

## Table of Contents

Introduction	1.1
0.	1.2
0.1 dble	1.2.1
0.2 dbleMyCat	1.2.2
0.3	1.2.3
0.4 docker	1.2.4
0.5 docker-compose	1.2.5
0.6	1.2.6
0.7 docker-composedble	1.2.7
1.	1.3
1.1 cluster.cnf	1.3.1
1.2 bootstrap.cnf	1.3.2
1.3 user.xml	1.3.3
1.4 db.xml	1.3.4
1.5 sharding.xml	1.3.5
1.6 log4j2.xml	1.3.6
1.7	1.3.7
1.7.1 MySQL offset-step	1.3.7.1
1.7.2 (Snowflake)	1.3.7.2
1.7.3 (Snowflake)	1.3.7.3
1.7.4 offset-step	1.3.7.4
1.8 cache	1.3.8
1.8.1 cache	1.3.8.1
1.8.2 ehcache	1.3.8.2
1.9	1.3.9
1.10	1.3.10
1.11	1.3.11
1.12	1.3.12
2.	1.4
2.0	1.4.1
2.1	1.4.2
2.1.1 select	1.4.2.1
2.1.2 set	1.4.2.2
2.1.3 show	1.4.2.3
2.1.4 switch	1.4.2.4
2.1.5 kill	1.4.2.5
2.1.6 stop	1.4.2.6
2.1.7 reload	1.4.2.7
2.1.8 rollback	1.4.2.8
2.1.9 offline	1.4.2.9
2.1.10 online	1.4.2.10
2.1.11 file	1.4.2.11
2.1.12 log	1.4.2.12
2.1.13	1.4.2.13
2.1.14 pause & resume	1.4.2.14
2.1.15	1.4.2.15
2.1.16 /	1.4.2.16
2.1.17 check @@metadata	1.4.2.17
2.1.18 release @@metadata	1.4.2.18
2.1.19 split	1.4.2.19
2.1.20 flow_control	1.4.2.20
2.1.21	1.4.2.21
2.2	1.4.3
2.2.1 MySQL offset-step	1.4.3.1
2.2.2	1.4.3.2
2.2.3	1.4.3.3
2.2.4 offset-step	1.4.3.4
2.3	1.4.4
2.4	1.4.5
2.5	1.4.6
2.5.1 XA	1.4.6.1
2.5.2 XA	1.4.6.2
2.5.3 XA	1.4.6.3
2.5.4 XA	1.4.6.4
2.5.5	1.4.6.5
2.6	1.4.7
2.7	1.4.8
2.8 &	1.4.9
2.9 gRPC	1.4.10
2.10 meta	1.4.11
2.10.1 Meta	1.4.11.1
2.10.2 Meta	1.4.11.2
2.10.3	1.4.11.3
2.10.4 View Meta	1.4.11.4
2.11	1.4.12
2.11.1	1.4.12.1
2.11.2	1.4.12.2
2.11.3	1.4.12.3
2.11.4	1.4.12.4
2.11.5 heartbeat	1.4.12.5
2.11.6	1.4.12.6
2.12	1.4.13
2.13	1.4.14
2.14 ER	1.4.15
2.15 global	1.4.16
2.16	1.4.17
2.17	1.4.18
2.18	1.4.19

2.19 reload	1.4.20
2.20	1.4.21
2.21 SQLtrace	1.4.22
2.22 KILL @@DDL_LOCK	1.4.23
2.23	1.4.24
2.23.1 MYSQL-HA	1.4.24.1
2.23.2	1.4.24.2
2.23.3	1.4.24.3
2.23.4 HA	1.4.24.4
2.24	1.4.25
2.25	1.4.26
2.26 client_found_rows	1.4.27
3.	1.5
3.1 DDL	1.5.1
3.1.1 DDL&Table Syntax	1.5.1.1
3.1.2 DDL&View Syntax	1.5.1.2
3.1.3 DDL&Index Syntax	1.5.1.3
3.1.4 DDL	1.5.1.4
3.1.5 DDL&Database_Syntax	1.5.1.5
3.1.6 ONLINE DDL	1.5.1.6
3.2 DML	1.5.2
3.2.1 INSERT	1.5.2.1
3.2.2 REPLACE	1.5.2.2
3.2.3 DELETE	1.5.2.3
3.2.4 UPDATE	1.5.2.4
3.2.5 SELECT	1.5.2.5
3.2.6 SELECT JOIN syntax	1.5.2.6
3.2.7 SELECT UNION Syntax	1.5.2.7
3.2.8 SELECT Subquery Syntax	1.5.2.8
3.2.9 LOAD DATA	1.5.2.9
3.2.10	1.5.2.10
3.3 Prepared SQL Syntax	1.5.3
3.4 Transactional, Savepoint and Locking Statements	1.5.4
3.4.1 Lock&unlock	1.5.4.1
3.4.2 XA	1.5.4.2
3.4.3	1.5.4.3
3.4.4 SET TRANSACTION Syntax	1.5.4.4
3.4.5 SAVEPOINT, ROLLBACK TO SAVEPOINT, and RELEASE SAVEPOINT Syntax	1.5.4.5
3.5 DAL	1.5.5
3.5.1 SET	1.5.5.1
3.5.2 SHOW	1.5.5.2
3.5.3 KILL	1.5.5.3
3.6	1.5.6
3.7 Utility Statements	1.5.7
3.8 Hint	1.5.8
3.9	1.5.9
3.10 (alpha)	1.5.10
3.11	1.5.11
4.	1.6
4.1	1.6.1
4.2	1.6.2
4.3	1.6.3
4.4 (Prepared Statements)	1.6.4
4.5	1.6.5
5.	1.7
5.1 druid	1.7.1
5.2	1.7.2
6.MySQL Server	1.8
6.1	1.8.1
6.2 INSERT	1.8.2
6.3 "show all tables"	1.8.3
6.4 message	1.8.4
6.5 information_schema	1.8.5
7.	1.9
7.1 SQL	1.9.1
7.2 dbleDemo	1.9.2
7.3	1.9.3
8.	1.10
8.1	1.10.1
8.2 MySQL-offset-step	1.10.2
9.sysbenchdble	1.11
9.1	1.11.1
9.2 dble	1.11.2
9.3 sysbench	1.11.3
A.Faq	1.12
A.1 ErrorCode	1.12.1
max Connections	1.12.1.1
Out Of Memory Error	1.12.1.2
The Problem Of Hint	1.12.1.3
NestLoop Parameters Lead To Temptable Exception	1.12.1.4
Can't Get Variables From ShardingNode	1.12.1.5
Port already in use:1984	1.12.1.6
Sharding Column Cannot Be Null	1.12.1.7
A.2	1.12.2
How To Use Explain To Resolve The Distribution Rules Of Group By	1.12.2.1
Hash And ConsistentHashing And Jumpstringhash	1.12.2.2
A.3	1.12.3
ToBeContinued2	1.12.3.1



## dble

gitbook SUMMARY.md

### PDF

[PDF](#)

dble

- github: [github.com/actiontech/dble](https://github.com/actiontech/dble)
- github: [github.com/actiontech/dble-test-suite](https://github.com/actiontech/dble-test-suite)
- github: [github.com/actiontech/dble-docs-cn](https://github.com/actiontech/dble-docs-cn)
- github pages: [actiontech.github.io/dble-docs-cn](https://actiontech.github.io/dble-docs-cn)
- [DBLE](#)
- QQ group: 669663113
- 



3.20.10.xdbletagerelease

dble

dble :

- : 400-820-6580
- : 86-13718877200,
- : 86-18503063188,
- : 86-18930110869,
- : 86-13540040119,

**0**

- [0.1 dble](#)
- [0.2 dbleMyCat](#)
- [0.3](#)
- [0.4 docker](#)
- [0.5 docker-compose](#)
- [0.6](#)
- [0.7 docker-composedble](#)

## 0.1 dble

### 0.1.1 dble

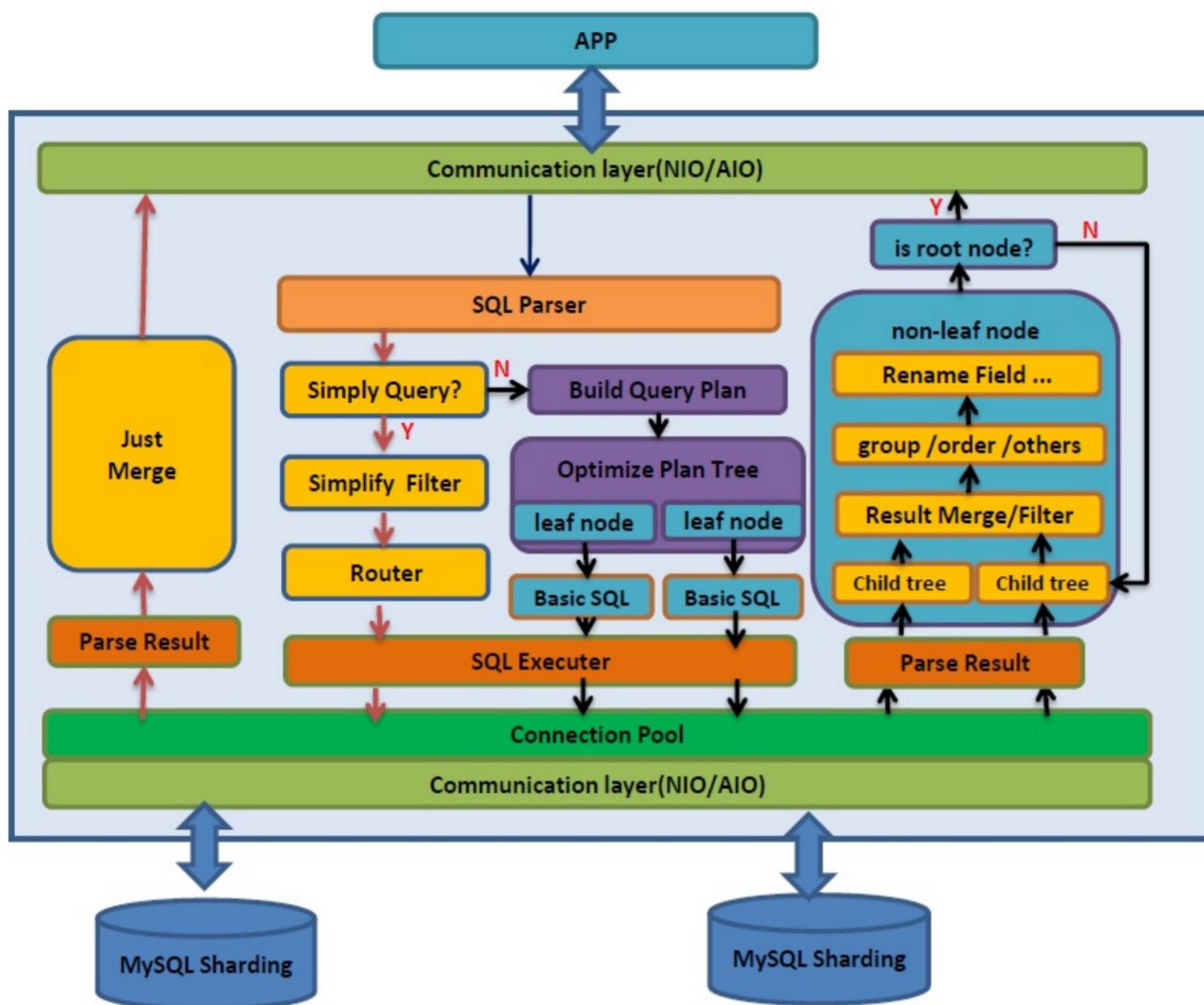
dblemysql

- dbleMySQL
- **Mysql MySQLMySQL**
- dble
- **SQL SQL 92MySQLSQLgroup by order by distinctjoinunionsub-query**
- ER
- XAXAMySQL-5.7XA TransactionMySQL

### 0.1.2 dble

- dble MySQLCatMySQLCat
- MySQL / bugs [dbleMySQLCat](#)

### 0.1.3 dble



## 0.2 dbleMyCat

dble

### 0.2.1 mycatmycatbug

- “double free”JVM #4
- XA #21
- wherewhereselect \* from customer wher id=1; #126
- insert into table values(1)(2)mycatSQL
- sql #92
- / #43,#31,#44
- between A and Bhash #23
- alter tableinsert...on duplicate...update...in () #24, #25,#26 ,#5
- #1
- ER, #13
- sharding:join#17

### 0.2.2

- SQLcreate table if not exists...alter table add/drop [primary] key...
- IO: dbleIO 2
- #56
- - ZK
  - ZK ID
  - #489
  - - insert into table1(id,name) values(next value for MYCATSEQ\_GLOBAL,'test');
    - 1insert into table1(name) values('test');
    - 2insert into table1 values('test');
    - bigint
- ERER
- ERER
- schemacheckSQLschema
- conf/index\_to\_charset.properties
- 
- SQLUPDATE/DELETE/INSERT

### 0.2.3

- ShareJoin(join,union,subquery)/
- showdescolumnsinsert #7
- - 
  -
- SQL
  - mysql> explain select \* from sharding\_two\_node a inner join sharding\_four\_node b on a.id =b.id;

```
+-----+-----+-----+
| SHARDING_NODE | TYPE    | SQL/REF           |
+-----+-----+-----+
| dn1.0          | BASE SQL | select `a`.`id`, `a`.`c_char`, `a`.`ts`, `a`.`si` from `sharding_two_node` `a` ORDER BY `a`.`id` ASC |
| dn2.0          | BASE SQL | select `a`.`id`, `a`.`c_char`, `a`.`ts`, `a`.`si` from `sharding_two_node` `a` ORDER BY `a`.`id` ASC |
| dn1.1          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| dn2.1          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| dn3.0          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| dn4.0          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| merge.1         | MERGE   | dn1.0, dn2.0      |
| merge.2         | MERGE   | dn1.1, dn2.1, dn3.0, dn4.0 |
| join.1          | JOIN    | merge.1, merge.2 |
+-----+-----+-----+
9 rows in set (0.00 sec)
```

- set
- set charset/names
- :XA
- 
- DUAL
- (CC++)
- 
- Druid,
- fastjson,
- reload
- MySQLGUI/Driver
- ,
- RocksDB
- mysqldumpslow pt-query-digest
- Trace
- MySQL
- Prepared SQL Statement Syntax
- - The Subquery as Scalar Operand
  - Comparisons Using Subqueries
  - Subqueries with ANY, IN, or SOME
  - Subqueries with ALL
  - Subqueries with EXISTS or NOT EXISTS
  - Derived Tables (Subqueries in the FROM Clause)
- dble View
- MySQL8.0
- 
- 
- 
- DDL
- 
- 

### 0.2.4

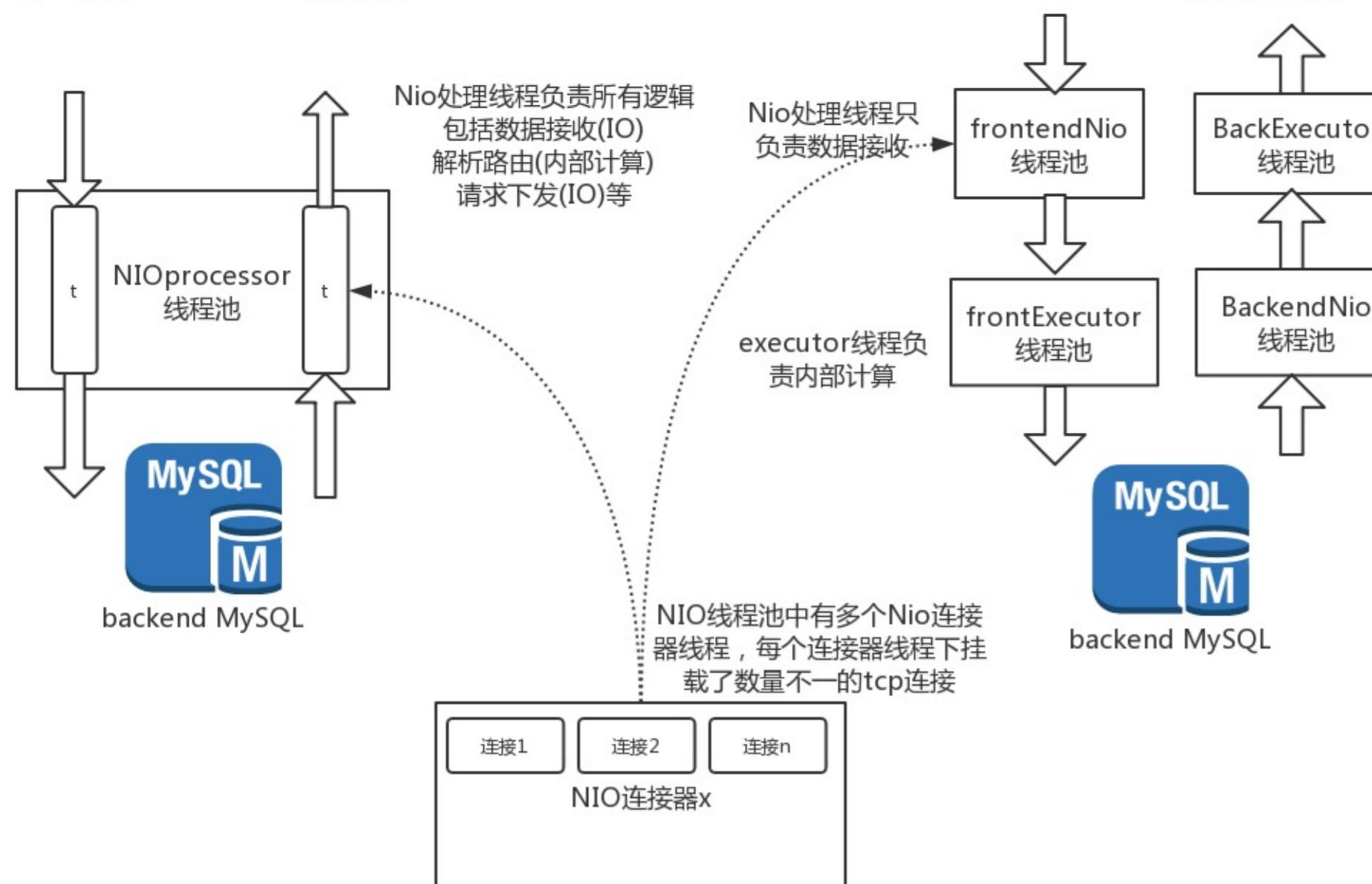
- HASH
- 
- set
- 
- writeType/writeType = 1
- handleDistributedTransactions

### 0.2.5

DBLEMyCAT

### 0.2.6 2

Mycat以及dble 2.18.02.0之前版本



Nio处理器 (处理线程) 内部为挂载不定数量个连接，并且循环响应每个连接的请求

在数据处理和数据接收进行线程分割之后(dble 2.18.02)，使得dble可以并发响应挂载在同一个NIO连接器(同一个processor线程)上的请求

e.g.

恰好我们存在场景连接1，2同时有请求过来，旧版本需要循环处理连接1，2的任务，在连接1的任务处理完成之前，连接2的任务无法进行处理

新的IO结构中，连接1的数据被接收完毕之后，NIO线程就可以接收连接2的数据，并且此时连接1的数据已经在executor线程池进行处理中，连接1，2之间的任务执行变成并行

## 0.3

### 0.3.1

- dbledbldble

### 0.3.2

dbe

- MySQL
 

```
dbledbldble
MySQL
A:$url=ip1:3306,$user=test,$password=testPsw
B:$url=ip2:3306,$user=test,$password=testPsw
/etc/hostsMySQL "NO ROUTE TO HOST"
```

- JVM

```
dbledbldblejava1.8JAVA_HOME
```

### 0.3.3

- <https://github.com/actiontech/dble/releases>

- dble

```
mkdir -p $working_dir
cd $working_dir
tar -xvf actiontech-dble-$version.tar.gz
cd $working_dir/dble/conf
mv cluster_template.cnf cluster.cnf
mv bootstrap_template.cnf bootstrap.cnf
mv db_template.xml db.xml
mv user_template.xml user.xml
mv sharding_template.xml sharding.xml
```

### 0.3.4 dble

- db.xml

```
<dbInstance name="instanceM1" url="ip1:3306" user="your_user" password="your_psw" maxCon="1000" minCon="10"
           primary="true" readWeight="1" id="xx1">
</dbInstance>
```

```
<dbInstance name="instanceM2" url="ip2:3306" user="your_user" password="your_psw" maxCon="1000" minCon="10"
           primary="true">
</dbInstance>
```

### 0.3.5

- 

```
cd $working_dir/dble
bin/dble start
```

- tail -f logs/wrapper.log
- mysql
 

```
dbledbldble654321 mysql -pP9066 -h 127.0.0.1 -u man1
```
- mysqlschema

```
create database @@shardingnode='dn$1-6';
```

- mysql
 

```
dbledbldble123456 mysql -p -P8066 -h 127.0.0.1 -u root
```
- mysql

```
use testdb;
drop table if exists tb_enum_sharding;
create table if not exists tb_enum_sharding (
    id int not null,
    code int not null,
    content varchar(250) not null,
    primary key(id)
)engine=innodb charset=utf8;
insert into tb_enum_sharding values(1,10000,'1'),(2,10010,'2'),(3,10000,'3'),(4,10010,'4');
```

## 0.4 (docker)

### 0.4.1

- dockerhubdbledbledemon

### 0.4.2

- docker
- mysql

### 0.4.3

docker

```
docker network create -o "com.docker.network.bridge.name"="dbe-net" --subnet 172.18.0.0/16 dbe-net
docker run --name backend-mysql1 --ip 172.18.0.2 -e MYSQL_ROOT_PASSWORD=123456 -p 33061:3306 --network=dbe-net -d mysql:5.7 --server-id=1
docker run --name backend-mysql2 --ip 172.18.0.3 -e MYSQL_ROOT_PASSWORD=123456 -p 33062:3306 --network=dbe-net -d mysql:5.7 --server-id=2
sleep 30
docker run -d -i -t --name dble-server --ip 172.18.0.5 -p 8066:8066 -p 9066:9066 --network=dbe-net actiontech/dble:latest
```

docker33061,33062mysql80669066

mysqldble

### 0.4.4

mysql80669066docker  
8066 (SQL) root/123456  
9066 () man1/654321

dble-server/opt/dble/conf/shardingxml

mysql

```
#dbe sql
mysql -P8066 -u root -p123456 -h 127.0.0.1
#dbe
mysql -P9066 -u man1 -p654321 -h 127.0.0.1
#mysql1
mysql -P33061 -u root -p123456 -h 127.0.0.1
#mysql2
mysql -P33062 -u root -p123456 -h 127.0.0.1
```

### 0.4.5

```
docker stop backend-mysql1
docker stop backend-mysql2
docker stop dble-server
docker rm backend-mysql1
docker rm backend-mysql2
docker rm dble-server
docker network rm dbe-net
```

### 0.4.6 docker-compose

docker-composegithub

```
wget https://raw.githubusercontent.com/actiontech/dble/master/docker-images/docker-compose.yml
```

docker-composemysqldble

```
docker-compose up
```

dblequick-startdble-server8066/9066 root/1234568066mysql33061/33062root/123456

```
docker-compose stop
docker-compose rm
```

### 0.4.7 docker-compose

dble,dble dble

dblevolumesdockerdble/confdble

docker-compose.yml dble-server

```
volumes:
  - /opt/test/conf:/opt/self_conf
```

/opt/test/conf/opt/self\_conf dble-server

```
command: ["/opt/dble/bin/wait.sh", "backend-mysql1:3306", "--", "/opt/self_conf/docker_init_start.sh"]
```

/opt/dble/bin/docker\_init\_start.sh self\_conf

- /opt/dble/conf
- /opt/dble/bin/dble startdble
- /opt/dble/bin/wait-for-it.sh dble 8066 dble
- mysqldble

## 0.5 (docker-compose)

### 0.5.1

- dbledocker-composedblequick start  
sqlable quick start

### 0.5.2

- docker
- docker-compose
- mysql

### 0.5.3

docker-composegithub

```
wget https://raw.githubusercontent.com/actiontech/dble/master/docker-images/docker-compose.yml
```

docker-composemysqldble

```
docker-compose up
```

dblequick-startdble-server8066/9066 root/1234568066mysql33061/33062root/123456

```
docker-compose stop  
docker-compose rm
```

### 0.5.4

mysql80669066docker  
8066 (SQL) root/123456  
9066 () man1/654321

dble-server/opt/dble/conf/sharding.xml

### 0.5.5 dble

dbledocker-composedble  
docker-compose.ymldble-comp osedble-server  
actiontech/dble:latest/opt/dble/bin/wait-for-it.shmysql1mysqlTCP  
docker\_init\_start.sh

```
#!/bin/sh  
  
echo "dble init&start in docker"  
  
sh /opt/dble/bin/dble start  
sh /opt/dble/bin/wait-for-it.sh 127.0.0.1:8066  
mysql -P9066 -u man1 -h 127.0.0.1 -p654321 -e "create database @@shardingNode ='dn1,dn2,dn3,dn4'"  
mysql -P8066 -u root -h 127.0.0.1 -p123456 -e "source /opt/dble/conf/testdb.sql" testdb  
  
echo "dble init finish"  
  
/bin/bash
```

dble->dble->dblecreate databasemysqldatabase->sqlable  
dble-server

- shdble
- dble-servervolumeslinuxdble-server(dble/opt/dble)
- dble-server

### e.g.

docker-compose.yml

```
dble-server:  
  image: actiontech/dble:latest  
  container_name: dble-server  
  hostname: dble-server  
  privileged: true  
  stdin_open: true  
  tty: true  
  volumes:  
    - ./:/opt/init/  
  command: ["/opt/dble/bin/wait-for-it.sh", "backend-mysql1:3306", "--", "/opt/init/customized_script.sh"]  
  ports:  
    - "8066:8066"  
    - "9066:9066"  
  depends_on:  
    - "mysql1"  
    - "mysql2"  
  networks:  
    net:  
      ipv4_address: 172.18.0.5
```

/

```
cluster.cnf bootstrap.cnf db.xml user.xml sharding.xml init.sql customized_script.sh
```

customized\_script.sh

```
#!/bin/sh  
  
echo "dble init&start in docker"  
  
cp cluster_template.cnf /opt/dble/conf/  
cp bootstrap_template.cnf /opt/dble/conf/  
cp db_template.xml /opt/dble/conf/  
cp user_template.xml /opt/dble/conf/  
cp sharding_template.xml /opt/dble/conf/  
  
sh /opt/dble/bin/dble start  
sh /opt/dble/bin/wait-for-it.sh 127.0.0.1:8066  
mysql -P9066 -u man1 -h 127.0.0.1 -p654321 -e "create database @@shardingNode ='dn1,dn2,dn3,dn4'"  
mysql -P8066 -u root -h 127.0.0.1 -p123456 -e "source /opt/init/init.sql" testdb  
  
echo "dble init finish"  
  
/bin/bash
```

### 0.5.6

docker-compose stop/down

**0.6****0.6.1**

- dble
  - mysql
  - mysql
  - mysql

**0.6.2**

- ,
- joinjoin
- :
  - 23
  - QPSTPS
- QPSTPSSQL

**0.6.3**

- 
-

## 0.7 docker-composedle

### 0.7.1

- docker-composemgr(mysql)dblequick startdble

### 0.7.2

- docker
- docker-compose
- mysql

### 0.7.3

#### docker-compose-mgr.yml 4 docker-compose up

1. dbledocker-images:

```
docker-images
  +-- dockerfile-Mgr      actiontech/dble-mgr:latest
  +-- docker-compose-mgr.yml  mysql(mgr)dble-server
  +-- quick-start
    +-- start-mgr.sh      mgr
    +-- docker_init_start.sh  start-mgr.sh dble
    +-- db_mgr.xml        6mysql
    +-- bootstrap.cnf      useOuterHafalse
  +-- ...
  +-- ...
```

2. bootstrap.cnfuseOuterHafalse

```
-DuseOuterHa=false
```

3. Dockerfile-M gactiontech/dble-mgr:latest

```
[root@localhost]docker-images# docker build -f="Dockerfile-Mgr" -t="actiontech/dble-mgr:latest" .
[root@localhost]docker-images# docker images
REPOSITORY          TAG           IMAGE ID            CREATED             SIZE
actiontech/dble-mgr  latest        d773d83fe832   46 hours ago       711MB
```

4. docker-compose-mgr.yml mysql(mgr)dble-server

```
[root@localhost]# cd docker-images
[root@localhost]docker-images# docker-compose -f docker-compose-mgr.yml up
[root@localhost]docker-images# docker-compose -f docker-compose-mgr.yml ps
     Name          Command          State          Ports
-----+-----+-----+-----+
dble-server /opt/dble/bin/wait-for-it ...  Up   0.0.0.0:8066->8066/tcp, 0.0.0.0:9066->9066/tcp
mgr-a-1   docker-entrypoint.sh mysql ...  Up   0.0.0.0:3307->3306/tcp, 33060/tcp
mgr-a-2   docker-entrypoint.sh mysql ...  Up   0.0.0.0:3308->3306/tcp, 33060/tcp
mgr-a-3   docker-entrypoint.sh mysql ...  Up   0.0.0.0:3309->3306/tcp, 33060/tcp
mgr-b-1   docker-entrypoint.sh mysql ...  Up   0.0.0.0:3310->3306/tcp, 33060/tcp
mgr-b-2   docker-entrypoint.sh mysql ...  Up   0.0.0.0:3311->3306/tcp, 33060/tcp
mgr-b-3   docker-entrypoint.sh mysql ...  Up   0.0.0.0:3312->3306/tcp, 33060/tcp
```

5. mgr

```
[root@localhost]docker-images# docker exec mgr-a-1 mysql -h127.0.0.1 -p3306 -uroot -p123456 \
-e "SHOW STATUS LIKE 'group_relication_primary_member';" \
-e "SELECT * FROM performance_schema.replication_group_members;"
```

Variable_name	Value
group_relication_primary_member   72da84d7-0c4b-11eb-9f0e-0242ac120002	

```
+-----+-----+-----+-----+
| CHANNEL_NAME | MEMBER_ID | MEMBER_HOST | MEMBER_PORT | MEMBER_STATE |
+-----+-----+-----+-----+
| group_rePLICATION_applier | 72da84d7-0c4b-11eb-9f0e-0242ac120002 | mgr-a-1 | 3306 | ONLINE |
| group_rePLICATION_applier | 7314effd-0c4b-11eb-ba28-0242ac120004 | mgr-a-3 | 3306 | ONLINE |
| group_rePLICATION_applier | 733b00fe-0c4b-11eb-bbea-0242ac120003 | mgr-a-2 | 3306 | ONLINE |
+-----+-----+-----+-----+
```

```
[root@localhost]docker-images# docker exec mgr-b-1 mysql -h127.0.0.1 -p3306 -uroot -p123456 \
-e "SHOW STATUS LIKE 'group_relication_primary_member';" \
-e "SELECT * FROM performance_schema.replication_group_members;"
```

Variable_name	Value
group_rePLICATION_primary_member   728c327d-0c4b-11eb-9300-0242ac120005	

```
+-----+-----+-----+-----+
| CHANNEL_NAME | MEMBER_ID | MEMBER_HOST | MEMBER_PORT | MEMBER_STATE |
+-----+-----+-----+-----+
| group_rePLICATION_applier | 728c327d-0c4b-11eb-9300-0242ac120005 | mgr-b-1 | 3306 | ONLINE |
| group_rePLICATION_applier | 732c5b3b-0c4b-11eb-9eb1-0242ac120007 | mgr-b-3 | 3306 | ONLINE |
| group_rePLICATION_applier | 733c6350-0c4b-11eb-b0fb-0242ac120006 | mgr-b-2 | 3306 | ONLINE |
+-----+-----+-----+-----+
```

6. dble, dble-server/opt/dble/logs/

```
[root@localhost]docker-images# docker exec -it dble-server bash
[root@dbe-server /]# less /opt/dble/logs/wrapper.log
[root@dbe-server /]# less /opt/dble/logs/dble.log
```

```
#bootstrap.cnfuseOuterHafalse
[root@dbe-server /]# less /opt/dble/logs/custom_mysql_ha.log
```

7.

```
[root@dbe-server /]# exit
[root@localhost]docker-images# docker-compose -f docker-compose-mgr.yml down
```

### 0.7.5

1. bootstrap.cnfuseOuterHafalse()

2. dble-server/opt/dble/conf/db.xml

```
[root@localhost]docker-images# docker exec -it dble-server bash
[root@dbe-server /]# cat /opt/dble/conf/db.xml
```

```
<dbGroup name="dbGroup1" rwSplitMode="2" delayThreshold="100">
  <heartbeat>show slave status</heartbeat>
  <dbInstance name="instanceM1" url="172.18.0.2:3306" user="root" password="123456" maxCon="300" minCon="10"
    primary="true" readWeight="1" id="xx1">
  </dbInstance>
  <dbInstance name="instanceS1" url="172.18.0.3:3306" user="root" password="123456" maxCon="1000" minCon="10" readWeight="2">
    <property name="testOnCreate">false</property>
  </dbInstance>
  <dbInstance name="instanceS2" url="172.18.0.4:3306" user="root" password="123456" maxCon="1000" minCon="10" readWeight="2">
    <property name="testOnCreate">false</property>
  </dbInstance>
</dbGroup>
```

```
[root@dbe-server /]# exit
```

3. mgr-a-1dble-servercustom\_mysql\_ha.log 172.18.0.2:3066...is not alive dbGroup1 instanceS1

```
[root@localhost]docker-images# docker-compose -f docker-compose-mgr.yml stop mgr-a-1
[root@localhost]docker-images# docker exec -it dble-server bash
[root@dbe-server /]# less /opt/dble/logs/custom_mysql_ha.log
...
```

```
2020-10-12 07:05:08 [DBLEDbGroupsCheck] [INFO] DbInstance 172.18.0.2:3306 in dbGroup1 is not alive!
2020-10-12 07:05:08 [DBLEDbGroupsCheck] [INFO] DbInstance 172.18.0.3:3306 in dbGroup1 is normal!
2020-10-12 07:05:08 [DBLEDbGroupsCheck] [INFO] DbInstance 172.18.0.4:3306 in dbGroup1 is normal!
2020-10-12 07:05:08 [DBLEDbGroupsCheck] [INFO] DbInstance 172.18.0.5:3306 in dbGroup2 is normal!
2020-10-12 07:05:08 [DBLEDbGroupsCheck] [INFO] DbInstance 172.18.0.6:3306 in dbGroup2 is normal!
2020-10-12 07:05:08 [DBLEDbGroupsCheck] [INFO] DbInstance 172.18.0.7:3306 in dbGroup2 is normal!
...

