wrocław University of Science and Technology

Large Scale Data Processing

Lecture 7 – Cassandra, Kafka, CQRS, ES

dr hab. inż. Tomasz Kajdanowicz, Piotr Bielak, Roman Bartusiak, Krzysztof Rajda

November 30, 2021