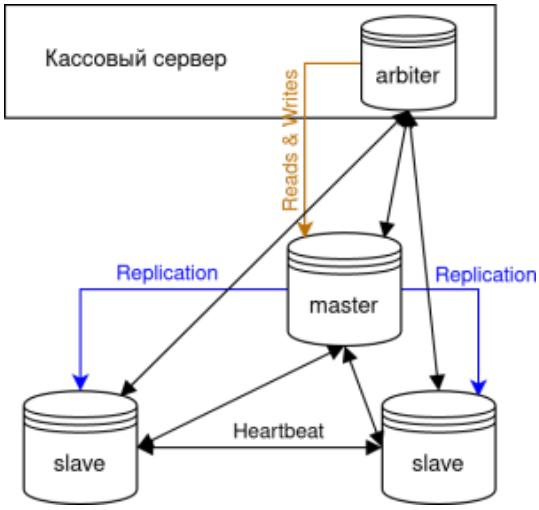
MongoDb Replica Set

Replica Set – кластер серверов MongoDb. Реализует механизм master-slave и автоматически выбирает нового мастера, если предыдущий отказал.

В Replica Set есть основной сервер (далее master), копии основного сервера (далее slave) и арбитр. Все операции "записи" производятся только в master, операции "чтения" могут происходить как с master так и со slave серверов. Откуда кассовый сервер будет производить чтение настраивается в applicaton. properties, далее будет описано как это сделать. Арбитр не копирует данные с master, и не может стать его приемником, этот сервер отвечает только за выбор нового master сервера. В теории можно обойтись и без арбитра, тогда выбор нового мастера будет осуществляться оставшимися в живых slave серверами после отказа master сервера, но с арбитром все будет куда быстрее. Так как арбитр содержит информацию о состоянии реплики и не копирует данные с master , то он довольно легковесный и его лучше поставить рядом с кассовым сервером чтобы исключить проблемы связанные с сетью и не потерять всю реплику.



Для начала необходимо установить mongodb на сервера и запустить их в одной реплике. На каждой машине (в том числе и на той где будет стоять арбитр) выполняем следующие действия (эти и дальнейшие действия выполняются от имени суперпользователя root, **имя реплики в примере rs0, можно заменить на свое**):

Установка mongodb

```
# Добавляем репу mongodb
curl http://update.artix.su/static/cs/install_scripts/stable/main/add_mongo_rep.sh -o /tmp/add_mongo_rep.sh
chmod +x /tmp/add_mongo_rep.sh
/tmp/add_mongo_rep.sh

apt-get update

# Устанавливаем mongodb
curl http://update.artix.su/static/cs/install_scripts/stable/cs_env/install_db.sh -o /tmp/install_db.sh
chmod +x /tmp/install_db.sh
/tmp/install_db.sh mongodb

# Добавляем mongodb в автозапуск
systemctl enable mongod

# Добавляем информацию о реплике
echo -e "\nreplication:\n    replSetName: rs0\n" >> /etc/mongod.conf

service mongod restart
```

Для инициализации реплики необходимо зайти на любой из настроенных серверов и прописать:

```
mongo

rs.initiate({ _id: "rs0", version: 1, members: [ { _id : 0, host : "<master_ip>:27017" } ] })
```

Сервер указанный сейчас в <master_ip> станет master-сервером.

В дальнейшем это может поменяться.

Если этот сервер станет недоступен, то master-сервером будет назначен один из slave серверов.

Далее необходимо добавить информацию об остальных серверах:

```
rs.add("<slave_ip>:27017")
```

Также добавляем арбитра:

```
rs.addArb(“<arbiter_ip>:27017”)
```

На этом настройка MongoDb ReplicaSet закончена.

Текущие настройки реплики можно посмотреть выполнив в mongo команду:

```
rs.status()
```

В завершении нужно настроить кассовый сервер на созданную реплику. Для этого открываем на редактирование файл /opt/artixcs-rest-split/application.properties и настраиваем подключение к арбитру (технически можно указать любой из серверов, но если связь с ним будет потеряна, то будет потеряна и вся реплика):

```
spring.data.mongodb.host=<arbiter_ip>
spring.data.mongodb.port=27017
spring.data.mongodb.replicaSet=rs0
```

Для настройки операций чтения преимущественно со slave-серверов, на кассовом сервере в конфигурационный файл /opt/artixcs-rest-split/application.properties добавить:

```
spring.data.mongodb.readPreference=secondaryPreferred
```

Для применения настроек необходимо перезапустить кассовый сервер.