```

```
[root@dbe-server /]# cat /opt/dble/conf/db.xml
```

```
<dbGroup name="dbGroup1" rwSplitMode="2" delayThreshold="100">
  <heartbeat>show slave status</heartbeat>
  <dbInstance name="instanceM1" url="172.18.0.2:3306" user="root" password="123456" maxCon="300" minCon="10"
    readWeight="1" id="xx1">
  </dbInstance>
  <dbInstance name="instanceS1" url="172.18.0.3:3306" user="root" password="123456" maxCon="1000" minCon="10" readWeight="2" primary="true">
    <property name="testOnCreate">false</property>
  </dbInstance>
  <dbInstance name="instanceS2" url="172.18.0.4:3306" user="root" password="123456" maxCon="1000" minCon="10" readWeight="2">
    <property name="testOnCreate">false</property>
  </dbInstance>
</dbGroup>
```

## 1.dble

- - cluster.cnf
  - bootstrap.cnf:JVM dble
  - user.xml:dble
  - db.xml
  - sharding.xml
  - log4j.xmllog4j2.xml
  - 
  - cache
  - 
  - 
  -
- - /logs/wrapper.logdble
  - /logs/dble.logdbledble
- -

dble 3.20.07.0 [2.20.04.0](#)

## 2.

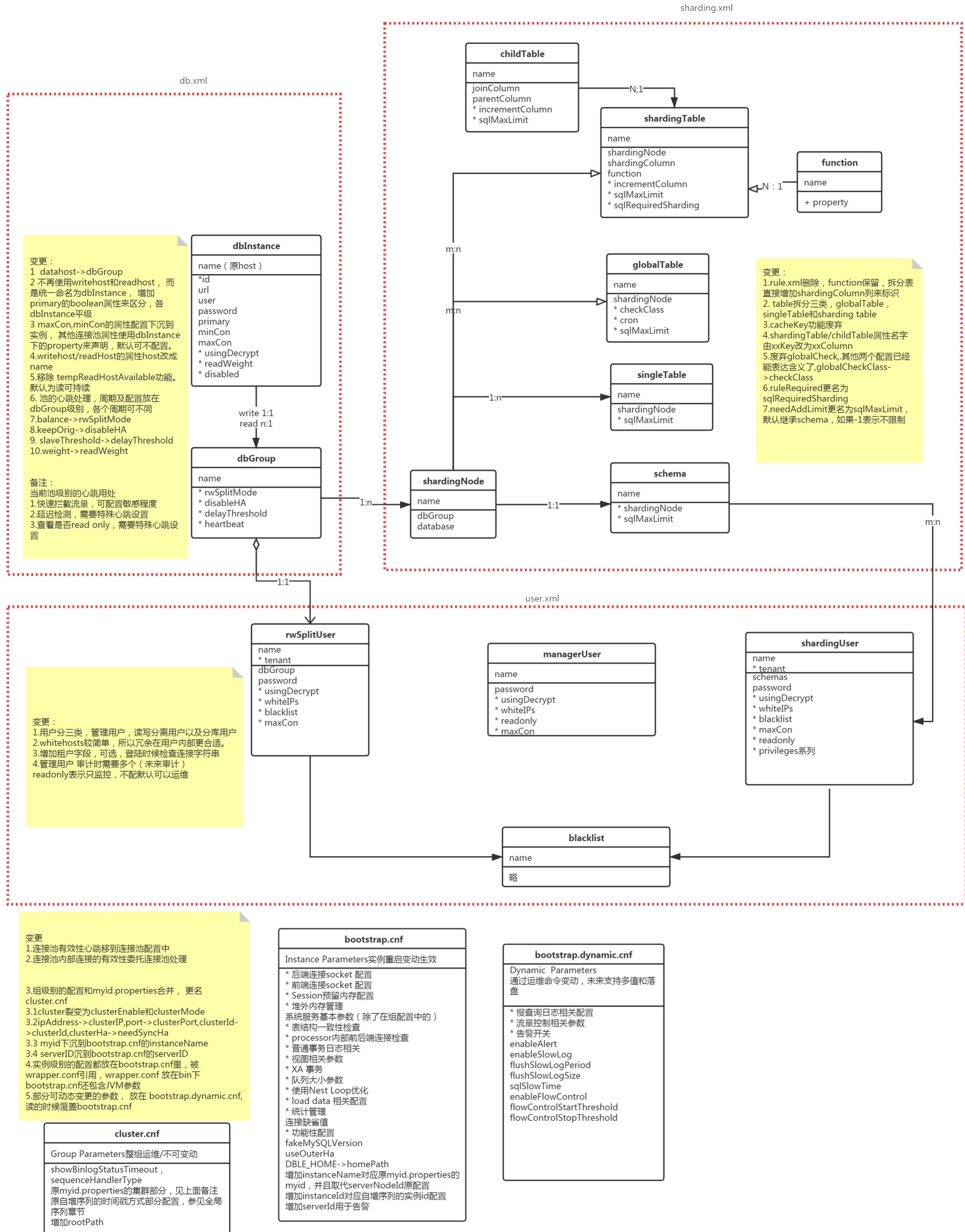
[dble\\_update\\_config](#) 2.20.04.0 3.20.07.02.20.04.0

```
dble_update_config [-i=read_dir] [-o=write_dir] [-p=rootPath]

read_dir/write_dir: rootPath:zk,          /dble , ucore,      universe/dble
:
my.id.properties
wrapper.conf
server.xml
schema.xml
rule.xml
log4j2.xml
cacheservice.properties(option)
sequence_distributed_conf.properties for type3 (option)
sequence_time_conf.properties for type2 (option)

:
cluster.cnf
bootstrap.cnf
user.xml
db.xml
sharding.xml
log4j2.xml
cacheservice.properties(option)
```

## 3.



## 1.1 cluster.cnf

```
key=valuedblekey/value
dbleDDL
.dble(zk).dble
```

	&	/	
clusterEnable		true/falsefalse	clusterIPclusterPortrootPathclusterID
clusterMode		zk/ucoreclusterEnable	zkzookeeperucore
clusterIP	IP	clusterEnable	clusterModezkzk10.186.19.aa:2281,10.186.60.bb:2281clusterModeucoreucoreip
clusterPort			clusterModezkclusterModeucoreucore
rootPath		clusterEnable	
clusterId	dble	clusterEnable	dbledble
needSyncHa	ha	true/falsefalse	trueuseOuterHatrue
showBinlogStatusTimeout	binlog	60000 ,	binlog
sequenceHandlerType		1~42	1MySQL offset-step sequence 2(Snowflake) 3time(Snowflake) 4offset-step
sequenceStartTime		2010-11-04 09:42:54	sequenceHandlerType23
sequenceInstanceByZk	timezkid	true/false,true	sequenceHandlerType3clusterModezk

### 1.1.1

```
clusterEnable=false
#showBinlogStatusTimeout=60000
sequenceHandlerType=2
#sequenceStartTime=2010-11-04 09:42:54
#sequenceInstanceByZk=true
```

### 1.1.2 ZK

```
clusterEnable=true
clusterMode=zk
clusterIP=10.186.19.aa:2281,10.186.60.bb:2281
rootPath=/dble
clusterId=cluster-1
#needSyncHa=false
#showBinlogStatusTimeout=60000
sequenceHandlerType=2
#sequenceStartTime=2010-11-04 09:42:54
#sequenceInstanceByZk=true
```

### 1.1.3 ucORE

```
clusterEnable=true
clusterMode=uCore
clusterIP=10.186.19.aa,10.186.60.bb
clusterPort=5700
rootPath=universe/dble
clusterId=cluster-1
#needSyncHa=false
#showBinlogStatusTimeout=60000
sequenceHandlerType=2
#sequenceStartTime=2010-11-04 09:42:54
#sequenceInstanceByZk=true
```

## 1.2 bootstrap.cnf

dblewrapper.cnfwrapper

### 1.2.1 jvm

JVM,

```
-agentlib:jdwp=transport=dt_socket,server=y,address=8088,suspend=n
-server
-XX:+AggressiveOpts
-Dfile.encoding=UTF-8
-Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=1984
-Dcom.sun.management.jmxremote.authenticate=false
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.host=127.0.0.1
-Xmx4G
-Xms1G
-XX:MaxDirectMemorySize=2G
```

JVM JVM

MaxDirectMemorySize 81917M 79G bufferPoolPageSize\*bufferPoolPageSize

bufferPoolPageSize = (MaxDirectMemorySize \* 0.8 / bufferPoolPageSize), 32767

bufferPoolPageSize = 2M

dble=0.6 ()

Xmx = 0.4 dble

MaxDirectMemorySize = 0.6 \* dble

bufferPoolPageSize bufferPoolPageSize MaxDirectMemorySize

### 1.2.2 dble

dble,-Dkey=value

JSWdbleJSW

wrapper.app.parameter\_file

		/			
homePath	slowlogsviewConfatxalogload data temp			,	
instanceName			xa		
instanceId	id			sequenceHandlerType=2 3. sequenceHandlerType=2 0~1023 sequenceHandlerType=3 0~511	
serverId		IP	dble		
bindIp	IP	"0.0.0.0"	IP		IP,
serverPort		8066			
managerPort		9066			
maxCon		0	maxCon0,maxCon,.maxcon	maxCommander	
processors	NIO	java ,	IO		
backendProcessors	NIO	java ,	IO		
processorExecutor		(2,)			
backendProcessorExecutor		(2,)			
complexExecutor		(2,88)			
writeToBackendExecutor	SQL	(2,)	SQL		
fakeMySQLVersion	dblemysql	NULL	MySql		MYSQL
serverBacklog	tcp backlog	2048	tcp backlog		
usePerformanceMode		0/	DbleCPU,		1-0-
useOuterHa		true	dble		true/false
charset		utf8mb4			

maxPacketSize		4×1024×1024	dble(+1024)dbInstancedbInstance-1024.1024SQL	
txIsolation		3	SQL dbledbInstancesessionsessionSQLsessionset	1-READ_UNCOMMITTED 2-READ_COMMITTED 3-REPEATABLE_READ 4-SERIALIZABLE
autocommit		1	dbledbInstance sessionSQLsessionset	0/1
useCompression		0	mysql	1 - 0 -
capClientFoundRows	Client_Found_Rows	false	dbleClient_Found_Rows	true - false -
usingAIO	AIO	0	AIONIO	1 - 0 -
useThreadUsageStat		0/	show @@thread_used	1-0-
useCostTimeStat		0/	BTraceCostTime.jav,show @@cost_time	1-0-
maxCostStatSize		100	show @@cost_time	
costSamplePercent		1%	costSamplePercent	
checkTableConsistency		0	1 DB	1-,0-
checkTableConsistencyPeriod		30×60×1000 ,		
sqlExecuteTimeout		300 ,		
processor	idleTimeout	10 × 60 × 1000 ,	processor	
	processorCheckPeriod	processor	1000 ,	processor
socket	backSocketSoRcvbuf	1024×1024×4 ,	buffer	
	backSocketSoSndbuf	1024×1024 ,	buffer	
	backSocketNoDelay	Nagle	1/	1-, 0-
socket	frontSocketSoRcvbuf	1024 × 1024 ,		
	frontSocketSoSndbuf	1024×1024×4 ,	buffer	
	frontSocketNoDelay	Nagle	1	1-, 0-
Session	orderMemSize	sessionorder	4,M	sessionorder by
	otherMemSize	session	4,M	sessionsubQuerydistinctd
	joinMemSize	sessionjoin	4M	sessionjoin
BufferPool	bufferPoolChunkSize	4096 ,		
	bufferPoolPageNumber	0.8 × MaxDirectMemory Size / bufferPoolPageSize(default 2M)	bufferPoolPageSize 1.4_wrapper.conf	
	bufferPoolPageSize	1024 * 1024 * 2,  <i>MaxDirectMemorySize( 1.4_wrapper.conf)bufferPoolPageNumber *</i> <i>bufferPoolPageSizeOOM</i>	bufferPoolPageNumbe,  <i>MaxDirectMemorySize( 1.4_wrapper.conf)bufferPoolPageNumber *</i> <i>bufferPoolPageSizeOOM</i>	
	mappedFileSize	1024×1024×64 ,	,	

	bufferUsagePercent		80,	resultSetMapClearclearBigSQLResultSetMaps	0-100
	useSqlStat	SQL	1/	SQL recycleSqlStat5SQL	1-0-
	clearBigSQLResultSetMaps		600×1000 ,	resultSetMapClear	
	sqlRecordCount		10 ,	recycleSqlStatsql,SQL	
	maxResultSet		512×1024 ,	SQL	
	recordTxn	log	0	log	1-0-
	transactionLogBaseDir	log	/txlogs	log	
	transactionLogBaseName	log	server-tx		
	transactionRotateSize		16 ,M		
XA	xaRecoveryLogBaseDir	xatm	dble/tmlogs/	XADblexa	
	xaRecoveryLogBaseName	xatm	tmlog		
	xaSessionCheckPeriod	XA	1000 ms	server session	
	xaLogCleanPeriod	XALog	1000 ms	server XA log	
	xaRetryCount	XA	0	XA0	
	viewPersistenceConfBaseDir		dble/viewConf	,	
	viewPersistenceConfBaseName		viewJson	,	
	joinQueueSize	join,	1024		
	mergeQueueSize	merge,	1024		
	orderByQueueSize	,	1024		
Nest Loop	useJoinStrategy	nest loop		joinwhereSQL	true false
	nestLoopConnSize		4		
	nestLoopRowsSize		2000		
	enableSlowLog		0		01
	slowLogBaseDir		dble/slowlogs		
	slowLogBaseName		slow-query	(.log)	
	flushSlowLogPeriod		1		
	flushSlowLogSize		1000	1	
	sqlSlowTime		100		0
load data	maxCharsPerColumn		65535		
	maxRowSizeToFile		10000	load dataOOM	
	enableFlowControl	true/false	false		true/false
	flowControlStartThreshold		4096		
	flowControlStopThreshold		256		

**1.2.3 bootstrap.dynamic.cnf**

```
bootstrap.dynamic.cnf
enableAlert
enableSlowLog
flushSlowLogPeriod
flushSlowLogSize
sqlSlowTime
enableFlowControl
flowControlStartThreshold
flowControlStopThreshold
```

**1.2.4**

```
#encoding=UTF-8
-agentlib:jdw=transport=dt_socket,server=y,address=8088,suspend=
-server
-XX:+AggressiveOpts
-Dfile.encoding=UTF-8
-Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=1984
-Dcom.sun.management.jmxremote.authenticate=false
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.host=127.0.0.1
-Xmx4G
-Xms1G
-XX:MaxDirectMemorySize=2G
# base config
-DhomePath=.
-DinstanceName=1
# valid for sequenceHandlerType=2 or 3
-DinstanceId=1
-DserverId=xxx1
# -BindIp=0.0.0.0
# -serverPort=8066
# -managerPort=9066
# -maxCon=1024
#-Dprocessors=4
#-DbackendProcessors=12
#-DprocessorExecutor=4
#-DbackendProcessorExecutor=12
#-DcomplexExecutor=8
#-DwriteToBackendExecutor=4

-DfakeMySQLVersion=5.7.11

# serverBacklog size,default 2048
-DserverBacklog=2048

#-DusePerformanceMode=0
# if need out HA
-DuseOuterHa=true

# connection
#-Dcharset=utf8mb4
-DmaxPacketSize=167772160
-DtxIsolation=2
#-Dautocommit=1

# option
#-DuseCompression=1
-DusingAIO=0

-DuseThreadUsageStat=1
# query time cost statistics
#-DuseCostTimeStat=0
#-DmaxCostStatsize=100
#-DcostSamplePercent=1

# consistency
# check the consistency of table structure between nodes,default not
-DcheckTableConsistency=0
# check period, he default period is 60000 milliseconds
-DcheckTableConsistencyPeriod=60000

# processor check conn
-DprocessorCheckPeriod=1000
-DsqlExecuteTimeout=3000
-DidleTimeout=1800000

#-DbackSocket unit:bytes
#-DbackSocketSoRcvbuf=4194304
#-DbackSocketSoSndbuf=1048576
#-DbackSocketNoDelay=1

# frontSocket
#-DfrontSocketSoRcvbuf=1048576
#-DfrontSocketSoSndbuf=4194304
#-DfrontSocketNoDelay=1

# query memory used for per session,unit is M
-DotherMemSize=4
-DorderMemSize=4
-DjoinMemSize=4

# off Heap unit:bytes
-DbufferPoolChunkSize=32767
-DbufferPoolPageNumber=512
-DbufferPoolPageSize=2097152
#-DmappedFileSize=2097152

# sql statistics
# 1 means use SQL statistics, 0 means not
-DuseSqlstat=1
#-DbufferUsagePercent=80
-DclearBigSQLResultSetMapMs=600000
#-DsqlRecordCount=10
#-DmaxResultSet=524288

# transaction log
# 1 enable record the transaction log, 0 disable ,the unit of transactionRotateSize is M
-DrecordTxn=0
#-DtransactionLogBaseDir=/txlogs
#-DtransactionLogBaseName=server-tx
#-DtransactionRotateSize=16
# XA transaction
# use XA transaction ,if the mysql service crash,the unfinished XA commit/rollback will retry for several times , it is the check period for ,default is 1000 milliseconds
-DxaSessionCheckPeriod=1000
# use XA transaction ,the finished XA log will removed. the default period is 1000 milliseconds
-DxaLogCleanPeriod=1000
# XA Recovery Log path
# -DxaRecoveryLogBaseDir=/tmlogs/
# XA Recovery Log name
#-DxaRecoveryLogBaseName=tmlog
# XA Retry count, retry times in backend, 0 means always retry until success
#-DxaRetryCount=0

#-DviewPersistenceConfBaseDir=/viewPath
#-viewPersistenceConfBaseName=viewJson

# for join tmp results
#-DmergeQueueSize=1024
#-DorderByQueueSize=1024
#-DjoinQueueSize=1024

# true is use JoinStrategy, default false
#-DuseJoinStrategy=true
-DnestLoopConnSize=4
-DnestLoopRowsSize=2000

# if enable the slow query log
```

```
-DenableSlowLog=1
# the slow query log location
#-DslowLogBaseDir=./slowlogs
#-DslowLogBaseName=slow-query
# the max period for flushing the slow query log from memory to disk after last time , unit is second
-DflushSlowLogPeriod=1
# the max records for flushing the slow query log from memory to disk after last time
-DflushSlowLogSize=1000
# the threshold for judging if the query is slow , unit is millisecond
-DsqlSlowTime=100

# used for load data,maxCharsPerColumn means max chars length for per column when load data
#-DmaxCharsPerColumn=65535
# used for load data, because dbale need save to disk if loading file contains large size
#-DmaxRowSizeToFile=10000

#enableFlowControl=false
#flowControlStartThreshold=4096
#flowControlStopThreshold=256
```

## 1.3 user.xml

### 1.3.1 XML

- managerUser()
- shardingUser()
- rwSplitUser()
- blacklist()

### 1.3.2 managerUser()

		/	
name		mysql	
password			
usingDecrypt	true/falsefalse	passwordcrypt.sh 0:{user}:{password} encrypt.sh 0:xxx:123456 fp/nl3XPXrSfWjpQzit5llOrRU1QRXuLTYtATUG0fGW2k5kdXUhKL5zf02hE6nGjdnSWrufVkJPUZpbQ2qX9uQ== password fp/nl3XPXrSfWjpQzit5llOrRU1QRXuLTYtATUG0fGW2k5kdXUhKL5zf02hE6nGjdnSWrufVkJPUZpbQ2qX9uQ== user xxx -u root -p123456	
whiteIPs	ip		whiteIPs
readOnly	true/falsefalse		showselect
maxCon			,0 maxCon

### 1.3.3 shardingUser()

		/	
name		mysql	
password			
usingDecrypt	true/falsefalse	passwordcrypt.sh 0:{user}:{password} encrypt.sh 0:xxx:123456 fp/nl3XPXrSfWjpQzit5llOrRU1QRXuLTYtATUG0fGW2k5kdXUhKL5zf02hE6nGjdnSWrufVkJPUZpbQ2qX9uQ== password fp/nl3XPXrSfWjpQzit5llOrRU1QRXuLTYtATUG0fGW2k5kdXUhKL5zf02hE6nGjdnSWrufVkJPUZpbQ2qX9uQ== user xxx -u root -p123456	
whiteIPs	ip		whiteIPs
readOnly	true/falsefalse		DMLshowselect
tenant			tenant
schemas	schema		schemaschemasharding.xmlschma
maxCon			,0 maxConmaxCon
blacklist	blacklist		blacklist
privileges	table		privileges

### 1.3.3.1 user.privileges.schema

user.privileges shemadml

		/	
name	schema		schema
dml	dml	0000	INSERT UPDATE SELECT DELETE 1- 0- 1111
table			tableschema

### 1.3.3.2 user.privileges.schema.table

		/	
name			key
dml	dml	0000	INSERT UPDATE SELECT DELETE 1- 0- 1111

### 1.3.4 rwSplitUser()

		/	
name		mysql	
password			
usingDecrypt	true/falsefalse	passwordcrypt.sh 0:{user}:{password} encrypt.sh 0:xxx:123456 fp/nl3XPXrSfWjpQzit5llOrRU1QRXuLTYtATUG0fGW2k5kdXUhKL5zf02hE6nGjdnSWrufVkJPUZpbQ2qX9uQ== password fp/nl3XPXrSfWjpQzit5llOrRU1QRXuLTYtATUG0fGW2k5kdXUhKL5zf02hE6nGjdnSWrufVkJPUZpbQ2qX9uQ== user xxx -u root -p123456	
whiteIPs	ip		whiteIPs

tenant		tenant	
dbGroup	dbGroup	db.xmlDbGroup	
maxCon		,0 maxConmaxCon	
blacklist	blacklist	blacklist	

**1.3.5 blacklist()**

		/	
name	blacklist		
property			property

**1.3.5.1 blacklist.property()**

&lt;property name="selectHavingAlwayTrueCheck"&gt;true&lt;/property&gt;

keyvalue

selectHavingAlwayTrueCheck	select having	true	true - false -	1having 2SQL() 3SQL  select * from test having id = id and hujh = hujh /*lxxddfsqldfsdqwesfct*/;	
selectWhereAlwayTrueCheck	select where	true	true - false -	1where 2SQL() 3SQL  select * from test where id = id and hujh = hujh/*lxxddfsqldfsdqwesfct*/;	
doPrivilegedAllow	druid	false	true - false -	druidflagdble	
wrapAllow	isWrapForunwrap	true	true - false -	druidflagdble	
metadataAllow	getmetadata	true	true - false -	druidflagdble	
completeInsertValuesCheck	dble1.0.31	false	true - false -	druidflagdble	
mergeAllow	merge (mysql)	true	true - false -	merge	Dble
conditionLikeTrueAllow	like	true	true - false -	SQLlike '%'	
conditionDoubleConstAllow		false	true - false -	SQL select * from suntest asdf where 1 = 1 and 2 = 1;	
conditionAndAlwayFalseAllow	and false	false	true - false -	whereand :select * from suntest where id = 567 and 1 = 1;	
conditionAndAlwayTrueAllow	and true	false	true - false -	whereand :select * from suntest where id = 567 and 1 = 1;	
selectAllColumnAllow		true	true - false -	sql* (x.*)	
multiStatementAllow	sql	false	true - false -	sql	Dble
constArithmeticAllow		true	true - false -	SQL select * from suntest asdf where id = 2 -1;	
alterTableAllow	alter table	true	true - false -	alter tableSQL SQLalter table	
commitAllow	commit	true	true - false -	commitSQL commit	
createTableAllow	create table	true	true - false -	create tableSQL SQLcreate table	
deleteAllow	delete	true	true - false -	deleteSQLdelete	
dropTableAllow	drop table	true	true - false -	drop tableSQL SQLdrop table	
insertAllow	insert	true	true - false -	insertSQLinsert	
intersectAllow	intersect	true	true - false -	intersectSQL intersect	
lockTableAllow	lock tables	true	true - false -	lock tablesSQL lock	
minusAllow	minus	true	true - false -	minusSQL minus	
callAllow	call	true	true - false -	querySQL SQLCALL	
selectIntoOutfileAllow	SELECT ... INTO OUTFILE	false	true - false -	querySQLSQL SELECT ... INTO OUTFILE	Dble

selectIntoAllow	select into	true	true - false -	querySQL SQLselect into	Dble
selelctAllow	select	true	true - false -	querySQL SQLselect	
renameTableAllow	rename table	true	true - false -	rename tableSQL SQLrename table	
replaceAllow	replace	true	true - false -	replaceSQL replace	
rollbackAllow	rollback	true	true - false -	rollbackSQL rollback	Dble
setAllow	set	true	true - false -	setSQL set	
describeAllow	describe	true	true - false -	SQLSQL describe	
limitZeroAllow	limit 0	false	true - false -	SQLSQL limit 0	
showAllow	show	true	true - false -	SQLSQL show	
hintAllow	sql hint	true	true - false -	sqlSQL SQLHint	
commentAllow	SQL	true	true - false -	SQLSQL	
mustParameterized		false	true - false -	SQLSQL name = 'sdfasdf' ,id = 1	
conditionOpXorAllow	SQLXOR	false	true - false -	SQLSQL	
conditionOpBitwseAllow	"&" "~" "  " "^^" "^"	true	true - false -	SQLSQL	
startTransactionAllow	START TRANSACTION	true	true - false -	START TRANSACTIONSQL START TRANSACTION :begin druid	
truncateAllow	truncate	true	true - false -	truncateSQL truncate	
updateAllow	update	true	true - false -	updateSQLupdate	
useAllow	use	true	true - false -	useSQLuse	
blockAllow		true	true - false -	SQLSQLSQL BEGIN; select * from suntest;END;//	
deleteWhereNoneCheck	deletewhere	false	true - false -	delete delete from suntest;	
updateWhereNoneCheck	updatewhere	false	true - false -	update update suntest set name = '33';	
deleteWhereAlwayTrueCheck	delete	true	true - false -	delete.sql where delete from suntest where id = id and name= name /*sdfaasd*/;	
updateWhereAlayTrueCheck	delete	true	true - false -	deletesql where update suntest set name = '33' where id = id,name = name /*sdfsdf*/;	
selectIntersectCheck	intersect check	true	true - false -	except left sqlfromright from :select * from sbtest1 where name = 'ff' INTERSECT; select * from dual;	
selectExceptCheck	except check	true	true - false -	except left sqlfromright from select * from sbtest1 where name = 'ff' except; select * from dual;	
selectMinusCheck	MINUS check	true	true - false -	MINUS left sqlfromright from select * from sbtest1 where namec = 'fff' minus select * from dual;	
selectUnionCheck	union check	true	true - false -	unoin select * from sbtest1 unoin select * from suntest;	
caseConditionConstAllow		false	true - false -	SQL :delete from suntest where id = 123 and 'name' = (select case 'fname' whe dsome' else 'good' end from xtest ) /*sdfaasd*/;	
strictSyntaxCheck		true	true - false -	Druid SQL Parser SQL	
schemaCheck	schema	true	true - false -	druidble	
tableCheck	table	true	true - false -	druidble	
functionCheck	function	true	true - false -	druidble	

objectCheck	object	true	true - false -	druidble	
variantCheck		true	true - false -	druidble	

### 1.3.6 tenant

dble

1. ::

```
mysql -u: -p -h
```

```
DriverManager.getConnection("jdbc:mysql://127.0.0.1:8066", "root2:tenant1", "123456");
```

2. JDBC-connectionAttributes connectionAttributes tenant

```
DriverManager.getConnection("jdbc:mysql://127.0.0.1:8066?connectionAttributes=tenant:tenant1", "root2", "123456");
```

1/21

### 1.3.7 whiteIPsIP

IP

ip192.168.1.2,192.168.2.22  
 IP192.168.1.10-192.168.1.100  
 192.168.1.%  
 IP/CIDR192.168.1.1/20

IPv4/IPv6

- 127.0.0.10:0:0:0:0:0:1
- IPv6IPv4

### 1.3.8

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE dble:use SYSTEM "user.dtd">
<dbl:use xmlns:dbl="http://dble.cloud/">
  <managerUser name="man1" password="654321" whiteIPs="127.0.0.1,0:0:0:0:0:1" readOnly="false"/>
  <managerUser name="user" usingDecrypt="true" readOnly="true" password="AqEKFEuIFAX6g2TJQnp4cJ2r7Yc0Z4/KBsZqKhT8qSz18Aj91e8lx049BKQElc60FfW4c38pCYa8QGFTub7pnw==" />
  <shardingUser name="root" password="123456" schemas="testdb" readOnly="false" blacklist="blacklist1" maxCon="20"/>
  <shardingUser name="root2" password="123456" schemas="testdb,testdb2" maxCon="20" tenant="tenant1">
    <privileges check="true">
      <schema name="testdb" dml="0110">
        <table name="tb01" dml="0000"/>
        <table name="tb02" dml="1111"/>
      </schema>
    </privileges>
  </shardingUser>
  <!--rwSplitUser not work for now-->
  <rwSplitUser name="rwsu1" password="123456" dbGroup="dbGroup1" blacklist="blacklist1" maxCon="20"/>
  <blacklist name="blacklist1">
    <property name="selectAllow">true</property>
  </blacklist>
</dbl:use>
```

## 1.4 db.xml

db.xml,

### 1.4.1 dbGroup

- dbGroup

	&	/	
name			dbGroup
rwSplitMode		0/1/2	0, 1 2
delayThreshold		-1	1:delayThresholdshow slave status 2:delayThreshold=-1
disableHA		true/falsefalse	
heartbeat	,		mysql. : 1.select 1 2. select @@read_only 3.show slave status,Seconds_Behind_Mastermysql
dbInstance			

- heartbeat

	&	/	
timeout	heartbeat,	0	dble heartbeatPeriodMillis dbInstance .timeout .heartbeatPeriodMillis)22s4s4stimeout( 2. dbletimeout
errorRetryCount	heartbeat,	10	/errorRetryCount 1. errorOK() 2. okerror 3.

- dbInstance

	&	/	
name			
url	ip:port		IPPORT
user			
password			
usingDecrypt	password	0/1,0	1passworddecrypt.sh 1:{name}:{user}:{password}
minCon			
maxCon			
readWeight	0	0	
primary	true	false	
disabled		false	
property			

- property

```

;
<?xml version="1.0"?>
<!DOCTYPE dble:db SYSTEM "db.dtd">
<db:db xmlns:dble="http://dble.cloud/">
  <dbGroup name="dbGroup1" rwSplitMode="2" delayThreshold="100">
    <heartbeat>show slave status</heartbeat>
    <dbInstance name="instanceM1" url="ip1:3306" user="your_user" password="your_psw" maxCon="1000" minCon="10"
      primary="true" readWeight="1" id="xx1">
      <property name="lifo">true</property>
    </dbInstance>
    <dbInstance name="instanceS1" url="ip2:3306" user="your_user" password="your_psw" maxCon="1000" minCon="10" readWeight="2" disabled="true">
      <property name="testOnCreate">false</property>
    </dbInstance>
  </dbGroup>

  <dbGroup name="dbGroup2" rwSplitMode="0" delayThreshold="100" disableHA="true">
    <heartbeat errorRetryCount="1" timeout="10">show slave status</heartbeat>
    <dbInstance name="instanceM2" url="ip4:3306" user="your_user" password="your_psw" maxCon="1000" minCon="10"
      primary="true">
      <property name="testOnCreate">true</property>
    </dbInstance>
    <!-- can have multi read instances -->
    <dbInstance name="instances2" url="ip5:3306" user="your_user" password="your_psw" maxCon="1000" minCon="10" usingDecrypt="true">
      <property name="testOnCreate">true</property>
    </dbInstance>
  </dbGroup>
</db:db>

```

### 1.4.2 MySQL

SELECT	
INSERT	
UPDATE	
DELETE	
FILE	load data
CREATE	0
DROP	
ALTER	
LOCK TABLES	lock tables
ALTER ROUTINE	(hint)/
CREATE ROUTINE	(hint)
EXECUTE	(hint)
INDEX	/
SUPER	KILL
SHOW DATABASES	GUIINFORMATION_SCHEMA SCHEMATA
PROCESS	show processlist
REPLICATION CLIENT	: 1., 2.show slave status 3.show @@binlog_status
REFERENCES	0



## 1.5 sharding.xml

### 1.5.1 XML

- schema (schema)
- shardingNode ()
- function ()

### 1.5.2 schema

#### 1.5.3 schema

- schema

		/	
name	schema		schema
shardingNode			xmtable; 1.tableschema2.schema,table single node table
sqlMaxLimit		-1	SQL 1 (order/group/ 2 schemasqlMaxLimit
shardingTable	shardingTable		
globalTable	globalTable		
singleTable	singleTable		

- shardingTable

		/	
name			;
shardingNode			xxx\$0-n1 xxxn0, ..., xxxnm, ..., xxxn1 dn\$1-6 dn1,dn2, dn3,dn4,dn5,dn6
shardingColumn			
function			function
incrementColumn			
sqlMaxLimit		schemasqlMaxLimit	schemas schema
sqlRequiredSharding	sql	false	truesql
childTable	childTable		ER

- childTable

		/	
name			;
joinColumn	joinjoin		
parentColumn	joinjoin		/
incrementColumn			
sqlMaxLimit		schemasqlMaxLimit	schemas schema
childTable	childTable		ER

- globalTable

		/	
name			;
shardingNode		xxx\$0-n1 xxxn0, ..., xxxnm, ..., xxxn1 dn\$1-6 dn1,dn2, dn3,dn4,dn5,dn6	
sqlMaxLimit	schemasqlMaxLimit	schemas schema	
checkClass		dbleCHECKSUM COUNT	
cron	0 0 0 * * ?	quartz <a href="http://www.quartz-scheduler.org/api/2.4.0-SNAPSHOT/org/quartz/CronScheduleBuilder.html">http://www.quartz-scheduler.org/api/2.4.0-SNAPSHOT/org/quartz/CronScheduleBuilder.html</a>	

- singleTable

		/	
name			;
shardingNode			shardingNode
sqlMaxLimit		schemasqlMaxLimit	schemas schema

### 1.5.4 shardingNode

- shardingNode

		&	
name	"dn,dn\$0-5"		,,,'X()05,"X\$0-5",\$namedatabasedbName
database	shardingNodemysqldb,"db,db\$0-5"		,,,'X()05,"X\$0-5",\$
dbGroup	shardingNodedb.xmldbGroup,,,"dh,dh\$0-5"		Host,'X()05,"X\$0-5",\$

```
<shardingNode name="dn1" dbGroup="localhost1" database="db1" />
name, dbGroup, database      xxx$0-n1, xxx, xxxn0..., xxxnm ... ,xxxn1, xxx,
n0 < nm < n1
```

```
<shardingNode name="dn1$0-19" dbGroup="localhost1$0-9" database="db1$0-1" />
```

```
:<shardingNode name="dn10" dbGroup="localhost10" database="db10" />
<shardingNode name="dn11" dbGroup="localhost10" database="db11" />
<shardingNode name="dn12" dbGroup="localhost11" database="db10" />
<shardingNode name="dn13" dbGroup="localhost11" database="db11" />
...
<shardingNode name="dn19" dbGroup="localhost19" database="db11" />
```

```
shardingNode(name)dbGroupdatabase,,name20,dbGroup10, database2;
```

```
<shardingNode name="dn,dn$0-19,dnx" dbGroup="localhost,localhost1$0-9" database="db1$0-1" />
```

```
,name22,dbGroup11, database2
```

shardingNodedatabaseGroup(shardingNode)

### 1.5.5 function

name: class property name

- function

name		
class		Enum,NumberRange,Hash,StringHash,Date,PatternRange,jump StringHash
property	function	

