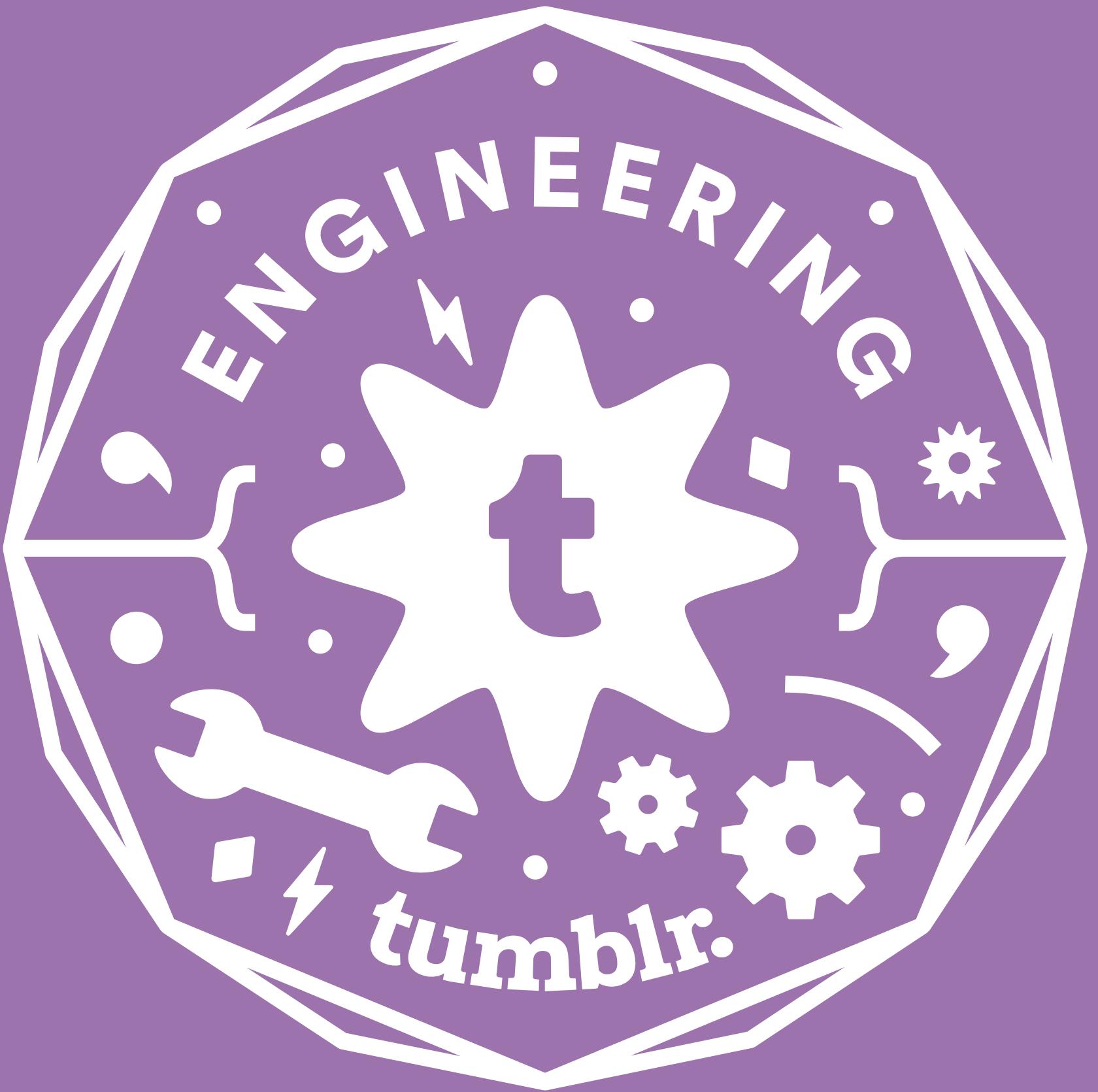


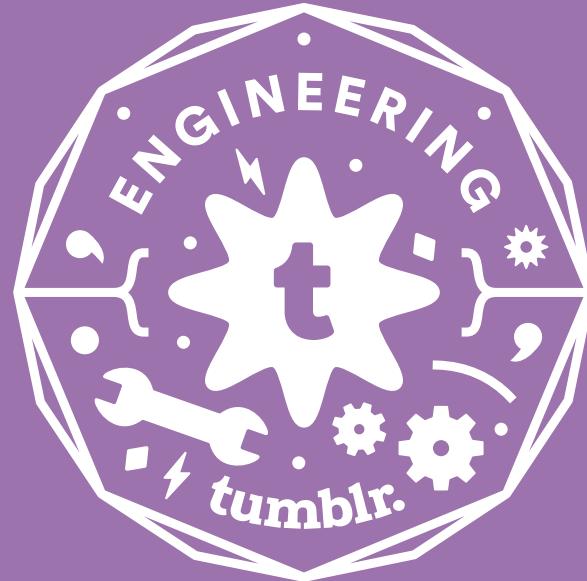
tumblr.



beta.tumblr.com







What are we working with?

- 150B+ posts
- 370M+ blogs
- 30M+ posts a day
- hundreds of database machines

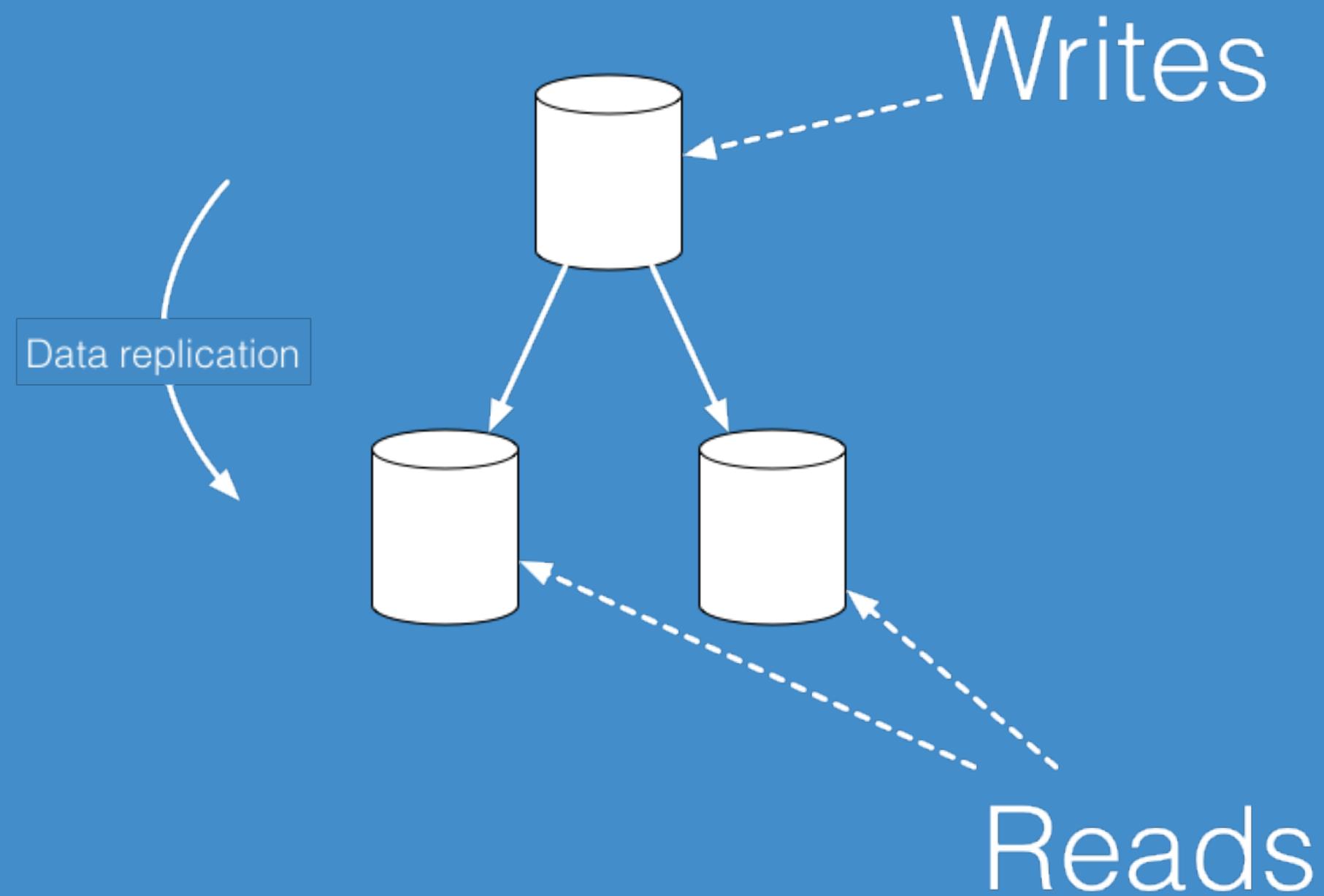
Jetpants



Single Server



Read-Write Splitting



A LOOK INTO

Ranged Sharding

Ranged Sharding

- Splits data across many shards
- Based on a primary ID
- Reduces blast radius for any one cluster
- Growable



