

MYSQL. РЕПЛИКАЦИЯ.

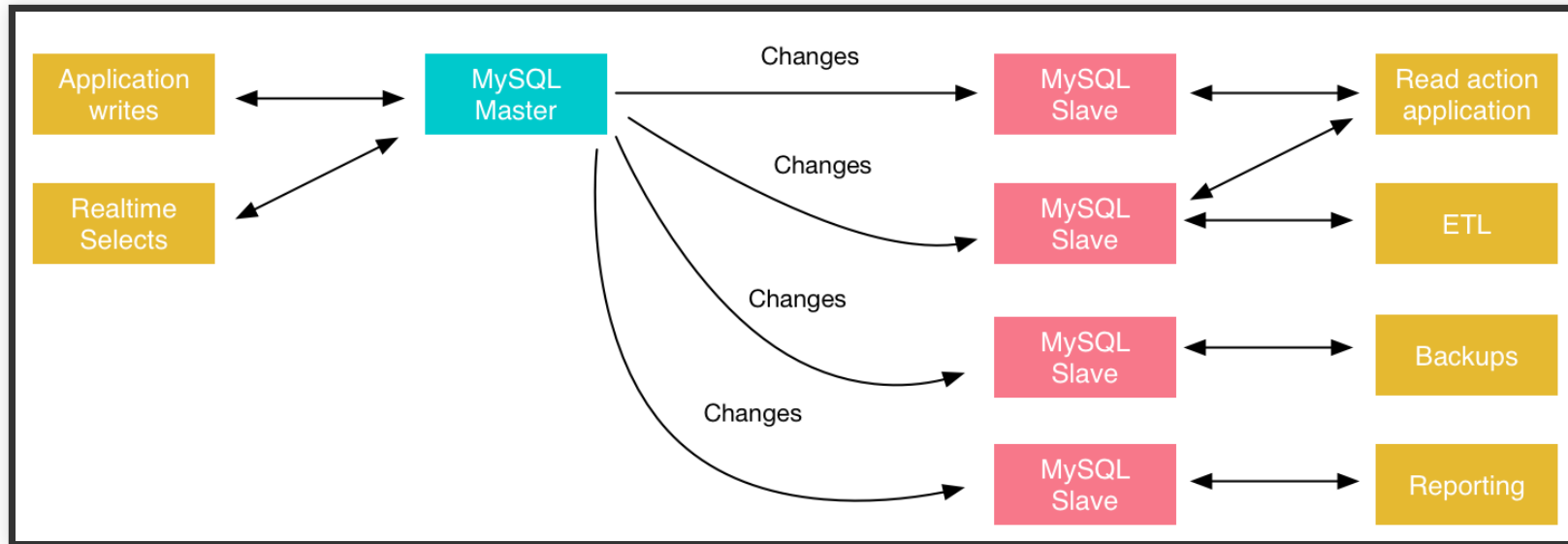
СЕГОДНЯ МЫ РАЗБЕРЕМ...

- Что такое репликация
- Зачем нужна репликация
- Виды репликации
- организация репликации