```
<function name="rang-long" class="com.actiontech.dble.route.function.AutoPartitionByLong">
    <property name="mapFile">auto-sharding-long.txt</property>
    ...
</function>
```

: hash, stringhash, enum, numberrange, patternrange, datejump stringhash.

#### 1.5.5.1.hash

function class="hash"" com.actiontech.dble.route.function.PartitionByLong"

```
<function name="hashLong" class="hash">
    <property name="partitionCount">C1[, C2, ...Cn]</property>
    <property name="partitionLength">L1[, L2, ...Ln]</property>
</function>
```

partitionCount: C1 [+C2 + ... + Cn].

partitionLength: [0, L1], [L1, 2L1], ..., [(C1-1)L1, C1L1], [C1L1, C1L1+L2], [C1L1+L2, C1L1+2L2], ...

F1

```
<property name="partitionCount">2,</property>
<property name="partitionLength">100,50</property>
```

[0 , 100) [100, 200) [200, 250) [250, 300) [300, 350)

,F2:

```
<property name="partitionCount">2</property>
<property name="partitionLength">1000</property>
```

[0 , 1000) [1000, 2000)

MC1 L1 + ... + Cn Ln. F1 M 350F2M2000 keyM

value = key mod M value F1,key =805 ,value = 105,51(0)

N C1 [+C2 + ... + Cn]. F1 N5F2N2

1. M 28802880:2, 3, 4, 5, 6, 8, 9, 10, 12, 15, 16, 18, 20, 24, 30, 32, 36, 40, 45, 48, 60, 64, 72, 80, 90, 96, 120, 144, 160, 180, 192, 240, 288, 320, 360, 480, 576, 720, 960, 14402880,
2. NshardingNodeshardingNodeshardingNode="dn1,dn2,dn3,dn4" N4
3. CnLn
- 4.
5. partitionLength1hashMNpartitionCount
6. NULL0

#### 1.5.5.2.stringhash

class="stringhash"" com.actiontech.dble.route.function.PartitionByString"

```
<function name="hashString" class="stringhash">
    <property name="partitionCount">C1[, C2, ...Cn]</property>
    <property name="partitionLength">L1[, L2, ...Ln]</property>
    <property name="hashSlice">l:r</property>
</function>
```

partitionCountpartitionLength hash hashSlicehashkey0

1.

n	n>=0		(0,n)
n	n<0		(n,0)
r			(0,r)
l:			(l,0)
:			(0:0)
lr			(l, r)

2.

a.l

l	l>=0	l		l
l	l<0	l=l+length	l<0	l=0

b.r

r	r>0	r	r>length	r=length
r	r<=0	r=r+length		r

length.

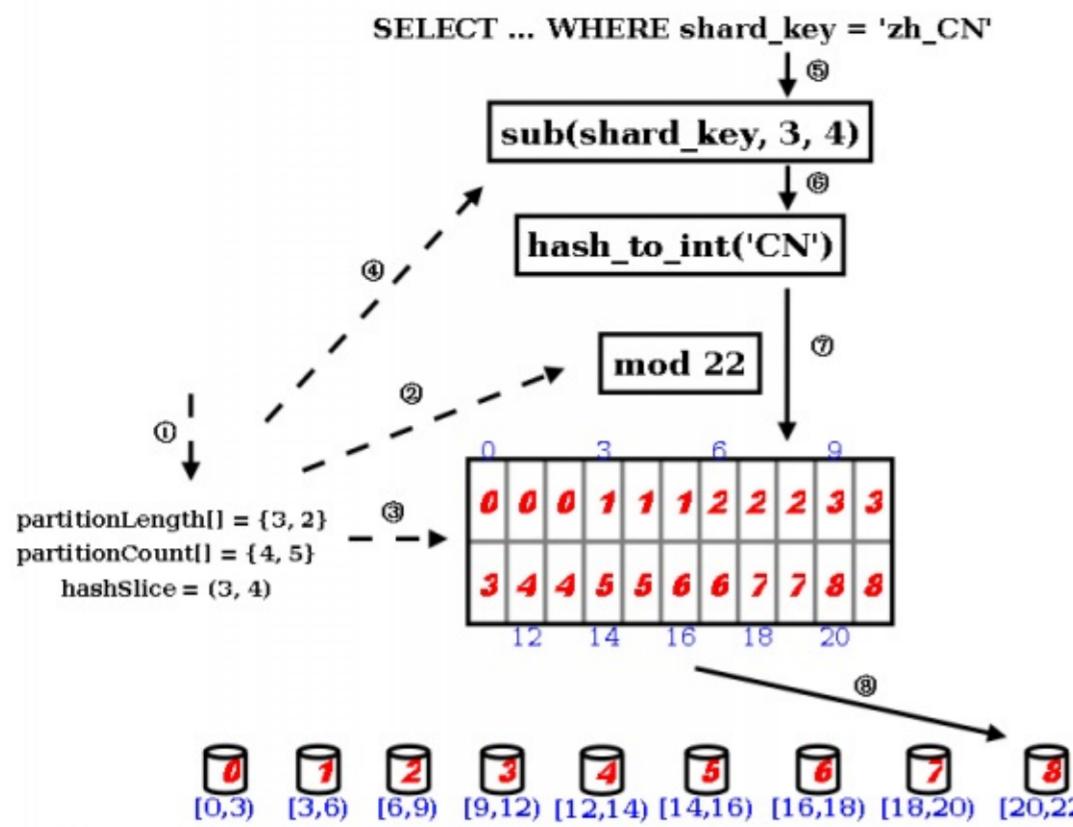
3.

			hash
l<r		[l, r)	hash
l>=r			0

hash hashhashhash

1. hash(3)
- 2.

stringhash unicode hash



- `partitionLength[] partitionCount[] hashSlice`
- DBLE
- `partitionCount[]`
- `hashSlice 4 5 0 3 4 4 5 " ->"`
- DBLE WHERE 4 5
- 0 31 unicode " ->"
- 
- 

### 1.5.5.3.enum

class="enum"“com.actiontech.dble.route.function.PartitionByFileMap”

```
<function name="enum" class="enum">
  <property name="mapFile">partition.txt</property>
  <property name="defaultNode">0</property>
  <property name="type">0</property>
</function>
```

mapFile: defaultNode-1 typekey0

a. type0

```
#comment //comment this line will be skiped
int1=node0
int2=node1
...
```

b. type0

```
#comment //comment this line will be skiped
string1=node0
string2=node1
...
```

defaultNodedefaultNode

1. “=”.
2. nodeX
- 3.
- 4.
5. NULLdefaultNodedefaultNode;mysqlnot null, "Sharding column can't be null when the table in MySQL column is not null"

### 1.5.5.4.numberrange

class="numberrange"“com.actiontech.dble.route.function.AutoPartitionByLong”

```
<function name="rangeLong" class="numberrange">
  <property name="mapFile">partition.txt</property>
  <property name="defaultNode">0</property>
</function>
```

mapFile: defaultNode-1

#comment //comment this line will be skiped start1-end1=node1 start2-end2=node2 ...

[start1, end1], [start2, end2], ... 1.defaultNodedefaultNode2.defaultNode

1. “=”
2. nodeX
- 3.
- 4.

5. NULLdefaultNodedefaultNode;mysqlnot null, "Sharding column can't be null when the table in MySQL column is not null"

### 1.5.5.5.patternrange

class="patternrange"“com.actiontech.dble.route.function.PartitionBy Pattern”

```
<function name="pattern" class="patternrange">
  <property name="mapFile">partition.txt</property>
  <property name="patternValue">1024</property>
  <property name="defaultNode">0</property>
</function>
```

mapFile: patternValue: 1024 defaultNode-1

#comment //comment this line will be skiped start1-end1=node1 start1-end2=node2 ...

numberrangepatternValuenumberrange

1. “=”
2. nodeX
- 3.
4. defaultNode defaultNode

5. NULLdefaultNodedefaultNode;mysqlnot null, "Sharding column can't be null when the table in MySQL column is not null"

### 1.5.5.6.date

class"date""com.actiontech.dble.route.function.PartitionByDate"

```
<function name="partbydate" class="date">
  <property name="dateFormat">yyyy-MM-dd</property>
  <property name="sBeginDate">2015-01-01</property>
  [<property name="sEndDate">2015-01-31</property>]
  <property name="sPartitionDay">10</property>
  <property name="defaultNode">0</property>
</function>
```

dateFormat: sBeginDate sEndDate("") sPartitionDay defaultNode-1

1sEndDatesEndDate"" sBeginDate sPartitionDay sBeginDatedefaultNodedefaultNode, 2sEndDate"" sBeginDate sPartitionDaysEndDate N sEndDate1sEndDateindex=((key - sBeginDate)/sPartitionDay)%N, defaultNode;

1. dateFormat

2. sPartitionDay86400000

3. 2 (sEndDate - sBeginDate)sPartitionDay0

4. NULLdefaultNodedefaultNode;mysqlnot null, "Sharding column can't be null when the table in MySQL column is not null"

### 1.5.5.7.

class"jumpstringhash""com.actiontech.dble.route.function.PartitionByJumpConsistentHash"

```
<function name="jumphash"
  class="JumpStringHash">
  <property name="partitionCount">2</property>
  <property name="hashSlice">0:2</property>
</function>
```

partitionCount: hashSlice: GoogleA Fast, Minimal Memory, Consistent Hash Algorithmhash1/n

1. NULL0;mysqlnot null, "Sharding column can't be null when the table in MySQL column is not null"

### 1.5.6

```
<?xml version="1.0"?>
<!!--
- Copyright (C) 2016-2020 ActionTech.
- License: http://www.gnu.org/licenses/gpl.html GPL version 2 or higher.
-->

<!DOCTYPE dble:sharding SYSTEM "sharding.dtd">
<dblesharding xmlns:dblesharding="http://dblesharding.com/" version="4.0">

<schema name="testdb" sqlMaxLimit="100">
  <shardingTable name="tb_enum_sharding" shardingNode="dn1,dn2" sqlMaxLimit="200" function="func_enum" shardingColumn="code"/>
  <shardingTable name="tb_range_sharding" shardingNode="dn1,dn2,dn3" function="func_range" shardingColumn="id"/>
  <!-- tables-->
  <shardingTable name="tb_hash_sharding" shardingNode="dn1,dn2" function="func_common_hash" shardingColumn="id"/>
  <shardingTable name="tb_hash_sharding_eri" shardingNode="dn1,dn2" function="func_common_hash" shardingColumn="id"/>
  <shardingTable name="tb_hash_sharding_eri2" shardingNode="dn1,dn2" function="func_common_hash" shardingColumn="id2"/>
  <shardingTable name="tb_hash_sharding_eri3" shardingNode="dn1,dn2" function="func_common_hash" shardingColumn="id" incrementColumn="id2"/>

  <shardingTable name="tb_uneven_hash" shardingNode="dn1,dn2,dn3" function="func_uneven_hash" shardingColumn="id"/>

  <shardingTable name="tb_mod" shardingNode="dn1,dn2,dn3,dn4" function="func_mod" shardingColumn="id" sqlRequiredSharding="true"/>

  <shardingTable name="tb_jump_hash" shardingNode="dn1,dn2" function="func_jumpHash" shardingColumn="code"/>

  <shardingTable name="tb_hash_string" shardingNode="dn1,dn2,dn3,dn4" function="func_hashString" shardingColumn="code"/>

  <shardingTable name="tb_date" shardingNode="dn1,dn2,dn3,dn4" function="func_date" shardingColumn="create_date"/>

  <shardingTable name="tb_pattern" shardingNode="dn1,dn2" function="func_pattern" shardingColumn="id"/>
  <!-- global tables-->
  <globalTable name="tb_global1" shardingNode="dn1,dn2" sqlMaxLimit="103" />
  <globalTable name="tb_global2" shardingNode="dn1,dn2,dn3,dn4" cron="0 0 0 * * ?" checkClass="CHECKSUM"/>
  <!-- single node table-->
  <singleTable name="tb_single" shardingNode="dn6" sqlMaxLimit="105"/>
  <!-- tables-->
  <shardingTable name="tb_parent" shardingNode="dn1,dn2" function="func_common_hash" shardingColumn="id">
    <childTable name="tb_child1" joinColumn="child1_id" parentColumn="id" sqlMaxLimit="201">
      <childTable name="tb_grandson1" joinColumn="grandson1_id" parentColumn="child1_id"/>
      <childTable name="tb_grandson2" joinColumn="grandson2_id" parentColumn="child1_id2"/>
    </childTable>
    <childTable name="tb_child2" joinColumn="child2_id" parentColumn="id"/>
    <childTable name="tb_child3" joinColumn="child3_id" parentColumn="id2"/>
  </shardingTable>
</schema>
<!-- sharding testdb2 route to database named dn5 in localhost -->
<schema name="testdb2" shardingNode="dn5">
<shardingNode name="dn2" dbGroup="dbGroup1" database="db_1"/>
<shardingNode name="dn2" dbGroup="dbGroup2" database="db_2"/>
<shardingNode name="dn3" dbGroup="dbGroup1" database="db_3"/>
<shardingNode name="dn4" dbGroup="dbGroup2" database="db_4"/>
<shardingNode name="dn5" dbGroup="dbGroup1" database="db_5"/>
<shardingNode name="dn6" dbGroup="dbGroup2" database="db_6"/>
<!-- enum partition -->
<function name="func_enum" class="Enum">
  <property name="mapFile">partition-enum.txt</property>
  <property name="defaultNode">0</property><!-- the default is -1, means unexpected value will report error-->
  <property name="type">0</property><!-- 0 means key is a number, 1 means key is a string-->
</function>
<!-- number range partition -->
<function name="func_range" class="NumberRange">
  <property name="mapFile">partition-number-range.txt</property>
  <property name="defaultNode">0</property><!-- the default is -1, means unexpected value will report error-->
</function>
<!-- Hash partition, when partitionLength=1, it is a mod partition, MAX(sum(count*length[i])) must not more than 2880-->
<function name="func_common_hash" class="Hash">
  <property name="partitionCount">2</property>
  <property name="partitionLength">512</property>
</function>
<!-- Hash partition, when partitionLength=1, it is a mod partition, MAX(sum(count*length[i])) must not more than 2880-->
<function name="func_uneven_hash" class="Hash">
  <property name="partitionCount">2,1</property>
  <property name="partitionLength">256,512</property>
</function>
<!-- eg: mod 4 -->
<function name="func_mod" class="Hash">
  <property name="partitionCount">4</property>
  <property name="partitionLength">1</property>
</function>
<!-- jumpStringHash partition for string-->
<function name="func_jumpHash" class="JumpStringHash">
  <property name="partitionCount">2</property>
  <property name="hashSlice">0:2</property>
</function>
<!-- Hash partition for string-->
<function name="func_hashString" class="StringHash">
  <property name="partitionCount">4</property>
  <property name="partitionLength">256</property>
  <property name="hashSlice">0:2</property>
  <!--<property name="hashSlice">-4:0</property> -->
</function>
<!-- date partition 4 case:
1.set sEndDate and defaultNode: input <sBeginDate ,router to defaultNode; input>sEndDate ,mod the period
2.set sEndDate, but no defaultNode:input <sBeginDate report error; input>sEndDate ,mod the period
3.set defaultNode without sEndDate: input <sBeginDate router to defaultNode; input>sBeginDate + (node size)*sPartitionDay-1 will report error(expected is defaultNode, but can't control now)

```

```
4.sEndDate and defaultNode are all not set: input <sBeginDate report error;input>sBeginDate + (node size)*sPartitionDay-1 will report error
-->
<function name="func_date" class="Date">
  <property name="dateFormat">yyyy-MM-dd</property>
  <property name="sBeginDate">2015-01-01</property>
  <property name="sEndDate">2015-01-31</property> <!--if not set sEndDate,then in fact ,the sEndDate = sBeginDate+ (node size)*sPartitionDay-1 -->
  <property name="sPartitionDay">10</property>
  <property name="defaultNode">0</property><!--the default is -1-->
</function>
<!-- pattern partition : mapFile must contains all value of 0-patternValue-1,key and value must be Continuous increase-->
<function name="func_pattern" class="PatternRange">
  <property name="mapFile">partition-pattern.txt</property>
  <property name="patternValue">1024</property>
  <property name="defaultNode">0</property><!--contains string which is not number,router to default node-->
</function>
</db:sharding>
```

## 1.6 log4j2.xml

### 1.6.1

Dblejavalog4j2.xml

#### 1.6.1.1

DefaultRolloverStrategy RollingRandomAccessFileRolloverStrategy

```
<DefaultRolloverStrategy max="100">
    <Delete basePath="logs" maxDepth="2">
        <IfFileName glob="*/dble-*.*.log.gz">
            <IfLastModified age="2d">
                <IfAny>
                    <IfAccumulatedFileSize exceeds="1 GB" />
                    <IfAccumulatedFileCount exceeds="10" />
                </IfAny>
            </IfLastModified>
        </IfFileName>
    </Delete>
</DefaultRolloverStrategy>
```

```
:  
basePath  
maxDepth basePath maxDepth.../logs/2018-01-01 .../log/2018-01-02  
glob  
age  
IfAccumulatedFileSize  
IfAccumulatedFileCount  
logSize "/dble-.log.gz" 21 GB 10
```

### 1.6.2

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration status="WARN" packages="com.actiontech.dble.log">
    <Appenders>
        <Console name="Console" target="SYSTEM_OUT">
            <PatternLayout pattern="%d [%p] [%t] %m %throwable{full} (%C:%F:%L) %n"/>
        </Console>
    </Appenders>
    <RollingRandomAccessfile name="RollingFile" fileName="logs/dble.log"
        filePattern="logs/${date:yyyy-MM}/dble-%d{MM-dd}-%i.log.gz">
        <PatternLayout>
            <Pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} %sp [%t] (%l) - %m%n</Pattern>
        </PatternLayout>
        <Policies>
            <OnStartupTriggeringPolicy/>
            <SizeBasedTriggeringPolicy size="250 MB"/>
            <TimeBasedTriggeringPolicy/>
        </Policies>
        <DefaultRolloverStrategy max="100">
            <Delete basePath="logs" maxDepth="2">
                <IfFileName glob="*/dble-*.*.log.gz">
                    <IfLastModified age="2d">
                        <IfAny>
                            <IfAccumulatedFileSize exceeds="1 GB" />
                            <IfAccumulatedFileCount exceeds="10" />
                        </IfAny>
                    </IfLastModified>
                </IfFileName>
            </Delete>
        </DefaultRolloverStrategy>
    </RollingRandomAccessfile>
    </Appenders>
    <Loggers>
        <asyncRoot level="debug" includeLocation="true">
            <AppenderRef ref="Console"/>
            <AppenderRef ref="RollingFile"/>
        </asyncRoot>
    </Loggers>
</Configuration>
```

**1.7**

```
dble
cluster.cnf
```

```
sequenceHandlerType=n
```

```
sequenceHandlerType
```

- 1:MySQL offset-step
- 2:(Snowflake)
- 3:(Snowflake)
- 4:offset-step

```
sharding.xml
```

```
: incrementColumn
```

```
//incrementColumn=pidpid
```

```
<shardingTable name="table1" shardingNode="dn1,dn2" function="func_common_hash" shardingColumn="id" incrementColumn="pid"/>
```

```
insert into table1(name) values('test');
```

```
insert into table1 set name = 'test';
```

**2.2****MYSQL**

```
MySQLdbledbledbleID
```

```
mysql
```

```
table1aid,bid,cid,did biddble
```

```
insert into table1 values(1,2,3)
```

```
sql
```

```
insert into table1 set aid = 1,cid = 2,did = 3
```

```
dbleupdatereplace)
```

## 1.7.1 MySQL-offset-step

### 1.7.1.1 MySQL-offset-step

mysqlsequence\_db\_conf.properties:

#this is comment

'schema1`.`table1'=node1

'schema1`.`table2'=node1

'schema2`.`table1'=node2

...

**schemaX**dbledble

**tableX** dbledble

**nodeX**

### 1.7.1.2

mysql nodeX schemaX`.**tableX**)

**nodeX**

a. mysql**nodeX**

mysql ...

b. **nodeX**db(1.2 schemal.xml)

use db;

c. dbsql.sqldbseq.sql1.7.2.3

source .../dbseq.sql;

d.

INSERT INTO DBLE\_SEQUENCE VALUES ('**schemaX`.**tableX**', 1, 1);**

...

### 1.7.1.3 dbseq.sql

dbseq.sql

```

DROP TABLE IF EXISTS DBLE_SEQUENCE;
CREATE TABLE DBLE_SEQUENCE ( name VARCHAR(64) NOT NULL, current_value BIGINT(20) NOT NULL, increment INT NOT NULL DEFAULT 1, PRIMARY KEY (name) ) ENGINE=InnoDB;

-- -----
-- Function structure for `dble_seq_currval`
-- -----
DROP FUNCTION IF EXISTS `dble_seq_currval`;
DELIMITER ;;
CREATE FUNCTION `dble_seq_currval`(seq_name VARCHAR(64)) RETURNS varchar(64) CHARSET latin1
DETERMINISTIC
BEGIN
DECLARE retval VARCHAR(64);
SET retval="-1,0";
SELECT concat(CAST(current_value AS CHAR),",",CAST(increment AS CHAR)) INTO retval FROM DBLE_SEQUENCE WHERE name = seq_name;
RETURN retval;
END
;;
DELIMITER ;

-- -----
-- Function structure for `dble_seq_nextval`
-- -----
DROP FUNCTION IF EXISTS `dble_seq_nextval`;
DELIMITER ;;
CREATE FUNCTION `dble_seq_nextval`(seq_name VARCHAR(64)) RETURNS varchar(64) CHARSET latin1
DETERMINISTIC
BEGIN
DECLARE retval VARCHAR(64);
DECLARE val BIGINT;
DECLARE inc INT;
DECLARE seq_lock INT;
set val = -1;
set inc = 0;
SET seq_lock = -1;
SELECT GET_LOCK(seq_name, 15) into seq_lock;
if seq_lock = 1 then
SELECT current_value + increment, increment INTO val, inc FROM DBLE_SEQUENCE WHERE name = seq_name for update;
if val != -1 then
UPDATE DBLE_SEQUENCE SET current_value = val WHERE name = seq_name;
end if;
SELECT RELEASE_LOCK(seq_name) into seq_lock;
end if;
SELECT concat((val - inc + 1) as CHAR),",",CAST(inc as CHAR)) INTO retval;
RETURN retval;
END
;;
DELIMITER ;

-- -----
-- Function structure for `dble_seq_setvals`
-- -----
DROP FUNCTION IF EXISTS `dble_seq_setvals`;
DELIMITER ;;
CREATE FUNCTION `dble_seq_setvals`(seq_name VARCHAR(64), count INT) RETURNS VARCHAR(64) CHARSET latin1
DETERMINISTIC
BEGIN
DECLARE retval VARCHAR(64);
DECLARE val BIGINT;
DECLARE seq_lock INT;
SET val = -1;
SET seq_lock = -1;
SELECT GET_LOCK(seq_name, 15) into seq_lock;
if seq_lock = 1 then
SELECT current_value + count INTO val FROM DBLE_SEQUENCE WHERE name = seq_name for update;
IF val != -1 THEN
UPDATE DBLE_SEQUENCE SET current_value = val WHERE name = seq_name;
END IF;
SELECT RELEASE_LOCK(seq_name) into seq_lock;
end if;
SELECT CONCAT(CAST((val - count + 1) as CHAR),",",CAST(count as CHAR)) INTO retval;
RETURN retval;
END
;;
DELIMITER ;

-- -----
-- Function structure for `dble_seq_setval`
-- -----
DROP FUNCTION IF EXISTS `dble_seq_setval`;
DELIMITER ;;
CREATE FUNCTION `dble_seq_setval`(seq_name VARCHAR(64), value BIGINT) RETURNS varchar(64) CHARSET latin1
DETERMINISTIC
BEGIN
DECLARE retval VARCHAR(64);
DECLARE inc INT;
SET inc = 0;
SELECT increment INTO inc FROM DBLE_SEQUENCE WHERE name = seq_name;
UPDATE DBLE_SEQUENCE SET current_value = value WHERE name = seq_name;
SELECT concat(CAST(value as CHAR),",",CAST(inc as CHAR)) INTO retval;
RETURN retval;
END
;;
DELIMITER ;

```



**1.7.2**

```
bootstrap.cnf cluster.cnf
bootstrap.cnfinstanceIdinstance id [0,1023]
cluster.cnfsequenceStartTime YYYY-MM-dd HH:mm:ss 2010-10-04 09:42:54
bootstrap.cnfinstanceIdddbleble
bigint63
```

### 1.7.3

```
bootstrap.cnf cluster.cnf
bootstrap.cnfinstanceId instance id [0,511]  cluster.cnfsequenceInstanceByZktrueidzk
cluster.cnfsequenceStartTime YYYY-MM-dd HH:mm:ss 2010-10-04 09:42:54
```

1. cluster.cnfsequenceInstanceByZktruezookeeper( 1.8 myid.properties)
2. bootstrap.cnfinstanceIdble
3. bigint63

#### 1.7.4 offset-step

```
offset-stepsequence_conf.properties
```

```
# this is comment  
`schema1`.`table1`.MINID=1001  
`schema1`.`table1`.MAXID=2000  
`schema1`.`table1`.CURID=1000  
  
`schema2`.`table2`.MINID=1001  
`schema2`.`table2`.MAXID=20000  
`schema2`.`table2`.CURID=1000  
  
schemaXmysqlable  
tableXmysqlable
```

```
1. zkMINID MAXID CURID
```

```
2. MINIDCURID1 MINID+1
```

```
3. MAXID - MINID + 1zookeeper
```

```
4. dblezookeeper 1.8 my.id.properties
```

## 1.8 cache

**1.8.1 cache****1.8.1.1 dblecache**

dblecache

- SQLRouteCacheSQL shema\_user\_SQL -> RouteResult
- ER\_SQL2PARENTIDSQLERjoinKey(parentKey)schema:select \* from where parentKey = (value of joinKey) -> shardingNode

**1.8.1.2 dblecache**

dblecache

- ehcache, ehcachecache
- leveldb leveldbcache
- mapdb MapDBcache
- rocksdbRocksDBcache

**1.8.1.3 dblecache**

dblecachecacheservice.properties

factory.cache\_type=cache\_type

keyvalue

**A.SQL**

pool.SQLRouteCache=type,max\_size,expire\_seconds

**B.ER**

pool.ER\_SQL2PARENTID=type,max\_size,expire\_seconds

**1.8.1.4 cache**

a. #

b. factory.cache\_type=cache\_typecache cacheehcacheleveldbmapdb rocksdbcache cache

,

factory.encache=ehcache  
pool.SQLRouteCache=encache,10000,1800  
pool.ER\_SQL2PARENTID=encache,1000,1800

typeehcache

factory.encache=ehcache  
factory.leveldb=leveldb  
pool.SQLRouteCache=encache,10000,1800  
pool.ER\_SQL2PARENTID=leveldb,1000,1800

typeencacheleveldb

c. pool.SQLRouteCache=type,max\_size,expire\_secondspool.ER\_SQL2PARENTID=type,max\_size,expire\_secondsSQLRouteCacheER\_SQL2PARENTID type max\_size expire\_seconds

d. default

**1.8.1.5**

- RocksDB cache dble rocksdb dble

## 1.8.2 ehcache

ehcachecacheservice.propertiesehcache

### 1.8.2.1 ehcache

dble2.6.11.

### 1.8.2.2 ehcache

ehcacheehcache.xml

<http://www.ehcache.org/documentation/ehcache-2.6.x-documentation.pdf>

```
<ehcache xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="ehcache.xsd" maxEntriesLocalHeap="100000000" maxBytesLocalDisk="50G" updateCheck="false">
  <defaultCache maxElementsInMemory="1000000" eternal="false" overflowToDisk="false" diskSpoolBufferSizeMB="30" maxElementsOnDisk="1000000" diskPersistent="false" diskExpiryThreadIntervalSeconds="120" memoryStoreEvictionPolicy="LRU"/>
</ehcache>
```

1.dbleehcachedefaultCachecache

2.maxEntriesLocalHeap

0 cacheservice.propertiesmax\_sizecacheservice.properties max\_size

3.timeToldleSeconds

cacheservice.properties expire\_seconds

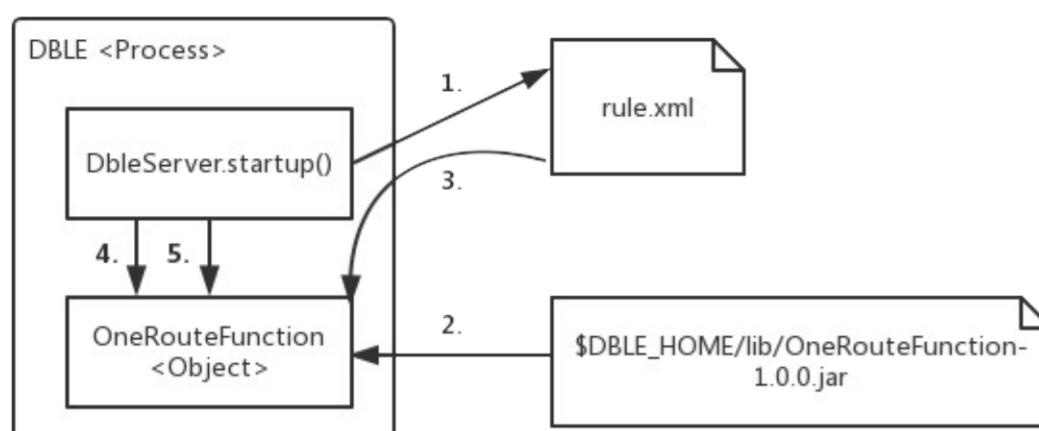
4.dbledefaultCachecache

## 1.9

### 1.9.1

#### 1.9.1.1

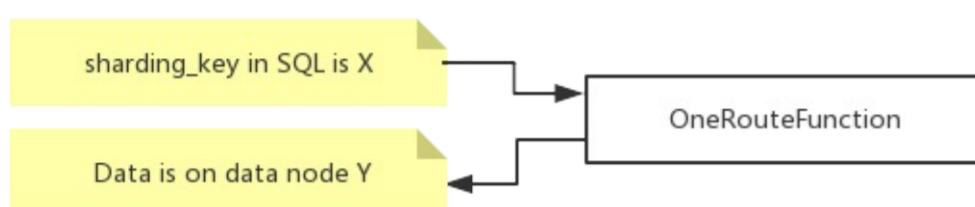
dbe



1. dblesharding.xml class
2. dbleJava\$DBLE\_HOME/libjarjarclass
3. dblenamesetter—— 2dblesetPartitionCount("2")
4. dbleselfCheck()
5. dbleinit()

#### 1.9.1.2

SQLDBLESQ



#### 1.9.1.3

9066SHOW @@ALGORITHM WHERE SCHEMA=? AND TABLE=?dbegetAllProperties()

```

mysql> show @@algorithm where schema=testdb and table=seqtest;
+-----+
| KEY          | VALUE           |
+-----+
| TYPE         | SHARDING TABLE |
| COLUMN       | ID              |
| CLASS        | com.actiontech.dble.route.function.PartitionByLong |
| partitionCount | 2               |
| partitionLength | 1              |
+-----+
5 rows in set (0.05 sec)

```

## 1.9.2

### 1.9.2.1

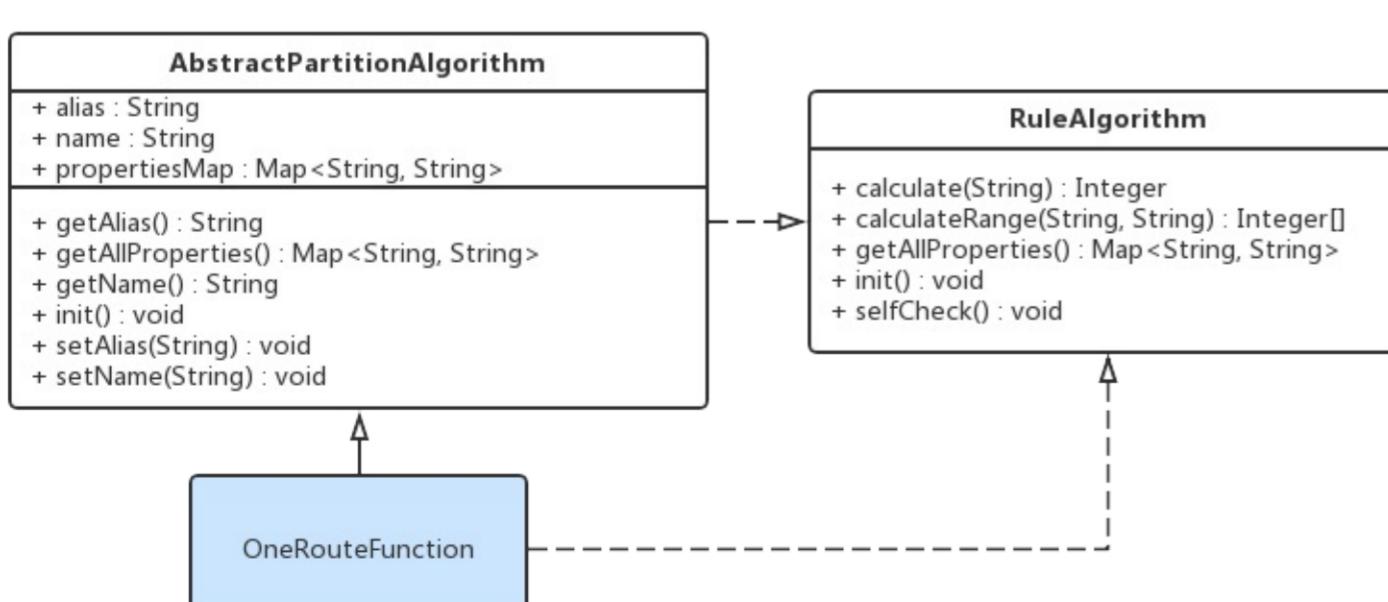
AbstractPartitionAlgorithmRuleAlgorithmAbstractPartitionAlgorithmTableConfigDBLE  
DBLE DBLEDBLE  
1. jarDBLEDBLE  
2. DBLEAbstractPartitionAlgorithmRuleAlgorithmDBLEDBLE  
3. DBLE

### 1.9.2.2

jar classlibraryjarclass  
DBLE  
1. jar\$DBLE\_HOME/lib  
2. jarchownchmod\$DBLE\_HOME/libjar  
3. sharding.xml classFully Qualified Namenet.john.dble.route.functions.NewFunction  
4. DBLE

## 1.9.3

AbstractPartitionAlgorithmRuleAlgorithmcom.actiontech.dble.route.function.PartitionByLong



#### 1.9.3.1 setters

sharding.xmlpartitionCountpartitionLength

```

<function name="hashmod" class="com">
<property name="partitionCount">4</property>
<property name="partitionLength">1</property>
</function>

```

dblepartitionCount4partitionLength1PartitionByLongsetPartitionCount()setPartitionLength()sharding.xmlXMLStringsetter

```

public void setPartitionCount(String partitionCount) {
    this.count = toIntArray(partitionCount);
}

