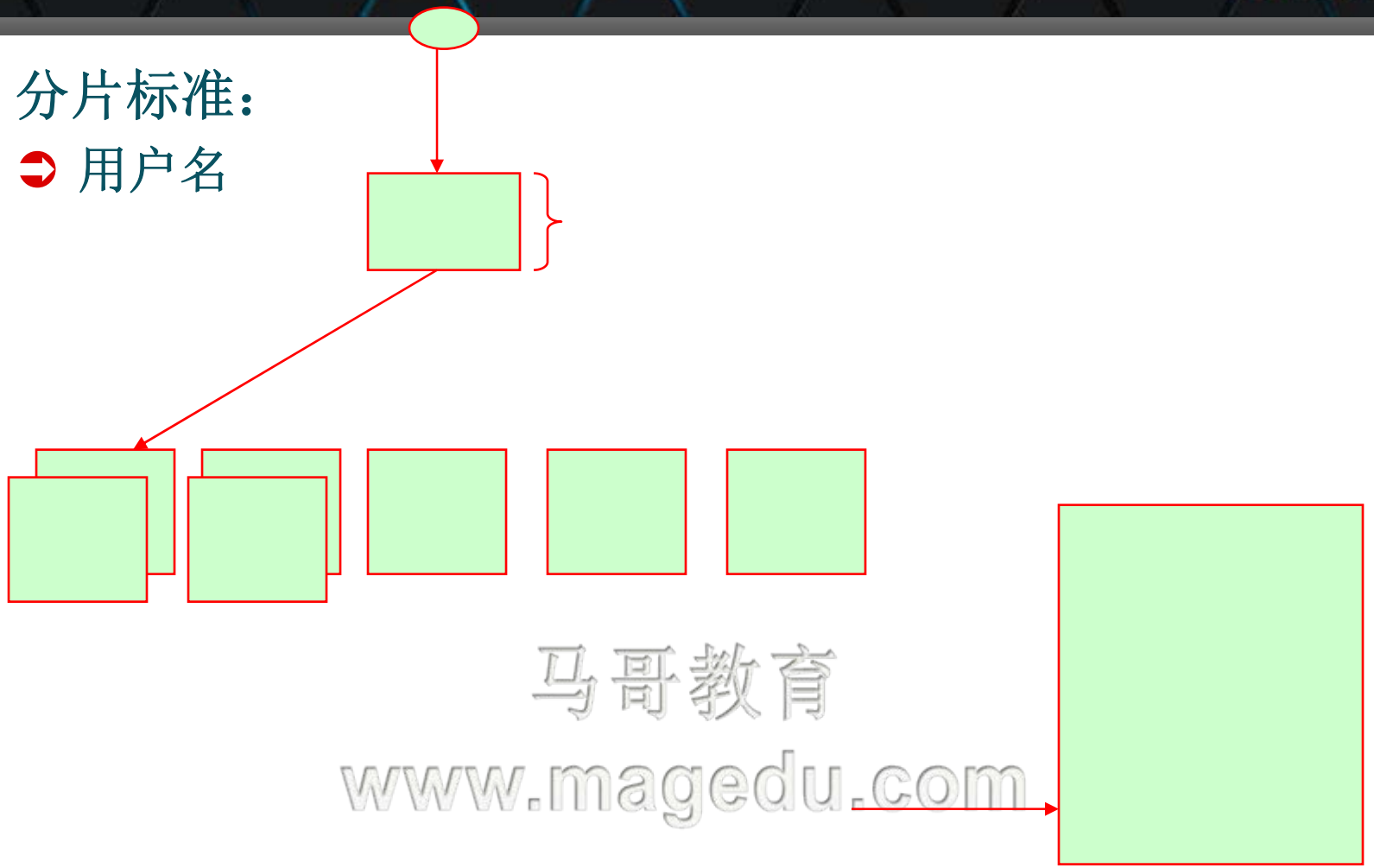
马哥教育

MySQL Replication

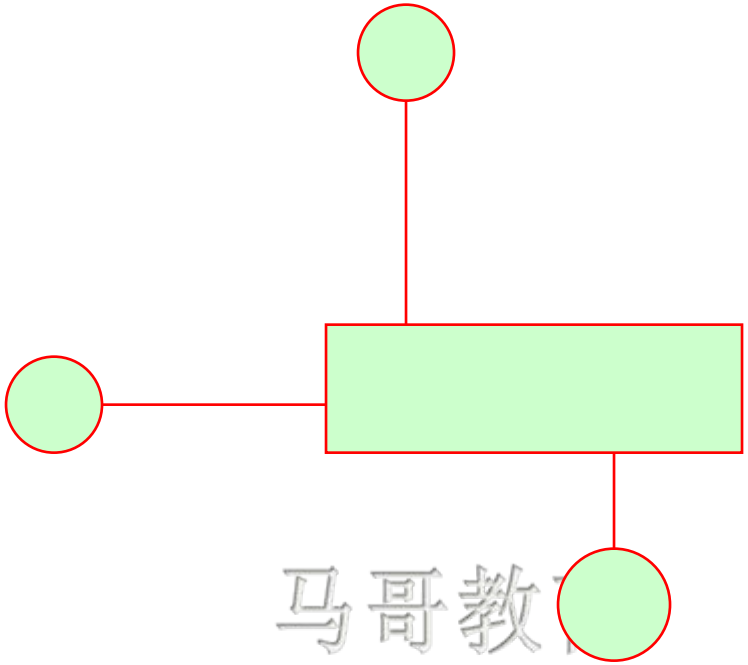
Include MySQL 5.6

主讲：马永亮(马哥)
QQ：1661815153
http://www.magedu.com

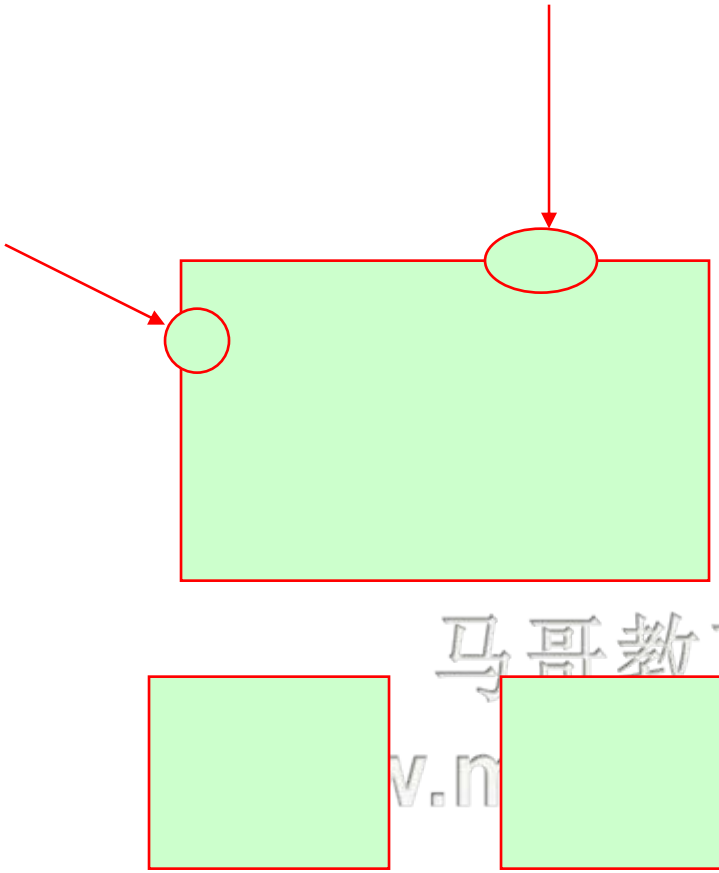
- 分片标准：
  - 用户名



马哥教育
www.magedu.com

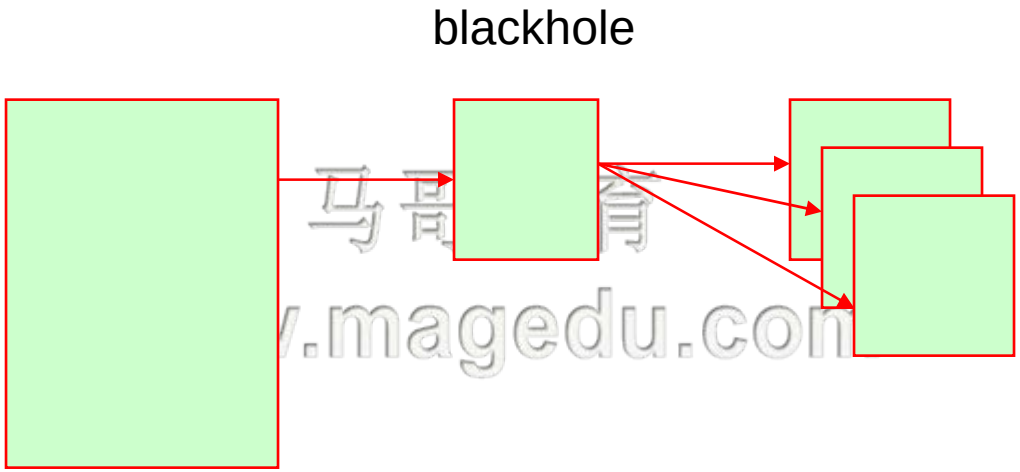


马哥教育
www.magedu.com

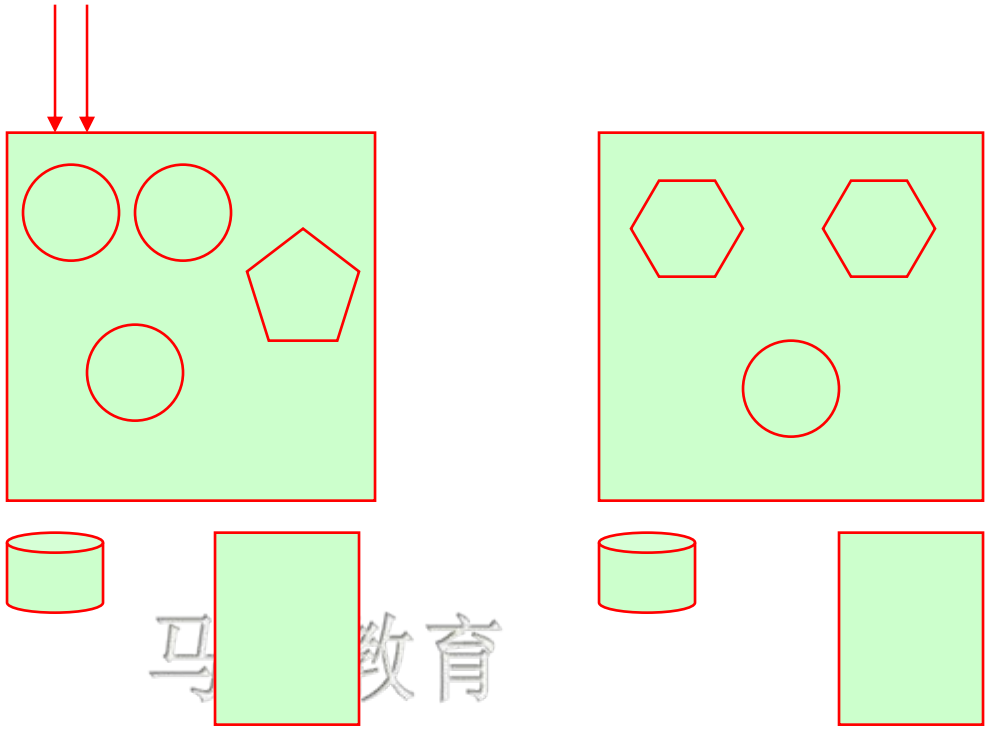


马哥教育
www.magedu.com

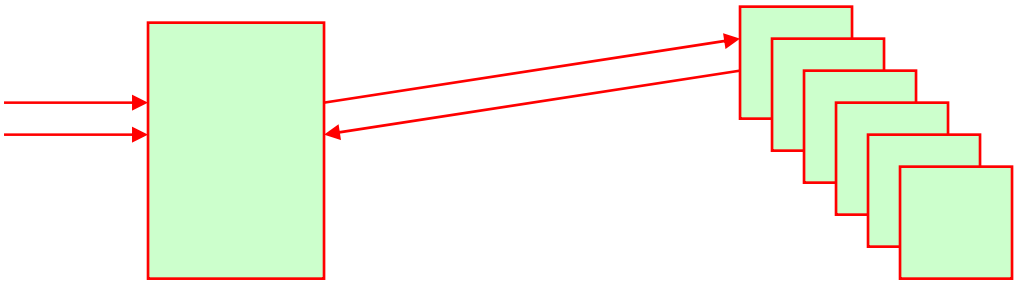
- 冗余，异地灾备
- 读、写分离
  - master：读/写
  - slave：读
    - ipvs、haproxy
- 升级测试



马哥教育
www.magedu.com

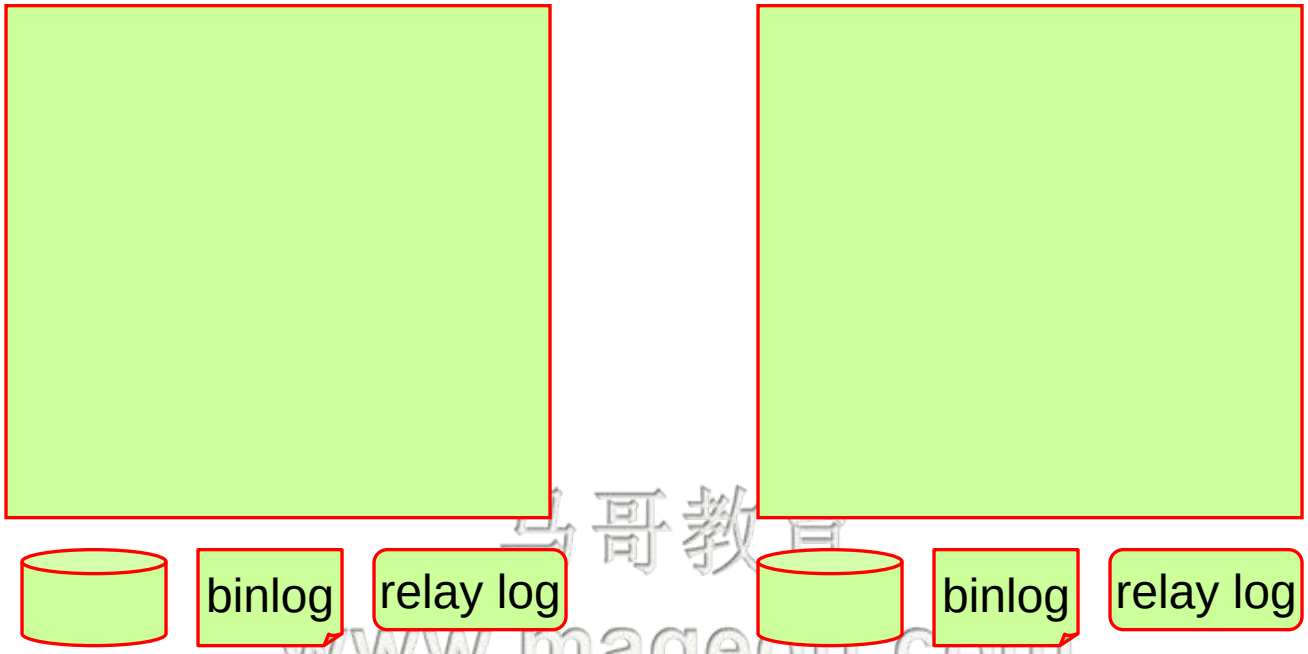


马[illegible]教育
www.magedu.com



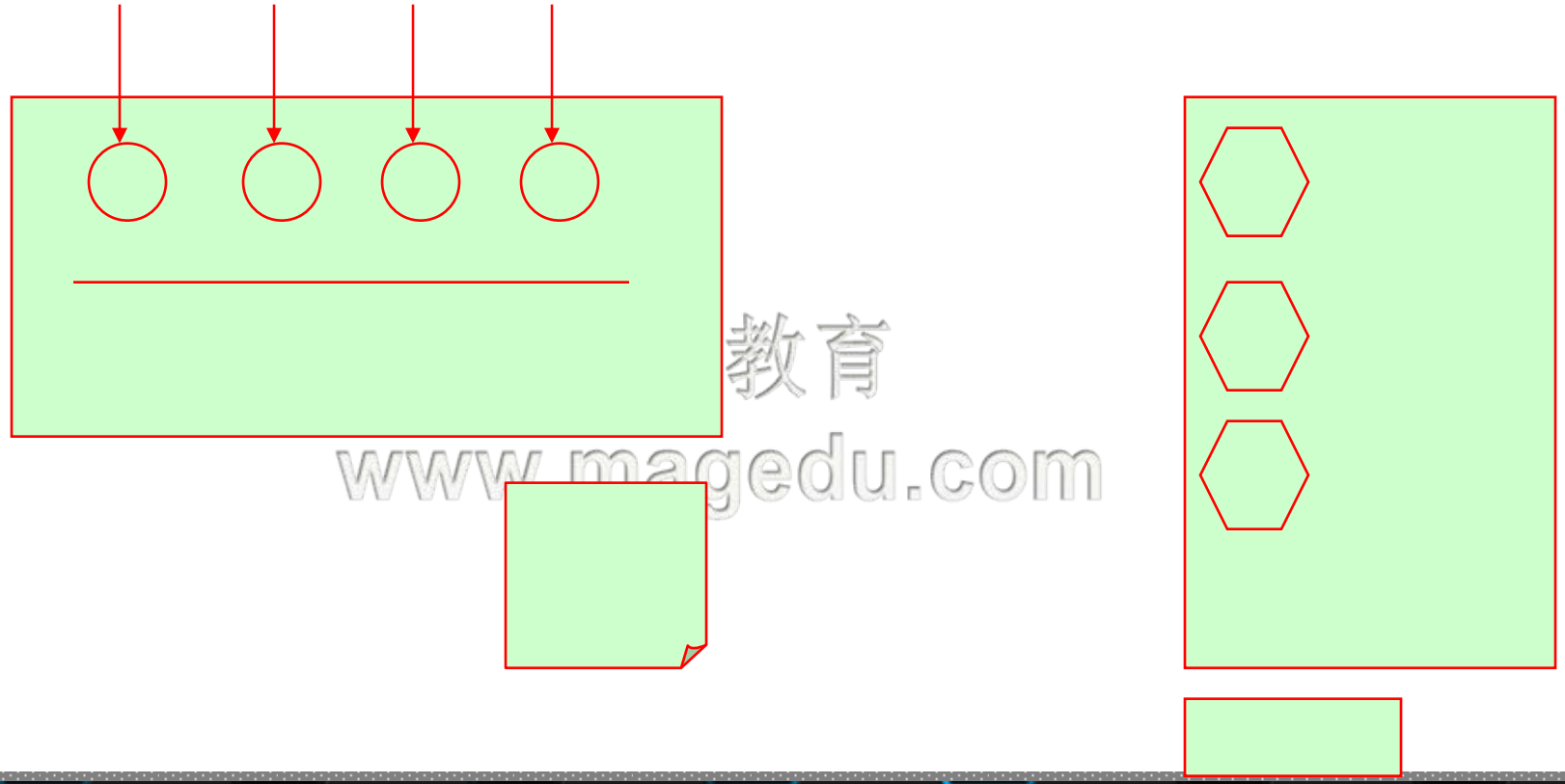
马哥教育
www.magedu.com

server-id



马哥教育
www.magedu.com

- TID
  - GTID
    - UUID



教育
www.magedu.com

- MySQL Replication
- Common Use Cases
- More Advanced Use Cases
- MySQL 5.6 Advanced Replication Features
- MySQL Utilities
- Summary

马哥教育
www.magedu.com

What is Replication?