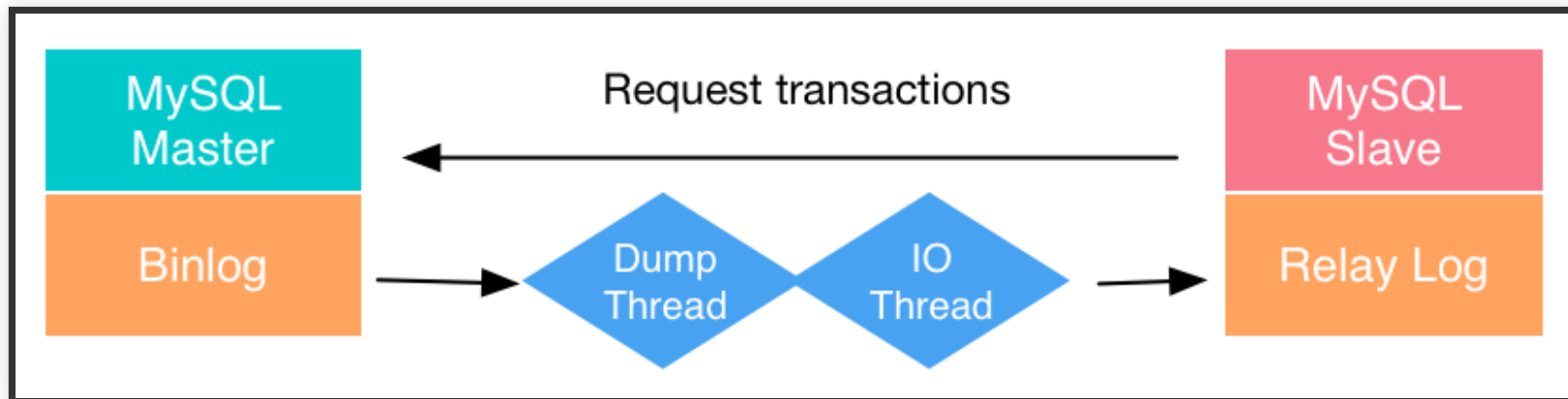
ПРЕИМУЩЕСТВА РЕПЛИКАЦИИ

- масштабирование нагрузки
 - нагрузку по чтению можно переложить на реплики
 - отчетные запросы также можно вынести на реплику
- бэкапы можно снимать со слейвов
- возможность переключения в случай сбоя мастера