```

```

    /* getAllProperties() */
    propertiesMap.put("partitionCount", partitionCount);
}

public void setPartitionLength(String partitionLength) {
    this.length = toIntArray(partitionLength);
    /* getAllProperties() */
    propertiesMap.put("partitionLength", partitionLength);
}

```

### 1.9.3.2 selfCheck()

```

dbleSelfCheck()RuntimeExceptiondbleRuntimeExceptiondble
selfCheck()RuleAlgorithmAbstractPartitionAlgorithm

```

```

@Override
public void selfCheck() {
}

```

### 1.9.3.3 init()

```

dbleInit()PartitionUtil
PartitionByLonginit()PartitionUtil

```

```

@Override
public void init() {
    partitionUtil = new PartitionUtil(count, length);

    initHashCode();
}

```

### 1.9.3.4 calculate()calculateRange()

```

dbleSQLSQLcalculate()calculateRange()SQL
IPOInput-Process-Outputcalculate()calculateRange()

```

- InputSQL
- OutputSQL
- ProcessInputOutput

calculate()calculateRange()

		Input	Output
calculate()	SQL ... WHERE sharding_key = 1	1String	1Integer
calculateRange()	SQL ... WHERE sharding_key BETWEEN 1 AND 5	2String	Integer

```

@Override
public Integer calculate(String columnName) {
    try {
        if (columnName == null || columnName.equalsIgnoreCase("NULL")) {
            return 0;
        }
        long key = Long.parseLong(columnName);
        return calculate(key);
    } catch (NumberFormatException e) {
        throw new IllegalArgumentException("columnName:" + columnName + " Please eliminate any quote and non number within it.", e);
    }
}

@Override
public Integer[] calculateRange(String beginValue, String endValue) {
    long begin = 0;
    long end = 0;
    try {
        begin = Long.parseLong(beginValue);
        end = Long.parseLong(endValue);
    } catch (NumberFormatException e) {
        return new Integer[0];
    }
    int partitionLength = partitionUtil.getPartitionLength();
    if (end - begin >= partitionLength || begin > end) { //TODO: optimize begin > end
        return new Integer[0];
    }
    Integer beginNode = calculate(begin);
    Integer endNode = calculate(end);

    if (endNode > beginNode || (endNode.equals(beginNode) && partitionUtil.isSingleNode(begin, end))) {
        int len = endNode - beginNode + 1;
        Integer[] re = new Integer[len];
        for (int i = 0; i < len; i++) {
            re[i] = beginNode + i;
        }
        return re;
    } else {
        int split = partitionUtil.getSegmentLength() - beginNode;
        int len = split + endNode + 1;
        if (endNode.equals(beginNode)) {
            //remove duplicate
            len--;
        }
        Integer[] re = new Integer[len];
        for (int i = 0; i < split; i++) {
            re[i] = beginNode + i;
        }
        for (int i = split; i < len; i++) {
            re[i] = i - split;
        }
        return re;
    }
}

```

### 1.9.3.5 getAllProperties()

```

dbleGetAllProperties()<, >

```

```

getAllProperties()RuleAlgorithmAbstractPartitionAlgorithmpropertiesMap"propertiesMap"getAllProperties()setters<, >put()propertiesMap

```

```

@Override
public Map<String, String> getAllProperties() {
    return propertiesMap;
}

```

## 1.9.4

```

DBLEcom.actiontech.dble.route.functions.sharding.xmlXMLLoaderRuleLoader?

```

date	PartitionByDate
enum	PartitionByFileMap
hash	PartitionByLong
jumpstringhash	PartitionByJumpConsistentHash
numberrange	AutoPartitionByLong
patternrange	PartitionByPattern
stringhash	PartitionByString

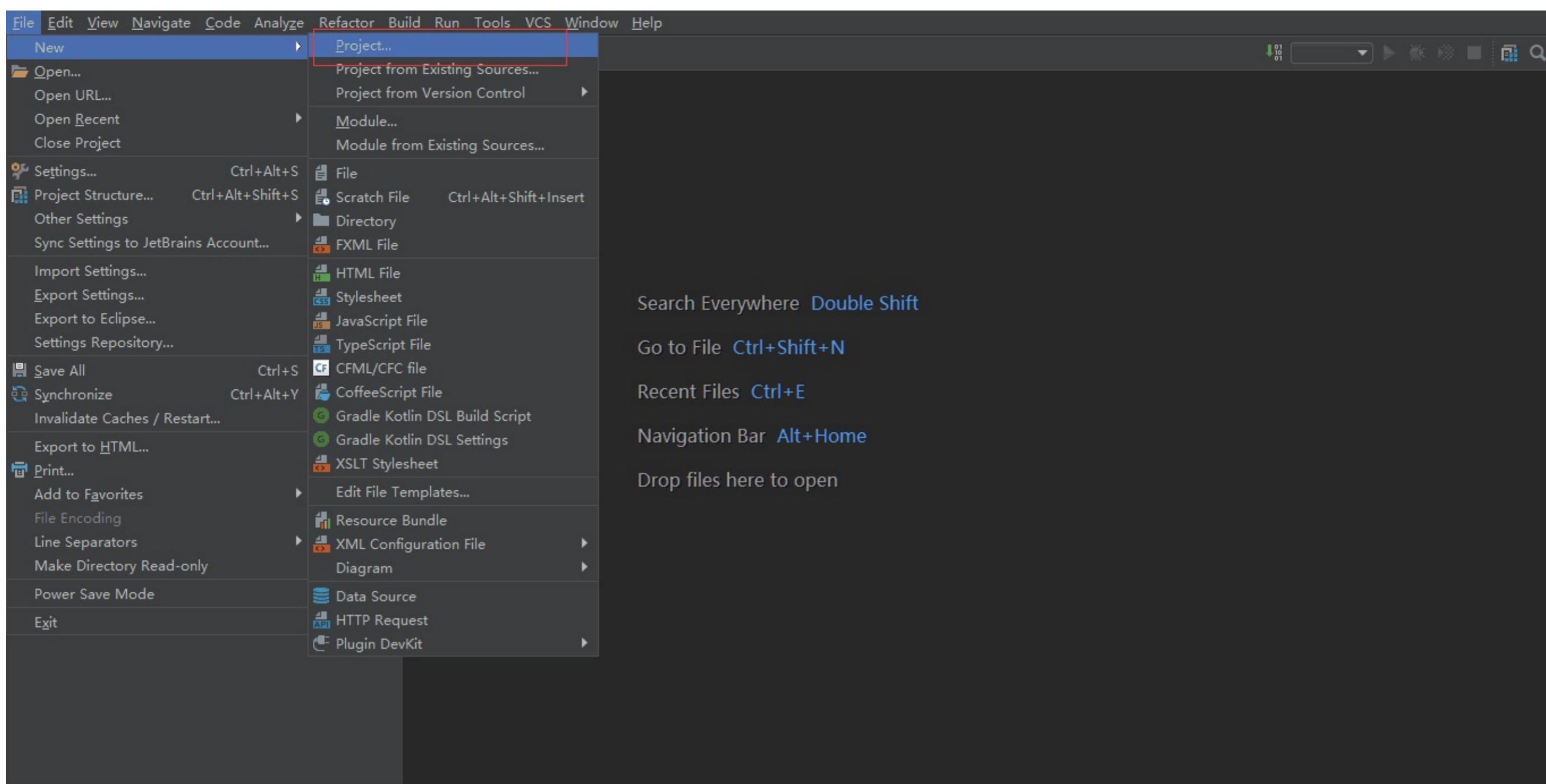
### 1.9.5 IntelliJ IDEA

#### 1.9.5.0

1. java  
 2. dblerleasejar2.19.05.0 <https://github.com/actiontech/dble/releases/download/2.19.05.0%2Ftag/actiontech-dble-2.19.05.0.tar.gz> libdblejar

### 1.9.5.1 java

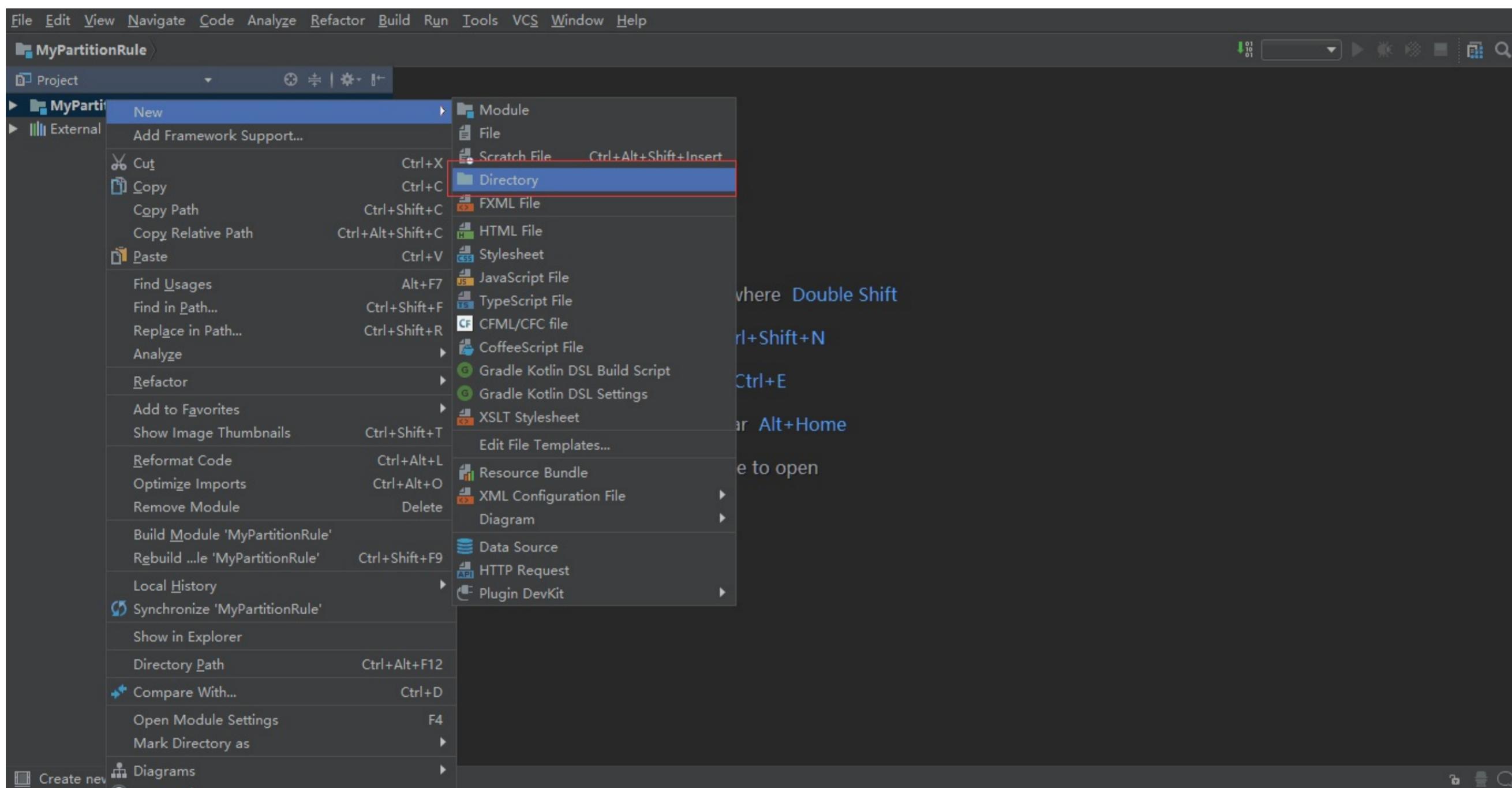
javaProject



Projectjavanext

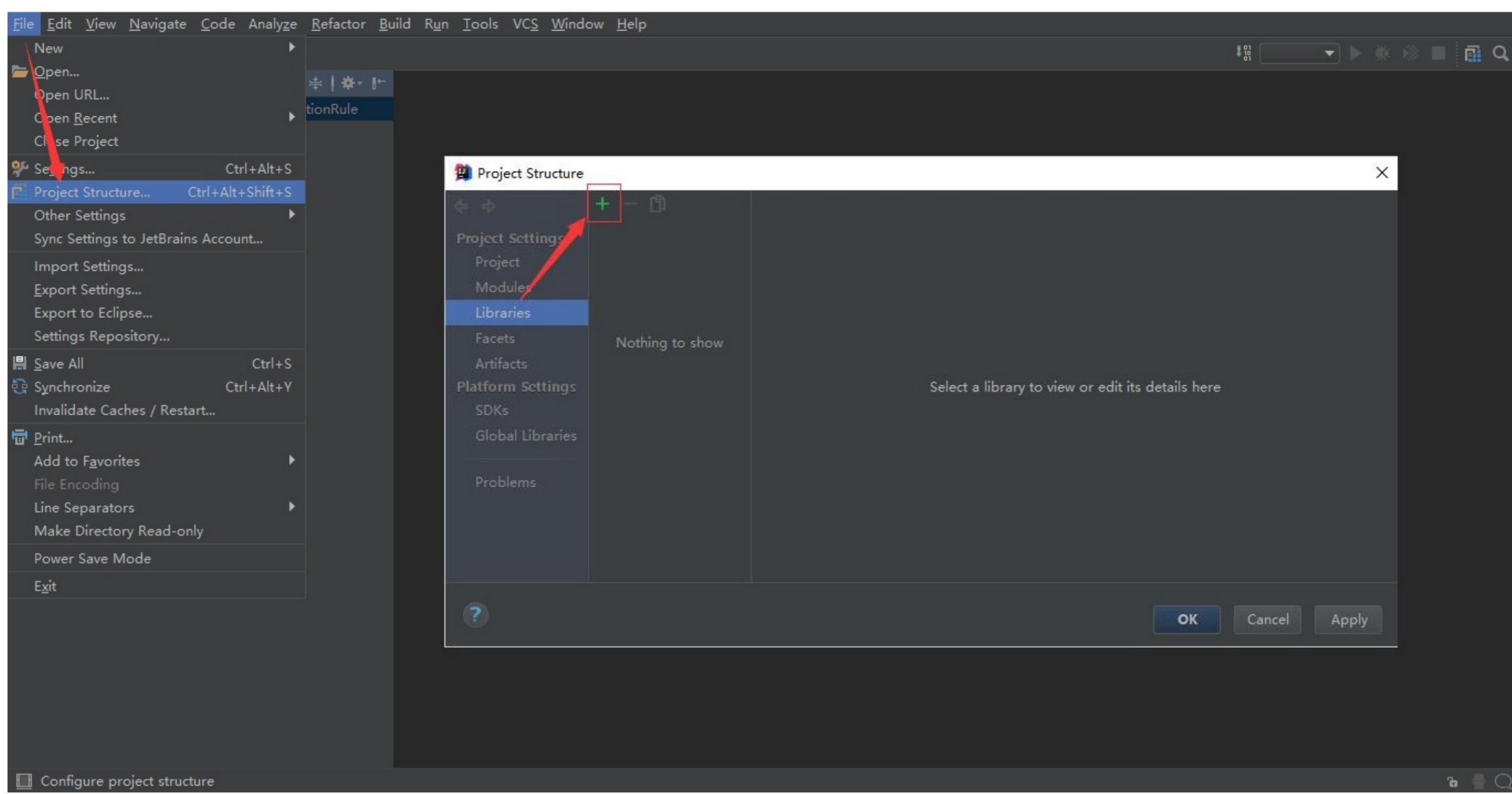
### 1.9.5.2 dble jar

javalib



dble jarlib

libdble jar



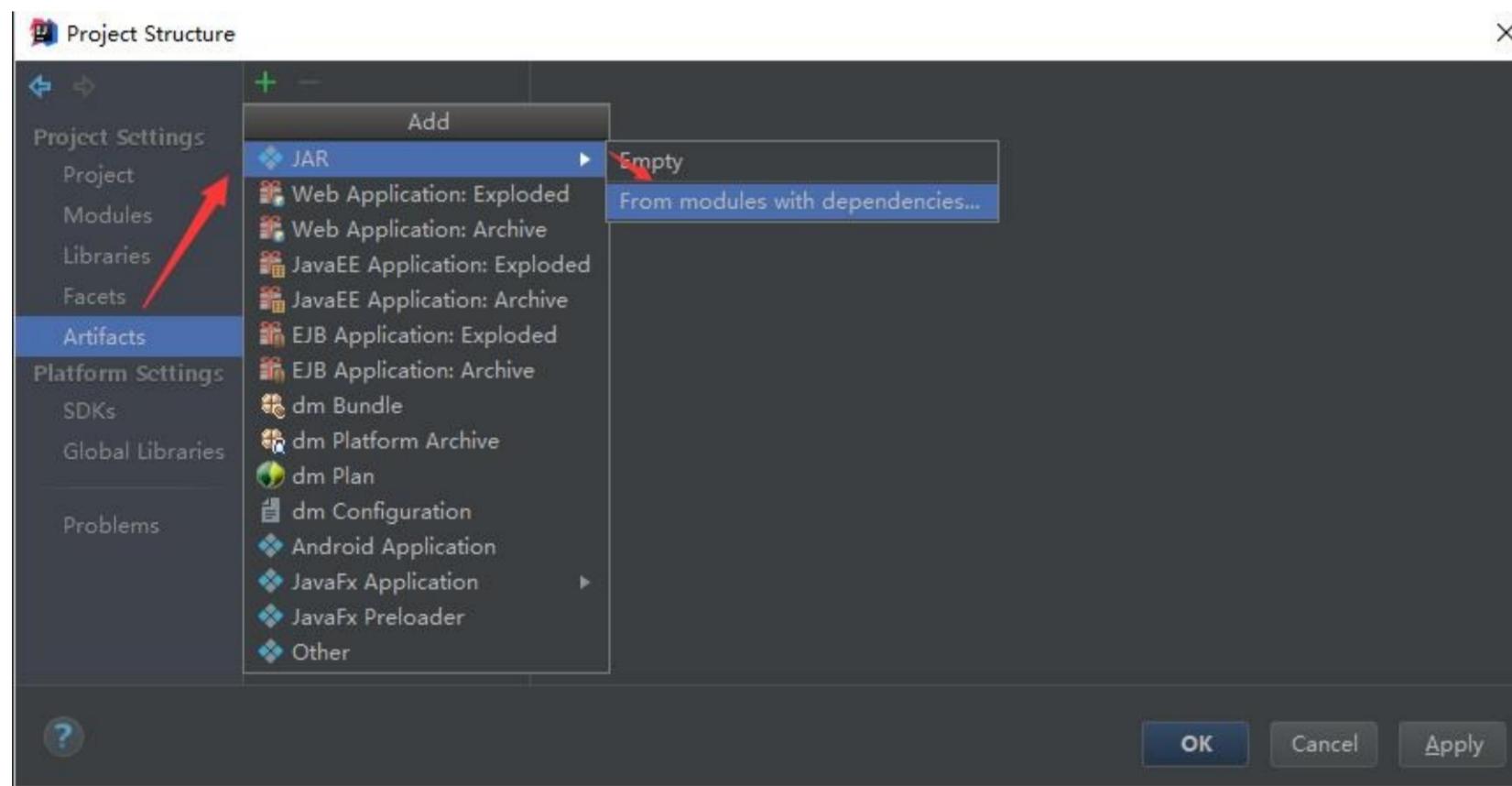
Configure project structure

lib

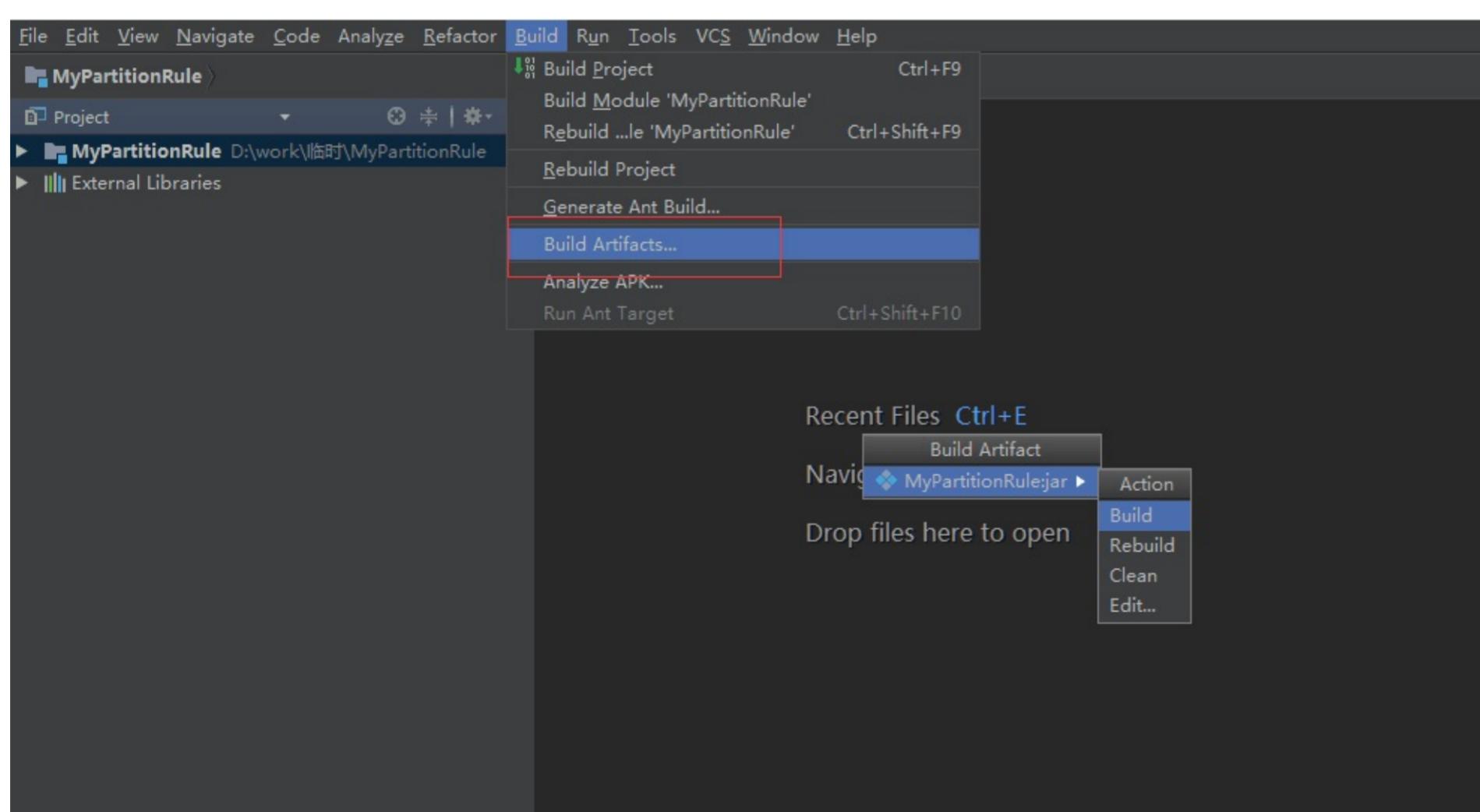
lib

**1.9.5.3****1.9.5.4 Artifacts**

Project StructureArtifactsOK

**1.9.5.5 jar**

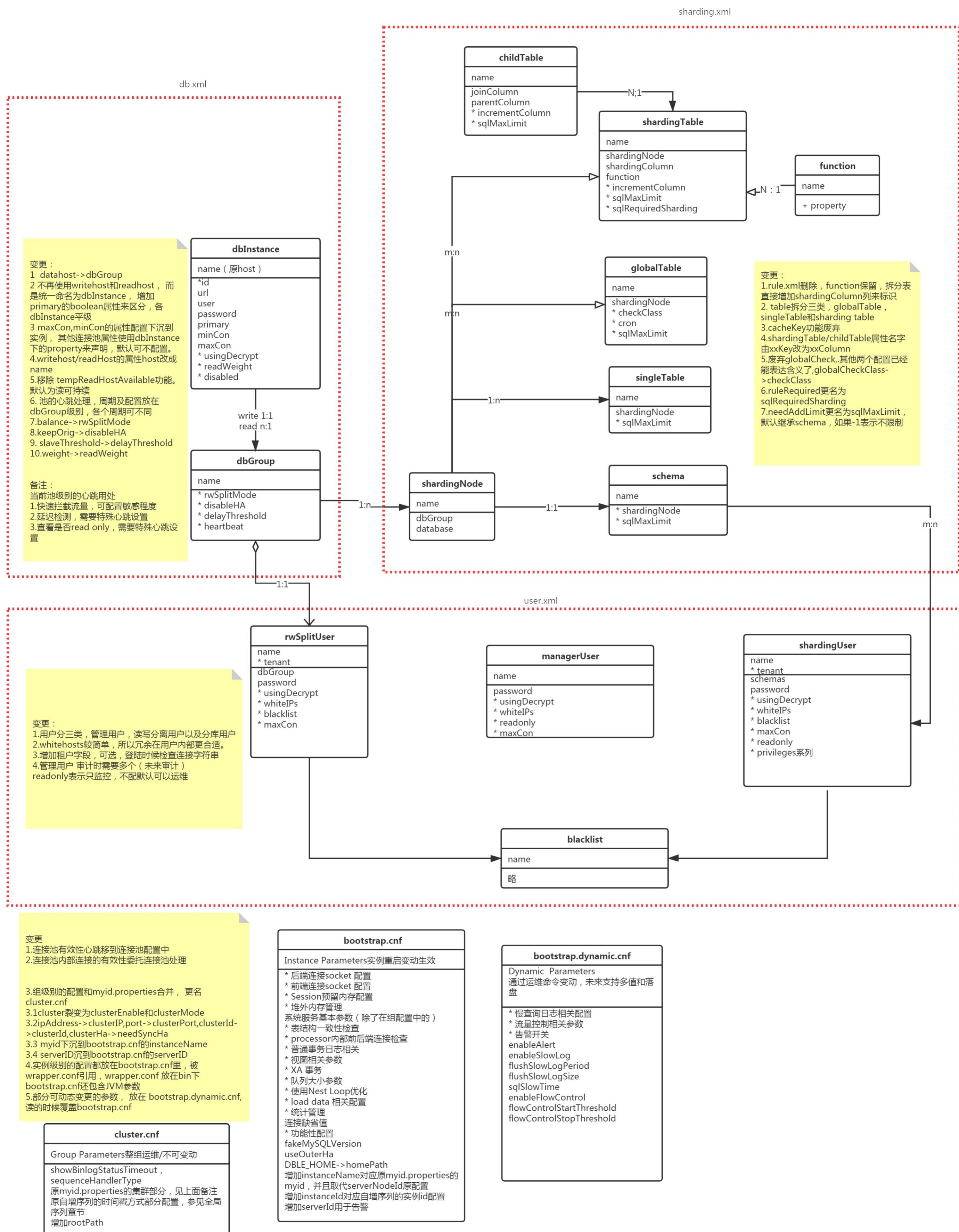
BuildBuild ArtifactsArtifactbuild



outartifactjar

## 1.10

db.xml 2.20.04.0

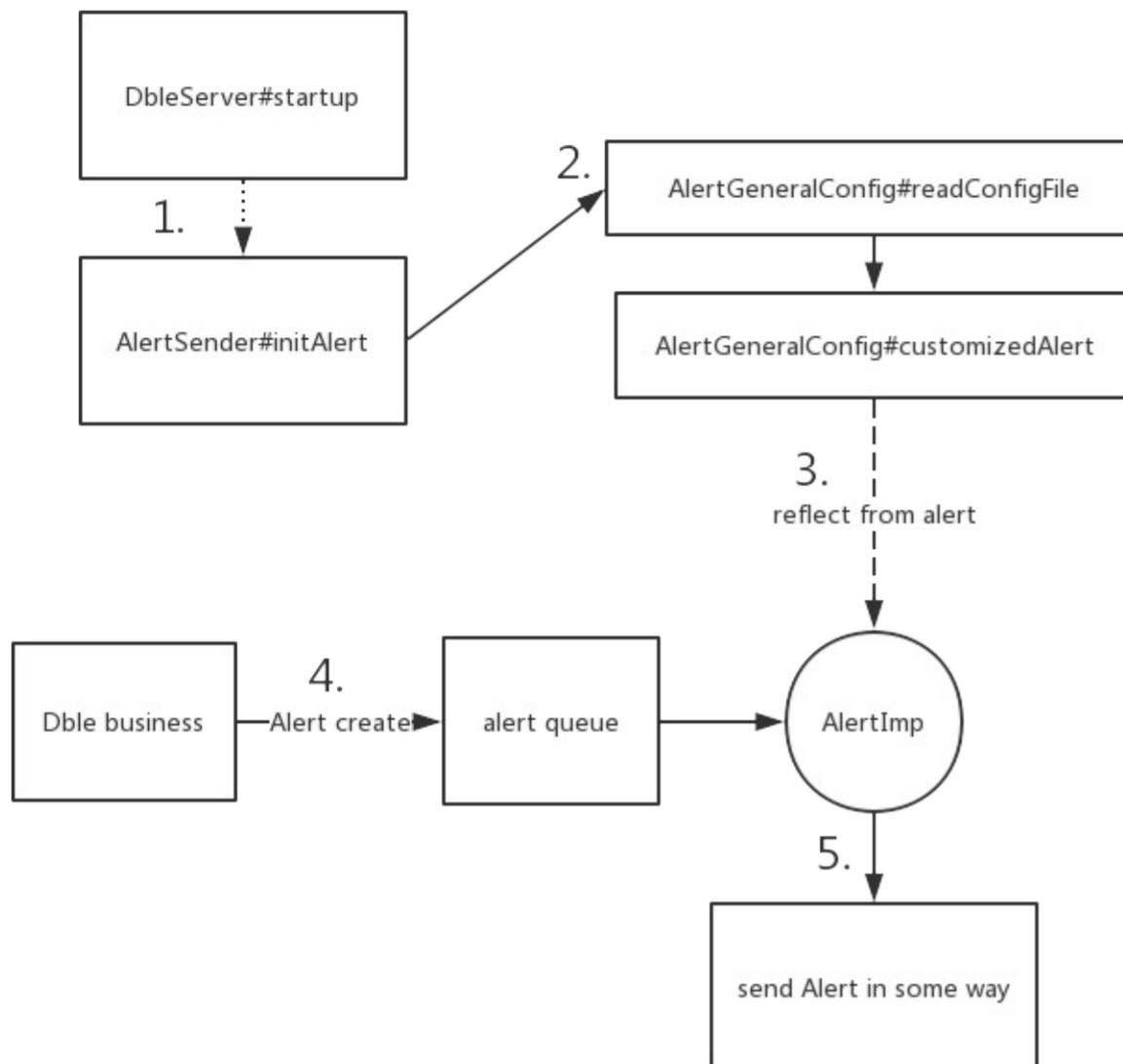


## 1.11

- 
- 
- 
- [Interface Alert](#)
- 
- 
- 
- 
- 

### 1.11.1

#### 1.11.1.1



#### dble

1. dble
2. AlertGeneralConfigdble\_alert.properties
3. alert
4. dble
5. dble

#### 1.11.1.2

##### dble

- 
- 
- dbleXA
- dblexa
- dble
- dble
- dble

“dble”dblekill “dble”dble

##### dble

- -dble
- -dble
- -dble
- -dble

##### dble

- dble
- dble
- dble
- dble

#### 1.11.1.3 Interface Alert

```

public interface Alert {
    void alertSelf(ClusterAlertBean bean);
    void alert(ClusterAlertBean bean);
    boolean alertResolve(ClusterAlertBean bean);
    boolean alertSelfResolve(ClusterAlertBean bean);
    void alertConfigCheck() throws Exception;
}
  
```

##### alertConfigCheck

- alert --- dble
- alertSelf --- dble
- alertResolve --- dble
- alertSelfResolve --- dble

##### dble

alertSelfalertClusterAlertBeanalertComponentTypealertComponentIdalertSelfDBLEserverAlert alertResolvealertSelfResolve

#### 1.11.1.4

```

public class ClusterAlertBean {
    String code;           //
    String level;          //
    String desc;           //
    String sourceComponentType; //
    String sourceComponentId; //ID
    String alertComponentType; //
    String alertComponentId; //ID
    String serverId;        //
    long timestampUnix;      //
  
```

```
long resolveTimestampUnix; //  
Map<String, String> labels; //  
}
```

dhle

1.11.2

1.11.2.1

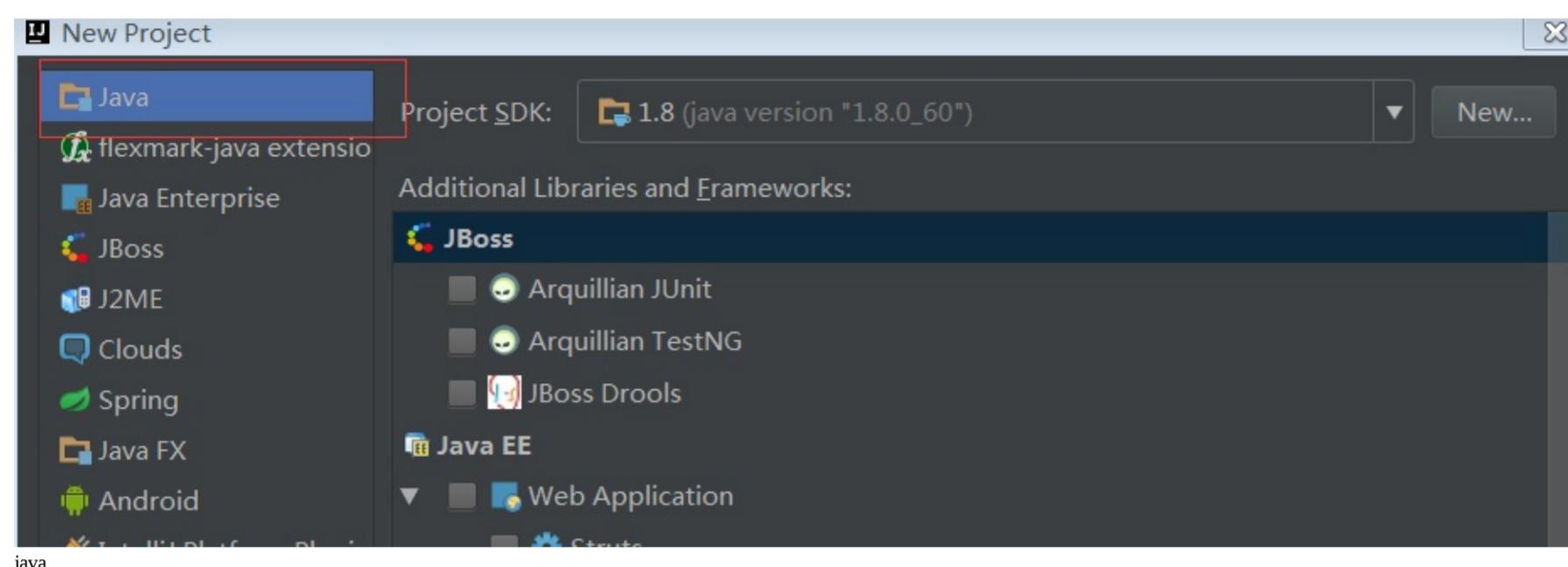
bledble

- dbleAlert
  - dbledble
  - dbleAlert
  - dble“”“”“”“”

