



Introduction to Tungsten Replicator

Neil Armitage, Cluster implementation Engineer, Continuent

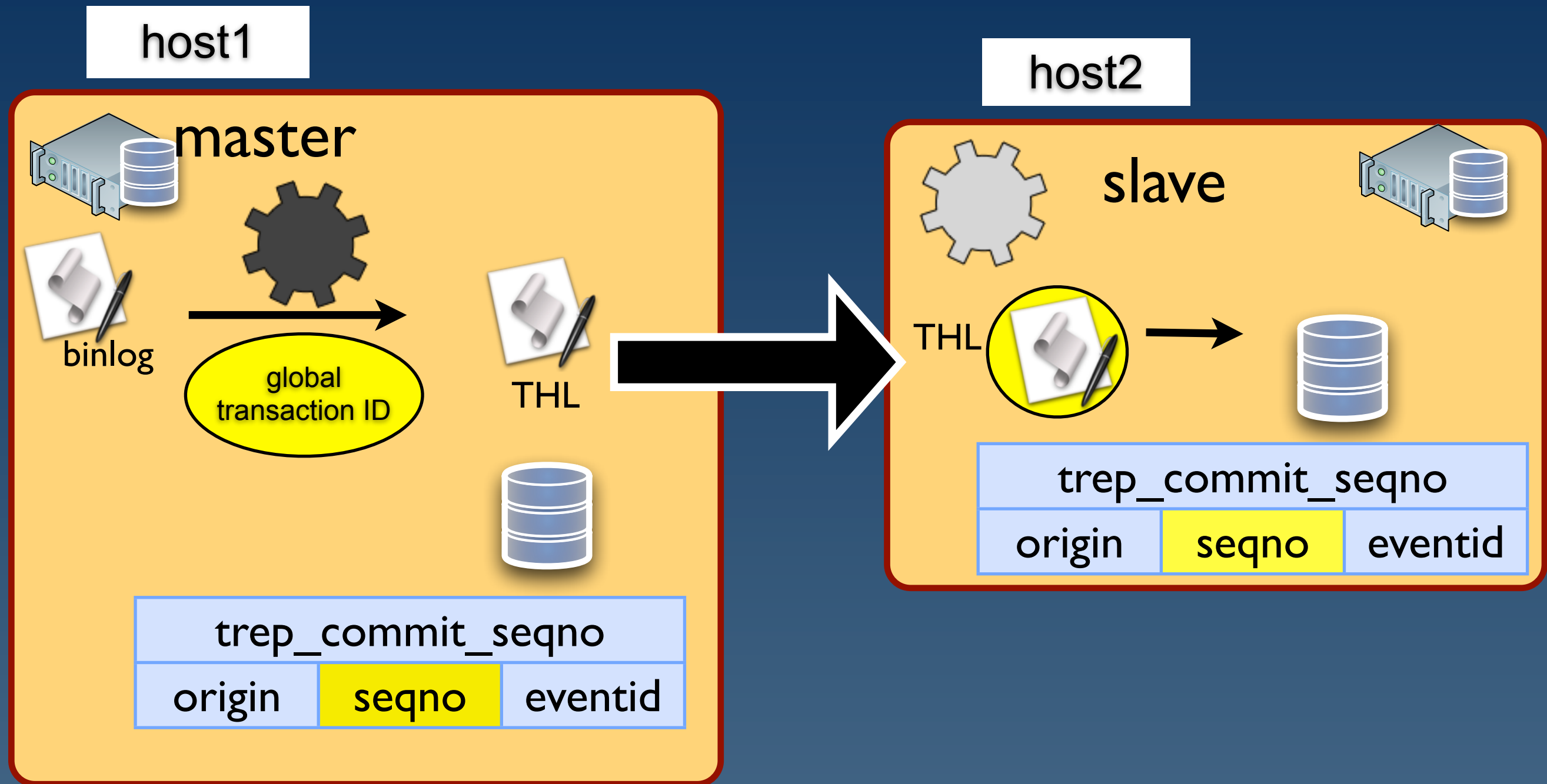
What is Tungsten replicator?

- Drop in replacement for native MySQL replication.
- Fully OpenSource under GPL v2
- Heterogeneous Replication
- Flexible topologies
- Parallel replication

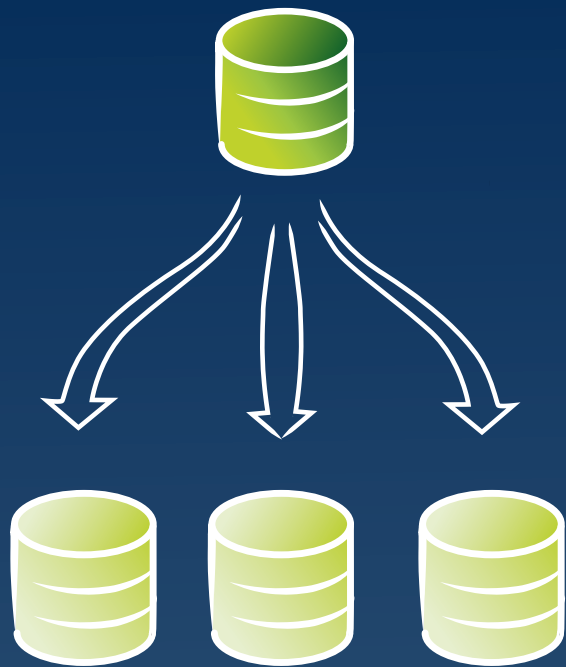
Tungsten replicator

- Global transaction ID
- Multiple masters
- Multiple sources (Fan In)

