```
25627:C 22 Mar 2023 15:37:17.704 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo
25627:C 22 Mar 2023 15:37:17.704 # Redis version=7.0.9, bits=64, commit=00000000, modified=0, pid=25627, just started
25627:C 22 Mar 2023 15:37:17.704 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
25627:M 22 Mar 2023 15:37:17.704 * Increased maximum number of open files to 10032 (it was originally set to 256).
25627:M 22 Mar 2023 15:37:17.704 * monotonic clock: POSIX clock_gettime
                _._
           _.-``__ ''-._
      _.-``    `.  `_.  ''-._           Redis 7.0.9 (00000000/0) 64 bit
  .-`` .-```.  ```\/    _.,_ ''-._
 (    '      ,       .-`  | `,    )     Running in standalone mode
 |`-._`-...-` __...-.``-._|'` _.-'|     Port: 6379
 |    `-._   `._    /     _.-'    |     PID: 25627
  `-._    `-._  `-./  _.-'    _.-'
 |`-._`-._    `-.__.-'    _.-'_.-'|
 |    `-._`-._        _.-'_.-'    |           https://redis.io
  `-._    `-._`-.__.-'_.-'    _.-'
 |`-._`-._    `-.__.-'    _.-'_.-'|
 |    `-._`-._        _.-'_.-'    |
  `-._    `-._`-.__.-'_.-'    _.-'
      `-._    `-.__.-'    _.-'
          `-._        _.-'
              `-.__.-'

25627:M 22 Mar 2023 15:37:17.706 # WARNING: The TCP backlog setting of 511 cannot be enforced because kern.ipc.somaxconn is set to the lower value of 128.
25627:M 22 Mar 2023 15:37:17.706 # Server initialized
25627:M 22 Mar 2023 15:37:17.706 * Loading RDB produced by version 7.0.9
25627:M 22 Mar 2023 15:37:17.706 * RDB age 445 seconds
25627:M 22 Mar 2023 15:37:17.706 * RDB memory usage when created 1.04 Mb
25627:M 22 Mar 2023 15:37:17.706 * Done loading RDB, keys loaded: 0, keys expired: 0.
25627:M 22 Mar 2023 15:37:17.706 * DB loaded from disk: 0.000 seconds
25627:M 22 Mar 2023 15:37:17.706 * Ready to accept connections
```

Memory First

An Introduction to Redis

Guy Royse
Developer Advocate

redis

@guyroyse

github.com/guyroyse

guy.dev

What is Redis?

Cache

Key-value store

Message broker

It's a memory-first, NoSQL database.

Memory First

Everything is in memory so it's wicked fast.