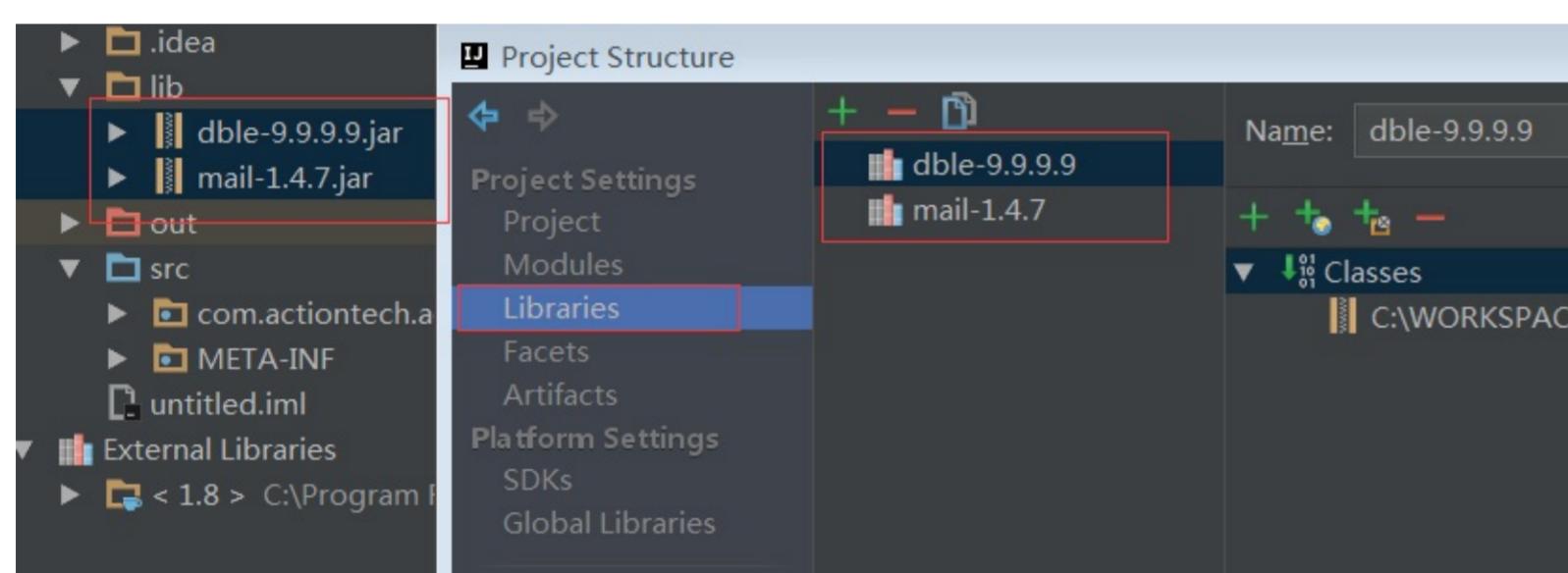
### dble -----Alert

IDEA

1 java



2 copylib



javamailjar

### 3.1 Alert

```
public MailAlert() {
    //init the mail data and read config file
    properties = AlertGeneralConfig.getInstance().getProperties();
}
```

```
AlertMailAlert.AlertGeneralConfig.getInstance().getProperties();dble(dble_alert.properties) dble_alert.properties.Alertke
```

3.2 Alert

```
@Override  
public void alertConfigCheck() throws ConfigException {  
    //check if the config is correct  
    if (properties.getProperty(MAIL_SENDER) == null  
        || properties.getProperty(SENDER_PASSWORD) == null  
        || properties.getProperty(MAIL_SERVER) == null  
        || properties.getProperty(MAIL_RECEIVE) == null) {  
        throw new ConfigException("alert check error, for some config is missing")  
    }  
}
```

dbl\_alert.propertiesMAIL\_SENDERLSENDER\_PASSWORDMAIL\_SERVERMAIL\_RECEIVE dbl

```
properties.getProperty(MATI_RECVTYPE)
```

3.3

send

```
private boolean sendMail(boolean isResolve, ClusterAlertBean clusterAlertBean) {
    try {
        Properties props = new Properties();
        props.setProperty("mail.debug", "true");
        props.setProperty("mail.smtp.auth", "true");
        props.setProperty("mail.host", properties.getProperty(MAIL_SERVER));
        props.setProperty("mail.transport.protocol", "smtp");

        MailSSLSocketFactory sf = new MailSSLSocketFactory();
        sf.setTrustAllHosts(true);
        props.put("mail.smtp.ssl.enable", "true");
        props.put("mail.smtp.ssl.socketFactory", sf);

        Session session = Session.getInstance(props);

        Message msg = new MimeMessage(session);
        msg.setSubject("DBLE " + (isResolve ? "RESOLVE\n" : "ALERT\n"));
        StringBuilder builder = new StringBuilder();
        builder.append(groupMailMsg(clusterAlertBean, isResolve));
        msg.setText(builder.toString());
        msg.setFrom(new InternetAddress(properties.getProperty(MAIL_SENDER)));

        Transport transport = session.getTransport();
        transport.connect(properties.getProperty(MAIL_SERVER), properties.getProperty(MAIL_SENDER), properties.getProperty(SENDER_PASSWORD));

        transport.sendMessage(msg, new Address[]{new InternetAddress(properties.getProperty(MAIL_RECEIVE))});
        transport.close();
        //send EMAIL SUCCESS return TRUE
        return true;
    } catch (Exception e) {
        e.printStackTrace();
    }
    //send fail reutrn false
    return false;
}

private String groupMailMsg(ClusterAlertBean clusterAlertBean, boolean isResolve) {
```

```

sb.append("Alert type:" + clusterAlertBean.getCode() + " " + (isResolve ? "RESOLVE\n" : "ALERT\n"));
sb.append("Alert message:" + clusterAlertBean.getDesc() + "\n");
sb.append("Alert component:" + clusterAlertBean.getAlertComponentType() + "\n");
sb.append("Alert componentID:" + clusterAlertBean.getAlertComponentId() + "\n");
sb.append("Alert source:" + clusterAlertBean.getAlertComponentId() + "\n");
sb.append("Alert server:" + clusterAlertBean.getServerId() + "\n");
sb.append("Alert time:" + TimeStamp2Date(clusterAlertBean.getTimestampUnix()) + "\n");
String detail = "";
if (clusterAlertBean.getLabels() != null) {
    for (Map.Entry<String, String> entry : clusterAlertBean.getLabels().entrySet()) {
        detail += entry.getKey() + ":" + entry.getValue();
    }
}
sb.append("Other detail:" + detail + "\n");
return sb.toString();
}

```

javaproPERTIES group MailMsgclusterAlertBean

3.4

```

@Override
public void alertSelf(ClusterAlertBean clusterAlertBean) {
    alert(clusterAlertBean.setAlertComponentType(COMPARTMENT_TYPE).setAlertComponentId(properties.getProperty(COMPARTMENT_ID)));
}

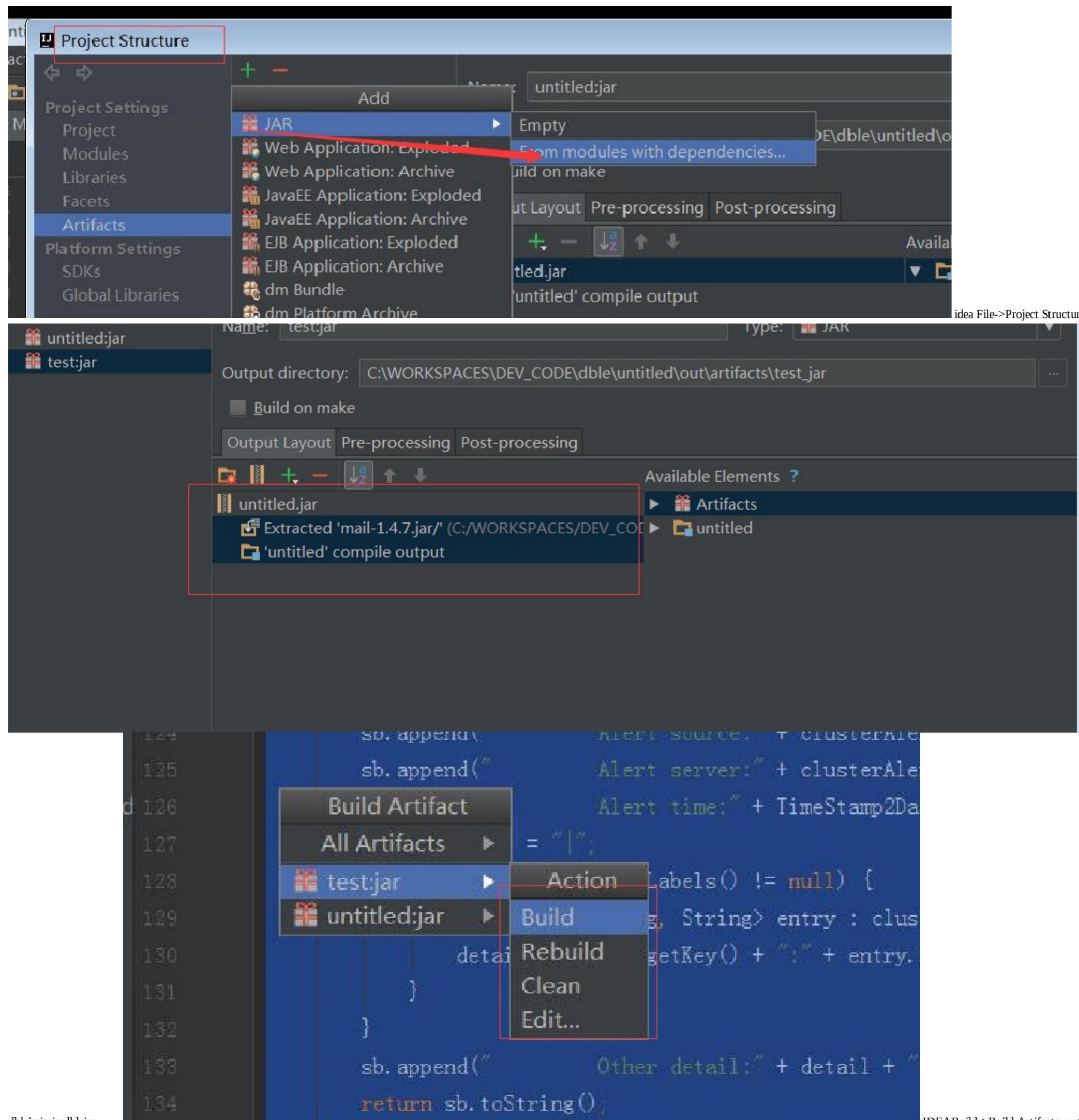
@Override
public void alert(ClusterAlertBean clusterAlertBean) {
    clusterAlertBean.setSourceComponentType(COMPARTMENT_TYPE);
    setSourceComponentId(properties.getProperty(COMPARTMENT_ID));
    setServerId(properties.getProperty(SERVER_ID));
    setTimestampUnix(System.currentTimeMillis() * 1000000);
    sendMail(false, clusterAlertBean);
}

@Override
public boolean alertResolve(ClusterAlertBean clusterAlertBean) {
    clusterAlertBean.setSourceComponentType(COMPARTMENT_TYPE);
    setSourceComponentId(properties.getProperty(COMPARTMENT_ID));
    setServerId(properties.getProperty(SERVER_ID));
    setTimestampUnix(System.currentTimeMillis() * 1000000);
    return sendMail(true, clusterAlertBean);
}

@Override
public boolean alertSelfResolve(ClusterAlertBean clusterAlertBean) {
    return alertResolve(clusterAlertBean.setAlertComponentType(COMPARTMENT_TYPE).setAlertComponentId(properties.getProperty(COMPARTMENT_ID)));
}

```

3.5 jar



## 1.11.2.2

jar

- jardblelib
- dble\_alert.properties,alert
- dble lib dble\_alert.properties,

```

alert=com.actiontech.additionAlert.MailAlert
mail_sender=12345678@qq.com
sender_pass=qwertyuiop
mail_server=smtpp.qq.com
mail_receive=yyyyyyyyyyyy@actionsky.com
server_id=dble-server-001
component_id=DBLE-FOR-XXX-01

```

dblecom.actiontech.addtionAlert.MailAlert

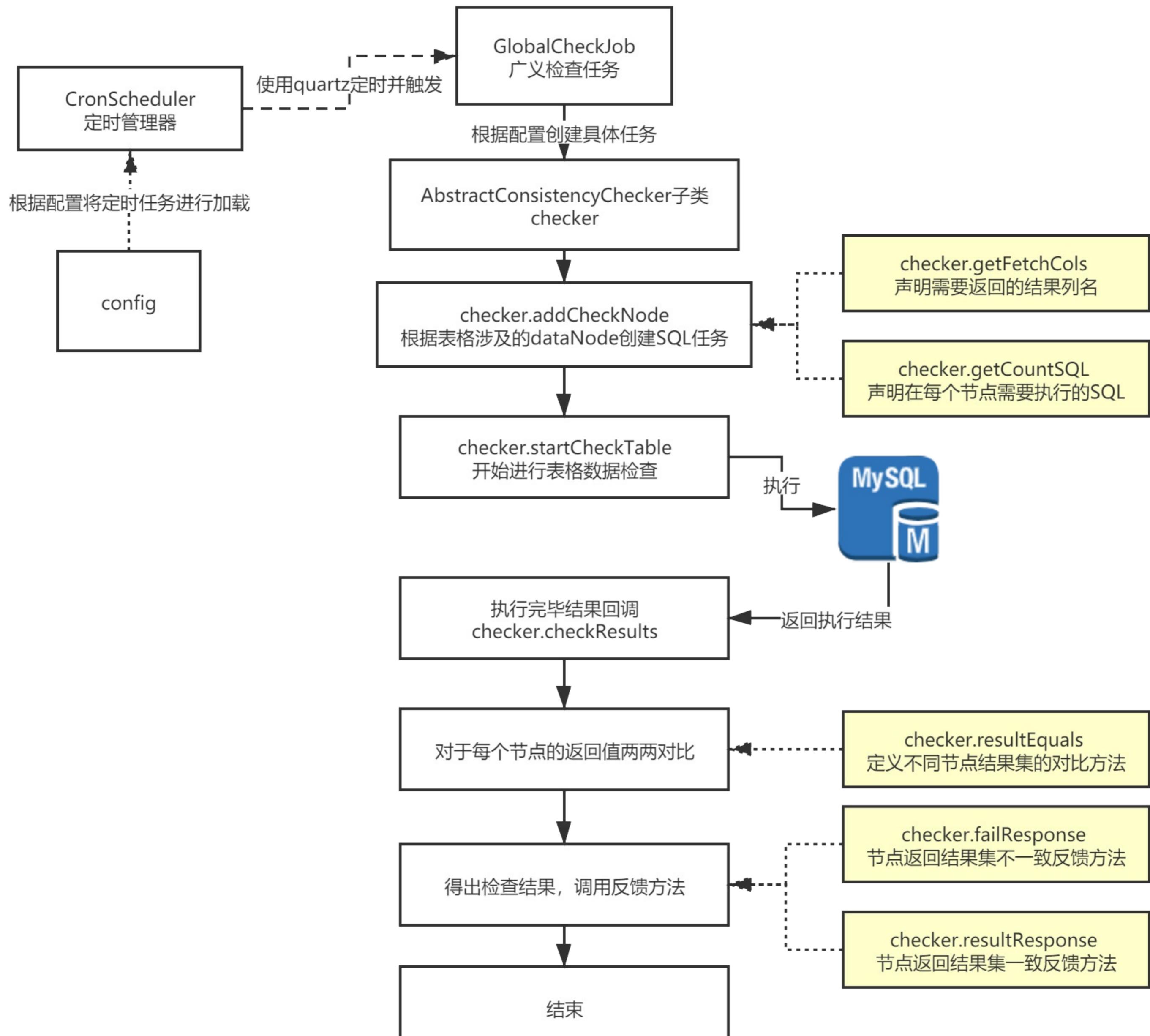
**1.11.2.3**[jar](#)**1.11.3 dbleCODE**

DBLE_WRITE_TEMP_RESULT_FAIL	/
DBLE_XA_RECOVER_FAIL	XA
XA_READ_XA_STREAM_FAIL	XA
DBLE_XA_READ_DECODE_FAIL	XA
DBLE_XA_READ_IO_FAIL	XA
DBLE_XA_WRITE_IO_FAIL	XA
DBLE_XA_WRITE_CHECK_POINT_FAIL	XA
DBLE_XA_BACKGROUND_RETRY_FAIL	XA
DBLE_REACH_MAX_CON	/
DBLE_TABLE_NOT_CONSISTENT_IN_SHARDINGS	/
DBLE_TABLE_NOT_CONSISTENT_IN_MEMORY	dble
DBLE_GLOBAL_TABLE_COLUMN_LOST	/
DBLE_CREATE_CONN_FAIL	mysql
DBLE_DB_INSTANCE_CAN_NOT_REACH	/
DBLE_KILL_BACKEND_CONN_FAIL	Kill
DBLE_NIOREACTOR_UNKNOWN_EXCEPTION	NIO
DBLE_NIOREACTOR_UNKNOWN_THROWABLE	NIO
DBLE_NIOCONNECTOR_UNKNOWN_EXCEPTION	NIO
DBLE_TABLE_LACK	/
DBLE_GET_TABLE_META_FAIL	/
DBLE_TEST_CONN_FAIL	/
DBLE_HEARTBEAT_FAIL	/
DBLE_SHARDING_NODE_LACK	shardingNode

- getCountSQL
- getFetchCols
- resultEquals
- failResponse
- resultResponse

dbletable\_atable\_a

dbletable\_a  
dbletable\_a



- reloadCronScheduler
- GlobalCheckJob
- GlobalCheckJobSQL
  - checkerSQL
  - shardingNodeSQL
  - SQLSQLSQLMySQL
- SQL0
- SQLcheckResults
- checkerSQL
- SQL(failResponseSQL(resultResponse

**1.SQLString getCountSQL(String dbName, String tableName)**

```
SQL
SQLMySQLdatabase
SQL

public String getCountSQL(String dbName, String tableName) {
    //tablechecksum
    return "checksum table " + tableName;
}
```

**2. getFetchCols()**

SQL

list

```
public String[] getFetchCols() {
    //checksumChecksum
    // mysql> checksum table suntest;
    //+-----+-----+
    //| Table | Checksum |
    //+-----+-----+
    //| db1.suntest |1290812451|
    //+-----+-----+
    //returnChecksum
    return new String[]{"Checksum"};
}
```

**3.SQL boolean resultEquals(result1,result2)**

resultresult1,result2

```
SQLQueryResult<List<Map<String, String>>> result

result
|
----- row(List)
|
-----Key-Value(Field-Value)
checksum
result
|
----- row(List<I> checksum table suntest)
|
-----Key-Value(checksum - 1290812451 getFetchCols)
```

list

```
public boolean resultEquals(SQLQueryResult<List<Map<String, String>>> or, SQLQueryResult<List<Map<String, String>>> cr) {
    //checksum
    //
    Map<String, String> oresult = or.getResult().get(0);
    Map<String, String> cresult = cr.getResult().get(0);
    //Mapchecksum
    return (oresult.get("Checksum") == null && cresult.get("Checksum") == null) ||
        (oresult.get("Checksum") != null && cresult.get("Checksum") != null &&
            oresult.get("Checksum").equals(cresult.get("Checksum")));
}
```

**4. failResponse(resultList)**

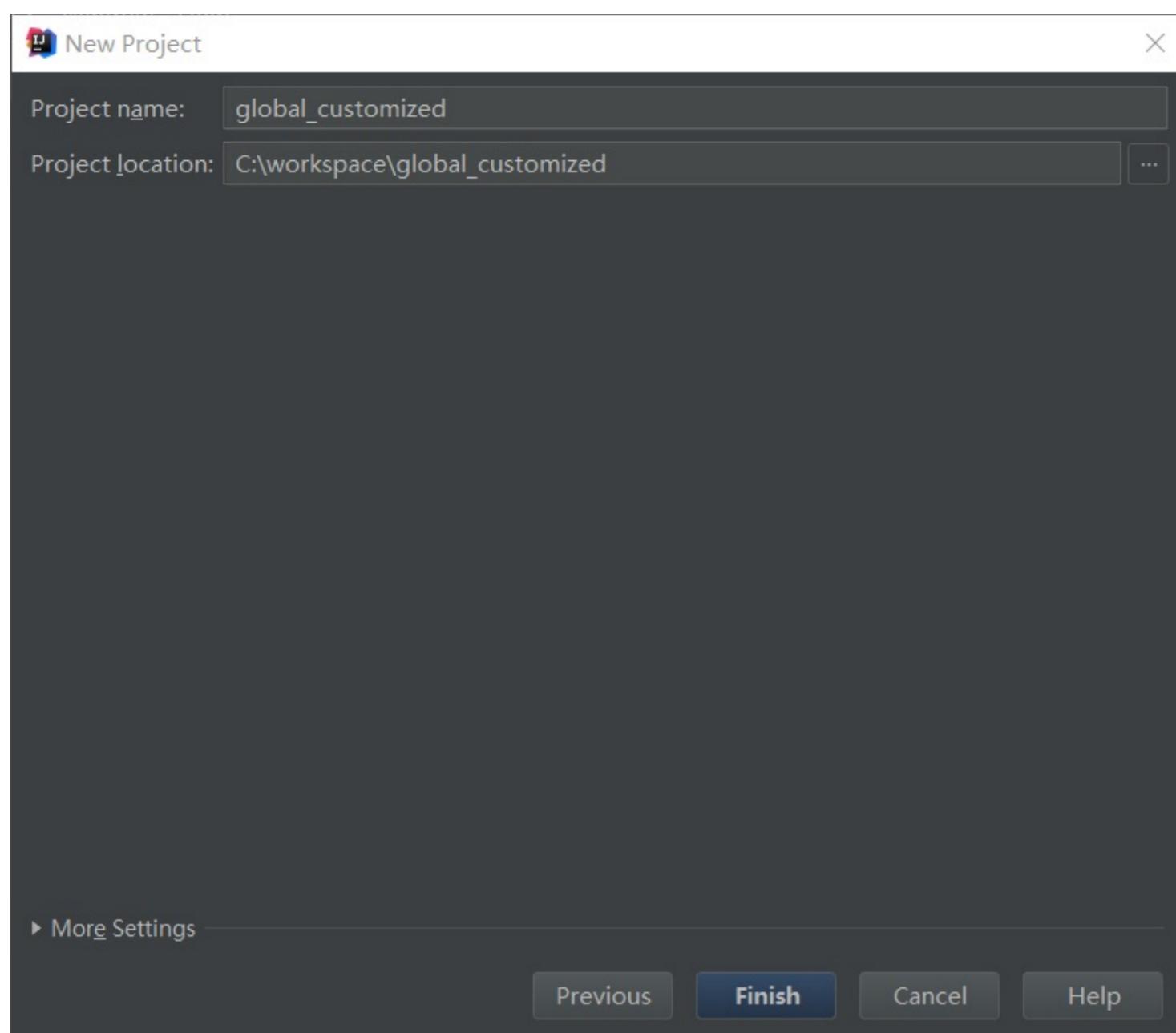
//

```
public void failResponse(List<SQLQueryResult<List<Map<String, String>>>> res) {
    /**
     * Global Consistency Check fail for table :" + schema + "-" + tableName;
     */
    System.out.println(errorMsg);
    for (SQLQueryResult<List<Map<String, String>>> r : res) {
        System.out.println("Checksum is : " + r.getResult().get(0).get("Checksum"));
    }
}
```

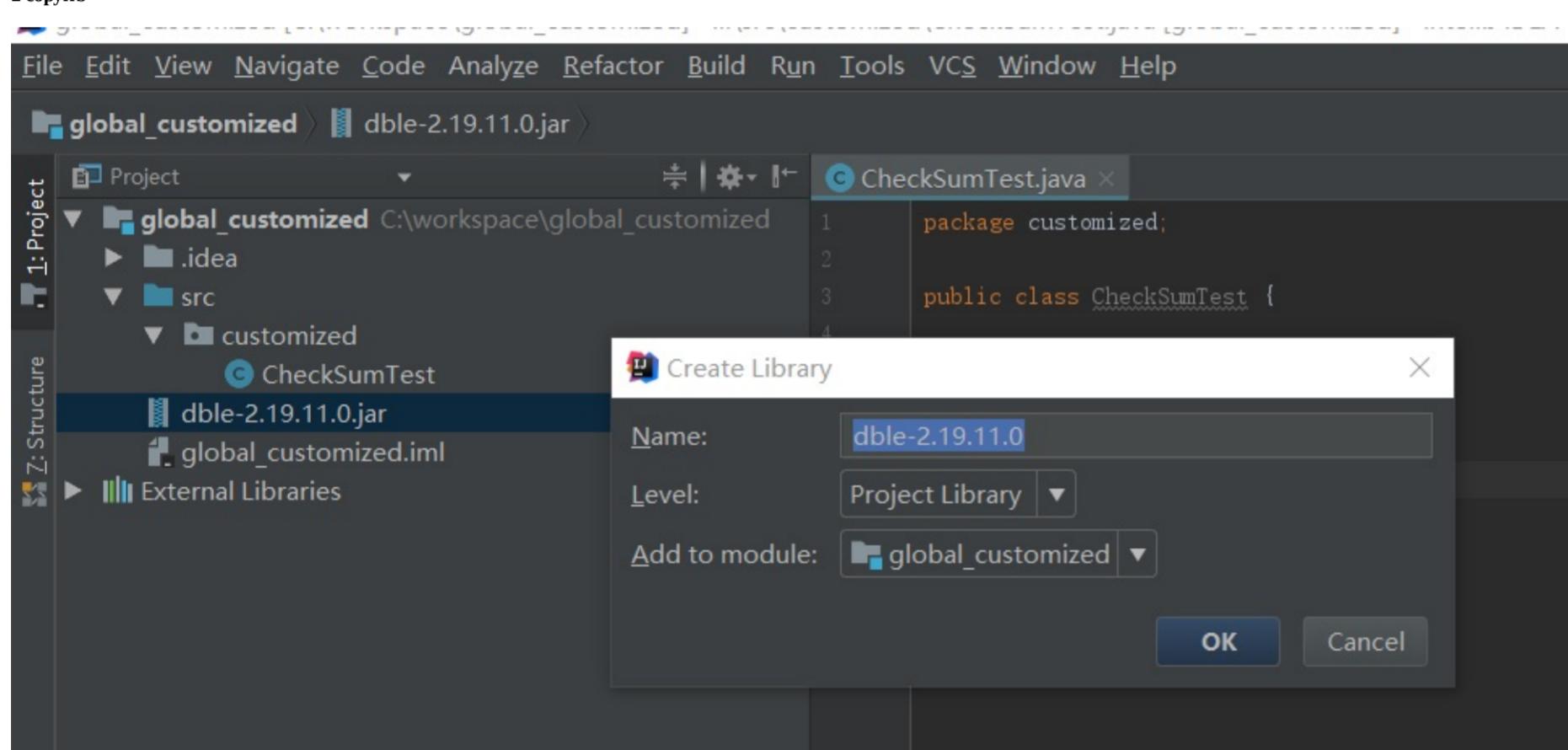
**5. void resultResponse(errorList)**//  
0

```
public void resultResponse(List<SQLQueryResult<List<Map<String, String>>>> elist) {
    //SQLlistSQL
    //SQLSQL
    //
    String tableId = schema + "." + tableName;

    if (elist.size() == 0) {
        System.out.println("Global Consistency Check success for table :" + schema + "-" + tableName);
    } else {
        System.out.println("Global Consistency Check fail for table :" + schema + "-" + tableName);
        StringBuilder sb = new StringBuilder("Error when check Global Consistency, Table ");
        sb.append(tableName).append(" shardingNode ");
        for (SQLQueryResult<List<Map<String, String>>> r : elist) {
            System.out.println("error node is : " + r.getTableName() + "-" + r.getShardingNode());
            sb.append(r.getShardingNode()).append(",");
        }
        sb.setLength(sb.length() - 1);
    }
}
```



2 copylib



35

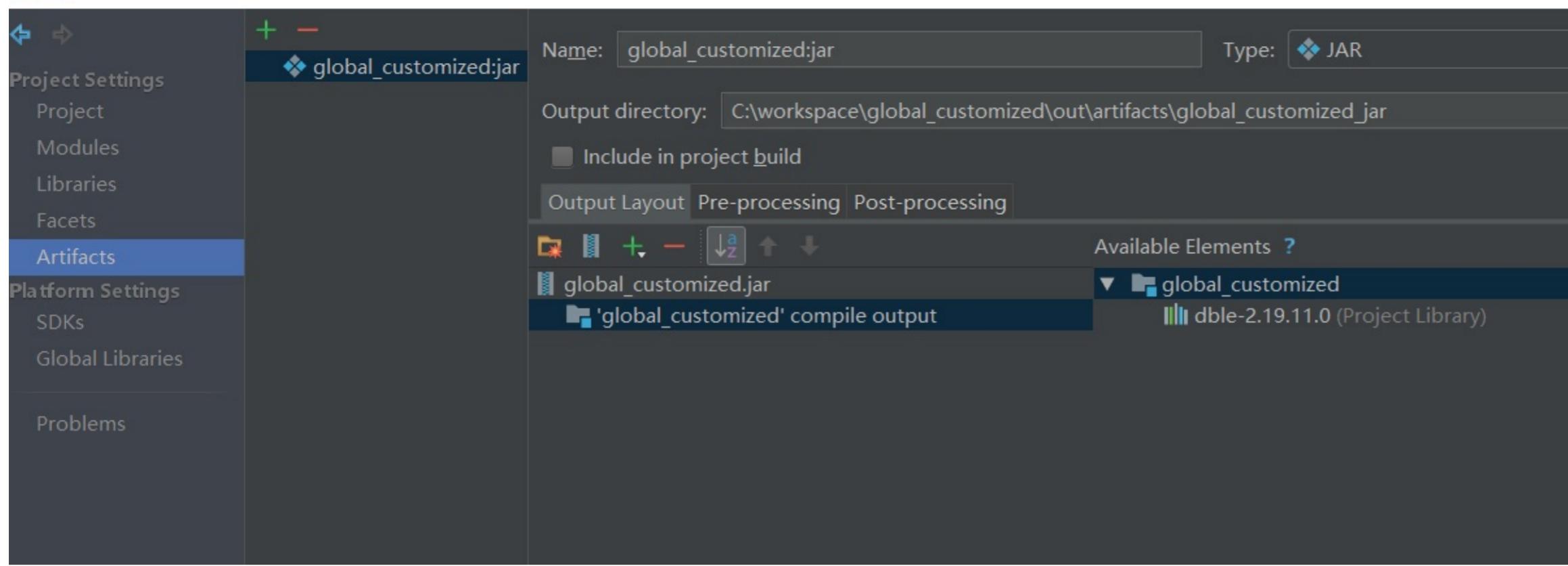
**按照自身的需要个实现这几个方法**

```

1 import com.actiontech.dble.backend.datasource.check.AbstractConsistencyChecker;
2 import com.actiontech.dble.sqlengine.SQLQueryResult;
3
4 import java.util.List;
5 import java.util.Map;
6
7 public class CustomizeTest extends AbstractConsistencyChecker {
8
9
10    @Override
11    public String[] getFetchCols() {
12        return new String[0];
13    }
14
15    @Override
16    public String getCountSQL(String s, String s1) {
17        return null;
18    }
19
20    @Override
21    public boolean resultEquals(SQLQueryResult<List<Map<String, String>>> sqlQueryResult, SQLQueryResult<List<Map<String, String>>> sqlQueryResult1) {
22        return false;
23    }
24
25    @Override
26    public void failResponse(List<SQLQueryResult<List<Map<String, String>>>> list) {
27    }
28
29    @Override
30    public void resultResponse(List<SQLQueryResult<List<Map<String, String>>>> list) {
31    }
32
33
34
35

```

4 jar

 Project Structure


5 jar

schema.table  
*:reload*

```
<!--dbleCHECKSUM-->
<globalTable name="tb_global1" shardingNode="dn1,dn2" cron = "0 * * * ?" globalCheckClass="CHECKSUM"/>

<!--dbleCOUNT-->
<globalTable name="tb_global2" shardingNode="dn1,dn2" cron = "0 * * * ?" globalCheckClass="COUNT"/>

<!--CustomizeTest-->
<globalTable name="tb_global3" shardingNode="dn1,dn2" cron = "0 * * * ?" globalCheckClass="CustomizeTest"/>
```

jardblejaralgorithmlibdblejavajardble  
*jardblereload*

## 2.

- 2.0
- 2.1
  - 2.1.1 select
  - 2.1.2 set
  - 2.1.3 show
  - 2.1.4 switch
  - 2.1.5 kill
  - 2.1.6 stop
  - 2.1.7 reload
  - 2.1.8 rollback
  - 2.1.9 offline
  - 2.1.10 online
  - 2.1.11 file
  - 2.1.12 log
  - 2.1.13
  - 2.1.14 pause & resume
  - 2.1.15
  - 2.1.16 /
  - 2.1.17 check @@metadata
  - 2.1.18 release @@reload\_metadata
  - 2.1.19 split
  - 2.1.20 flow\_control
  - 2.1.21
- 2.2
  - 2.2.1 MySQL offset-step
  - 2.2.2
  - 2.2.3
  - 2.2.4 offset-step
- 2.3
- 2.4
- 2.5
  - 2.5.1 XA
  - 2.5.2 XA
  - 2.5.3 XA
  - 2.5.4 XA
  - 2.5.5
- 2.6
- 2.7
- 2.8 &
- 2.9 grpc
- 2.10 meta
  - 2.10.1 Meta
  - 2.10.2 Meta
  - 2.10.3
  - 2.10.4 View Meta
- 2.11
  - 2.11.1
  - 2.11.2
  - 2.11.3
  - 2.11.4
  - 2.11.5 heartbeat
  - 2.11.6
- 2.12
- 2.13
- 2.14 ER
- 2.15 global
- 2.16
- 2.17
- 2.18
- 2.19 reload
- 2.20
- 2.21 SQLtrace
- 2.22 KILL @@DDL\_LOCK
- 2.23
- 2.24
- 2.25
- 2.26 client\_found\_rows

## 2.0 dble\_information

### 2.0.0

```
dble_information dbledbleuse dble_information
:
(select)
(where)
(join)
where
```

```
use dble_information
show tables [ like ]
desc|describe table xxx
show databases; show @@database
INSERT/UPDATE/DELETE
```

### 2.0.1 dble\_information

#### 2.0.1.0 version

- version
- dble
- 

version	true	

- - dble

#### 2.0.1.1 dble\_variables

- dble\_variables
- 
- 

variable_name	true	
variable_value		
comment		
read_only		

- - version\_comment
  - isOnline:
  - heap\_memory\_max (mb)
  - direct\_memory\_max: -XX:MaxDirectMemorySize

```
show @@sysparam
```