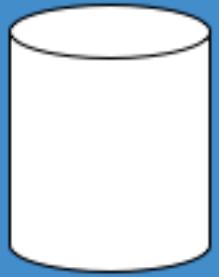
1-10



11-20



21-30



A new
post:

14

1-10



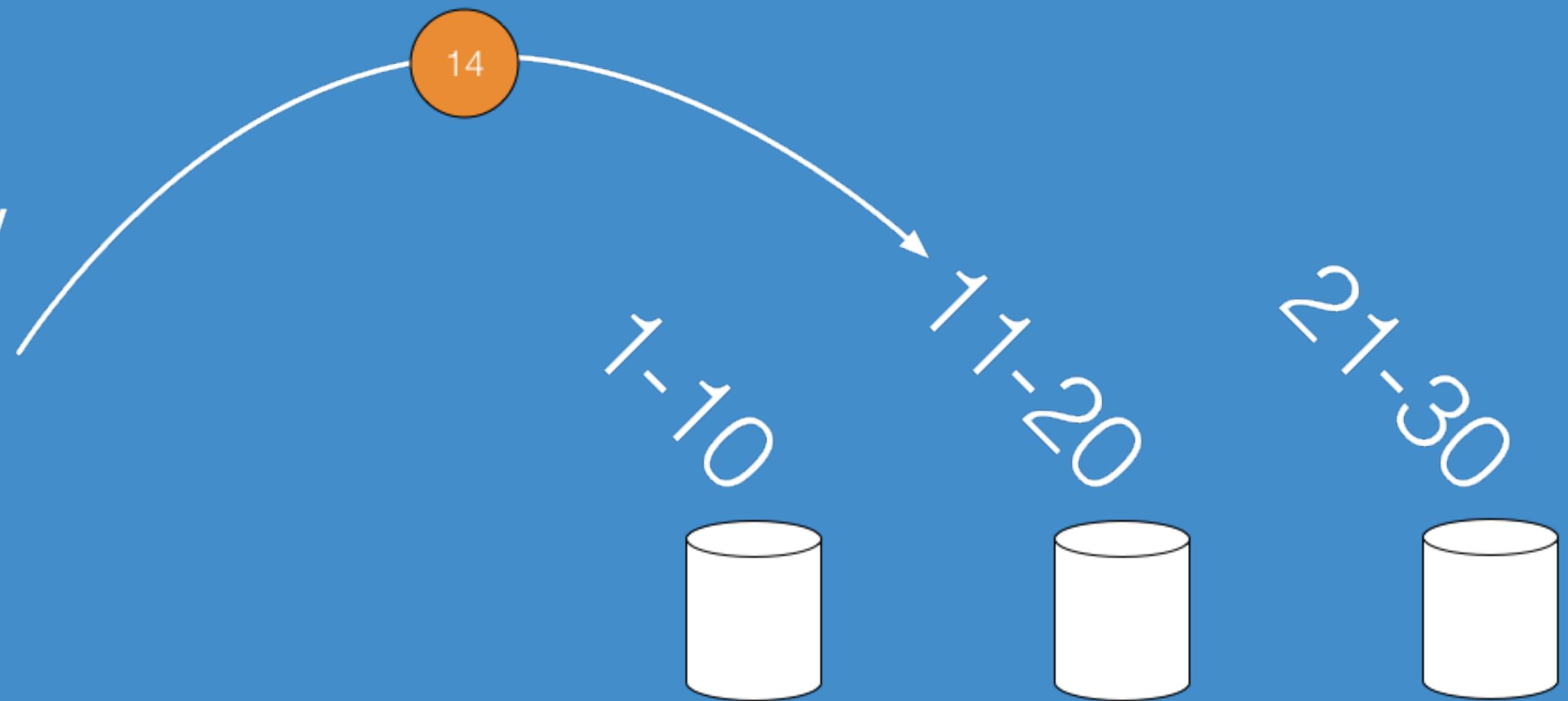
11-20



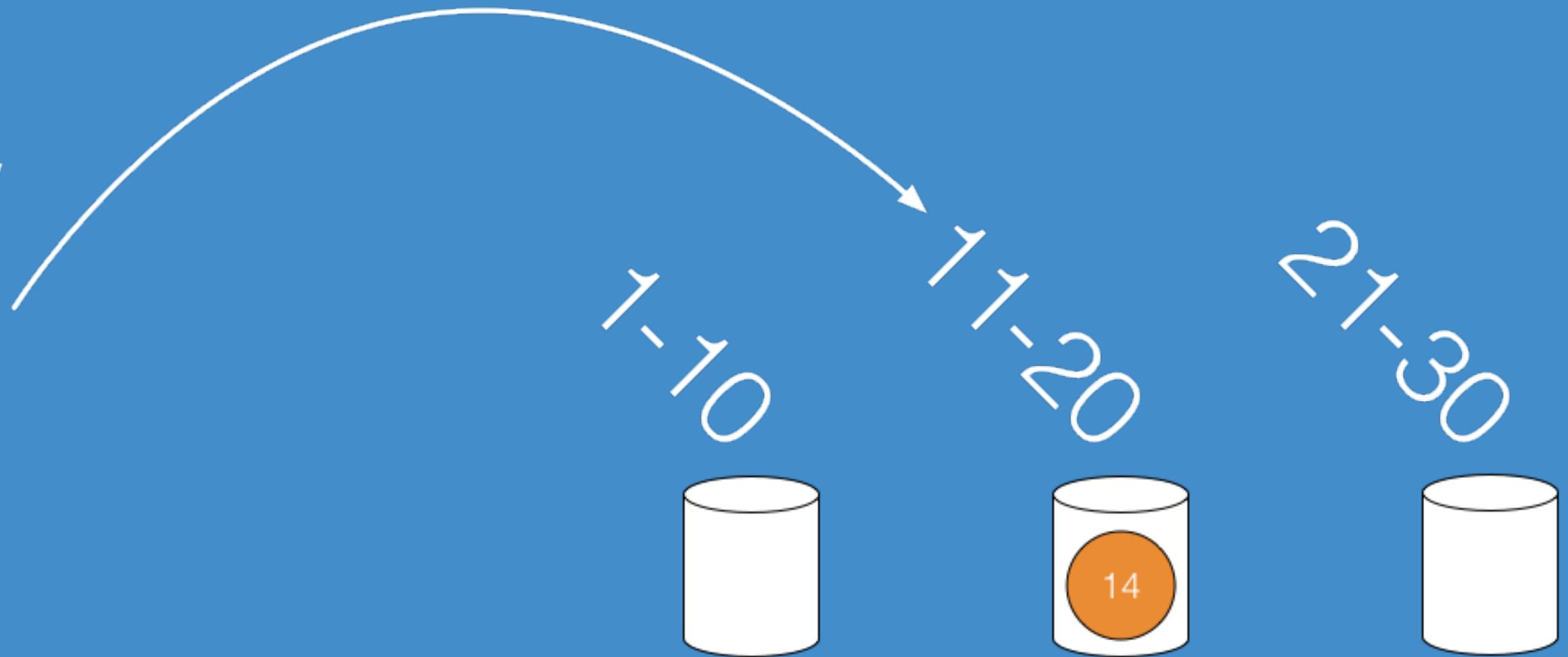
21-30

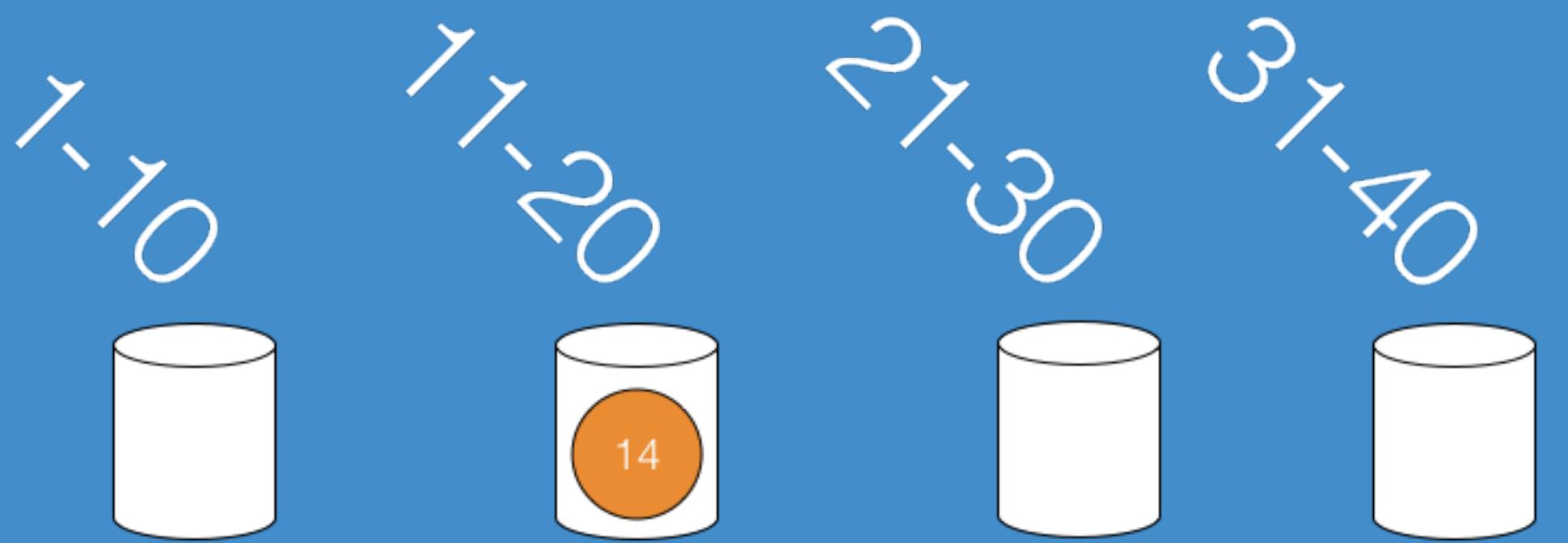


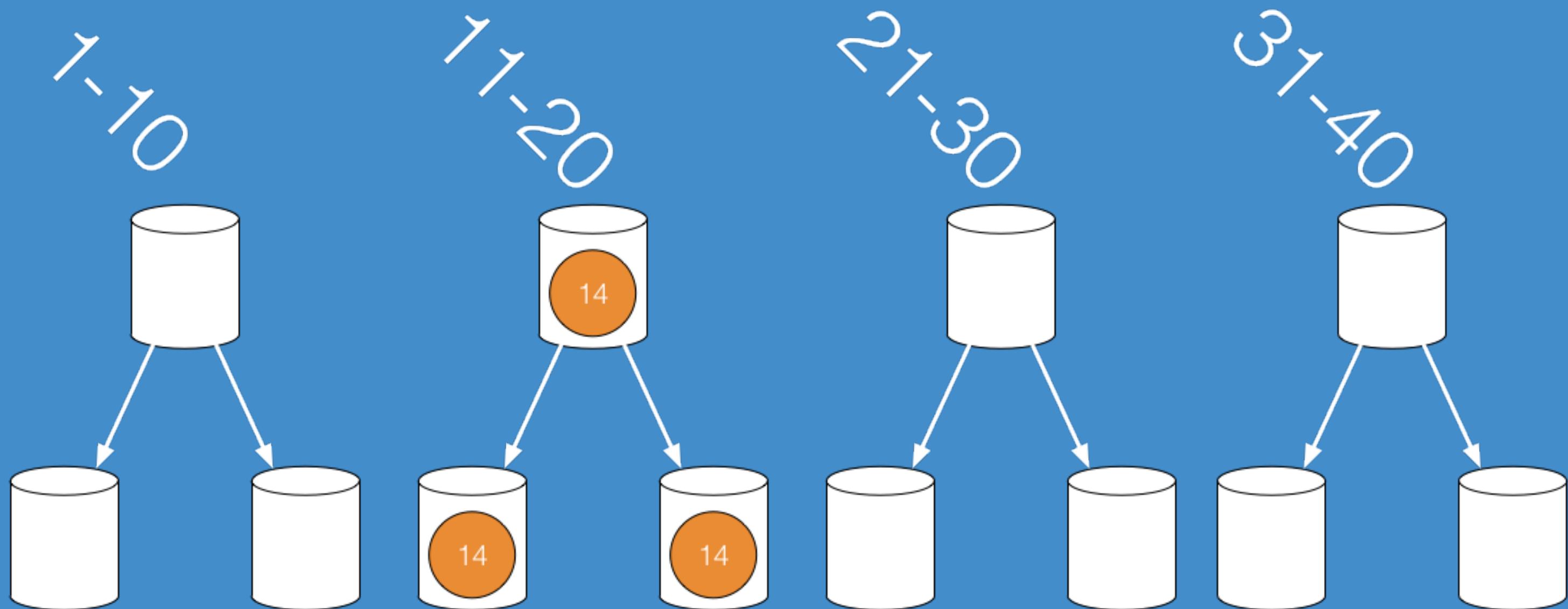
A new
post:



A new
post:

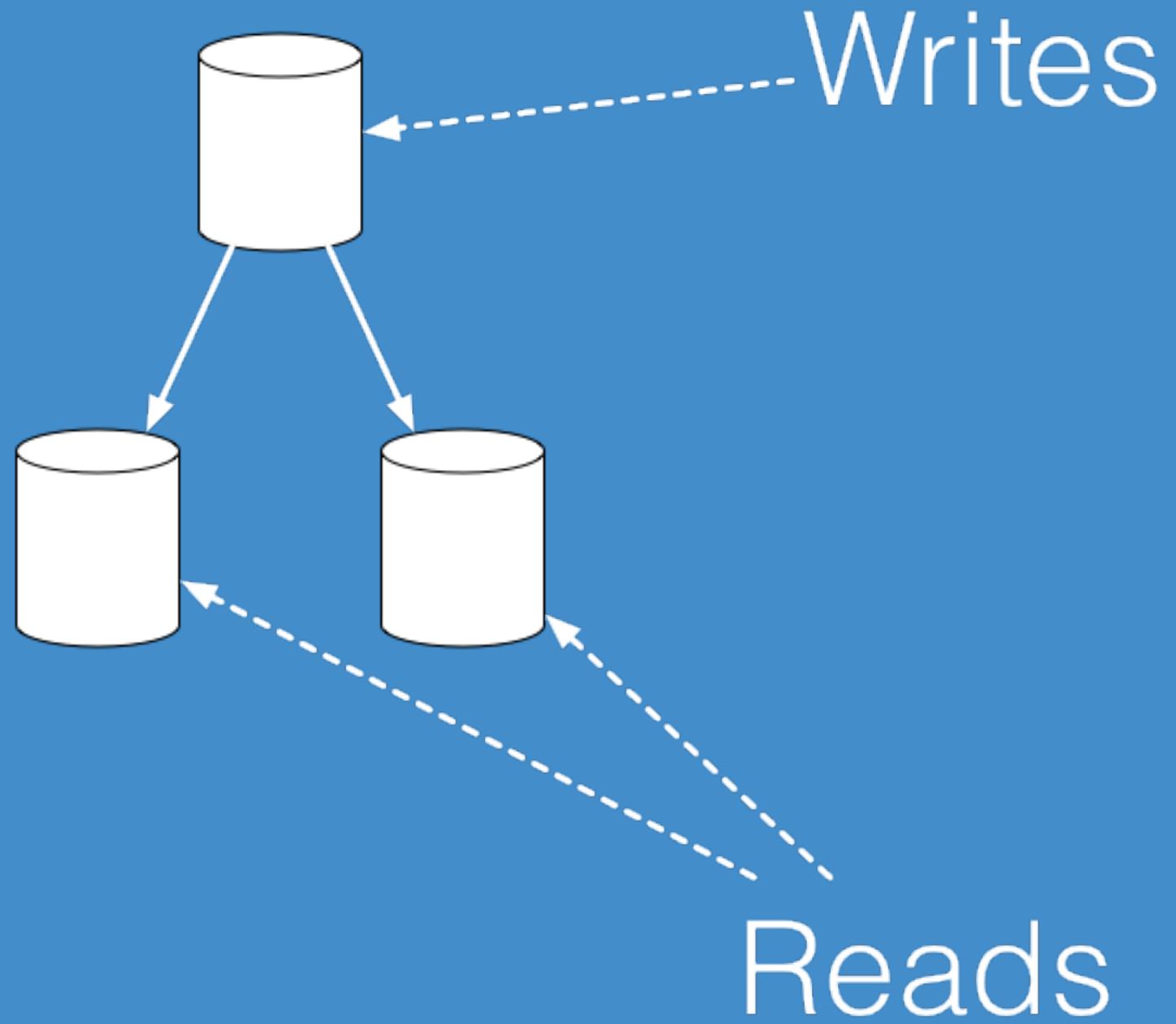


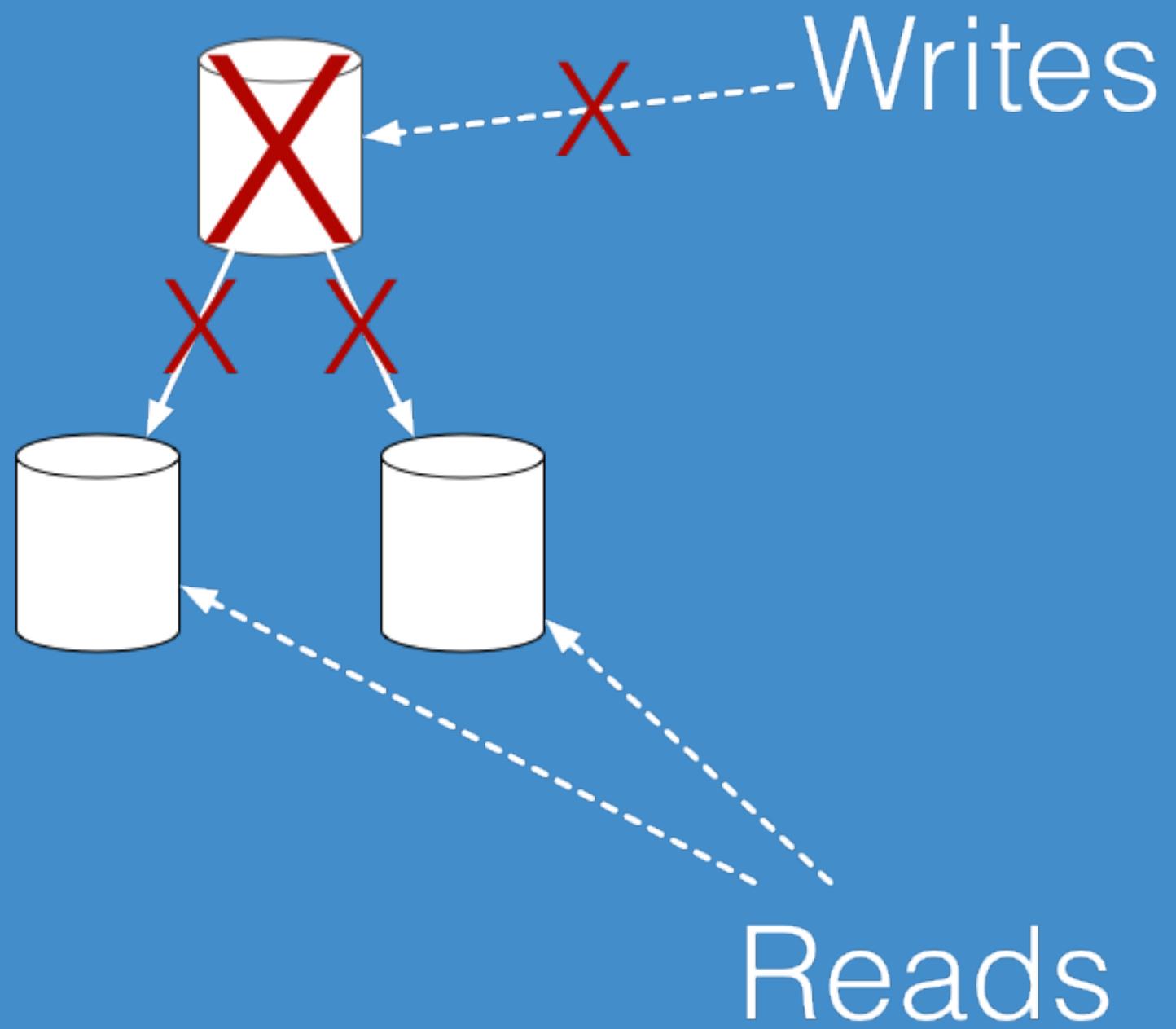


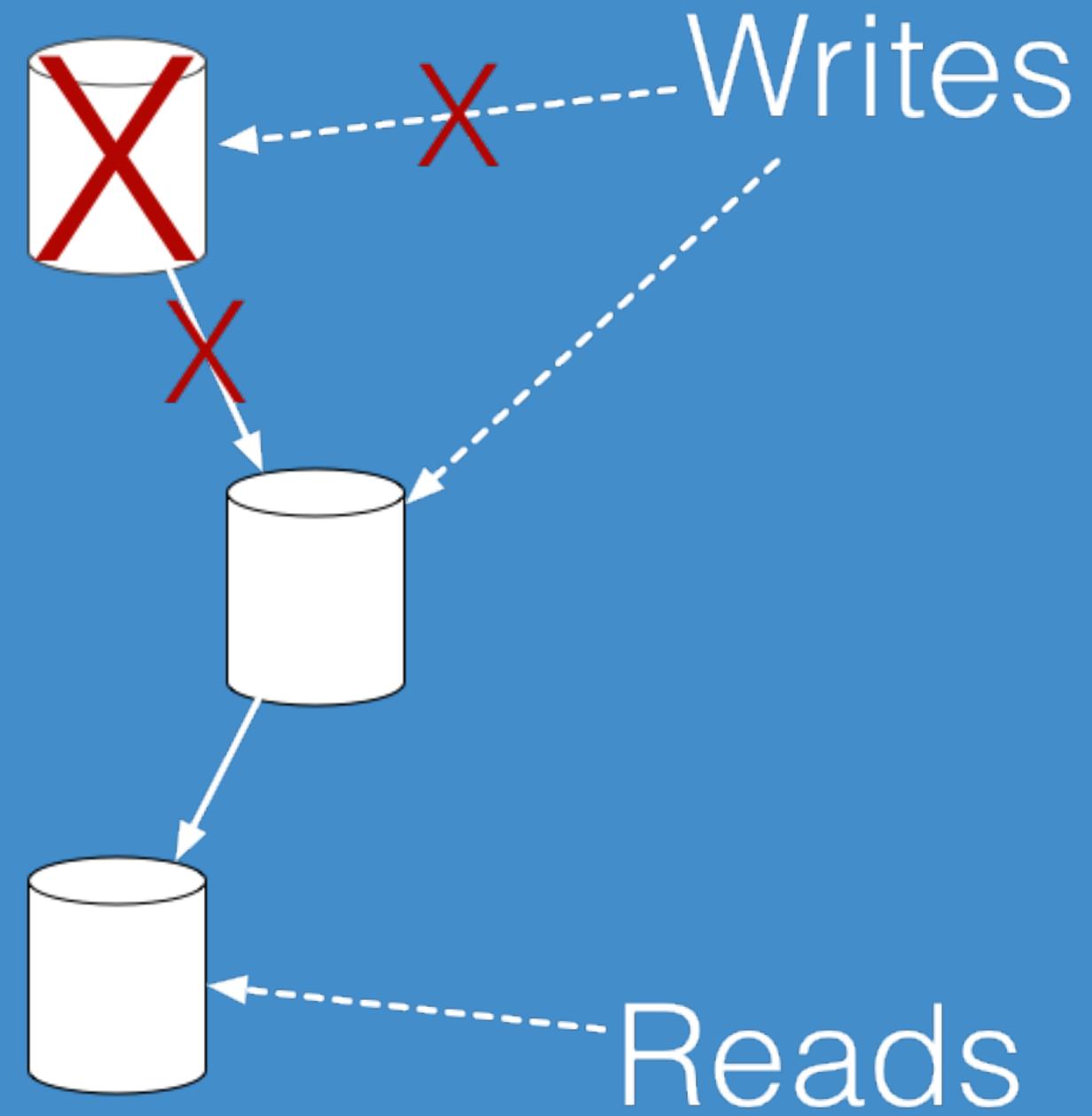


A LOOK INTO

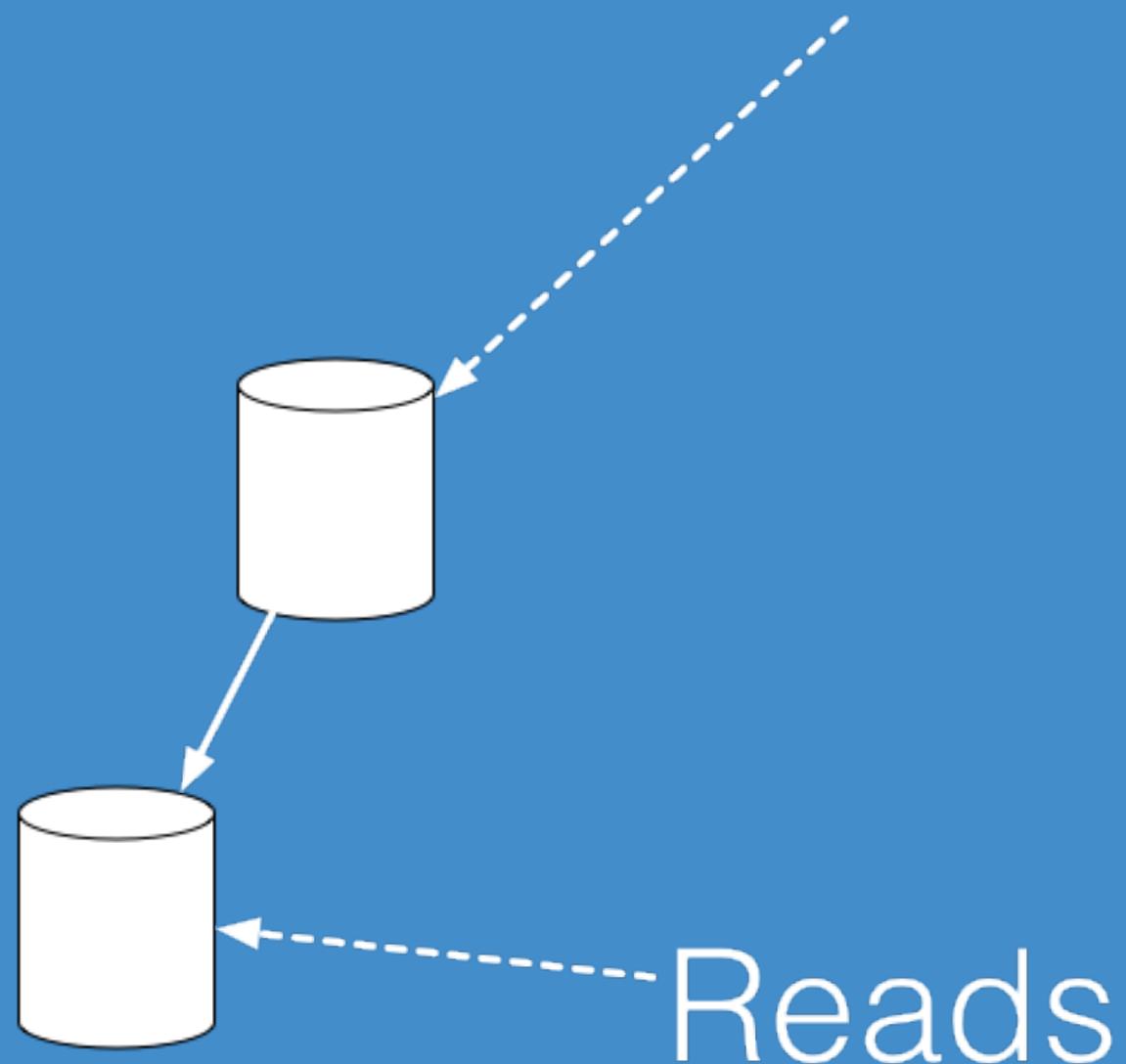
Master Promotions





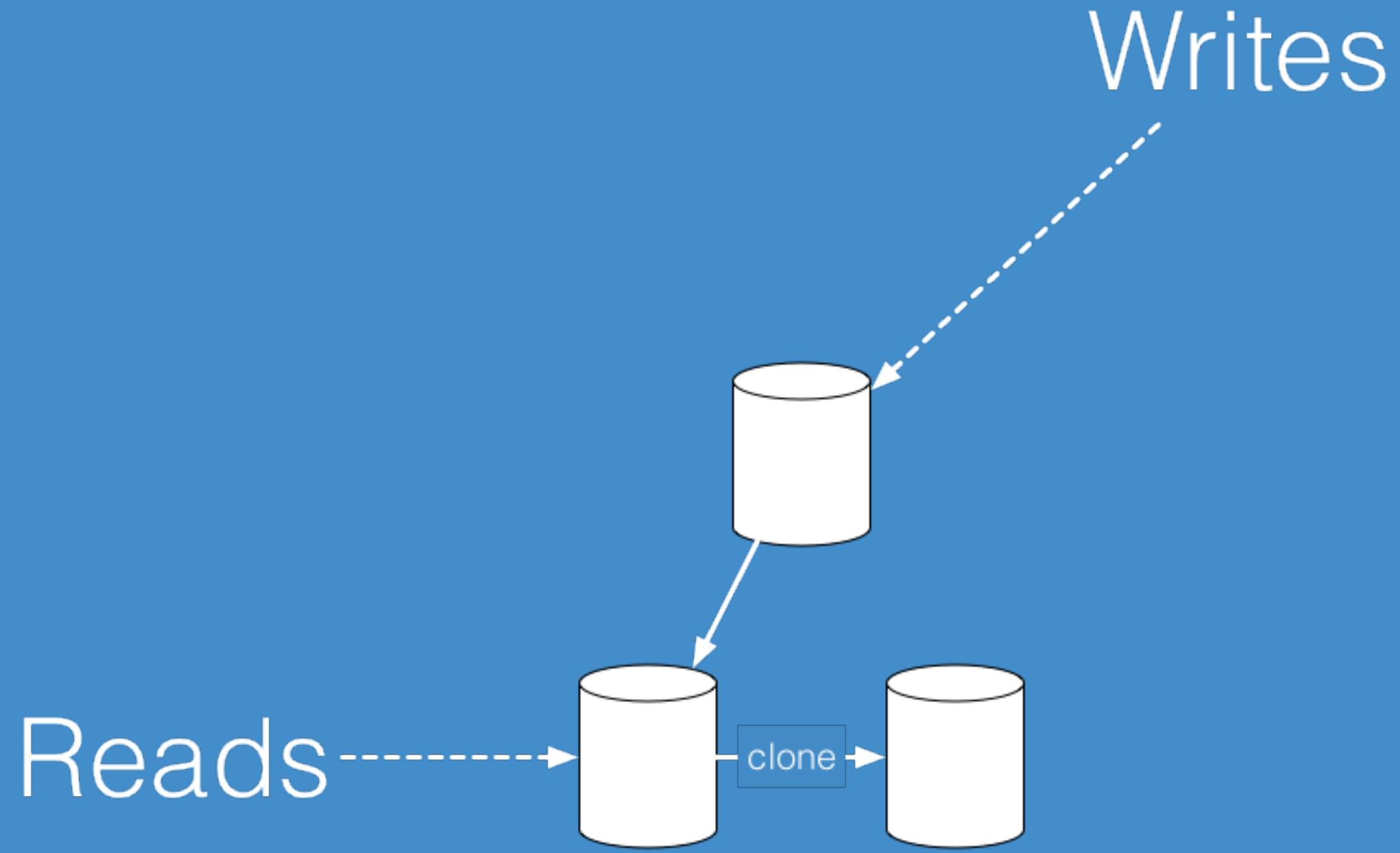


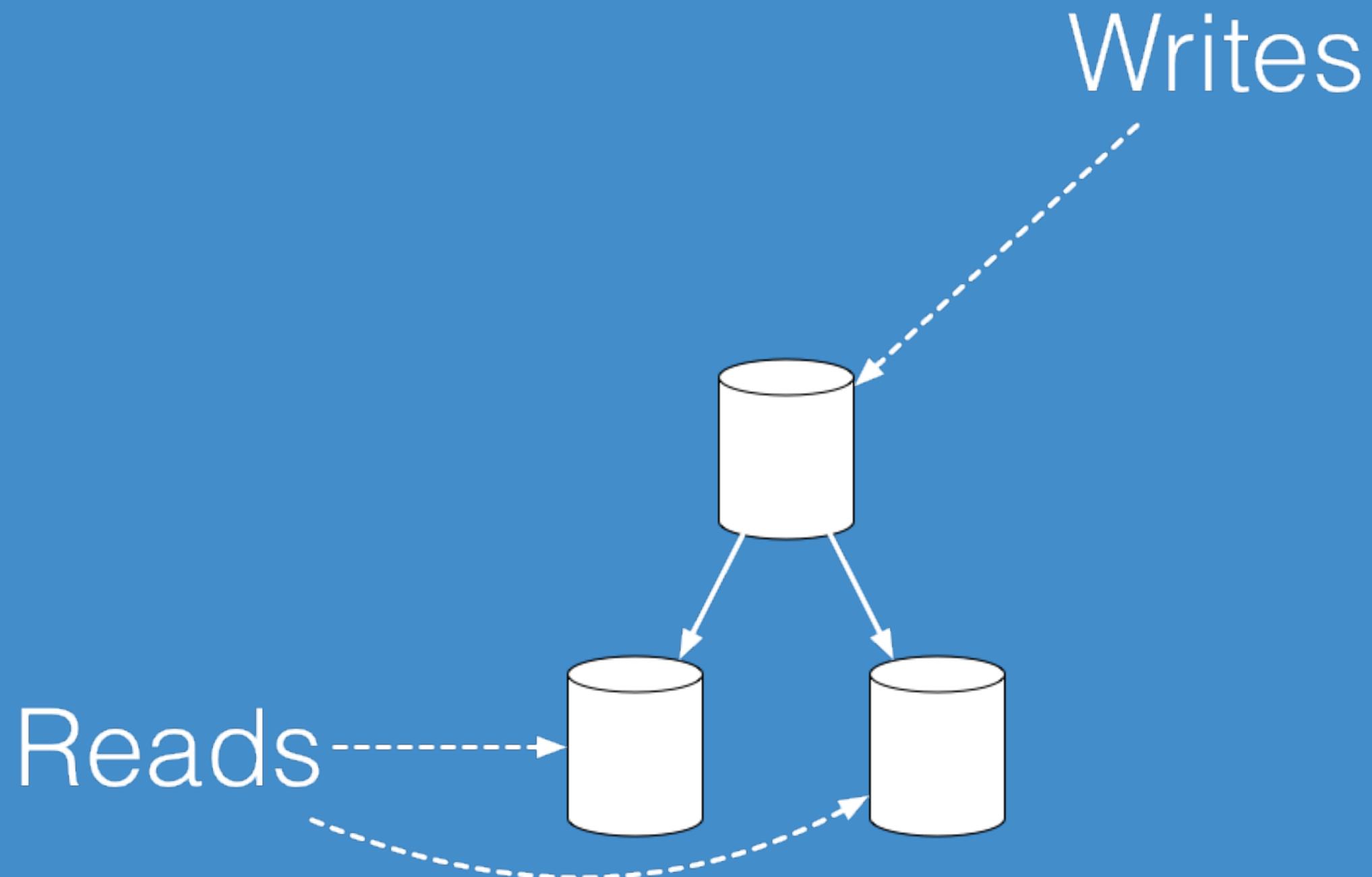
Writes



Reads











t

- Collins, an asset tracker
- Real Linux systems with SSH
- Functioning network
- Actual MySQL



```
import ./make-test.nix ({ ... }: {  
    # this will never work  
    name = "jetpants-test";
```



```
$ nix-build ./jetpants-test.nix
```



```
$ nix-build ./jetpants-test.nix
. . . snip . . .
/nix/store/brz8in768iax30kkwn6mry1121j4-jetpants-test
```





```
import ../../make-test.nix  ({ helpers, ... }:
{
  name = "verify-default-environment";

  starting-spare-dbs = 0;
  Starting-replica-dbs = 2;

  test-phases = with helpers; [
    (assert-shard-exists "POSTS-1-INFINITY")
    (assert-master-has-n-replicas
      "POSTS-1-INFINITY" 2)
  ];
})
```

