Redis:~

Mahdiye Tadayon

What is Redis?

- is an open source (BSD licensed), in-memory data structure store.
- works on the concept of key-value pair.
- used as a database, cache, and message broker.
- provides data structures such as strings, hashes, lists, sets, hyperloglogs etc.

History

- Early 2009 - Salvatore Sanfilippo, started the Redis project.
- June 2009 - Redis was deployed
- March 2010 - VMWare hired Sanfilippo to work full-time on Redis
- June 2015 - development became sponsored by Redis Labs

Features

- Speed

  memory > SSD > HDD

  pipeline

  can perform about 110000 SETs per second, about 81000 GETs per second.
- Supported Languages

  C, C++, C#, Go, Java, JavaScript , PHP, Python etc
- Data structures
- Keys with a limited time-to-live
- Pub/Sub
- Transactions
- Pipeline
- Persistence

Download and Install

1. Go to https://github.com/microsoftarchive/redis/releases

2. Download file Redis-x64-3.0.504.msi

3. Install and run file: redis-cli.exe

   127.0.0.1:6379>

1. $sudo apt-get install redis-server

2. $redis-server or $sudo systemctl start redis

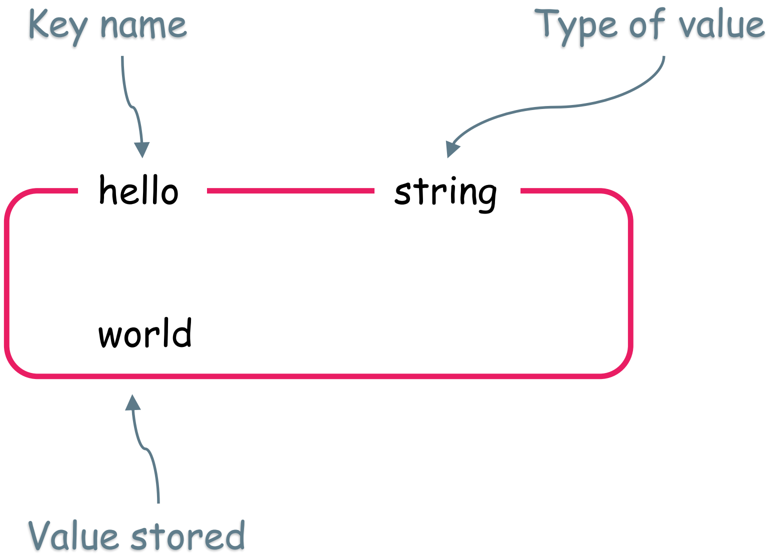
3. $redis-cli

   redis 127.0.0.1:6379>

Key value

key	value
firstName	bugs
lastName	bunny
location	earth

key	value
Key1	{1 , 2 , “foo” , “bar”}
Key2	“f1” → “v1” , “f2” → “v2”

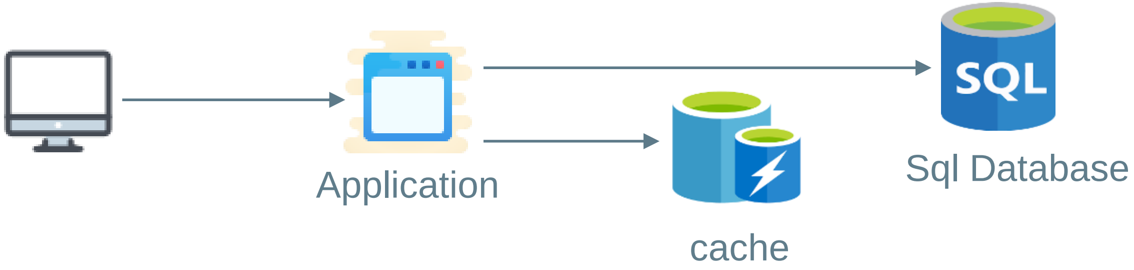


Redis vs RDBMS

Redis	RDBMS
stores everything in primary memory	stores everything in secondary memory
Read and Write operations are extremely fast	Read and Write operations are slow
Primary memory is in lesser in size and much expensive than secondary so, Redis cannot store large files.	Secondary memory is in abundant in size and cheap than primary memory so, RDBMS can easily deal with these type of files.
For store those small information which needs to be accessed, modified and inserted at a very fast rate.	For hold large data which has less frequently usage and not required to be very fast.