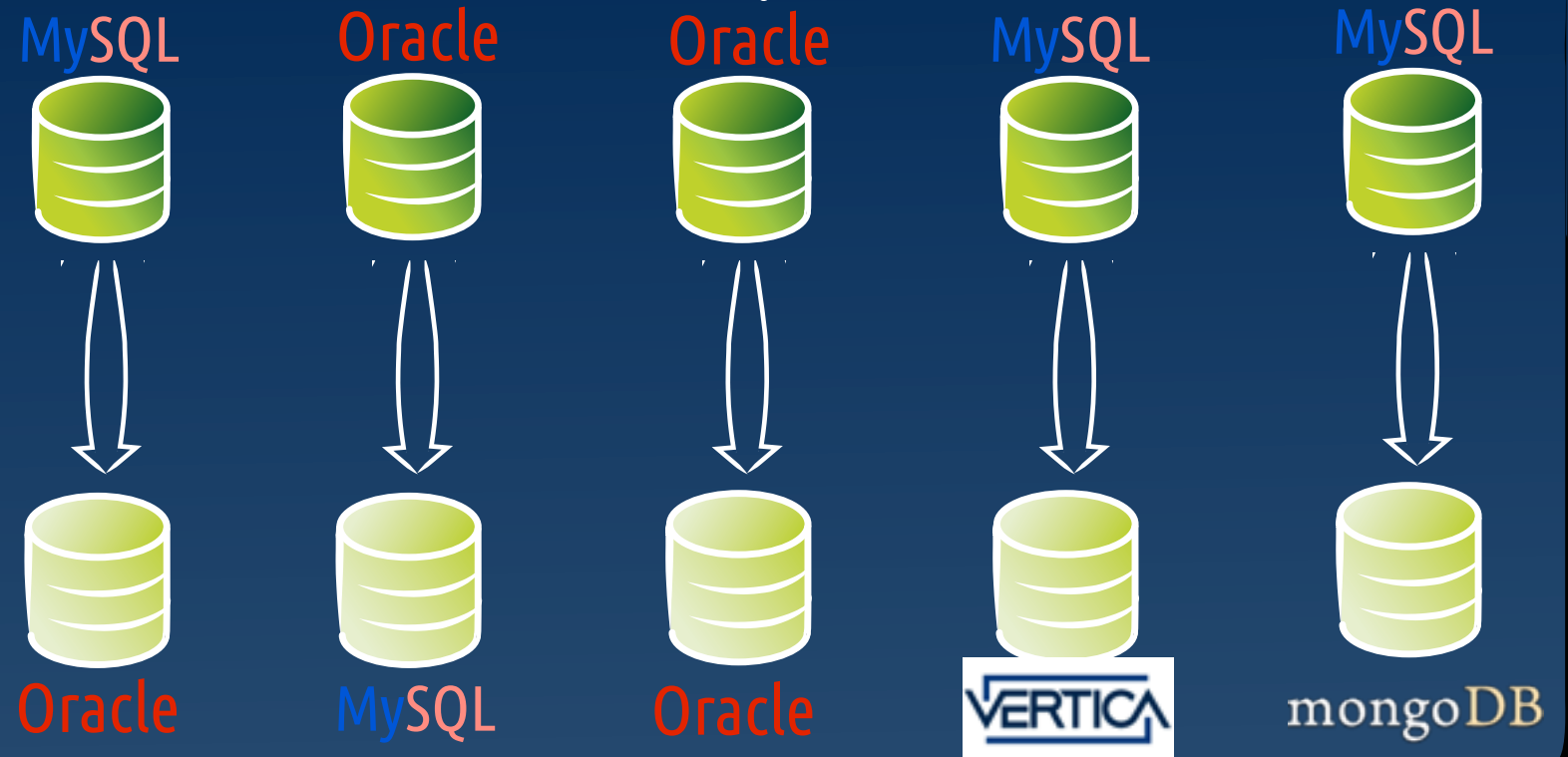
Tungsten Replicator in a nutshell



master-slave



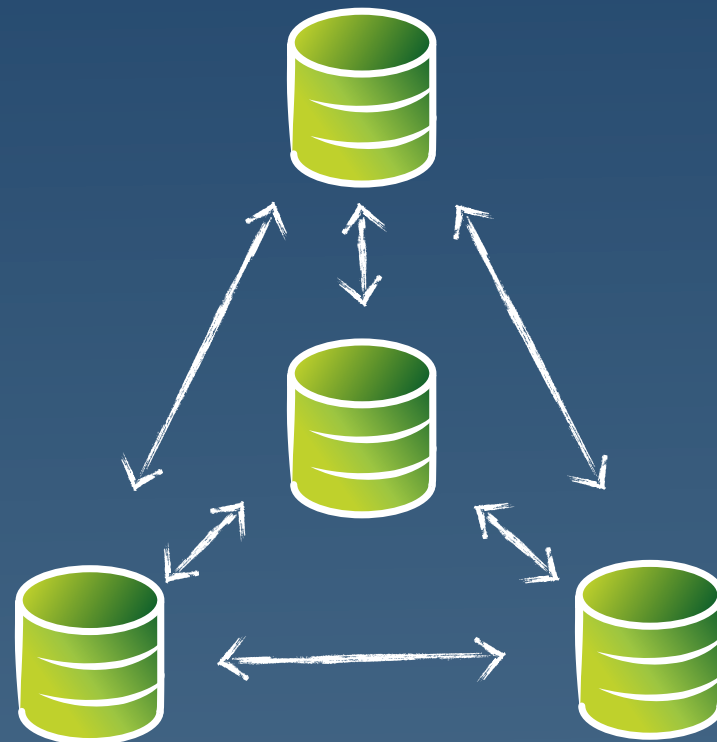
Heterogeneous



fan-in slave



all-masters



star-schema



Why Use Tungsten Replicator?

A failover scenario

I: MySQL native replication

1. one Master, two slaves

- Loading the “employees” test database

2. Master goes away

- * Stop replication
- * Slaves are updated at different levels

```
# 2  
select count(*) from titles  
333,145
```

```
# 3  
select count(*) from titles  
443,308
```

3. Look into Slave #2 binary logs

- find the last transaction

4. Look into Slave #3 binary logs

1. find the transaction that was last in slave #2
2. Recognize that last transaction in the log of slave #3 (This can actually take you a LOOOOONG TIME)
3. Get the position immediately after this transaction
4. (e.g. 134000 in file mysql-bin.000018)

5. promote Slave #3 to master

* in slave #2

```
CHANGE MASTER TO  
master_host='slave_3_IP',  
master_user='slavename',  
master_password='slavepassword',  
master_log_file='mysql-bin.000018',  
master_log_pos=134000;
```

A failover scenario

II: Tungsten Replicator

1. one master, two slaves

- loading the 'employees' test database

2. Master goes away

- * Stop replication
- * Slaves are updated at different levels

```
# 2  
select count(*) from titles  
333,145
```

```
# 3  
select count(*) from titles  
443,308
```

3. no need to find the last transaction

simply change roles

```
trepctl -host slave3 setrole -role master
```

```
trepctl -host slave2 setrole \  
-role slave -uri th1://slave3
```

```
trepctl -host slave3 online  
State: ONLINE
```

```
trepctl -host slave2 online  
State: GOING-ONLINE:SYNCHRONIZING
```

4. Check that the slave has synchronized

```
# new master  
select seqno from tungsten.trep_commit_seqno;  
78
```

```
# new slave  
select seqno from tungsten.trep_commit_seqno;  
64
```

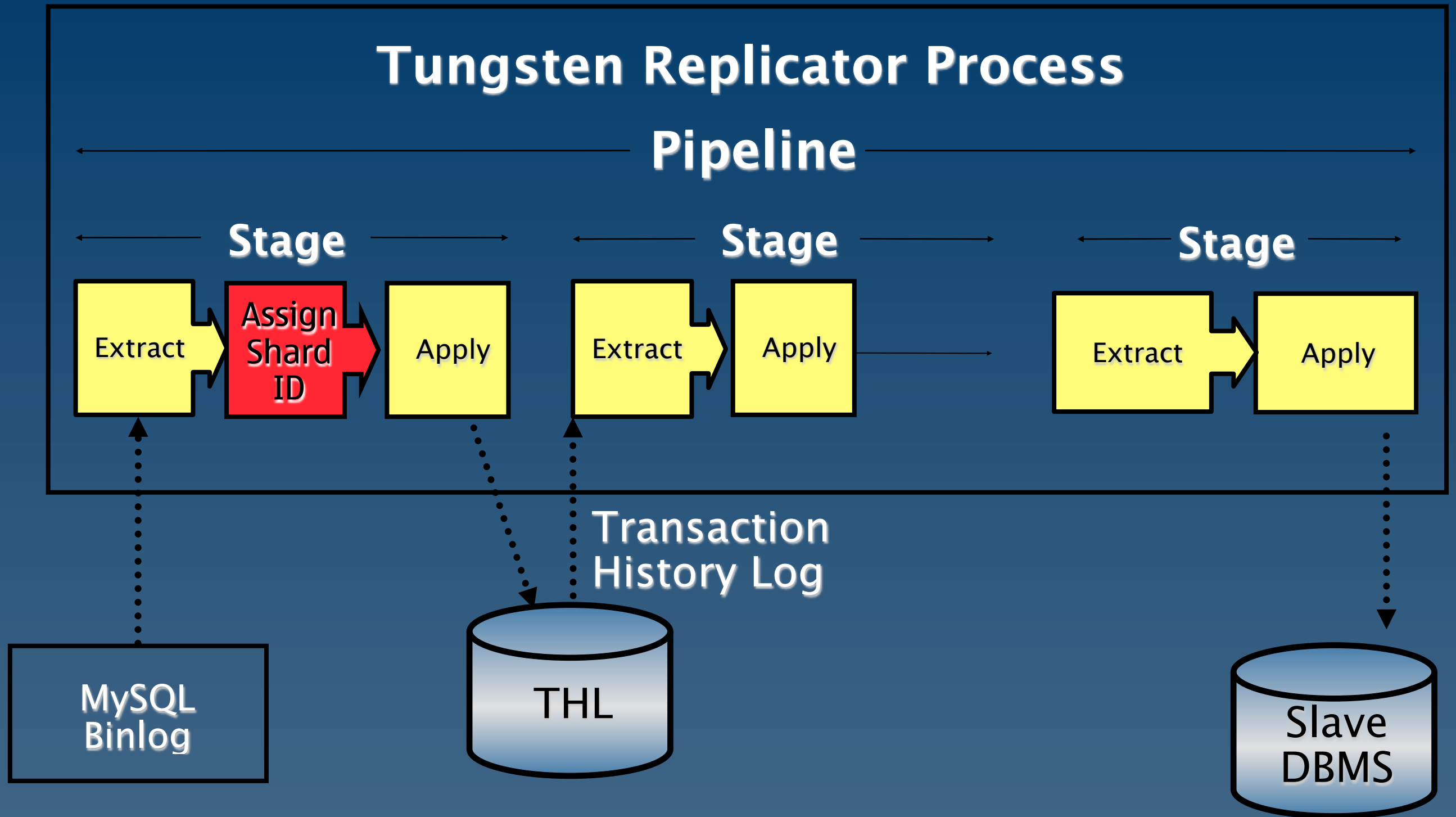
4. ... and we're done

```
# new master  
select count(*) from employees.titles  
count(*)  
443308
```

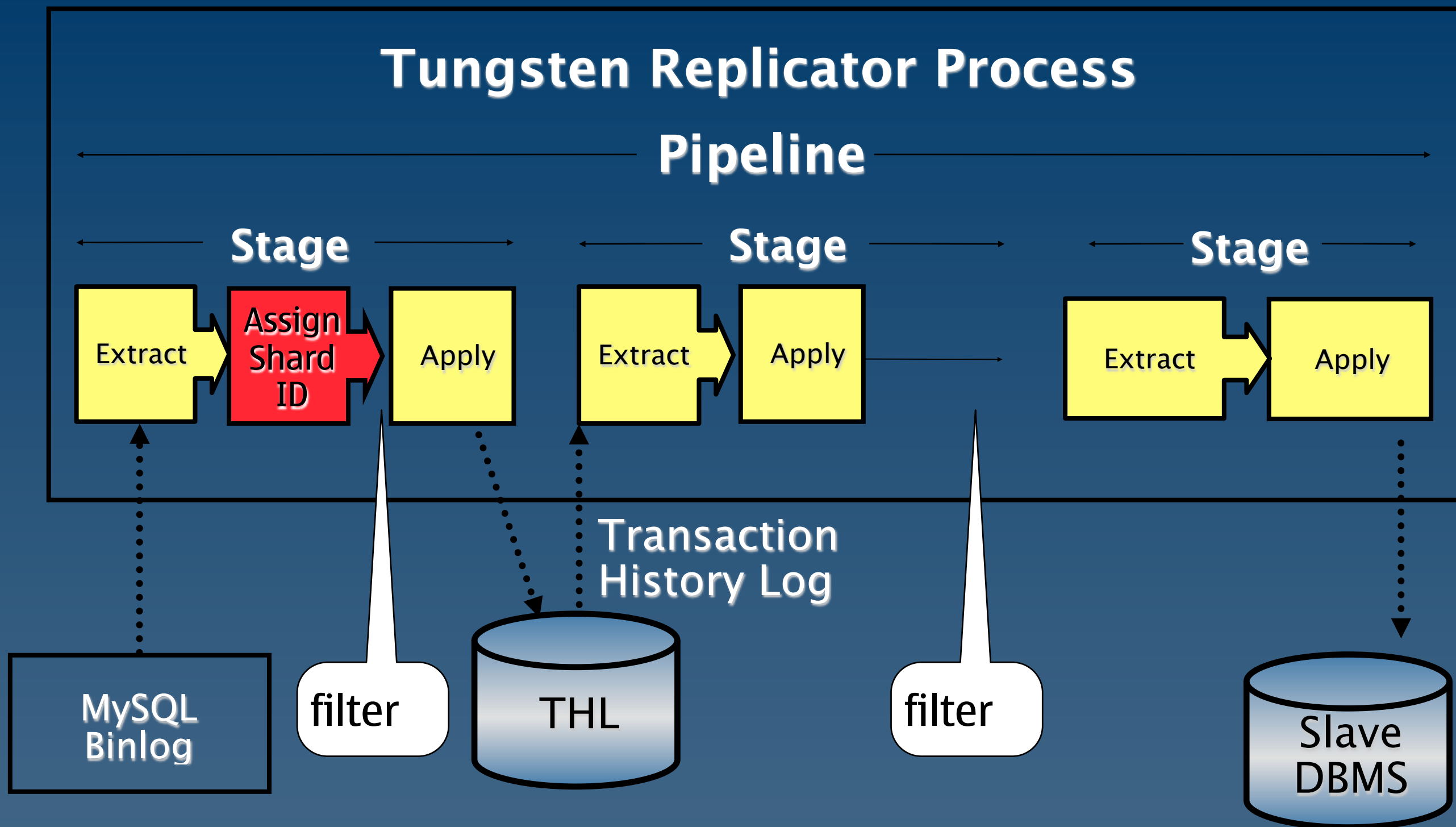
```
# new slave:  
count(*)  
443308
```

