



APACHE
kafka®

A distributed streaming platform

Michael Sieber

Hochschule München – IF6

SS2019

Gliederung

1. Grundprinzip
2. Vor- / Nachteile
3. Live Coding
4. Fazit

Kafka

- Open Source Streaming Plattform
- „Publish Subscribe“ System
- verschiedene API's: u.a. Consumer API, Producer API und Streaming API
- Kann auf mehrere Server verteilt in einem Cluster laufen

Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Records

- Key – Value Paar
- Binär abgespeichert -> Serializer und Deserializer
- Timestamp

Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Topic

- Records werden Topics zugeordnet
- 1 Record → 1 Topic
- Producer schreiben Records in Topics
- Consumer konsumieren Records aus Topics

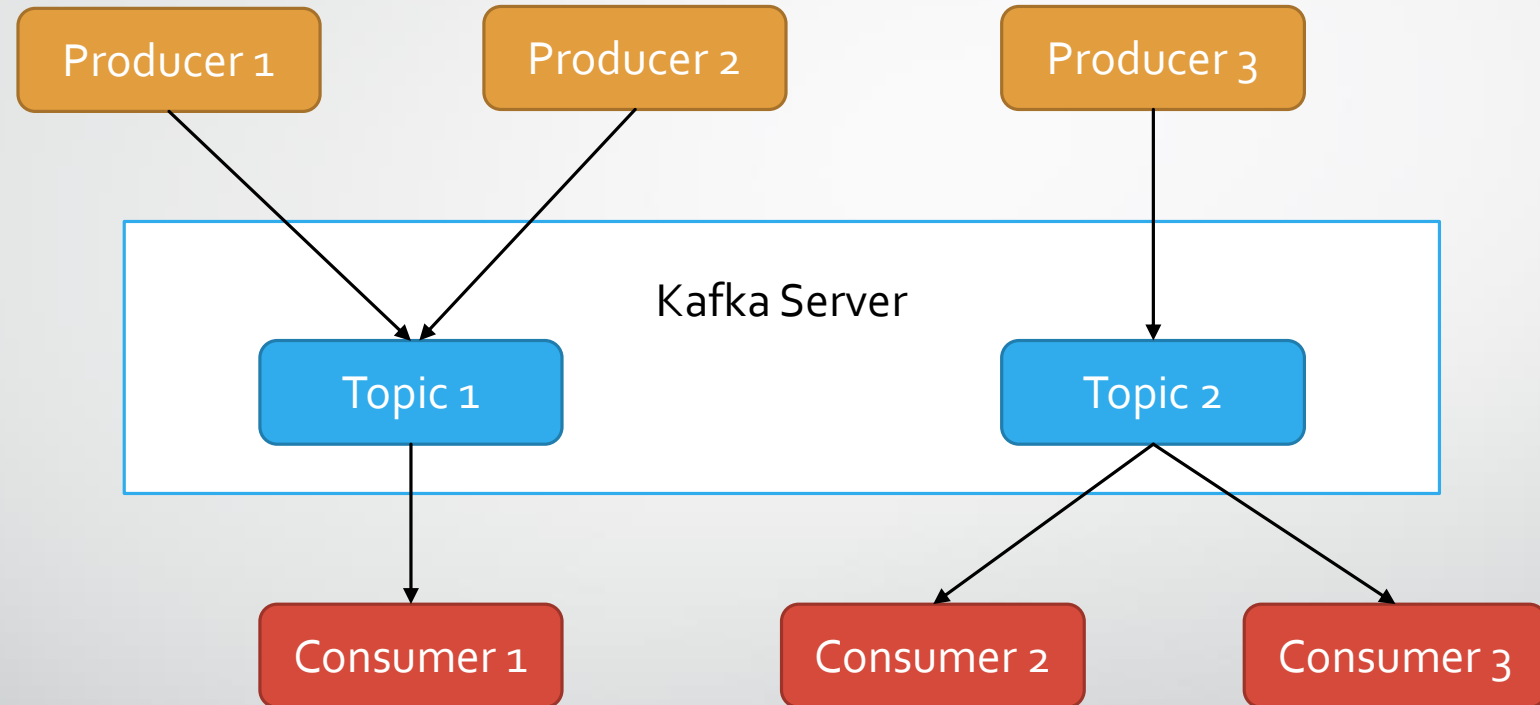
Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Übersicht



Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Pratition

- Ein Topic wird in mehrere Partitionen unterteilt
 - Records werden abhängig vom Key einer Partition zugewiesen
- Topics können über mehrere Server verteilt werden → Lastverteilung und Topics können größer als ein einzelner Server werden

Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Replica

- Partitionen können repliziert werden um gleiche Daten auf mehreren Servern zu speichern
- Ein Replica ist dabei immer der Leader in den die Producer schreiben (die anderen folgen dem Leader)

→ Ausfallsicherheit

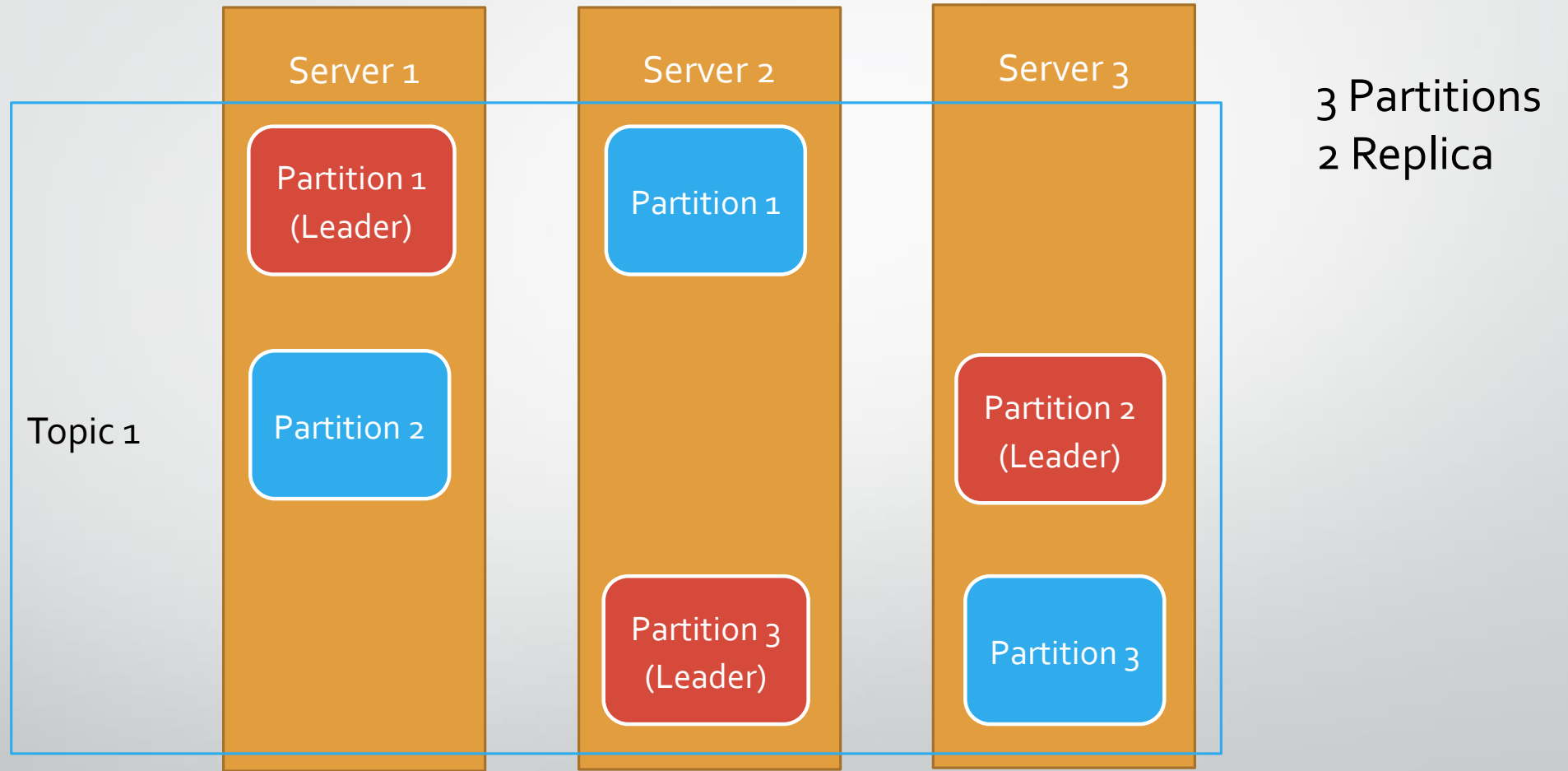
Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Übersicht