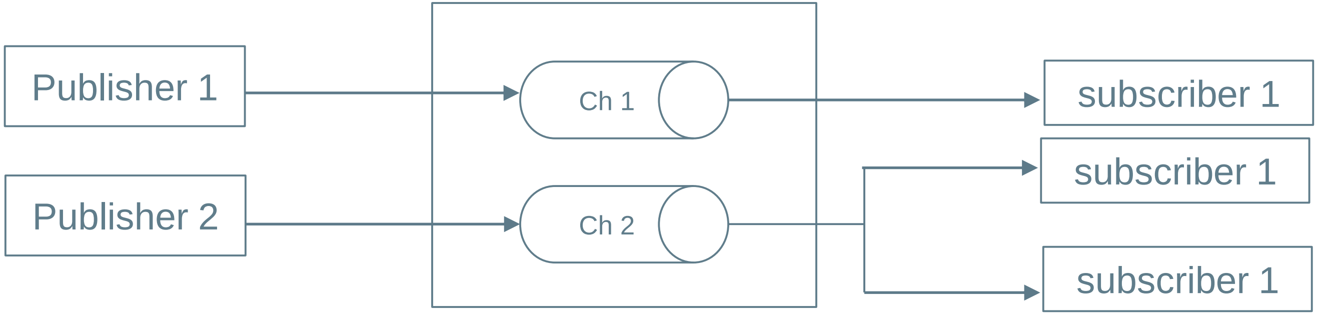
Cache

- **Caching** improves application response time by storing copies of the most frequently used data on ephemeral but very fast storage