#### 2.0.1.2 dble\_status

- dble\_status
- 
- 

variable_name	true	
variable_value		
comment		

- - uptime: dble()
  - current\_timestamp: dble
  - startup\_timestamp: dble
  - heap\_memory\_max:
  - heap\_memory\_used:
  - heap\_memory\_total:
  - config\_reload\_timestamp: config
  - config\_reload\_rollback\_timestamp: config rollback
  - direct\_memory\_max: -XX:MaxDirectMemorySize
  - direct\_memory\_pool\_size: bufferpoolpagesizebufferpoolpagenumber
  - direct\_memory\_pool\_used: directmemory
  - questions:
  - transactions:

#### 2.0.1.3 dble\_thread\_pool

- dble\_thread\_pool
- 
- 

name	true	
size		
active_count		
waiting_task_count		(,)

- 

#### 2.0.1.4 dble\_processor

- dble\_processor
- processor
- 

name	true	
type		(session/backend)
conn_count		
conn_net_in	0	
conn_net_out	0	

- 

### 2.0.1.5 dble\_sharding\_node

- dble\_sharding\_node
- sharding\_node
- 

name	true	
db_group		db_group
db_schema		db_schema
pause		

- 

### 2.0.1.6 dble\_db\_group

- dble\_db\_group
- db\_group
- 

name	true	
heartbeat_stmt		sql
heartbeat_timeout		0
heartbeat_retry		
rw_split_mode		
delay_threshold		
disable_ha		
active		

- 

### 2.0.1.7 dble\_db\_instance

- dble\_db\_instance
- db\_instance
- 

name	true	
db_group	true	db_group
addr		
port		
primary		
user		
password_encrypt		
encrypt_configured		
active_conn_count		
idle_conn_count		
read_conn_request		
write_conn_request		
disabled		disabled
last_heartbeat_ack_timestamp		
last_heartbeat_ack		init/ok/error/timeout
heartbeat_status		idle/checking
heartbeat_failure_in_last_5min		5,
min_conn_count		
max_conn_count		
read_weight		
id		id
connection_timeout		
connection_heartbeat_timeout		
test_on_create		
test_on_borrow		
test_on_return		
test_while_idle		
time_between_eviction_runs_millis		
evictor_shutdown_timeout_millis		
idle_timeout		
heartbeat_period_millis		

- 

### 2.0.1.8 dble\_schema

- dble\_schema
- schema
- 

name	true	
sharding_node		sharding_node
sql_max_limit		

- 

### 2.0.1.9 session\_variables

- session\_variables
- 
-

session_conn_id	true	id
variable_name	true	
variable_value		
variable_type		(sys/user)

•

```

- tx_read_only(mysql8.0)
- transaction_read_only(mysql5.7)
- character_set_client
- collation_connection
- character_set_results
- tx_isolation_level(mysql8.0)
- transaction_isolation (mysql5.7)
- autocommit
-

```

**2.0.1.10 session\_connections**

- session\_connections
- 
- 

session_conn_id	true	id
remote_addr		
remote_port		
local_port		
processor_id		id
user		
tenant		
schema		schema? (/)
sql		sql(10241024)
sql_execute_time		sql, sql(ms)20ms
sql_start_timestamp		sql
sql_stage		,finished
conn_net_in		
conn_net_out		
conn_estab_time		0
conn_recv_buffer		0(,,)
conn_send_task_queue		0(,,)
in_transaction		
entry_id		id

•

**2.0.1.11 backend\_variables**

- backend\_variables
- 
- 

backend_conn_id	true	id
variable_name	true	
variable_value		
variable_type		(sys/user)

•

```

- tx_read_only, (mysql8.0)
- transaction_read_only, (mysql5.7)
- character_set_client,
- collation_connection,
- character_set_results,
- tx_isolation_level, (mysql8.0)
- transaction_isolation, (mysql5.7)
- autocommit,
-

```

**2.0.1.12 backend\_connections**

- backend\_connections
- 
- 

backend_conn_id	true	id
db_group_name		db
db_instance_name		db
remote_addr		
remote_port		
remote_processlist_id		mysqld
local_port		
processor_id		id
user		
schema		schema? (/)
session_conn_id		id,
sql		sql(10241024)
sql_execute_time		sql20ms
mark_as_expired_timestamp		,,
conn_net_in		
conn_net_out		
conn_estab_time		0
borrowed_from_pool		0
conn_recv_buffer		0(,,)

conn_send_task_queue		0()
used_for_heartbeat		
conn_closing		
xa_status		xa
in_transaction		

•

**2.0.1.13 dble\_table****2.0.1.13.0 dble\_table**

- dble\_table
- table
- 

id	true	tableCschemaMySqltableM
name		
schema		schema
max_limit		
type		global/single/sharding/child/no sharding

- nameschema
- 

**2.0.1.13.1 dble\_global\_table**

- dble\_global\_table
- 
- 

id	true	dble_tableid
check		
checkClass		class
cron		

•

**2.0.1.13.2 dble\_sharding\_table**

- dble\_sharding\_table
- 
- 

id	true	dble_tableid
increment_column		
sharding_column		
sql_required_sharding		sqlRequiredSharding
algorithm_name		

•

**2.0.1.13.3 dble\_child\_table**

- dble\_child\_table
- 
- 

id	true	dble_tableid
parent_id		id
increment_column		
join_column		
parent_column		

•

**2.0.1.13.4 dble\_table\_sharding\_node**

- dble\_table\_sharding\_node
- sharding\_node
- 

id	true	dble_tableid
sharding_node	true	
order		sharding_node(0)

•

**2.0.1.14 dble\_algorithm**

- dble\_algorithm
- 
- 

name	true	
key	true	
value		
is_file		map filetruefilefalsecontentmap file1024

•

**2.0.1.15 dble\_entry****2.0.1.15.0 dble\_entry**

- dble\_entry

- (+)

id	true	
type		(username/conn_attr),
user_type		//sharding
username		
password_encrypt		
encrypt_configured		
conn_attr_key		tenant
conn_attr_value		
white_ips		
readonly		-
max_conn_count		
blacklist		

- 

#### 2.0.1.15.1 dble\_entry\_schema

- dble\_entry\_schema
- schema

- 

id	true	dble_entryid
schema	true	schema

- 

#### 2.0.1.15.2 dble\_rw\_split\_entry

- dble\_rw\_split\_entry
- schema

- 

id	true	
type		(username/conn_attr),
username		
password_encrypt		
encrypt_configured		
conn_attr_key		tenant
conn_attr_value		
white_ips		
max_conn_count		
blacklist		
db_group		db_group

- usernameconn\_attr\_keyconn\_attr\_value

- 

#### 2.0.1.15.3 dble\_entry\_table\_privilege

- dble\_entry\_table\_privilege
- privilege

- 

id	true	dble_entryid
schema	true	schema
table	true	table
exist_metas		tableable
insert		insert
update		update
select		select
delete		delete
is_effective		

- 

#### 2.0.1.16 dble\_blacklist

- dble\_blacklist
- 
- 

name	true	
property_key	true	key
property_value		value
user_configured		

- 

#### 2.0.1.17 processlist

- processlist
- (NULL)
- 

front_id	true	id
sharding_node		sharding node
db_instance		name

mysql_id	true	mysql id
user		
front_host		
mysql_db		mysql 'show processlist' db
command		mysql mysql 'show processlist' command
time		mysqlstate mysql 'show processlist' time
state		mysql mysql 'show processlist' state
info		mysql mysql 'show processlist' info

- (show @@processlist)

### 2.0.1.18 dble\_thread\_usage

- dble\_thread\_usage
- 
- 

thread_name	true	
last_quarter_min		15s
last_minute		1min
last_five_minute		5min

- (show @@thread\_used;)

### 2.0.1.19 dble\_reload\_status

- dble\_reload\_status
- reload
- 

index	true	reload[ RL ]
cluster		dble
reload_type		reload reload_metadata/reload_all/manager_insert/manager_update/mamager_delete
reload_status		reloadnot_reloading/self_reload/meta_reload/waiting_others
last_reload_start		
last_reload_end		
trigger_type		local_command/cluster_notify
end_type		

- (show @@reload\_status)

### 2.0.1.20 dble\_xa\_session

- dble\_xa\_session
- xa
- 

front_id	true	id
xa_id		xaid
xa_state		xa
sharding_node	true	xasharding_node,

- (show @@session.xa)

### 2.0.1.21 dble\_ddl\_lock

- dble\_ddl\_lock
- dbleddl
- 

schema	true	schema
table	true	table
sql		ddl sql

- (show @@ddl)

## 2.0.2 INSERT/UPDATE/DELETE&

### 2.0.2.0 INSERT Syntax

```

INSERT
  [INTO] tbl_name
  [(col_name [, col_name] ...)]
  {VALUES | VALUE} (value_list) [, (value_list)] ...
  
```

```

INSERT
  [INTO] tbl_name
  SET assignment_list
  
```

- INSERT ...SELECT ,LOW\_PRIORITY,DELAYED,HIGH\_PRIORITY,IGNORE,ON DUPLICATE KEY UPDATE,PARTITION

### 2.0.2.1 UPDATE Syntax

```

UPDATE table_reference
  SET assignment_list
  WHERE where_condition
  value:
    {expr | DEFAULT}
  assignment:
    col_name = value
  assignment_list:
    assignment [, assignment]
  
```

- 
- 
-

- LOW\_PRIORITYIGNOREORDER BYLIMIT ,PARTITION
- where
- 

### 2.0.2.2 DELETE Syntax

```
DELETE FROM tbl_name WHERE where_condition
```

- 
- 
- LOW\_PRIORITYIGNOREORDER BYLIMIT ,PARTITION
- where
- 

### 2.0.2.3 INSERT/UPDATE/DELETE

#### 2.0.2.3.0 dble\_db\_group

- active

#### 2.0.2.3.1 dble\_db\_instance

- active\_conn\_countidle\_conn\_countread\_conn\_requestwrite\_conn\_requestlast\_heartbeat\_ack\_timestamplast\_heartbeat\_ackheartbeat\_statusheartbeat\_failure\_in\_last\_5min

#### 2.0.2.3.2 dble\_rw\_split\_entry

- idblacklisttype

## 2.1

- [2.1.1 select](#)
- [2.1.2 set](#)
- [2.1.3 show](#)
- [2.1.4 switch](#)
- [2.1.5 kill](#)
- [2.1.6 stop](#)
- [2.1.7 reload](#)
- [2.1.8 rollback](#)
- [2.1.9 offline](#)
- [2.1.10 online](#)
- [2.1.11 file](#)
- [2.1.12 log](#)
- [2.1.13](#)
- [2.1.14 pause & resume](#)
- [2.1.15](#)
- [2.1.16 create database](#)
- [2.1.17 check @@metadata](#)
- [2.1.18 release @@reload\\_metadata](#)
- [2.1.19 split](#)
- [2.1.20 flow\\_control](#)
- [2.1.21](#)

**2.1.1 select****2.1.1.1 select @@VERSION\_COMMENT**

```
select @@VERSION_COMMENT;
```

db1e

```
MySQL [(none)]> select @@VERSION_COMMENT;
+-----+
| @@VERSION_COMMENT      |
+-----+
| db1e Server (ActionTech) |
+-----+
1 row in set (0.02 sec)
```

**2.1.1.2 select @@SESSION.TX\_READ\_ONLY / select @@SESSION.Transaction\_READ\_ONLY**

```
select @@SESSION.TX_READ_ONLY;
```

```
select @@SESSION.Transaction_READ_ONLY
```

readonly

**2.1.1.3 select @@max\_allowed\_packet**

```
select @@max_allowed_packet;
```

1048576

**2.1.1.4 select TIMEDIFF(NOW(), UTC\_TIMESTAMP())**

```
select TIMEDIFF(NOW(), UTC_TIMESTAMP())
```

00:00:00

### 2.1.2 set xxx

**set xxx;**

xxx

OK

**2.1.3 show****2.1.3.1 show @@time.current**

```
show @@time.current;
```

**2.1.3.2 show @@time.startup**

```
show @@time.startup;
```

**2.1.3.3 show @@version**

```
show @@version;
```

dble

**2.1.3.4 show @@server**

```
show @@server;
```

dble

```
mysql> show @@server;
+-----+-----+-----+-----+-----+-----+-----+
| UPTIME | USED_MEMORY | TOTAL_MEMORY | MAX_MEMORY | RELOAD_TIME | ROLLBACK_TIME | CHARSET | STATUS |
+-----+-----+-----+-----+-----+-----+-----+
| 1h 4m 47s | 17414592 | 87031808 | 1840256880 | 2017/10/17 16:42:09 | -1 | utf8 | ON |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.05 sec)
```

UPTIME:  
USED\_MEMORY:  
TOTAL\_MEMORY:  
MAX\_MEMORY:  
RELOAD\_TIME: config  
ROLLBACK\_TIME: config rollback  
CHARSET:  
STATUS:

**2.1.3.5 show @@threadpool**

```
show @@threadpool;
```

```
mysql> show @@threadpool;
+-----+-----+-----+-----+-----+-----+
| NAME | POOL_SIZE | ACTIVE_COUNT | TASK_QUEUE_SIZE | COMPLETED_TASK | TOTAL_TASK |
+-----+-----+-----+-----+-----+-----+
| Timer | 1 | 0 | 0 | 22596 | 22596 |
| BusinessExecutor | 8 | 1 | 0 | 216 | 217 |
| complexQueryExecutor | 0 | 0 | 0 | 0 | 0 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.03 sec)
```

NAME:  
POOL\_SIZE:  
ACTIVE\_COUNT:  
TASK\_QUEUE\_SIZE:  
COMPLETED\_TASK:  
TOTAL\_TASK:

**2.1.3.6 show @@database**

```
show @@database;
```

schema

**2.1.3.7 show @@shardingnode**

```
show @@shardingnode;
```

shardingnode

```
mysql> show @@shardingnode;
+-----+-----+-----+-----+-----+-----+-----+
| NAME | DB_GROUP | SCHEMA_EXISTS | ACTIVE | IDLE | SIZE | EXECUTE | RECOVERY_TIME |
+-----+-----+-----+-----+-----+-----+-----+
| dn1 | dh1/db1e_test | true | 0 | 0 | 1000 | 34 | -1 |
| dn2 | dh2/db1e_test | true | 0 | 0 | 1000 | 34 | -1 |
| dn3 | dh1/db2e_test | false | 0 | 0 | 1000 | 26 | -1 |
| dn4 | dh2/db2e_test | true | 0 | 0 | 1000 | 26 | -1 |
| dn5 | dh1/nosharding | true | 0 | 0 | 1000 | 9 | -1 |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.09 sec)
```

NAME:  
DB\_GROUP: dbGroupName/schema  
SCHEMA\_EXISTS: true/false  
ACTIVE:  
IDLE: (bug)  
SIZE: maxCon  
EXECUTE:  
RECOVERY\_TIME: (stop @@heartbeat )

schemashardingnode

```
show @@shardingnode where schema=xxx;
```

xxxschema

**2.1.3.8 show @@dbinstance**

```
show @@dbinstance;
```

dbinstance

```
mysql> show @@dbinstance;
```

```
+-----+-----+-----+-----+-----+-----+-----+
| DB_GROUP | NAME | HOST      | PORT | W/R | ACTIVE | IDLE | SIZE | EXECUTE | READ_LOAD | WRITE_LOAD | DISABLED |
+-----+-----+-----+-----+-----+-----+-----+
| localhost | hostS1 | 10.18x.2x.63 | 3307 | W   | 1 | 9 | 100 | 11 | 0 | 0 | true |
| localhost | hostM1 | 10.18x.2x.64 | 3306 | W   | 1 | 9 | 100 | 17 | 0 | 0 | false |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.09 sec)
```

```
DB_GROUP:dbinstanceDB_GROUP
NAME: dbinstance
HOST: host
PORT:
W/R:
ACTIVE: ,dbinstance
IDLE: ,dbinstance(bug)
SIZE: maxCon
EXECUTE: ,dbinstance
READ_LOAD: ()
WRITE_LOAD: ()
DISABLED: db.xmldbinstance (2.19.09.0disabledtrue)
```

shardingnodedbinstance

```
show @@dbinstance where shardingnode=xxx;
xxxshardingnode
```

**2.1.3.9 show @@dbinstance.synstatus**

```
show @@dbinstance.synstatus;
dbinstance
:heartbeat show slave statusdb.xml
```

```
mysql> show @@dbinstance.synstatus;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| DB_GROUP | NAME | HOST      | PORT | MASTER_HOST | MASTER_PORT | MASTER_USER | SECONDS_BEHIND_MASTER | SLAVE_IO_RUNNING | SLAVE_SQL_RUNNING | SLAVE_IO_STATE | CONNECT_RETRY | LAST_IO_ERROR |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| localhost | hostM1 | 10.18x.2x.63 | 3320 | 10.18x.2x.61 | 3320 | qrep    | NULL | No        | No          | 60           | 60           |
| localhost | hostM2 | 10.18x.2x.64 | 3320 | 10.18x.2x.62 | 3320 | qrep    | NULL | No        | No          | 60           | 60           |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 row in set (0.00 sec)
```

```
DB_GROUP:dbinstanceDB_GROUP
NAME: dbinstance
HOST: /ip
PORT:
```

mysqlshow slave status

**2.1.3.10 show @@dbinstance.syndetail where name=?**

```
show @@dbinstance.syndetail where name=xxx;
xxdbinstance
```

24dbinstance

```
mysql> show @@dbinstance.syndetail WHERE name =hostM2;
+-----+-----+-----+-----+-----+-----+-----+
| DB_GROUP | NAME | HOST      | PORT | MASTER_HOST | MASTER_PORT | MASTER_USER | TIME               | SECONDS_BEHIND_MASTER |
+-----+-----+-----+-----+-----+-----+-----+
| localhost2 | hostM2 | 10.18x.2x.64 | 3320 | 10.18x.2x.62 | 3320 | qrep    | 2017-10-17 18:31:27 | -1 |
| localhost2 | hostM2 | 10.18x.2x.64 | 3320 | 10.18x.2x.62 | 3320 | qrep    | 2017-10-17 18:31:57 | -1 |
| localhost2 | hostM2 | 10.18x.2x.64 | 3320 | 10.18x.2x.62 | 3320 | qrep    | 2017-10-17 18:32:27 | -1 |
| localhost2 | hostM2 | 10.18x.2x.64 | 3320 | 10.18x.2x.62 | 3320 | qrep    | 2017-10-17 18:32:57 | -1 |
+-----+-----+-----+-----+-----+-----+-----+
4 row in set (0.05 sec)
```

:

```
DB_GROUP:dbinstanceDB_GROUP
NAME: dbinstance
HOST: /ip
PORT:
```

mysqlshow slave status

**2.1.3.11 show @@datasource.cluster**

```
show @@datasource.cluster;
2.20.04.0
```

**2.1.3.12 show @@processor**

```
show @@processor;
dbleprocessor
```

```
mysql> show @@processor ;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| NAME   | NET_IN | NET_OUT | REACT_COUNT | R_QUEUE | W_QUEUE | FREE_BUFFER | TOTAL_BUFFER | BU_PERCENT | BU_WARNS | FC_COUNT | BC_COUNT |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Processor0 | 6834 | 2749 | 0 | 0 | 0 | 687194767360 | 687194767360 | 0 | 0 | 0 | 0 |
| Processor1 | 7221 | 2862 | 0 | 0 | 0 | 687194767360 | 687194767360 | 0 | 0 | 0 | 0 |
| Processor2 | 6830 | 31141 | 0 | 0 | 0 | 687194767360 | 687194767360 | 0 | 0 | 1 | 0 |
| Processor3 | 6375 | 2681 | 0 | 0 | 0 | 687194767360 | 687194767360 | 0 | 0 | 0 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.06 sec)
```

```
NAME:
NET_IN:
NET_OUT:
REACT_COUNT: 0
R_QUEUE: 0
W_QUEUE:
FREE_BUFFER: BufferPool_free
TOTAL_BUFFER: BufferPool
BU_PERCENT: BufferPool
BU_WARNS: 0
FC_COUNT:
BC_COUNT:
```

**2.1.3.13 show @@command**

```
show @@command;
processor
```

```
mysql> show @@command;
+-----+-----+-----+-----+-----+
```

```
PROCESSOR: processor  
INIT_DB: COM_INIT_DB  
QUERY: COM_QUERY  
STMT_PREPARE: COM_STMT_P  
STMT_EXECUTE: COM_STMT_E  
STMT_CLOSE: COM_STMT_C  
PING: COM_PING  
KILL: COM_PROCESS_KILL  
QUIT: COM_QUIT  
OTHER:
```

**2.1.3.14 show @@connection where processor=? and front\_id=? and host=? and user=?**

**show @@connection where processor=? and front\_id=? and host=? and user=?**

```
PROCESSOR: PROCESSOR
FRONT_ID: ID
HOST: host
PORT: ()
LOCAL_PORT:
USER:
SCHEMA: schema
CHARACTER_SET_CLIENT:
COLLATION_CONNECTION:
CHARACTER_SET_RESULTS
NET_IN:
NET_OUT:
ALIVE_TIME(S):
RECV_BUFFER: ()
SEND_QUEUE:
TX_ISOLATION_LEVEL:
AUTOCOMMIT:
SYS_VARIABLES:
USER_VARIABLES:
```

### 2.1.3.15 show @@cache

**show @@cache;**

cache

```
mysql> show @@cache;
+-----+-----+-----+-----+-----+-----+
| CACHE          | MAX   | CUR   | ACCESS | HIT    | PUT    | LAST_ACCESS | LAST_PUT |
+-----+-----+-----+-----+-----+-----+
| ER_SQL2PARENTID | 1000  | 0     | 0      | 0      | 0      |             |           |
| SQLRouteCache  | 10000 | 0     | 0      | 0      | 0      |             |           |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.09 sec)
```

```
CACHE: cache  
MAX:  
CUR:  
ACCESS:  
HIT:  
PUT:  
LAST_ACCESS: (yyyy/mm/dd hh:mm:ss)  
LAST_INPUT: (yyyy/mm/dd hh:mm:ss)
```

**2.1.3.16 show @@backend where processor=? and backend\_id=? and mysqlid=? and host=? and port=?**

```
show @@backend where processor=? and backend_id=? and mysqlid=? and host=? and port=?
```

```
show @@session.processorbackend_idmy sqlidhostport
```

```
processor: processor
BACKEND_ID: ID
MYSQLID: mysqlid(show processlistMYSQLID)
HOST:
PORT:
LOCAL_TCP_PORT: tcp
NET_IN:
NET_OUT:
ACTIVE_TIME(S):
CLOSED:
BORROWED:
SEND_QUEUE:
SCHEMA: schema
CHARACTER_SET_CLIENT:
COLLATION_CONNECTION:
CHARACTER_SET_RESULTS:
TX_ISOLATION_LEVEL: -1
AUTOCOMMIT:
SYS_VARIABLES:
USER_VARIABLES:
XA_STATUS: xa
DEAD_TIME
USED_FOR_HEARTBEAT:
```

**2.1.3.17 show @@session**

show @@session;

session

```
mysql> show @@session ;
+-----+-----+
| FRONT_ID | DN_COUNT | DN_LIST
+-----+-----+
| 2 | 2 | MySQLConnection [backendId=59, lastTime=1508233042917 [... ] |
+-----+-----+
1 row in set (0.00 sec)
```

FRONT\_ID: ID  
DN\_COUNT:  
DN\_LIST: ,

DN\_LIST

```
MySQLConnection [backendId=59, lastTime=1508233042917, user=xxxx , schema=dble_test, old schema=dble_test, borrowed=true, fromSlaveDB=false, mysqlId=23201, character_set_client=utf8, character_set_results=utf8, collation_connection=utf8_general_ci, txIsolation=3, autocommit=false, attachment=dn{select * from sharding_two_node LIMIT 100}.0, respHandler=com.actiontech.dble.backend.mysql.nio.handler.MultiNodeQueryHandler@181db802, host=10.18x.2x.64, port=3320, stat usSync=com.actiontech.dble.backend.mysql.nio.MySQLConnection$StatusSync@7da548, writeQueue=0, modifiedSQLExecuted=false] MySQLConnection [id=58, lastTime=1508233042917, user=qrep, schema=dble_test, old schema=dble_test, borrowed=true, fromSlaveDB=false, threadId=11112, character_set_client=utf8, character_set_results=utf8, collation_connection=utf8_general_ci, t xIsolation=3, autocommit=false, attachment=dn{select * from sharding_two_node L IMIT 100}.0, respHandler=com.actiontech.dble.backend.mysql.nio.handler.MultiNodeQueryHandler@181db802, host=10.18x.2x.63, port=3320, statusSync=com.actiontech.d ble.backend.mysql.nio.MySQLConnection$StatusSync@5882b3d, writeQueue=0, modified SQLExecuted=false]
```

:

```
backendId: id
lastTime
user:
schema: schema
old_schema: schema
borrowed:
fromSlaveDB:
mysqlId: idshow processlist
charset:
txIsolation:
autocommit:
attachment:
respHandler: handler
host: host ip
port:
statusSync:
writeQueue:
modifiedSQLExecuted:
```

**2.1.3.18 show @@connection.sql**

show @@connection.sql;

sessionSQL

```
mysql> show @@connection.sql;
+-----+-----+-----+-----+-----+-----+-----+
| FRONT_ID | HOST | USER | SCHEMA | START_TIME | EXECUTE_TIME | SQL | STAGE |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 0:0:0:0:0:0:1 | man | NULL | 2017/10/17 17:00:58 | 139 | show @@connection.sql | Read SQL |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.13 sec)
```

:

FRONT\_ID: ID  
HOST: host  
USER:  
SCHEMA: schema  
START\_TIME:  
EXECUTE\_TIME: SQL20ms  
SQL10241024  
STAGE: ,finished

**2.1.3.19 show @@sql**

show @@sql;

50sql(5)

```
mysql> show @@sql;
+-----+-----+-----+-----+
| ID | USER | START_TIME | EXECUTE_TIME | SQL |
+-----+-----+-----+-----+
| 1 | root | 2017/10/17 17:37:22 | 381 | select * from sharding_two_node LIMIT 100 |
+-----+-----+-----+-----+
1 row in set (0.02 sec)
```

:

ID:  
USER:  
START\_TIME:  
EXECUTE\_TIME:  
SQL

show @@sql true;

**2.1.3.20 show @@sql.high**

show @@sql.high;

sql(10245)

```
mysql> show @@sql.high;
+-----+-----+-----+-----+-----+-----+-----+
| ID | USER | FREQUENCY | AVG_TIME | MAX_TIME | MIN_TIME | EXECUTE_TIME | LAST_TIME | SQL |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | root | 1 | 381 | 381 | 381 | 2017/10/17 17:37:23 | SELECT * FROM sharding_two_node LIMIT ? |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.06 sec)
```

ID:  
USER:  
FREQUENCY: sql  
AVG\_TIME:  
MAX\_TIME:  
MIN\_TIME:  
EXECUTE\_TIME:  
LAST\_TIME:  
SQL

show @@sql.high true;

**2.1.3.21 show @@sql.slow**

show @@sql.slow;

(100reload)sql(10sqlRecordCount5

```
mysql> show @@sql.slow;
+-----+-----+-----+
| USER | START_TIME | EXECUTE_TIME | SQL
+-----+-----+-----+
| root | 2017/10/17 17:37:22 |      381 | select * from sharding_two_node LIMIT 100 |
+-----+-----+-----+
1 row in set (0.07 sec)
```

USER:  
START\_TIME:  
EXECUTE\_TIME:  
SQL

show @@sql.slow true;

**2.1.3.22 show @@sql.resultset**

show @@sql.resultset;

(512KmaxResultSet)sql

```
mysql> show @@sql.resultset;
+-----+-----+-----+-----+
| ID   | USER | FREQUENCY | SQL           | RESULTSET_SIZE |
+-----+-----+-----+-----+
| 1    | root |       1 | SELECT * FROM sharding_two_node | 1048576        |
+-----+-----+-----+-----+
1 row in set (0.05 sec)
```

ID:  
USER:  
FREQUENCY:sql  
SQL:  
RESULTSET\_SIZE:

**2.1.3.23 show @@sql.sum**

show @@sql.sum;

sql,.user.true

```
mysql> show @@sql.sum;
+-----+-----+-----+-----+-----+-----+-----+-----+
| ID   | USER | R   | W   | R%  | MAX  | NET_IN | NET_OUT | TIME_COUNT | TTL_COUNT | LAST_TIME   |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1    | root | 1  | 0  | 1.00 | 1    | 41    | 840   | [0, 0, 1, 0] | [0, 0, 1, 0] | 2017/10/17 17:37:23 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.26 sec)
```

;  
ID:  
USER:  
R:  
W:()  
R%:W0100%  
MAX:  
NET\_IN:  
NET\_OUT:  
TIME\_COUNT:query22-06 ,06-13 ,13-18,18-22  
TTL\_COUNT:query10,10 - 200,1, 1  
LAST\_TIME:SQL

show @@sql.sum true;

**2.1.3.24 show @@sql.sum.user**

show @@sql.sum;

show @@sql.sum.user true;

**2.1.3.25 show @@sql.sum.table**

show @@sql.sum.table;

```
mysql> show @@sql.sum.table;
+-----+-----+-----+-----+-----+-----+
| ID   | TABLE  | R   | W   | R%  | RELATABLE | RELACOUNT | LAST_TIME   |
+-----+-----+-----+-----+-----+-----+
| 1    | sharding_two_node | 1  | 0  | 1.00 | NULL     | NULL      | 2017/10/17 17:37:23 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.06 sec)
```

;  
ID:  
TABLE:(;,bug)  
R:  
W:()  
R%:W0100%  
RELATABLE:(,NULL)  
RELACOUNT:(,NULL)  
LAST\_TIME:SQL

show @@sql.sum.table true;

**2.1.3.26 show @@heartbeat**

show @@heartbeat;

dbinstanceheartbeat

```
mysql> show @@heartbeat;
+-----+-----+-----+-----+-----+-----+-----+
| NAME | HOST | PORT | RS_CODE | RETRY | STATUS | TIMEOUT |
+-----+-----+-----+-----+-----+-----+-----+
| hostM | 10.18x.2x.63 | 3320 | OK | 0 | idle | 0 | 8,8,8 | NULL | false | NULL |
| hostM2 | 10.18x.2x.64 | 3320 | OK | 0 | idle | 0 | 9,9,9 | NULL | false | NULL |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.07 sec)
```

```
NAME:dbGroup
HOST:/IP
PORT:
RS_CODE:INIT, OK, ERROR, TIMEOUT
RETRY:
STATUS:checking/idle
TIMEOUT:(db.xmlheartbeattimeout0)
EXECUTE_TIME:311030
LAST_ACTIVE_TIME:
STOP:stop,stop
RS_MESSAGE:RS_CODEINIT, OK, TIMEOUTmessagenullRS_CODEERRORmessage
```

**2.1.3.27 show @@heartbeat.detail where name=**

show @@heartbeat.detail where name=xxx;

xxdbinstance

dbinstanceheartbeat

shardingNodeHeartbeatPeriod

:

```
mysql> show @@heartbeat.detail where name='hostM1';
+-----+-----+-----+-----+
| NAME | HOST | TIME | EXECUTE_TIME |
+-----+-----+-----+-----+
| hostM | 10.18x.2x.63 | 3320 | 2017-10-17 17:31:58 | 7 |
| hostM | 10.18x.2x.63 | 3320 | 2017-10-17 17:32:59 | 9 |
+-----+-----+-----+-----+
2 row in set (0.00 sec)
```

:

```
NAME:dbGroup
HOST:/IP
PORT:
TIME:
EXECUTE_TIME:()
```

**2.1.3.28 show @@sysparam**

show @@sysparam;

sysconfig

**2.1.3.29 show @@syslog limit=?**

show @@syslog limit=N;

N

:dble.logN

**2.1.3.30 show @@white**

show @@white;

```
mysql> show @@white;
+-----+-----+
| IP | USER |
+-----+-----+
| 0:0:0:0:0:0:0:1 | root |
| 127.0.0.1 | root |
| 0:0:0:0:0:0:0:1 | test |
| 127.0.0.1 | test |
+-----+-----+
4 rows in set (0.00 sec)
```

**2.1.3.31 show @@directmemory**

show @@directmemory;

```
+-----+-----+-----+
| DIRECT_MEMORY_MAXED | DIRECT_MEMORY_POOL_SIZE | DIRECT_MEMORY_POOL_USED |
+-----+-----+-----+
| 3GB | 1024MB | 44KB |
+-----+-----+-----+
1 row in set (0.16 sec)
```

:

```
DIRECT_MEMORY_MAXED:-XX:MaxDirectMemorySize
DIRECT_MEMORY_POOL_SIZE:bufferPoolPageSizebufferPoolPageNumber
DIRECT_MEMORY_POOL_USEDDirectMemory
```

**2.1.3.32 show @@command.count**

show @@command.count;

**2.1.3.33 show @@connection.count**

show @@connection.count;

**2.1.3.34 show @@backend.statistics**

show @@backend.statistics;

```
MySQL [(none)]> show @@backend.statistics;
+-----+-----+-----+
| HOST | PORT | ACTIVE | TOTAL |
+-----+-----+-----+
| 192.168.2.177 | 3307 | 0 | 10 |
| 192.168.2.177 | 3308 | 0 | 10 |
+-----+-----+-----+
2 rows in set (0.02 sec)
```

```
HOSTip  
PORT  
ACTIVE  
TOTAL
```

**2.1.3.35 show @@backend.old**

```
show @@backend.old;
reload @@config_all
show @@backend
```

**2.1.3.36 show @@binlog.status**

```
show @@binlog.status;
sharding.xmlmyqlbinlog
```

```
mysql> show @@binlog.status;
+-----+-----+-----+-----+-----+
| Url | File | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
+-----+-----+-----+-----+-----+
| 10.18x.2x.63:3320 | mysql-bin.000024 | 14128 | | 7ad71aab-de94-11e5-9488-3a935460da28:1-67646 |
| 10.18x.2x.64:3320 | mysql-bin.000049 | 604440 | | ba8f8b5c-debf-11e5-a87b-26b8a61f9012:1-91 |
+-----+-----+-----+-----+-----+
2 rows in set (0.11 sec)
```

```
Url: Url
show master status
```

**2.1.3.37 show @@help**

```
show @@help;
```

**2.1.3.38 show @@sql.large**

```
show @@sql.large;
10000sql(10,5)
```

```
mysql> show @@sql.large;
+-----+-----+-----+
| USER | ROWS | START_TIME | EXECUTE_TIME | SQL |
+-----+-----+-----+
| root | 20000 | 2017/10/17 17:37:23 | 381 | SELECT * FROM sharding_two_node LIMIT ? |
+-----+-----+-----+
1 row in set (0.06 sec)
```

```
:
```

```
USER:
ROWS:
START_TIME: EXECUTE_TIME:
SQL
```

```
show @@sql.large true;
```

**2.1.3.39 show @@sql.condition**

```
show @@sql.condition;
reload @@query_cf table&columnsql.100000
select from sharding_two_node where id =0; select from sharding_two_node where id =1;
```

```
mysql> show @@sql.condition;
+-----+-----+-----+
| ID | KEY | VALUE | COUNT |
+-----+-----+-----+
| 2 | sharding_two_node.id | 0 | 1 |
| 3 | sharding_two_node.id | 1 | 2 |
| 2 | sharding_two_node.id.valuekey | size | 2 |
| 3 | sharding_two_node.id.valuecount | total | 3 |
+-----+-----+-----+
4 rows in set (0.05 sec)
```

```
:
```

```
ID:
KEY: schema.table schema.table.valuekey schema.table.valuecount
VALUE: keyvalue
COUNT:
```

