1. Настройка асинхронной репликации

Запускаю Docker-compose файл с двумя контейнерами Mysql:

version: '3'  
services:  
 mysql\_master:  
 image: mysql:5.7  
 env\_file:  
 - ./master/mysql\_master.env  
 container\_name: "mysql\_master"  
 restart: "no"  
 ports:  
 - 4406:3306  
 volumes:  
 - ./master/data:/var/lib/mysql  
 command: --server-id=1 --log-bin='mysql-bin-1.log' --binlog\_do\_db='mydb'  
 networks:  
 - overlay  
  
 mysql\_slave:  
 image: mysql:5.7  
 env\_file:  
 - ./slave/mysql\_slave.env  
 container\_name: "mysql\_slave"  
 restart: "no"  
 ports:  
 - 5506:3306  
 depends\_on:  
 - mysql\_master  
 volumes:  
 - ./slave/data:/var/lib/mysql  
 command: --server-id=2 --log-bin='mysql-bin-1.log' --relay-log='mysql-relay-bin.log' --binlog\_do\_db='mydb'  
 networks:  
 - overlay  
  
networks:  
 overlay:

Запускаю на мастере. Даю права для юзера слейва на мастере:

$ docker exec -it mysql\_master mysql -uroot -p111

mysql> GRANT REPLICATION SLAVE ON \*.\* TO "mydb\_slave\_user"@"%" IDENTIFIED BY "mydb\_slave\_pwd";  
mysql> FLUSH PRIVILEGES;

Запускаю слейв:

$ docker exec -it mysql\_slave mysql -uroot -p111

mysql> CHANGE MASTER TO MASTER\_HOST='192.168.32.1', MASTER\_USER='mydb\_slave\_user', MASTER\_PASSWORD='mydb\_slave\_pwd', MASTER\_LOG\_FILE='mysql-bin-1.000003', MASTER\_LOG\_POS=895;  
mysql> START SLAVE;

Делаю insert на мастере и проверяю изменения на слейве.

Настраиваю репликацию в приложении (первый хост – мастер)

spring:  
 datasource:  
 url: jdbc:mysql:replication://localhost:4406,localhost:5506/mydb   
 username: root  
 password: 111

Даю нагрузку на чтение до репликации и после:

wrk -t10 -c100 -d200s --latency -s scripts/otus.lua http://172.21.80.1

Результаты метрик (сначала нагрузка только на мастер, после репликации и переноса чтения на слейв нагрузка перешла слейв):



1. Настройка полусинхронной репликации

Запускаю Docker-compose файл с тремя контейнерами Mysql:

version: '3'  
services:  
 mysql\_master:  
 image: mysql:5.7  
 env\_file:  
 - ./master/mysql\_master.env  
 container\_name: "mysql\_master"  
 restart: "no"  
 ports:  
 - 4406:3306  
 volumes:  
 - ./master/data:/var/lib/mysql  
 command: --server-id=1 --log-bin='mysql-bin-1.log' --binlog\_format='ROW' --binlog\_do\_db='mydb' --gtid\_mode='ON' --enforce\_gtid\_consistency='ON'  
 networks:  
 - overlay  
  
 mysql\_slave:  
 image: mysql:5.7  
 env\_file:  
 - ./slave/mysql\_slave.env  
 container\_name: "mysql\_slave"  
 restart: "no"  
 ports:  
 - 5506:3306  
 depends\_on:  
 - mysql\_master  
 volumes:  
 - ./slave/data:/var/lib/mysql  
 command: server-id=2 --log-bin='mysql-bin-1.log' --relay-log='mysql-relay-bin.log' --binlog\_do\_db='mydb' --gtid\_mode='ON' --enforce\_gtid\_consistency='ON'  
 networks:  
 - overlay  
  
 mysql\_slave2:  
 image: mysql:5.7  
 env\_file:  
 - ./slave2/mysql\_slave2.env  
 container\_name: "mysql\_slave2"  
 restart: "no"  
 ports:  
 - 6606:3306  
 depends\_on:  
 - mysql\_master  
 volumes:  
 - ./slave2/data:/var/lib/mysql  
 command: --server-id=3 --log-bin='mysql-bin-1.log' --relay-log='mysql-relay-bin.log' --binlog\_do\_db='mydb' --gtid\_mode='ON' --enforce\_gtid\_consistency='ON'  
 networks:  
 - overlay  
  
networks:  
 overlay:

Команда --gtid\_mode='ON' --enforce\_gtid\_consistency='ON' запускает режим GTID

Команда --binlog\_format='ROW' включает row-based репликацию.

Запускаю на мастере. Даю права для юзеров слейвов на мастере:

GRANT REPLICATION SLAVE ON \*.\* TO "mydb\_slave\_user"@"%" IDENTIFIED BY "mydb\_slave\_pwd";

GRANT REPLICATION SLAVE ON \*.\* TO "mydb\_slave\_user2"@"%" IDENTIFIED BY "mydb\_slave\_pwd2";  
FLUSH PRIVILEGES;

Запускаю слейв 1:

CHANGE MASTER TO MASTER\_HOST='192.168.32.1', MASTER\_USER='mydb\_slave\_user', MASTER\_PASSWORD='mydb\_slave\_pwd', MASTER\_LOG\_FILE='mysql-bin-1.000003', MASTER\_LOG\_POS=895;  
START SLAVE;

Запускаю слейв 2:

CHANGE MASTER TO MASTER\_HOST='192.168.32.1', MASTER\_USER='mydb\_slave\_user2', MASTER\_PASSWORD='mydb\_slave\_pwd2', MASTER\_LOG\_FILE='mysql-bin-1.000003', MASTER\_LOG\_POS=895;  
START SLAVE;

Делаю insert на мастере и проверяю изменения на слейвах.

Устанавливаю плагин для полусинхронной репликации на мастере:

INSTALL PLUGIN rpl\_semi\_sync\_master SONAME 'semisync\_master.so';

SET GLOBAL rpl\_semi\_sync\_master\_enabled = 1;

SET GLOBAL rpl\_semi\_sync\_master\_wait\_for\_slave\_count = 2;

Устанавливаю плагин для полусинхронной репликации на слейве 1:

INSTALL PLUGIN rpl\_semi\_sync\_slave SONAME 'semisync\_slave.so'

SET GLOBAL rpl\_semi\_sync\_slave\_enabled = 1;

То же самое на слейве 2:

INSTALL PLUGIN rpl\_semi\_sync\_slave SONAME 'semisync\_slave.so'

SET GLOBAL rpl\_semi\_sync\_slave\_enabled = 1;

Настраиваю репликацию в приложении (первый хост – мастер)

spring:  
 datasource:  
 url: jdbc:mysql:replication://localhost:4406,localhost:5506,localhost:6606/mydb   
 username: root  
 password: 111

Даю нагрузку на запись (в логах приложения отмечаю сколько записей было создано):

wrk -t10 -c100 -d200s --latency http://172.21.80.1/user

Останавливаю контейнер мастера:

docker container stop mysql\_master

Перевожу слейв 1 в мастер:

STOP SLAVE;  
RESET MASTER;  
GRANT REPLICATION SLAVE ON \*.\* TO "mydb\_slave\_user2"@"%" IDENTIFIED BY "mydb\_slave\_pwd2";  
FLUSH PRIVILEGES;

Перевожу слейв 2 на новый мастер:

STOP SLAVE;  
CHANGE MASTER TO MASTER\_HOST='192.168.32.2', MASTER\_USER='mydb\_slave\_user2', MASTER\_PASSWORD='mydb\_slave\_pwd2';  
START SLAVE;

Проверяю кол-во записей в таблице USER на мастере и репликах. Потери транзакций не обнаружено. Кол-во строк в каждой базе оказалось одинкаковым.