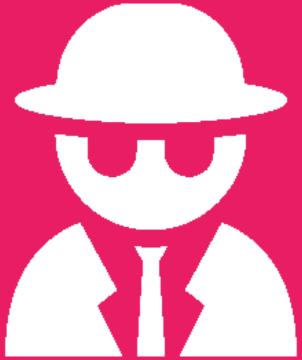


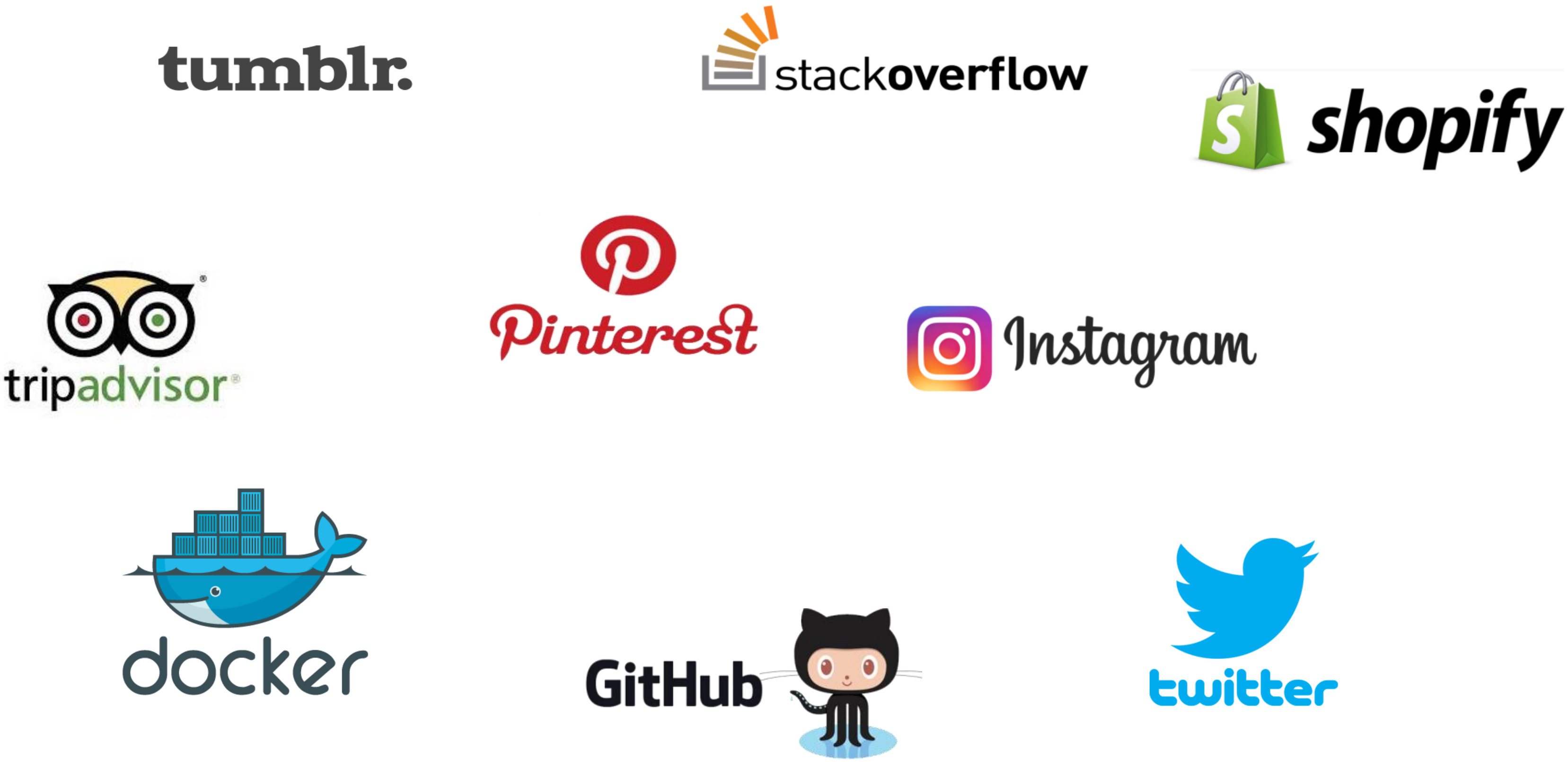
Massage broker

- **message broker** enables applications, systems, and services to communicate with each other and exchange information



Who is using redis?





source:http://techstacks.io/tech/redis

Data Types

Data Model

- Key
    - Printable ASCII
- Value
    - Primitives
        - Strings
    - Containers (of strings)
        - Hashes
        - Lists
        - Sets
        - Sorted Sets
        - hyperloglog

Keys

- SET key value
- GET key
- DEL key
- EXIST key
- RENAME key newKey
- KEYS pattern
- RANDOMKEY

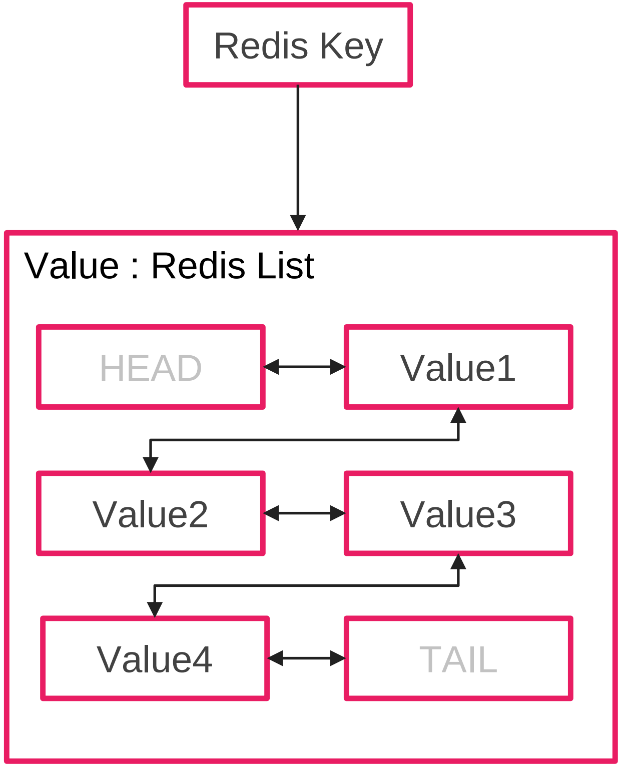
Strings

- MSET key1 value1 key2 value2 …
- MGET key1 key2 …
- GETSET key value
- APPEND key value
- STRLEN key