**2.1.3.40 show @@cost\_time;**

```
show @@cost_time;
query.bootstrap.cnfuseCostTimeStat
```

```
mysql> show @@cost_time;
+-----+-----+
| OVER_ALL(us) | FRONT_PREPARE | BACKEND_EXECUTE |
+-----+-----+
| 71496 | Id:9,Time:53135;Id:12,Time:54056 | Id:9,Time:16924;Id:12,Time:16006 |
| 15316 | Id:17,Time:2301;Id:11,Time:3196 | Id:17,Time:10691;Id:11,Time:11397 |
+-----+-----+
2 rows in set (0.05 sec)
```

```
:
```

```
OVER_ALL:
FRONT_PREPARE: dble
BACKEND_EXECUTE:
```

**2.1.3.41 show @@shardingNodes where schema=? and table=?;**

show @@shardingNodes

```
mysql> show @@shardingNodes where schema=testdb and table=seqtest;
+-----+-----+-----+-----+-----+
| NAME | SEQUENCE | HOST      | PORT | PHYSICAL_SCHEMA | USER | PASSWORD |
+-----+-----+-----+-----+-----+
| dn1  | 0       | 10.186.24.113 | 3309 | db1           | root | 123456  |
| dn2  | 1       | 10.186.24.113 | 3309 | db2           | root | 123456  |
+-----+-----+-----+-----+-----+
2 rows in set (0.05 sec)
```

NAME:  
SEQUENCE:  
HOST:IP  
PORT  
PHYSICAL\_SCHEMA  
USER  
PASSWORD

**2.1.3.42 show @@algorithm where schema=? and table=?;**

show @@algorithm

```
mysql> show @@algorithm where schema=testdb and table=seqtest;
+-----+-----+
| KEY    | VALUE   |
+-----+-----+
| TYPE   | SHARDING TABLE |
| COLUMN | ID      |
| CLASS  | com.actiontech.dble.route.function.PartitionByLong |
| partitionCount | 2          |
| partitionLength | 1          |
+-----+-----+
5 rows in set (0.05 sec)
```

KEY:  
VALUE

**2.1.3.43 show @@thread\_used;**

show @@thread\_used;

```
mysql> show @@thread_used;
+-----+-----+-----+-----+
| THREAD_NAME        | LAST_QUARTER_MIN | LAST_MINUTE | LAST_FIVE_MINUTE |
+-----+-----+-----+-----+
| BusinessExecutor3 | 0%              | 0%          | 0%            |
| $_NIO_REACTOR_BACKEND-2 | 0%              | 0%          | 0%            |
| BusinessExecutor1 | 0%              | 0%          | 0%            |
| $_NIO_REACTOR_BACKEND-3 | 0%              | 0%          | 0%            |
| $_NIO_REACTOR_BACKEND-0 | 0%              | 0%          | 0%            |
| $_NIO_REACTOR_FRONT-0 | 0%              | 0%          | 0%            |
| $_NIO_REACTOR_BACKEND-1 | 0%              | 0%          | 0%            |
+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

THREAD\_NAME:  
LAST\_QUARTER\_MIN15  
LAST\_MINUTE  
LAST\_FIVE\_MINUTE

**2.1.3.44 show @@ddl;**

show @@ddl;

dbleDDL

```
mysql> show @@ddl;
+-----+-----+
| Schema | Table      |
+-----+-----+
| testdb | sharding_two_node |
| mytest | sharding_four_node |
+-----+-----+
2 rows in set (0.00 sec)
```

Schema:Schema  
TableTable  
Sqlddl sql

**2.1.3.45 show @@processlist;**

show @@processlist;

NULL

```
mysql> show @@processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Front_Id | shardingNode | MysqlId | User | Front_Host | db | Command | Time | State | Info |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | dn2      | 2903 | root | 127.0.0.1:33222 | db2 | Sleep   | 17 |  | NULL |
| 2 | NULL     | NULL | man1 | 127.0.0.1:34882 | NULL | NULL    | 0 |  | NULL |
| 3 | dn3      | 2259 | root | 127.0.0.1:33226 | db1 | Sleep   | 4 |  | NULL |
| 3 | dn2      | 2308 | root | 127.0.0.1:33226 | db2 | Sleep   | 4 |  | NULL |
| 3 | dn1      | 2304 | root | 127.0.0.1:33226 | db1 | Sleep   | 4 |  | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.05 sec)
```

Front\_IdID  
shardingNode shardingNode  
MysqlId mysql ID  
User  
Front\_Host  
db mysql 'show processlist' db  
Commandmysql mysql 'show processlist' Command  
Timemysqlstate mysql 'show processlist' Time  
Statemysql mysql 'show processlist' State  
Infomysql mysql 'show processlist' Info

**2.1.3.46 show @@session.xa;**

show @@session.xa;

xa

```
mysql> show @@session.xa;
+-----+-----+-----+
| FRONT_ID | XA_ID | XA_STATE | SHARDING_NODES |
+-----+-----+-----+
| 1 | 'Dble_Server.1.1' | TX_COMMIT_FAILED_STATE | dn1,dn3 |
+-----+-----+-----+
1 rows in set (0.00 sec)
```

:

```
FRONT_IDID
XA_IDxaid
XA_STATExa
SHARDING_NODEsxashardingNode
```

**2.1.3.47 show @@reload\_status**

show @@reload\_status

dblereload

```
+-----+-----+-----+-----+-----+-----+
| INDEX | CLUSTER | RELOAD_TYPE | RELOAD_STATUS | LAST_RELOAD_START | LAST_RELOAD_END | TRIGGER_TYPE | END_TYPE |
+-----+-----+-----+-----+-----+-----+
| 0 | No Cluster | RELOAD_ALL | NOT_RELOADING | 2020/06/19 14:28:04 | 2020/06/19 14:28:05 | LOCAL_COMMAND | RELOAD_END |
+-----+-----+-----+-----+-----+-----+
```

:

```
INDEX:reload[RL]
CLUSTER:dble
RELOAD_TYPE:reload RELOAD_ALL/RELOAD_META/MANAGER_INSERT/MANAGER_UPDATE/MANAGER_DELETE
RELOAD_STATUS:reloadnot_reloading/self_reload/meta_reload/waiting_others
LAST_RELOAD_START
LAST_RELOAD_END
TRIGGER_TYPE LOCAL_COMMAND/CLUSTER_NOTIFY
END_TYPE RELOAD_END/INTERRUPTED
```

release @@reload\_metadata

**2.1.3.48 show @@user**

show @@user

dble

```
mysql> show @@user;
+-----+-----+-----+
| Username | Manager | Readonly | Max_con |
+-----+-----+-----+
| man1 | Y | N | no limit |
| root | N | N | no limit |
| user | N | N | no limit |
+-----+-----+-----+
3 rows in set (0.03 sec)
```

:

```
Username
Manager
Readonly
Max_con
```

**2.1.3.49 show @@user.privilege**

show @@user.privilege

dble

```
mysql> show @@user.privilege;
+-----+-----+-----+-----+-----+
| Username | Schema | Table | INSERT | UPDATE | SELECT | DELETE |
+-----+-----+-----+-----+-----+
| root | testdb1 | * | Y | Y | Y | Y |
| root | testdb | * | Y | Y | Y | Y |
| user | testdb | * | N | Y | Y | N |
+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

:

```
Username
Schema
Tabledml, *
INSERT
UPDATE
SELECT
DELETE
```

**2.1.3.50 show @@data\_distribution where table ='schema.table'**

show @@data\_distribution where table ='schema.table'

```
+-----+-----+
| SHARDING_NODE | COUNT |
+-----+-----+
| dn1 | 100 |
| dn2 | 101 |
| dn3 | 98 |
| dn4 | 104 |
+-----+-----+
4 rows in set (0.09 sec)
```

:

```
SHARDING_NODE
COUNT
```

**2.1.3.51 show @@Questions**

show @@Questions

SQLQUERYTransaction

```
mysql> show @@Questions;
+-----+-----+
| Questions | Transactions |
+-----+-----+
|      0 |          0 |
+-----+-----+
```

:  
Questions:  
Transactions

**2.1.3.52 show @@connection\_pool**

show @@connection\_pool

```
mysql> show @@connection_pool;
+-----+-----+-----+-----+
| DB_GROUP | DB_INSTANCE | PROPERTY | VALUE |
+-----+-----+-----+-----+
| dbGroup1 | instanceM1 | minCon | 2 |
| dbGroup1 | instanceM1 | maxCon | 4 |
| dbGroup1 | instanceM1 | testOnCreate | false |
| dbGroup1 | instanceM1 | testOnBorrow | false |
| dbGroup1 | instanceM1 | testOnReturn | false |
| dbGroup1 | instanceM1 | testWhileIdle | false |
| dbGroup1 | instanceM1 | connectionHeartbeatTimeout | 20 |
| dbGroup1 | instanceM1 | connectionTimeout | 10000 |
| dbGroup1 | instanceM1 | heartbeatPeriodMillis | 10000 |
| dbGroup1 | instanceM1 | idleTimeout | 600000 |
| dbGroup1 | instanceM1 | evictorShutdownTimeoutMillis | 10000 |
| dbGroup1 | instanceM1 | timeBetweenEvictionRunsMillis | 30000 |
+-----+-----+-----+-----+
12 rows in set (0.01 sec)
```

:  
DB\_GROUPdbinstanceDB\_GROUP  
DB\_INSTANCEdbinstance  
PROPERTY  
VALUE

**2.1.3.52 show @@cap\_client\_found\_rows**

show @@cap\_client\_found\_rows

cap\_client\_found\_rows

```
mysql> show @@cap_client_found_rows;
+-----+
| @@cap_client_found_rows |
+-----+
| 0 |
+-----+
1 row in set (0.02 sec)
```

:  
0-1-

#### **2.1.4 switch**

2.20.04.0

**2.1.5 kill****2.1.5.1 kill @@connection**

```
kill @@connection id1,id2,...;
```

```
idx idshow @@connection
```

OK

**2.1.5.2 kill @@xa\_session;**

```
kill @@xa_session id1,id2,...;
```

```
idx session idshow @@session.xa
```

```
sessionxasession
```

OKsession

**2.1.5.3 KILL @@DDL\_LOCK where schema=? and table=?**

```
KILL @@DDL_LOCK where schema=? and table=?;
```

```
schematableddl 2.22 KILL @@DDL_LOCK
```

OK

## 2.1.6 stop

### 2.1.6.1 stop @@heartbeat

stop @@heartbeat keys:dbGroup

keys:dbGroup key dbGroup      key:dbGroupdbGroup\$0-n dbGroup\$0-2dbGroup[0],dbGroup[1],dbGroup[2]BUG

value:

dbGroupkeyhostheartbeat n

OK

**2.1.7 reload****2.1.7.1 reload @@config**

```
reload @@config;
2.19.09.0()2.19.09.0reload @@config_all
```

**2.1.7.2 reload @@config\_all**

```
reload @@config_all [-s] [-f] [-r];
user.xmldb.xmlsharding.xml
-s ERROR
-f dbGroup-rdbGroup,
-r,
2.19 reload\_all
OK ERROR
metameta
•
•
• shardingNode
• shardingNodedbGroup/dbInstance
• schema
• schema
• schema shardingNode
• schemashardingNodedbGroup/dbInstance
-rmeta
-r-s,metadataadbGroup/dbInstance
.metadata #1002
shardingNodeschemaglobal
(systemouterHAFalse), ... 2.12
```

**2.1.7.3 reload @@metadata**

```
reload @@metadata;
```

OK

```
reload @@metadata where schema=? [ and table=? ]
```

```
schema
OK
reload @@metadata where table in ('schema1.table1','schema2.table2','schema1.table3',...
schema1table1,table3schema2table2
OK
```

**2.1.7.4 reload @@sqlslow=N;**

```
reload @@sqlslow=N;
slow sqlN OK
```

**2.1.7.5 reload @@user\_stat**

```
reload @@user_stat;
```

```
:
show @@sql;
show @@sql.sum;
show @@sql.slow;
show @@sql.high;
show @@sql.large;
show @@sql.resultset;
OK
```

**2.1.7.6 reload @@query\_cf**

```
reload @@query_cf=table&column;
tablecolumn
show @@sql.conditiont
OK

reload @@query_cf;
reload @@query_cf=NULL;
```

## 2.1.8 rollback

### 2.1.8.1 rollback @@config

```
rollback @@config;  
reload @@config reload @@config_all,  
OKERROR
```

### 2.1.9 offline

**offline;**

dble, dblepingselect user

OK

### 2.1.10 online

**online**

dbleoffline

OK

**2.1.11 file****2.1.11.1 file @@list**

conf

```
MySQL [(none)]> file @@list;
+-----+
| DATA
+-----+
| 1 : wrapper.conf  time:2019-10-17 11:13 |
| 2 : myid.properties time:2019-10-17 11:11 |
+-----+
2 rows in set (0.00 sec)
```

**2.1.11.2 file @@show filename**

```
filename
filename conf/filename
```

**2.1.11.3 file @@upload filename content**

```
contentfileNamefileNamexml
```

### 2.1.12 log

```
log @@[file=logfile limit=numberOfRow key=keyword regex=regex]
```

logsfilename

- logfile.log
- rowLimit:10000
- keyword :key, key'key word'
- regexStr : regexStr's

logsfilename

## 2.1.13 dryrun

reloaddryrun

sharding.xml reloaddryrun sharding.xml:

```
<?xml version="1.0"?>
<!DOCTYPE dble:sharding SYSTEM "sharding.dtd">
<db:sharding xmlns:dble="http://dble.cloud/">
  <schema name="testdb" sqlMaxLimit="100" >
    <shardingTable name="sharding_two_node" shardingNode="dn1,dn2" shardingColumn="id" function="two-long" />
    <shardingTable name="sharding_two_node2" shardingNode="dn1,dn2" shardingColumn="id" function="two-long" />
    <shardingTable name="sharding_two_node3" shardingNode="dn1,dn2" shardingColumn="id" function="two-long" />
    <shardingTable name="sharding_four_node" shardingNode="dn1,dn2,dn3,dn4" shardingColumn="id" function="rule_simple" />
    <globalTable name="test_table" shardingNode="dn1-2"/>
    <shardingTable name="a_test" shardingNode="dn1,dn2,dn3,dn4" shardingColumn="id" function="rule_simple" />
    <shardingTable name="a_order" shardingNode="dn1,dn2,dn3,dn4" shardingColumn="id" function="rule_simple" />
    <shardingTable name="test_shard" shardingNode="dn1,dn2,dn3,dn4" shardingColumn="id" function="rule_simple" />
    <globalTable name="test_global" shardingNode="dn1,dn2,dn3,dn4"/>
    <shardingTable name="sbtest1" shardingNode="dn1,dn2,dn3,dn4" shardingColumn="id" function="rule_simple" />
  </schema>
  <schema name="nosharding_test" sqlMaxLimit="100" shardingNode="dn5">
  </schema>
  <shardingNode name="dn1" dbGroup="dh1" database="ares_test" />
  <shardingNode name="dn2" dbGroup="dh2" database="db1e_test" />
  <shardingNode name="dn3" dbGroup="dh1" database="mycat_test" />
  <shardingNode name="dn4" dbGroup="dh2" database="mycat_test" />
  <shardingNode name="dn5" dbGroup="dh1" database="nosharding" />
  <shardingNode name="dn8" dbGroup="dh1" database="xxxxxxx" />
</db:sharding>
<function name="rule_simple" class="Hash">
  <property name="partitionCount">4</property>
  <property name="partitionLength">1</property>
</function>
</db:sharding>
<function class="Hash" name="two-long">
  <property name="partitionCount">2</property>
  <property name="partitionLength">1</property>
</function>
```

dryrun

```
mysql> dryrun;
+-----+-----+
| TYPE | LEVEL | DETAIL
+-----+-----+
| Xml | WARNING | shardingNode dn9 is useless |
| Xml | WARNING | shardingNode dn8 is useless |
+-----+-----+
2 rows in set (0.58 sec)
```

TYPE: XML

BACKEND:WARNING

DETAIL :

**2.1.14 shardingNode**

shardingNodedbGroupshardingNodeshardingNode

**2.1.14.0**

```
pausedblequeuewait_limitresume,
pausepausetimeoutsqldblesql
pausedble servershardingNodereoloadble
reload
```

**2.1.14.1**

```
pause @@shardingNode = 'dn1,dn2' and timeout = 10 ,queue = 10,wait_limit = 10;
:
timeout:timeoutpause
queue:
wait_limit:wait_limit
```

**2.1.14.2**

RESUME; OK "No shardingNode paused"

**2.1.14.3**

show @@pause;

```
mysql> show @@pause;
+-----+
| PAUSE_SHARDING_NODE |
+-----+
| dn1                 |
| dn2                 |
+-----+
2 rows in set (0.15 sec)
```

**2.1.15**

:

**2.1.15.1**

```
mysql> show @@slow_query_log;
+-----+
| @@slow_query_log |
+-----+
| 0 |
+-----+
1 row in set (0.00 sec)
```

**2.1.15.2**

```
mysql> enable @@slow_query_log;
Query OK, 1 row affected (0.09 sec)
enable slow_query_log success
```

**2.1.15.3**

```
mysql> disable @@slow_query_log;
Query OK, 1 row affected (0.03 sec)
disable slow_query_log success
```

**2.1.15.4**

```
mysql> show @@slow_query.time;
+-----+
| @@slow_query.time |
+-----+
| 100 |
+-----+
1 row in set (0.00 sec)
```

**2.1.15.5**

```
mysql> reload @@slow_query.time=200;
Query OK, 1 row affected (0.10 sec)
reload @@slow_query.time success

mysql> show @@slow_query.time;
+-----+
| @@slow_query.time |
+-----+
| 200 |
+-----+
1 row in set (0.00 sec)
```

**2.1.15.6**

```
mysql> show @@slow_query.flushperiod;
+-----+
| @@slow_query.flushperiod |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)
```

**2.1.15.7**

```
mysql> reload @@slow_query.flushperiod=2;
Query OK, 1 row affected (0.05 sec)
reload @@slow_query.flushPeriod success

mysql> show @@slow_query.flushperiod;
+-----+
| @@slow_query.flushperiod |
+-----+
| 2 |
+-----+
1 row in set (0.00 sec)
```

**2.1.15.8**

```
mysql> show @@slow_query.flushsize;
+-----+
| @@slow_query.flushsize |
+-----+
| 1000 |
+-----+
1 row in set (0.01 sec)
```

**2.1.15.9**

```
mysql> reload @@slow_query.flushsize=1100;
Query OK, 1 row affected (0.03 sec)
reload @@slow_query.flushSize success

mysql> show @@slow_query.flushsize;
+-----+
| @@slow_query.flushsize |
+-----+
| 1100 |
+-----+
1 row in set (0.00 sec)
```

**2.1.15.10**

```
show @@connection.sql.status where FRONT_ID=?; query      trace queryquery
```

```
mysql> show @@connection.sql.status where FRONT_ID= 1;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF |
+-----+-----+-----+-----+-----+
|       Read_SQL | 0.0 | 0.082598 | 0.082598 | - | - |
|     Parse_SQL | 0.082598 | 0.676424 | 0.593826 | - | - |
| Route_Calculation | 0.676424 | 0.895382 | 0.218958 | - | - |
| Prepare_to_Push/Optimize | 0.895382 | 6743.838628 | 6742.943246 | - | - |
| Execute_SQL | 6743.838628 | 6751.478422 | 9.649794 | dn1 | select * from sharding_4_t1 |
| Execute_SQL | 6743.838628 | 6751.472835 | 7.634207 | dn3 | select * from sharding_4_t1 |
| Execute_SQL | 6743.838628 | 6750.981646 | 7.143018 | dn4 | select * from sharding_4_t1 |
| Execute_SQL | 6743.838628 | 6753.31394 | 9.475312 | dn2 | select * from sharding_4_t1 |
| Fetch_result | 6753.488422 | 6754.383316 | 0.894894 | dn1 | select * from sharding_4_t1 |
| Fetch_result | 6751.472835 | 6751.656604 | 0.183769 | dn3 | select * from sharding_4_t1 |
| Fetch_result | 6750.981646 | 6751.188385 | 0.206739 | dn4 | select * from sharding_4_t1 |
| Fetch_result | 6753.31394 | 6754.286055 | 0.972115 | dn2 | select * from sharding_4_t1 |
| Write_to_Client | 6750.981646 | unfinished | unknown | - | - |
+-----+-----+-----+-----+-----+
13 rows in set (0.04 sec)
```

join

```
mysql> show @@connection.sql.status where FRONT_ID= 1;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF |
+-----+-----+-----+-----+-----+
|       Read_SQL | 0.0 | 0.039588 | 0.039588 | - | - |
+-----+-----+-----+-----+-----+
```

```

| Parse_SQL | 0.039588 | 0.756578 | 0.71699 | - | - | |
| Route_Calculation | 0.756578 | 1.5547 | 0.798122 | - | - | |
| Prepare_to_Push/Optimize | 1.5547 | 3.428551 | 1.873851 | - | - | |
| Execute_SQL | 3.428551 | 2362.10579 | 2358.677239 | dn1_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Fetch_result | 2362.10579 | unfinished | unknown | dn1_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Execute_SQL | 3.428551 | 2362.122497 | 2358.693856 | dn2_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Fetch_result | 2362.122497 | unfinished | unknown | dn2_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Execute_SQL | 3.428551 | 2362.307153 | 2358.878602 | dn3_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Fetch_result | 2362.307153 | unfinished | unknown | dn3_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Execute_SQL | 3.428551 | 2364.523615 | 2361.095064 | dn4_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
| Fetch_result | 2364.523615 | unfinished | unknown | dn4_0 | select `a`.`age`, `a`.`id`, `a`.`name` from `sharding_4_t1` `a` ORDER BY `a`.`id` ASC |
MERGE_AND_ORDER | 2362.639612 | unfinished | unknown | merge_and_order_1 | dn1_0; dn2_0; dn3_0; dn4_0 |
SHUFFLE_FIELD | 4178.383366 | unfinished | unknown | shuffle_field_1 | merge_and_order_1 |
Execute_SQL | 3.428551 | 2365.71371 | 2362.285159 | dn1_1 | select `b`.`id` from `sharding_2_t1` `b` ORDER BY `b`.`id` ASC |
Fetch_result | 2365.71371 | unfinished | unknown | dn1_1 | select `b`.`id` from `sharding_2_t1` `b` ORDER BY `b`.`id` ASC |
Execute_SQL | 3.428551 | 2365.952707 | 2362.524156 | dn2_1 | select `b`.`id` from `sharding_2_t1` `b` ORDER BY `b`.`id` ASC |
Fetch_result | 2365.952707 | unfinished | unknown | dn2_1 | select `b`.`id` from `sharding_2_t1` `b` ORDER BY `b`.`id` ASC |
MERGE_AND_ORDER | 2366.164823 | unfinished | unknown | merge_and_order_2 | dn1_1; dn2_1 |
SHUFFLE_FIELD | not started | unfinished | unknown | - | - |
JOIN | not started | unfinished | unknown | - | - |
SHUFFLE_FIELD | not started | unfinished | unknown | - | - |
Write_to_Client | not started | unfinished | unknown | - | - |
+-----+-----+-----+-----+-----+-----+
23 rows in set (0.04 sec)

```

**2.1.16 /****2.1.16.1**

dbleshardingNode

create database @@shardingNode ='dn.....'

shardingNode dn\$1-4

shardingNodeshardingNode \$Name does not exists.

shardingNode create database if not exists \$databaseNameOK

**2.1.16.2**

shardingNode

drop database @@shardingNode ='dn.....'

shardingNode dn\$1-4

shardingNodeshardingNode \$Name does not exists.

shardingNode drop database if exists \$databaseNameOKshow @@shardingNodeSCHEMA\_EXISTfalse

## check

### 2.1.17.0 check @@metadata

meta

- check @@metadata
  - reload @@metadata datatime reload @@config\_alldatatime metadatotime
- check full @@metadata ;:
  - where schema=? and table=?
  - where schema=?
  - where reload\_time='yyyy-MM-dd HH:mm:ss', where reload\_time>='yyyy-MM-dd HH:mm:ss', where reload\_time<='yyyy-MM-dd HH:mm:ss'
  - where reload\_time is null
  - where consistent\_in\_sharding\_nodes=0
  - where consistent\_in\_sharding\_nodes = 1
  - where consistent\_in\_memory=0
  - where consistent\_in\_memory = 1
  - If no where, return all results.
- check full @@metadata :

schema	table	reload_time	table_structure	consistent_in_sharding_nodes	consistent_in_memory
schema	table	2018-09-18 11:01:04	CREATE TABLE table`(...	1	1

column table\_structure show create table  
 column consistent\_in\_sharding\_nodes 01  
 column consistent\_in\_memory meta01

table\_structurenulconsistent\_in\_sharding\_nodesconsistent\_in\_memory  
 consistent\_in\_sharding\_nodes0consistent\_in\_memory

### 2.1.17.1 check @@global schema = " [and table = "]

```
mysql> check @@global schema = 'testdb';
+-----+-----+-----+
| SCHEMA | TABLE | DISTINCT_CONSISTENCY_NUMBER | ERROR_NODE_NUMBER |
+-----+-----+-----+-----+
| testdb | tb_global1 | 0 | 0 |
+-----+-----+-----+-----+
```

SCHEMA: SCHEMA A

TABLE: TABLE

DISTINCT\_CONSISTENCY\_NUMBER

ERROR\_NODE\_NUMBER SQL

SQLDISTINCT\_CONSISTENCY\_NUMBER1

## 2.1.18 release

### 2.1.18.1 release @@reload\_metadata

```
reload_metadata reload/reload config_all/reload metadata/rollback configreload metadata  
hangreload metadataable OK ERROR
```

- configurable metareload @@metadatameta
- reloadreload5999SQLCODE“meta”
- dblereload [show @@reload\\_status](#)

## 2.1.19 split

POCdblesqldble 2.19.09.0my sqldump

### dump

1. create database
2. ddldump bigint
3. insertdbledump
4. ddldump
- 5.
- 6.

```
mysql > split src dest [-sschema] [-r500] [-w500] [-l10000] [--ignore]
- srcdump
- destdump
- -sdumpschemaschema
- -r500
- -w500
- -lsplitinsertvalues,4000
- --ignoreinsert
```

-shardingNode-.dump

dump /tmp/mysql\_dump.sql /tmp/dump/

```
split /tmp/mysql_dump.sql /tmp/dump/
```

sp lldb.logsp litlog4j.xml

```
<Configuration status="WARN">
<Appenders>
  <!-- dblelog4j.xmlAppenders -->
  <RollingFile name="DumpFileLog" fileName="logs/dump.log"
    filePattern="logs/${date:yyyy-MM}/dump-%d{MM-dd}-%i.log.gz">
    <PatternLayout>
      <Pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} %5p [%t] (%l) - %m%n</Pattern>
    </PatternLayout>
  <Policies>
    <OnStartupTriggeringPolicy/>
    <SizeBasedTriggeringPolicy size="250 MB"/>
    <TimeBasedTriggeringPolicy/>
  </Policies>
  <DefaultRolloverStrategy max="10"/>
</RollingFile>
</Appenders>
<Loggers>
  <!-- dblelog4j.xmlLoggersleveldebug -->
  <Logger name="dumpfilelog" level="info" additivity="false" includeLocation="false" >
    <AppenderRef ref="DumpFileLog" />
    <AppenderRef ref="RollingFile"/>
  </Logger>
</Loggers>
</Configuration>
```

"dump file has been read d%"

debug

dump file idletimeout kill @@connection id dump file

- 1.
- 2.
3. dump

## 2.1.20 flow\_control

### 2.1.20.1

```
mysql> flow_control @@show;
+-----+-----+-----+
| FLOW_CONTROL_ENABLE | FLOW_CONTROL_START | FLOW_CONTROL_END |
+-----+-----+-----+
| false | 4096 | 256 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

- FLOW\_CONTROL\_ENABLE
- FLOW\_CONTROL\_START
- FLOW\_CONTROL\_END

### 2.1.20.2

flow\_control @@set [enableFlowControl = true/false] [flowControlStart = ?] [flowControlEnd = ?]

```
MySQL [(none)]> @@set enableFlowControl = true flowControlStart= 100 flowControlEND = 30;
Query OK, 0 rows affected (0.02 sec)
```

- enableFlowControl bootstrap.cnfenableFlowControl
- flowControlStartbootstrap.cnfflowControlStartThreshold
- flowControlEndbootstrap.cnfflowControlStopThreshold

### 2.1.20.3

```
MySQL [(none)]> flow_control @@list;
+-----+-----+-----+
| CONNECTION_TYPE | CONNECTION_ID | CONNECTION_INFO | WRITE_QUEUE_SIZE |
+-----+-----+-----+
| ServerConnection | 2 | 127.0.0.1:66772/testdb user = root | 2920 |
+-----+-----+-----+
```

- CONNECTION\_TYPE MySQLConnection/ServerConnection
- CONNECTION\_ID dbleIDID
- CONNECTION\_INFO IPMySQLID
- WRITE\_QUEUE\_SIZE

## 2.1.21

### 2.1.21.1 &

mysql/dble

### 2.1.21.2

**fresh conn [forced] where dbGroup ='groupName' [and dbInstance='instanceName'];**

- forced
- forced
- dbInstancedbGroupdbInstance
- dbInstancedbInstancedbGroupdbInstance

## 2.2

dble cluster.cnf

```
sequenceHandlerType=n
```

sequenceHandlerType

- 1 MySQL offset-step
- 2
- 3
- 4 offset-step

```
/*id*/  
insert into table1(name) values('test');
```

### 2.2.1 MySQL offset-step

MySQL offset-step

1. 24
2. SELECT dble\_seq\_nextval(\seqName\);    **seqName increment**(current\_value, current\_value+increment)
3. 2
- 4.

[1.7.1 offset-step](#)

**2.2.2**

dbleID  
bigint63(63Java0)  
63bits

a.29bits	b.10bits	c.12bits	d.12bits
----------	----------	----------	----------

- a - e
- a4129
- b10instance idbootstrap.cnfinstanceId
- c12
- d4112

1. bootstrap.cnfinstanceId1023
2. 40954095
3. java,411288834974657L(2010)
4. 4169

### 2.2.3

Zookeeper(ID63ID  
PS:63Java0

#### 2.2.3.1

63bits

a.9bits	b.9bits	c.6bits	d.39bits
---------	---------	---------	----------

:

- a - e
- aid9
- b9 id( bootstrap.cnfinstanceIdzookeeper, [1.7.3 time](#))
- c6
- d39(17)

#### 2.2.3.2

cluster.cnfsequenceInstanceByZktrue,      bootstrap.cnfinstanceId([2.2.2](#))

#### 2.2.3.3

cluster.cnfsequenceInstanceByZktruezookeeper,zk, % 32 INSTANCEID.

#### 2.2.4 offset-step

offset-stepzookeeper

1. (`schemaX`:`tableX`)

2. zookeepermin

3. ( [1.7.4 offset-step](#)),max

4. max+1zookeepermin

5.

## 2.3

3.20.10.0dble3.20.10.0

### 2.3.1

#### 2.3.1.1

dble user.xml rwSplitUserdbGroup dbGroup db.xml user.xml user.xml

```
<dble:user xmlns:dble="http://dble.cloud/" version="4.0">
  <managerUser name="man1" password="654321" maxCon="100"/>
  <shardingUser name="root" password="123456" schemas="testdb" readOnly="false" maxCon="20"/>
  <rwSplitUser name="rwsu1" password="123456" dbGroup="rwGroup" maxCon="20"/>
</dble:user>
```

1. user.xml rwSplitUserdbGroup dbGroup db.xml user.xml user.xml
2. dbGroup dbGroup rwSplitUserdbGroup db.xml shardingUser dbGroup schema sharding.xml shardingNode
3. rwSplitUserdbGroup
4. dbGroup instance dbGroup instance

#### 2.3.1.2

db.xml sharding.xml dbGroup db.xml sharding.xml

### 2.3.2

dbledbInstancerwSplitMode0db.xml

1. dbInstance
- 2.

#### 2.3.2.1 dbInstance

dbInstances dbInstancedbInstances show slave status dbledelayThreshold dbInstances delayThreshold=-1

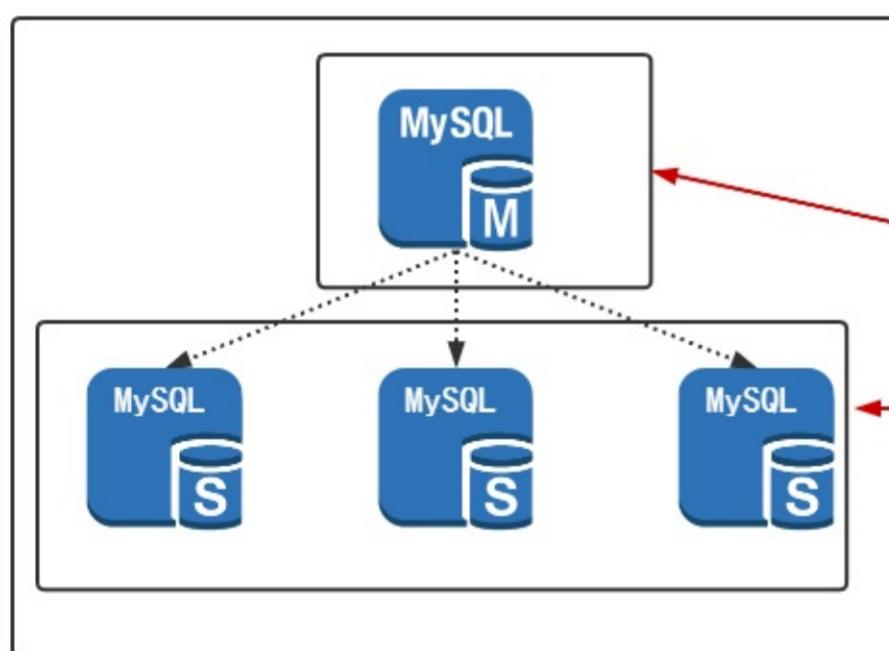
- (primary="true")
  - rwSplitMode2dbInstances
  - (primary primary="false")
    - dbInstances
    - dbInstances
- -

#### 2.3.2.2

dbInstancedbInstance

- dbInstance
- dbInstance
  - dbInstance(readWeight), ,
  - dbInstance,

#### 2.3.2.3 dbGroup rwSplitMode



### 2.3.3

#### 2.3.3.1

1. ddl
2. dml
3. prepared statement
- 4.

#### 2.3.3.2

1. SQL select show

#### 2.3.4

1. druid - set
2. druid - set session transaction read write, isolation level repeatable read
- 3.
4. set transaction read write
5. select
6. select ... into load datadble
- 7.
8. clientdbleschemadble
9. mysqlnddb
10. prepared statament
11. set hint
- 12.

## 2.4 /Hint

Hint, :SQLSQL“”

- 
- dbleinsert...select...

SQLSQLSQLSQLSQLSQLSQLSQL

### 2.4.1 Hint

Hint

1. /\*!dble:type=...\*/
2. /\*#dble:type=...\*/
3. /\* \*/()