- For the purposes of this paper we define "replication" as the duplication of data to one or more locations
- These can be co-located within a data center, or geographically distributed
- Replication enables a database to copy or duplicate changes from one physical location or system to another

马哥教育
www.magedu.com

Replication

- Happens at MySQL level, not Storage EngineLevel(*NDB)
- Asynchronous!(Semi-syncavailablein5.5)
- A server can have only 1 master
- IO Thread: Fetches from master
- SQL Thread: Executes on slave Single Threaded Execution (Expect enhancements in 5.6)
- Different schema's are possible between master and slave (Watchout!!):
  - different indexes
  - storage engines
  - data types
  - columns

马哥教育
www.magedu.com

Binary Logs

- Set of files
- Contains all writes and schema changes
- != REDO/Transaction log
- Rotated when full (Set max_binlog_size)
- Incrementing numbers (000001, 000002, 000003, ...)
- Relay Logs are also binary logs
- 2 Formats:
  - Statement Based (SBR)
  - Row based (RBR, since mysql 5.1)
-

马哥教育
www.magedu.com

- MySQL supports two kinds of replication: statement-based replication and row-based replication
  - Statement-based (or "logical") replication has been available since MySQL 3.23, and it's what most people are using in production today
  - Row-based replication is new in MySQL 5.1
- Both kinds work by recording changes in the master's binary log and replaying the log on the slave
- MySQL's replication is mostly backward compatible. That is, a newer server can usually be a slave of an older server without trouble

马哥教育
www.magedu.com

- Replication is relatively good for scaling reads, which you can direct to a slave, but it's not a good way to scale writes unless you design it right
- Attaching many slaves to a master simply causes the writes to be done many times, once on each slave
- Replication is also wasteful with more than a few slaves, because it essentially duplicates a lot of data needlessly

马哥教育
www.magedu.com

Problems Solved by Replication

- Data distribution
  - MySQL's replication is usually not very bandwidth-intensive,† and you can stop and start it at will
- Load balancing
  - Distribute read queries across several servers, which works very well for read-intensive applications
- Backups
- High availability and failover
- Testing MySQL upgrades

马哥教育
www.magedu.com

马哥教育

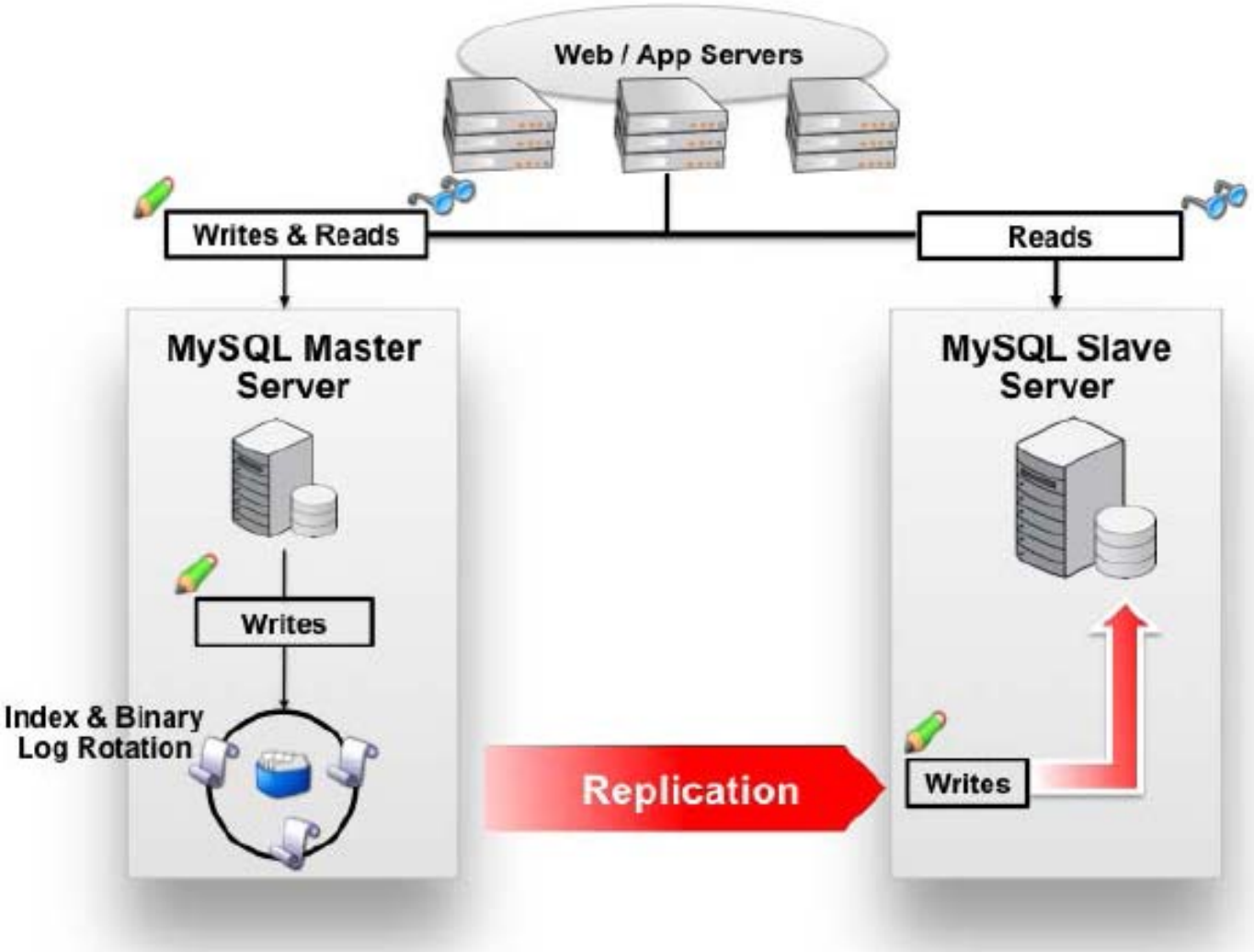
Replication Architecture

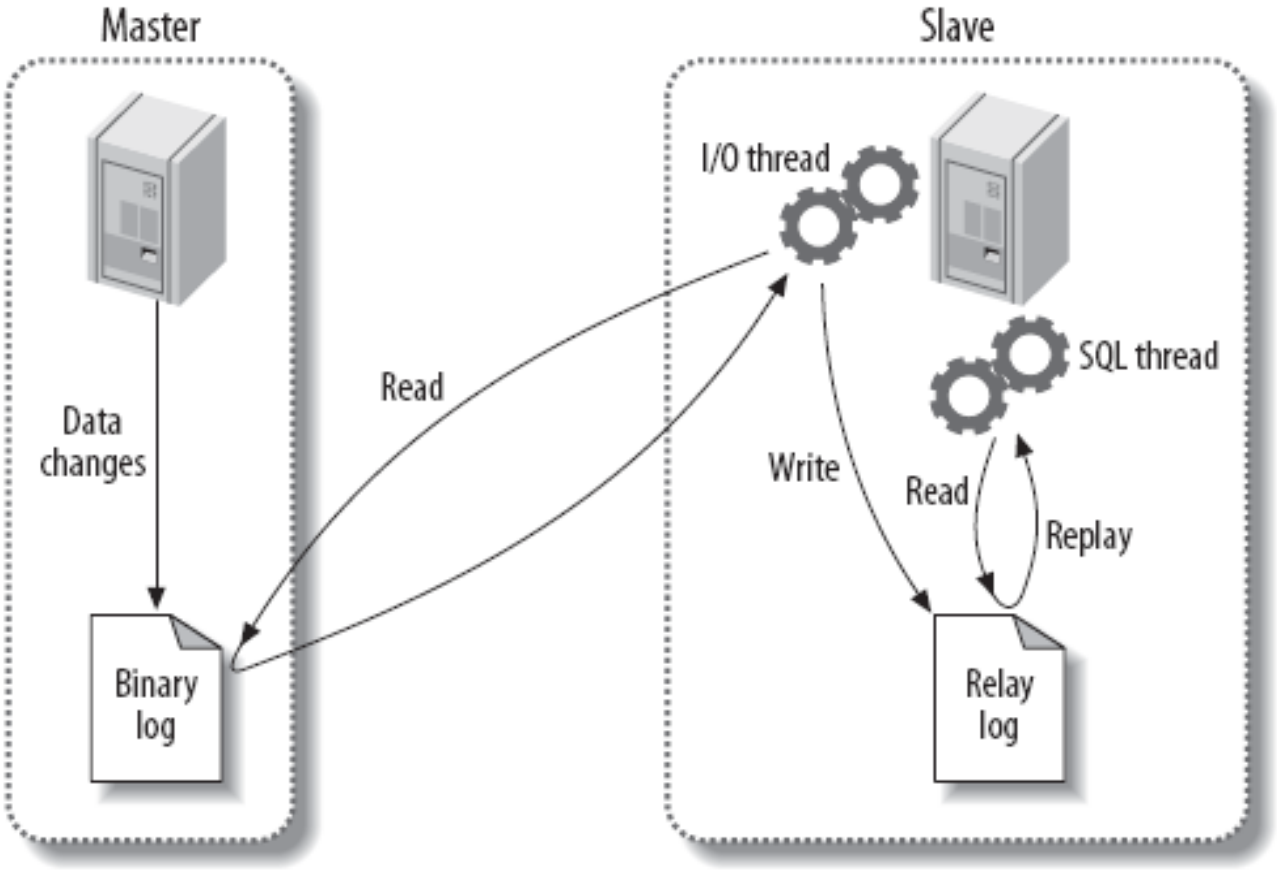
Replication: Master-Slave

- MySQL Master Server
  - Changes data
  - Sends changes to slave
- MySQL Slave Server
  - Receives changes from master
  - Applies received changes to database
- Data is propagated asynchronously
  - Possibly semi-synchronously on MySQL 5.5+
- Binary Log, holds transactions that are copied...

马哥教育
www.magedu.com

Replication: Master-Slave

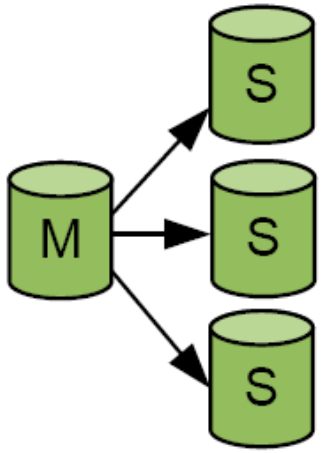


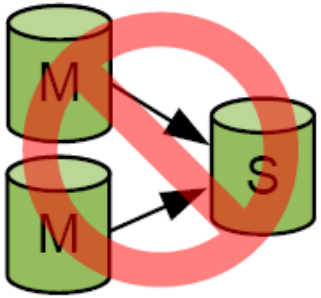

Master
Slave
I/O thread
SQL thread
Data changes
Read
Write
Read
Replay
Binary log
Relay log

Replication: Master-Slave

Server can be **master, slave or both**

Master can have **multiple slaves**

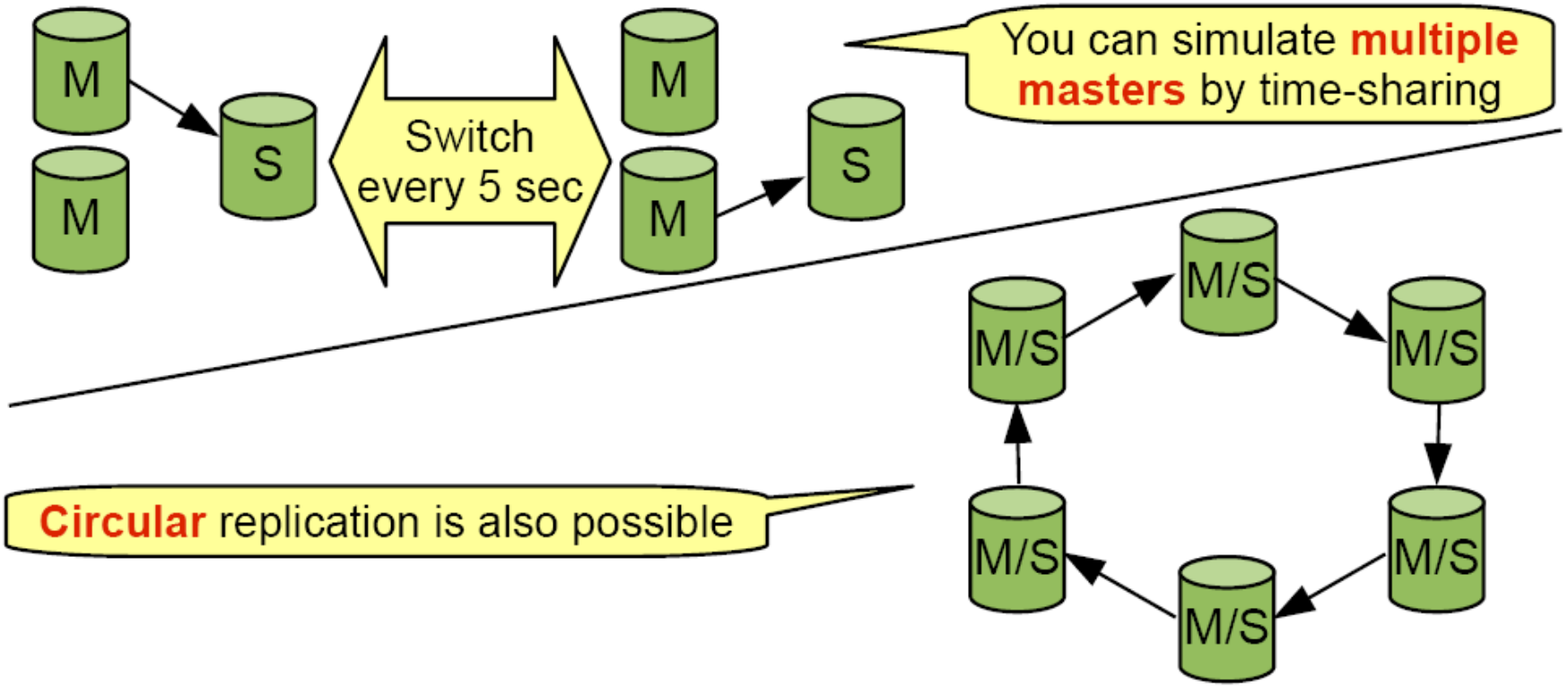




Slave can only have **one master** (at a time)

www.magedu.com

Replication: Architecture

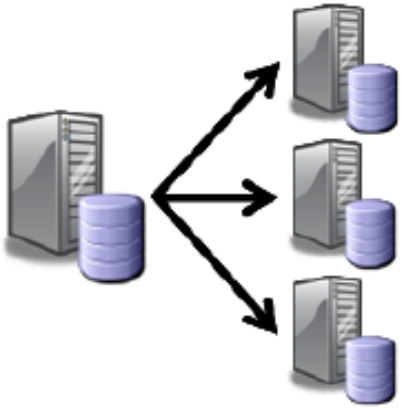


www.magedu.com

Replication Topologies

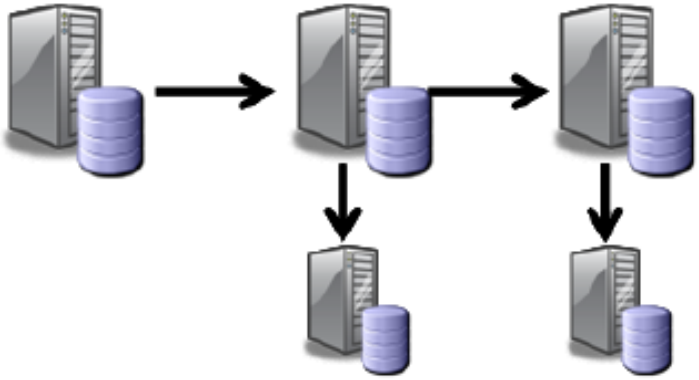
Master to Slave

Master to Multiple Slaves

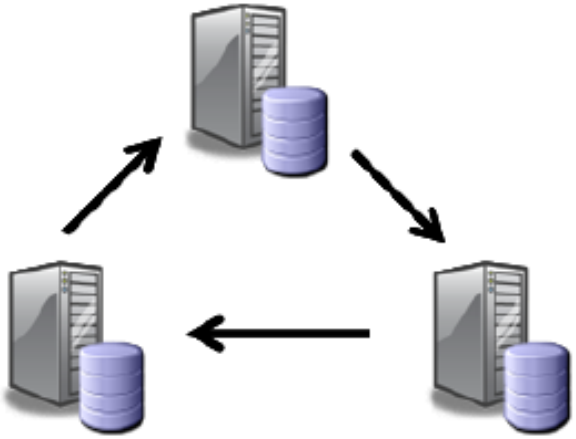


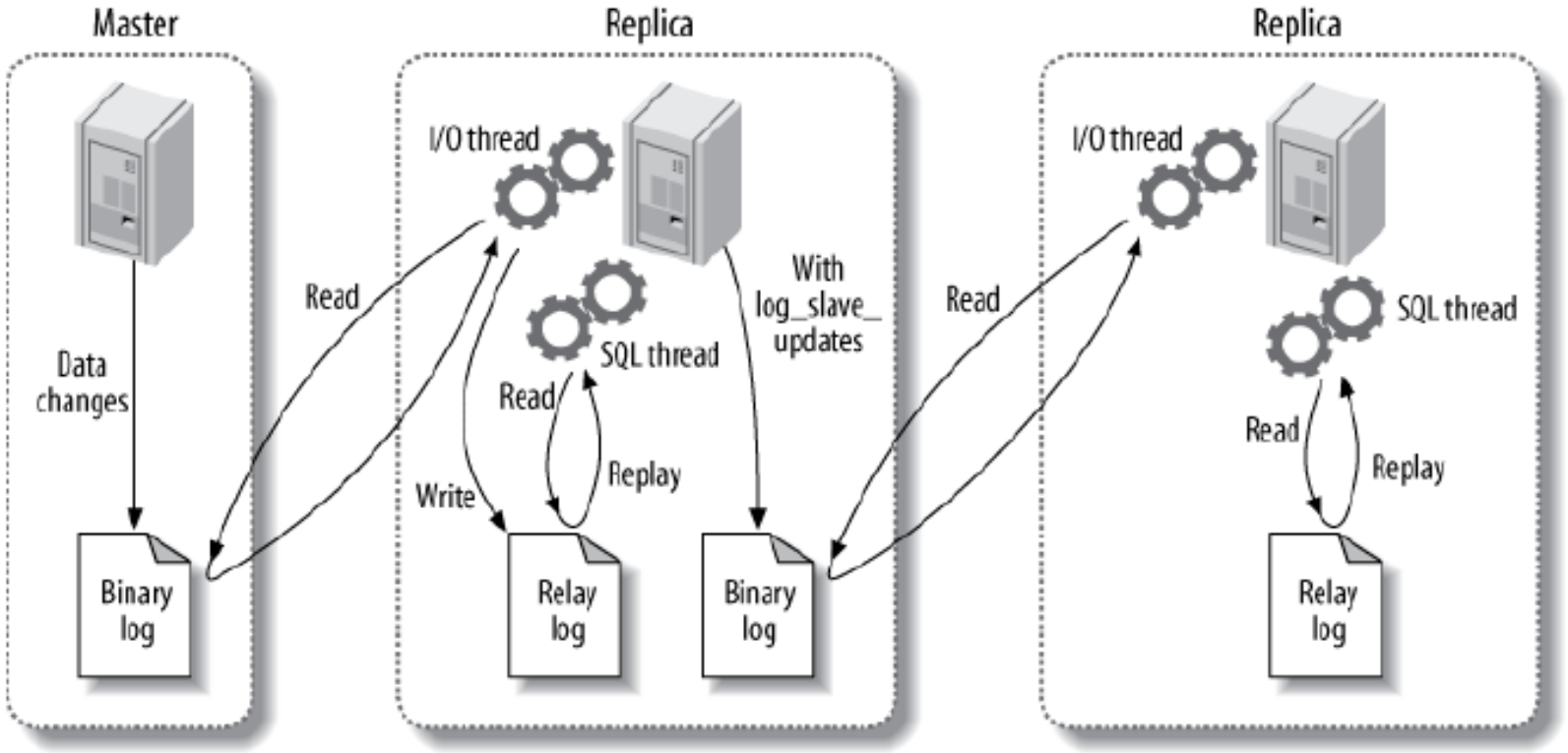
Multi-Master

Master to Slave(s) to Slave(s)