- INC key
- INCBY key x
- DECR key
- DECRBY key x

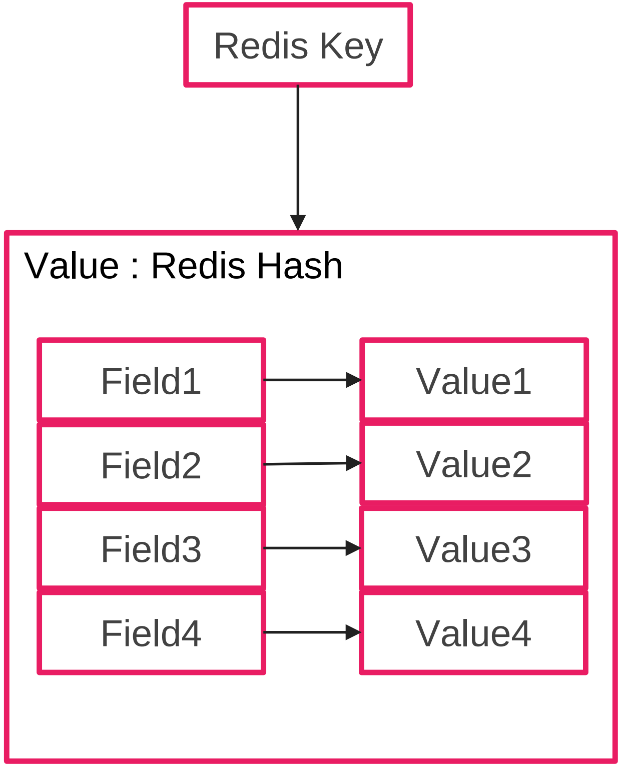
Lists

- LPUSH key value1 value2 …
- RPUSH key value1 value2 …
- LPOP
- RPOP
- LLEN key
- LINDEX key index
- LPOS key value
- BLPOP key timeout



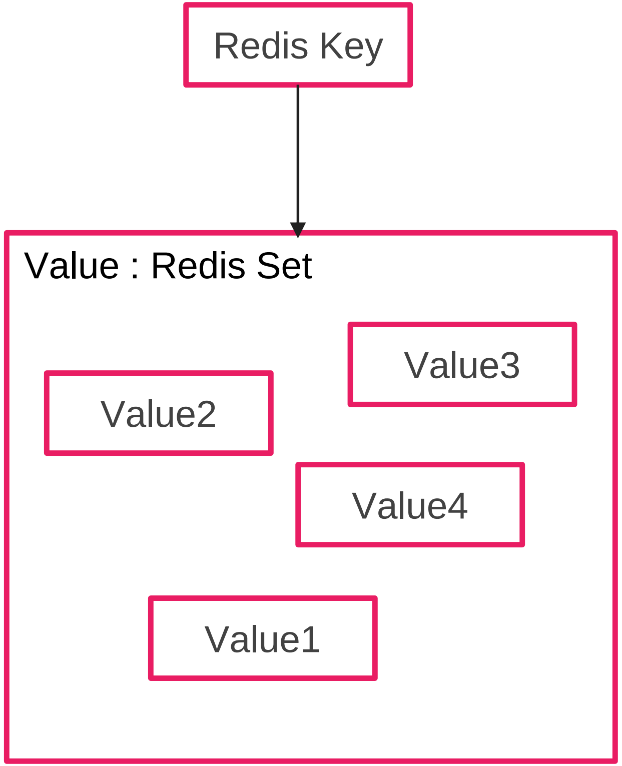
Hash

- HSET key, field1 value1, field2 value2 ….
- HGET key field
- HGETALL key
- HMGET key field1 field2 …
- HDEL key field1
- HKEYS key
- HVALS key
- HLEN key



