Advanced Data Management

Redis Intro

Peyer Lars

Berger Adrian

Version 1.0

29. Oktober 2019

1 Key Value Store

1.1 Definition

Ein Key Value Store basiert auf einer Tabelle mit genau zwei Spalten:
In der einen befindet sich ein eindeutiges Identifikationsmerkmal, der Schlüssel (Key), in der anderen der Wert (Value). Somit wird jeder Wert eindeutig einem Schlüssel zugewiesen. Welche Datentypen im Wert gespeichert werden können, ist abhängig vom verwendeten Key-Value Store. Es handelt sich somit um eine NoSQL Datenbank.

1.2 Vorteile und Nachteile

1.2.1 Vorteile

- Geschwindigkeit
- Skalierbarkeit
- Simples Model, einfach verständlich

1.2.2 Nachteile

- Für Daten mit vielen relationalen Abhängikeiten sehr komplex
- Keine Lookup Optimierungen für das Suchen

1.3 Einsatzgebiete

Überall wo schnelle Zugriffszeiten bei großen Datenmengen benötigt werden, eignen sich Key Value Stores. Typische Einsatzgebiete sind deshalb Warenkörbe in Onlineshops oder das Speichern von Session-Daten.

1.4 In-Memory

Disk und SSD - Charakteristiken, Stärken und Grenzen Die Bedeutung von In Memory Datenbanken

1.5 Top 5 Key Value Stores

Gemäss db-engines.com (Stand 27.10.2019) sind die folgenden Datenbanksysteme führend:

1. Redis
2. Amazon DynamoDB
3. Microsoft Azure Cosmos DB
4. Memcached
5. Hazelcast

2 Redis

2.1 Charakterisierung

Entwickler Lizenzmodell Standards weitere Charakteristiken

2.1.1 Platformen, welche Redis verwenden

https://redis.io/topics/whos-using-redis

2.2 Architektur anhand von Redis

2.3 Installation

Server und Client

2.4 Datentypen

https://redis.io/topics/data-types

2.5 Abfragesprache

https://www.cheatography.com/tasjaevan/cheat-sheets/redis/

2.5.1 Transaktionen

Transkaktionsverhalten - Theoretisch und Testszenario (Demo in der Präsentation) https://redis.io/topics/transactions

2.6 Benchmarks

https://redis.io/topics/benchmarks

2.7 Persistenz

Concurrency Verhalten, Serialisierbarkeit https://redis.io/topics/persistence

2.8 Replikation

Verteilte Datenhaltung, Skalierbarkeit, Protokolle, Architektur https://redis.io/topics/replication

2.9 Optimierungsmöglichkeiten

2.10 GUI

2.11 API

mit Demo