Multi-Master Ring

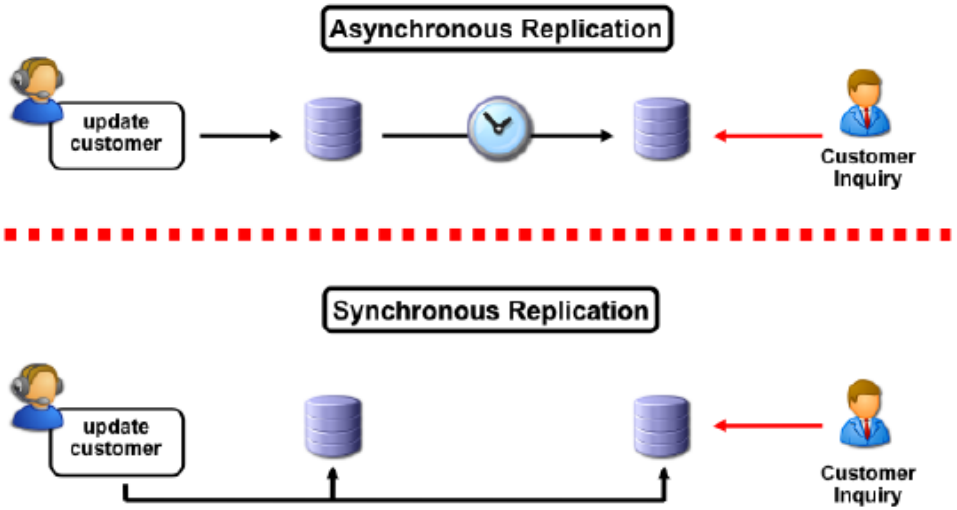




www.magedu.com

Replication Modes and Data Consistency

- Asynchronous Replication
  - By default, MySQL is asynchronous.
  - Updates are committed to the database on the master and then relayed to the slave where they are also applied
  - The master does not wait for the slave to receive the update, and so is able to continue processing further write operations without as it waits for acknowl- edgement from the slav-

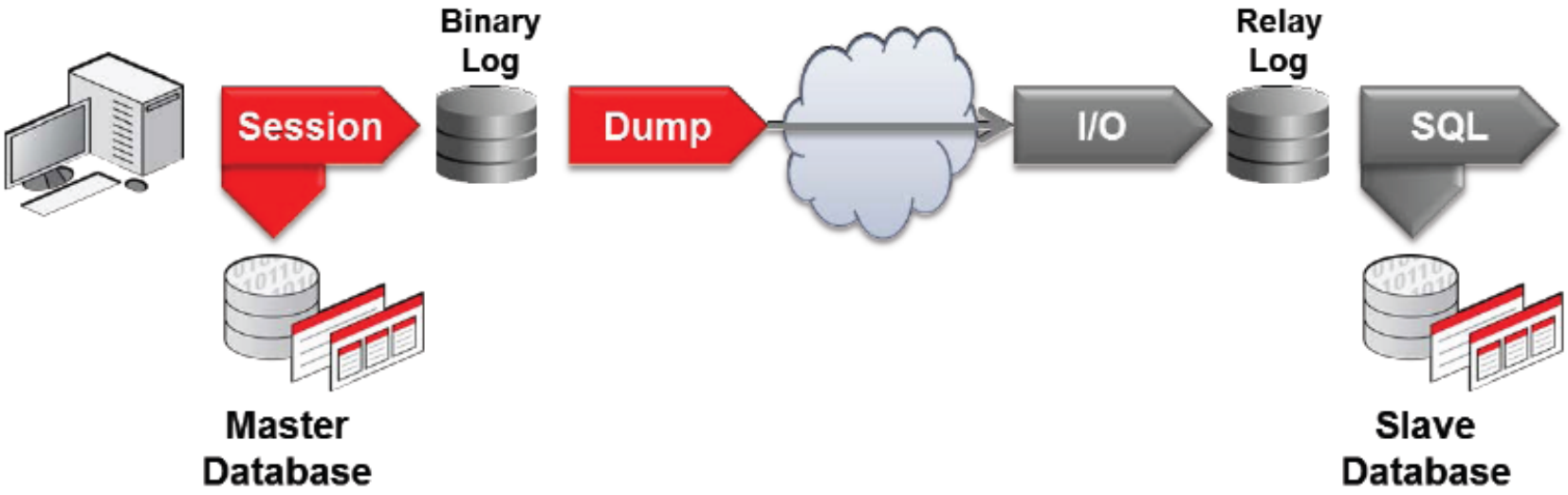


www.mag

- Semi-Synchronous Replication
  - Using semi-synchronous replication, a commit is returned to the client only when a slave has received the update, or a timeout occurs
  - Therefore it is assured that the data exists on the master and at least one slave (note that the slave will have received the update but not necessarily applied it when a commit is returned to the master)

马哥教育
www.magedu.com

MySQL Replication workflow

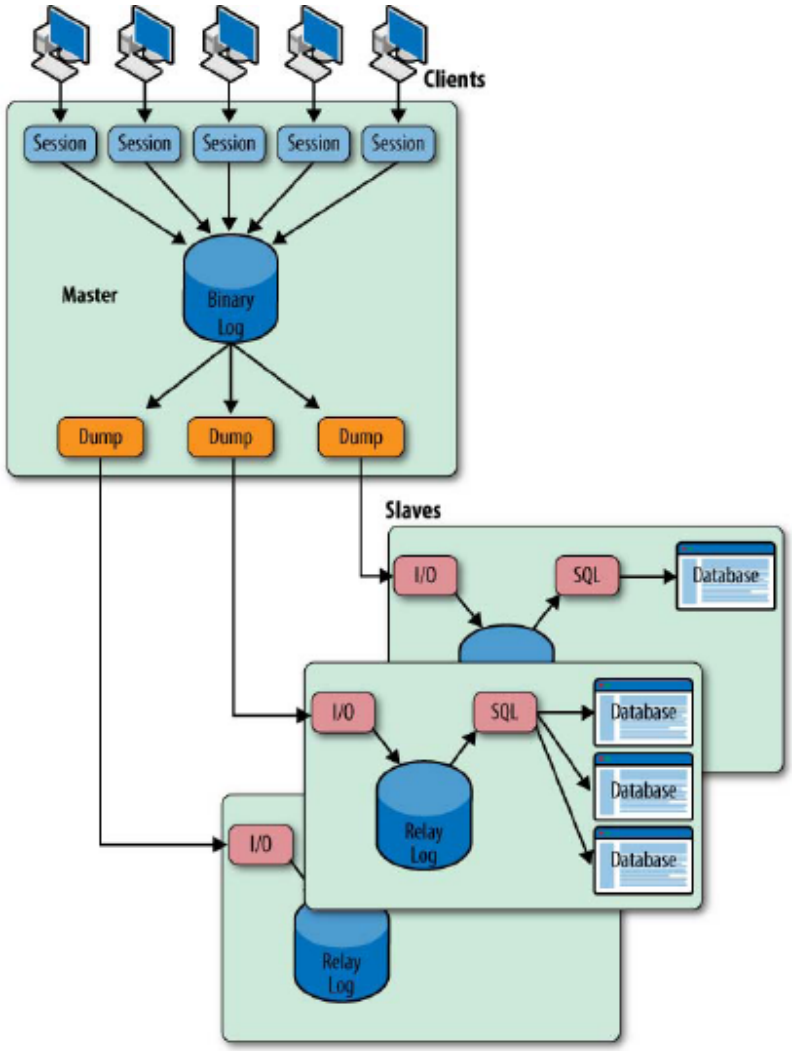


马哥教育
www.magedu.com

Replication Threads

- Binlog Dump Thread
  - To send the binary log contents to the slave
  - A master that has multiple slaves "attached" to it creates one binlog dump thread for each currently connected slave, with each slave having its own I/O and SQL threads
- Slave I/O Thread
  - A START SLAVE statement creates an I/O thread, which connects to the master and asks it to send the updates recorded in its binary logs
- Slave SQL Thread
  - To read the relay logs that were written by the slave I/O thread and executes the updates contained in the relay logs
  - If using multi-threaded slaves, multiple SQL threads will be created on the slave

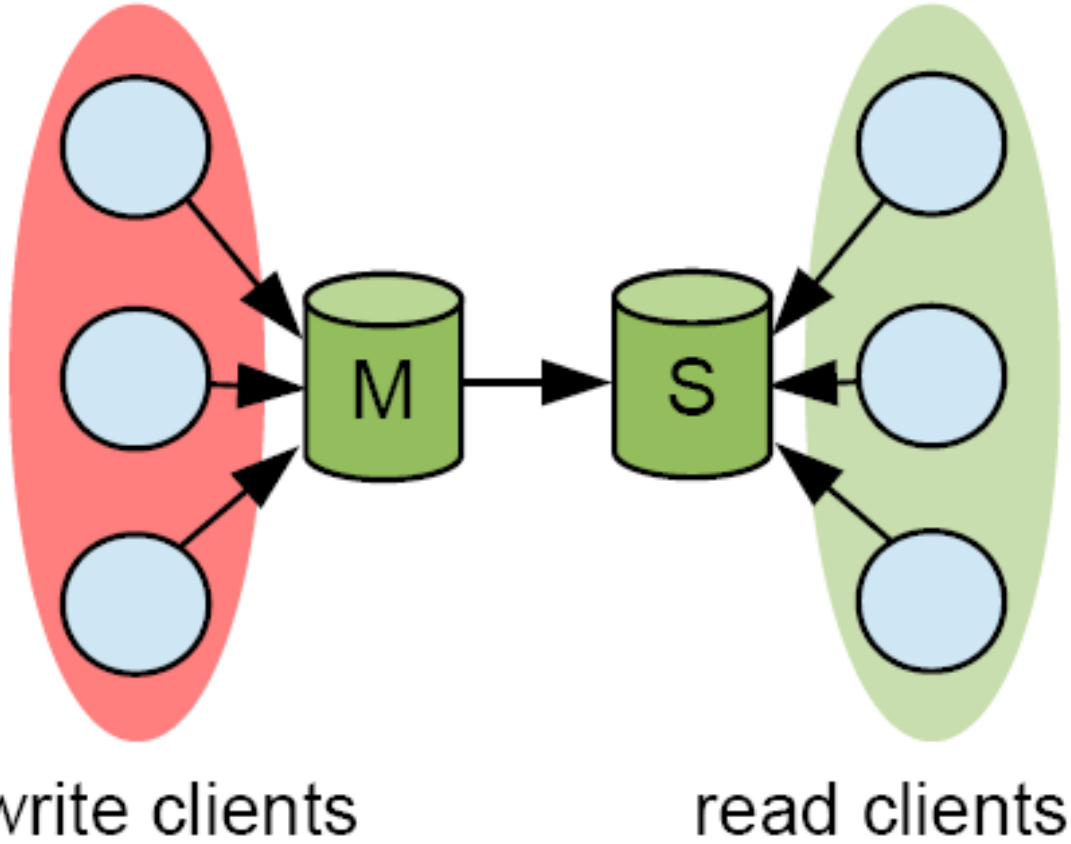
马哥教育
www.magedu.com


Clients
Session
Session
Session
Session
Session
Master
Binary Log
Dump
Dump
Dump
Slaves
I/O
SQL
Database
I/O
SQL
Database
Database
Database
Relay Log
I/O
Relay Log

