The Etsy Shard Architecture

Starts With S and Ends With Hard

jgoulah@etsy.com / @johngoulah

Etsy

1.5B page views / mo.

525MM sales in 2011

40MM unique visitors/mo.

800K shops / 150 countries


APACHE
HBASE


MySQL®


redis


SQLite


MySQL®

25K+ queries/sec avg

3TB InnoDB buffer pool

15TB+ data stored

99.99% queries under 1ms

50+ MySQL servers

Server Spec

HP DL 380 G7

96GB RAM

16 spindles / 1TB RAID 10

24 Core

Ross Snyder

Scaling Etsy - What Went Wrong, What Went Right

http://bit.ly/rpcxtP

Matt Graham

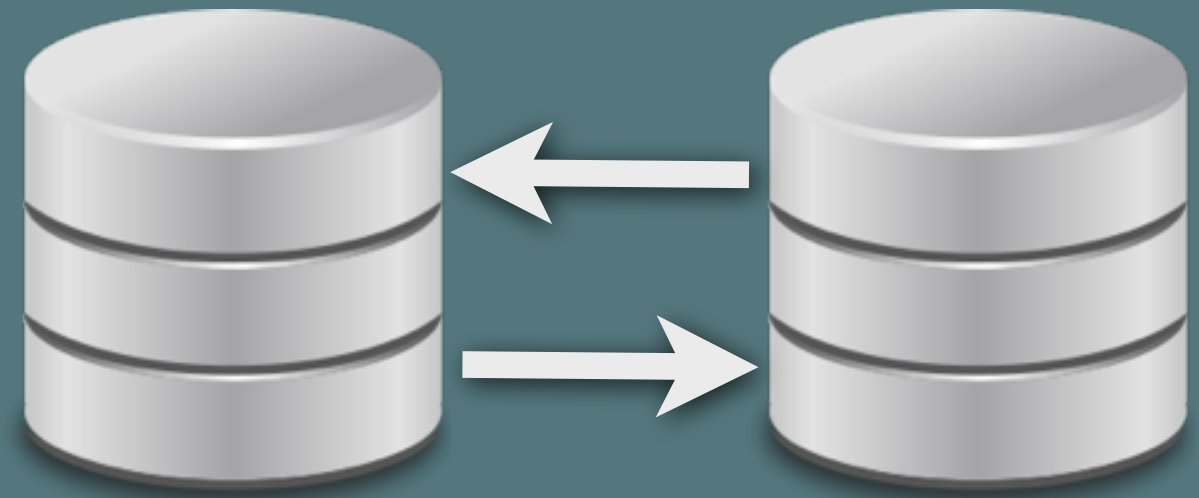
Migrating From PG to MySQL Without Downtime