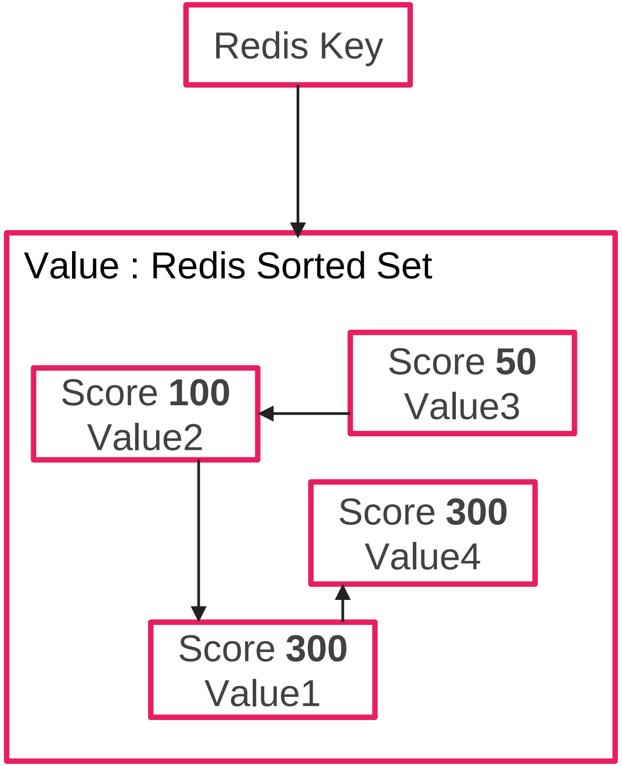
Set

- SADD key value1 value2 …
- SPOP key x
- SCARD key
- SREM key value1 value2 …
- SMEMBERS key
- SDIFF key1 key2
- SINTER key1 key2
- SMOVE key1 key2 member



Sorted Set

- SADD key value1 value2 …
- SPOP key x
- SCARD key
- SREM key value1 value2 …
- SMEMBERS key
- SDIFF key1 key2
- SINTER key1 key2
- SMOVE key1 key2 member



HyperLogLog

- is a probabilistic data structure used to count unique values
- a measure of a set's size, meaning the number of elements in the set

- PFADD key value value …
- PFCOUNT key
- PFMERG keyNew key1 key2

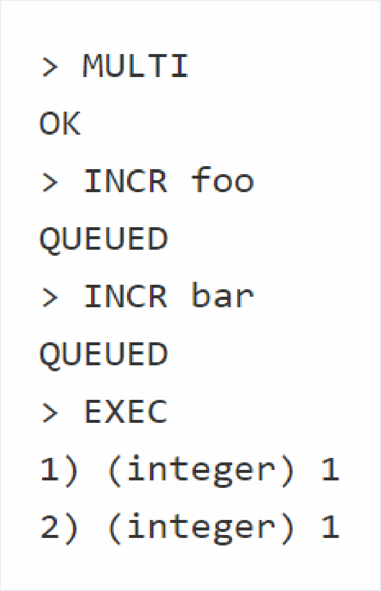
Transaction

Transaction

It is possible to group commands together so that they are executed as a single transaction

- Isolate
- atomic

- MULTI
- EXEC
- DISCARD
- WATCH key
- UNWATCH key



```
> MULTI
OK
> INCR foo
QUEUED
> INCR bar
QUEUED
> EXEC
1) (integer) 1
2) (integer) 1
```

