

redis: NoSQL Simplified

Who I Am

- Work: Hypothetical Labs
- Twitter: @kevsmith
- Blog: <http://weblog.hypotheticallabs.com>

NoSQL

It's So
Amazingly
Cool!

It's Easy!

Document
Map/Reduce
Column Store
Super Column
JSON

Document
Map/Reduce
Column Store
Super Column
JSON

ERLANG

Document
Map/Reduce
Column Store
Super Column
JSON

My brain hurts!



Redis is...

Redis is...

- A key/value store

Redis is...

- A key/value store
- An in-memory database

```
top - 17:21:27 up 1:20, 4 users, load average: 1.90, 2.23, 1.91
Tasks: 79 total, 1 running, 78 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.0%us, 0.0%sy, 0.0%ni, 100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 71680140k total, 66641604k used, 5038536k free, 400628k buffers
Swap: 0k total, 0k used, 0k free, 816344k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6609	root	15	0	60.7g	60g	416	S	0	88.7	18:41.56	redis-server /etc/redis.conf
6688	root	16	0	57456	20m	1872	S	0	0.0	1:03.70	irb
6709	root	15	0	58084	18m	1880	S	0	0.0	0:36.62	irb

Redis is...

Redis is...

- A key/value store

Redis is...

- A key/value store
- An in-memory database

Redis is...

- A key/value store
- An in-memory database
- Fast!

How fast is Redis?

Redis Performance

Operation	Performance
GET	~19,600/sec
SET	~13,900/sec

Redis has...

Redis has...

A simple protocol

Redis has...

A simple protocol

An Easy to understand data model

Redis has...

A simple protocol

An Easy to understand data model

Insanely fast performance

Redis doesn't have...

Redis doesn't have...

A sophisticated persistence engine

Redis doesn't have...

A sophisticated persistence engine
Fancy distributed capabilities

Redis doesn't have...

A sophisticated persistence engine

Fancy distributed capabilities

A 133t query engine

Redis is like
memcached.