Offset

- Innerhalb einer Partition bekommt ein Record ein Offset zugewiesen
 - Nach Lesen des Records wird dieser nicht gelöscht, sondern der Consumer selbst schiebt seinen Offset weiter
- Records werden (je nach Konfiguration) für eine bestimmte Zeit gespeichert unabhängig ob schon gelesen oder nicht

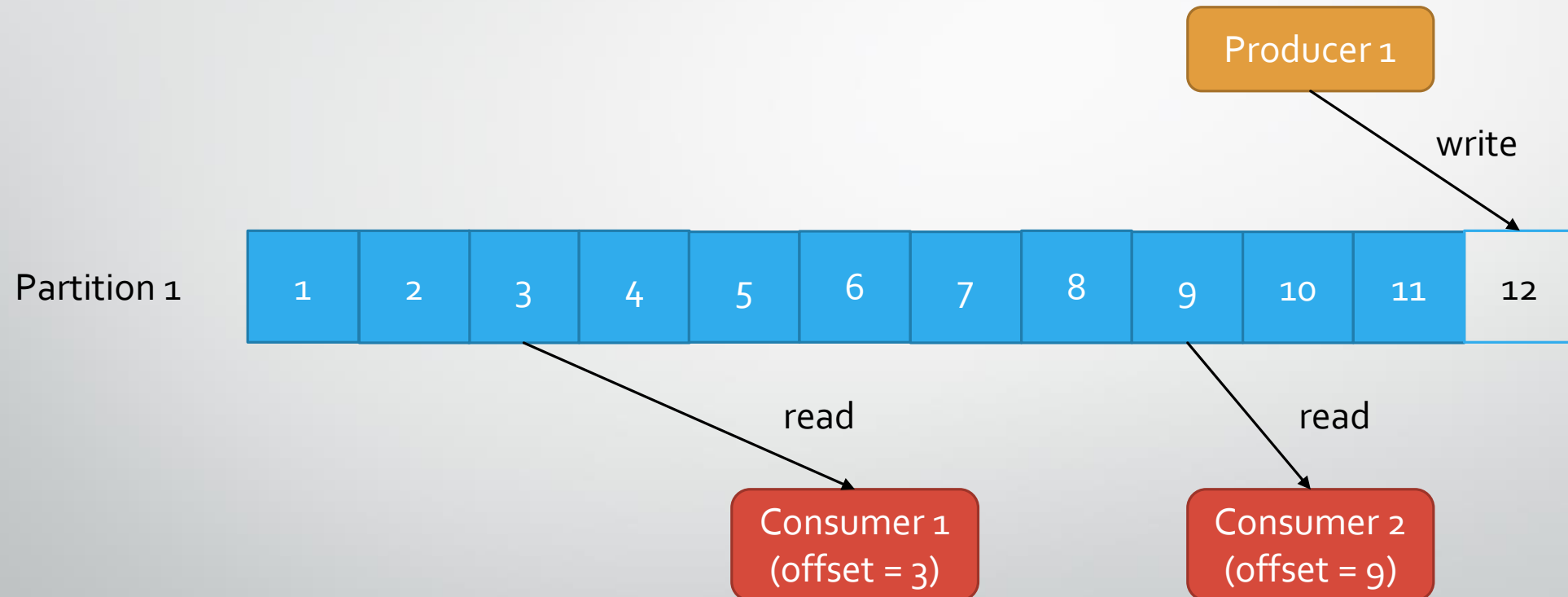
Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Offset



Consumergruppen

- Mehrere Consumer können sich zu seiner Gruppe zusammenschließen
 - Ein Record wird innerhalb einer Gruppe nur einmal gelesen
 - Ein Consumer innerhalb einer Gruppe wird einer oder mehreren Partitionen zugeordnet
- Anzahl Partitions \geq Anzahl Consumer einer Gruppe