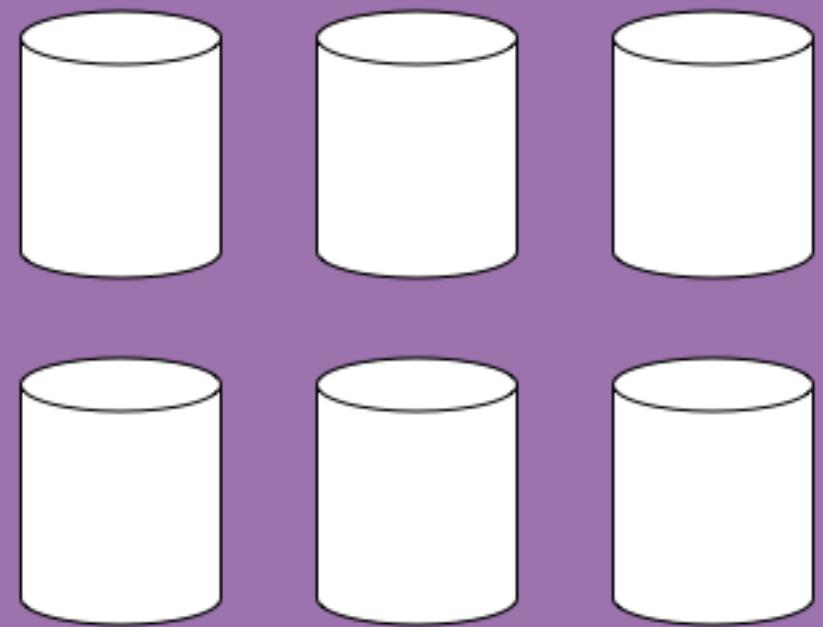


```
import ../../make-test.nix ({ helpers, ... }:
{
  name = "verify-default-environment";

starting-spare-dbs = 0;

starting-replica-dbs = 2;

  test-phases = with helpers; [
    (assert-shard-exists "POSTS-1-INFINITY")
    (assert-master-has-n-replicas
      "POSTS-1-INFINITY" 2)
  ];
})
```



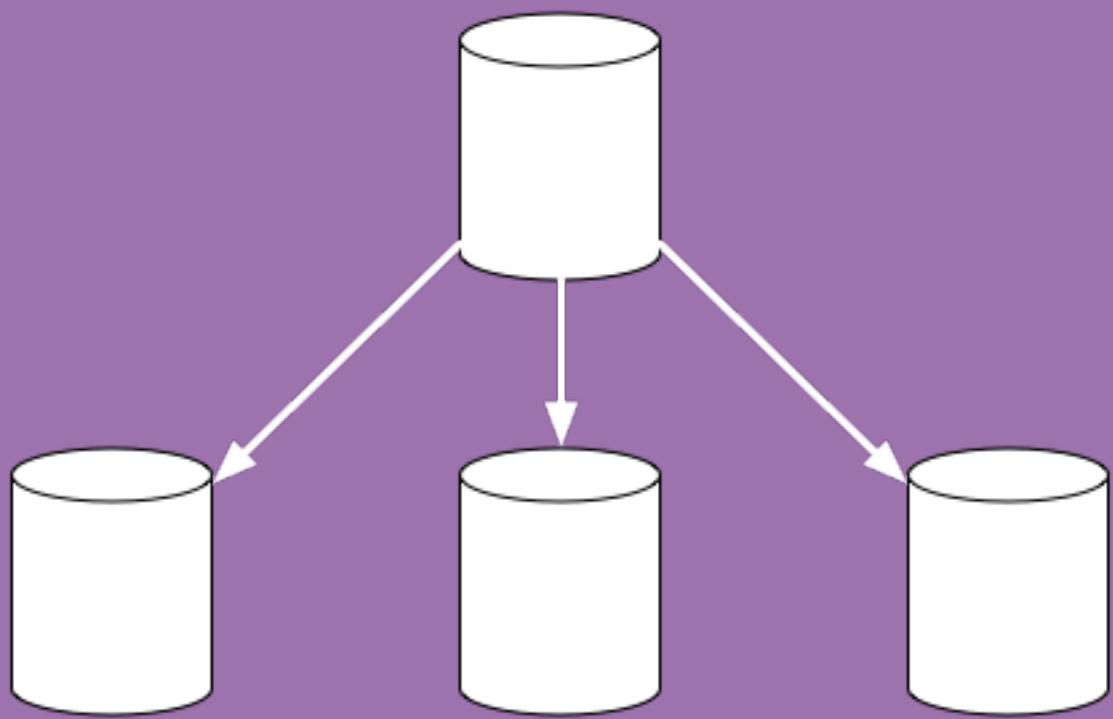
starting-spare-dbs = 6

```
import ../../make-test.nix ({ helpers, ... }:
{
  name = "verify-default-environment";

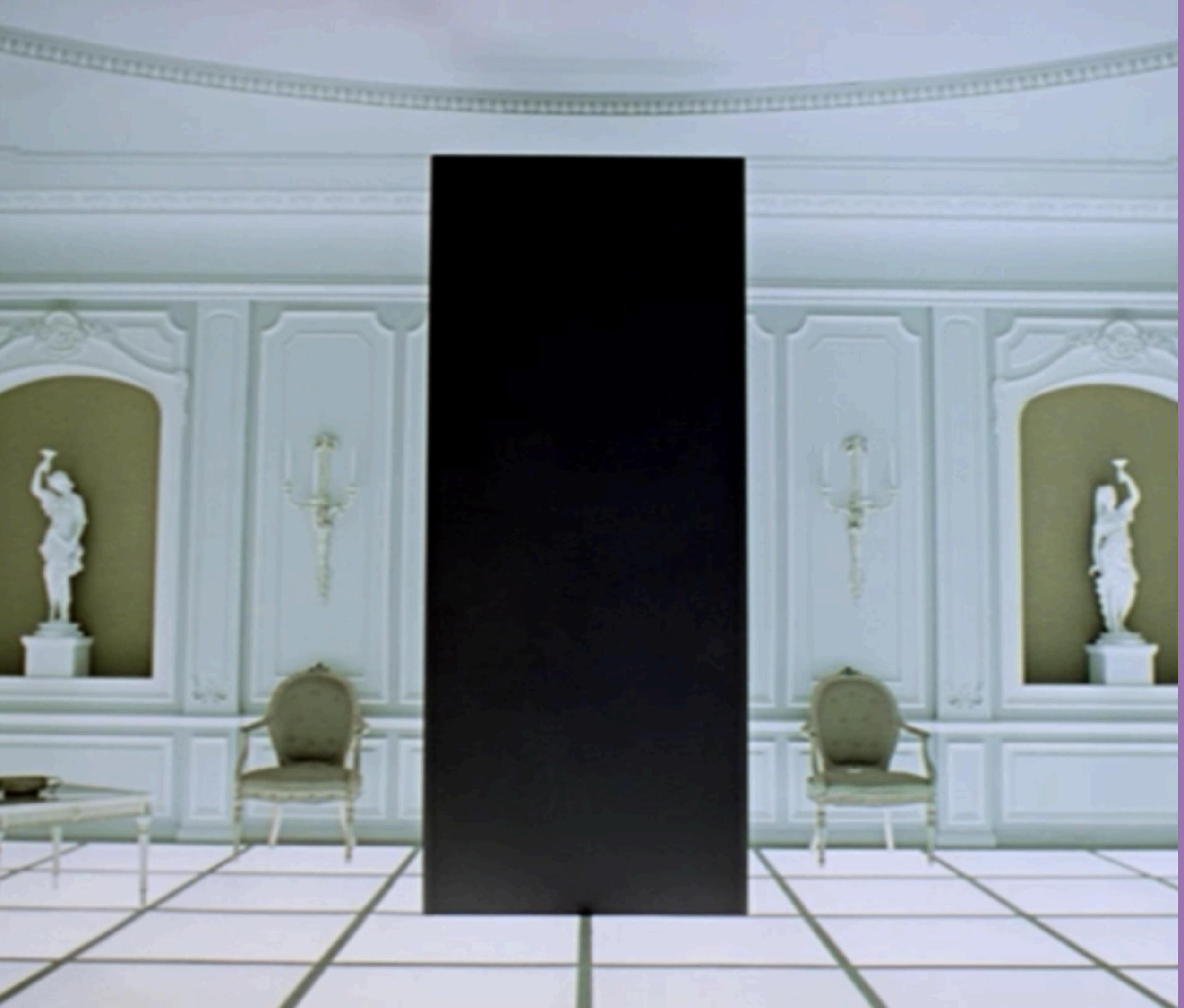
  starting-spare-dbs = 0;

starting-replica-dbs = 2;

  test-phases = with helpers; [
    ...
  ];
})
```