http://bit.ly/rQpqZG

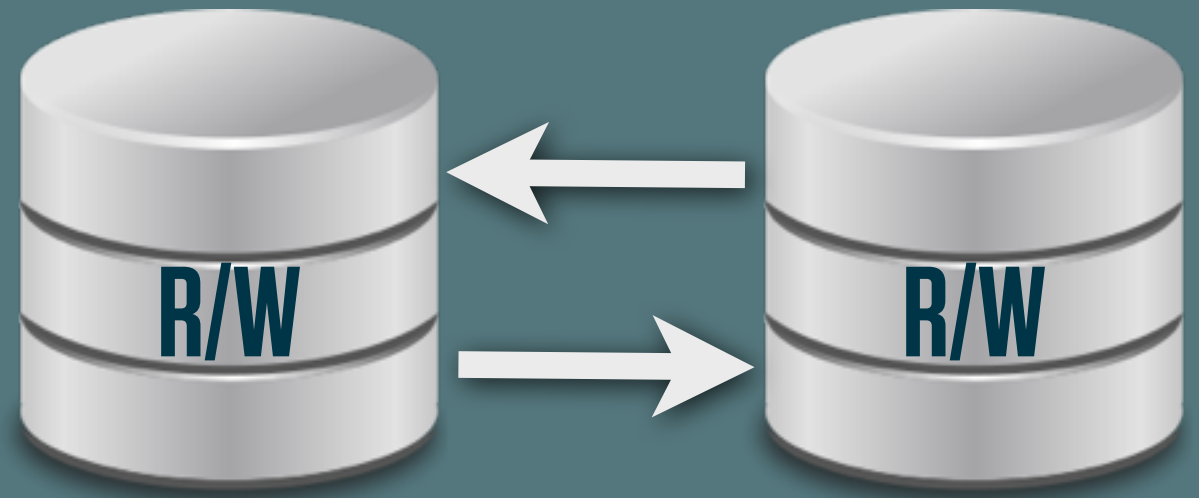
Architecture

Redundancy

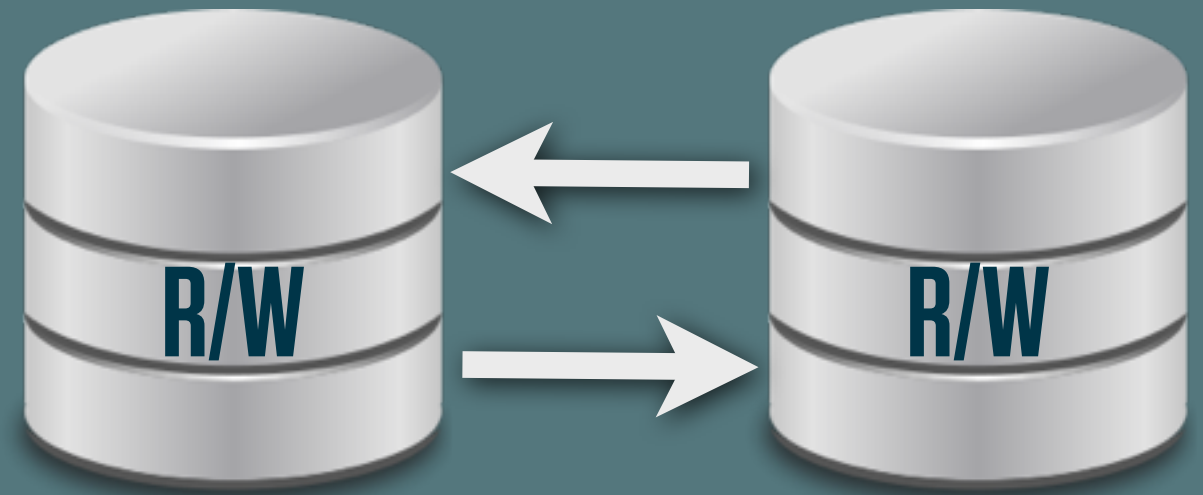
Master - Master



Master - Master

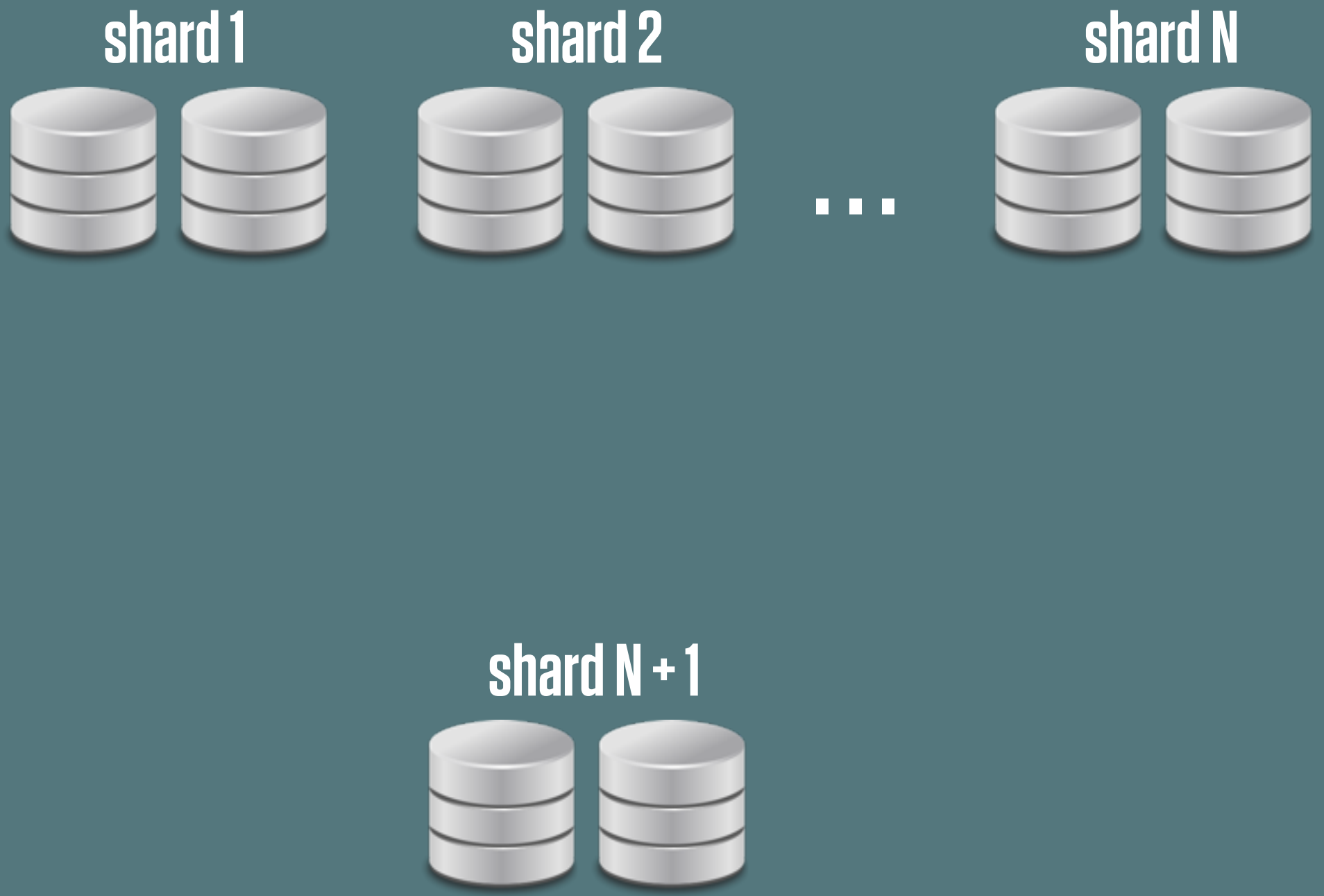


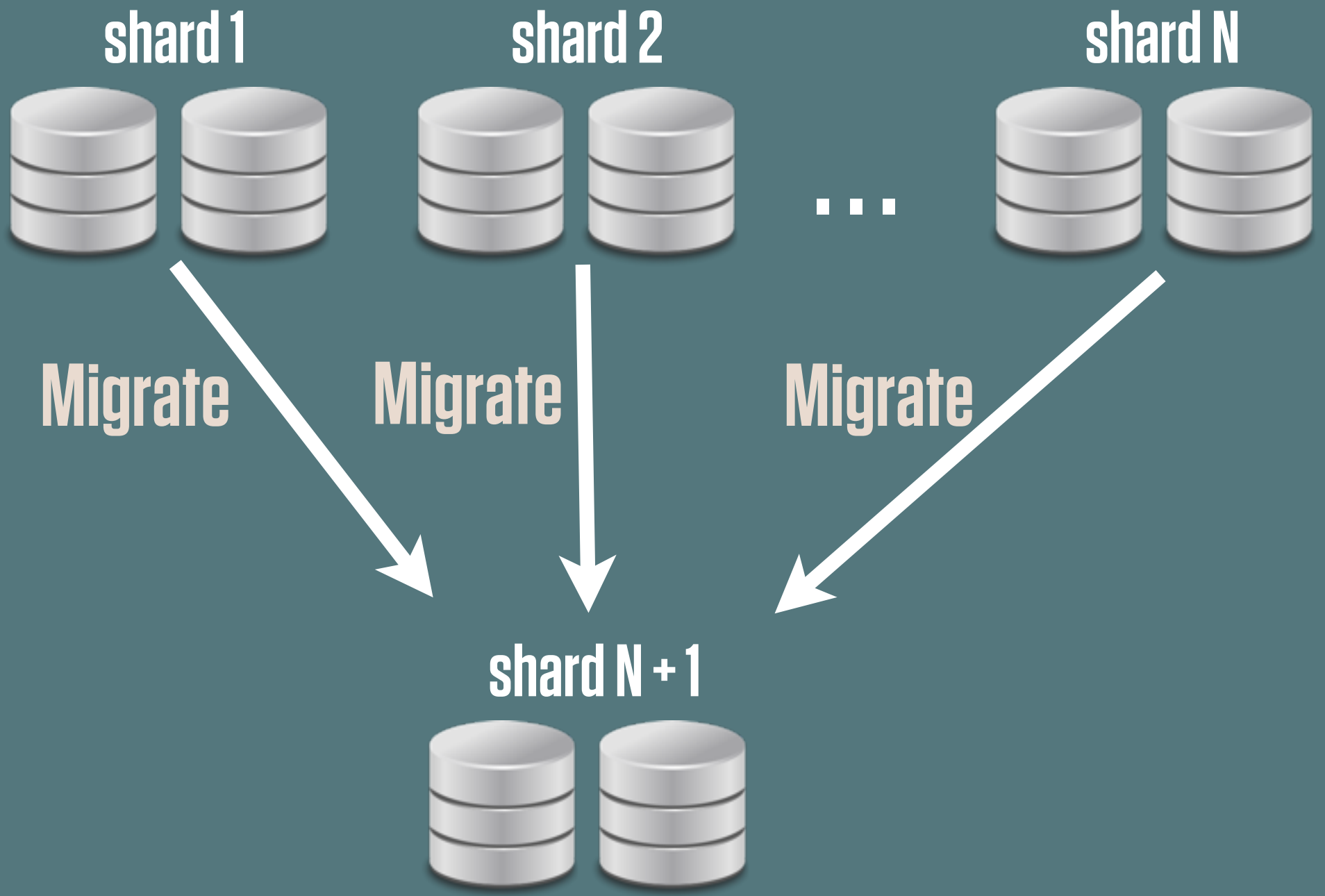
Master - Master



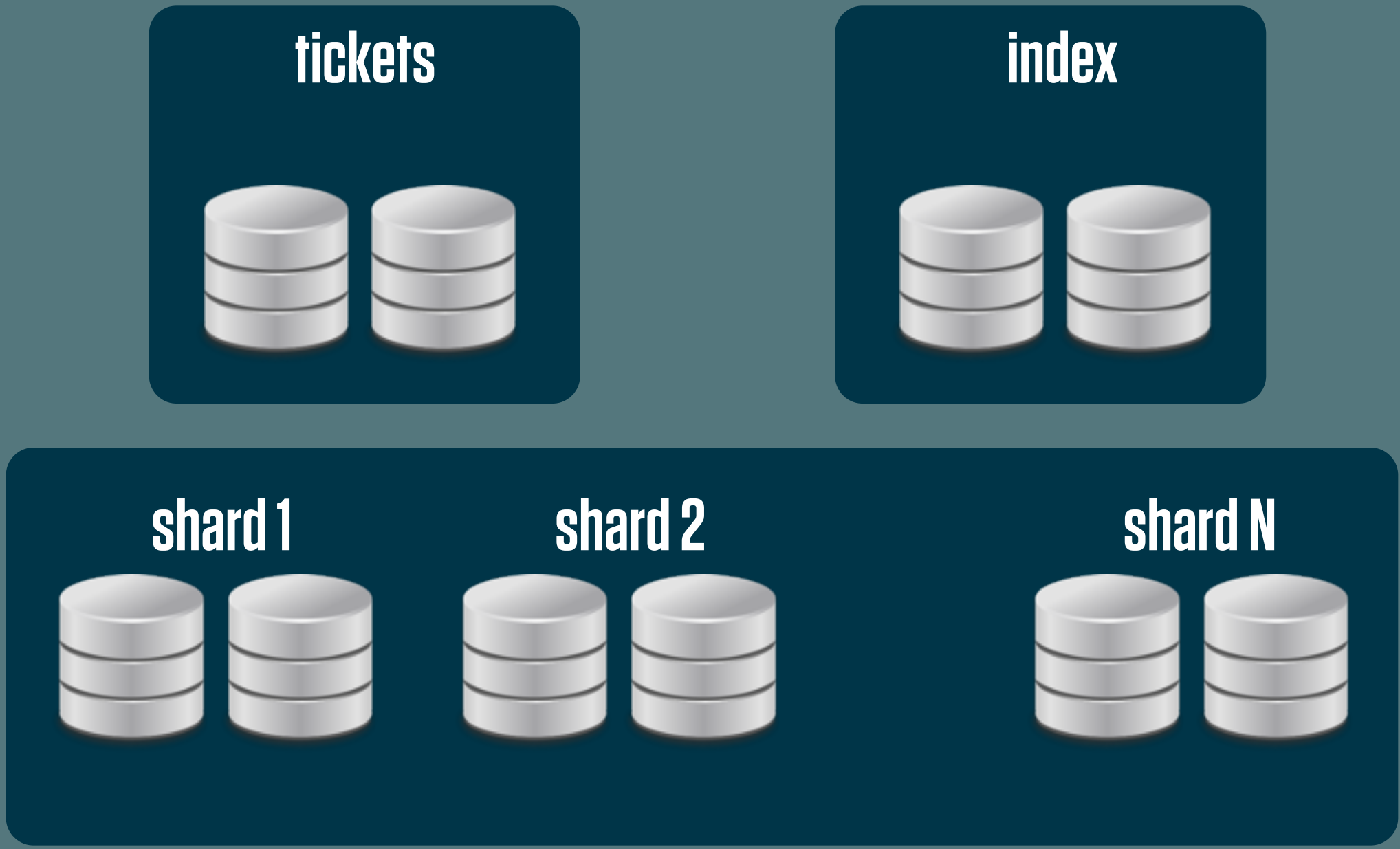
Scalability

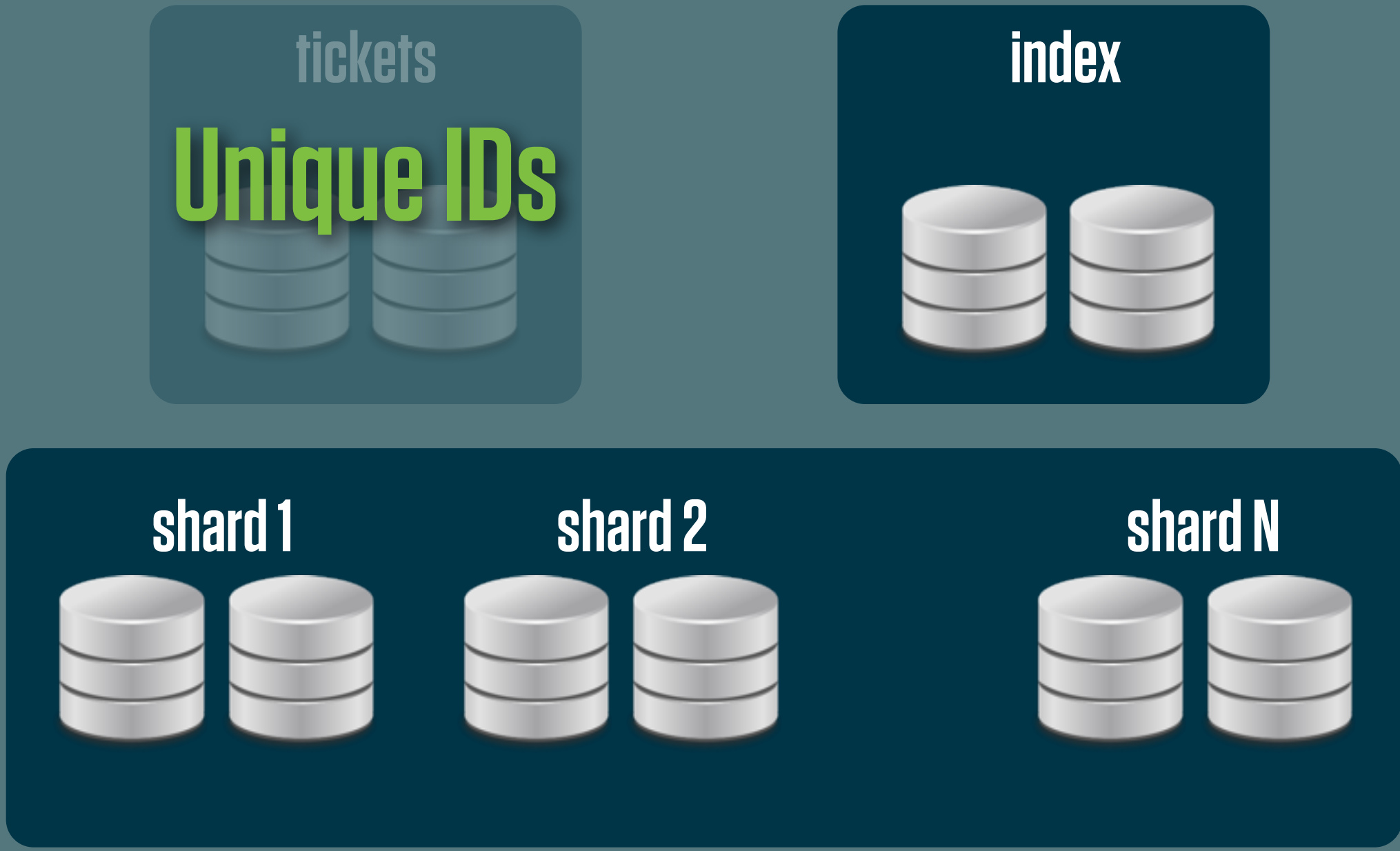

shard 1
shard 2
shard N
...