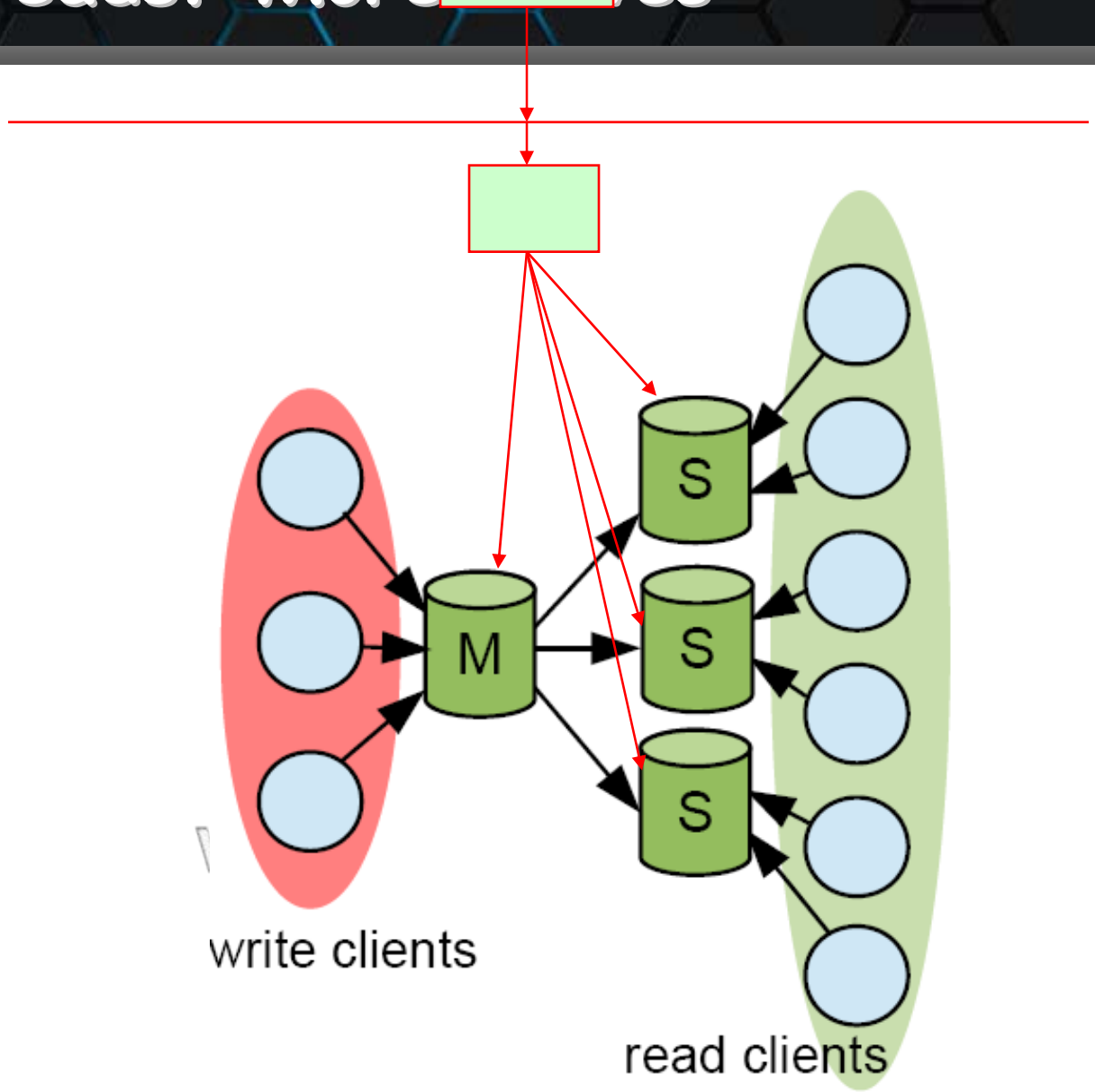
马哥教育

Common Use Cases

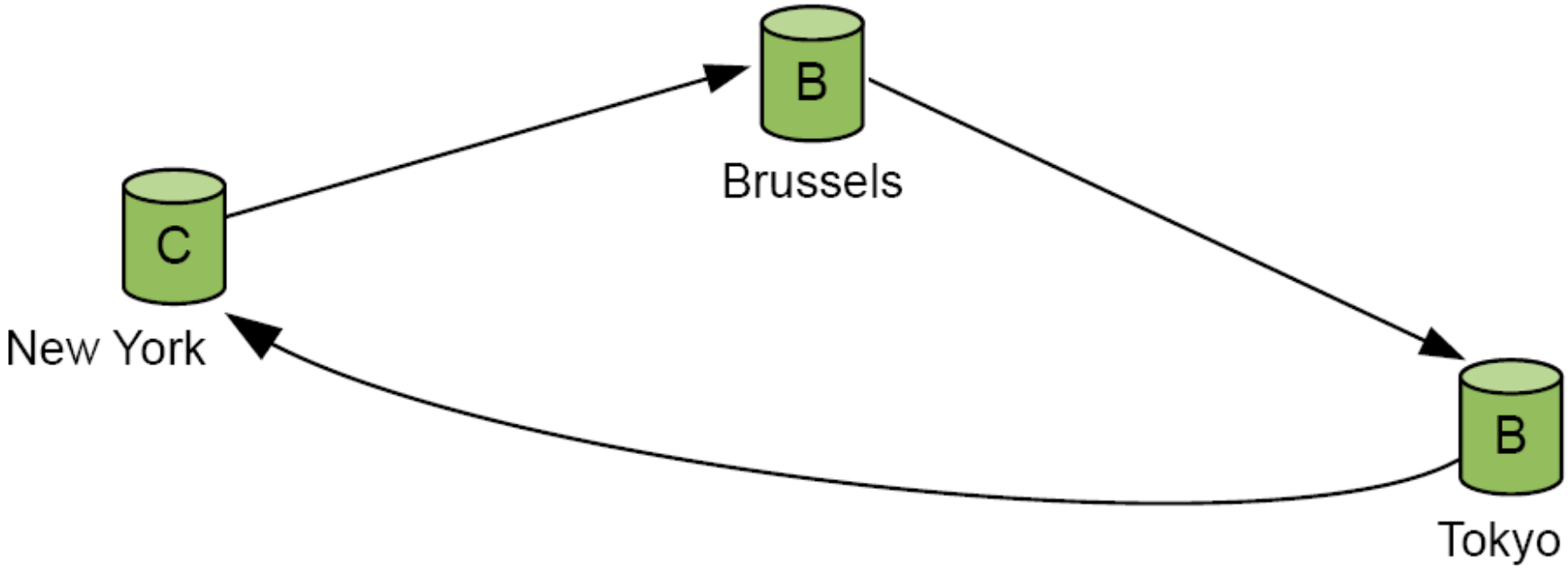
Read Scale-out



More reads? More [redacted]ves

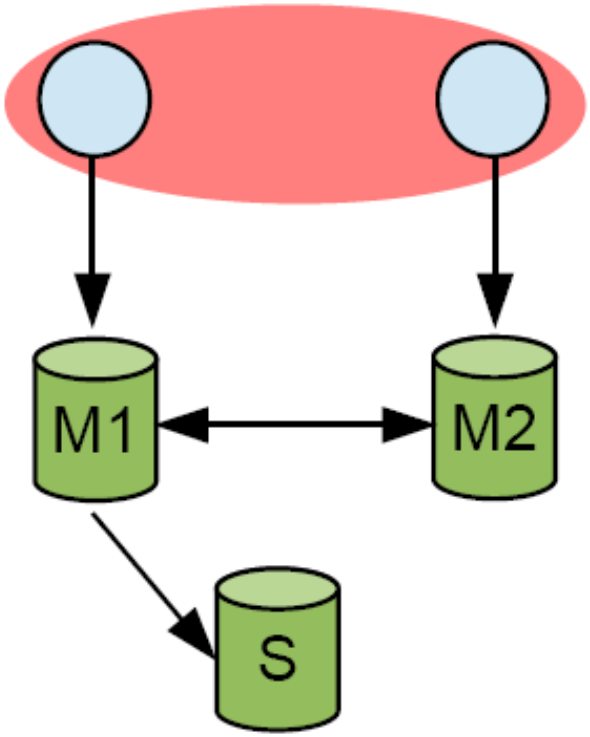


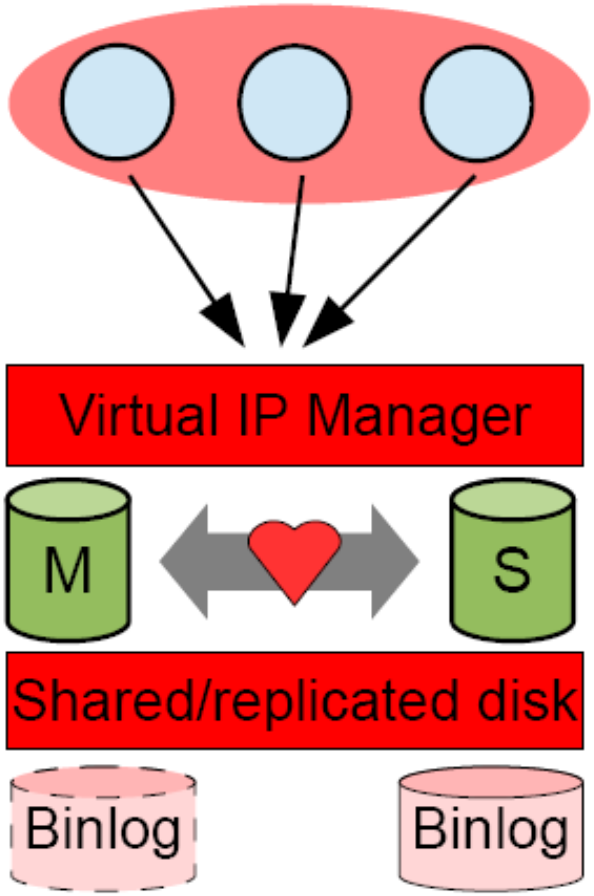
Disaster Recovery



马哥教育
www.magedu.com

High Availability

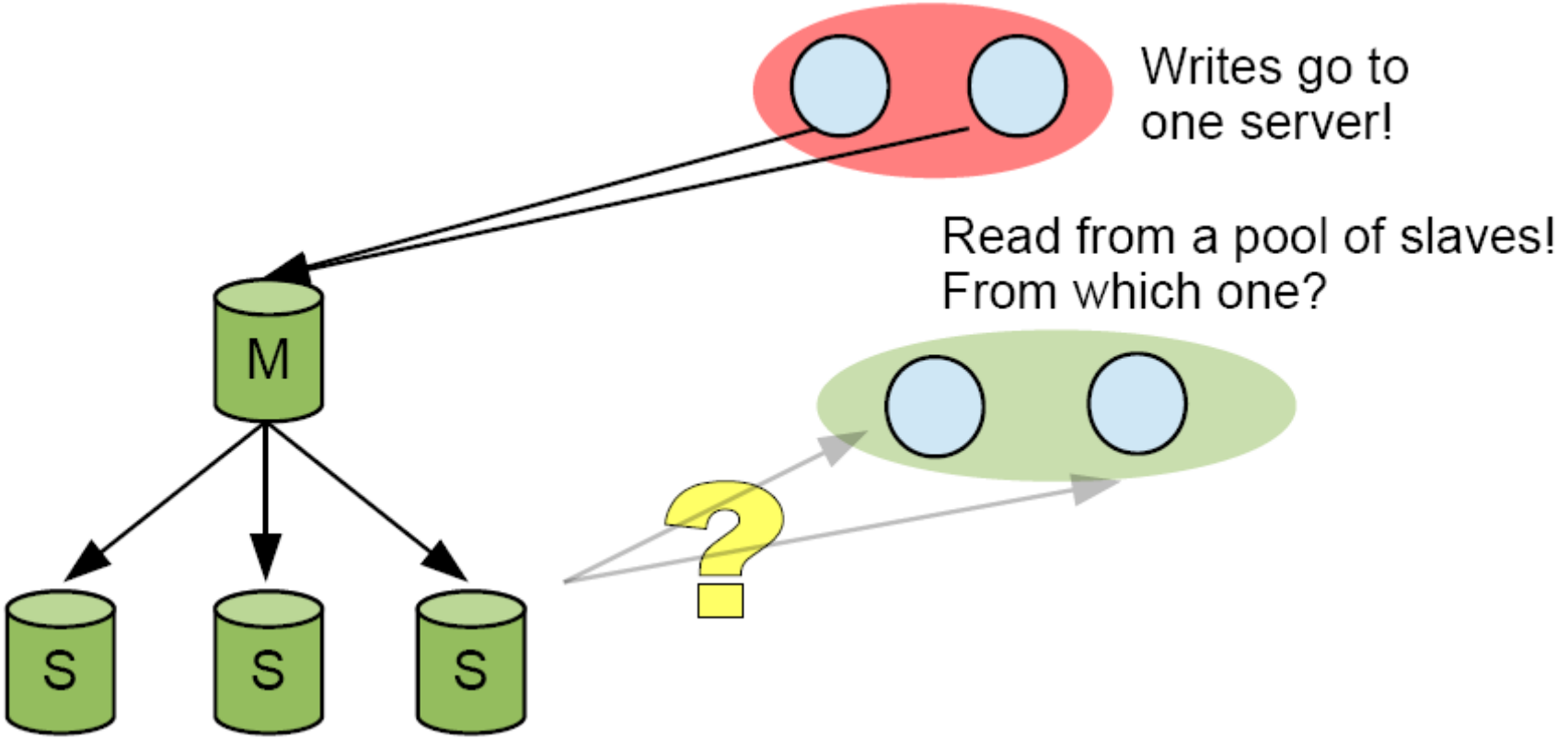




More Advanced Use Cases

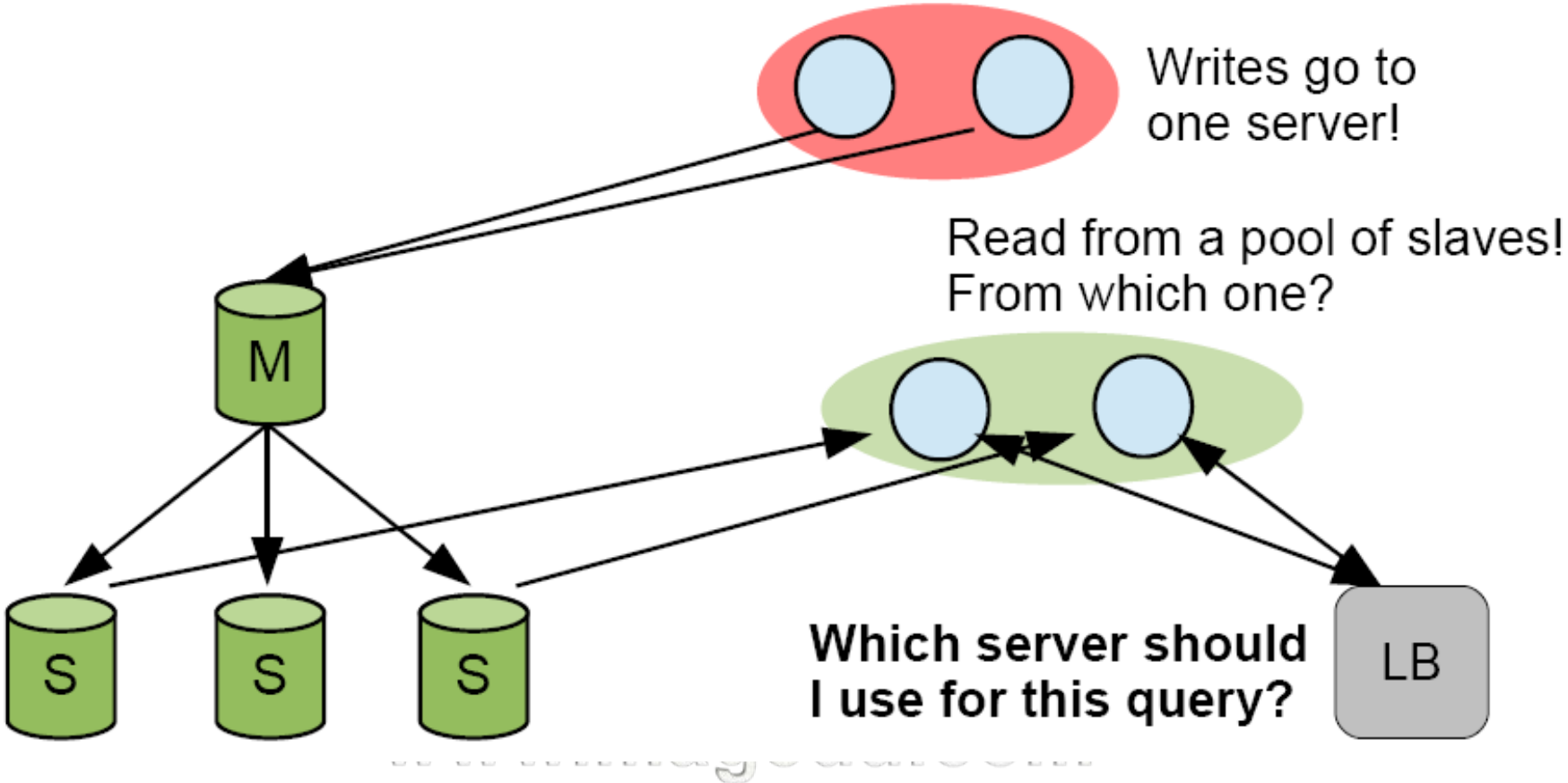
马哥教育

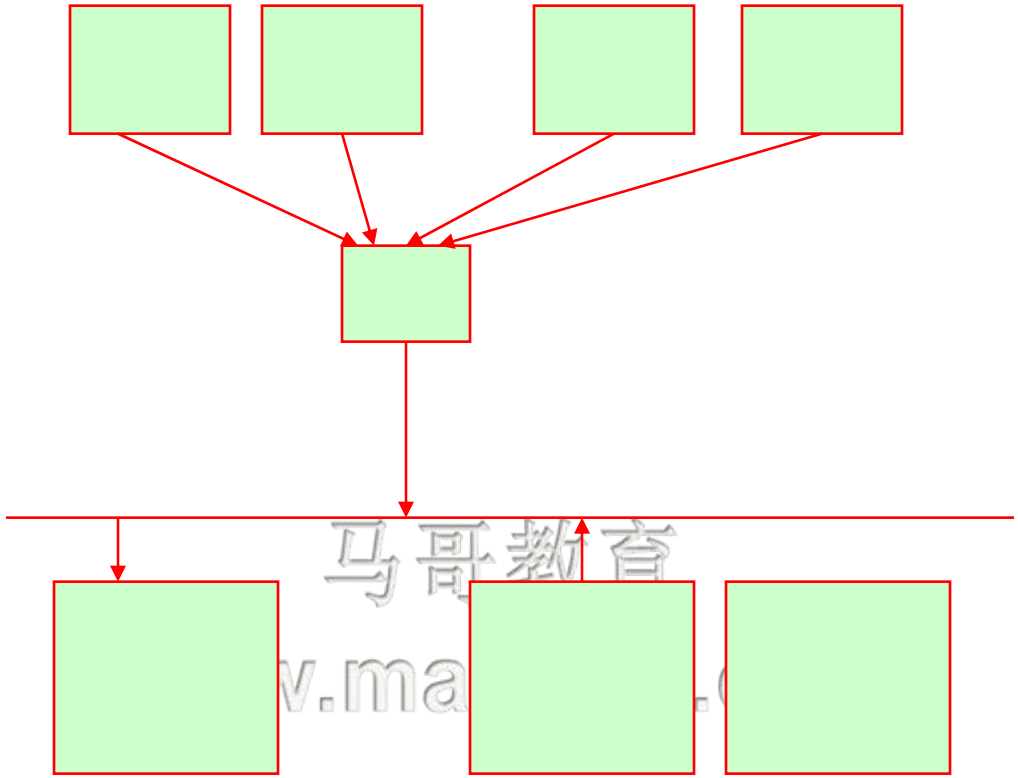
Load Balancing and Query Splitting



www.magedu.com

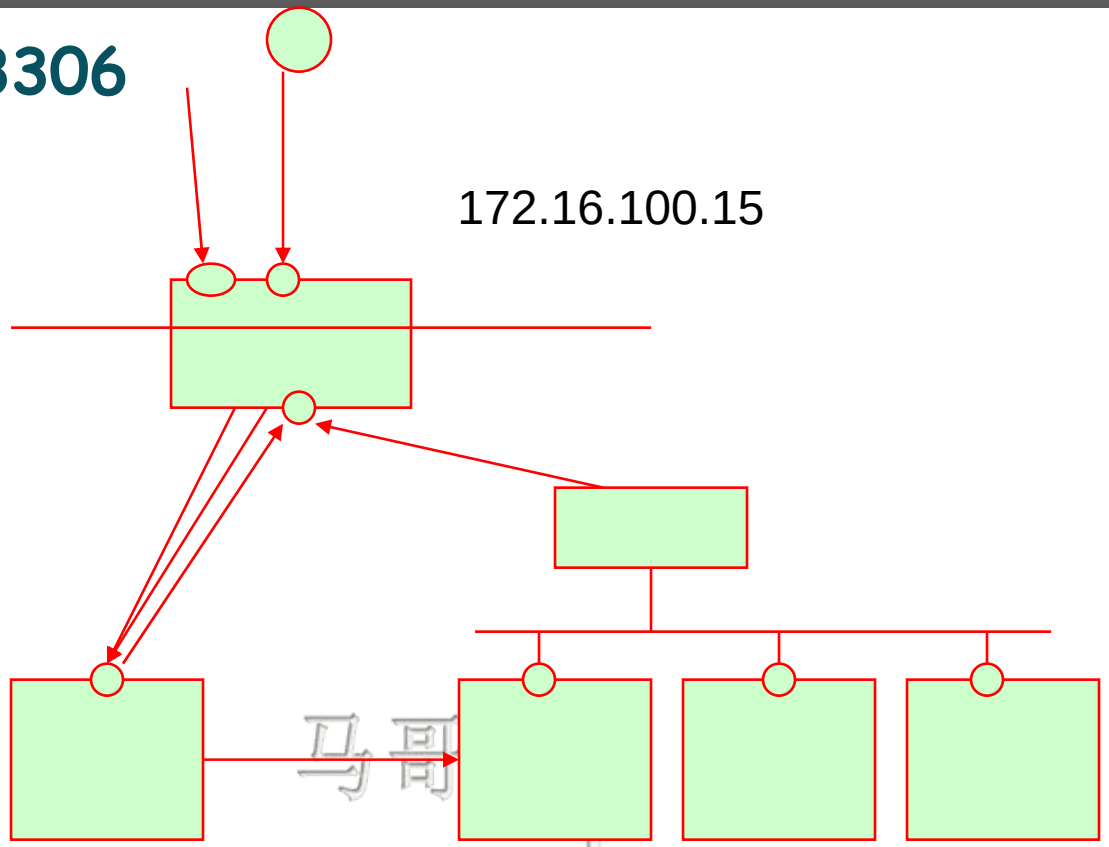
Load Balancing and Query Splitting





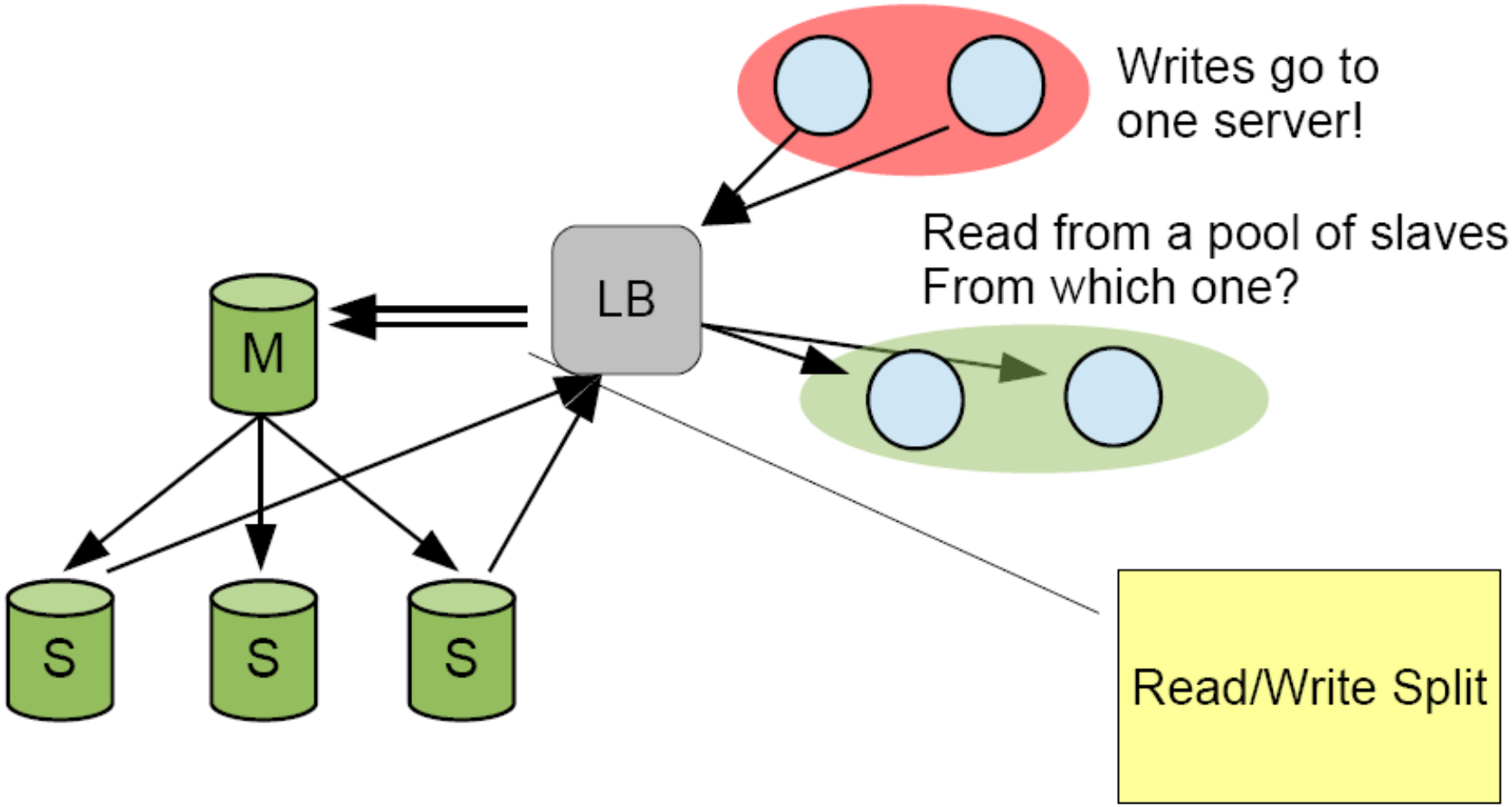
马哥教育
www.ma

- 8066 → 3306
- 9066



马哥
www.magedu.com

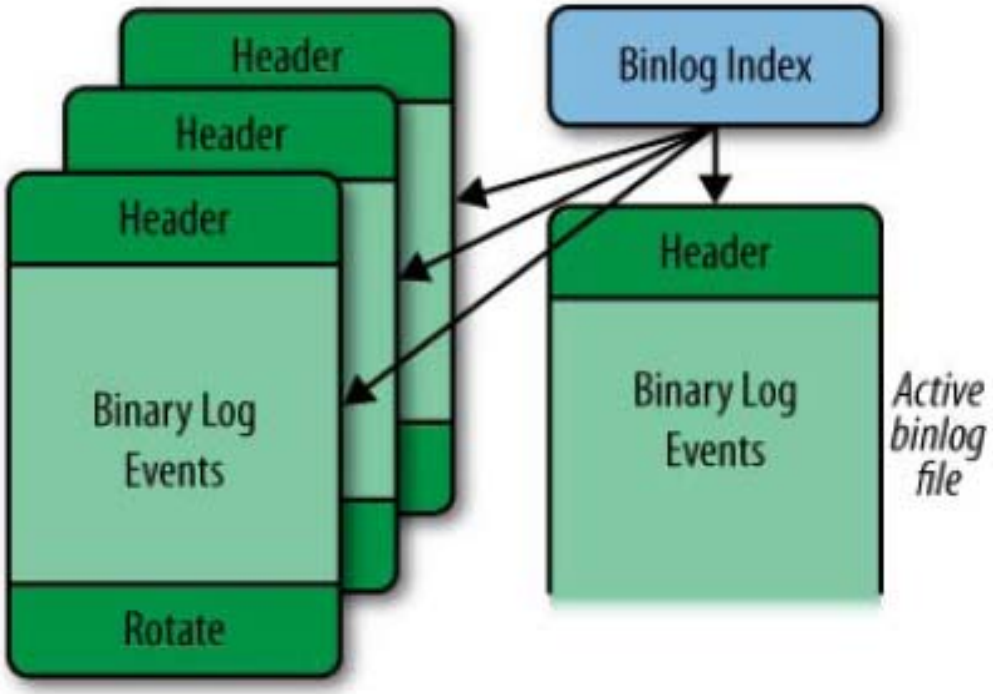
Load Balancing and Query Splitting



马哥教育

Replication Configuration

The Binary Log's Structure and Content



www.magedu.com

Basic Steps in Replication

- Setting up basic replication can be summarized in three easy steps:
  - Configure one server to be a master
  - Configure one server to be a slave
  - Connect the slave to the master
- This assumes that many default settings will suffice, which is true if you've just installed the master and slave and they have the same data

