

**SEE HOW FAST FEELS**



# Memory first: An introduction to Redis

Guy Royse



● Developer Advocate

Guy Royse

@guyroyse X

github.com/guyroyse 

guy.dev 

# What's a 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.

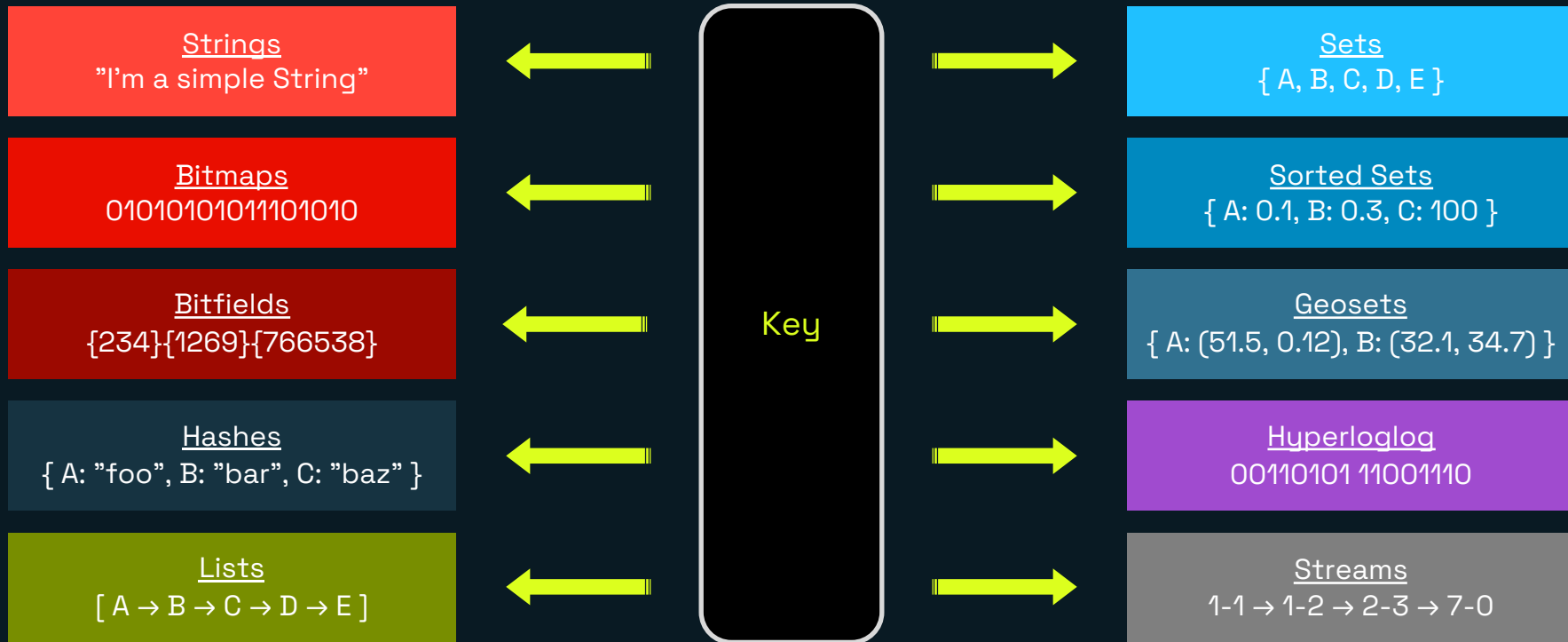
## NoSQL

Serves up data  
structures that we  
know and love.

## Database

Persistable  
Clusterable  
Replicable

# A giant hash table



# Redis is extensible



Modules

New commands

New data structures



JSON



Search



Bloom



Time  
Series



# redis stack



Redis



JSON



Search



Bloom



Time  
Series



Redis  
Insight

[redis.io/try-free](https://redis.io/try-free)



# Talking to Redis



PING

TIME

FLUSHALL

PONG

1) "1679598546"

2) "805638"

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



1010  
1010



```
> 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 and slides

[github.com/guyroyse/memory-first](https://github.com/guyroyse/memory-first)





X @guyroyse

 github.com/guyroyse

 guy.dev

Thank you.

**SEE HOW FAST FEELS**