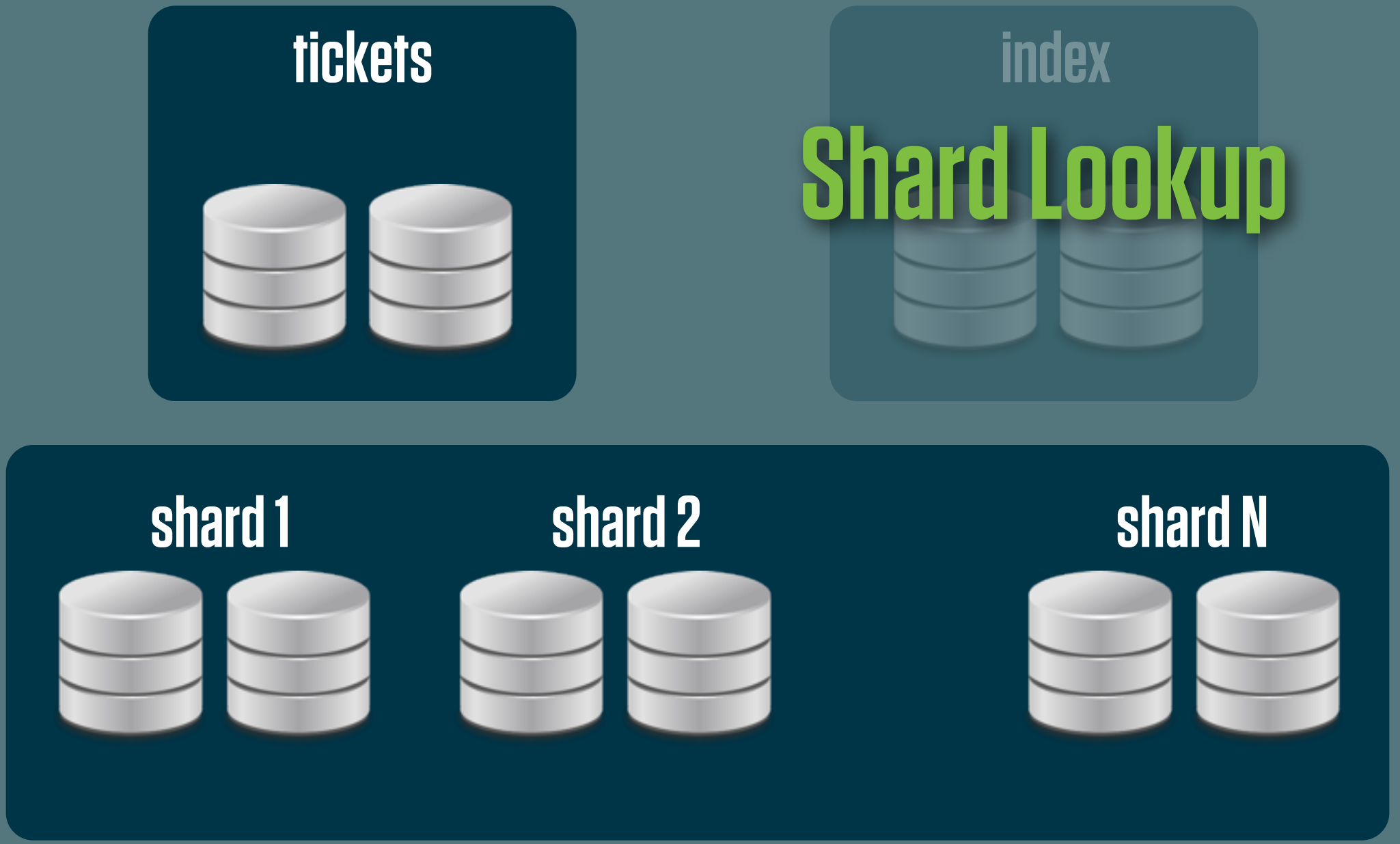

shard 1
shard 2
shard N
...
shard N + 1

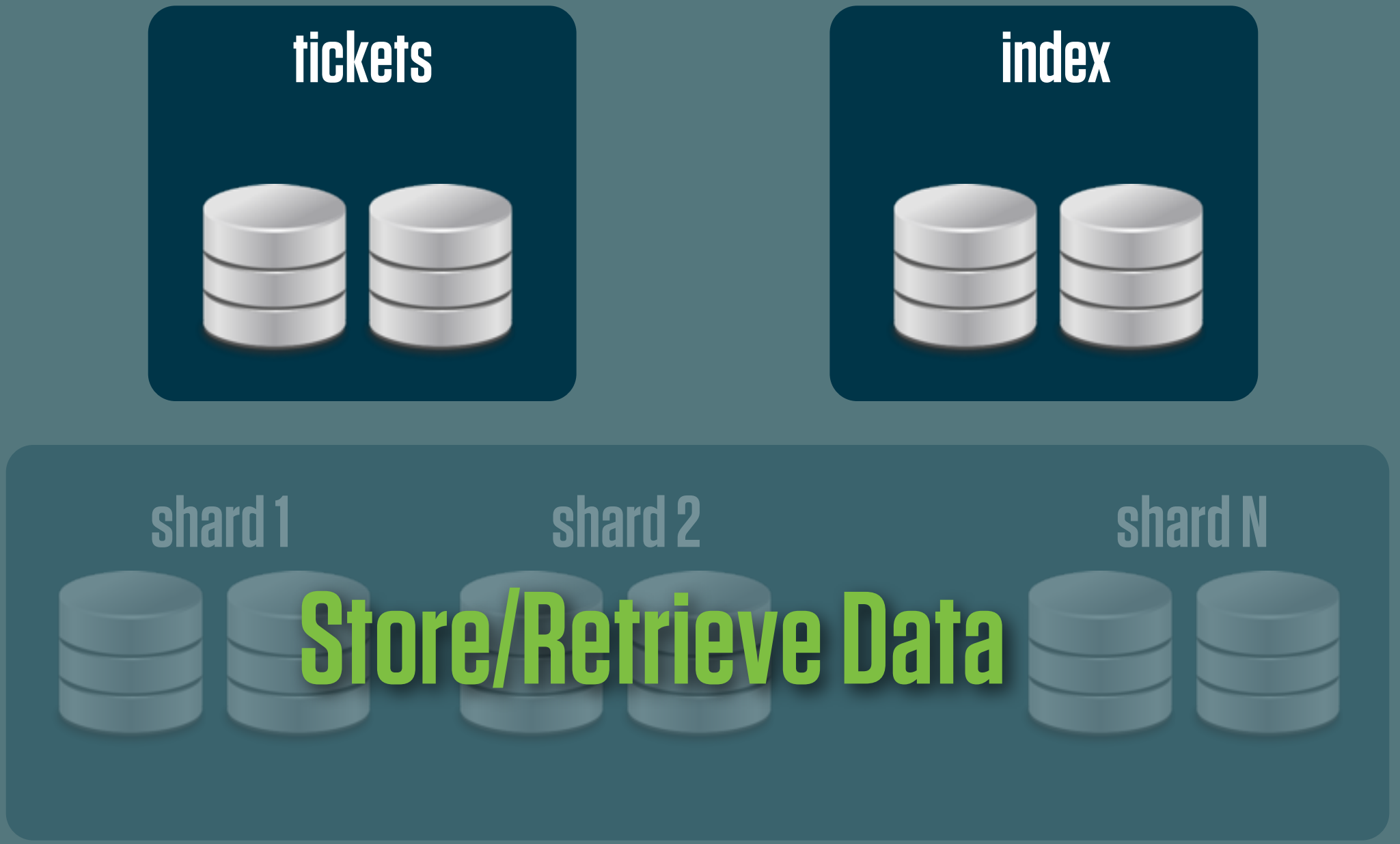

shard 1
shard 2
shard N
...
Migrate
Migrate
Migrate
shard N + 1

Bird's-Eye View


tickets
index
shard 1
shard 2
shard N


tickets
Unique IDs
index
shard 1
shard 2
shard N


tickets
index
Shard Lookup
shard 1
shard 2
shard N


tickets
index
shard 1
shard 2
shard N
Store/Retrieve Data

Basics

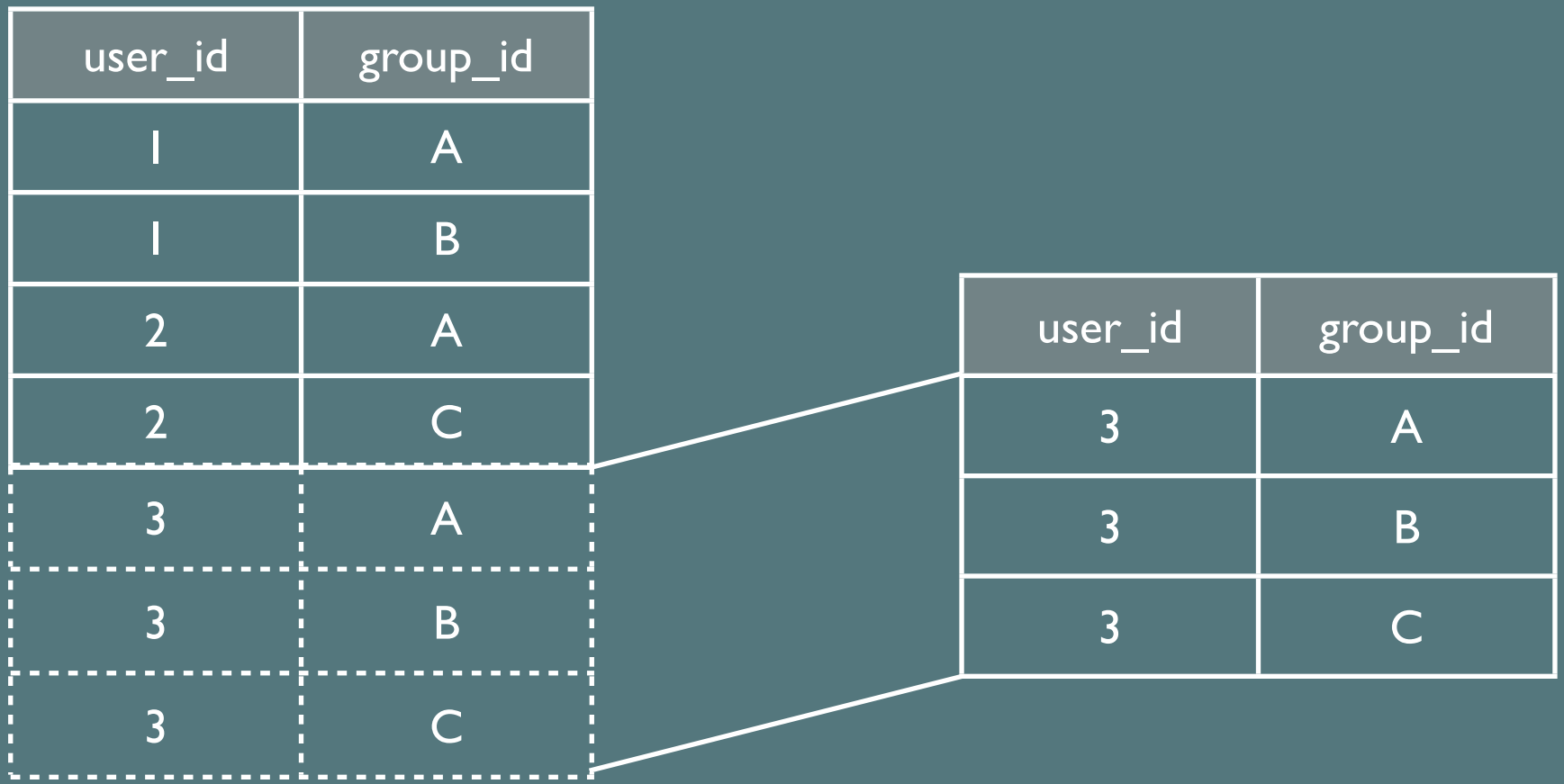
users_groups

user_id	group_id
1	A
1	B
2	A
2	C
3	A
3	B
3	C

users_groups

user_id	group_id
1	A
1	B
2	A
2	C
3	A
3	B
3	C

users_groups



users_groups

shard 1

user_id	group_id
1	A
1	B
2	A
2	C

shard 2

user_id	group_id
3	A
3	B
3	C


Index Servers

Shards NOT Determined by

key hashing

range partitions

partitioning by function

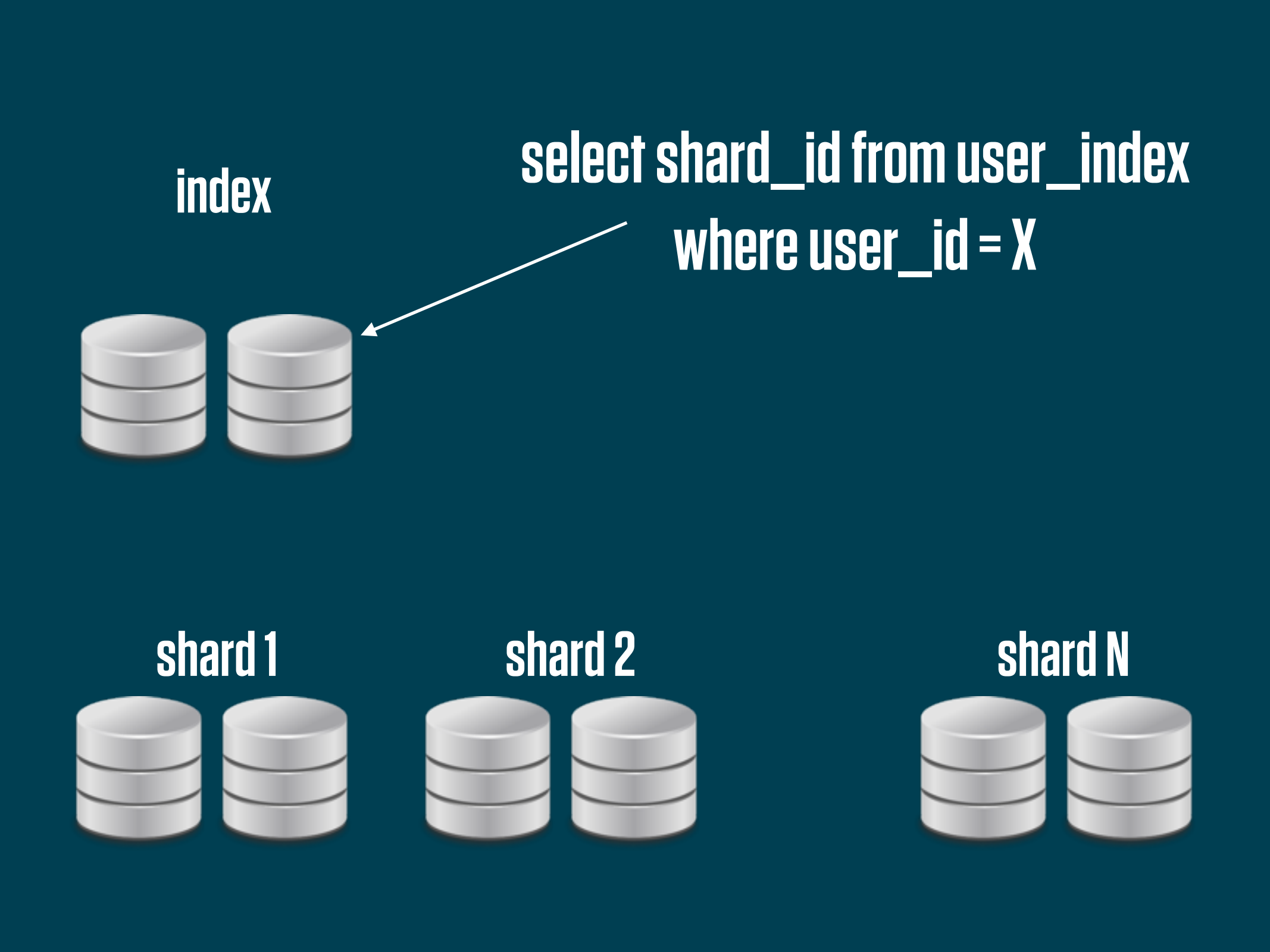
Look-Up Data

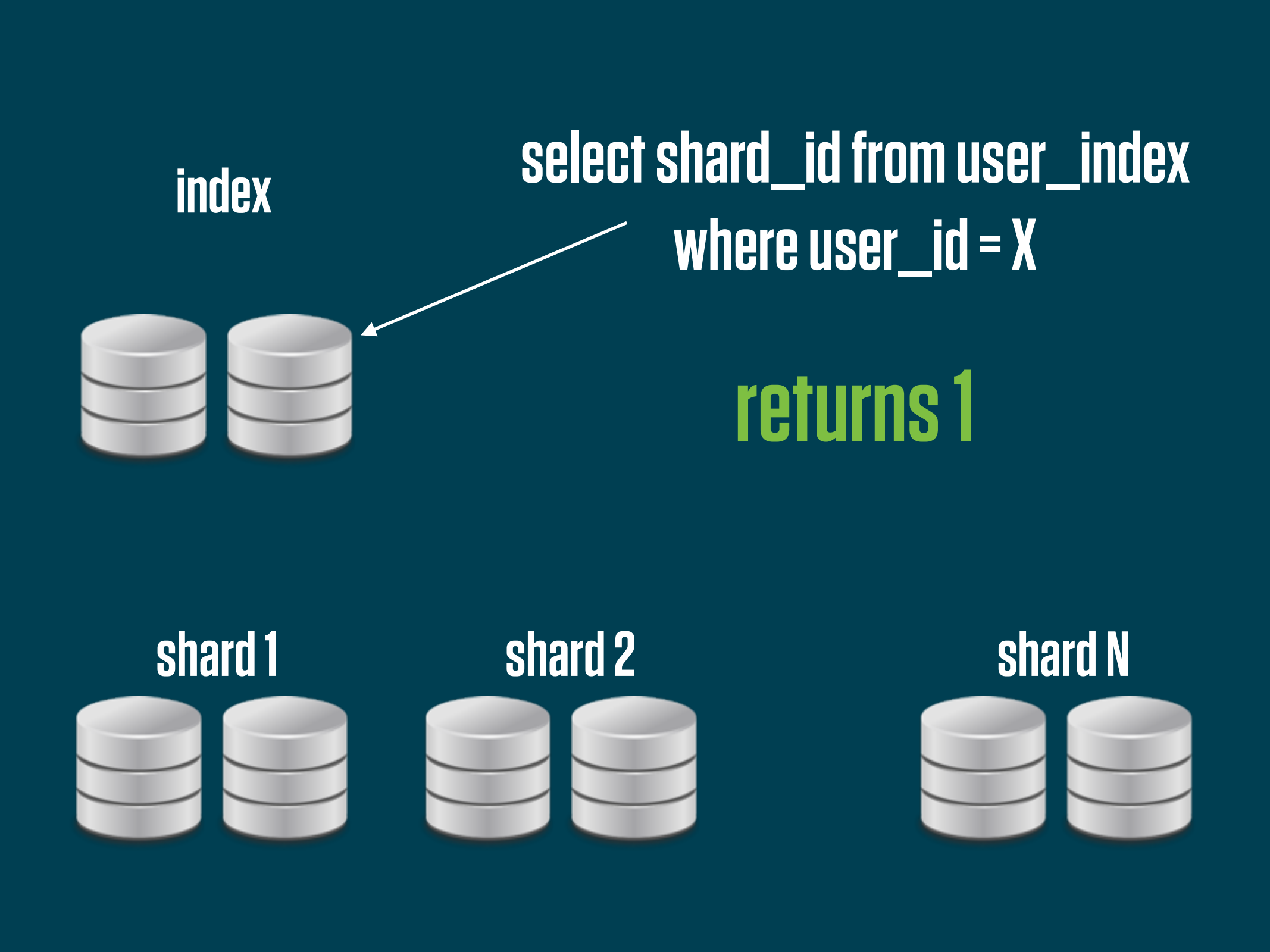
index

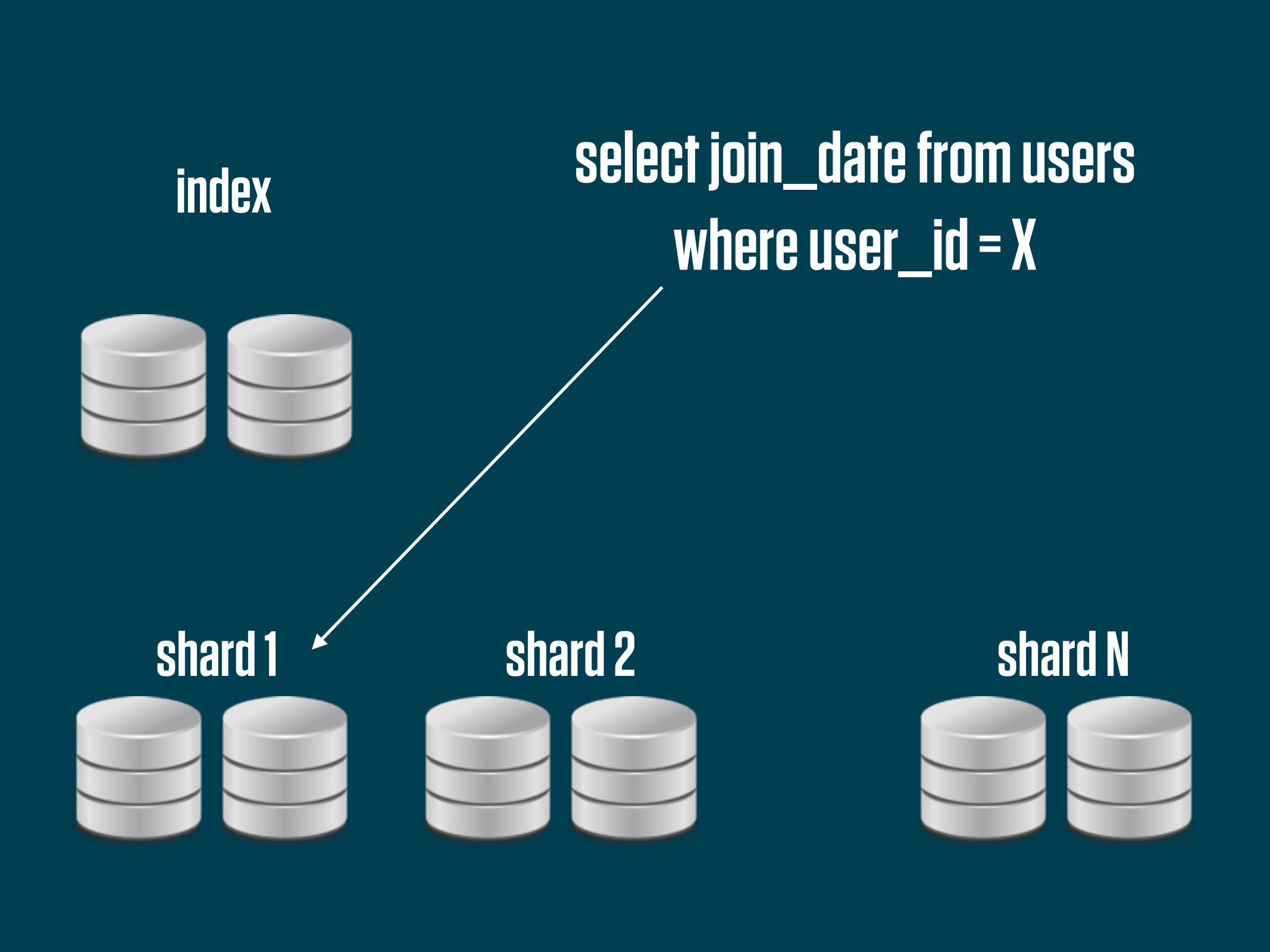
shard 1

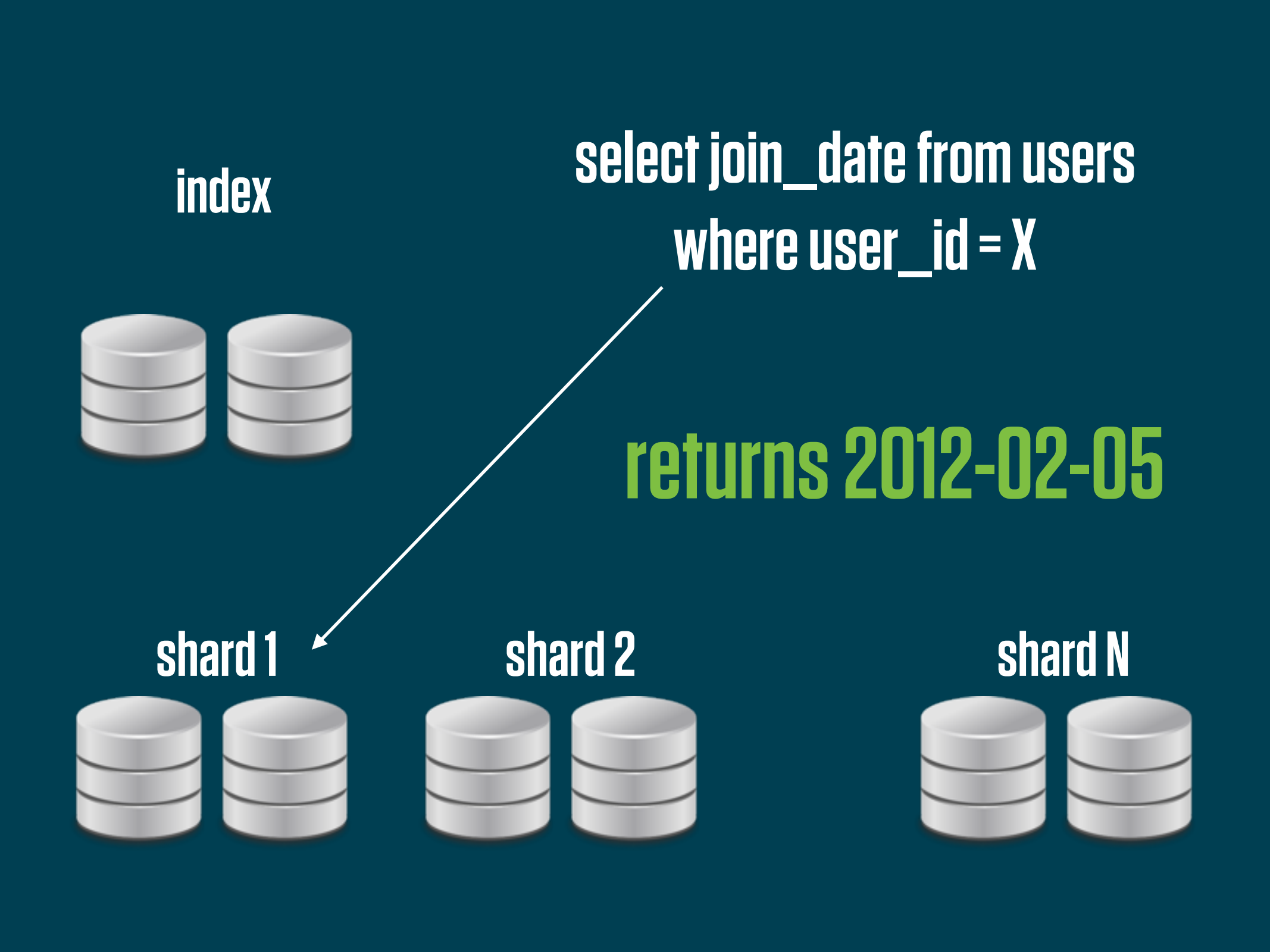
shard 2

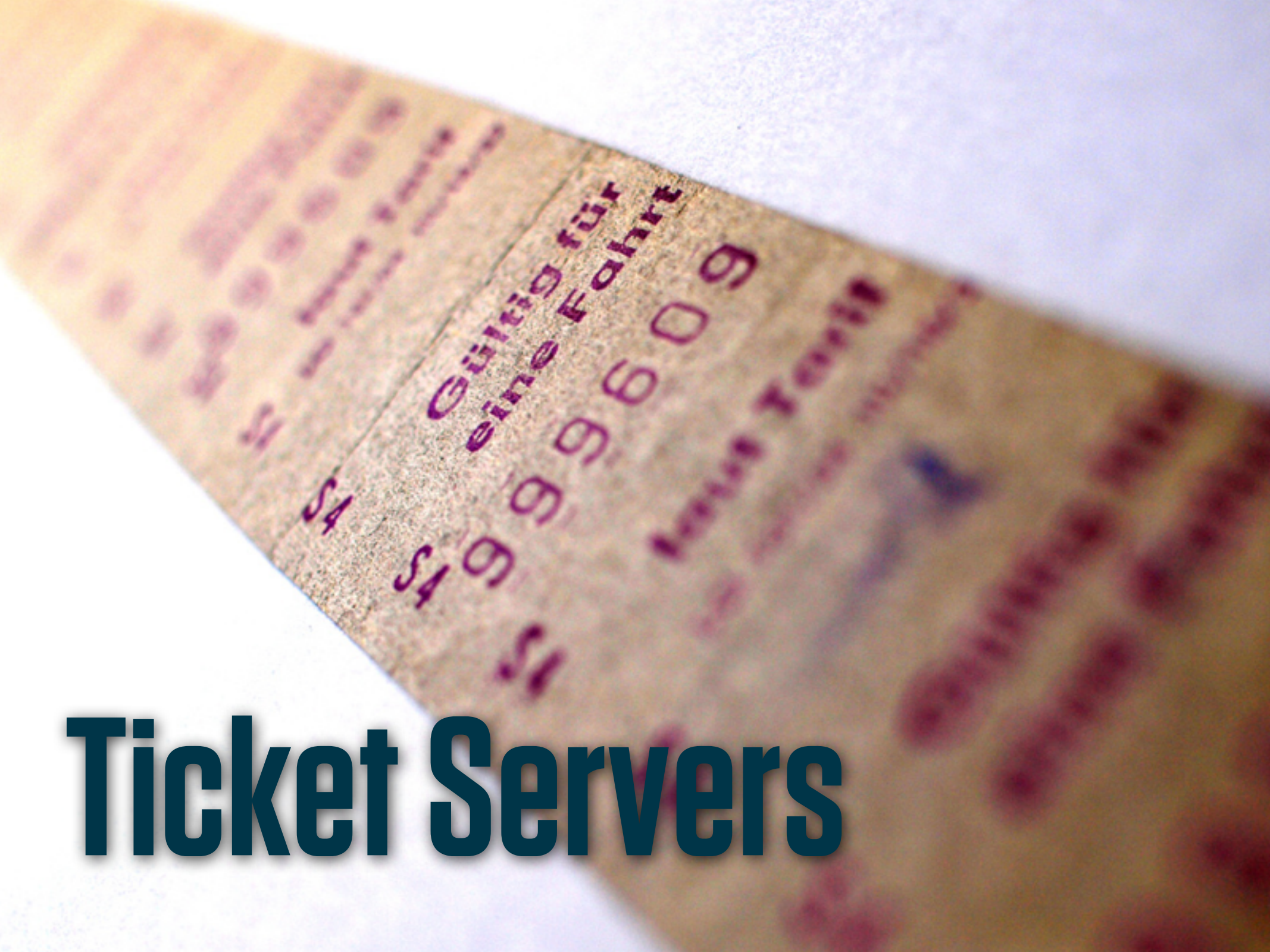
shard N


index
select shard_id from user_index
where user_id = X
shard 1
shard 2
shard N


index
select shard_id from user_index
where user_id = X
returns 1
shard 1
shard 2
shard N


index
select join_date from users
where user_id = X
shard 1
shard 2
shard N


index
select join_date from users
where user_id = X
returns 2012-02-05
shard 1
shard 2
shard N



Ticket Servers

Globally Unique ID

```
CREATE TABLE `tickets` (
  `id` bigint(20) unsigned NOT NULL auto_increment,
  `stub` char(1) NOT NULL default '',
  PRIMARY KEY (`id`),
  UNIQUE KEY `stub` (`stub`)
) ENGINE=MyISAM
```

Ticket Generation

```
REPLACE INTO tickets (stub) VALUES ('a');
SELECT LAST_INSERT_ID();
```

Ticket Generation

```
REPLACE INTO tickets (stub) VALUES ('a');
SELECT LAST_INSERT_ID();

SELECT * FROM tickets;
```

id	stub
4589294	a

tickets A

auto-increment-increment = 2

auto-increment-offset = 1