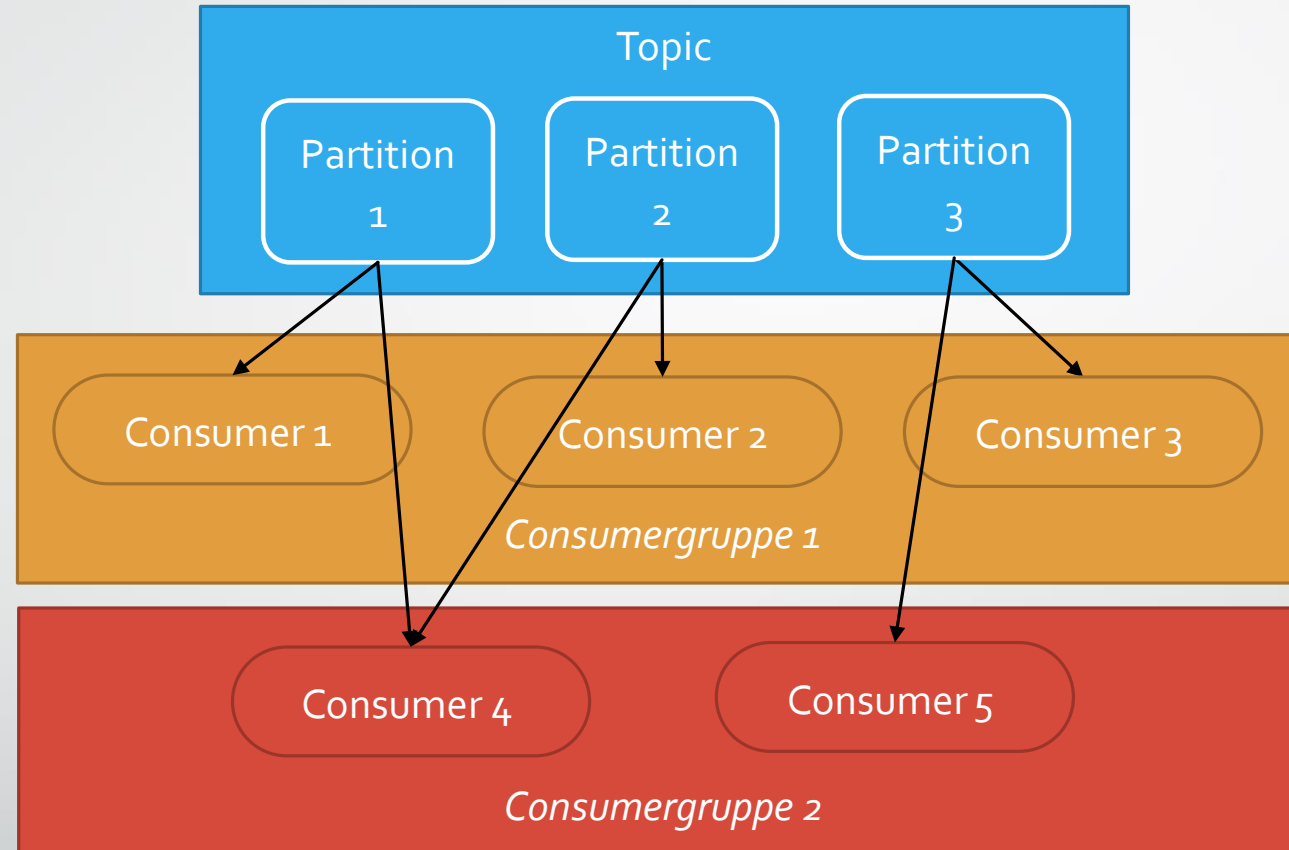
Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Übersicht



Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Vorteile

- Extrem ausfallsicher
- Sehr gut skalierbar
- Hohe Verfügbarkeit
- Parallele Verarbeitung möglich (Consumergruppen)
- Bietet Streaming API an

Nachteile

- Zusätzliche Technologie die eingeführt und gewartet werden muss
 - Zeit- und Kostenaufwand
 - Zusätzliche Kompetenzen nötig

Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Live Coding

Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit

Fazit

- Gute Möglichkeit, um Daten zwischen Microservices auszutauschen (siehe Vorteile)
- Einarbeitungszeit (bis auf Konfiguration) im Verhältnis zum Nutzen überschaubar
- Nicht in allen Architekturen sinnvoll



Grundprinzip

Vor- /
Nachteile

Live
Coding

Fazit



APACHE
kafka®

A distributed streaming platform

Fragen ?