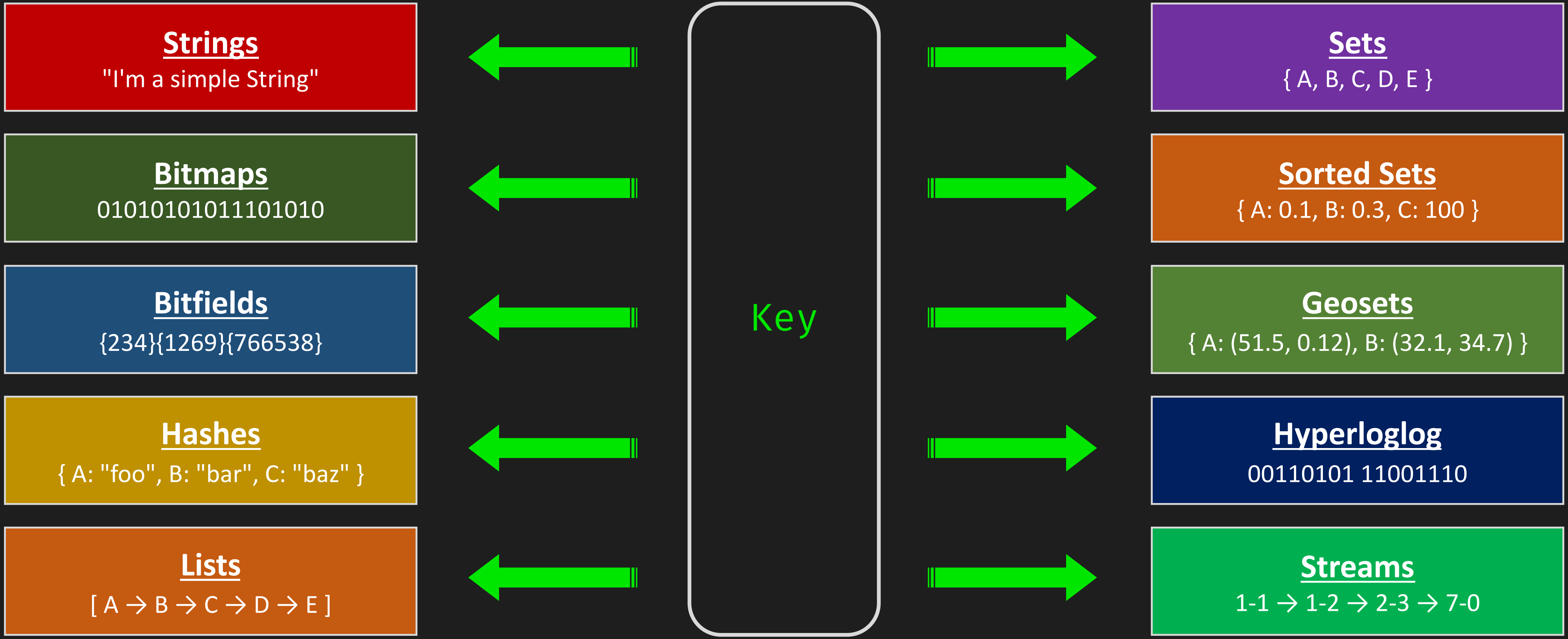
NoSQL

Serves up data structures that we know and love.

Database

Persistable
Clusterable
Replicatable

A Giant Hash Table



Redis is Extensible

Modules

New data structures

New commands

JSON

Search

Bloom

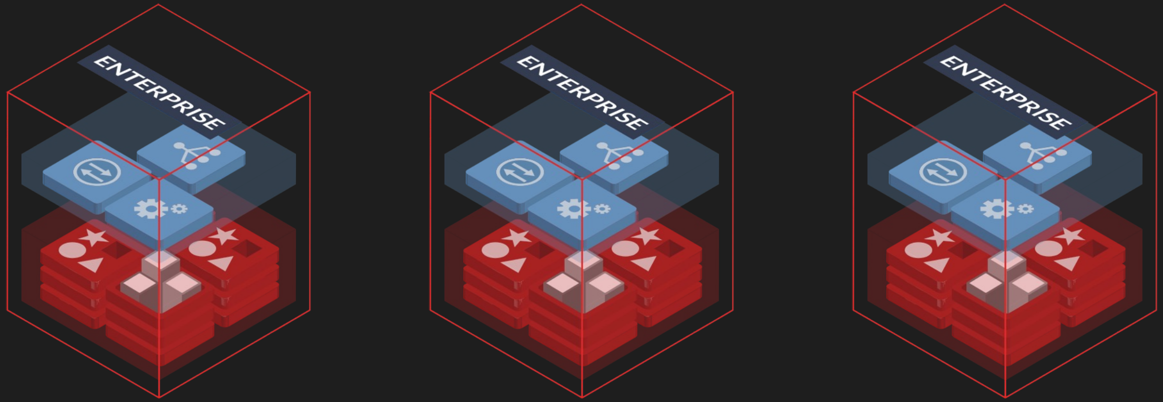
Time Series

redis stack

Redis JSON Search Bloom Time Series Redis Insight

```
docker run -d --name redis-stack \
           -p 6379:6379 -p 8001:8001 \
           redis/redis-stack:latest
```