Filters

Replicator Pipeline Architecture



Replicator Pipeline Architecture



Restrict replication to some schemas and tables

```
./tools/tungsten-installer \  
  --master-slave -a \  
  ...  
  --svc-extractor-filters=replicate \  
  "--property=replicator.filter.replicate.do=test,*.foo" \  
  ...  
  --start-and-report
```

```
# test="test.*"    -> same drawback as binlog-do-db in MySQL  
# *.foo = table 'foo' in any database  
# employees.dept_codes,employees.salaries => safest way
```

Exclude some schemas and tables from replication

```
./tools/tungsten-installer \  
  --master-slave -a \  
  ...  
  --svc-extractor-filters=replicate \  
  "--property=replicator.filter.replicate.ignore=test,*.foo" \  
  ...  
  --start-and-report
```

```
# test="test.*" -> same drawback as binlog-ignore-db in MySQL  
# *.foo = table 'foo' in any database  
# employees.dept_codes,employees.salaries => safest way
```

```
# DO NOT MIX .do and .ignore!  
# (you can do it, but it may not do what you mean)
```

Change name of replicated schema

```
-a --svc-applier-filters=dbtransform \  
  --property=replicator.filter.dbtransform.from_regex1=stores \  
  --property=replicator.filter.dbtransform.to_regex1=playground
```

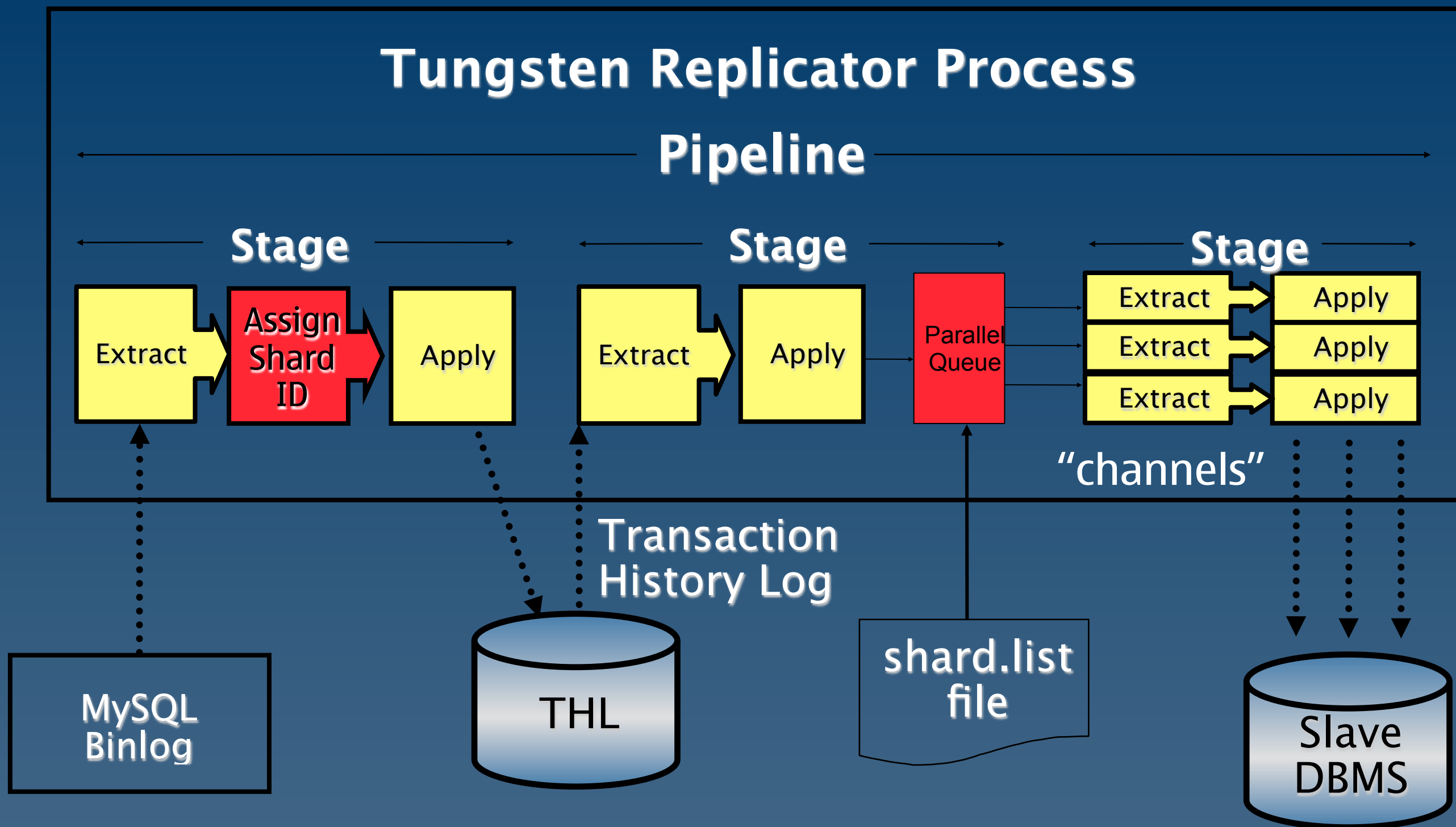
```
# from_regex1=stores -> name of the schema in the master
```

```
# to_regex1=playground -> name of the schema in the slave
```

```
# WARNING: requires "USE schema_name" to work properly.
```