tickets B

auto-increment-increment = 2

auto-increment-offset = 2

tickets A

auto-increment-increment = 2

auto-increment-offset = 1

tickets B

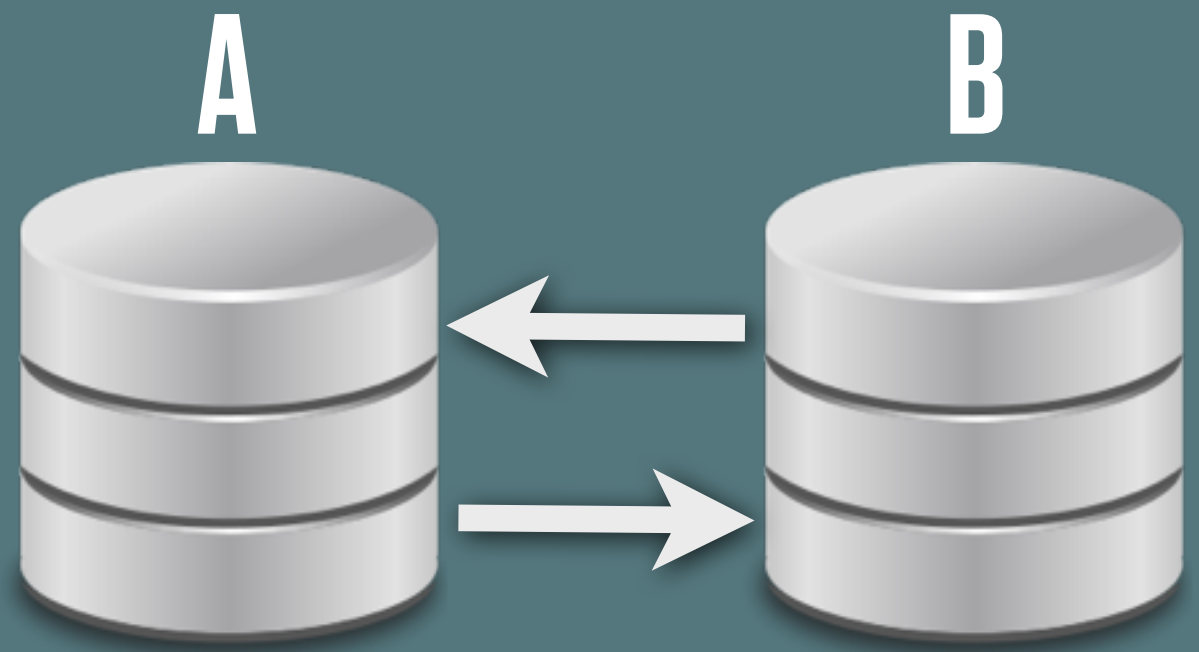
auto-increment-increment = 2

auto-increment-offset = 2

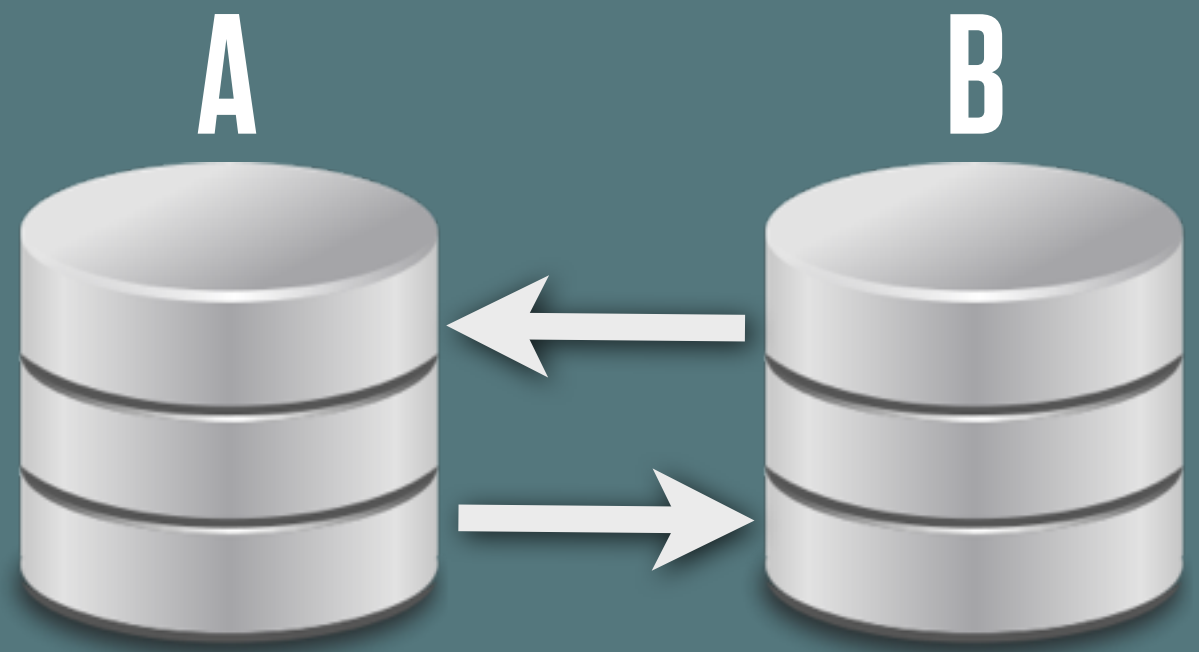
NOT master-master


Shards

Object Hashing



user_id : 500



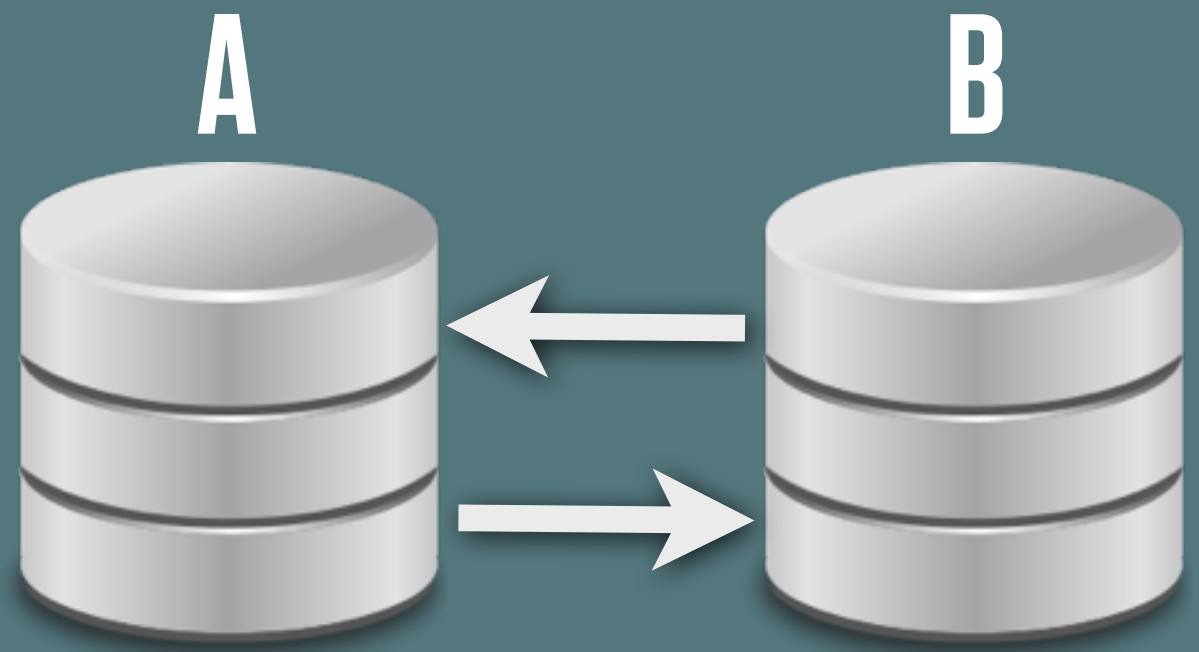
user_id : 500 % (# active replicants)

```
'etsy_index_A' => 'mysql:host=dbindex01.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_index_B' => 'mysql:host=dbindex02.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_shard_001_A' => 'mysql:host=dbshard01.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_001_B' => 'mysql:host=dbshard02.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_A' => 'mysql:host=dbshard03.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_B' => 'mysql:host=dbshard04.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_A' => 'mysql:host=dbshard05.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_B' => 'mysql:host=dbshard06.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
```

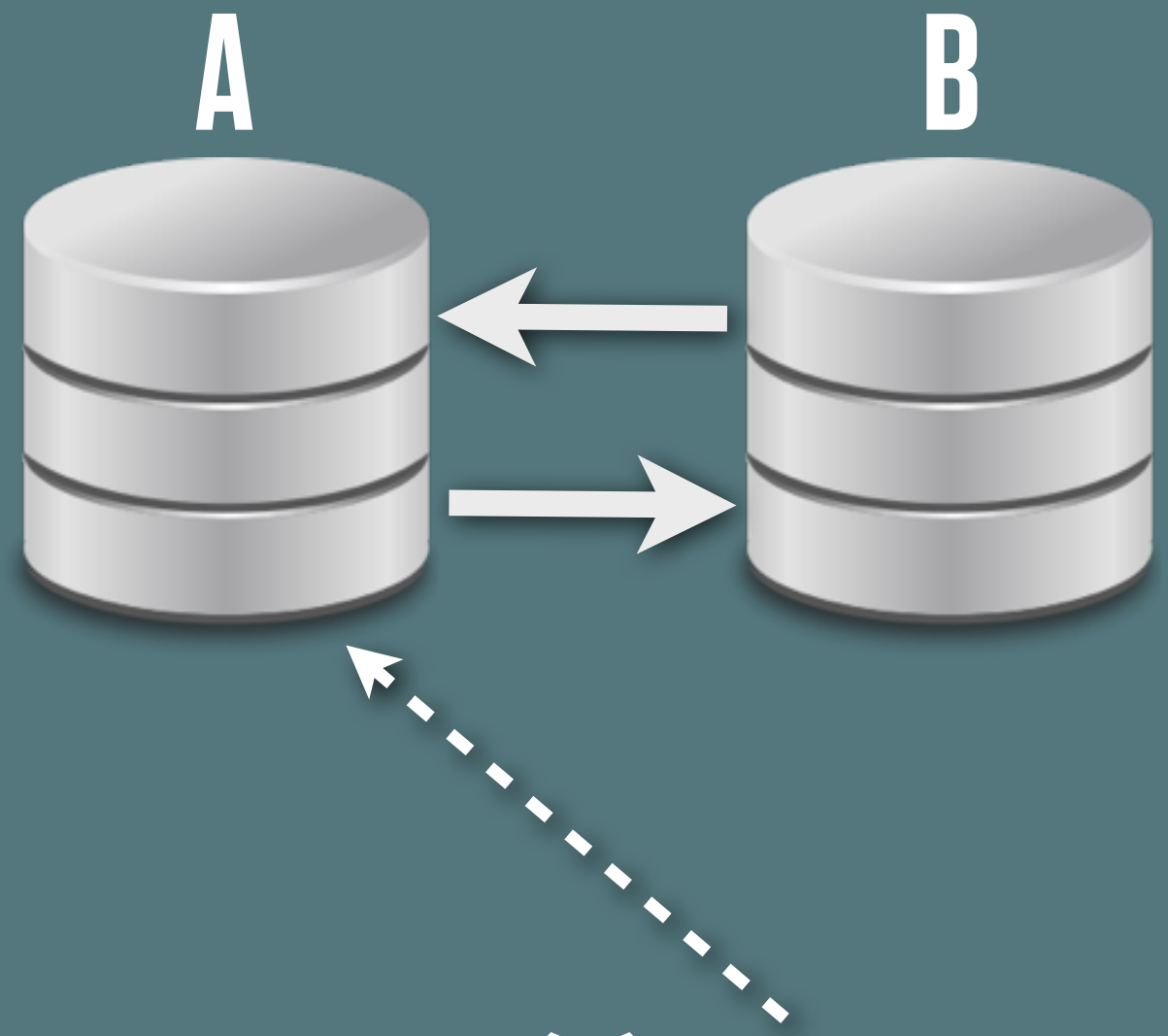
user_id : 500 % (# active replicants)

```
'etsy_index_A' => 'mysql:host=dbindex01.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_index_B' => 'mysql:host=dbindex02.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_shard_001_A' => 'mysql:host=dbshard01.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_001_B' => 'mysql:host=dbshard02.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_A' => 'mysql:host=dbshard03.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_B' => 'mysql:host=dbshard04.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_A' => 'mysql:host=dbshard05.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_B' => 'mysql:host=dbshard06.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
```

