

# REDIS – Letem světem

Robert Havelka

# What?

- „key-value” or „key-datastructure” storage
- **string, hash, list, set a sorted set**
- master - slave
- transactions

# What REDIS means?

- It means **RE**mote **D**ictionary **S**erver
- It is a joke on the word Redistribute (instead to use just a Relational DB redistribute your workload among Redis servers)

# String

- binary safe (JPEG, serialized objects)
- max 512M
- atomic counters
  - INCR, DECR (64bit signed int)
- random access vector
  - GETRANGE, SETRANGE
- <http://redis.io/commands/#string>

# List

- list of Strings
- ordered
- max length of a List is  $2^{32}-1$
- B(L|R)POP
- <http://redis.io/commands#list>

# Set

- set of Strings
- max members in a Set is  $2^{32}-1$
- extract elements at random
  - SPOP, SRANDMEMBER
- <http://redis.io/commands#set>

# Hash

- map between String fields and String values
- Hash can store up to  $2^{32}-1$  field-value pairs

- <http://redis.io/commands#hash>

# Sorted set

- set of Strings
- members with score
- „most advanced Redis data types“
- [http://redis.io/commands#sorted\\_set](http://redis.io/commands#sorted_set)



# Transaction

- WATCH – check and set
- UNWATCH
- MULTI – start transaction block
- EXEC – commit transaction, unWATCH all
- DISCARD – rollback

# Expiration

- ❑ EXPIRE (s)
- ❑ EXPIREAT (timestamp)
- ❑ PERSIST(cancel)

# Persistence

- RDB
  - memory snapshot
  - key changes
- AOF
  - fast (~1s)
  - redis-check-aof + diff

# Replication

- master – multiple slaves
  - slave – slave
  - graph-like structure
- cluster

# Clients

- C – hiredis
- Java – Jedis
- LUA – redis-lua
- Perl – Redis
- PHP – Predis
- **Python – redis-py**
- Ruby – redis-rb

# Sources

- <http://redis.io>
- Interactive tutorial - <http://try.redis-db.com>