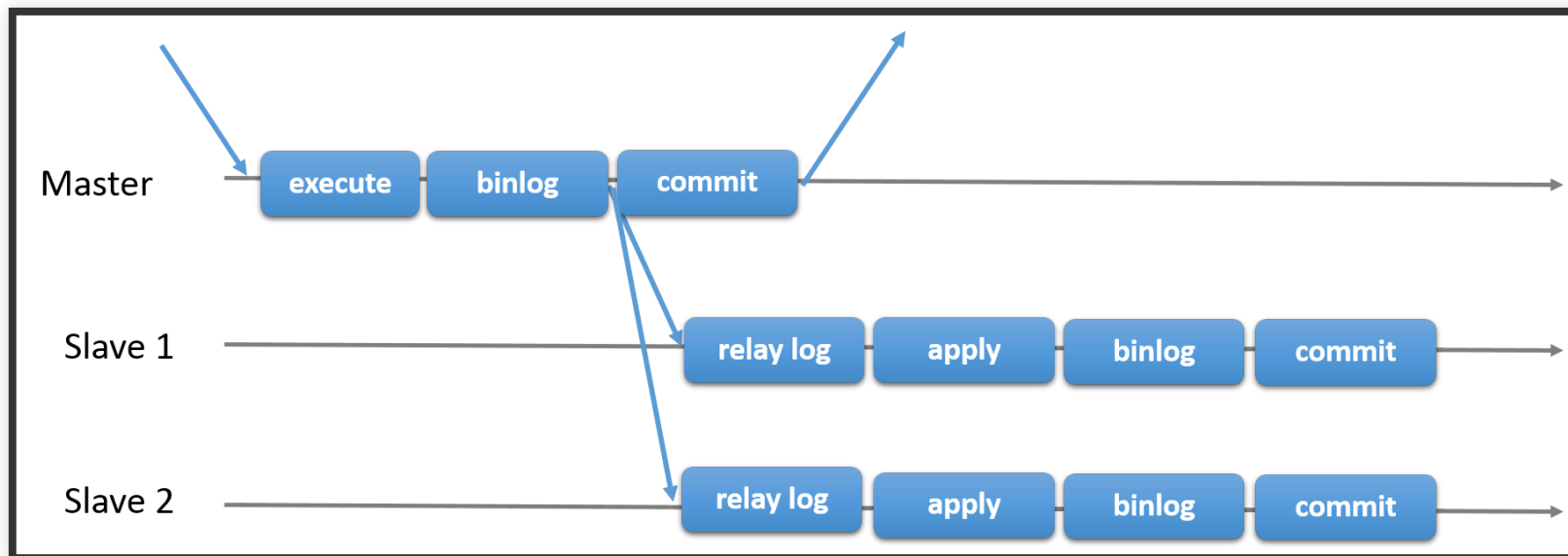
MASTER-SLAVE РЕПЛИКАЦИЯ



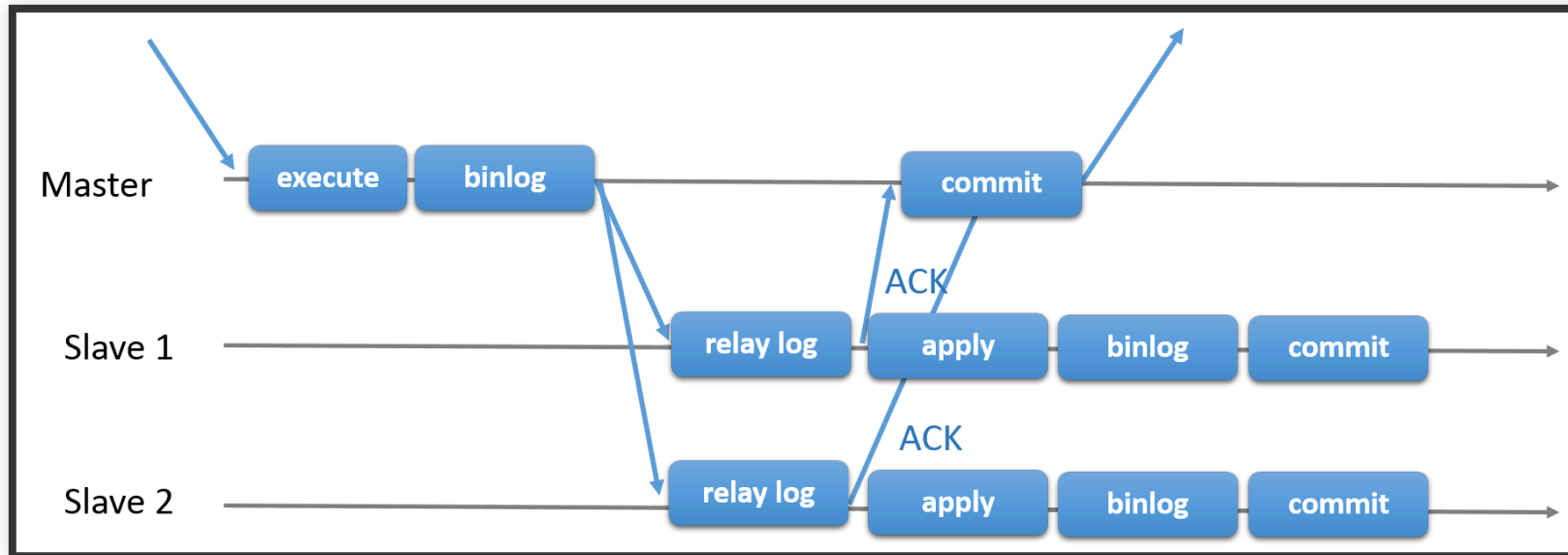
ПРОЦЕССЫ MASTER-SLAVE РЕПЛИКАЦИИ



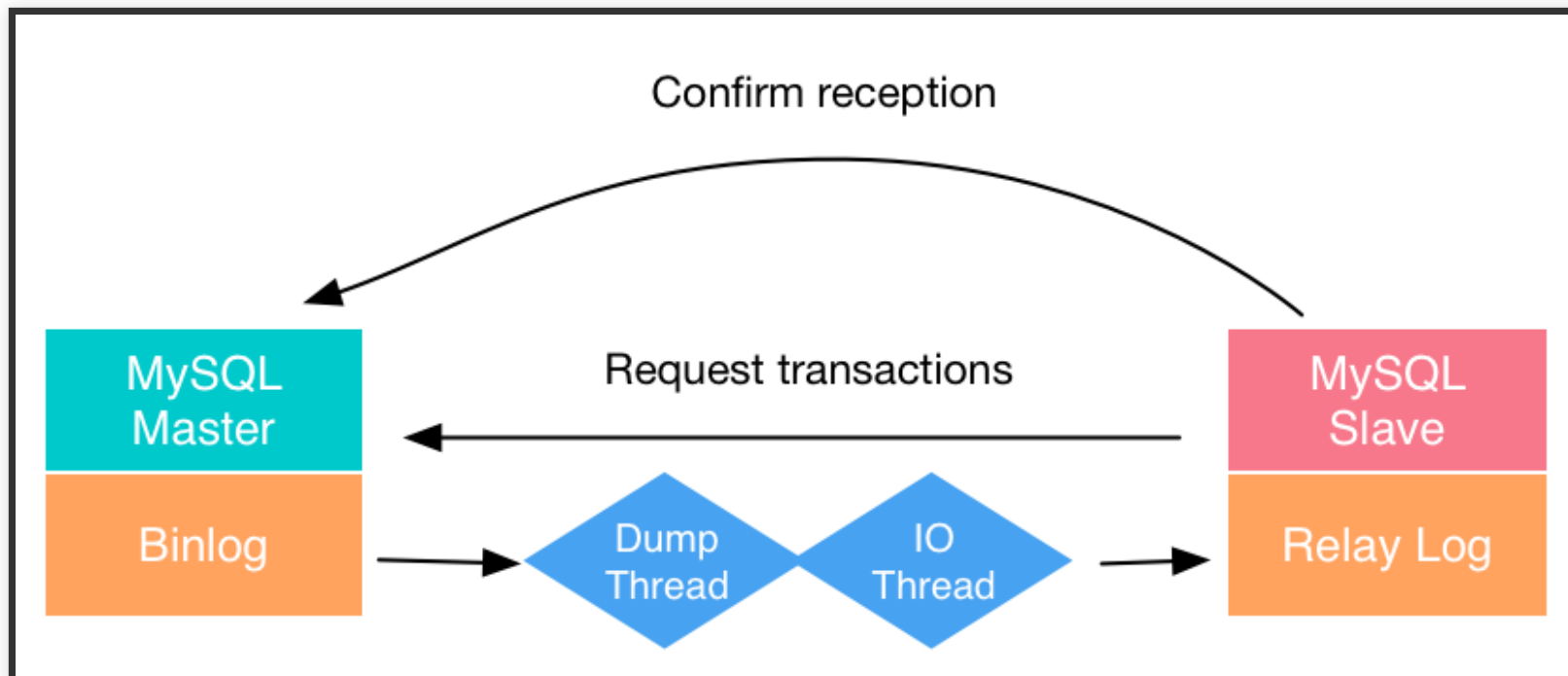
АСИНХРОННАЯ РЕПЛИКАЦИЯ



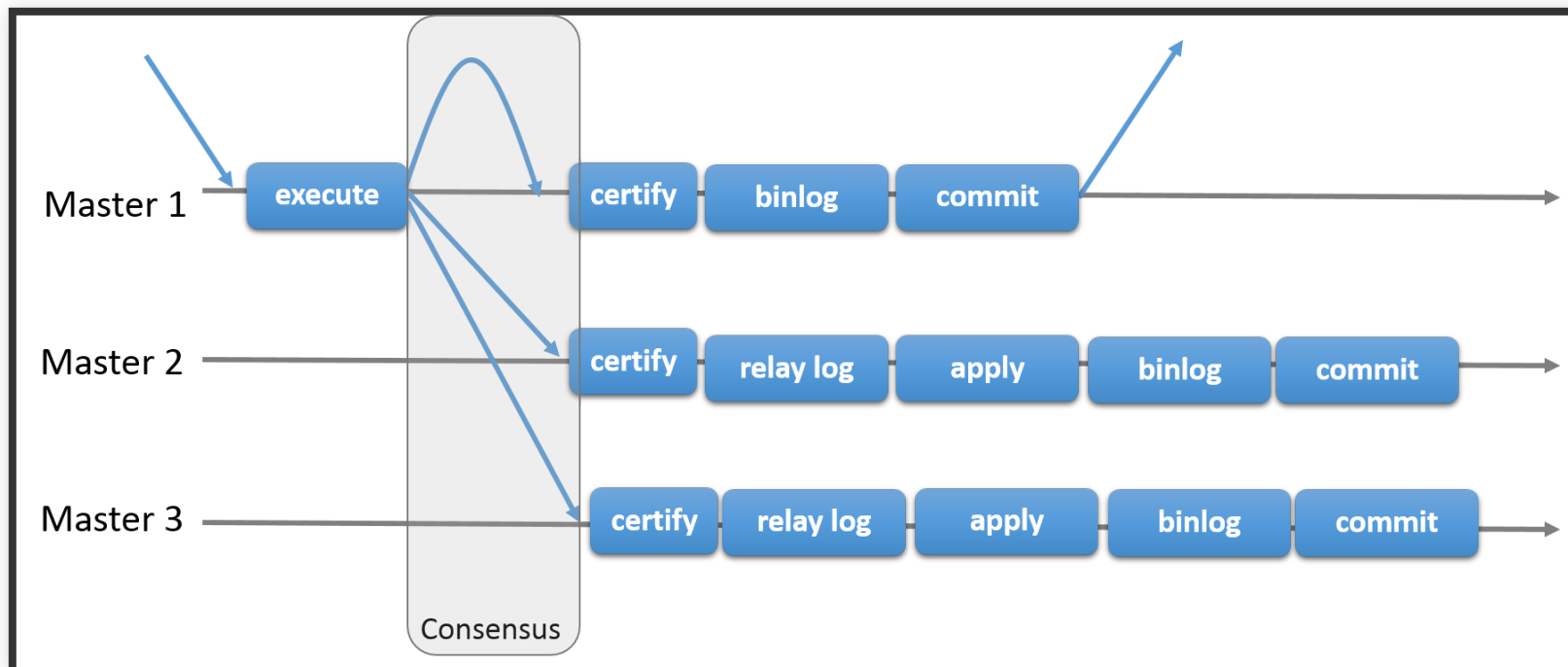
ПОЛУСИНХРОННАЯ РЕПЛИКАЦИЯ



ПОЛУСИНХРОННАЯ РЕПЛИКАЦИЯ



ГРУППОВАЯ РЕПЛИКАЦИЯ



ФОРМАТ БИНАРНОГО ЛОГА

- **документация**
- statement-based
 - в лог пишутся SQL
- row-based
 - в лог пишутся строки
 - получаются большие логи
- mixed
 - для safe стейтментов пишутся SQL
 - для unsafe - строки

ОПЦИИ ДЛЯ УПРАВЛЕНИЯ БИНАРНЫМИ ЛОГАМИ

```
# размер кэша в памяти
binlog_cache_size

# формат
binlog_format = ROW/STATEMENT/MIXED

# Шифрование
binlog_encryption

# Реакция на ошибку записи
binlog_error_action = ABORT_SERVER/IGNORE_ERROR

# таймоут хранения в секундах
binlog_expire_logs_seconds
```

BINARY LOG FILE POSITION BASED REPLICATION

- включаем binary logs на мастере
- снимаем копию с мастера
- проверяем позицию бинлога
- создаем пользователя для репликации
- поднимаем копию на слейве
- меняем **server_id** на слейве
- запускаем команду CHNAGE MASTER