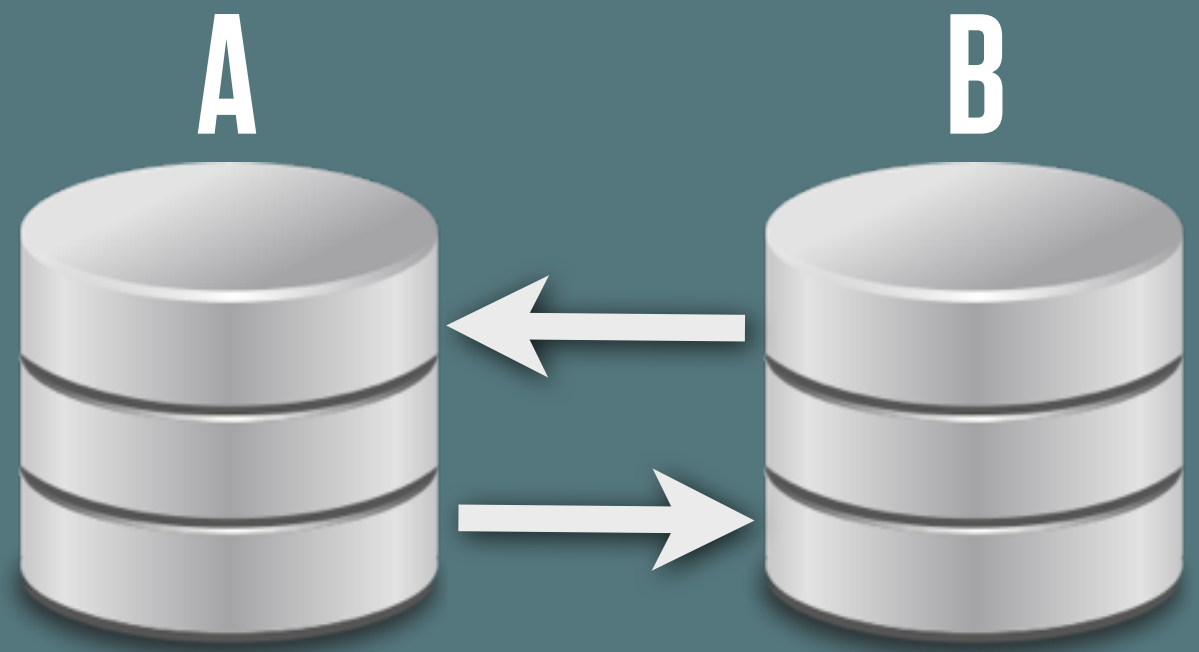
user_id : 500 % (# active replicants)



user_id : 500 % (2)



user_id : 500 % (2) == 0

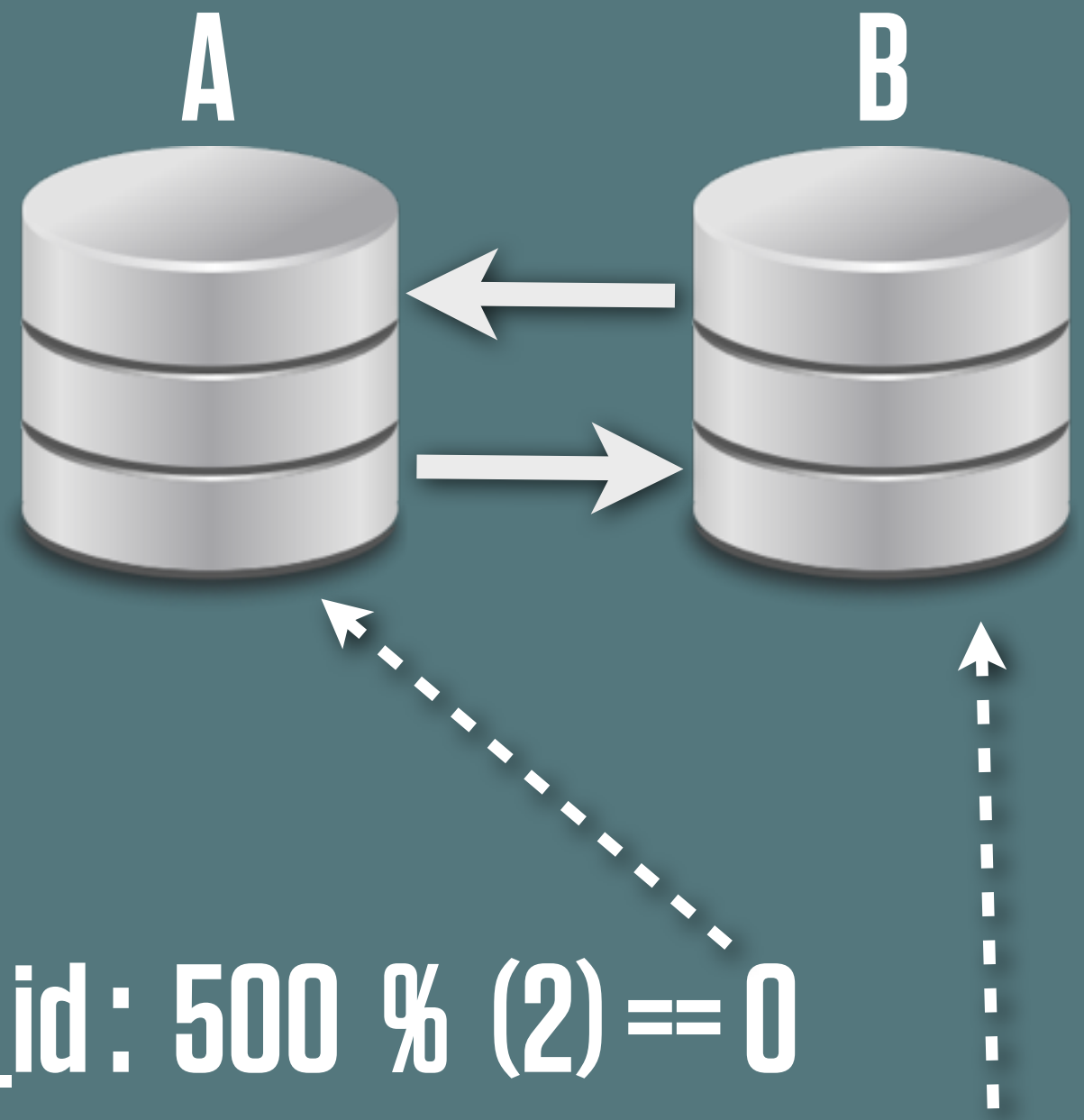


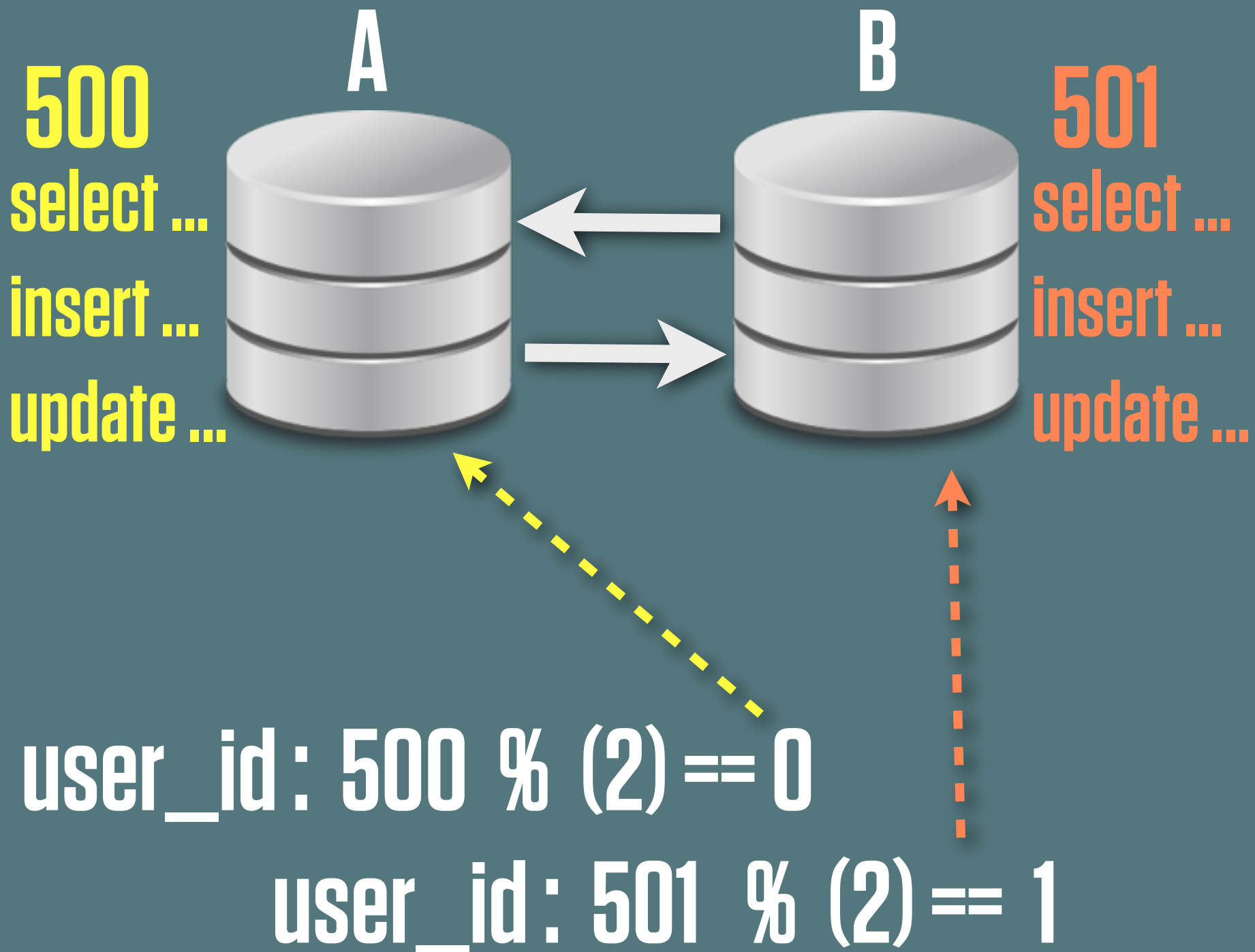
user_id : 500 % (2) == 0

select ...

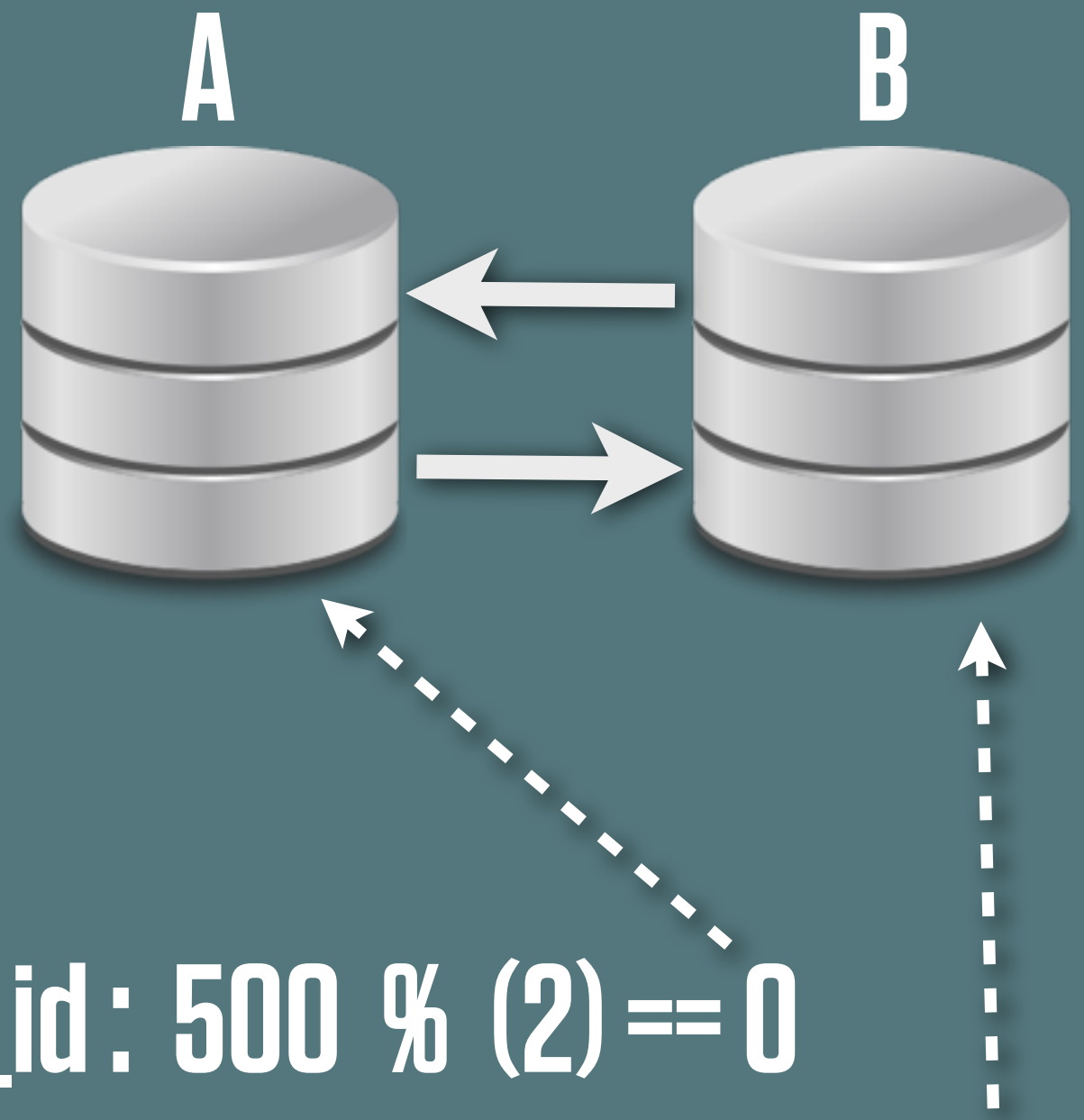
insert ...

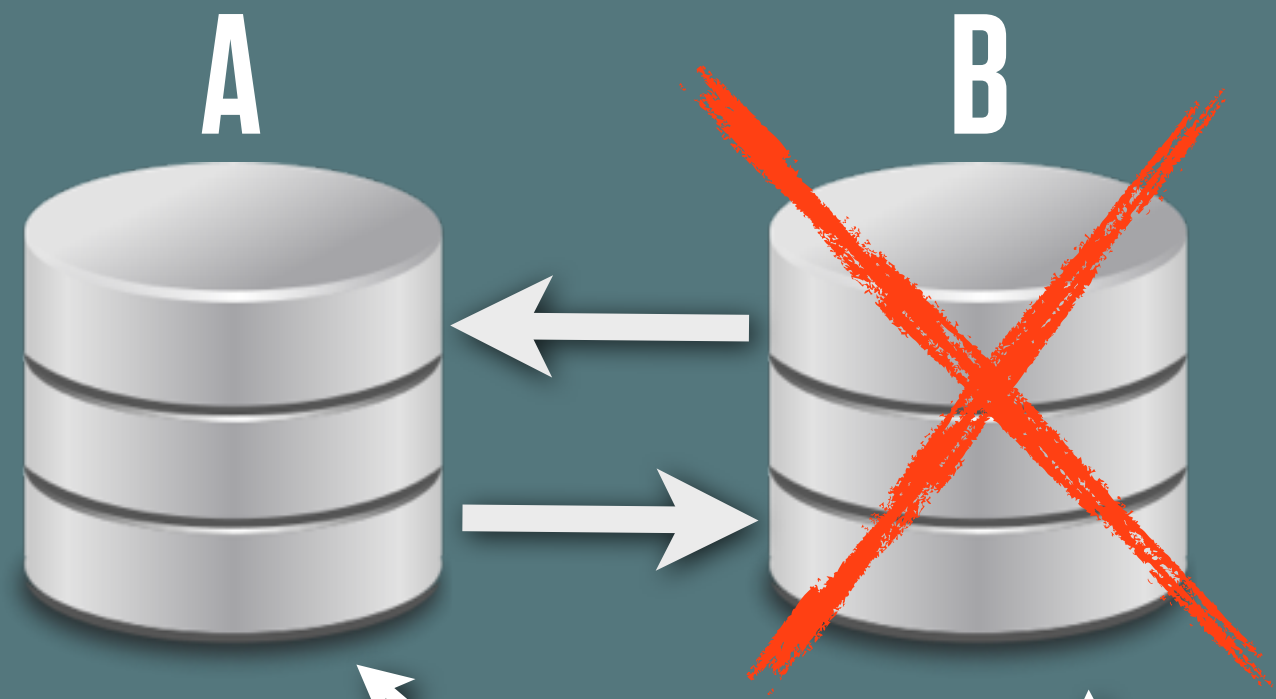
update ...