parallel replication

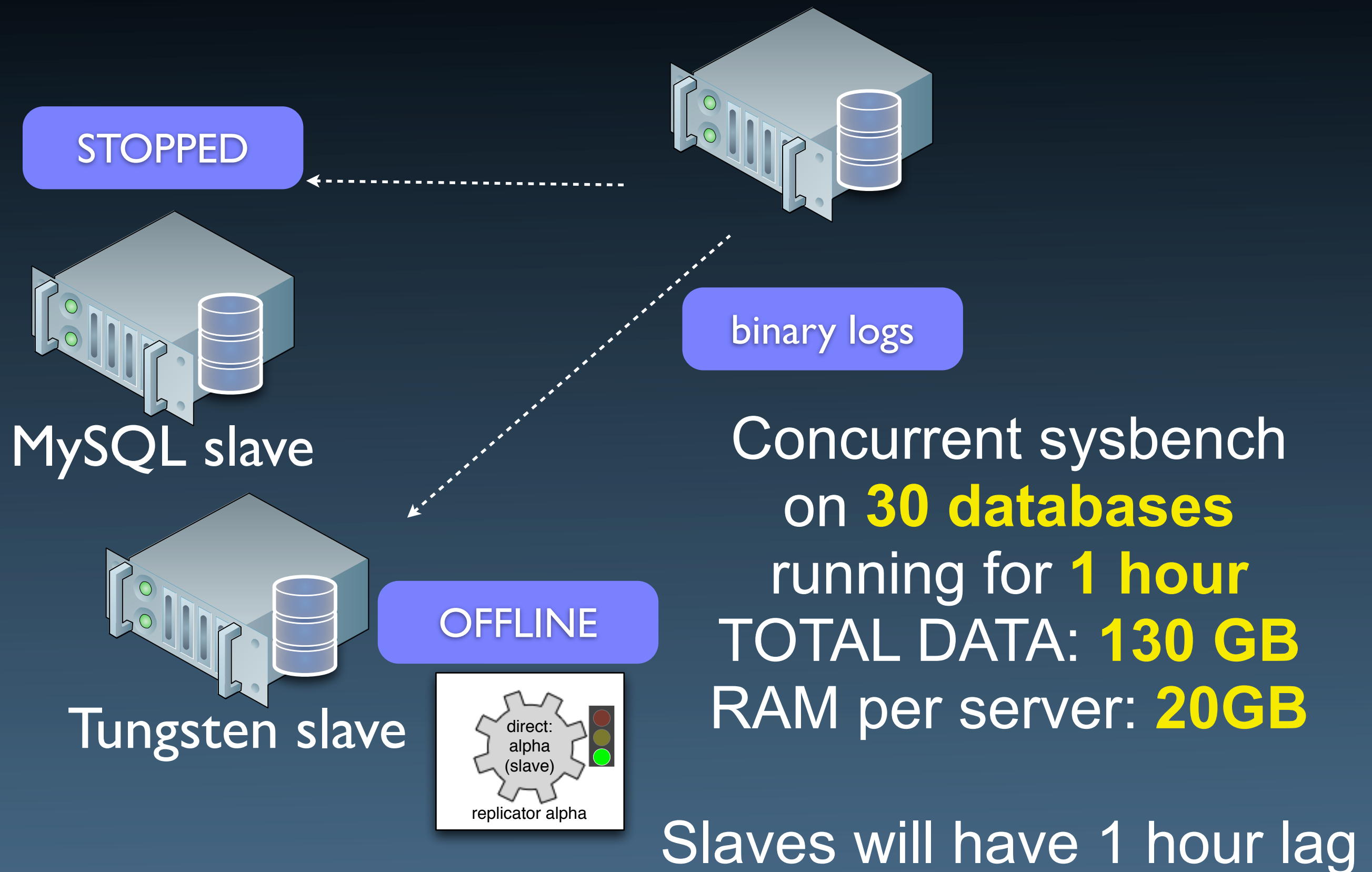
Replicator Pipeline Architecture



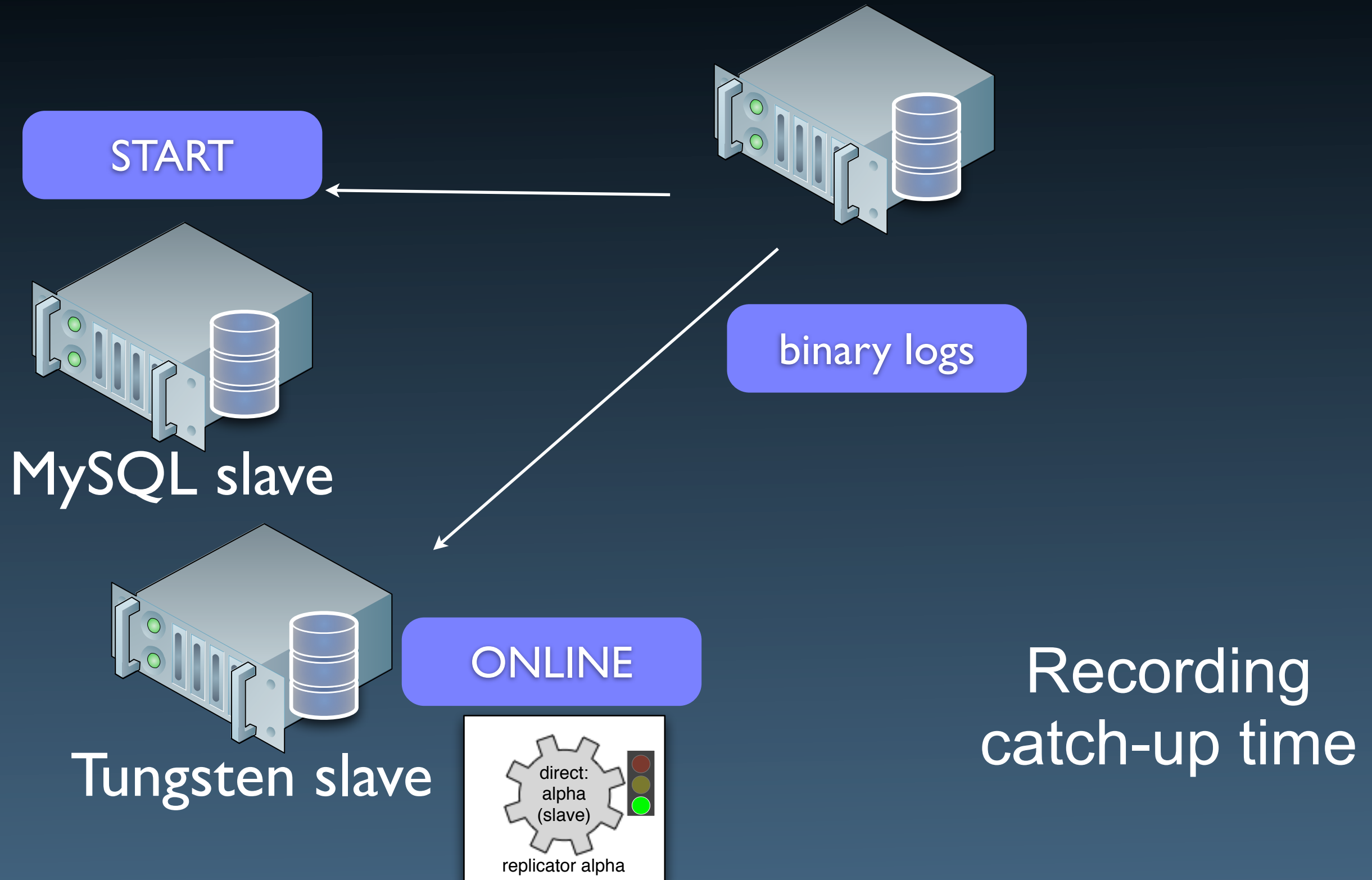
Parallel replication facts

- ✓ Sharded by database
- ✓ Good choice for slave lag problems
- ❖ Bad choice for single database projects