马哥教育
www.magedu.com

Configuring the Master

- To configure a server so that it can act as master, ensure the server has an active binary log and a unique server ID
  - The server ID is used to distinguish two servers from each other
  - To set up the binary log and server ID, you have to take the server down and add the log-bin, log-bin-index, and server-id options to the *my.cnf* configuration file

```
[mysqld]
user            = mysql
pid-file        = /var/run/mysqld/mysqld.pid
socket          = /var/run/mysqld/mysqld.sock
port            = 3306
basedir         = /usr
datadir         = /var/lib/mysql
tmpdir          = /tmp
log-bin         = master-bin
log-bin-index   = master-bin.index
server-id       = 1
```

- The log-bin option gives the base name for all the files created by the binary log
  - It is not necessary to give a name in the log-bin option. The default value is hostname-bin
  - It is a good idea to create a name that is unique for the server and not tied to the machine the server is running on, since it can be confusing to work with a series of binlog files that suddenly change name midstream
- The log-bin-index option gives the name of the binary log index file, which keeps a list of all binlog files

马哥教育
www.magedu.com

- If no value is provided for log-bin-index, the default value will be the same base name as for the binlog files
- Each server is identified by a unique server ID, so if a slave connects to the master and has the same server-id as the master, an error will be generated indicating that the master and the slave have the same server ID

马哥教育
www.magedu.com

Creating Replication Accounts

- MySQL has a few special privileges that let the replication processes run
- Must create a user account on the master and give it the proper privileges, so the I/O thread can connect as that user and read the master's binary log

```
mysql> GRANT REPLICATION SLAVE, REPLICATION CLIENT ON *.*
    -> TO repl@'192.168.0.%' IDENTIFIED BY 'p4ssword';
```

马哥教育
www.magedu.com

Configuring the Slave

- As with the master server, you need to assign each slave a unique server ID
- You may also want to consider adding the names of the relay log and the relay log index files to the my.cnf file using the options relay-log and relay-log-index

```
[mysqld]
user            = mysql
pid-file        = /var/run/mysqld/mysqld.pid
socket          = /var/run/mysqld/mysqld.sock
port            = 3306
basedir         = /usr
datadir         = /var/lib/mysql
tmpdir          = /tmp
server-id       = 2
relay-log-index = slave-relay-bin.index
relay-log       = slave-relay-bin
```

Connecting the Master and Slave

- Now you can perform the final step in setting up basic replication: directing the slave to the master so that it knows where to replicate
- Need four pieces of information about the master:
  - A hostname
  - A port number
  - A user account on the master with replication slave privileges
  - A password for the user account

马哥教育
www.magedu.com

- You should not use the my.cnf file for this; instead, use the CHANGE MASTER TO statement. This statement replaces the corresponding my.cnf settings completely
- It also lets you point the slave at a different master in the future, without stopping the server

```
mysql> CHANGE MASTER TO MASTER_HOST='server1',
    -> MASTER_USER='repl',
    -> MASTER_PASSWORD='p4ssword',

slave> START SLAVE;
Query OK, 0 rows affected (0.15 sec)
```

Monitoring Replication


```
mysql> SHOW MASTER STATUS;
+------------------+----------+--------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------+----------+--------------+------------------+
| mysql-bin.000001 |       98 |              |                  |
+------------------+----------+--------------+------------------+
1 row in set (0.00 sec)

mysql> SHOW SLAVE STATUS\G
*************************** 1. row ***************************
             Slave_IO_State: Waiting for master to send event
                Master_Host: server1
                Master_User: repl
                Master_Port: 3306
              Connect_Retry: 60
            Master_Log_File: mysql-bin.000001
        Read_Master_Log_Pos: 164
             Relay_Log_File: mysql-relay-bin.000001
              Relay_Log_Pos: 164
      Relay_Master_Log_File: mysql-bin.000001
           Slave_IO_Running: Yes
          Slave_SQL_Running: Yes
                            ...omitted...
```

```
mysql> SHOW PROCESSLIST\G
*************************** 1. row ***************************
     Id: 1
   User: system user
   Host:
     db: NULL
Command: Connect
   Time: 611116
  State: Waiting for master to send event
   Info: NULL
*************************** 2. row ***************************
     Id: 2
   User: system user
   Host:
     db: NULL
Command: Connect
   Time: 33
  State: Has read all relay log; waiting for the slave I/O thread to update it
   Info: NULL
```

Watching Replication in Action


```
master> SHOW BINLOG EVENTS\G
*************************** 1. row ***************************
   Log_name: master-bin.000001
        Pos: 4
 Event_type: Format_desc
  Server_id: 1
End_log_pos: 106
       Info: Server ver: 5.1.33, Binlog ver: 4
```

- Event_type
  - This is the type of the event
- Server_id
  - This is the server ID of the server that created the event
- Log_name
  - The name of the file that stores the event

马哥教育
www.magedu.com

- Pos
  - This is the position of the file where the event starts; that is, it's the first byte of the event
- End_log_pos
  - This gives the position in the file where the event ends and the next event starts
- Info
  - This is human-readable text with information about the event

马哥教育
www.magedu.com

- One binlog file is the active binlog file
  - This is the file that is currently being written to
- Each binlog file starts with a format description event and ends with a rotate event
  - The format description log event contains, among other things, the version of the server that produced the file and general information about the server and binary log
  - The rotate event tells where the binary log continues by giving the filename of the next file in the sequence

马哥教育
www.magedu.com

Creating New Slaves

- When we configured the slave, we provided no information about where to start replication, so the slave will start reading the binary logs on the master from the beginning
- That's clearly not a very good idea if the master has been running for some time:
  - In addition to making the slave replay quite a lot of events just to ramp up, you might not be able to obtain the necessary logs, because they might have been stored somewhere else for safekeeping and removed from the master

马哥教育
www.magedu.com

- The CHANGE MASTER TO command has two parameters that will help us here: MASTER_LOG_FILE and MASTER_LOG_POS
- You can use these to specify the binlog position at which the master should start sending events instead of starting from the beginning

马哥教育
www.magedu.com

- Using these parameters to CHANGE MASTER TO, we can bootstrap a slave using the following steps:
  - Configure the new slave
  - Make a backup of the master (or of a slave that has been replicating the master)
  - Write down the binlog position that corresponds to this backup (in other words, the position following the last event leading up to the master's current state)
  - Restore the backup on the new slave
  - Configure the slave to start replication from this position

马哥教育
www.magedu.com

Cloning the Master

- Since the master is probably running and has a lot of tables in the cache, it is necessary to flush all tables and lock the database to prevent changes before checking the binlog position
- You can do this using the FLUSH TABLES WITH READ LOCK command:

```
master> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0.02 sec)
```

www.magedu.com

- Once the database is locked, you are ready to create a backup and note the binlog position
- Since no changes are occurring on the master, the SHOW MASTER STATUS command will correctly reveal the current file and position in the binary log
- The position of the next event to write is master-bin.000042, 456552, which is where replication should start, since everything before this point will be in the backup

马哥教育

```
master> SHOW MASTER STATUS\G
*************************** 1. row ***************************
            File: master-bin.000042
        Position: 456552
    Binlog Do DB:
Binlog_Ignore_DB:
1 row in set (0.00 sec)
```

- Once you have jotted down the binlog position, you can create your backup
  - mysqldump --all-databases --host=master-1 >backup.sql
- Since you now have a faithful copy of the master, you can unlock the tables of the database on the master and allow it to continue processing queries
  - UNLOCK TABLES;

马哥教育
www.magedu.com

- Next, restore the backup on the slave using the mysql utility:
  - mysql --host=slave-1 <backup.sql
- You have now restored the backup of the master on the slave and can start the slave
- Recalling the binlog position of the master that you wrote down previously, configure the slave using CHANGE MASTER TO and start the slave

马哥教育
www.magedu.com

```
slave> CHANGE MASTER TO
    ->   MASTER_HOST = 'master-1',
    ->   MASTER_PORT = 3306,
    ->   MASTER_USER = 'slave-1',
    ->   MASTER_PASSWORD = 'xyzzy',
    ->   MASTER_LOG_FILE = 'master-bin.000042',
    ->   MASTER_LOG_POS = 456552;
Query OK, 0 rows affected (0.00 sec)

slave> START SLAVE;
Query OK, 0 rows affected (0.25 sec)
```

www.magedu.com

Common Options for the CHANGE MASTER TO Command

Parameter	What It Means
MASTER_HOST	Host name for the master server
MASTER_USER	Slave name to use when connecting to the master
MASTER_PASSWORD	Slave's password to connection to master
MASTER_PORT	Port number to connect to master
MASTER_LOG_FILE	Name of master's binary log file from which to start reading when replication begins
MASTER_LOG_POS	Position in the master's binary log file from which to start reading when replication begins
MASTER_CONNECT_RETRY	Number of seconds to wait between connection attempts
RELAY_LOG_FILE	Name of the slave relay log from which to begin execution when replication begins
RELAY_LOG_POS	Position in slave relay log from which to begin execution when replication begins
MASTER_SSL	Whether to connect to the master server using SSL

Binary Log Filters on Master

- It is possible to filter out statements from the binary log using two options: binlog-do-db and binlog-ignore-db
  - The binlog-do-db is used when you want to filter only statements belonging to a certain database
  - The binlog-ignore-db is used when you want to ignore a certain database but replicate all other databases
  - These options can be given multiple times

马哥教育
agedu.com

```
[mysqld]
binlog-ignore-db=one_db
binlog-ignore-db=two_db
```

Binary Log Filters on Slave

- replicate-do-db=*db*
  - If the current database of the statement is *db*, execute the statement
- replicate-ignore-db=db
  - If the current database of the statement is *db*, discard the statement
- replicate-do-table=table *and* replicate-wild-do-table=db_pattern.tbl_pattern
  - If the name of the table being updated is *table* or matches the pattern, execute updates to the table
- replicate-ignore-table=table *and* replicate-wild-ignore-table=db_pattern.tbl_pattern
  - If the name of the table being updated is *table* or matches the pattern, discard updates to the table

马哥教育
www.magedu.com

Security and the Binary Log

- A user with REPLICATION SLAVE privileges has privileges to read everything that occurs on the master and should therefore be secured so that the account cannot be compromised
  - Make it impossible to log into the account from outside the firewall
  - Log all attempts to log into the account, and place the log on a separate secure server
  - Encrypt the connection between the master and the slave using, for example, MySQL's built-in SSL (Secure Sockets Layer) support

马哥教育
www.magedu.com

Binary Log Options and Variables

- expire-log-days=*days*
  - The number of days that binlog files should be kept.
  - By default this option is 0, meaning that binlog files are never removed
- log-bin[=*basename*]
  - The binary log is turned on by adding the log-bin option in the *my.cnf* file
  - This option gives a base name for the binlog files; that is, the portion of the filename before the dot
- log-bin-index[=*filename*]
  - Gives a name to the index file
  - The default is the same as the base name used for log-bin

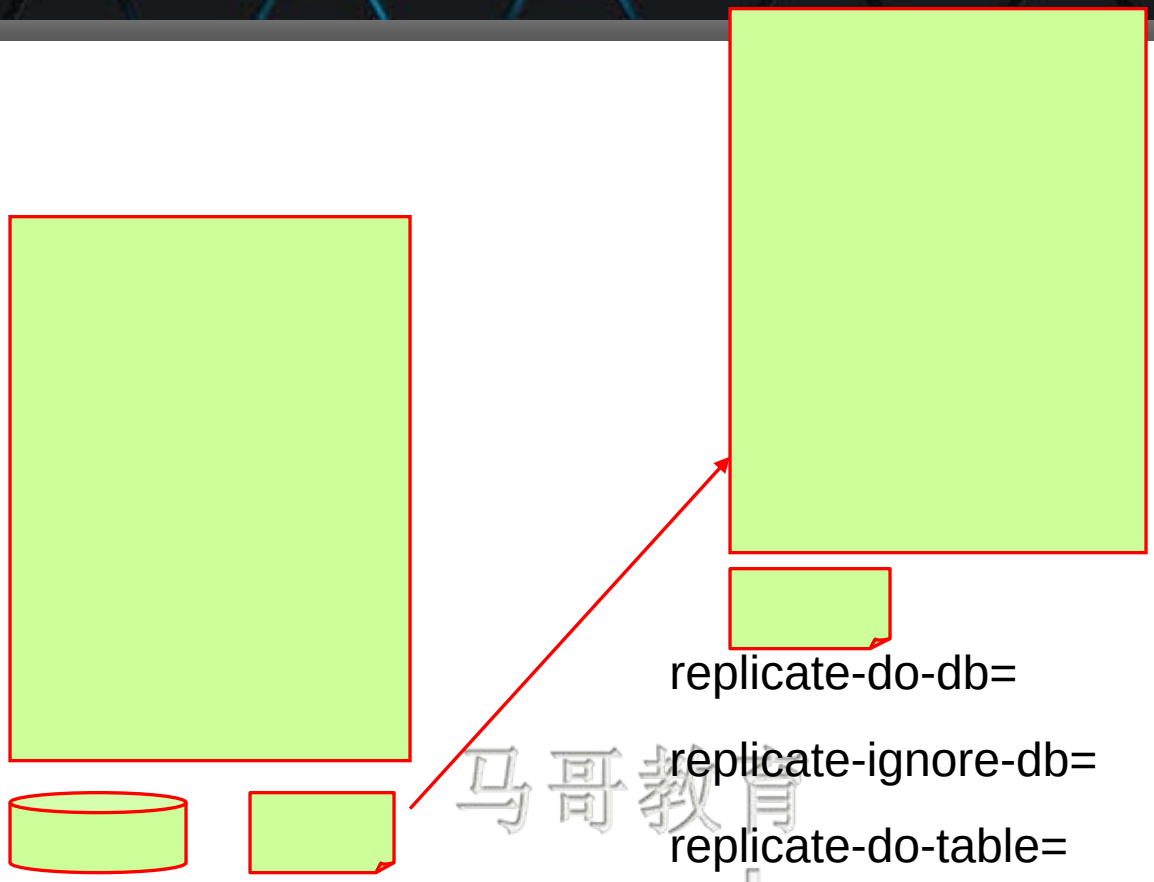
马哥教育
www.magedu.com

- log-bin-trust-function-creators
  - To disable the SUPER privilege requirement for creating stored functions
- binlog-cache-size=*bytes*
  - The size of the in-memory part of the transaction cache in bytes
  - Increasing the value of this option can improve performance if you use many large transactions
- max-binlog-cache-size=*bytes*
  - To restrict the size of each transaction in the binary log
- max-binlog-size=*bytes*
  - Specifies the size of each binlog file

马哥教育
www.magedu.com

- sync-binlog=*period*
  - Specifies how often to write the binary log to disk using *fdatasync(2)*
  - The value given is the number of transaction commits for each real call to *fdatasync(2)*
- read-only
  - Prevents any client threads—except the slave thread and users with SUPER privileges—from updating any data on the server
  - This is useful on slave servers to allow replication to proceed without data being corrupted by clients that connect to the slave

马哥教育
www.magedu.com



%, _

replicate-do-db=

replicate-ignore-db=

replicate-do-table=

replicate-ignore-table=

replicate-wild-do-table=

replicate-wild-ignore-table=

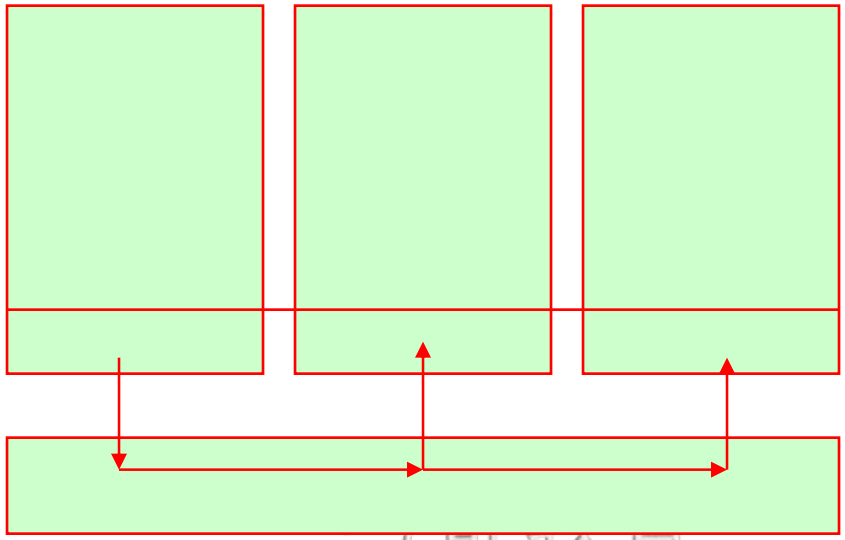
马哥教育
www.magedu.com

binlog-do-db: 仅将指定数据库的相关修改操作记入二进制日志

binlog-ignore-db

马哥教育

GTID



www.magedu.com

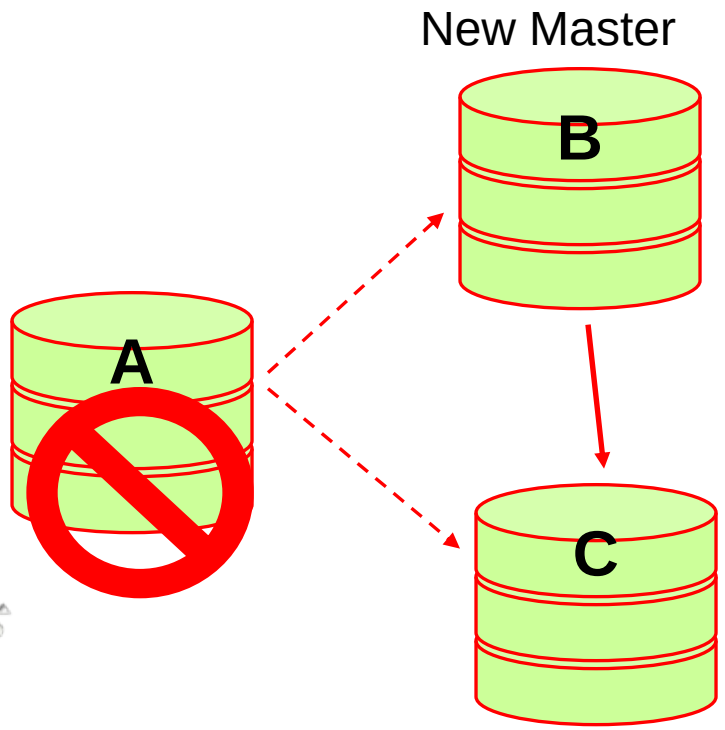
About GTID

- GTIDs are unique identifiers comprising the server UUID (of the original master) and a transaction number
- They are automatically generated as a header for every transaction and written with the transaction to the binary log
- GTIDs make it simple to track and compare replicated transactions between the master and slaves, which in turn enables simple recovery from failures of the master
- The default InnoDB storage engine must be used with GTIDs to get the full benefits of HA

马哥教育
www.magedu.com

Global Transaction Identifiers (GTIDs)

- In the event of Server "A" crashing, we need to failover to one of the slaves, promoting it to master, while the remaining server becomes a slave of that new master
- As MySQL replication is asynchronous by default, servers B and C may not have both replicated and executed the same set of transactions



马哥教育
www.magedu.com

- GTIDs apply a unique identifier to each transaction, so it is becomes easy to track when it is executed on each slave
- When the master commits a transaction, it generates a GTID consisting of two components:
  - the UUID of the server (128bits)
  - an automatically incremented integer
  - UUID:N, such as
  - 26096A54-FE03-6C14-95E7-BD3C1100AF21:1011

马哥教育
www.magedu.com

GTID

- The GTID is written to the binary log prior to the transaction
- The GTID and the transaction are replicated to the slave
- The slave does not generate a new GTID, even if the slave is configured as a master to other slaves in an n-tier (hierarchical or cascading) replication topology