starting-replica-dbs = 3



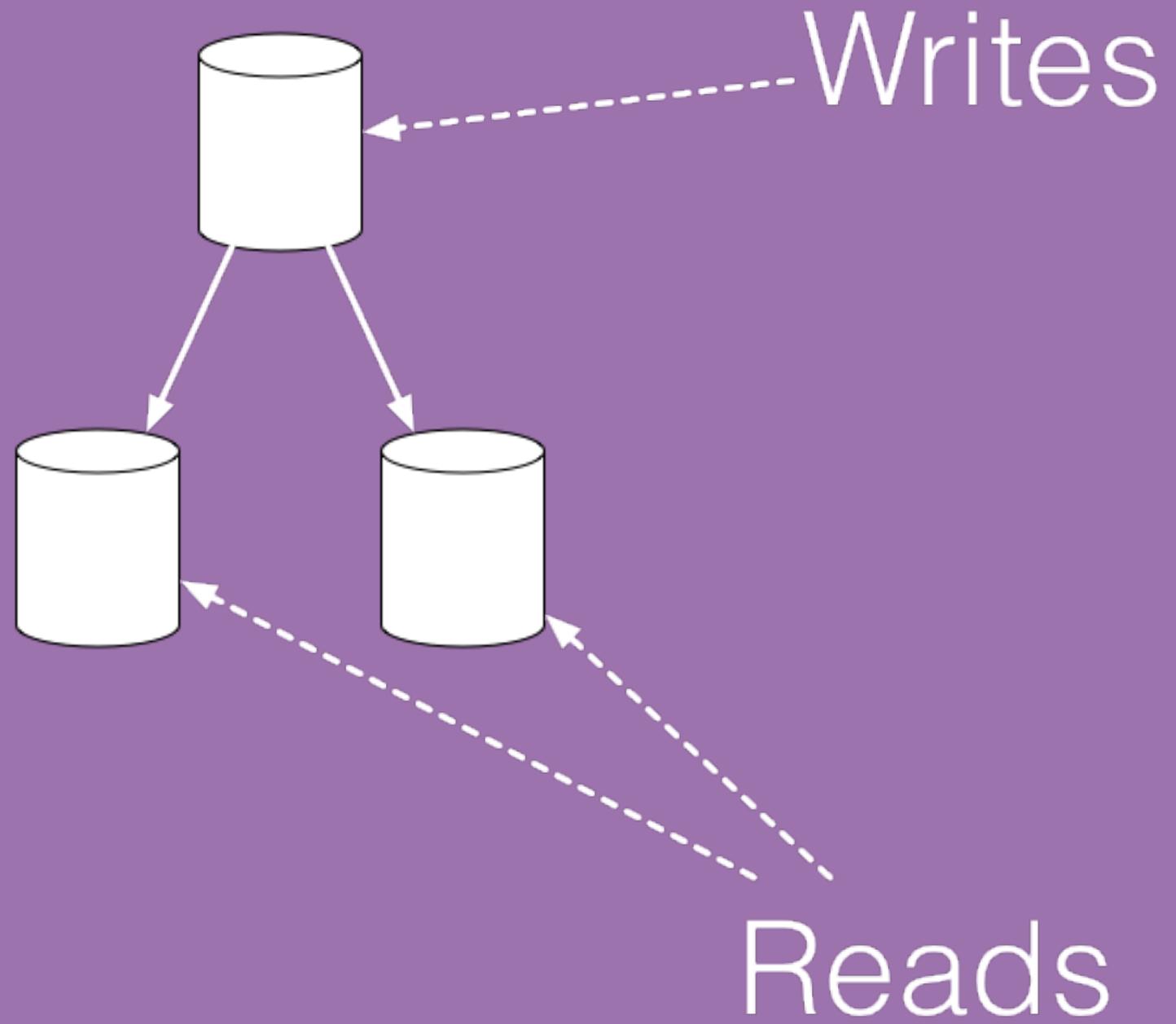
t

Hard Promotion Test

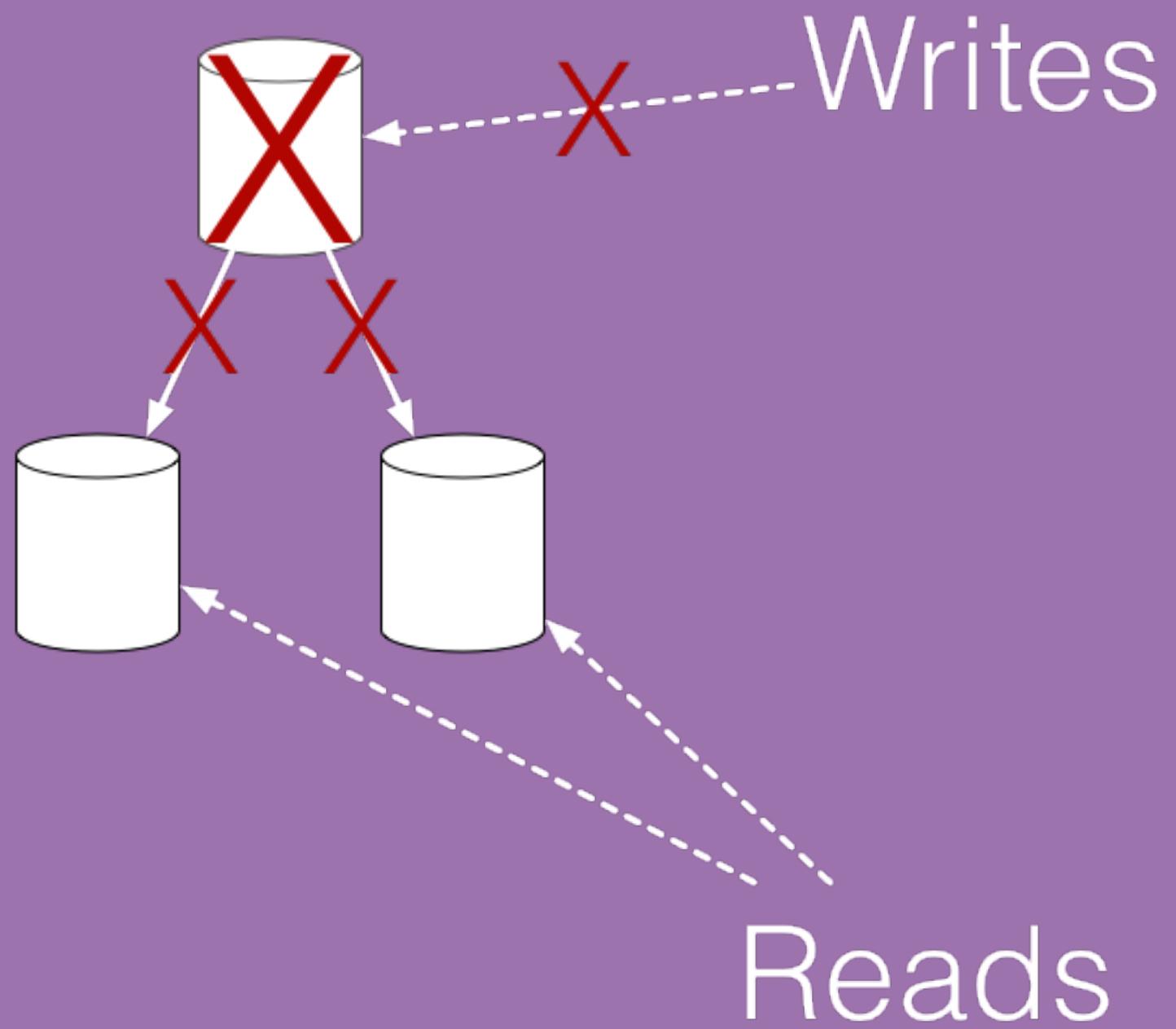


```
import ./make-test.nix  ({ helpers, pkgs, ... }:  
{  
    name = "shard-master-promotion";  
    starting-slave-dbs = 2;  
  
    test-phases = with helpers; [  
        shardMasterPromotion;  
    ];  
};)  
in  
pkgs.lib.mkDerivation {  
    pname = "shard-master-promotion";  
    version = "1.0";  
    src = ./shard-master-promotion.nix;  
    buildPhase = "shardMasterPromotion";  
    buildInputs = [ pkgs.lib.mkShell ];  
    meta = {  
        description = "A Nix script for shard master promotion";  
        homepage = "https://github.com/graahc/shard-master-promotion";  
        license = pkgs.licenses.mit;  
        maintainers = [ pkgs.lib.maintainer "Graham Christensen" ];  
    };  
}
```