Parallel Replication test



measuring results



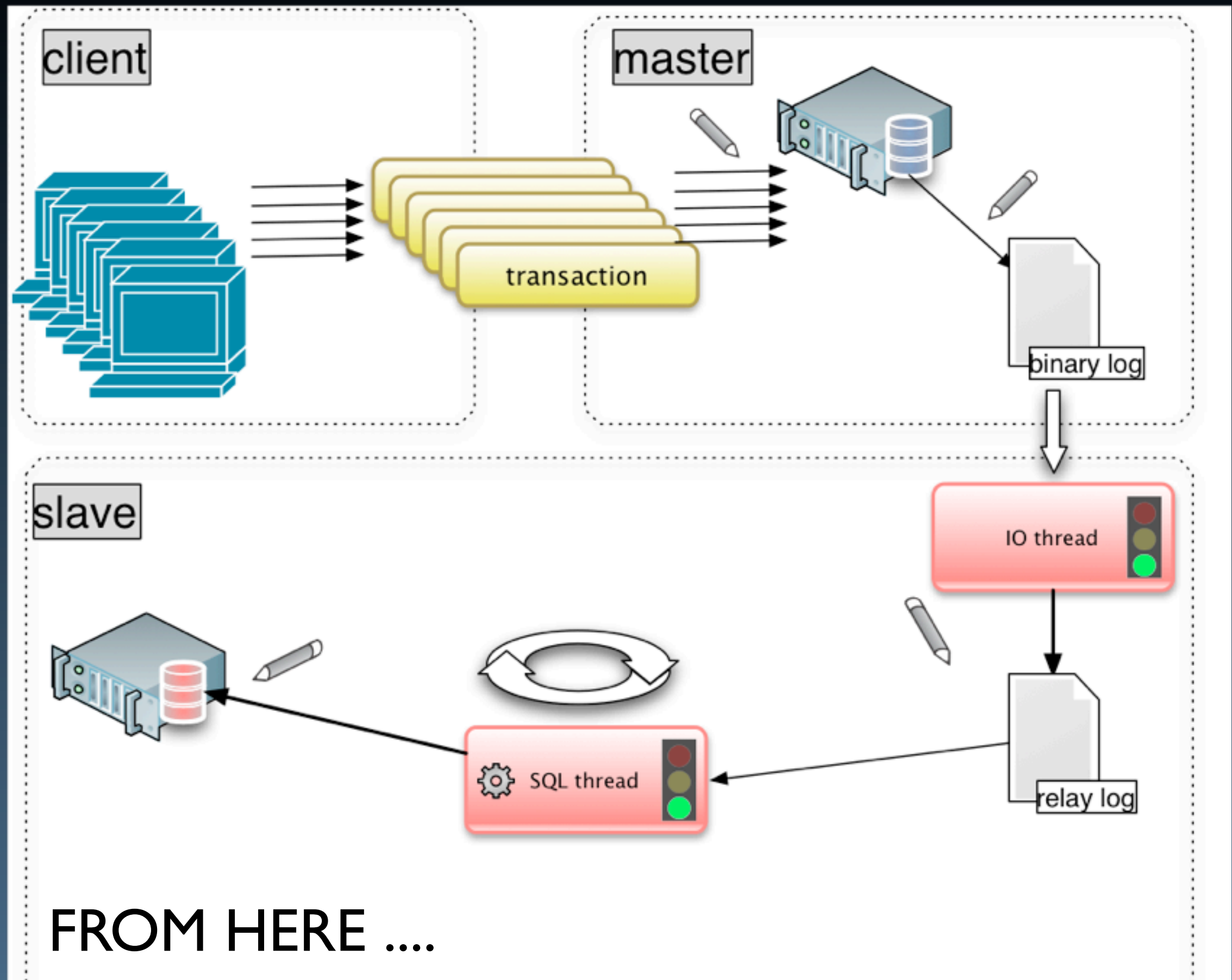
MySQL native replication

slave catch up in **04:29:30**

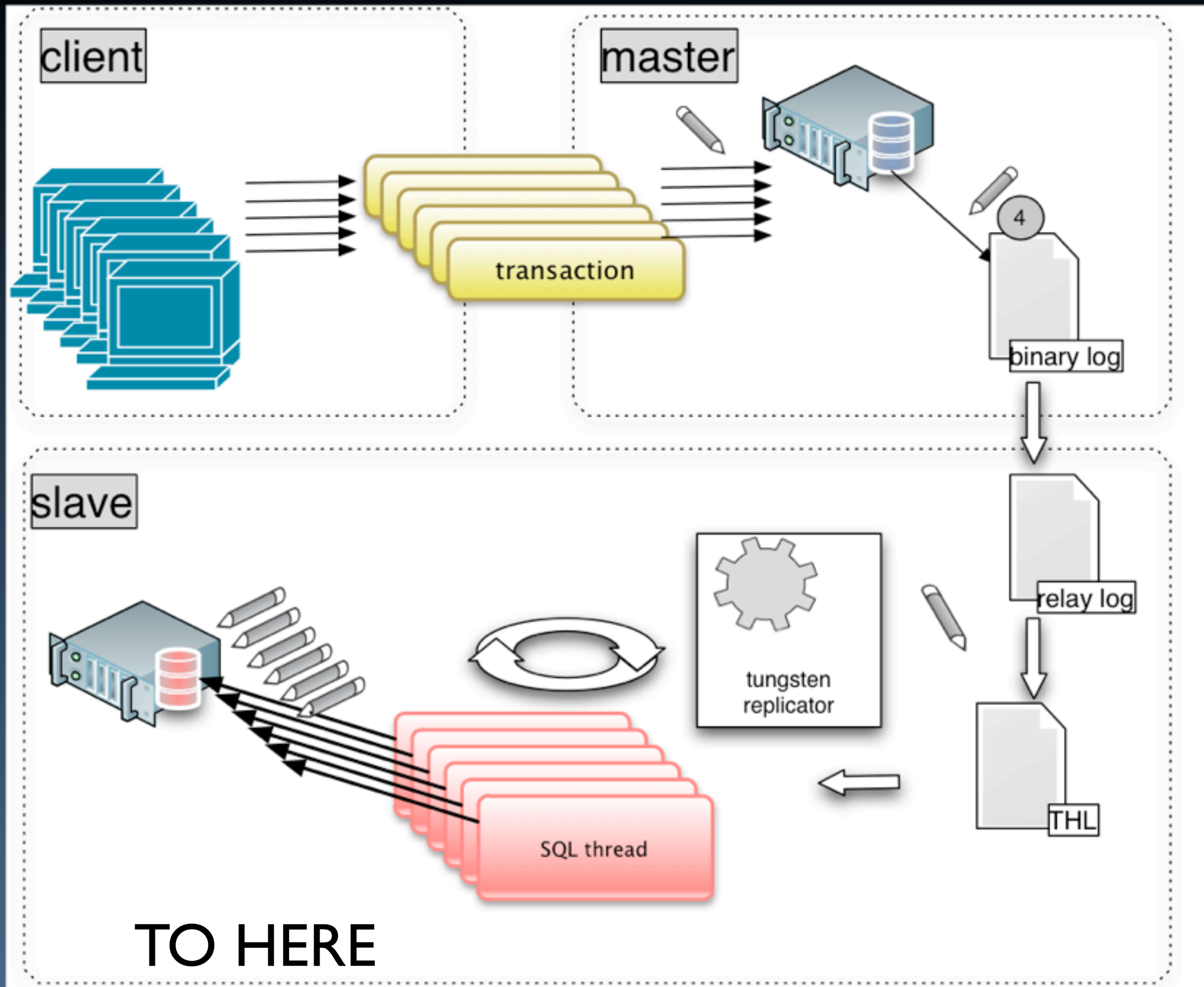
Tungsten parallel replication

slave catch up in **00:55:40**

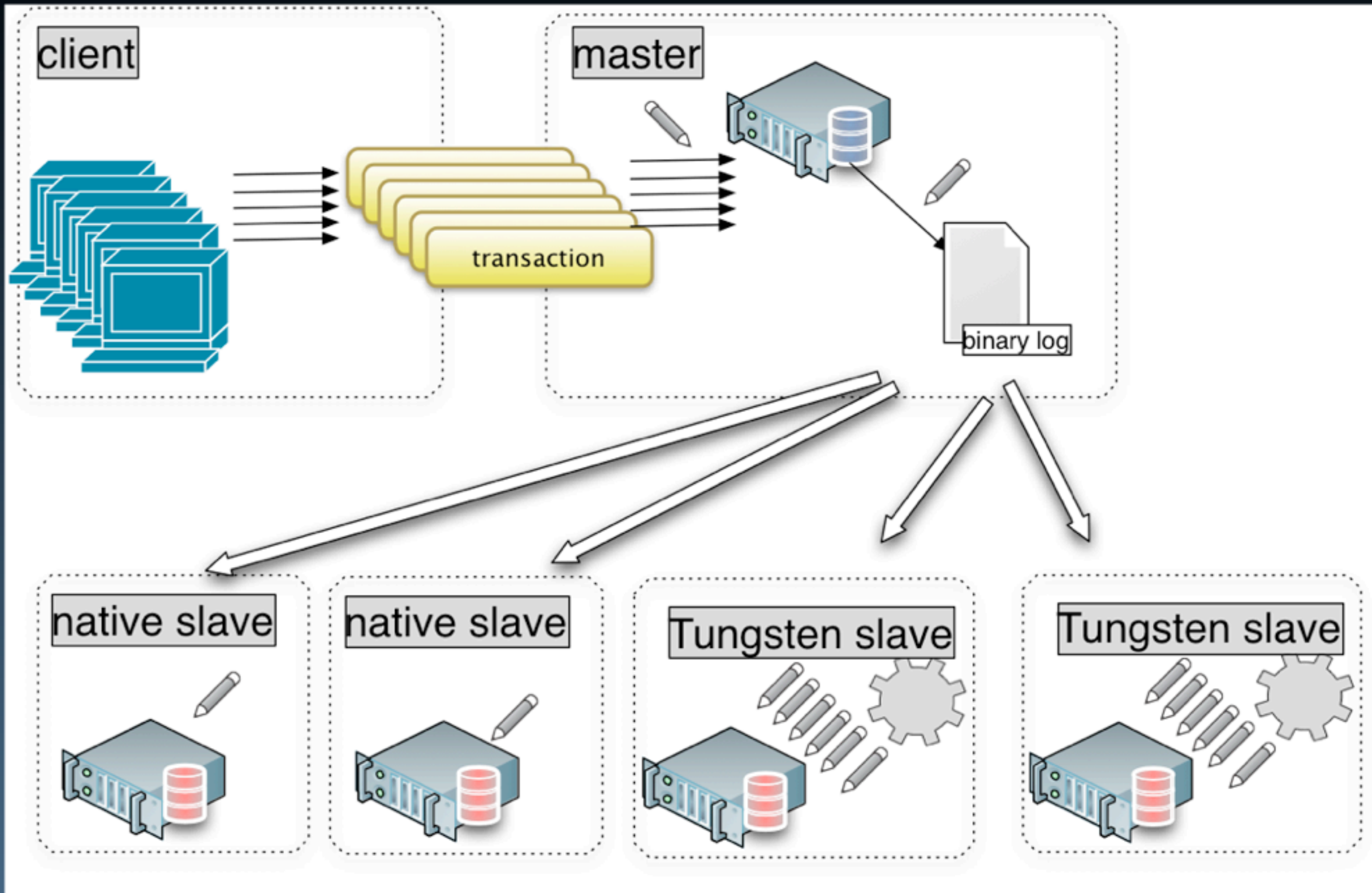
Parallel replication made simpler



Parallel replication made simpler



Parallel replication made simpler



parallel replication direct slave facts

parallel replication direct slave facts

- ✓ No need to install Tungsten on the master

parallel replication direct slave facts

- ✓ No need to install Tungsten on the master
- ✓ Tungsten runs only on the slave

parallel replication direct slave facts

- ✓ No need to install Tungsten on the master
- ✓ Tungsten runs only on the slave
- ✓ Replication can revert to native slave with two commands (`trepctl offline; start slave`)

parallel replication

direct slave facts

- ✓ No need to install Tungsten on the master
- ✓ Tungsten runs only on the slave
- ✓ Replication can revert to native slave with two commands (`trepctl offline; start slave`)
- ✓ Native replication can continue on other slaves

parallel replication

direct slave facts

- ✓ No need to install Tungsten on the master
- ✓ Tungsten runs only on the slave
- ✓ Replication can revert to native slave with two commands (`trepctl offline; start slave`)
- ✓ Native replication can continue on other slaves
- ❖ Failover (either native or Tungsten) becomes a manual task

Checking parallel replication

```
trepctl status
```

```
trepctl status -name tasks
```