But better.

Datatypes

Datatypes

- String-y things

Datatypes

- String-y things
- Lists

Datatypes

- String-y things
- Lists
- Sets

Datatypes

- String-y things
- Lists
- Sets
- Skip lists (coming soon!)

String Ops

String Ops

- GET/SET/SETNX

String Ops

- GET/SET/SETNX
- GETSET

String Ops

- GET/SET/SETNX
- GETSET
- MGET

String Ops

- GET/SET/SETNX
- GETSET
- MGET
- INCR/DECR

String Ops

- GET/SET/SETNX
- GETSET
- MGET
- INCR/DECR
- DEL

String Ops

- GET/SET/SETNX
- GETSET
- MGET
- INCR/DECR
- DEL
-and several more

Things You Can Build With Strings

Things You Can Build With Strings

- General-purpose cache

Things You Can Build With Strings

- General-purpose cache
- Primitive message queue

Things You Can Build With Strings

- General-purpose cache
- Primitive message queue
- Statistics Tracker

List Ops

List Ops

- PUSH/POP

List Ops

- PUSH/POP
- LEN

List Ops

- PUSH/POP
- LEN
- RANGE

List Ops

- PUSH/POP
- LEN
- RANGE
- TRIM

List Ops

- PUSH/POP
- LEN
- RANGE
- TRIM
-and a whole bunch more

Lists are efficient

Things You Can Build With Lists

Things You Can Build With Lists

- Even better caches

Things You Can Build With Lists

- Even better caches
- Smarter queues

Things You Can Build With Lists

- Even better caches
- Smarter queues
- Job streaming

Set Ops

Set Ops

- **ADD/REM/SPOP/SMEMBERS**

Set Ops

- **ADD/REM/SPOP/SMEMBERS**
- **Intersection**

Set Ops

- **ADD/REM/SPOP/SMEMBERS**
- Intersection
- Union

Set Ops

- **ADD/REM/SPOP/SMEMBERS**
- **Intersection**
- **Union**
- **Diff**

Redis does persistence

Persistence

Persistence

- Performs background save every X changes in Y seconds

Persistence

- Performs background save every X changes in Y seconds
- **SAVE/BGSAVE** for explicit saving

Persistence

- Performs background save every X changes in Y seconds