A
B
user_id : 500 % (2) == 0
user_id : 501 % (2) == 1


A
B
500
select ...
insert ...
update ...
501
select ...
insert ...
update ...
user_id : 500 % (2) == 0
user_id : 501 % (2) == 1

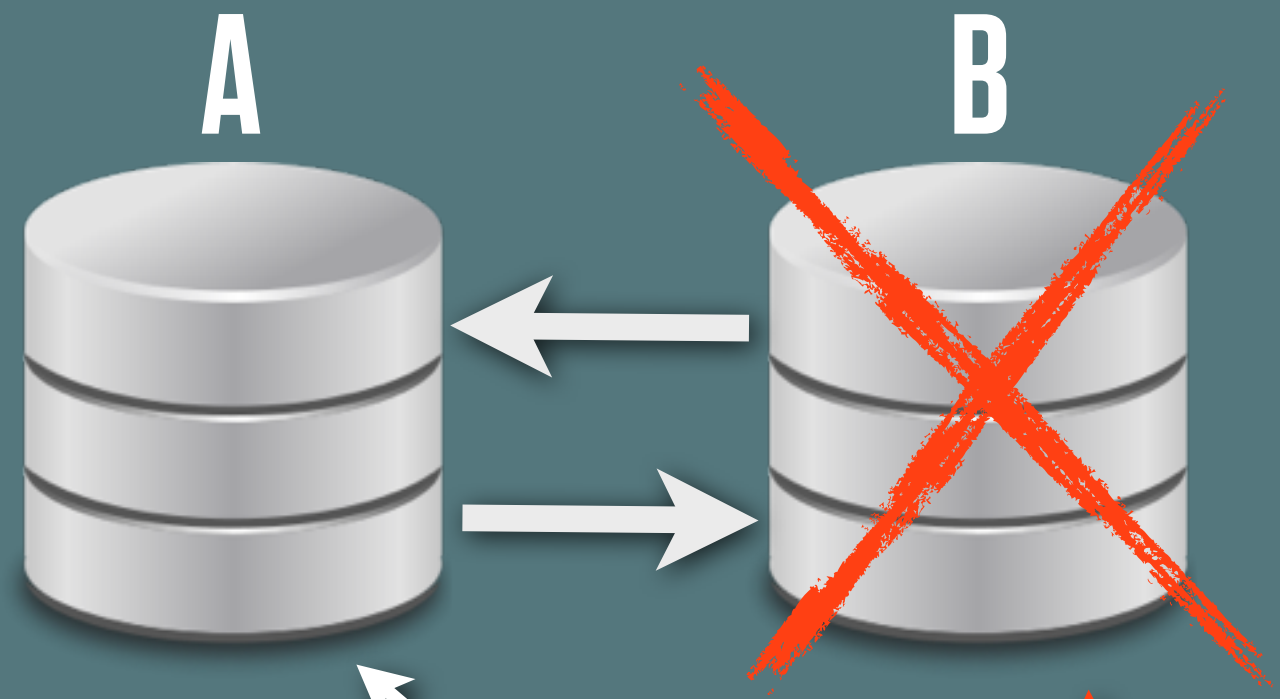

Failure


A
B
user_id : 500 % (2) == 0
user_id : 501 % (2) == 1



user_id : 500 % (2) == 0

user_id : 501 % (2) == 1



user_id : 500 % (2) == 0

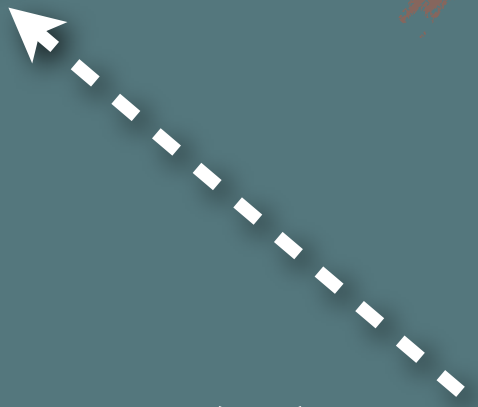
user_id : 501 % (2) == 1

```
'etsy_index_A' => 'mysql:host=dbindex01.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_index_B' => 'mysql:host=dbindex02.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_shard_001_A' => 'mysql:host=dbshard01.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_001_B' => 'mysql:host=dbshard02.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_A' => 'mysql:host=dbshard03.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_B' => 'mysql:host=dbshard04.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_A' => 'mysql:host=dbshard05.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_B' => 'mysql:host=dbshard06.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
```

user_id : 500 % (2) == 0

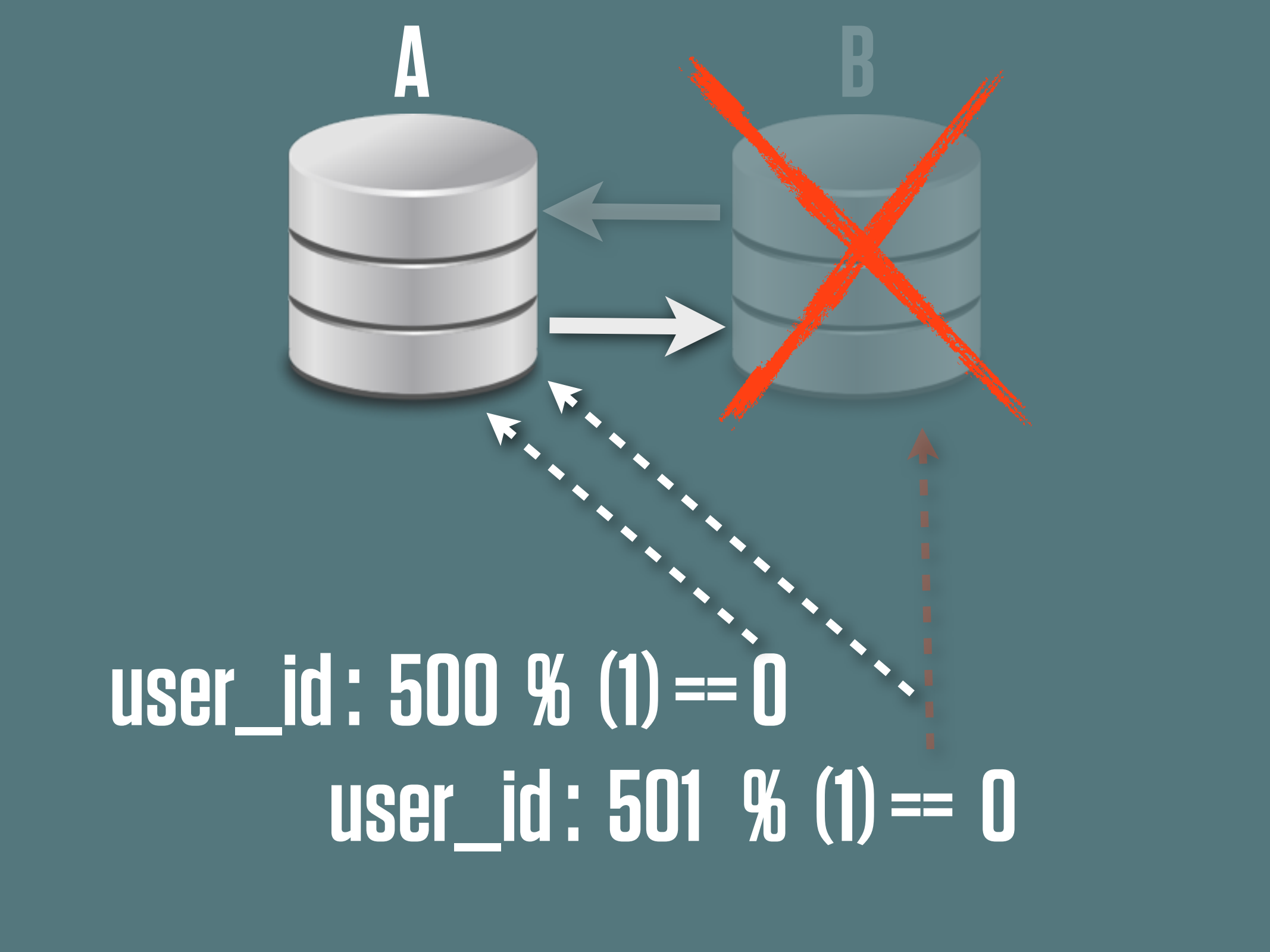
user_id : 501 % (2) == 1

'etsy_index_A' => 'mysql:host=dbindex01.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_index_B' => 'mysql:host=dbindex02.ny4.etsy.com;port=3306;dbname=etsy_index;user=etsy_rw',
'etsy_shard_001_A' => 'mysql:host=dbshard01.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
~~'etsy_shard_001_B' => 'mysql:host=dbshard02.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',~~
'etsy_shard_002_A' => 'mysql:host=dbshard03.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_002_B' => 'mysql:host=dbshard04.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_A' => 'mysql:host=dbshard05.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',
'etsy_shard_003_B' => 'mysql:host=dbshard06.ny4.etsy.com;port=3306;dbname=etsy_shard;user=etsy_rw',



user_id : 500 % (2) == 0

user_id : 501 % (2) == 1


A
B
user_id : 500 % (1) == 0
user_id : 501 % (1) == 0

ORM

connection handling

shard lookup

replicant selection

CRUD

cache handling

data validation

data abstraction

Shard Selection

Non-Writable Shards

```
$config["non_writable_shards"] = array(1, 2, 3, 4);
```

```
  public static function getKnownWritableShards(){
    return array_values(
      array_diff(
        self::getKnownShards(),
        self::getNonwritableShards()
    ));
  }
```

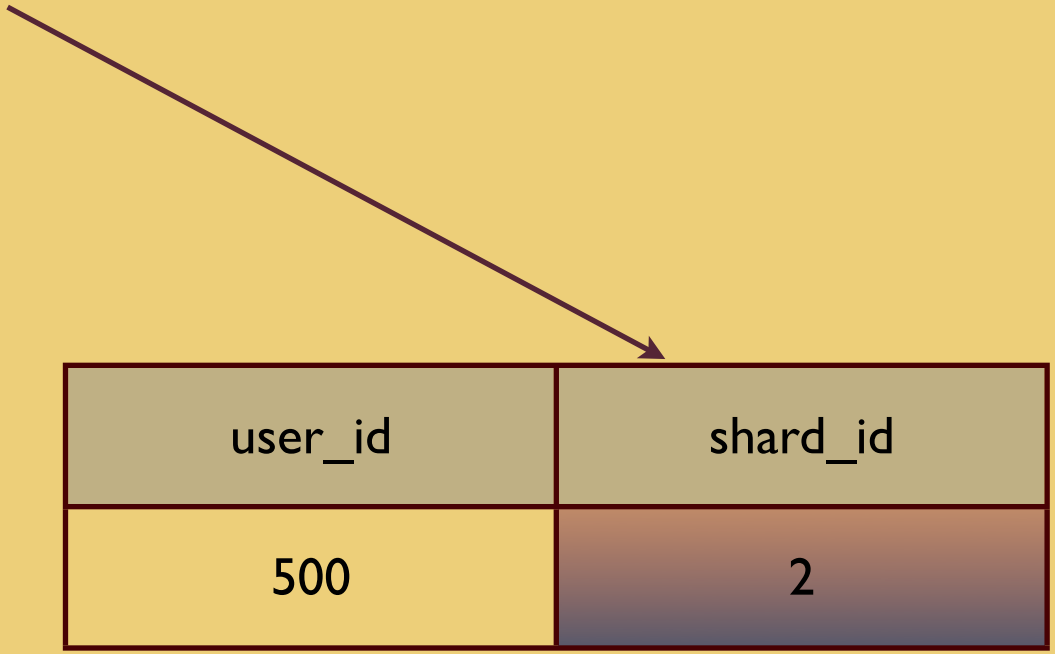
Initial Selection

```
$shards = EtsyORM::getKnownWritableShards();

$user_shard = $shards[rand(0, count($shards) - 1)];
```

user_id	shard_id
500	

Initial Selection

```
$shards = EtsyORM::getKnownWritableShards();

$user_shard = $shards[rand(0, count($shards) - 1)];
```



Later....

select shard_id from user_index
where user_id = X


Variants

shard 1

user_id	group_id
1	A
1	B
2	A
2	C

shard 2

user_id	group_id
3	A
3	B
4	A
5	C

SELECT user_id FROM users_groups WHERE group_id = 'A'

shard 1

user_id	group_id
1	A
1	B
2	A
2	C

shard 2

user_id	group_id
3	A
3	B
4	A
5	C

SELECT user_id FROM users_groups WHERE group_id = 'A'

Broken!

shard 1

user_id	group_id
1	A
1	B
2	A
2	C

JOIN?

shard 2

user_id	group_id
3	A
3	B
4	A
5	C

SELECT user_id FROM users_groups WHERE group_id = 'A'

Broken!

shard 1

user_id	group_id
1	A
1	B
2	A
2	C

shard 2

user_id	group_id
3	A
3	B
4	A
5	C

JOIN?

SELECT user_id FROM users_groups WHERE group_id = 'A'

Broken!

users_groups

user_id	group_id
1	A
1	B
2	A
2	C
3	A
3	B
3	C

groups_users

group_id	user_id
A	1
A	3
A	2
B	3
B	1
C	2
C	3

index

users_groups_index

user_id	shard_id
1	1
2	1
3	2
4	3

groups_users_index

group_id	shard_id
A	1
B	2
C	2
D	3

separate indexes for different slices of data

index

users_groups_index

user_id	shard_id
1	1
2	1
3	2
4	3

groups_users_index

group_id	shard_id
A	1
B	2
C	2
D	3

shard 3

user_id	group_id
4	A
4	B
4	C
4	D

Schema Changes


shard 1
shard 2
shard N


shard 1
shard 2
shard N

Schemanator

Shard Config Status

Last updated:
Sat, 31 Mar 12 16:50:26 -0400

Etsy_aux

A	B

Etsy_index

A	B

Tickets

A	B

Etsy_shard

001_A	001_B
002_A	002_B
003_A	003_B
004_A	004_B
005_A	005_B
006_A	006_B
007_A	007_B
008_A	008_B
009_A	009_B
010_A	010_B
011_A	011_B
012_A	012_B
013_A	013_B
014_A	014_B
015_A	015_B
016_A	016_B

Side Splitter

Locked by: jgoulah Clear Lock

It helps you change database schemas in three steps.

1. Platform
   etsy_shard

2. Sides
Operating on etsy_shard.

3. Deploy

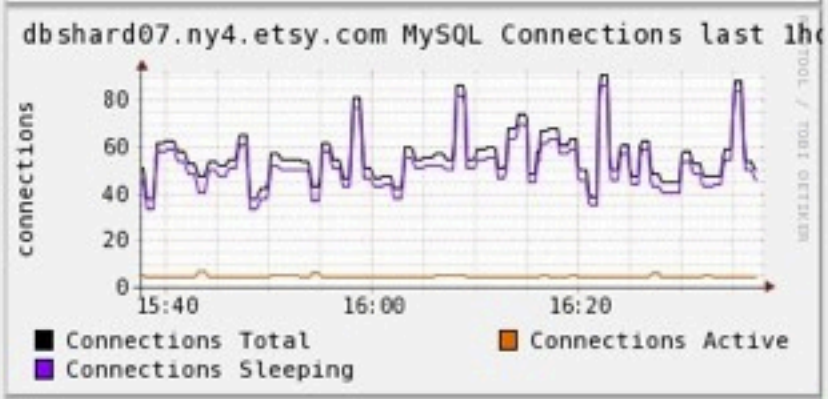
etsy_shard status:

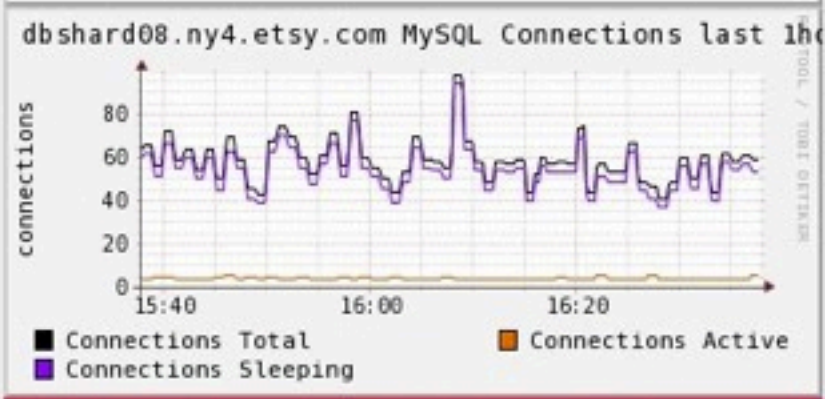
No sides are out.