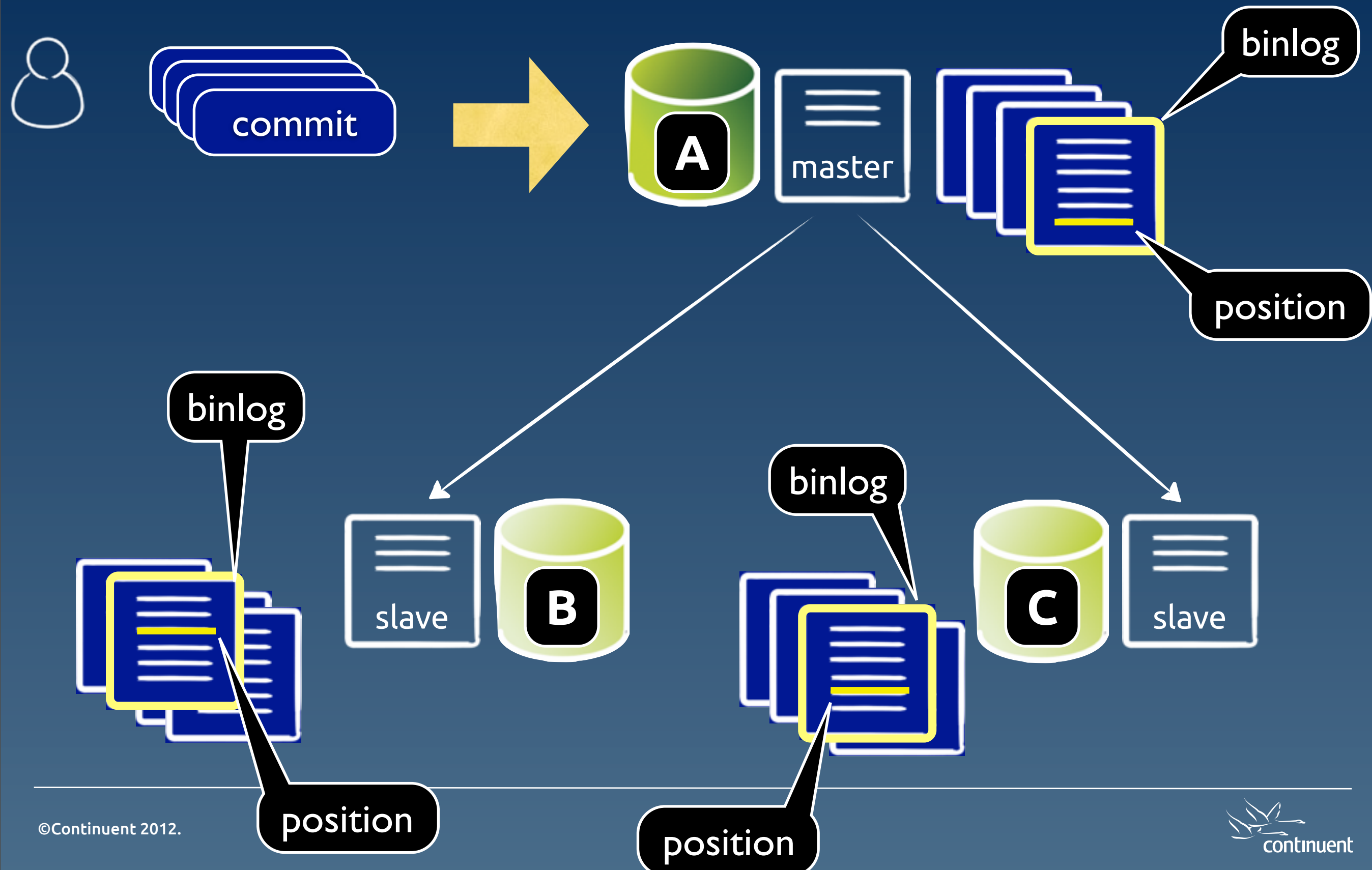
```
trepctl status -name shards
```

```
trepctl status -name stores
```

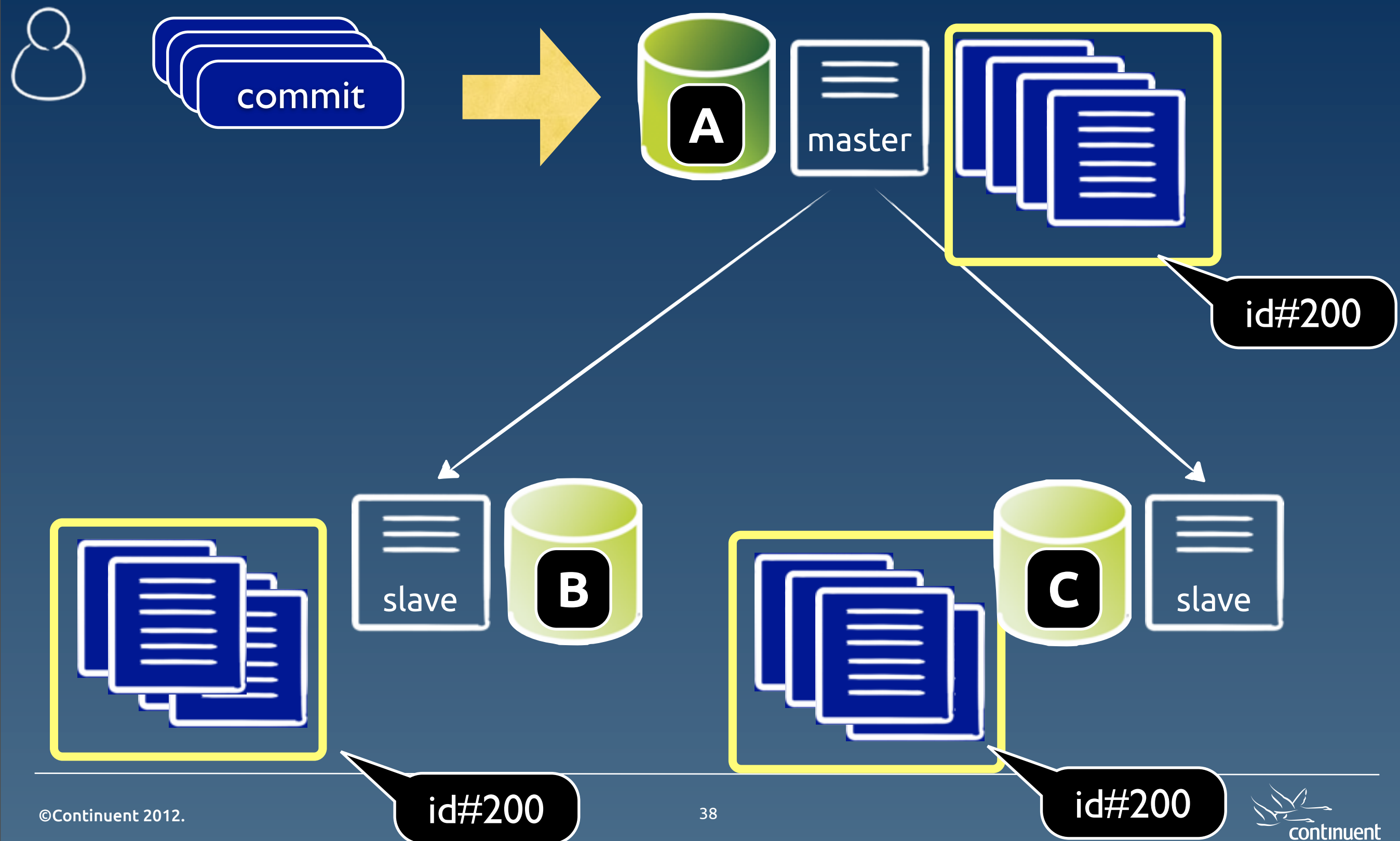
Tungsten GTID vs MySQL 5.6 GTID

- What is GTID
- How it works in Tungsten
- How it works (or not) in MySQL 5.6

without global transaction ID



with global transaction ID



Tungsten and global transaction ID: activation

(none)
active by default

Tungsten and global transaction ID: status

```
trepctl status
```

```
Processing status command...
```

NAME	VALUE
-----	-----
appliedLastEventId	: mysql-bin.000002:000000000000001442;0
appliedLastSeqno	: 6 ←
appliedLatency	: 0.862
clusterName	: default
currentEventId	: NONE
currentTimeMillis	: 1354304680923
dataServerHost	: qa.r4.continuent.com

Tungsten and global transaction ID: seeing transactions

```
thl list -seqno 6
```

```
SEQ# = 6 / FRAG# = 0 (last frag)
```

- TIME = 2012-11-30 20:44:35.0
- EPOCH# = 0
- EVENTID = mysql-bin.000002:00000000000000001442;0
- SOURCEID = qa.r1.continuent.com
- SQL(0) = insert into test.v1 values (1, 'inserted by node #1') /* ____SERVICE____ = [cookbook] */

Tungsten and global transaction ID: changing master connection

```
trepctl offline  
trepctl online -seqno 105
```

Tungsten and Global transaction ID: crash-safe slave tables

```
mysql -e 'select * from tungsten_cookbook.trep_commit_seqno\G'
***** 1. row *****
      task_id: 0
      seqno: 6
      fragno: 0
      last_frag: 1
      source_id: qa.r1.continuent.com
      epoch_number: 0
      eventid: mysql-bin.000002:00000000000000001442;0
      applied_latency: 0
      update_timestamp: 2012-11-30 20:44:35
      shard_id: test
      extract_timestamp: 2012-11-30 20:44:35
```

Tungsten and Global transaction ID: crash-safe tables and parallel replication

```
mysql -e 'select seqno, source_id, shard_id,update_timestamp from  
tungsten_cookbook.trep_commit_seqno'
```

seqno	source_id	shard_id	update_timestamp
7	qa.r1.continuent.com	db1	2012-11-30 20:54:14
8	qa.r1.continuent.com	db2	2012-11-30 20:54:14
9	qa.r1.continuent.com	db3	2012-11-30 20:54:14
10	qa.r1.continuent.com	db4	2012-11-30 20:54:14
11	qa.r1.continuent.com	db5	2012-11-30 20:54:14
12	qa.r1.continuent.com	db6	2012-11-30 20:54:14
13	qa.r1.continuent.com	db7	2012-11-30 20:54:14
14	qa.r1.continuent.com	db8	2012-11-30 20:54:14
15	qa.r1.continuent.com	db9	2012-11-30 20:54:14
16	qa.r1.continuent.com	db10	2012-11-30 20:54:14