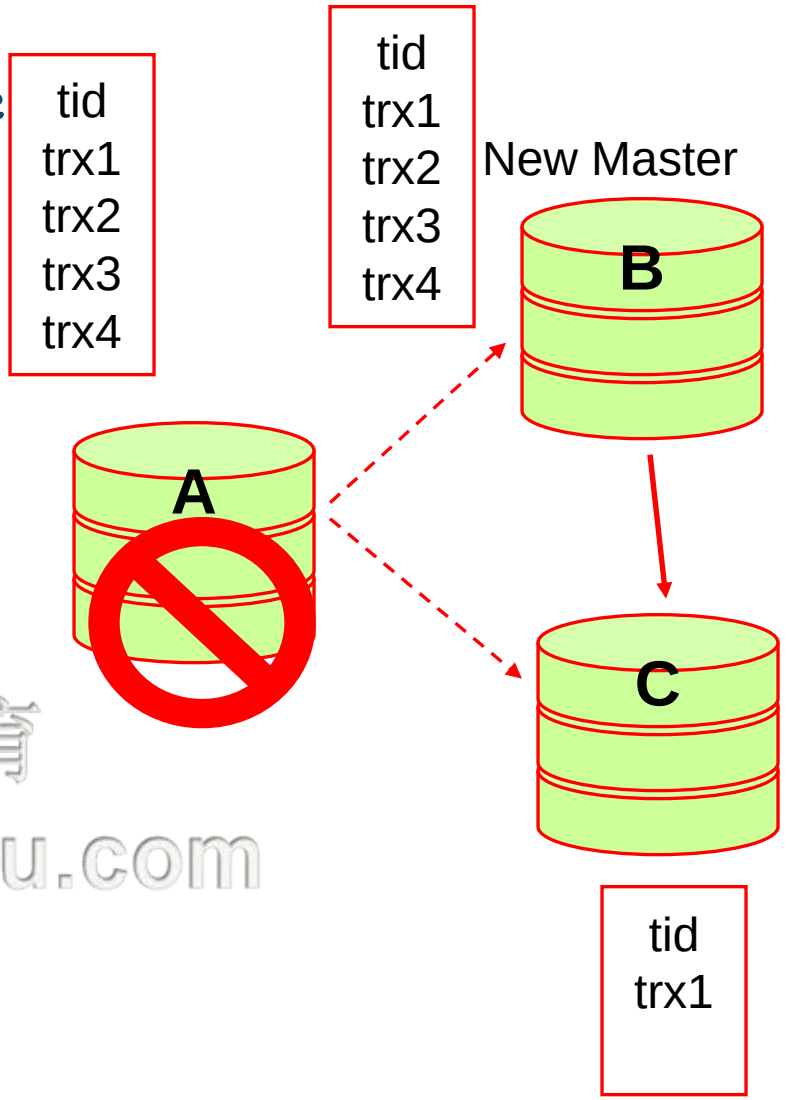
马哥教育
www.magedu.com

gtid_executed

- The set of GTIDs executed on each slave is exposed to the user in a new, read-only, global server variable, **gtid_executed**
- The variable can be used in conjunction with GTID_SUBTRACT() to determine if a slave is up to date with a master, and if not, which transactions are missing

马哥教育
www.magedu.com

- A new replication protocol was created to make the process above automatic
- When a slave connects to the master, the new protocol ensures it sends the range of GTIDs that the slave has executed and committed and requests any missing transactions
- The master then sends all other transactions, i.e. those that the slave has not yet executed

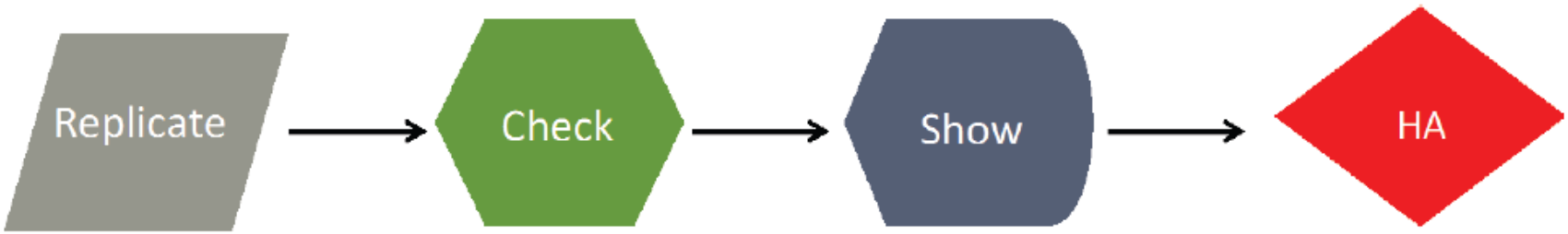


马哥教育
www.magedu.com

Utilities for Simplifying Replication

- For convenience and ease-of-use, additional MySQL Replication Utilities are provided to accelerate the provisioning of new replication clusters
- The utilities are available under the GPLv2 license, and are extendable using a supplied library
- They are designed to work with Python 2.7 and above
- https://launchpad.net/mysql-utilities

马哥教育



Utilities for Simplifying Replication

- Replication Utility: **mysqlreplicate**
  - Enables fast and simple introduction of replication slaves, the mysqlreplicate utility is used to start the replication process
  - Any GTIDs that have already been executed on the slave will be skipped
  - The utility also checks storage engine compatibility
- Replication Utility: **mysqlrplcheck**
  - Provides simple verification of deployment and fast fault resolution
  - Checks that the binlog is enabled and displays any configured exceptions
  - Checks slave access and privileges to master and slave connection status

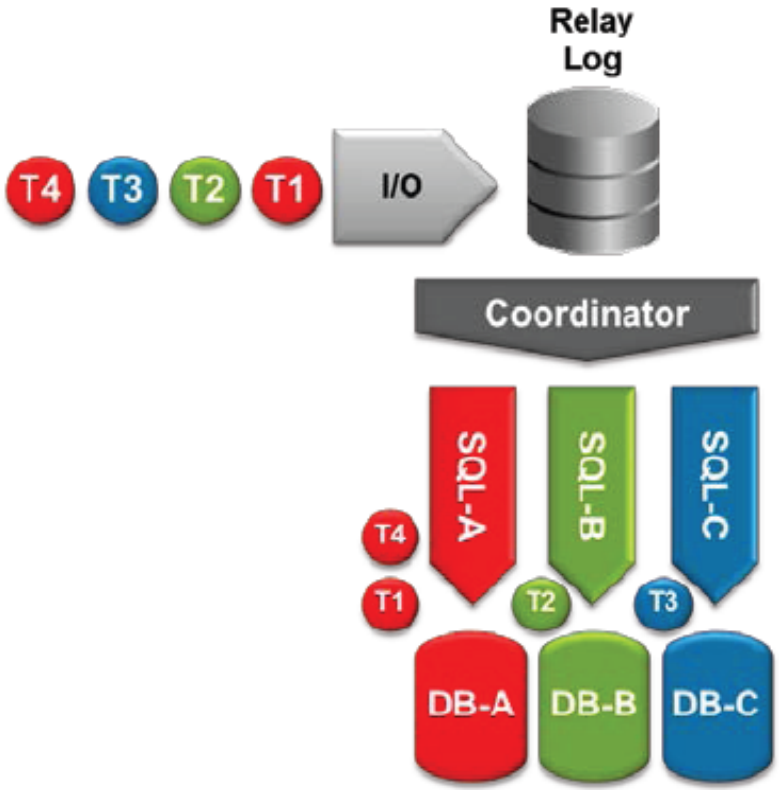
马哥教育
www.magedu.com

- Replication Utility: **mysqlrplshow**
  - Discovers and displays the replication topology on-demand
  - Shows slaves attached to each master and labels each slave with hostname and port number
- Replication Utility: **mysqlfailover**
  - Enables automatic or manual failover to a slave in the event of an outage to the master
- Replication Utility: **mysqlrpladmin**
  - If a user needs to take a master offline for scheduled maintenance, mysqlrpladmin can perform a switchover to a specific slave (called the new master)

马哥教育
www.magedu.com

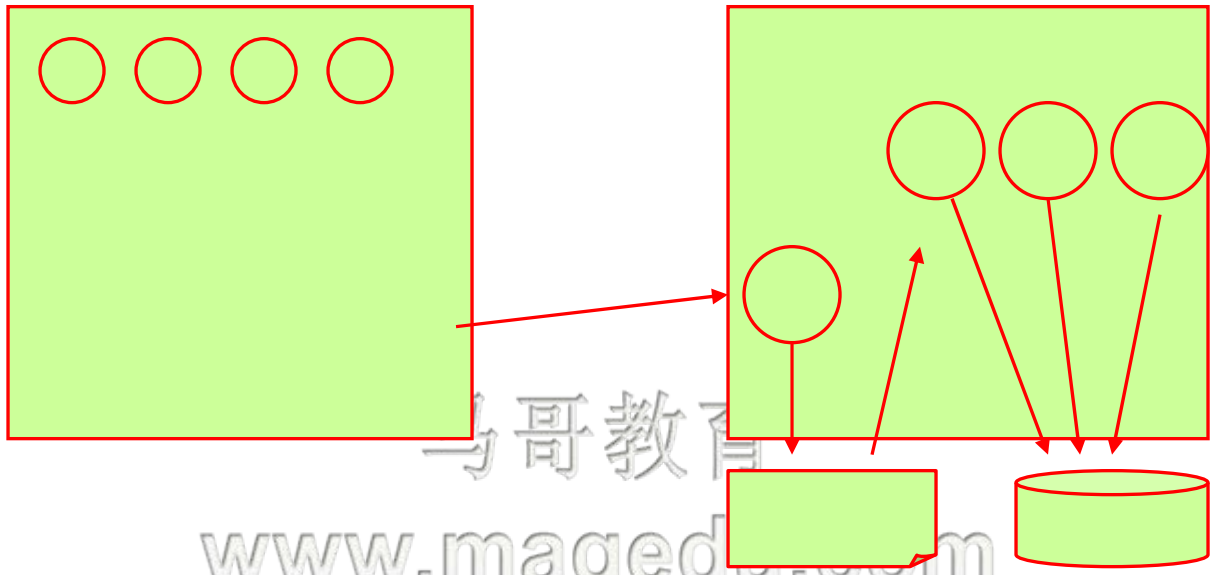
Multi-Threaded Slaves

- The multi-threaded slave splits processing between worker threads based on schema, allowing updates to be applied in parallel, rather than sequentially
- The number of worker threads is configured using the **slave-parallel-workers** parameter



马哥教育
www.magedu.com

- 每个数据库仅能使用一个线程
  - 复制涉及到多个数据库时多线程复制才有意义



马哥教育
www.magedu.com

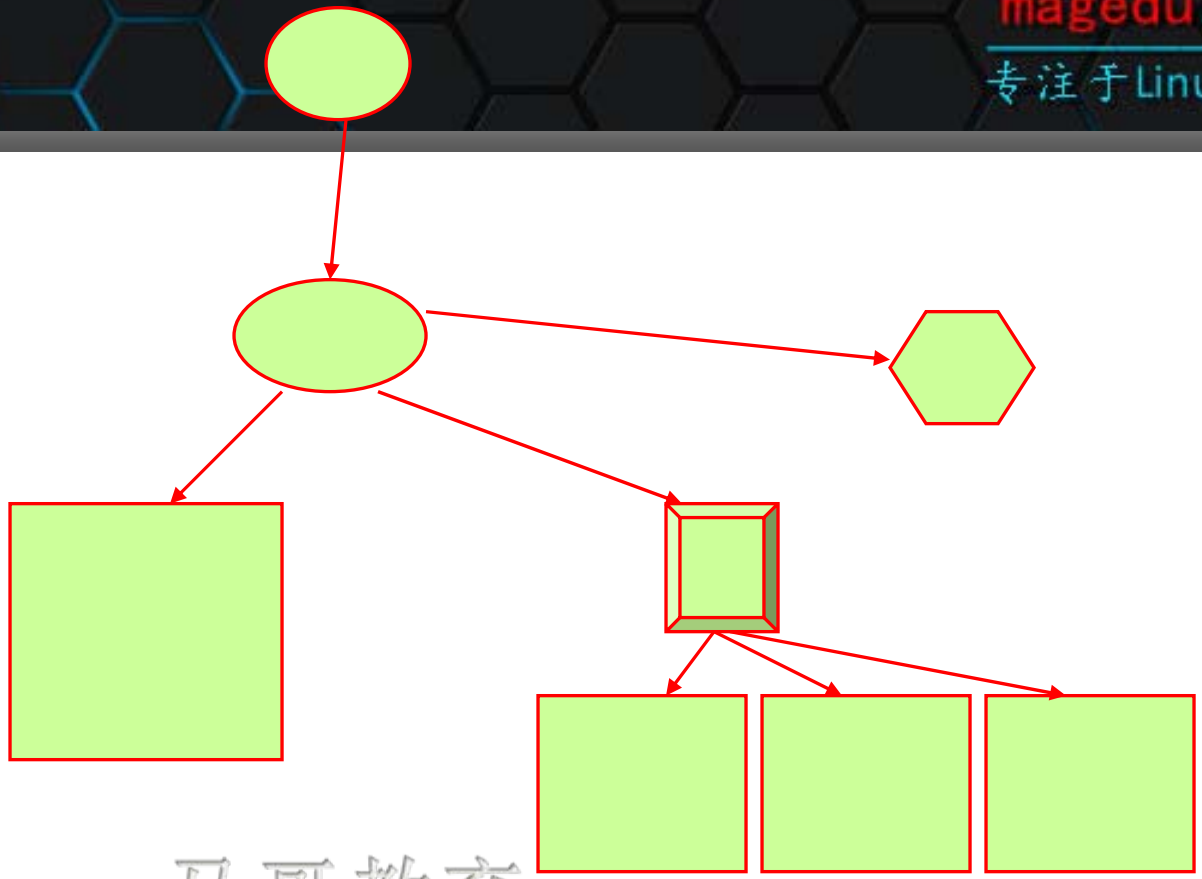
Crash-Safe Replication

- The Binary Log positioning information and the table data can now be written as part of the same transaction – ensuring that they are transactionally consistent when using InnoDB
- This enables the slave to automatically roll back replication to the last committed event before a crash, and resume replication without administrator intervention
- If a crash to the master causes corruption of the binary log, the server will automatically recover it to a position where it can be read correctly
- This functionality is enabled by setting the **master-info-repository** and **relay-log-inforepository** parameters to TRUE

马哥教育
www.magedu.com

马哥教育

mysql-proxy



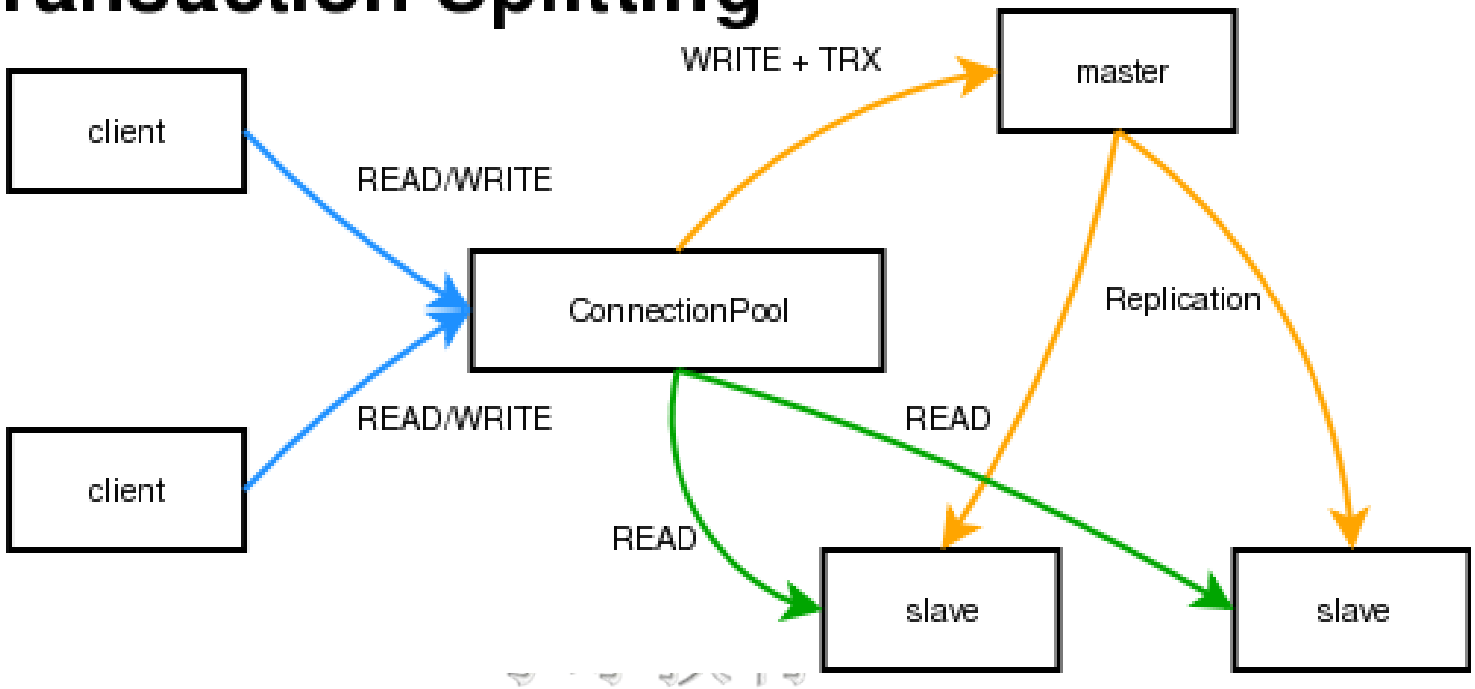
马哥教育
www.magedu.com

读、写分离

- MySQL Proxy (lua)
  - 连接路由、Query 分析、查询过滤和修改、负载均衡、HA
- Amoeba (Java)
  - 查询路由、查询分析、查询过滤、读写分离、负载均衡、HA
  - xml
- Cobar (Java)

马哥教育
www.magedu.com

Transaction Splitting



www.magedu.com

马哥教育

MySQL Other Topics

主讲：马永亮(马哥)
QQ：1661815153
http://www.magedu.com

Monitoring Master

- SHOW MASTER STATUS;
- SHOW MASTER LOGS;
- SHOW BINLOG EVENTS;
  - SHOW BINLOG EVENTS [IN <log>] [FROM <pos>] [LIMIT [<offset>,] <rows>]

    mysql> SHOW BINLOG EVENTS IN 'master-bin.000001' FROM 2571 LIMIT 4 \G
- SHOW BINARY LOGS

马哥教育
www.magedu.com

Master Status Variables

- Com_change_master
  - the number of times the CHANGE MASTER command was issued
- Com_show_master_status
  - the number of times the SHOW MASTER STATUS command was issued

马哥教育
www.magedu.com

Monitoring Slaves

- SHOW SLAVE STATUS
  - Connect_Retry
  - Exec_Master_Log_Pos
    - the position of the last event executed from the master's binary log
  - Relay_Log_Space
    - The total size of all of the relay logfiles
  - Seconds_Behind_Master
    - The number of seconds between the time an event was executed and the time the event was written in the master's binary log
    - pt-heartbeat命令 (percona-tools)

马哥教育
www.magedu.com

Slave Status Variables

- Com_show_slave_hosts
- Com_show_slave_status
- Com_slave_start
- Com_slave_stop
- Slave_heartbeat_period
- Slave_open_temp_tables
- Slave_received_heartbeats
- Slave_retried_transactions
- Slave_running

马哥教育
www.magedu.com

Causes for Slave Lag

- the main reason for lag is the single-threaded nature of the slave
  - mysql 5.6 multithread replication
    - one thread per database
- long-running queries with inefficient joins
- I/O-bound reads from disk, lock contention
- InnoDB thread concurrency issues

马哥教育
www.magedu.com

Replication Breaking - Causes

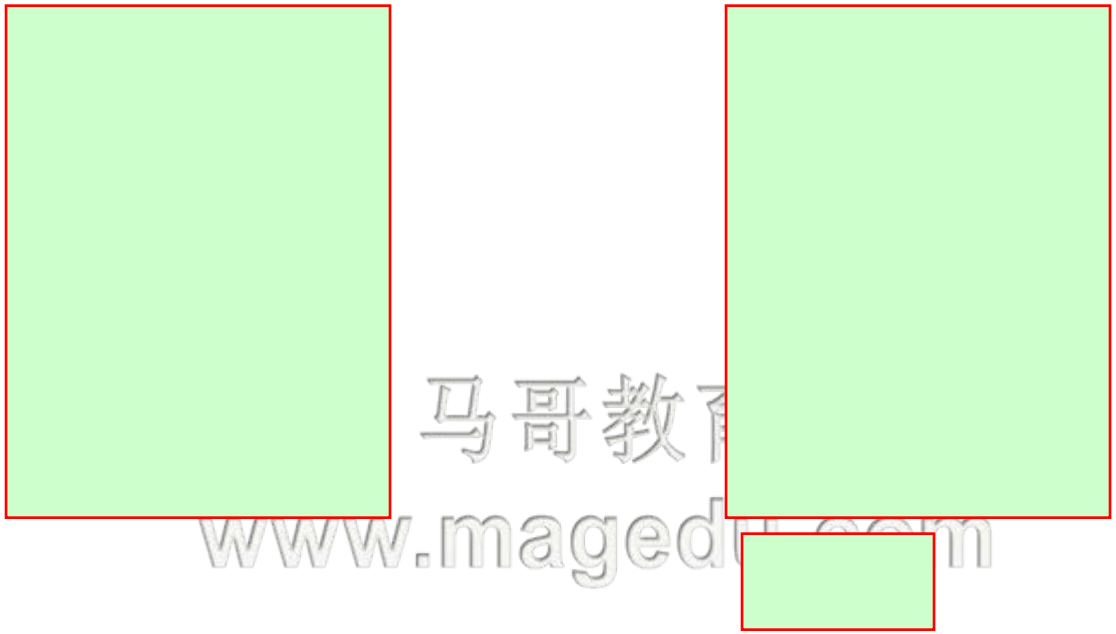
- Writes happening on Slaves
- SBR with bad statements
- Crashes: Relay Log Corruption
- Different schemas
- Temporary tables
- Mixing MyISAM with InnoDB
- Different behavior between MySQL versions (minor/major)
- Killing queries that change MyISAM tables will be partially executed, and break replication
- ...