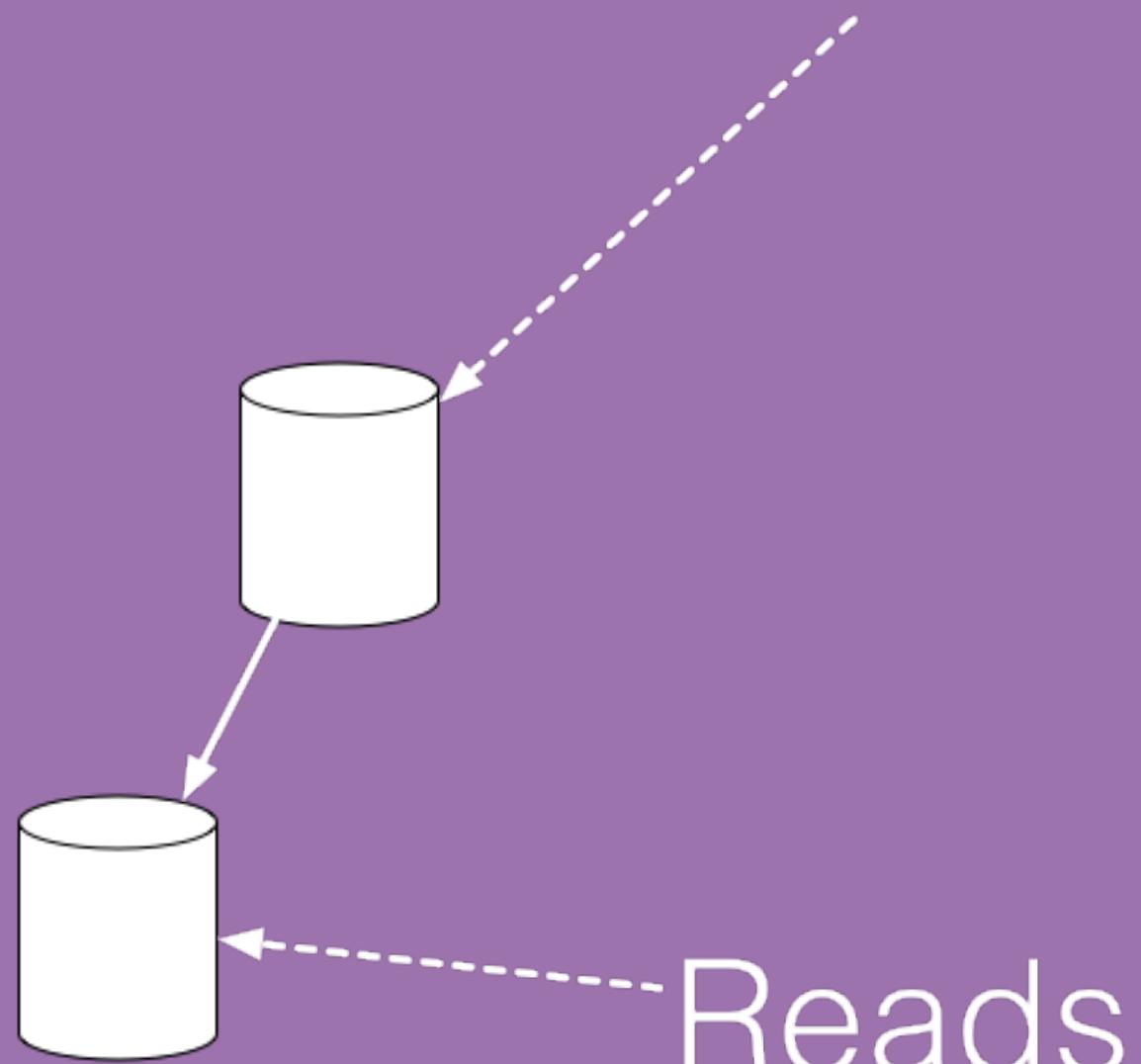


```
test-phases = with helpers; [
  (jetpants-phase "shutdown"
    ""
    Jetpants.pool(
      "POSTS-1-INFINITY"
      ).master.stop_mysql
    ""
  )
]
```

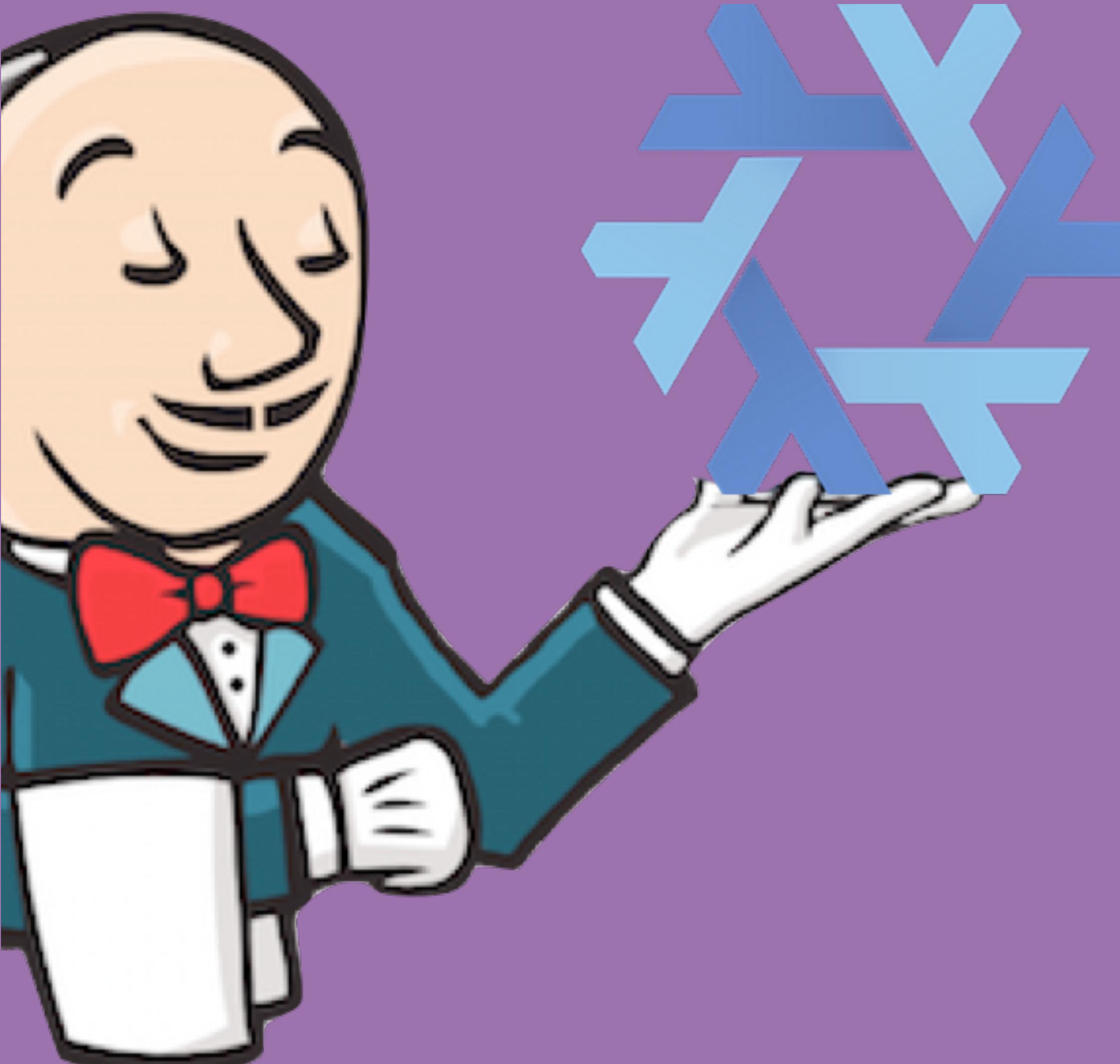


```
(jetpants-phase "shutdown-master" ''
  Jetpants.pool(
    "POSTS-1-INFINITY"
  ).master.stop_mysql
'')
(phase "jetpants-promotion"
  ''
  jetpants promotion \
    --demote=10.50.2.10 \
    --promote=10.50.2.11
'')
```

Writes



```
(phase "jetpants-promotion"
  "
    jetpants promotion \
      --demote=10.50.2.10 \
      --promote=10.50.2.11
  ")
(asserT-shard-master
  "POSTS-1-INFINITY"
  "10.50.2.11")
(asserT-shard-replica
  "POSTS-1-INFINITY"
  "10.50.2.10")
```



An Internal DB



- Golang dev environment
- Dozens of extra dev-time tools
- Involved build process



- 3 developers using Nix to manage the environment
- No time wasted on Jenkins
- Nix-shell for dev, using same build phases Jenkins does



- Tests integrate with Jetpants
- Nix's modularity made this easy



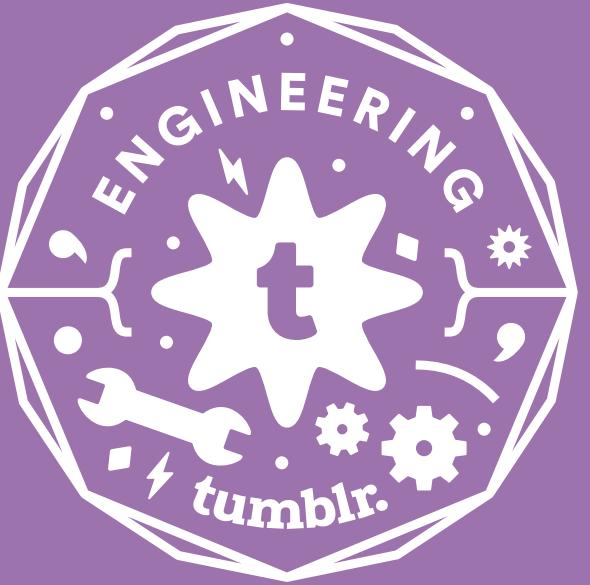
- Mature traceability out of the gate
- Nix produces a deployable artifact
- Deployed with Puppet



A possibility for the future...

Memcache





What are we working with?

- 3M+ cache hits a second
- Many nodes, many purposes



A LOOK INTO

Memcache

- A simple in-memory
- get, set, increment, decrement



- Currently deployed on stateful nodes
- Annoying, when disks fail...
- Slow to reprovision after a failure(Puppet)



- Added NixOS support to the iPXE infrastructure
- Work in progress, likely a production pilot in early 2018







t beta
@grhmc
github.com/grahamc

<https://github.com/tumblr/jetpants>

<https://tumblr.com/jobs>