MySQL 5.6 and global transaction ID activation

```
mysqld --log-slave-updates \  
      --gtid-mode=on \  
      --enforce-gtid-consistency
```

MySQL 5.6 and global transaction ID seeing transactions

```
#121203 11:15:49 server id 1  end_log_pos 344 CRC32 0x45b25c8f
  GTID [commit=yes]
SET @@SESSION.GTID_NEXT= '7A77A490-3D3A-11E2-8CC9-7DCF9991097B:
2' /*!*/;
# at 344
#121203 11:15:49 server id 1  end_log_pos 423 CRC32 0x873c8fac
  Query thread_id=3 exec_time=0 error_code=0
SET TIMESTAMP=1354533349 /*!*/;
BEGIN
/*!*/;
# at 423
#121203 11:15:49 server id 1  end_log_pos 522 CRC32 0xb4bf4372
  Query thread_id=3 exec_time=0 error_code=0
SET TIMESTAMP=1354533349 /*!*/;
insert into t1 values (1)
```

MySQL 5.6 and global transaction ID status

```
show slave status\G
```

```
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 127.0.0.1
Master_User: rsandbox
Master_Port: 13233
Connect_Retry: 60
Master_Log_File: mysql-bin.000002
Read_Master_Log_Pos: 1837
Relay_Log_File: mysql_sandbox13234-relay-bin.000005
Relay_Log_Pos: 2047
Relay_Master_Log_File: mysql-bin.000002
```

```
...
```

```
Retrieved_Gtid_Set: 46E13434-3B28-11E2-BF47-6C626DA07446:1-7
Executed_Gtid_Set: 46E13434-3B28-11E2-BF47-6C626DA07446:1-7
```

MySQL 5.6 and global transaction ID changing master connection

```
CHANGE MASTER TO master_log_file='mysql-bin-000003',  
master_log_pos='1234'
```

No global transaction ID is used

MySQL 5.6 and global transaction ID crash-safe slave table

```
select * from slave_relay_log_info\G
***** 1. row *****
Number_of_lines: 7
Relay_log_name: ./mysql_sandbox13234-relay-bin.000005
Relay_log_pos: 2047
Master_log_name: mysql-bin.000002
Master_log_pos: 1837
Sql_delay: 0
Number_of_workers: 5
Id: 1
```

NO Global transaction ID is used!

MySQL 5.6 and global transaction ID crash-safe slave table + parallel

```
select * from mysql.slave_worker_info\G
      Id: 12
      Relay_log_name: ./mysql_sandbox13234-relay-bin.000007
      Relay_log_pos: 4299
      Master_log_name: mysql-bin.000002
      Master_log_pos: 7155
Checkpoint_relay_log_name: ./mysql_sandbox13234-relay-bin.000007
Checkpoint_relay_log_pos: 1786
Checkpoint_master_log_name: mysql-bin.000002
Checkpoint_master_log_pos: 4642
Checkpoint_seqno: 9
Checkpoint_group_size: 64
Checkpoint_group_bitmap: ?
```

NO Global transaction ID is used!

Installation

Installation

- System Requirements
- Validate first
- Deploying from a single location

Installation - tools

- tools/ tungsten-installer
 - tools/ configure-service
 - tools/update
-
- (Using the cookbook recipes, you hardly see them)

