

Non-Relational Databases

Redis

Prof. Dr. Jürgen Heym

Hof University of Applied Sciences

1 – NRD Introduction

- Important questions

Which database/datastore or which combination of databases/datastores may solve a given problem?

- Q 1: *What is the type of the datastore?*
- Q 2: *What was the motor for the development of a given datastore?*
- Q 3: *How do we talk to the datastore?*
- Q 4: *What makes a datastore unique?*
- Q 5: *What kind of solutions provides a datastore?*
- Q 6: *How scales a datastore?*

1 – NRD Introduction

- Q 1: *What is the type of the datastore?*
 - relational (PostgreSQL)
 - Key / value (Riak, Redis, Hazelcast)
 - column-oriented (Hbase)
 - document-oriented (MongoDB, CouchDB)
 - graph-based (Neo4J)

1 – NRD Introduction

- Q 6: *How scales a datastore?*
 - Scaling features are in tight connection to performance of the datastore!
 - Scaling features need a context!
 - Horizontal scaling features (MongoDB, Hbase, Riak)
 - Vertical scaling features (PostgreSQL, Neo4J, Redis, HazelCast)
 - Mixed scaling features

1 – NRD Introduction

- Types of datastores
 - key/value or KV datastore
 - Relate keys to values
 - comparable to hash tables or maps in programming languages
 - Pro: sometimes we can complex data types directly
 - hashes or lists
 - Pro: in some cases we can iterate over keys
 - Pro: Huge performance when sending simple requests
 - Con: not helpful for complex requests and aggregation tasks
 - Example: filesystem (key=filename, value=content)
 - More examples: Redis, Hazelcast, Riak, memcached, memcachedb, membase, Voldemort , ...

2 – Redis – Introduction

- Redis --- in short
 - In-Memory-Datastore
 - Data structure Datastore
 - Open-Source (BSD-License)
 - May be used as a database
 - May be used as a cache
 - May be used as a message broker

2 – Redis – Introduction

■ Redis --- Features

- Redis is not only a simple key-value-datastore that permits only strings.
- Redis is a data structure datastore that permits complex data structures!
- Supported Datastructures
 - Strings, hashes, lists, sets, sorted sets with range queries, bitmaps, hyperloglogs and geospatial indexes with radius queries
- Replication, Lua Scripting, LRU eviction, Transactions
- Different levels of on-disk persistence
- High availability via Redis Sentinel
- Automatic partitioning with Redis Cluster.

2 – Redis – Introduction

■ Lab activity

Answer the assigned question within at most 2 PPT slides.
Use the given PPT template (see moodle).

- Q 1: What are “Redis --- hyperloglogs”?
- Q 2: What are “Redis --- geospatial indexes with radius queries”?
- Q 3: What means “Redis --- Replication”?
- Q 4: What means “Redis --- Lua Scripting”?
- Q 5: What means “Redis --- LRU eviction”?
- Q 6: What means “Redis --- Transactions”?
- Q 7: What means “Redis --- Different levels of on-disk persistence”?
- Q 8: What means “High availability via Redis Sentinel”?
- Q 9: What means “Automatic partitioning with Redis Cluster”?
- Q 10: What means horizontal / vertical scaling for a datastore?
- Q 11: What was the motor for the development of Redis?
- Q 12: How can we talk to the Redis (APIs)?
- Q 13: What makes a Redis unique?

Literature

- Redis Essentials (M. da Silva, H. Lopez Tavarez)
ISBN-13: 978-1784392451
- Mastering Redis (J. Nelson)
ISBN-13: 978-1783988181