Replication Breaking - Fixing

- Questions to ask yourself: Why? Where? Impact?
- Quick 'fix':

```
SET GLOBAL SQL_SLAVE_SKIP_COUNTER=1; START SLAVE;
```

  - Causes 1 event to be skipped
  - Causes inconsistencies!!!
- Look at:
  - Check error & Investigate data on master/slave
  - MySQL errorlogs
  - `SHOW SLAVE STATUS\G`
  - `mysqlbinlog` on relaylogs/binarylogs and investigate
  - `pt-slave-restart`
- More On: http://www.percona.com/files/presentations/percona-live/london-2011/PLUK2011-diagnosing-and-fixing-mysql-replication.pdf



马哥教育
www.magedu.com

- percona-tools
  - pt-table-checksum
    - 检测主从节点表是否一致；
  - pt-table-sync
    - 主从节点上表的重新同步；
  - pt-slave-restart
    - 从节点意外关闭时，用于安全启动复制进程

马哥教育
www.magedu.com

Crash Safe Replication

- On master: `sync_binlog=1` to fsync at every binlog write
  - Performance impact, mainly on:
    - Ext3 filesystem
      http://www.mysqlperformanceblog.com/2009/01/21/beware-ext3-and-sync-binlog-do-not-play-well-together/
    - High write-response time disksubsystem (no WriteBackCache& BBU)
- On slave:
  - `relay-log.info/master.info`: not durable/consistent!
  - When OS crash, old information might be in the file
  - If InnoDB only: use `innodb_overwrite_relay_log_info` in Percona Server:
    http://www.percona.com/doc/percona-server/5.5/reliability/innodb_recovery_update_relay_log.html

www.magedu.com

Consistency

- Replication can cause inconsistencies:
  - Statement Based Replication
    - Certain functions like `UUID()`...
    - `UPDATE/DELETE` with `LIMIT` without `ORDER BY` unique key
  - Writing to multiple masters
  - Writing to slaves (by accident? use `read_only`)
  - Failover done wrong
  - MyISAM with killing queries and/ord MySQL Crashes
  - Wrongly restored slaves
  - Replication broke and used `SQL_SLAVE_SKIP_COUNTER`
  - ...
- How to know if my data is consistent?

Consistency

- Checking Consistency: `pt-table-checksum`
  - Checksums chunks of data on master and slaves
  - Monitors replication lag
  - Low checksum response time to lower impact on production
  - Uses Replication to ensure consistent view of data
- Fixing Consistency: `pt-table-sync`
  - Can checksum and use `pt-table-checksum` results
  - Fixes inconsistencies

马哥教育
www.magedu.com

High Availability

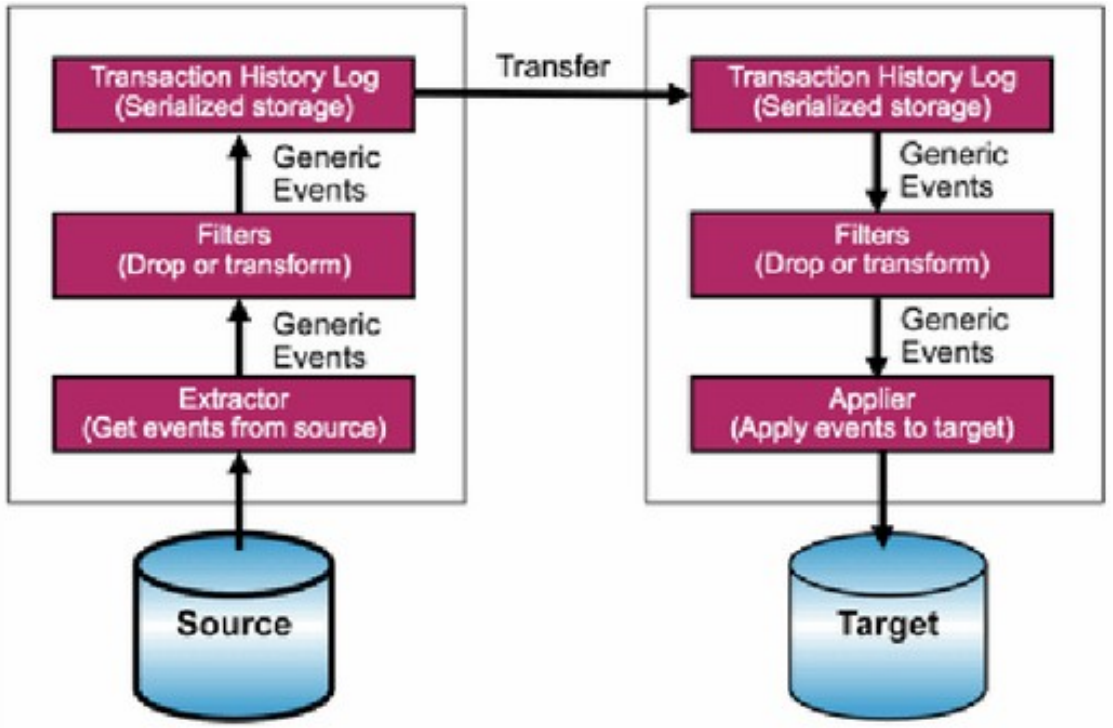
- Failover:
  - Virtual IPs
  - Change IP in application servers
- Warning: Split-Brain possible!!!
  - Ensure no writes happen on old master before moving
  - Make sure new master is in sync
- Scripts available to automate failover:
  - MySQL-MHA http://code.google.com/p/mysql-master-ha/
  - MMM (not-recommended) http://mysql-mmm.org/
  - Percona-PRM

    http://www.mysqlperformanceblog.com/2011/11/29/percona-replication-manager-a-solution-for-mysql-high-availability-with-replication-using-pacemaker/

Other Replication Implementations

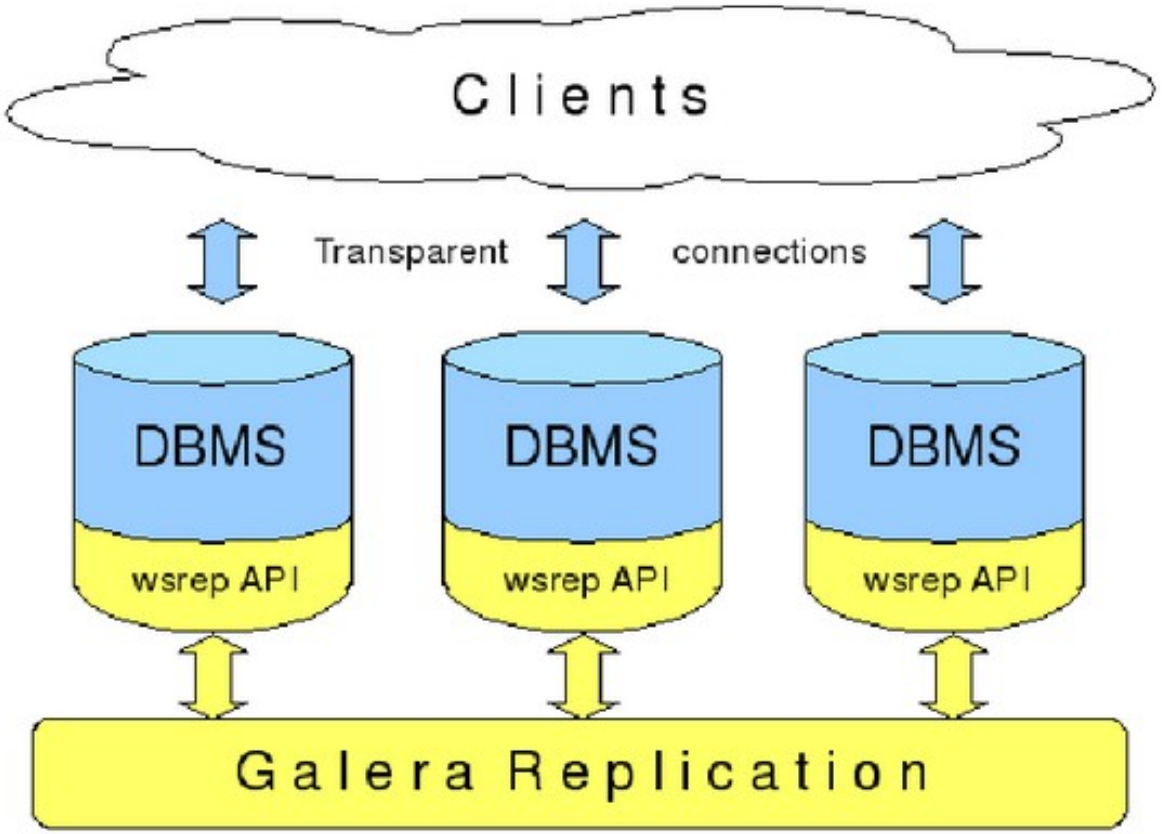
- Tungsten Replicator
  - Uses binary log
  - Parallel replication
  - Enhanced filtering
  - Multi-master
  - Replication between different databases
- Galera Cluster
  - Does not use binary log
  - Synchronous replication
  - Active-active master
  - Parallel replication
  - Percona XtraDB Cluster! http://www.mysqlperformanceblog.com/2012/01/09/announcement-of-percona-xtradb-cluster-alpha-release/

Tungsten Replicator



http://continuent.com/
http://code.google.com/p/tungsten-replicator/

Galera Cluster, Percona



http://codership.com/

Delayed slave

- Deliberately lag a slave for a predefined amount of time
- Decrease recovery time in case of bad changes caused by:
  - Humans (`DROP DATABASE`)
  - Application (`DELETE FROM table;`)
- How to configure:
  - `pt-slave-delay --delay 60m`
    (http://www.percona.com/doc/percona-toolkit/2.0/pt-slave-delay.html)
  - 5.6 Built-in feature:

```
> CHANGE MASTER TO MASTER_DELAY=3600;
```

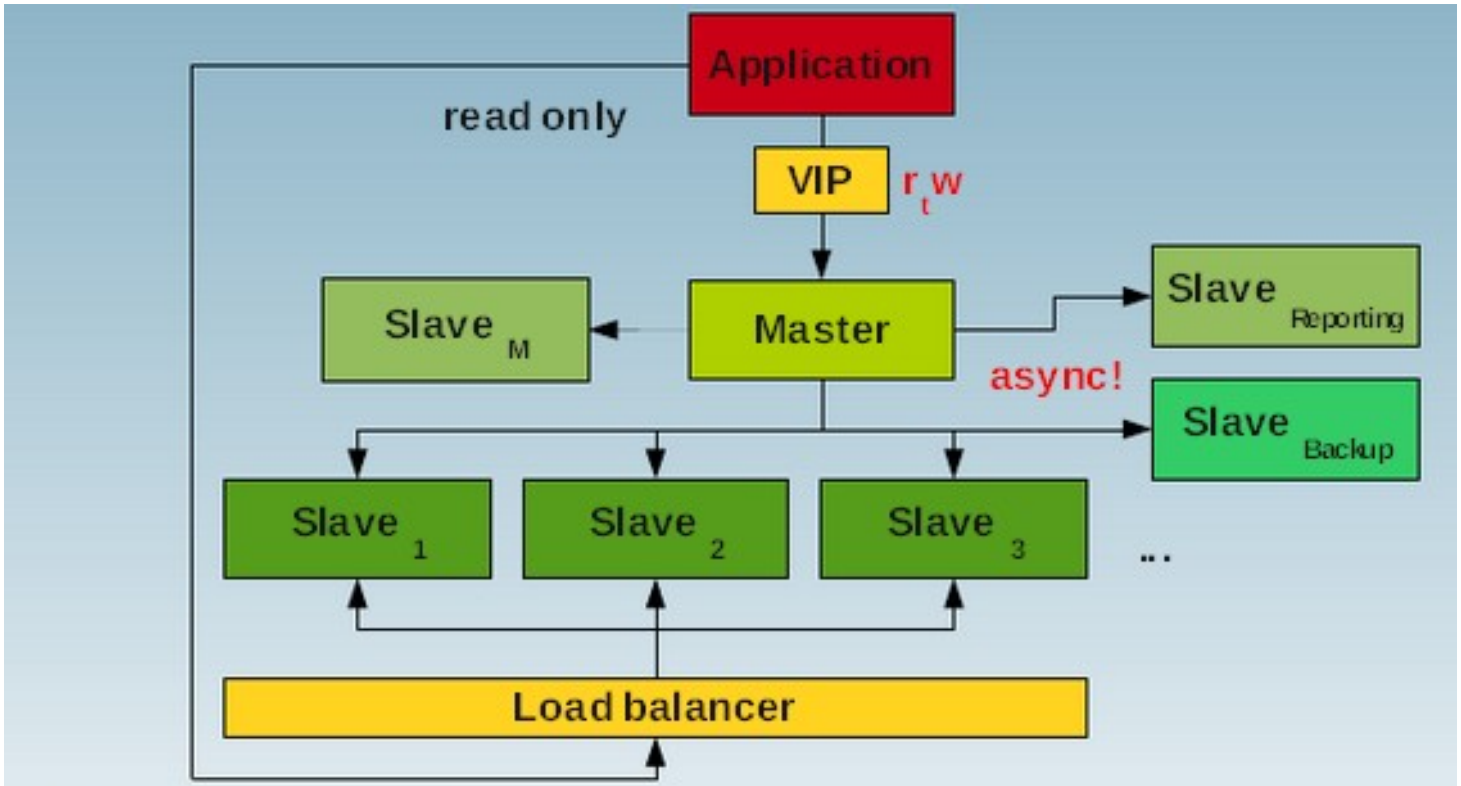
- When a problem happens stop replication just before the malicious statement.

```
START SLAVE UNTIL MASTER_LOG_FILE='log_name',
MASTER_LOG_POS = pos;
```

Tools & MoreInfo

- Percona Toolkit http://www.percona.com/software/percona-toolkit/
  `pt-slave-find, pt-table-checksum, pt-table-sync, pt-slave-delay, pt-slave-restart, pt-heartbeat`
- Percona Server http://www.percona.com/software/percona-server/
- OpenArk kit http://code.openark.org/forge/openark-kit
- Replication Booster https://github.com/yoshinorim/replication-booster-for-mysql
- High Availability
  - MMM http://mysql-mmm.org/
  - Percona-PRM
    http://www.mysqlperformanceblog.com/2011/11/29/percona-replication-manager-a-solution-for-mysql-high-availability-with-replication-using-pacemaker/
  - MySQL-MHA http://code.google.com/p/mysql-master-ha/
- High Performance MySQL, 2nd Edition:
  http://shop.oreilly.com/product/9780596101718.do

High-Availability with Replication



- Fail-over?

Advantages/Disadvantages

- Simple „standard“ Set-up
- Master is a SpoF! (Single Point of Failure)
- If master fails → which Slave becomes new master?
  - Switch → a lot of work, delicate!
  - There are tools to help (MMM ~~v1~~/v2, MHA, Tungsten, ...)
- Fail-over Site is already warm/hot!
- Works very well if r >> w
- Data inconsistencies (mk-table-check/sync)
- Delay Master/Slave
- Slave lagging (Slave as bottleneck)

- 5.5
  - semi-synchronous replication
- 5.6
  - delayed replication
  - server UUID
  - crash-safe slave
  - multi-thread slave
  - Global transaction identifers