Tungsten in practice Installation

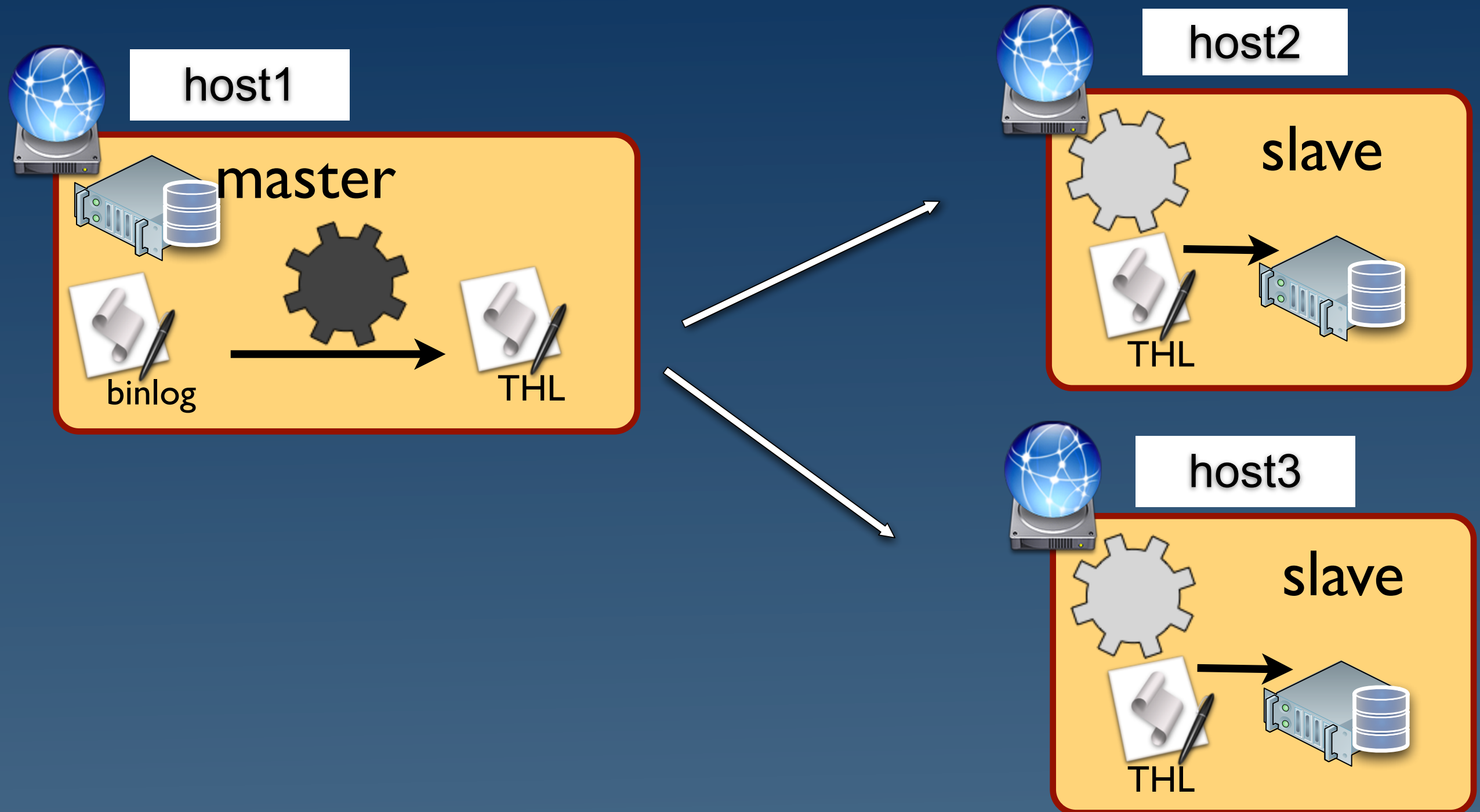
Installation

- Check the requirements
- Get the binaries
- Expand the tarball
- Run cookbook

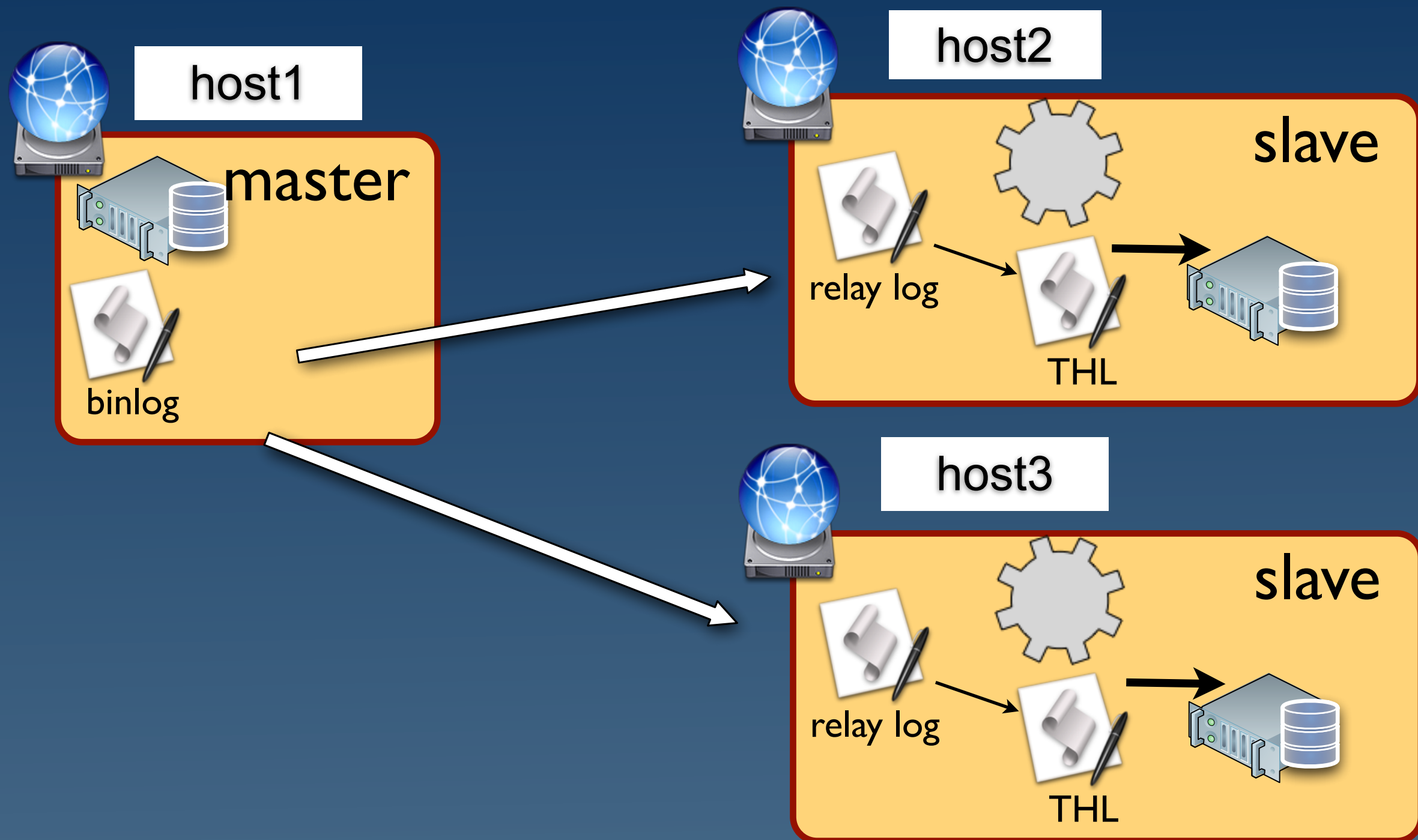
REQUIREMENTS

- Java JRE or JDK (Sun/Oracle or Open-jdk)
- Ruby 1.8 (only during installation)
- ssh access to the same user in all nodes
- MySQL user with all privileges

master-slave



direct



Demo

- Install Master Slave using the Cookbook

Overview of Tungsten cookbook

tungsten cookbook

tungsten-replicator-2.0.7-xx

|

+-- /cluster-home

+-- /**cookbook**

+-- /tools

+-- /tungsten-replicator

tungsten cookbook

tungsten-replicator-2.0.7-xx

|

+ - - / **cookbook**

|

+ - - COMMON_NODES.sh

+ - - USER_VALUES.sh

+ - - NODES_MASTER_SLAVE.sh

+ - - show_master_slave.sh

+ - - test_master_slave.sh

+ - - clear_cluster_master_slave.sh

...

tungsten cookbook

tungsten-replicator-2.0.7-xx

|

+ - - / **cookbook**

|

+ - - COMMON_NODES.sh

+ - - USER_VALUES.sh

+ - - NODES_STAR.sh

+ - - show_star.sh

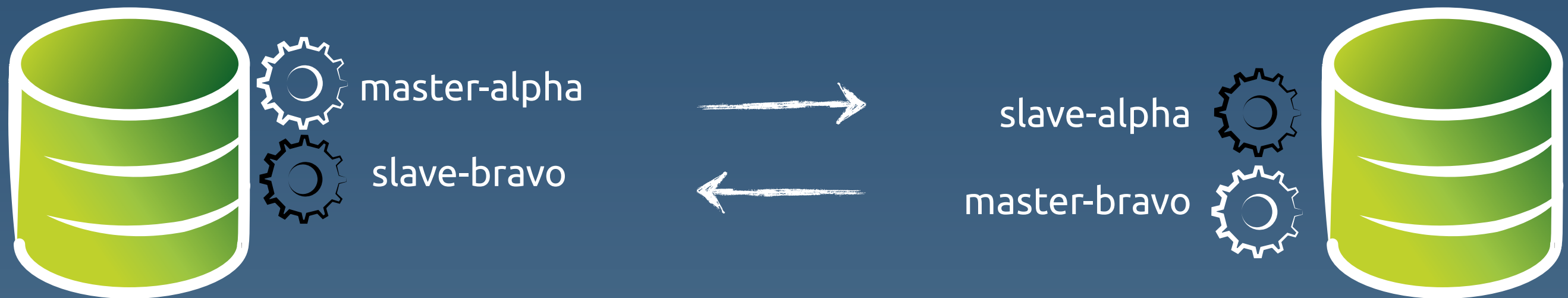
+ - - test_star.sh

+ - - clear_cluster_star.sh

...

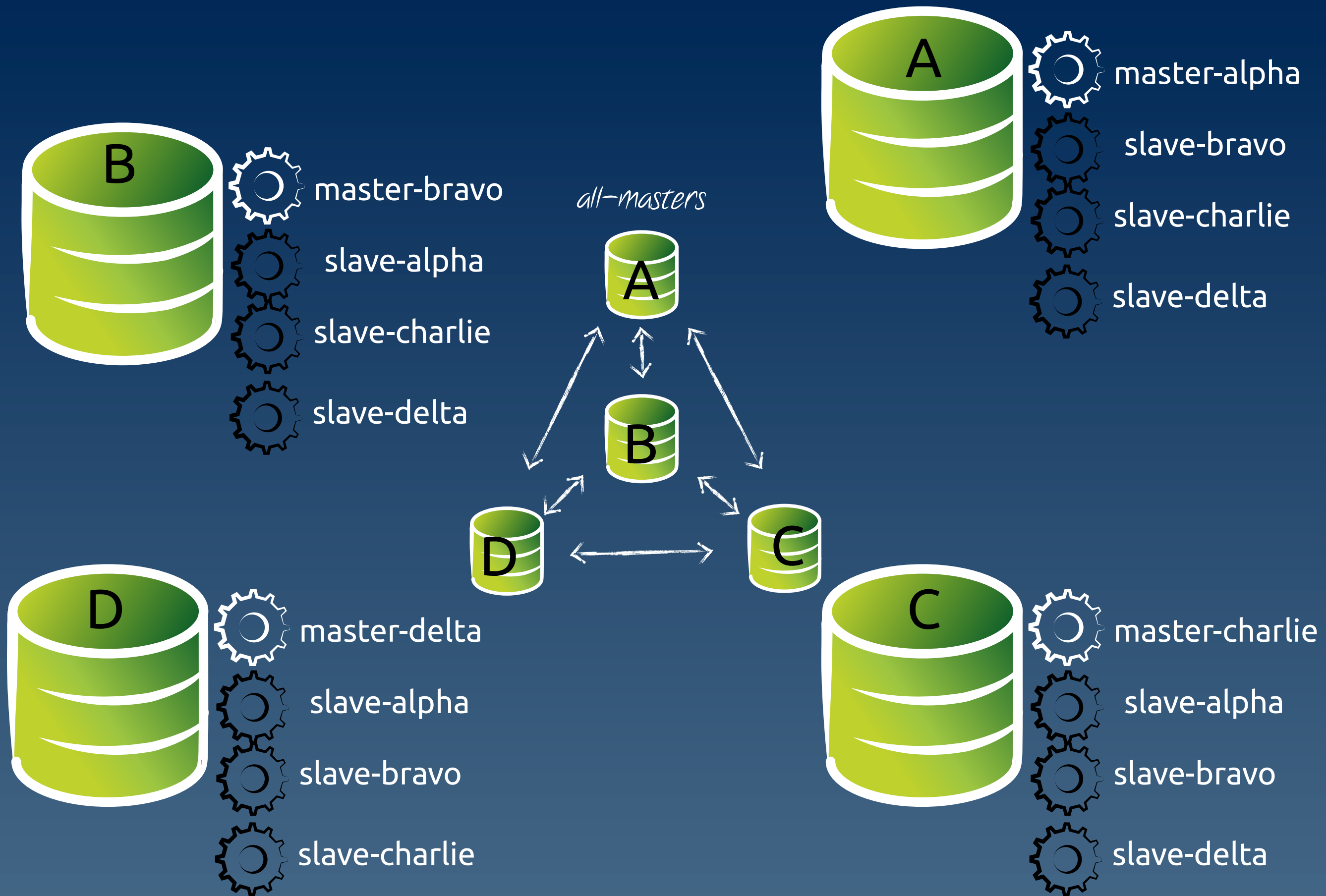
Master - to - master

- Bi-directional installation
- Operational steps:
 - install a master service in both servers
 - install the corresponding slave service in the other server



BI-DIR: the painless way

- edit cookbook/COMMON_NODES.sh
- edit cookbook/USER_VALUES.sh
 - remove two nodes
- edit the variables in cookbook/
NODES_ALL_MASTERS.sh
- cookbook/install_all_masters.sh



Multiple masters

- fan-in
- Steps:
 - install a master service in each node
 - install a slave service for each master in the fan-in node
 - or :
 - `cookbook/install_fan_in.sh`

Master - to - master

- star schema
- Steps:
 - install a master service in each server
 - in the hub, install a slave service for each spoke
 - in each spoke, install a slave service for the hub, using bypass option
 - `cookbook/install_star.sh`

For more information download the
slides from Mondays Tutorial



560 S. Winchester Blvd., Suite 500
San Jose, CA 95128
Tel +1 (866) 998-3642
Fax +1 (408) 668-1009
e-mail: sales@continuent.com

Our Blogs:

<http://scale-out-blog.blogspot.com>

<http://datacharmer.blogspot.com>

<http://flyingclusters.blogspot.com>

Continuent Website:

<http://www.continuent.com>

Tungsten Replicator 2.0:

<http://code.google.com/p/tungsten-replicator>