redisenterprise
CLOUD



redis.com/try-free

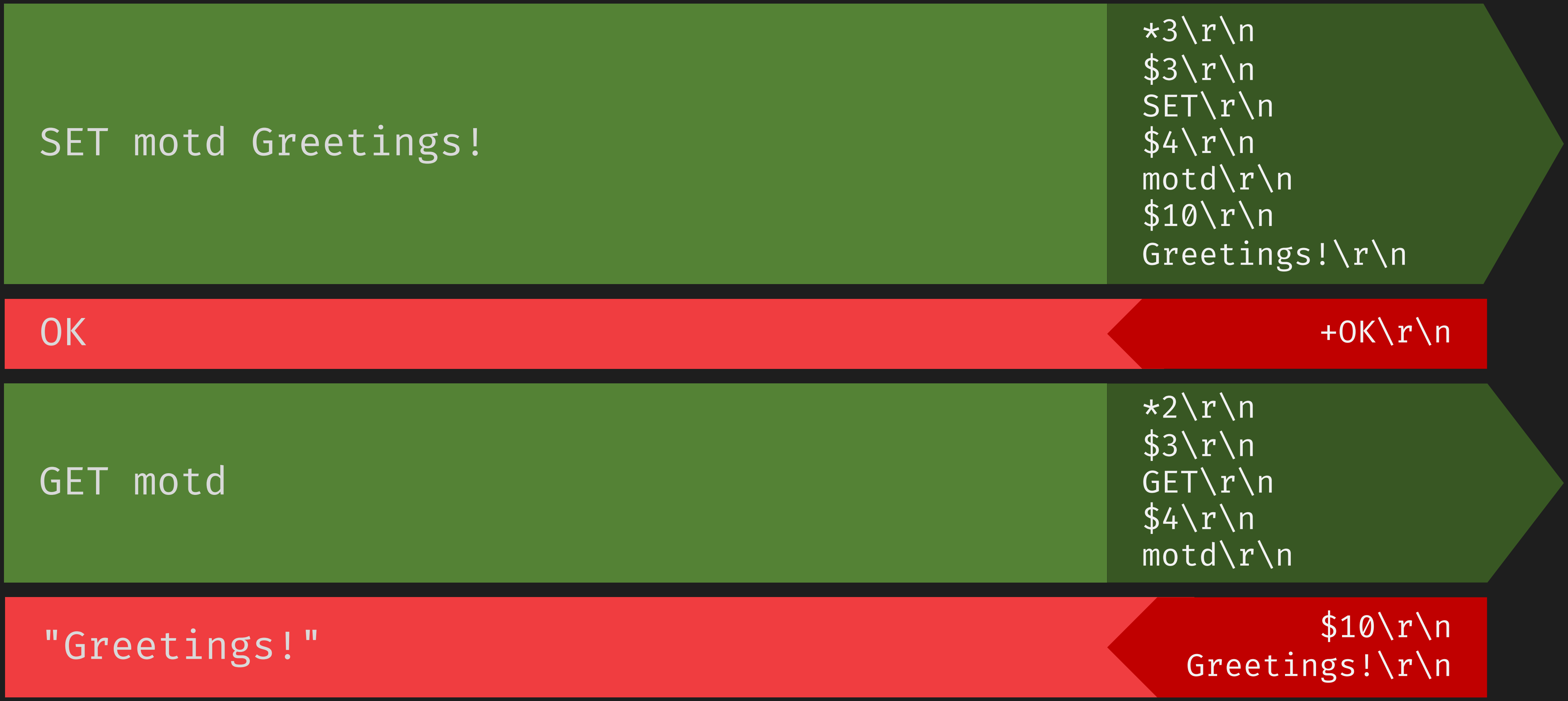
Talking to Redis

```
PING              PONG

TIME              1) "1679598546"
                  2) "805638"

FLUSHALL          OK
```

RESP: The Wire Protocol



```
SET motd Greetings!
```

```
*3\r\n
$3\r\n
SET\r\n
$4\r\n
motd\r\n
$10\r\n
Greetings!\r\n
```

```
OK
```

```
+OK\r\n
```

```
GET motd
```

```
*2\r\n
$3\r\n
GET\r\n
$4\r\n
motd\r\n
```

```
"Greetings!"
```

```
$10\r\n
Greetings!\r\n
```

Demo

Strings

```
> SET foo alfa
> GET foo
```

```
$ redis-cli -x SET foo < data.bin
```

```
> SET foo 42
> INCR foo
> INCRBY foo 23
```

Stored as either raw or embedded strings

Stored as integers

Demo

redis.io/commands

Persistent vs. Volatile Keys

Stick around forever.
Keys are persistent by default.

Have a time-to-live.
Are removed on or after that TTL.

Persistent vs. Volatile Keys

```
> SET foo alfa
```

```
> EXPIRE foo 3600
> EXPIREAT foo 2147483647
```

```
> TTL foo
> EXPIRETIME foo
```

```
> PERSIST foo
```

Demo

Lists, Sets, and Hashes

```
> LPUSH foo alfa
> RPOP foo
```

```
> SADD foo alfa bravo charlie
> SISMEMBER foo alfa
> SMEMBERS foo
> SUNION foo bar
```

```
> HSET foo alfa a
> HGET foo alfa
> HGETALL foo
```

Demo

Storing JSON

""

```
> SET foo '{ "bar": "alfa" }'
> GET foo

...parse string...

> SET foo '{ "bar": 42 }'
```

{}

```
> JSON.SET foo $ '{ "bar": "alfa"}'
> JSON.GET foo $.bar
> JSON.SET foo $.bar 42
```

Demo

Persistence

RDB

redis.conf

```
save 3600 1
save 300 100
save 60 10000
```

```
> BGSAVE
```

AOF

redis.conf

```
appendonly yes
appendfsync always | everysec | no
auto-aof-rewrite-percentage 100
auto-aof-rewrite-min-size 64mb
```

```
> BGREWRITEAOF
```

Questions?

Code & Slides

https://github.com/guyroyse/memory-first

Check Out Our Stuff

Redis

https://redis.io

Redis Insight

https://redis.com/redisinsight

Redis Discord Server

https://discord.gg/redis

Guy Royse
Developer Advocate

redis

@guyroyse

github.com/guyroyse

guy.dev

redis