ON MASTER

```
create user repl@10.51.21.42 IDENTIFIED BY 'oTUSlave#2020';  
GRANT REPLICATION SLAVE ON *.* TO repl@10.51.21.42;  
  
SHOW MASTER STATUS;
```

ON SLAVE

```
CHANGE MASTER TO MASTER_HOST='10.51.21.40',  
MASTER_USER='repl', MASTER_PASSWORD='oTUSlave#2020',  
MASTER_LOG_FILE='binlog.000027',  
MASTER_LOG_POS=707;  
  
START SLAVE;
```

КОМАНДЫ ДЛЯ УПРАВЛЕНИЯ СОСТОЯНИЕМ

```
STOP SLAVE;  
START SLAVE;  
RESET SLAVE;  
SHOW SLAVE STATUS;  
  
RESET MASTER;  
SHOW MASTER STATUS;  
SHOW BINLOG STATUS;
```

GTID

- Global transaction identifier
- Имеет формат : GTID = source_id:transaction_id
 - 3E11FA47-71CA-11E1-9E33-C80AA9429562:2
- Позволяет убедиться, что транзакция принадлежит только одному серверу.
- Позволяет убедиться, что транзакция применена только один раз в системе

GTID LIFECYCLE

- транзакция выполняется и коммитится на мастере
 - назначается GTID, записывается в binlog
- slave считывает GTID с релэй лога и прописывает его в `gtid_next`
- slave производит проверку что такой `gtid` не исполнялся
- для записи в бинлог slave использует `gtid_next`
- после коммита транзакция добавляется в `gtid-executed`

ВКЛЮЧАЕМ GTID

```
# включает режим GTID
gtid-mode = on

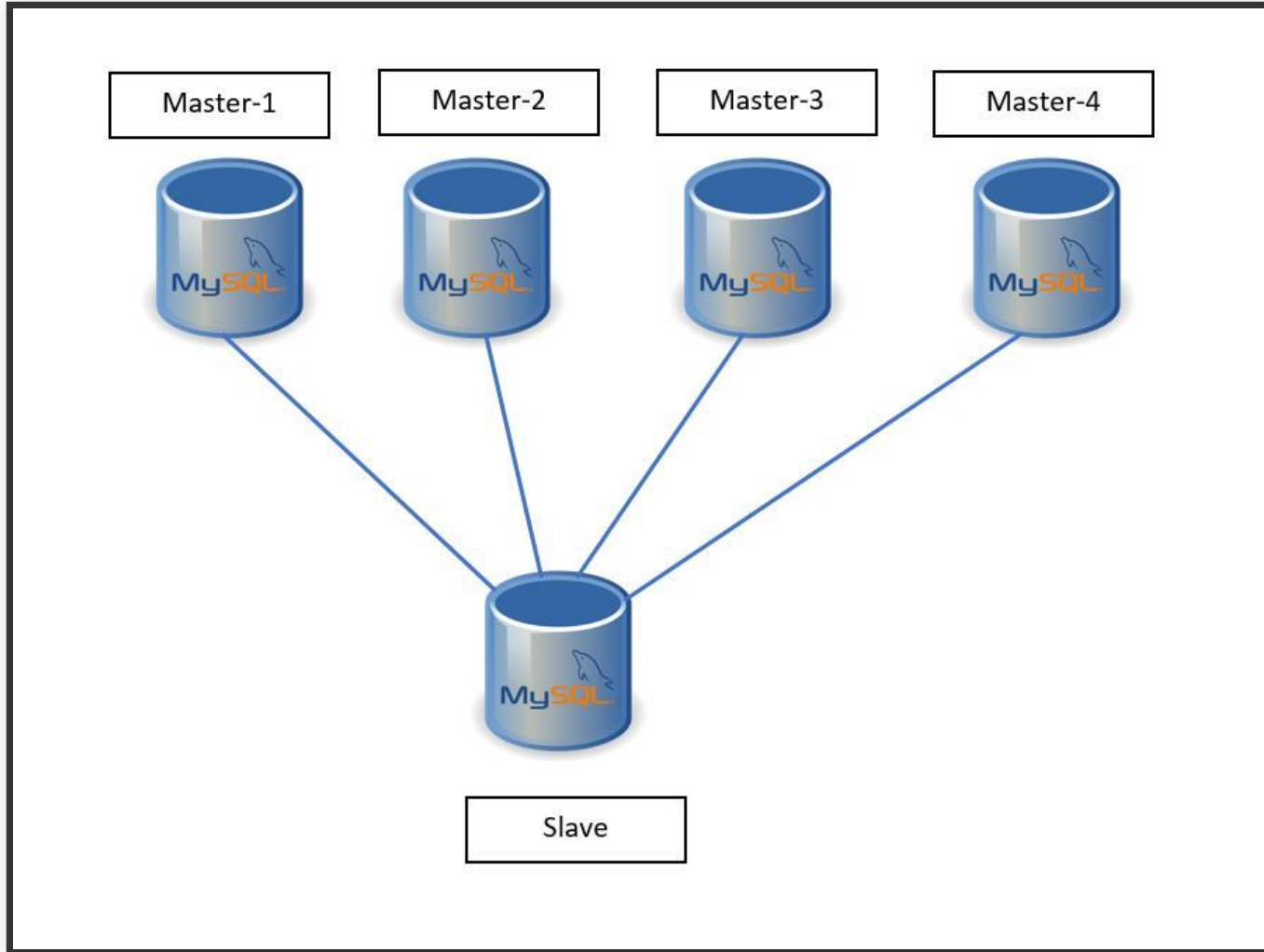
# разрешает выполнение только безопасных транзакций
enforce-gtid-consistency=true
```