Clear all checked sides | Pick all side A | Pick all side B

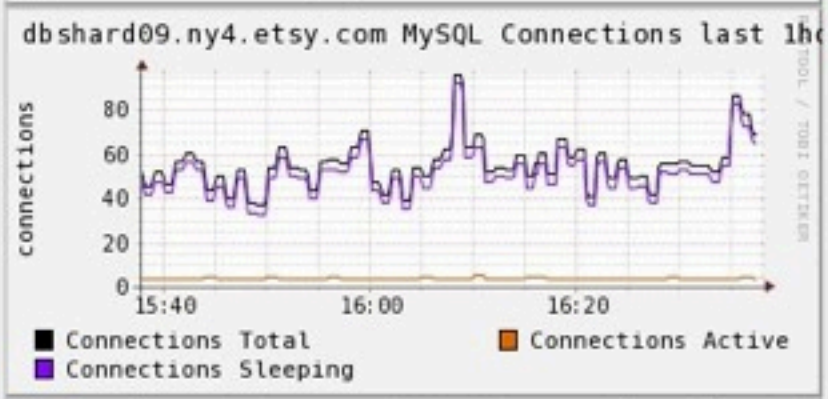
#	Side A	Side B
1	☐ etsy_shard_001_A	☐ etsy_shard_001_B
2	☐ etsy_shard_002_A	☐ etsy_shard_002_B
3	☐ etsy_shard_003_A	☐ etsy_shard_003_B
4	☐ etsy_shard_004_A	☐ etsy_shard_004_B
5	☐ etsy_shard_005_A	☐ etsy_shard_005_B
6	☐ etsy_shard_006_A	☐ etsy_shard_006_B
7	☐ etsy_shard_007_A	☐ etsy_shard_007_B
8	☐ etsy_shard_008_A	☐ etsy_shard_008_B
9	☐ etsy_shard_009_A	☐ etsy_shard_009_B
10	☐ etsy_shard_010_A	☐ etsy_shard_010_B
11	☐ etsy_shard_011_A	☐ etsy_shard_011_B
12	☐ etsy_shard_012_A	☐ etsy_shard_012_B
13	☐ etsy_shard_013_A	☐ etsy_shard_013_B
14	☐ etsy_shard_014_A	☐ etsy_shard_014_B
15	☐ etsy_shard_015_A	☐ etsy_shard_015_B
16	☐ etsy_shard_016_A	☐ etsy_shard_016_B
17	☐ etsy_shard_017_A	☐ etsy_shard_017_B

shard004

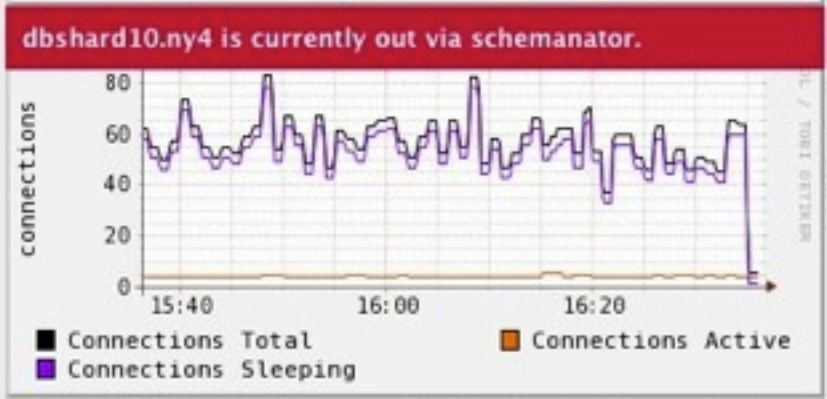




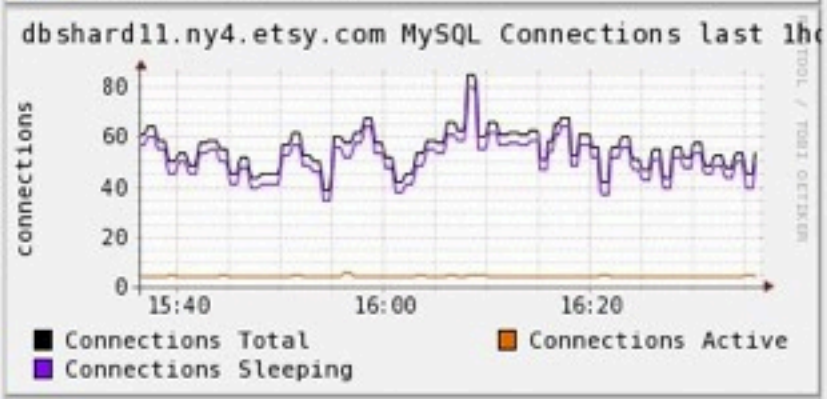
shard005

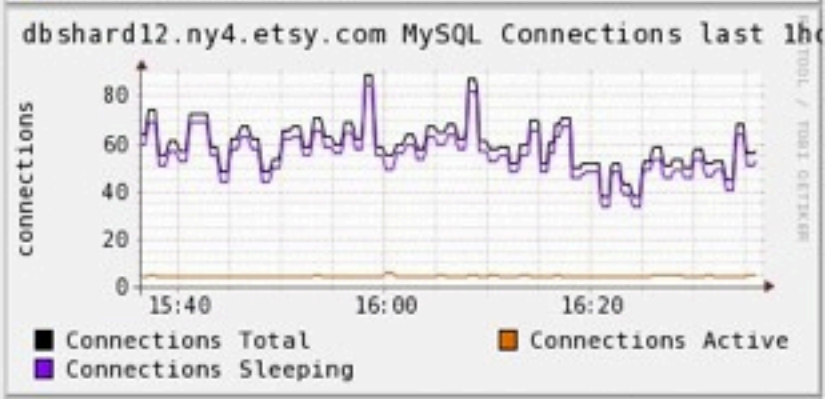


dbshard10.ny4 is currently out via schemanator.

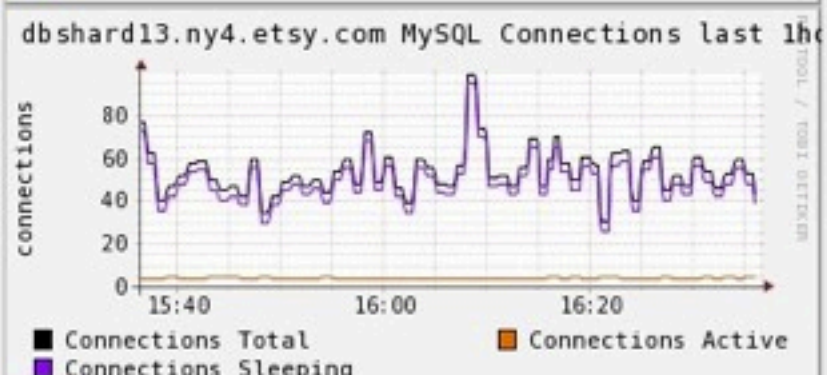


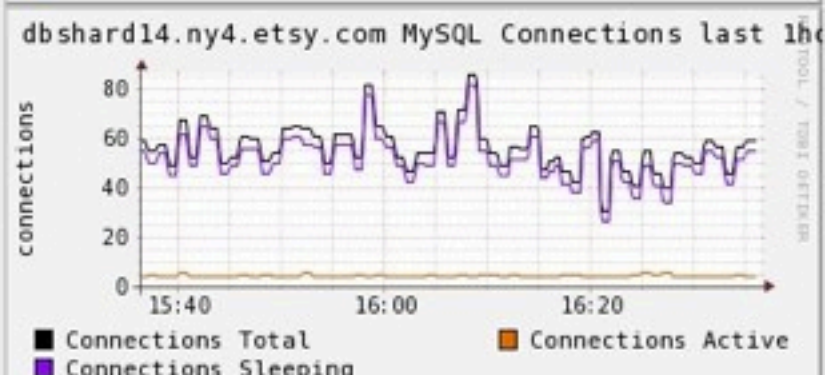
shard006



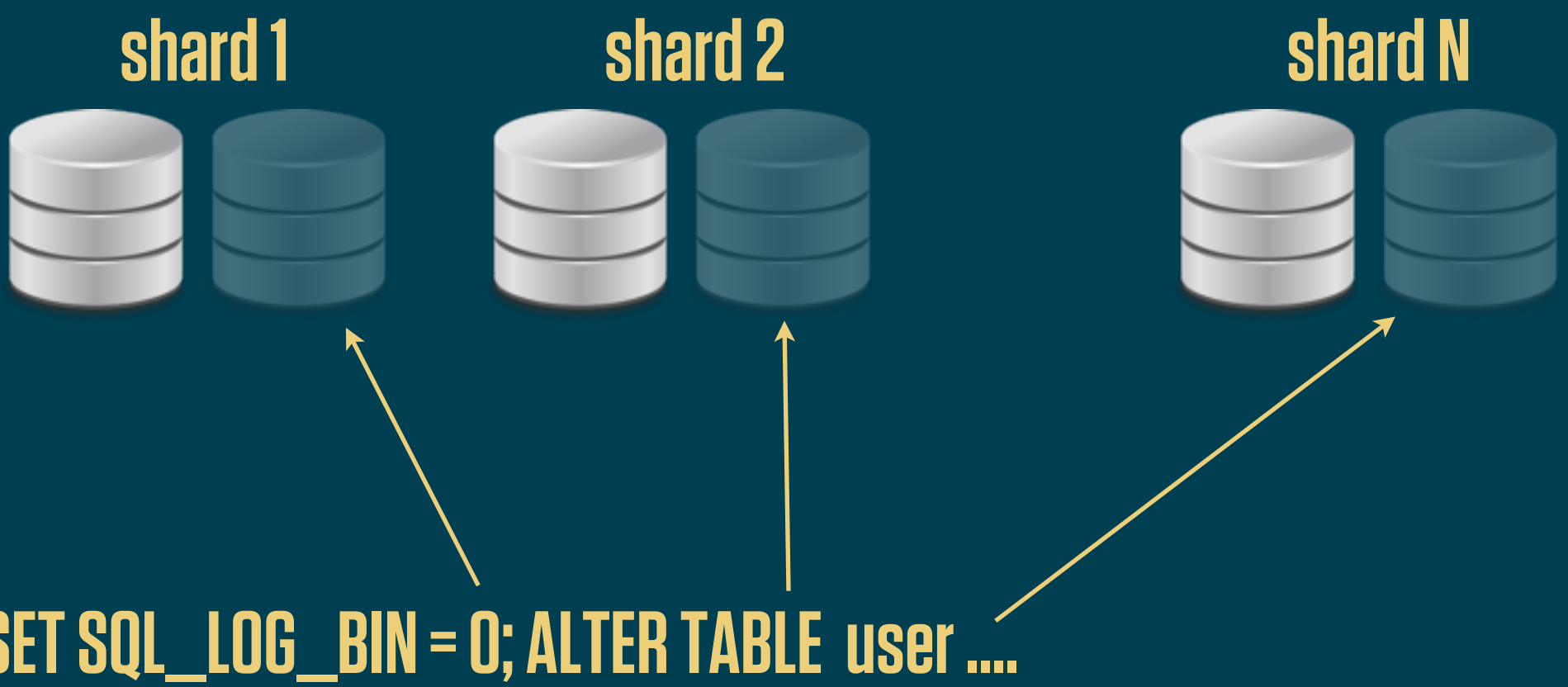


shard007






shard 1
shard 2
shard N


shard 1
shard 2
shard N
SET SQL_LOG_BIN = 0; ALTER TABLE user ....

shard migration

Why?

Prevent disk from filling

Prevent disk from filling

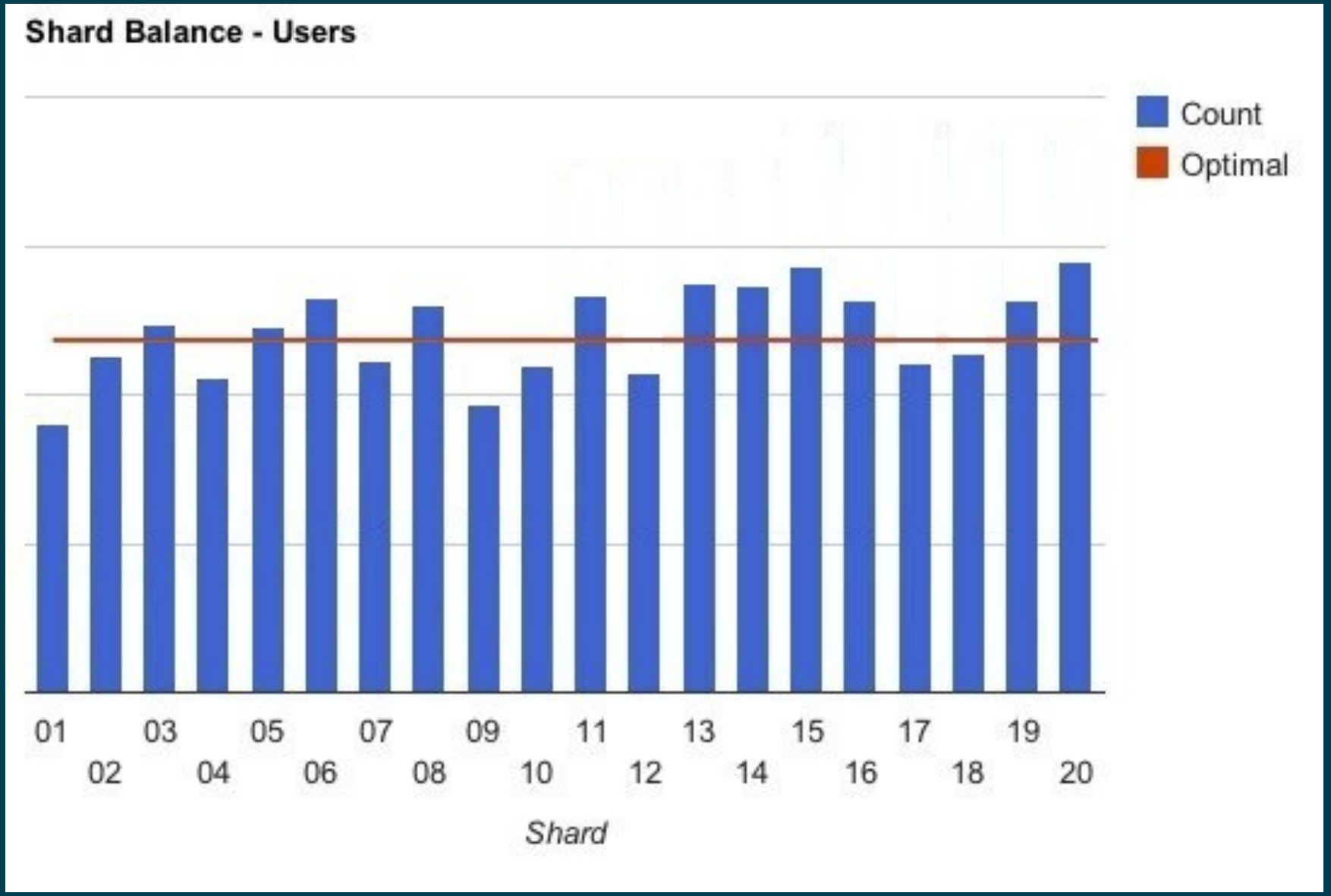
High traffic objects (shops, users)