Databases

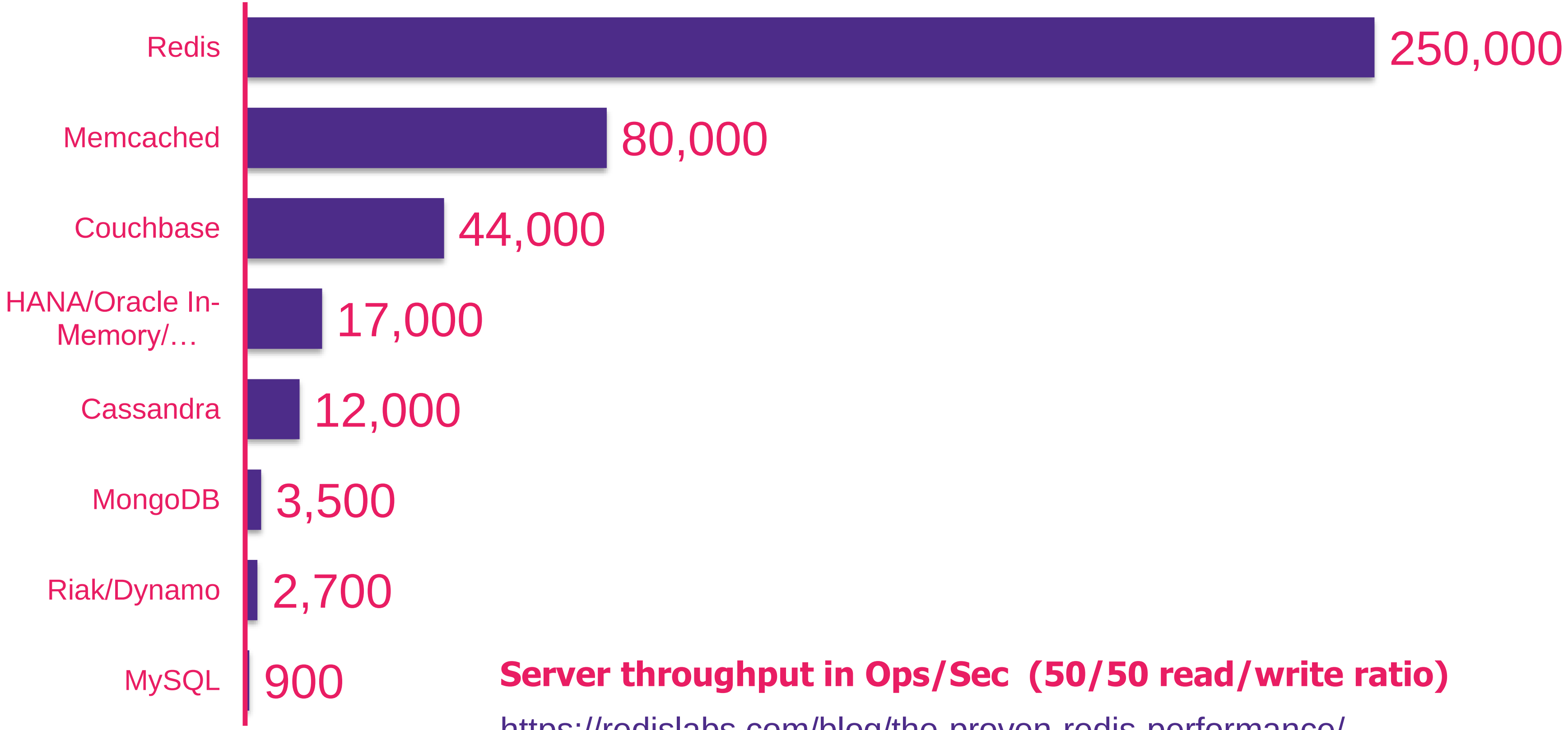
- every **Redis** instance supports 16 **databases**
- The default **database** is 0 but you can change that to any number from 0-15
- and you can configure **Redis** to support more **databases**

- CONFIG GET databases
- SELECT index(0-15)
- SWAPDB index1 index2
- DBSIZE
- FLUSHDB
- FLUSHALL

Why Redis

?

Redis Performance



Server throughput in Ops/Sec (50/50 read/write ratio)

https://redislabs.com/blog/the-proven-redis-performance/

Resources

1. https://redis.io/
2. https://www.tutorialspoint.com/redis/index.html
3. redis in action by josiah carlson
4. Cristian Andrei BARON. (2015). NoSQL Key-Value DBs Riak and Redis

Thanks