马哥教育
www.magedu.com

马哥教育

案例：Data Storage at Craigslist

主讲：马永亮(马哥)
QQ：1661815153
http://www.magedu.com

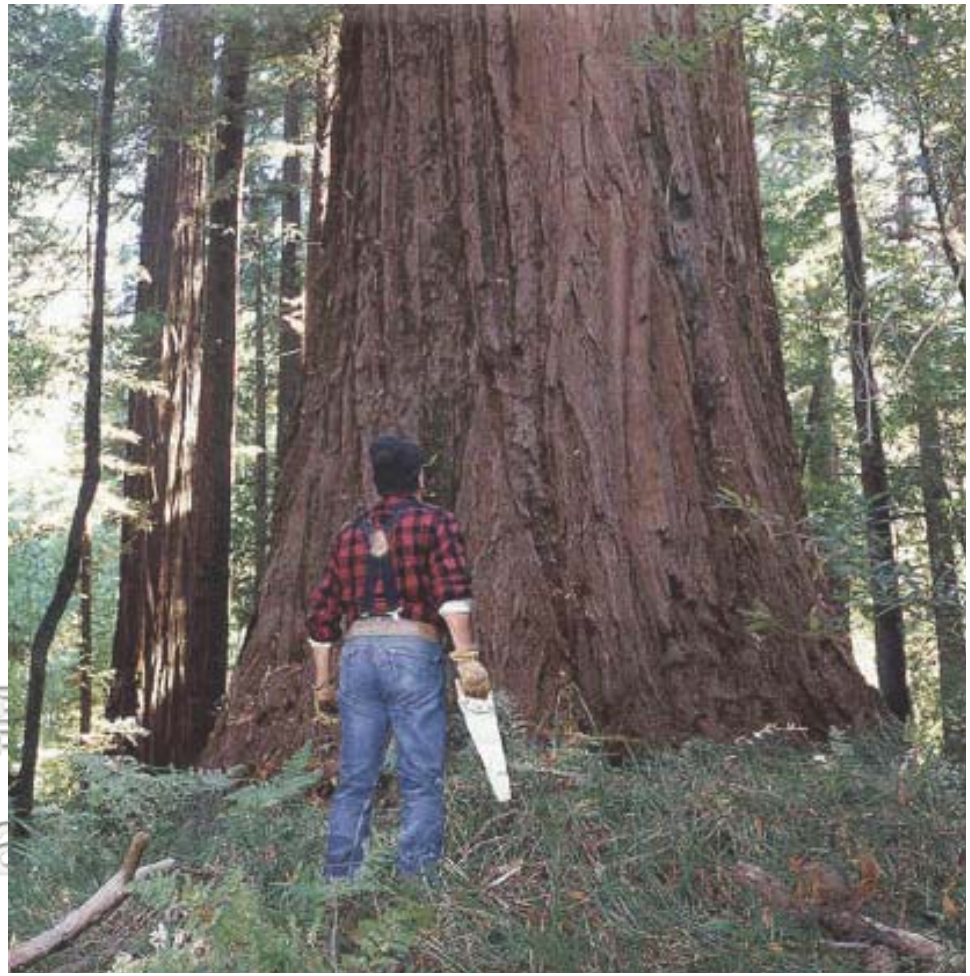
Data Storage at Craigslist

- MySQL
- Memcached
- Redis
- MongoDB
- Sphinx
- Filesystem

马哥
www.mag

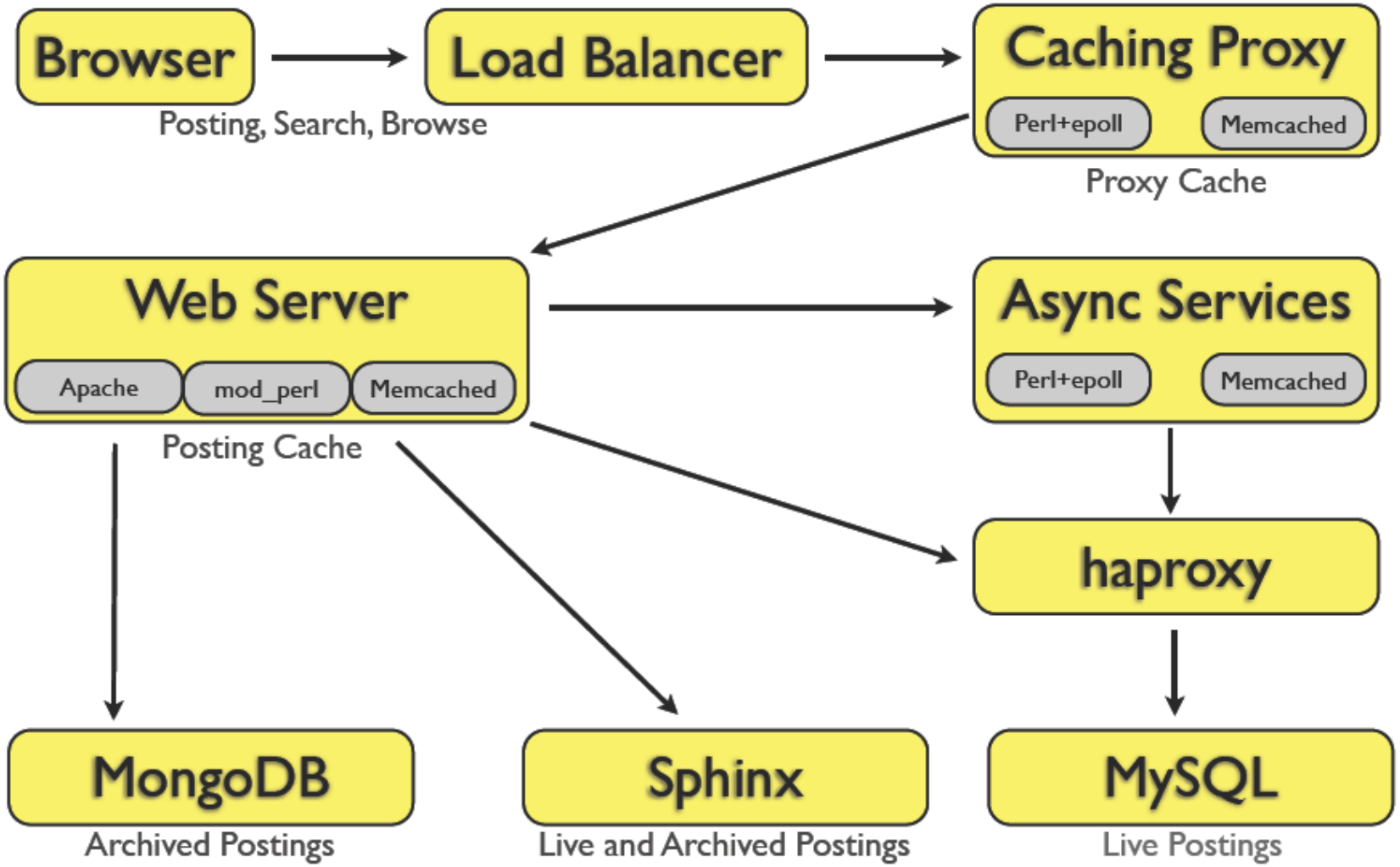
Choosing the Right Tool

- Durability
- Performance
- Query API
- Features
- Complexity
- Support

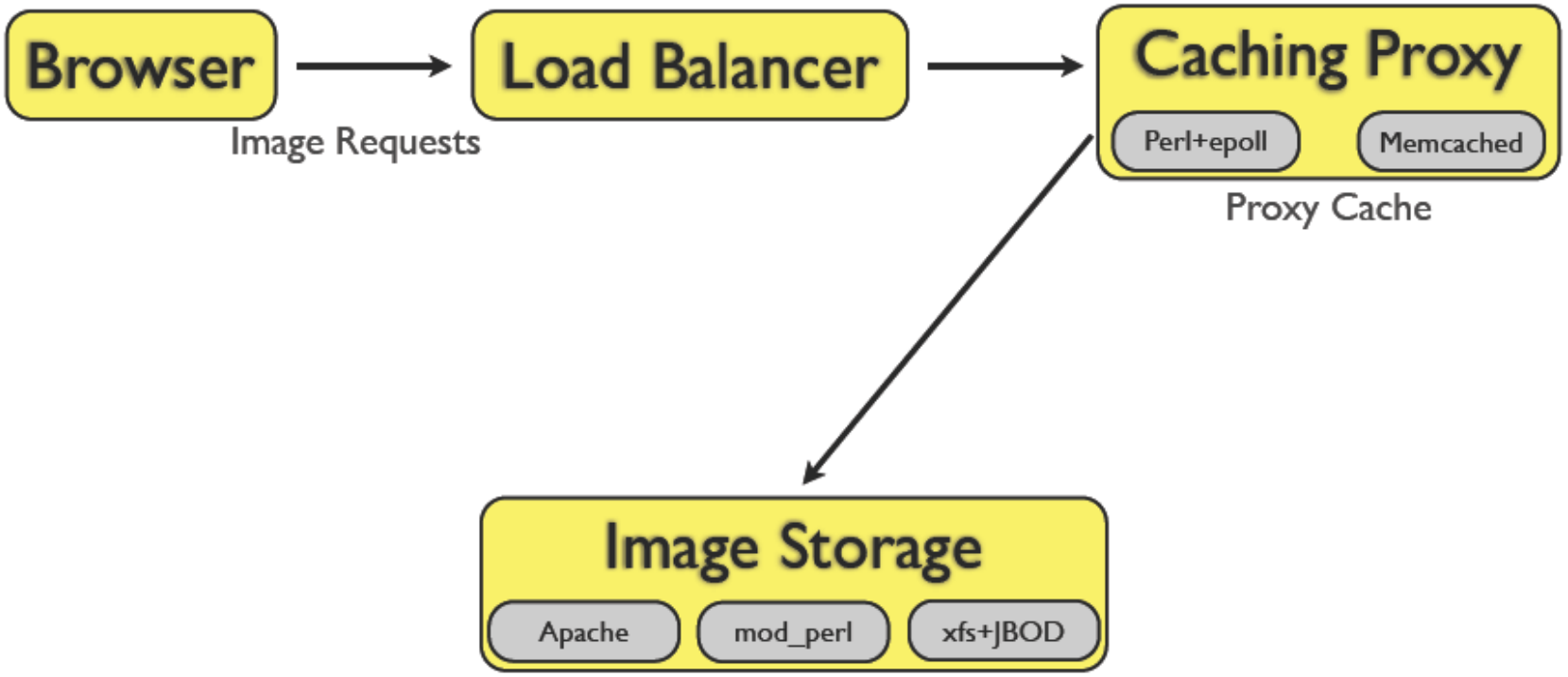


马哥
www.ma

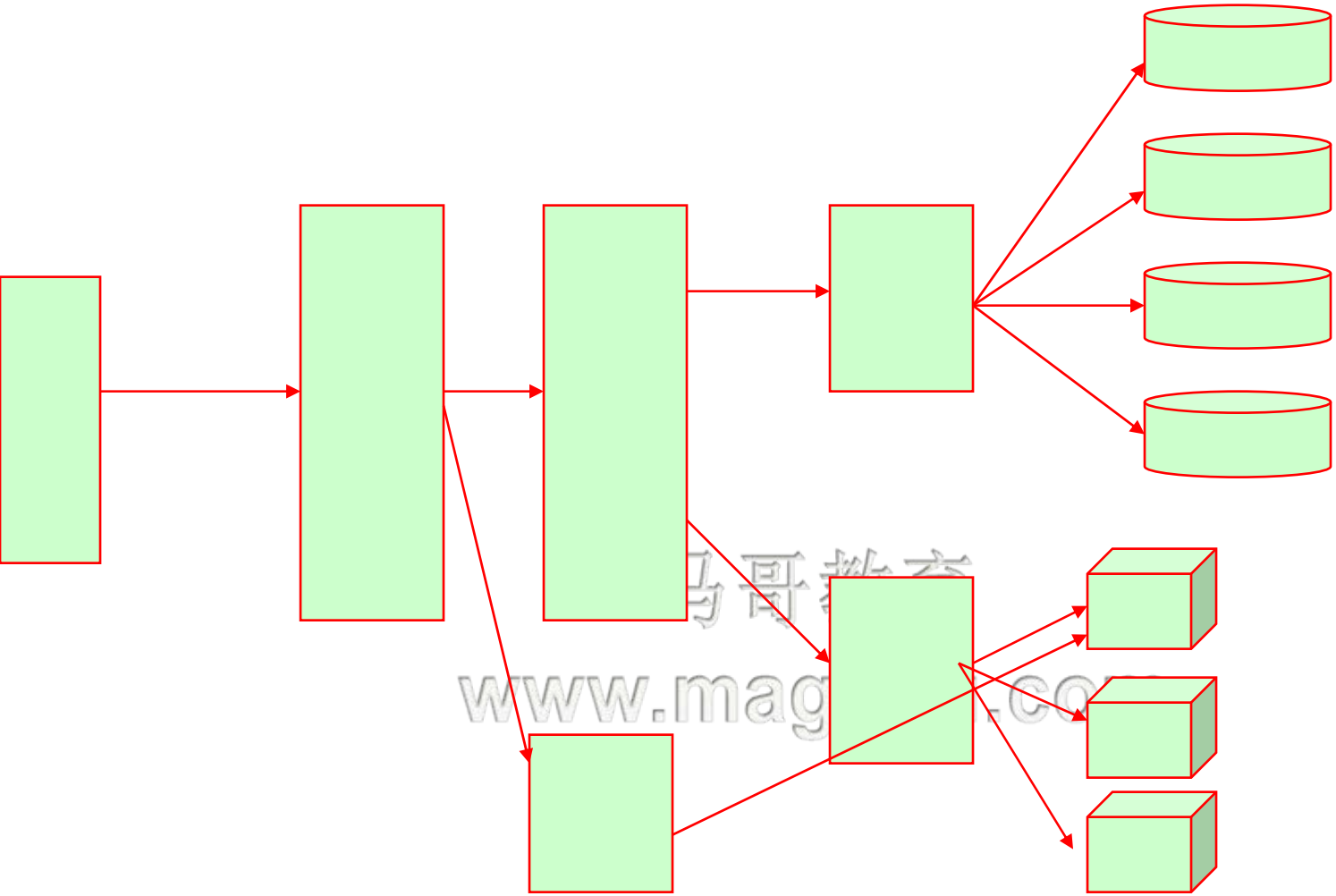
Request Flow (reads)



Request Flow (reads)

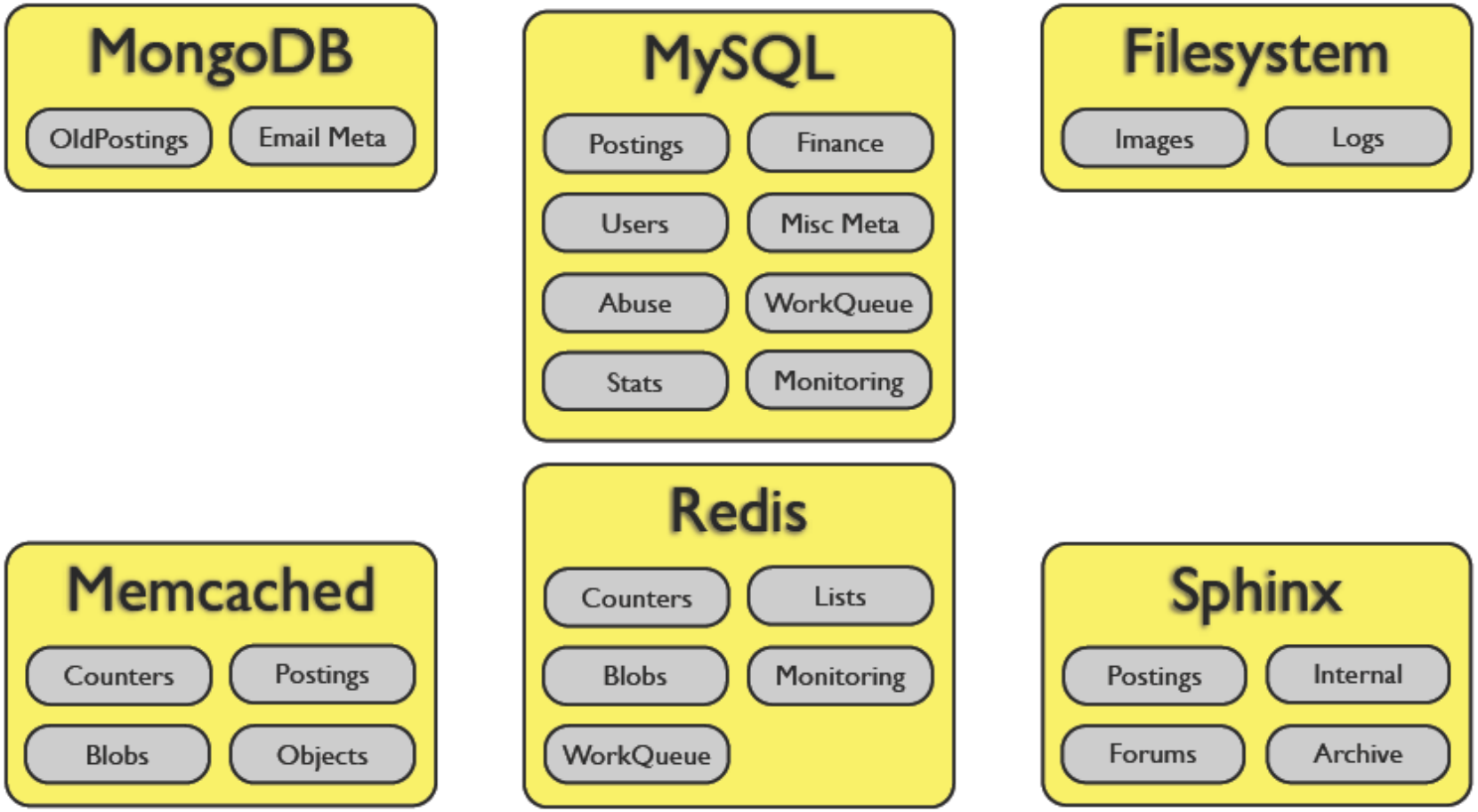


www.magedu.com



马哥教育
www.mag

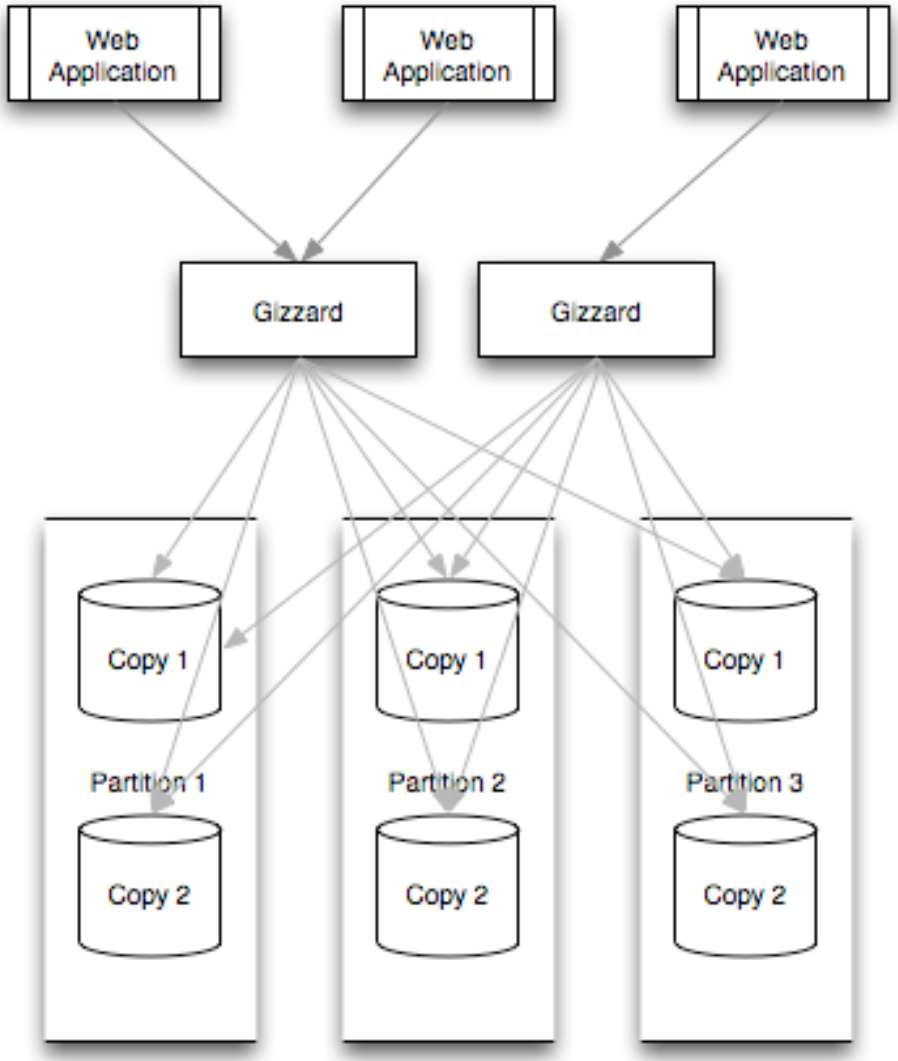
Data Repositories



马哥教育

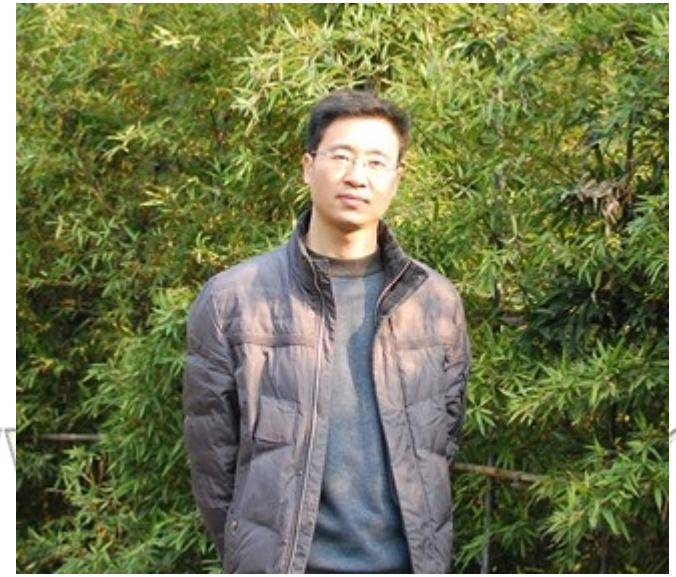
MySQL sharding framework

Gizzard


Web Application
Web Application
Web Application
Gizzard
Gizzard
Copy 1
Copy 1
Copy 1
Partition 1
Partition 2
Partition 3
Copy 2
Copy 2
Copy 2

关于马哥教育

- 博客：http://mageedu.blog.51cto.com
- 主页：http://www.magedu.com
- QQ：1661815153，113228115
- QQ群：203585050，279599283



马哥教育

Thank You!