- `SAVE/BGSAVE` for explicit saving
- Dumping to a file descriptor

Replication

Replication

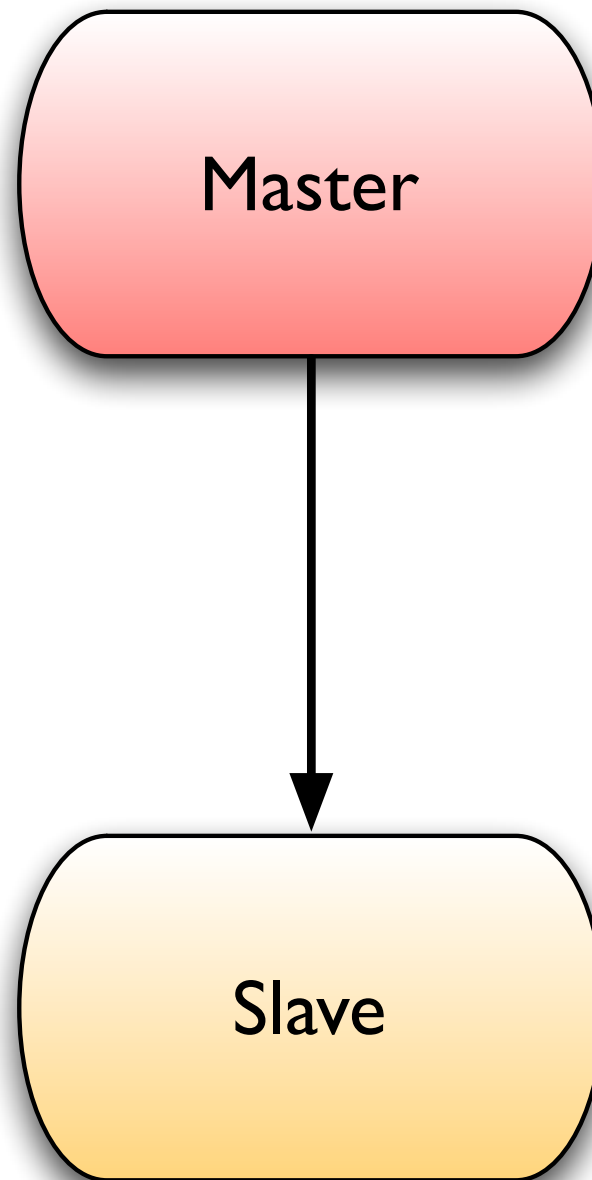
- Dumping to another file descriptor

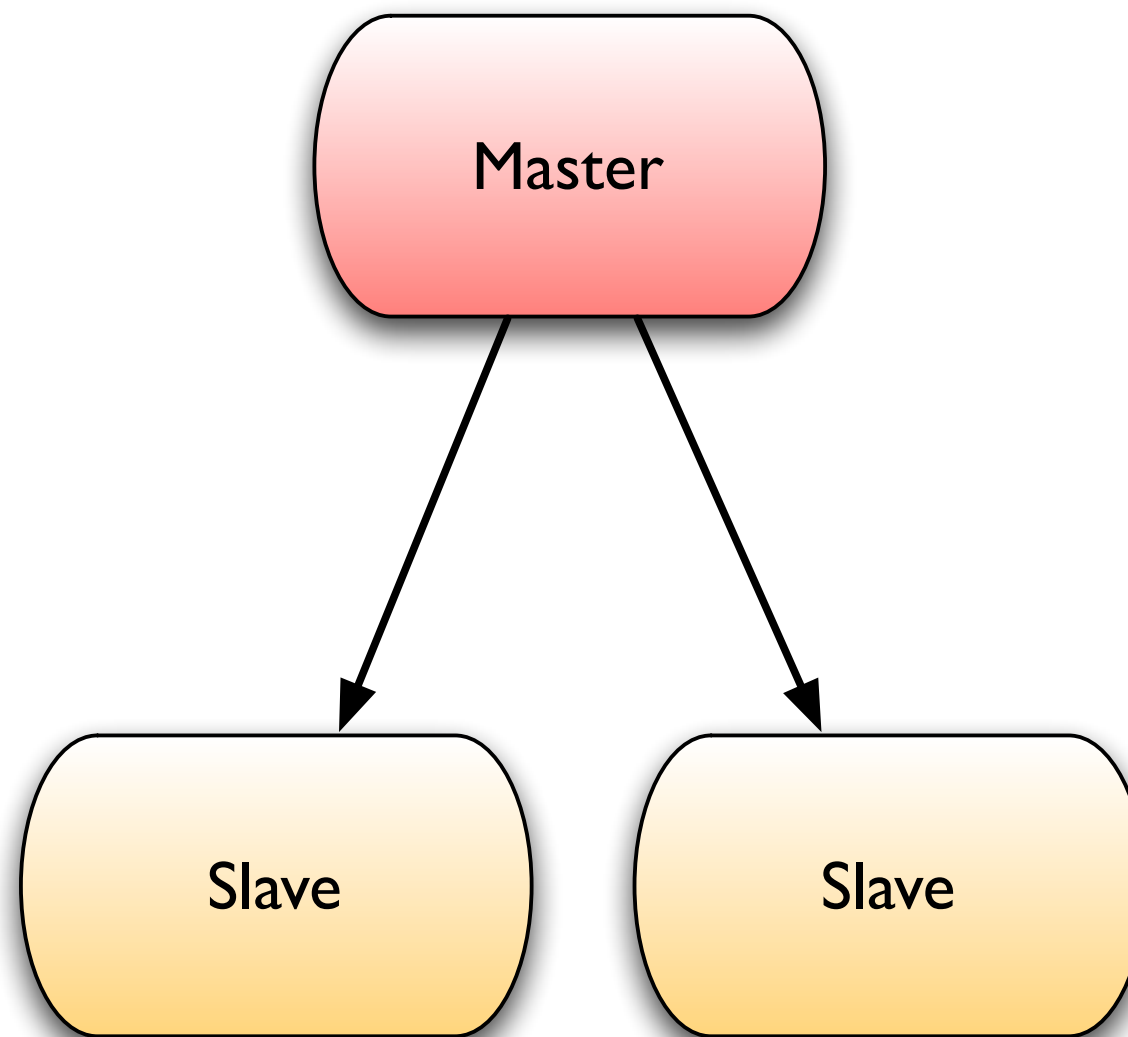
Replication

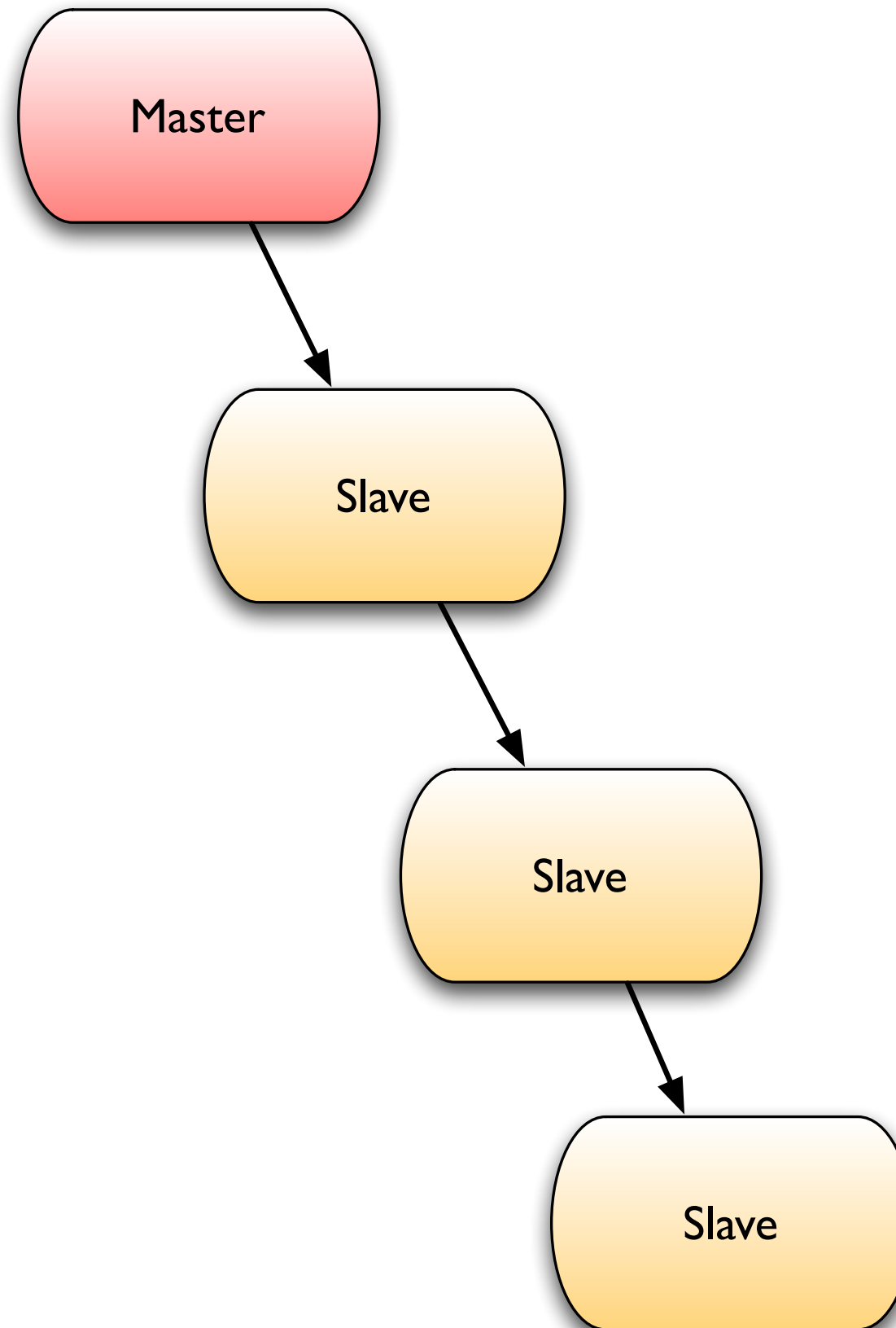
- Dumping to another file descriptor
- Configured at startup

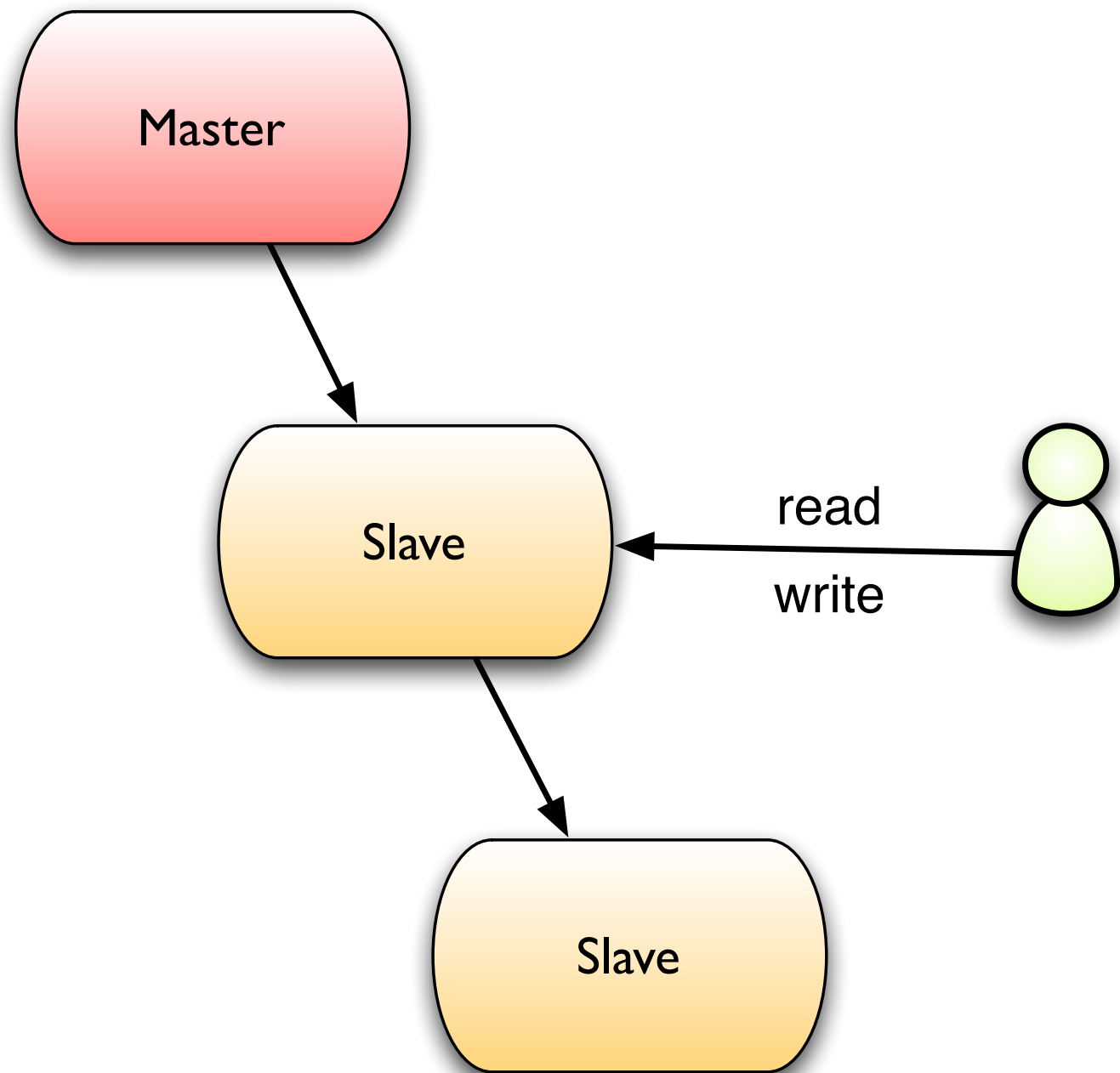
Replication

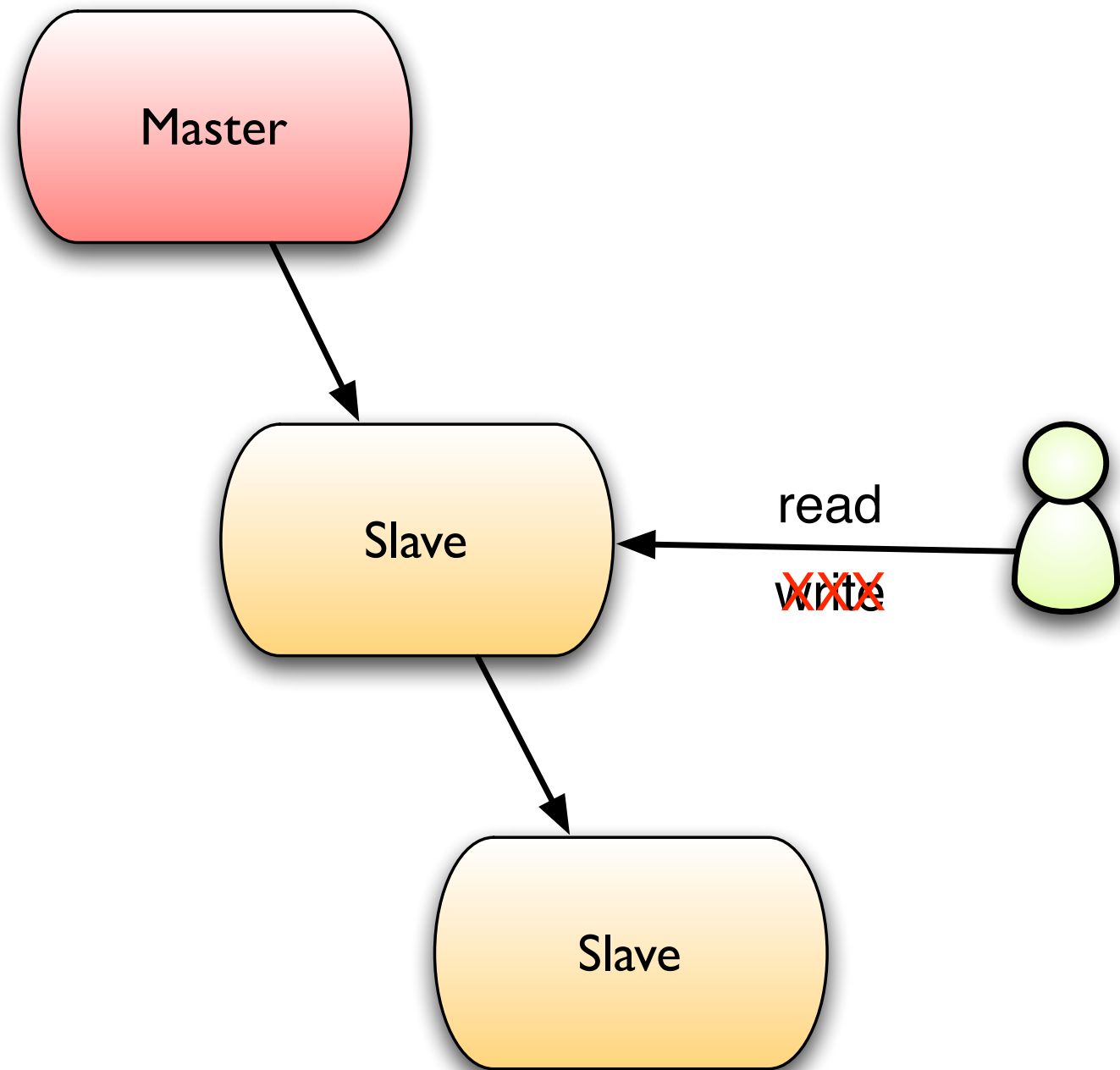
- Dumping to another file descriptor
- Configured at startup
- Toggled at runtime via SLAVEOF











When To Use Redis

When To Use Redis

- Caches

When To Use Redis

- Caches
- Statistics

When To Use Redis

- Caches
- Statistics
- Work Queues

When To Use Redis

- Caches
- Statistics
- Work Queues
- “Social” metadata - click tracking, backlinking, etc

Thank You

redis:

<http://github.com/antirez/redis/>

redis-rb:

<http://github.com/ezmobius/redis-rb/>