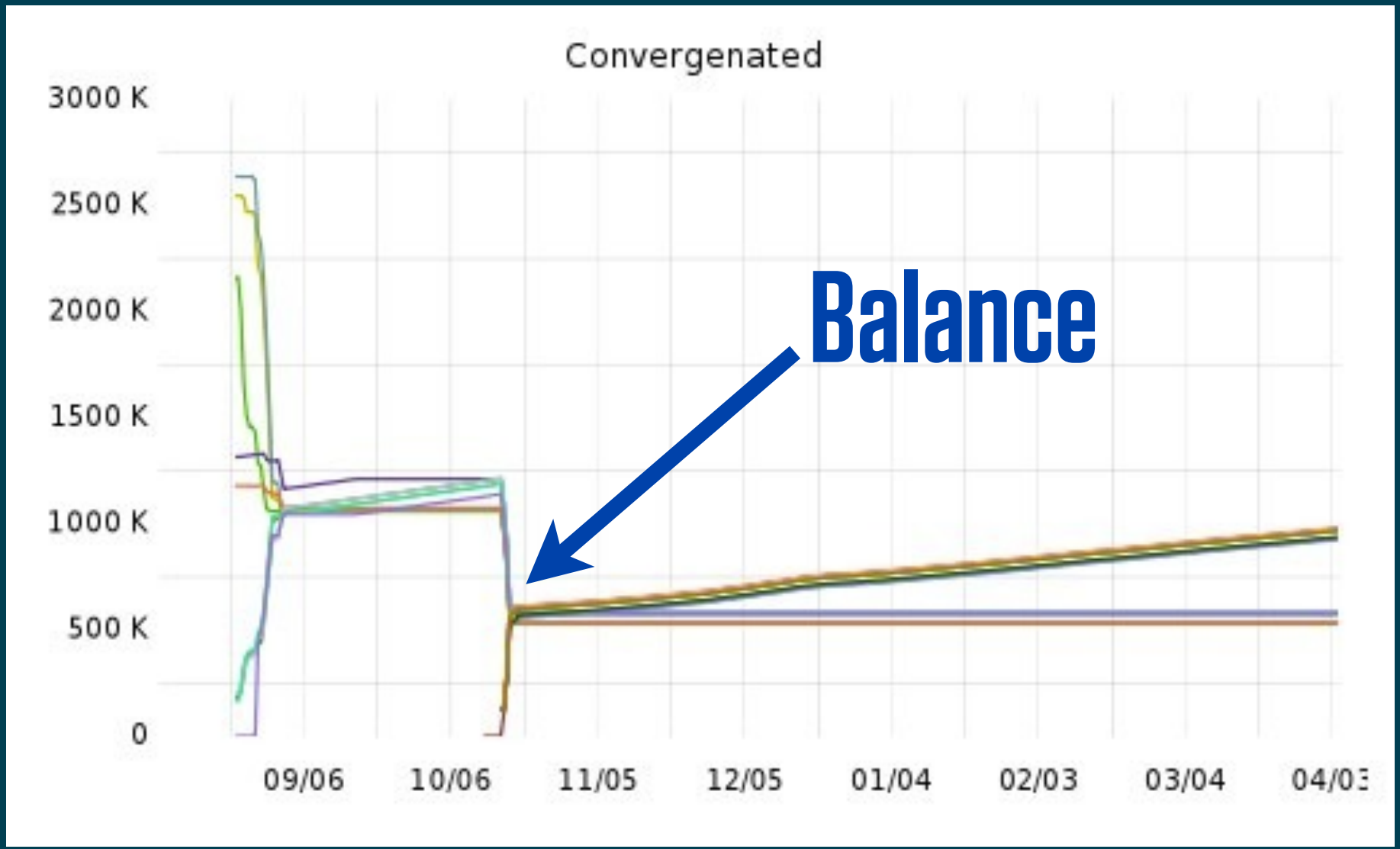
Prevent disk from filling

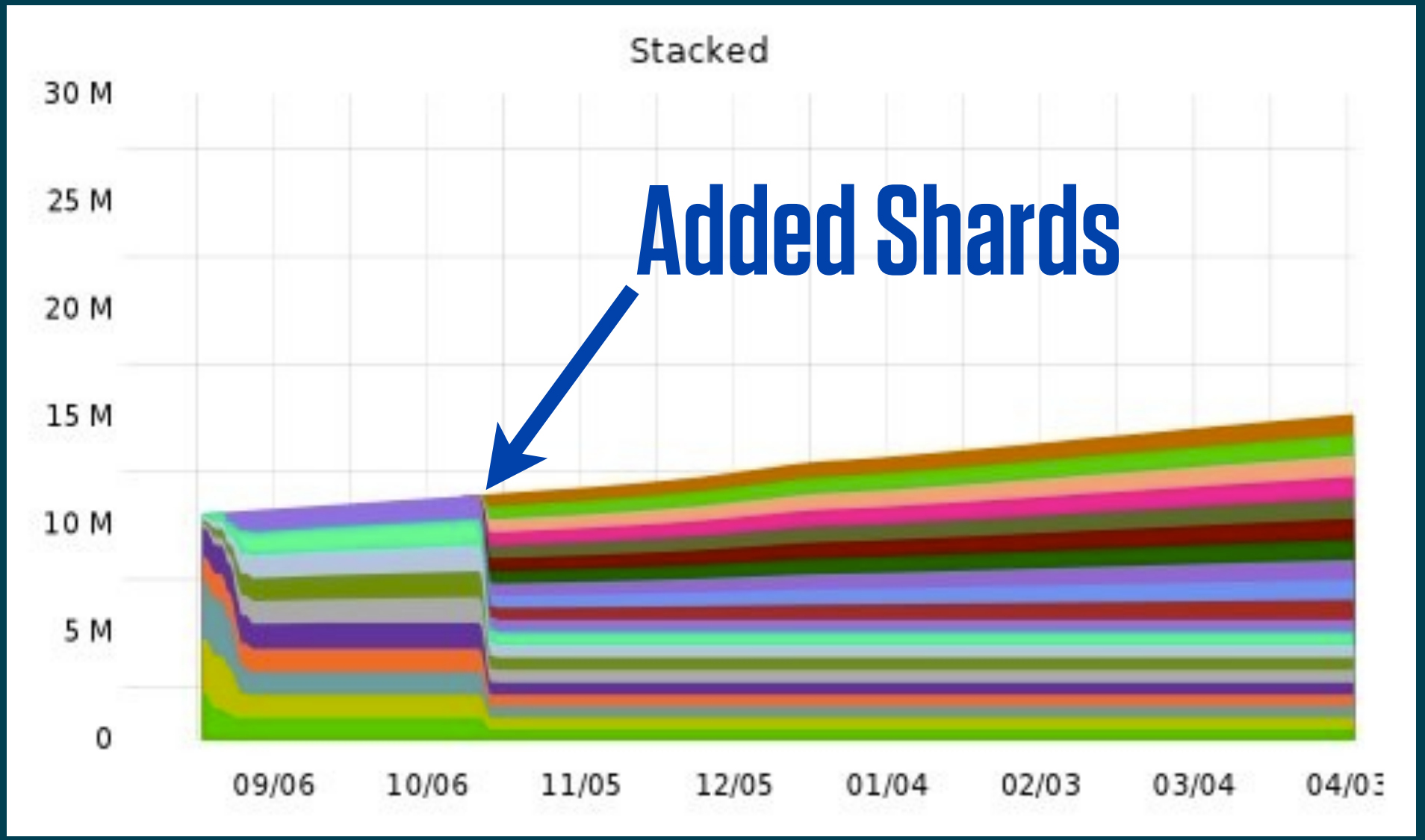
High traffic objects (shops, users)

Shard rebalancing

When?


Shard Balance - Users
Count
Optimal
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
Shard


Convergenated
3000 K
2500 K
2000 K
1500 K
1000 K
500 K
0
09/06
10/06
11/05
12/05
01/04
02/03
03/04
04/03
Balance


Stacked
30 M
25 M
20 M
15 M
10 M
5 M
0
Added Shards
09/06
10/06
11/05
12/05
01/04
02/03
03/04
04/03

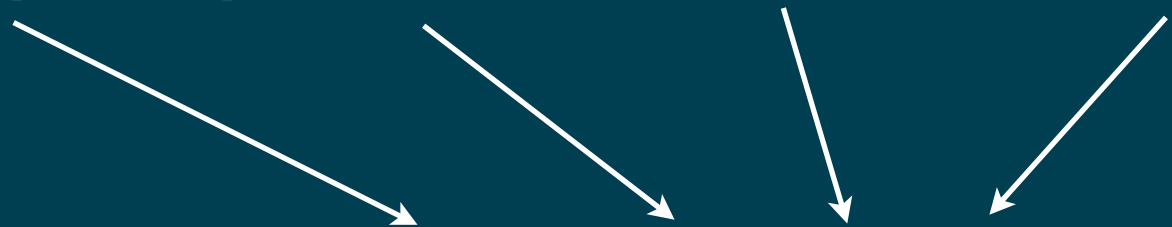
per object migration

<object type> <object id> <shard>

```
# migrate_object User 5307827 2
```

percentage migration

<object type> <percent> <old shard> <new shard>



```
# migrate_pct User 25 3 6
```

index

user_id	shard_id	migration_lock	old_shard_id
1	1	0	0

shard 1

shard 2

shard N

index

user_id	shard_id	migration_lock	old_shard_id
1	1	1	0

- Lock

shard 1

shard 2

shard N

index

user_id	shard_id	migration_lock	old_shard_id
1	1	1	0

- Lock
- Migrate

shard 1

shard 2

shard N

index

user_id	shard_id	migration_lock	old_shard_id
1	1	1	0

- Lock
- Migrate
- Checksum

shard 1

shard 2

shard N

index

user_id	shard_id	migration_lock	old_shard_id
1	1	1	0

- Lock
- Migrate
- Checksum

shard 1

shard 2

shard N

index

user_id	shard_id	migration_lock	old_shard_id
1	2	0	1

- Lock
- Migrate
- Checksum
- Unlock

shard 1

shard 2

shard N

index

user_id	shard_id	migration_lock	old_shard_id
1	2	0	1

- Lock
- Migrate
- Checksum
- Unlock
- Delete (from old shard)

shard 1

shard 2

shard N

Usage Patterns

Arbitrary Key Hash

tag1	tag2	co_occurrence _count
“red”	“cloth”	666

tag1	tag2	shard_id
“red”	“cloth”	1
“vintage”	“doll”	3
“antique”	“radio”	5
“gift”	“vinyl”	2
“toy”	“car”	1
“wool”	“felt”	2
“floral”	“wreath”	5
“wood”	“table”	8
“box”	“wood”	4
“doll”	“happy”	5
“smile”	“clown”	3
“radio”	“vintage”	10
“blue”	“luggage”	8
“shoes”	“green”	12
...	...	...

OR

hash_bucket	shard_id
1	2
2	3
3	1
4	2
5	3

1. provide some key

1. provide some key
2. compute corresponding hash bucket

1. provide some key
2. compute corresponding hash bucket
3. lookup hash bucket on index to find shard

1,000,000 'buckets' each with a row in arbitrary_key_index which points to a shard

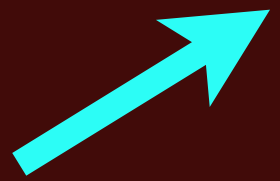
hash_bucket	shard_id
1	2
2	3
3	1
4	2
5	3

hash_bucket == hash('red', 'cloth') % BUCKETS

1,000,000 'buckets' each with a row in arbitrary_key_index which points to a shard

hash_bucket	shard_id
1	2
2	3
3	1
4	2
5	3

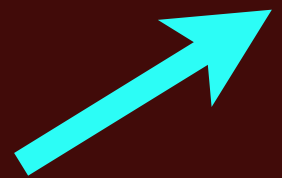
hash_bucket == hash('red', 'cloth') % BUCKETS



1,000,000 'buckets' each with a row in arbitrary_key_index which points to a shard

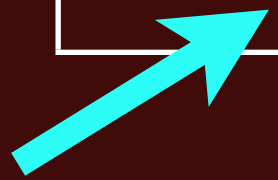
hash_bucket	shard_id
1	2
2	3
3	1
4	2
5	3

hash_bucket == hash('red', 'cloth') % BUCKETS



1,000,000 'buckets' each with a row in arbitrary_key_index which points to a shard

hash_bucket	shard_id
1	2
2	3
3	1
4	2
5	3



hash_bucket == hash('red', 'cloth') % BUCKETS

Partitions

```
PARTITION BY RANGE (reference_timestamp)(
  PARTITION P5 VALUES LESS THAN (1317441600),
  PARTITION P6 VALUES LESS THAN (1320120000),
  PARTITION P7 VALUES LESS THAN (1322715600),
  PARTITION P8 VALUES LESS THAN (1325394000));
```

Deleting a large partition:
few hours, tons of disk IO

Deleting a large partition:

few hours, tons of disk IO

Dropping a 2G partition with 2M rows :

Deleting a large partition:

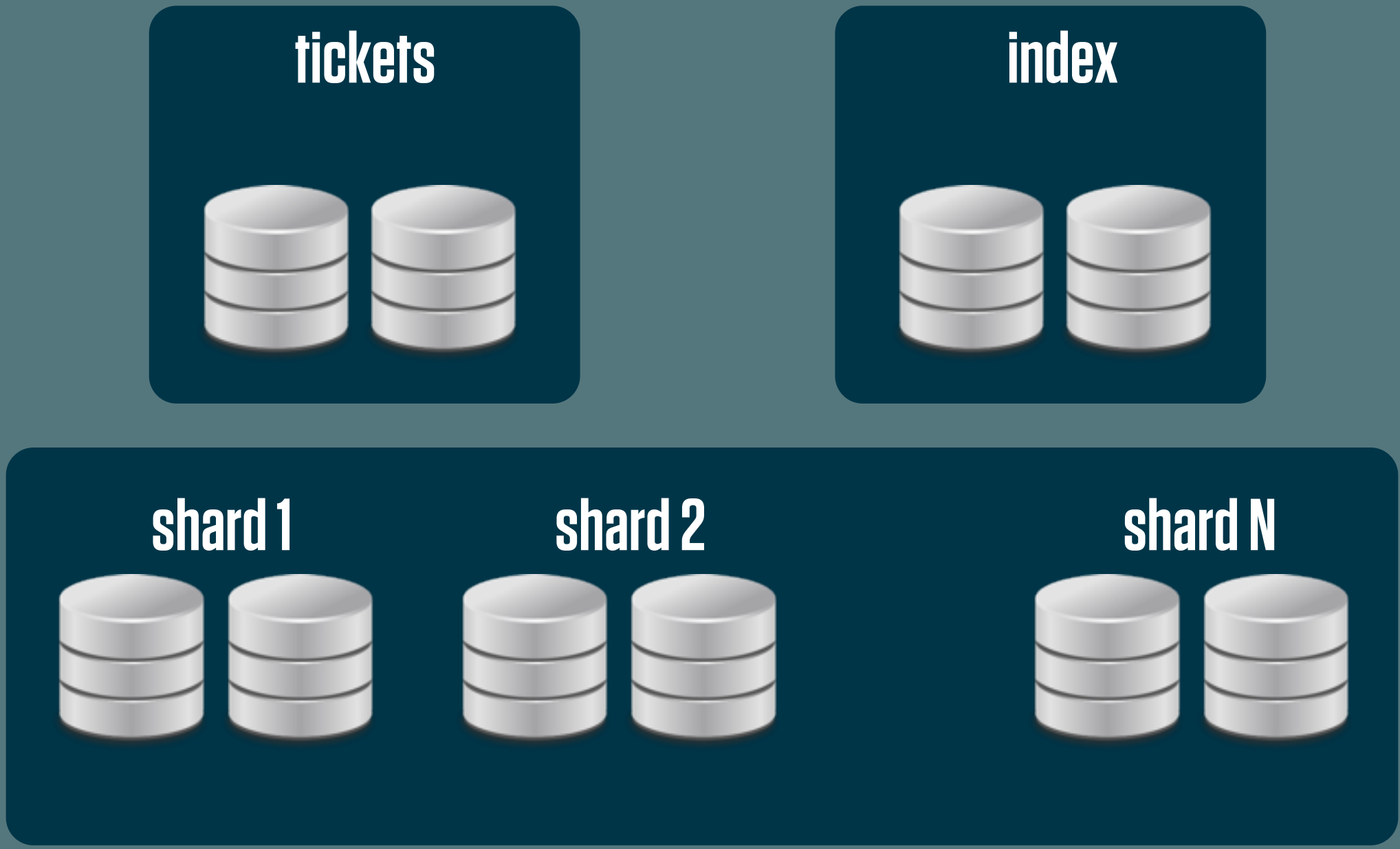
few hours, tons of disk IO

Dropping a 2G partition with 2M rows :

< 1s

```
# file= "shop_stats_syndication_hourly#P#P13458672OO.ibd"
# In $file  $file.remove"
```

```
# file= "shop_stats_syndication_hourly#P#P1345867200.ibd"
# ln $file  $file.remove"


# stat "shop_stats_syndication_hourly#P#P1345867200.ibd"
  File: `shop_stats_syndication_hourly#P#P1345867200.ibd'
  Size: 65536         Blocks: 136        IO Block: 4096   regular file
Device: 6804h/26628d    Inode: 41321[illegible]63    Links: 2
Access: (0660/-rw-rw----)  Uid: (  104/   mysql)   Gid: (  106/   mysql)
```


tickets
index
shard 1
shard 2
shard N

Thank you

etsy.com/jobs