- на слейве

```
CHANGE MASTER TO MASTER_HOST='10.51.21.40',
MASTER_USER='repl', MASTER_PASSWORD='oTUSlave#20
MASTER_AUTO_POSITION=1;
START SLAVE;
```

МОНИТОРИМ GTID

```
SHOW @@gtid_executed;  
SHOW @@gtid_purged;
```



МОНИТОРИНГ РЕПЛИКАЦИИ

таблицы в Performance Schema

```
replication_applier_configuration  
replication_applier_filters  
replication_applier_global_filters  
replication_applier_status  
replication_applier_status_by_coordinator  
replication_applier_status_by_worker  
replication_connection_configuration  
replication_connection_status  
replication_group_member_stats  
replication_group_members
```

ПОЛУСИНХРОННАЯ РЕПЛИКАЦИЯ

```
# на мастере и слейвах
mysql> INSTALL PLUGIN rpl_semi_sync_master SONAME 'semisync_master.so';

# включение режима репликации
rpl_semi_sync_master_enabled

# таймоут ожидания ответа в мс
rpl_semi_sync_master_timeout

# включение дебага
rpl_semi_sync_master_trace_level

# ожидание подтверждения от заданного кол-ва слейвов
rpl_semi_sync_master_wait_for_slave_count

# коммит на матере после синхронизации биологов на слейв, либо коммит и дожидает ответа от слейва
rpl_semi_sync_master_wait_point = AFTER_SYNC/AFTER_COMMIT
```

ФИЛЬТРЫ НА СЛЕЙВЕ

```
CHANGE REPLICATION FILTER filter[, filter]
    [, ...] [FOR CHANNEL channel]

filter:
    REPLICATE_DO_DB = (db_list)
    | REPLICATE_IGNORE_DB = (db_list)
    | REPLICATE_DO_TABLE = (tbl_list)
    | REPLICATE_IGNORE_TABLE = (tbl_list)
    | REPLICATE_WILD_DO_TABLE = (wild_tbl_list)
    | REPLICATE_WILD_IGNORE_TABLE = (wild_tbl_list)
    | REPLICATE_REWRITE_DB = (db_pair_list)
```

Отus: учиcь у лучших!