```
/*/*#dble: */" for mybatis and /*!*!dble: */" for mysql
```

Hint

type4shardingnodedb\_typesql\_db\_instance\_url

### 2.4.2 shardingnode

- 1.
- shardingnode=node  
node,(node1,2 schema.xml)
- 2.

### 2.4.3 db\_type

- 1.
- db\_type=masterdb\_type=slave
- 2.
- 3.
- delete, insert, replace, update, ddldb\_type=slave

### 2.4.4 sql

- 1.
- sql=sql\_statement
- 2.
- sql\_statementsql

### 2.4.5 db\_instance\_url

- 1.
- db\_instance\_url=ip:port
- 2.
- mysql
- 3.
- delete, insert, replace, update, ddlmysqlread\_only

### 2.4.6

- dbleMySQL, MySQL #1169
- selectSQLdelete/update/insert delete/update/insert SQL
- SQL
- hintDDLreload @@metadata
- hintsession
- SQLSQL select id from tab\_a where id='10000'
- 
- / xxx /(sql)uproxyslavedbmaster

## 2.5

- 2.5.1 XA
- 2.5.2 XA
- 2.5.3 XA
- 2.5.4 XA
- 2.5.5

## 2.5.1 XA

### 2.5.1.1 XA

```

2                               DbleMysqlXAMySQL5.7XAMySQL 5.7dbleXA

1. set autocommit=0;
2. XA set xa=on;
3. SQL
4. commit/rollback;

SQL dble SQL 100

JDBCXAdemo
0

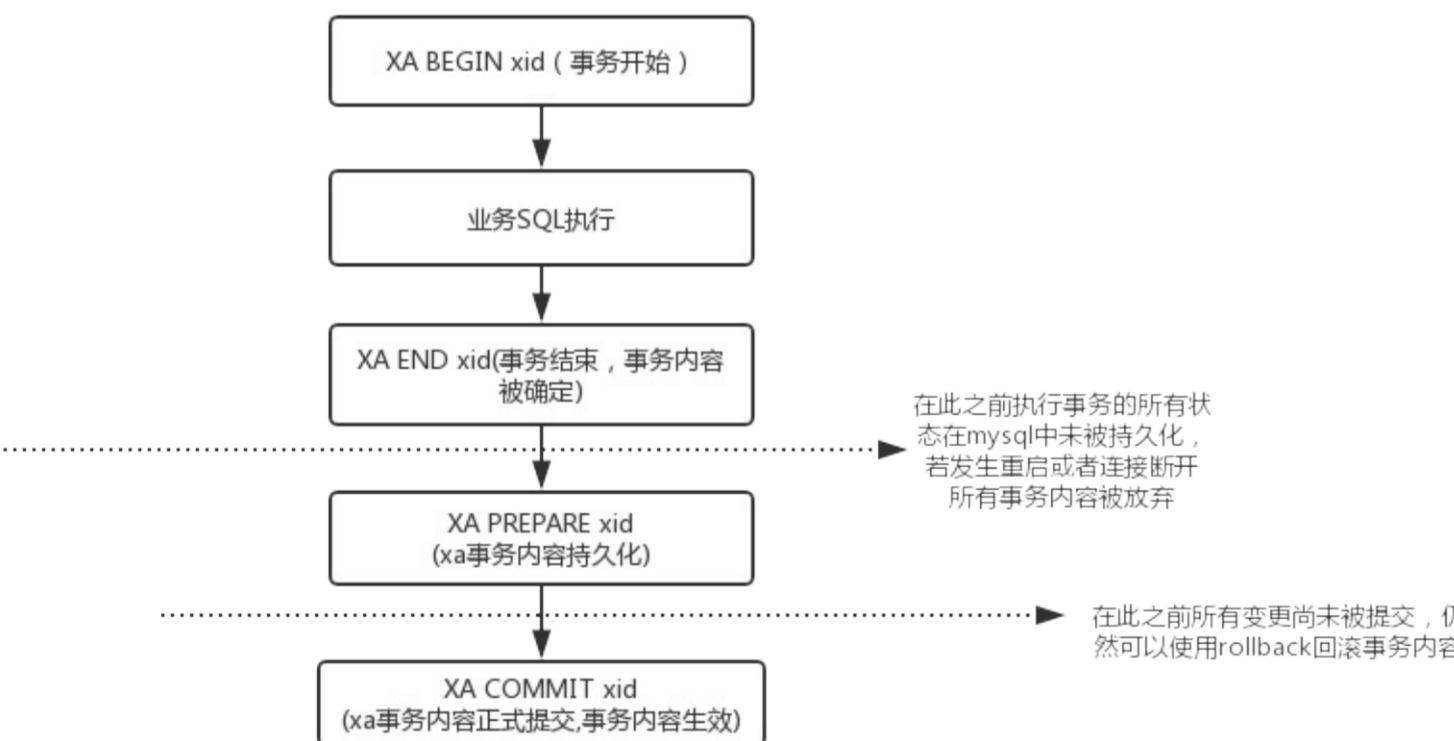
public class XaDemo {
    public static final String URL = "jdbc:mysql://localhost:8066/testdb";
    //jdbc:mysql://127.0.0.1:8066?sessionVariables=xa=1
    //set xa = 1
    public static final String USER = "root";
    public static final String PASSWORD = "123456";

    public static void main(String[] args){
        try {
            //1.
            Class.forName("com.mysql.jdbc.Driver");
            //2.
            Connection conn = DriverManager.getConnection(URL, USER, PASSWORD);
            //3.
            Statement stmt = conn.createStatement();
            stmt.execute("set xa = 1");
            //xa
            stmt.execute("begin");
            try {
                //catch
                //rollback
                stmt.execute("insert into xa_test set id = 11,name = '3333'");
                stmt.execute("insert into xa_test set id = 22,name = '333'");
                stmt.execute("insert into xa_test set id = 3,name = '33'");
                //
                stmt.execute("commit");
            }catch (Exception e){
                System.out.println(" error "+e);
                //
                stmt.execute("rollback");
            }finally {
                stmt.close();
                conn.close();
            }
        }catch(Exception e){
            }
        }
    }
}

```

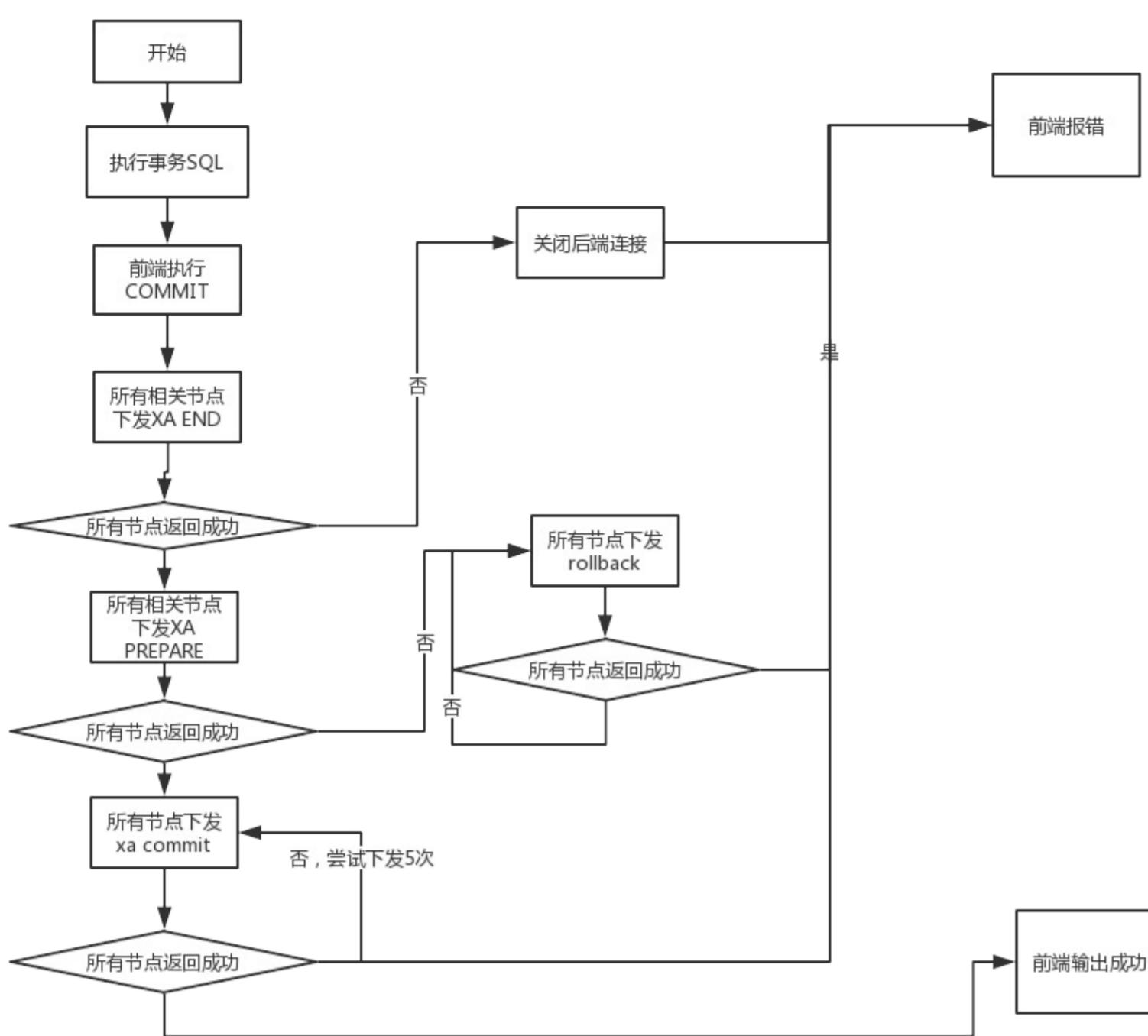
### 2.5.1.2 XA

xamysql5.7xa



**2.5.2.1 XA**

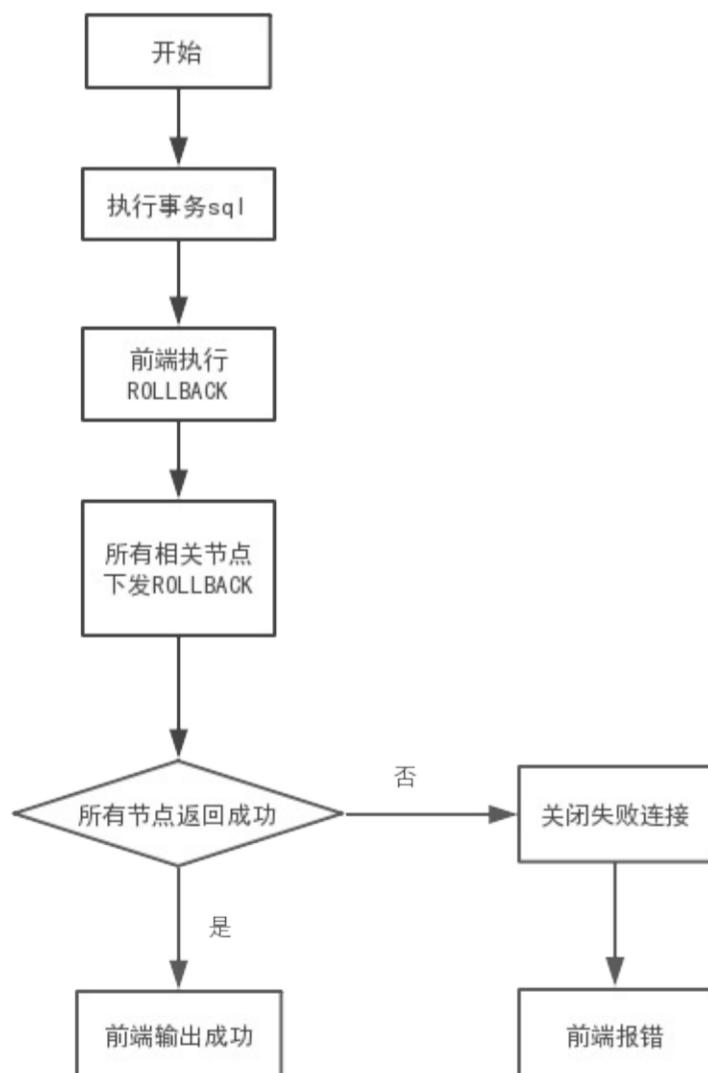
DbleXA



1. XAEND PREPARE COMMIT
2. PREPARE
3. PREPAREROLLBACK
4. COMMIT

**2.5.2.2 XA**

rollback

**2.5.2.3 XA**

2.19.03.0 dble2.19.03.0

**2.5.2.3.1**

2.19.03.0 bootstrap.cnfxaRetryCountxa

1. xaRetryCount 0
2. xaRetryCount 0xaRetryCount

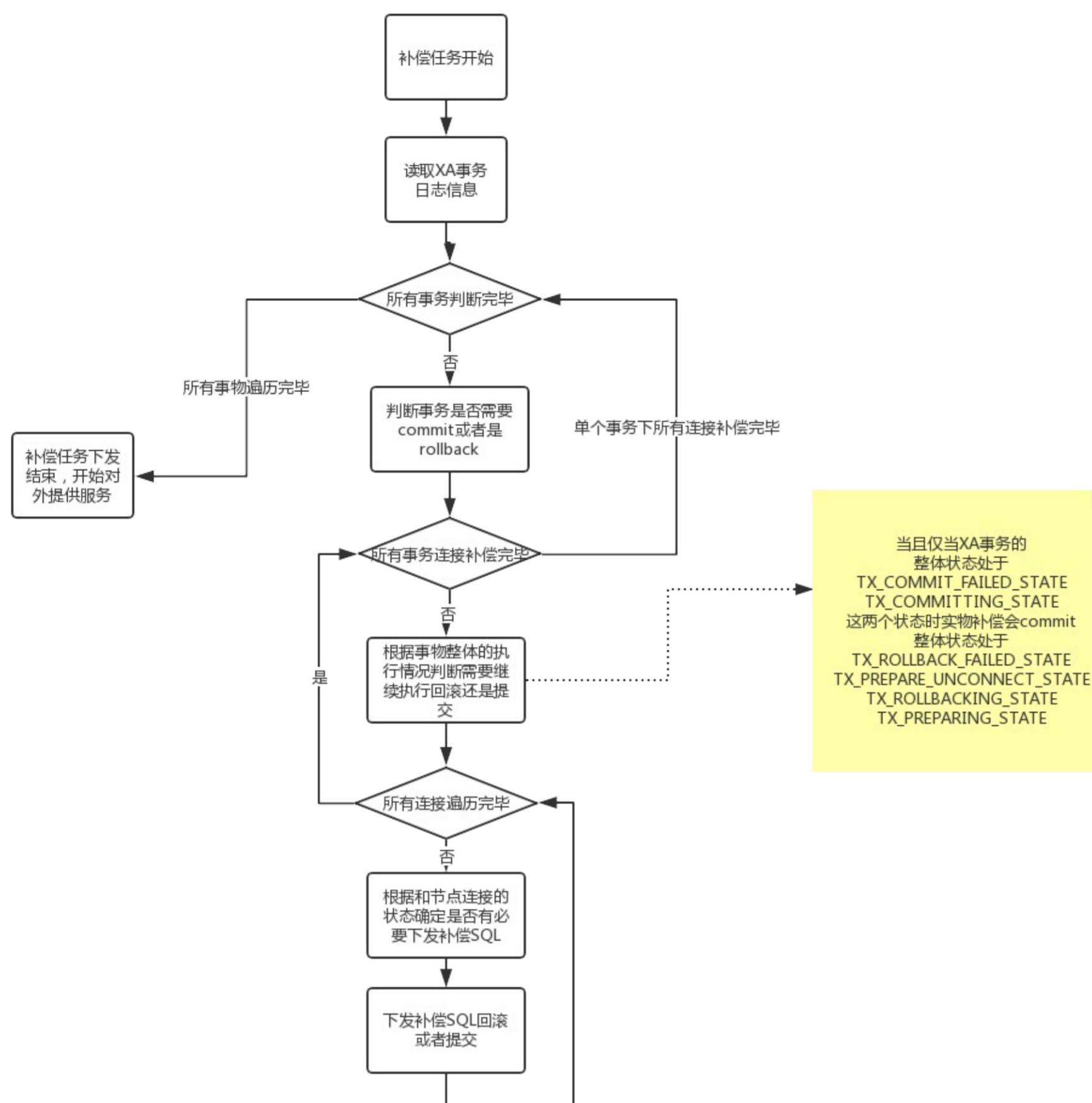
**2.5.2.3.2**

2.19.03.0

1. show @@session.xa xa
2. kill @@xa\_session id1,id2... sessionxa

**2.5.3 XA****2.5.3.1 XA**

XA dbleXA dbleXAXA



XASQLdbe

**2.5.3.2 XAxaLogClean**

xa bootstrap.cnf xaLogCleanPeriod

**2.5.3.3 XAxaSessionCheck**

dblecommit(commit,XApreparecommit) rollback xaprepare

## 2.5.4 XA

### 2.5.4.1 XA

DbleXADBLEXA

```

1. ID
2.
3. host
4.
5.
6. ()
7.

```

```
{
  "id": "'Dble_Server.1.15'",
  "state": "8",
  "participants": [
    {
      "host": "10.186.24.37",
      "port": "3308",
      "p_state": "8",
      "expires": 0,
      "schema": "db3"
    },
    {
      "host": "10.186.24.37",
      "port": "3306",
      "p_state": "8",
      "expires": 0,
      "schema": "db2"
    },
    {
      "host": "10.186.24.37",
      "port": "3308",
      "p_state": "8",
      "expires": 0,
      "schema": "db2"
    },
    {
      "host": "10.186.24.37",
      "port": "3306",
      "p_state": "8",
      "expires": 0,
      "schema": "db1"
    }
  ]
}
```

### 2.5.4.2 XAstatus

status		
0	TX_INITIALIZE_STATE	XA
1	TX_STARTED_STATE	XA XA
2	TX_ENDED_STATE	XA END
3	TX_PREPARED_STATE	XA PREPARED
4	TX_PREPARE_UNCONNECT_STATE	XA PREPARED
5	TX_COMMIT_FAILED_STATE	XA COMMIT
6	TX_ROLLBACK_FAILED_STATE	XA ROLLBACK
7	TX_CONN_QUIT	mysql
8	TX_COMMITED_STATE	XA
9	TX_ROLBACKED_STATE	XA
10	TX_COMMITTING_STATE	XA
11	TX_ROLLBACKING_STATE	XA
12	TX_PREPARING_STATE	XA prepare

### 2.5.4.3 XA

xa

bootstrap.cnf{xaRecoveryLogBaseDir}/{XaRecoveryLogBaseName}.log/tmlogs/tmlogs-1.log

Dble

ZK

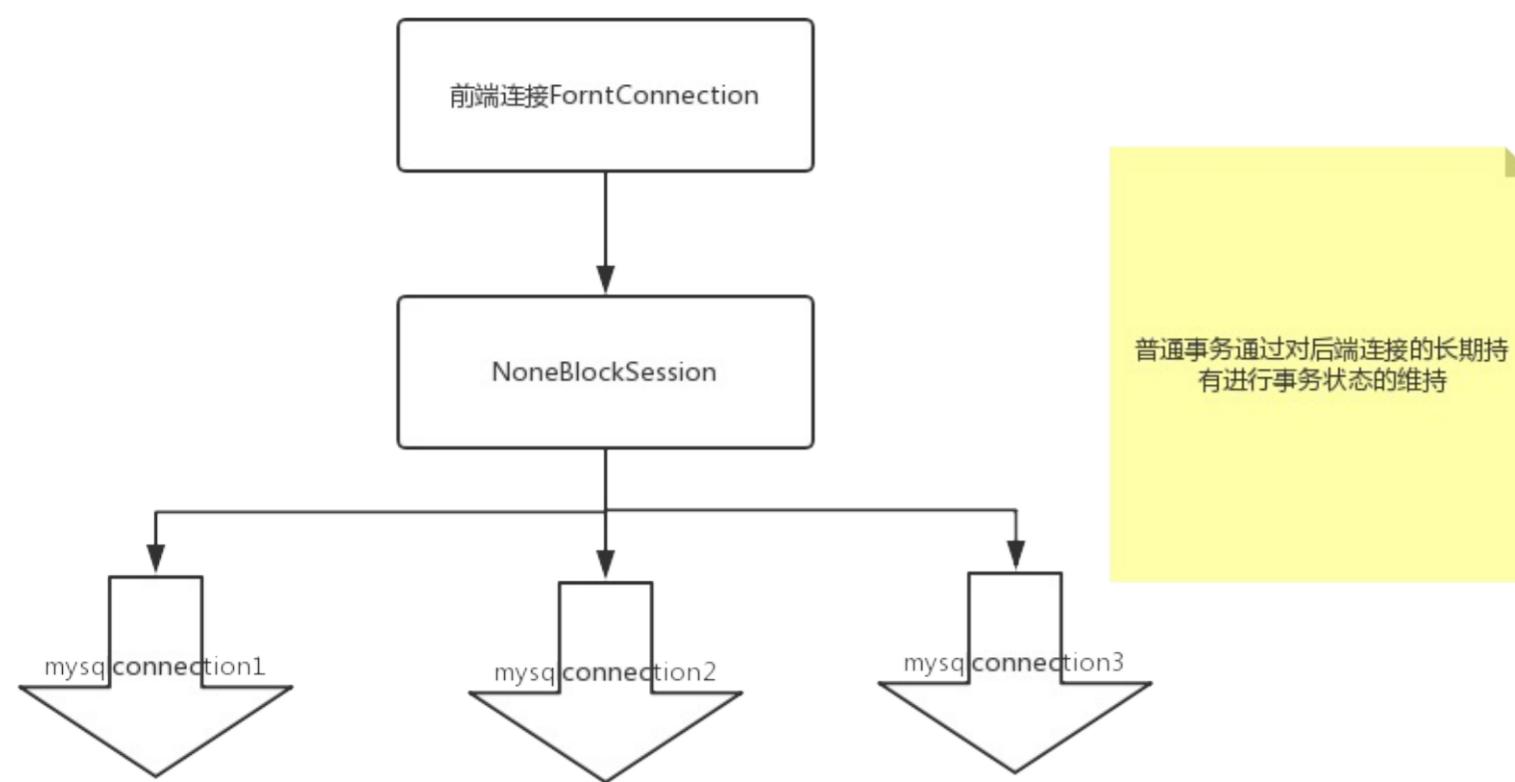
ZKDbleZKXAZK

XAdble/{clusterId}/XALOG/{myid} Key

## 2.5.5

```
mysqlfrontconnectionsessiondblesessionsessiontargetautocommittargetSQLconnectiontargetsessioncommitrollback
```

```
Dblemysqlmysqldn1,dn2,dn3,dn4commitdn1,dn2,dn3dn4dn1,dn2.bn3
```



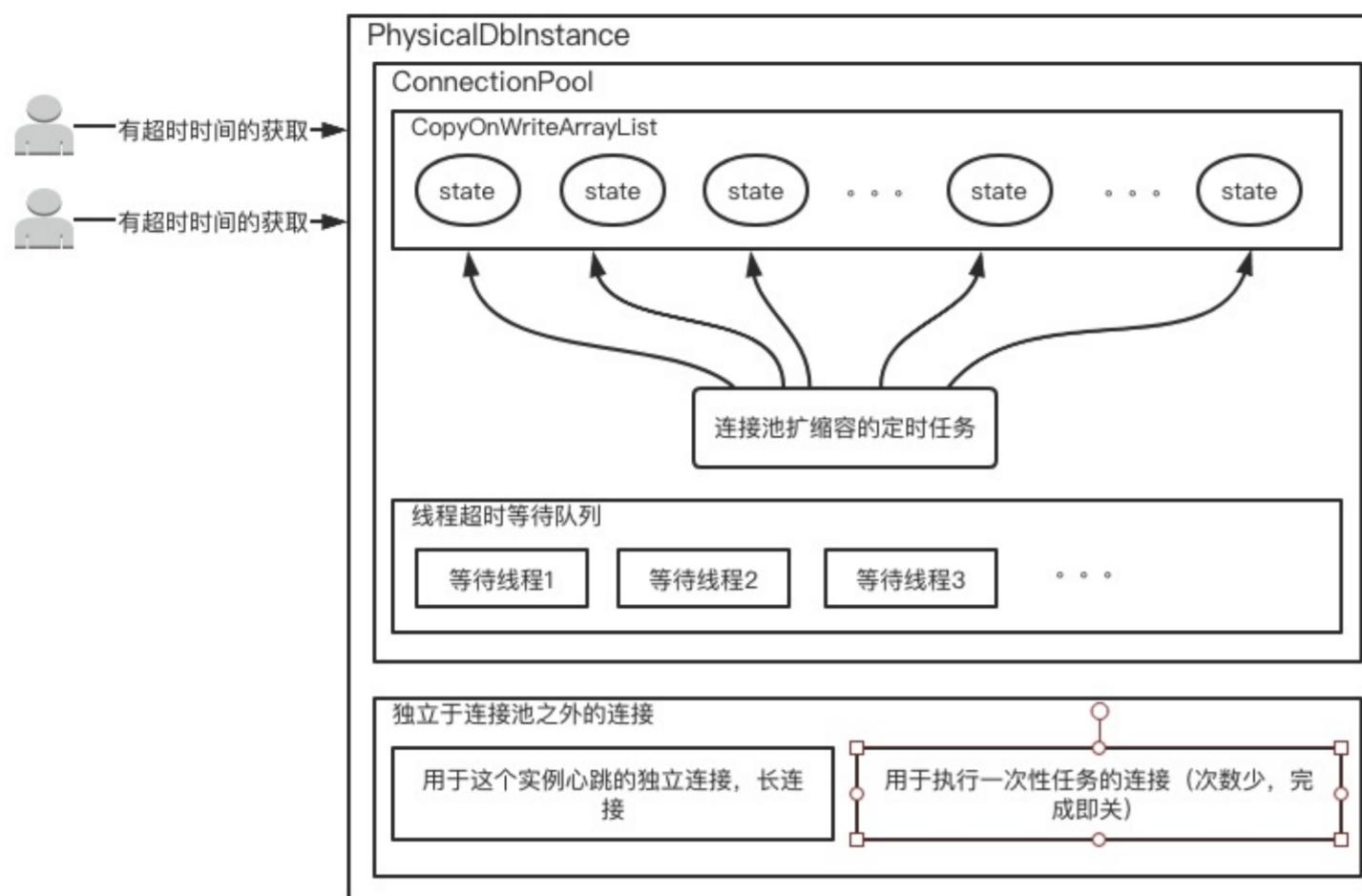
## 2.6

dbleMySQL PhysicalDbInstance PhysicalDbInstance

- 1.
2. MySQLOneTimeJob

### 2.6.1 dble

CopyOnWriteArrayList MySQLstateevictor



#### 2.6.1.1

#### 2.6.1.2

#### 2.6.1.3

evictorevictor

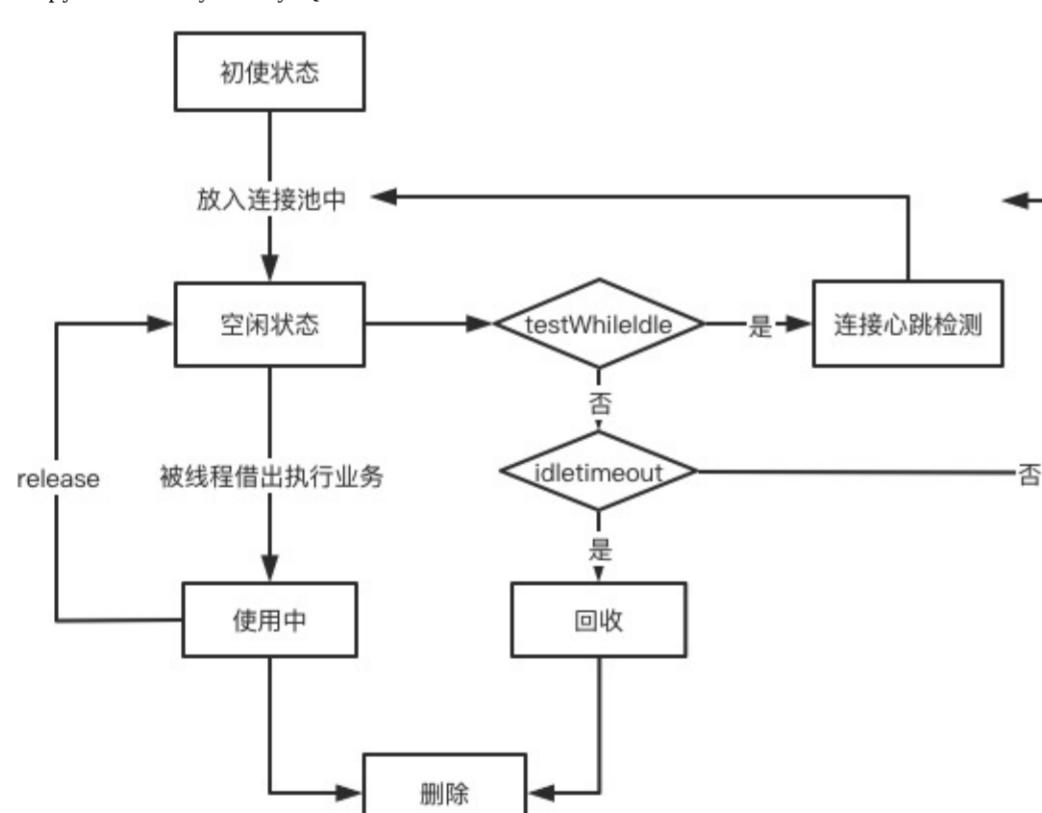
```
minConminConmin( - ) - 0
minConminCon - > 0 && idleTimeout
```

#### 2.6.1.4

- testOnCreate true ping connectionHeartbeatTimeout
- testOnBorrow true ping connectionHeartbeatTimeout
- testOnReturn true ping connectionHeartbeatTimeout
- testWhileIdle true ping connectionHeartbeatTimeout

## 2.6.2

CopyOnWriteArrayList MySQLstate



## 2.6.3

testOnCreate	false		
testOnBorrow	false		
testOnReturn	false		
testWhileIdle	false		
connectionTimeout	30000 (30s)		
connectionHeartbeatTimeout	20		
timeBetweenEvictionRunsMillis	30000 (30s)		
idleTimeout	600000 (10 minute)		
heartbeatPeriodMillis	10000 (10s)		
evictorShutdownTimeoutMillis	10000 (10s)		

### 2.6.4 dble

dbleMySQLheartbeatPeriodMillis dbleMySQLLevictor

#### 2.6.4.1

- 
- 

#### 2.6.4.2

dble

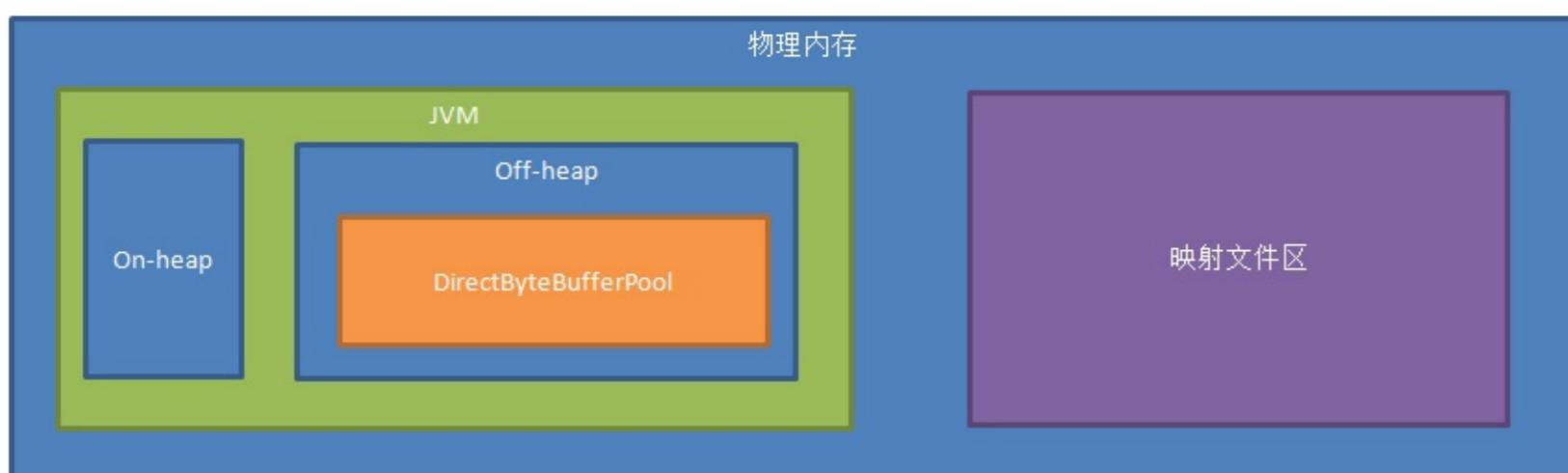
- ok
- timeoutHeartbeatTimeout
- errordble

#### 2.6.4.2

- dblerrorerrorRetryCountok
- dblerrorRetryCounterror

## 2.7

### 2.7.1



- On-Heap JVM Xms ,Xmx jvm
- Off-Heap JVM XX:MaxDirectMemorySize
- DirectByteBufferPool = bufferPoolPageNumber\*bufferPoolPageSize

bufferPoolPageNumberbufferPoolPageSizebootstrap.cnf bufferPoolPageSize2M, bufferPoolPageNumber(MaxDirectMemorySize \* 0.8 /bufferPoolPageSize),

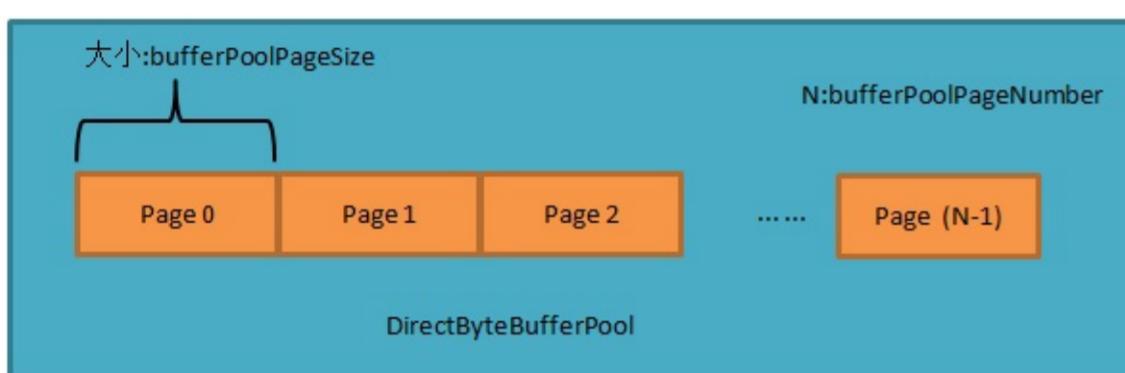
- JVM

```
tmpMin= Min( free)
=((tmpMin/mappedFileSize))
= *mappedFileSize (mappedFileSize64M)

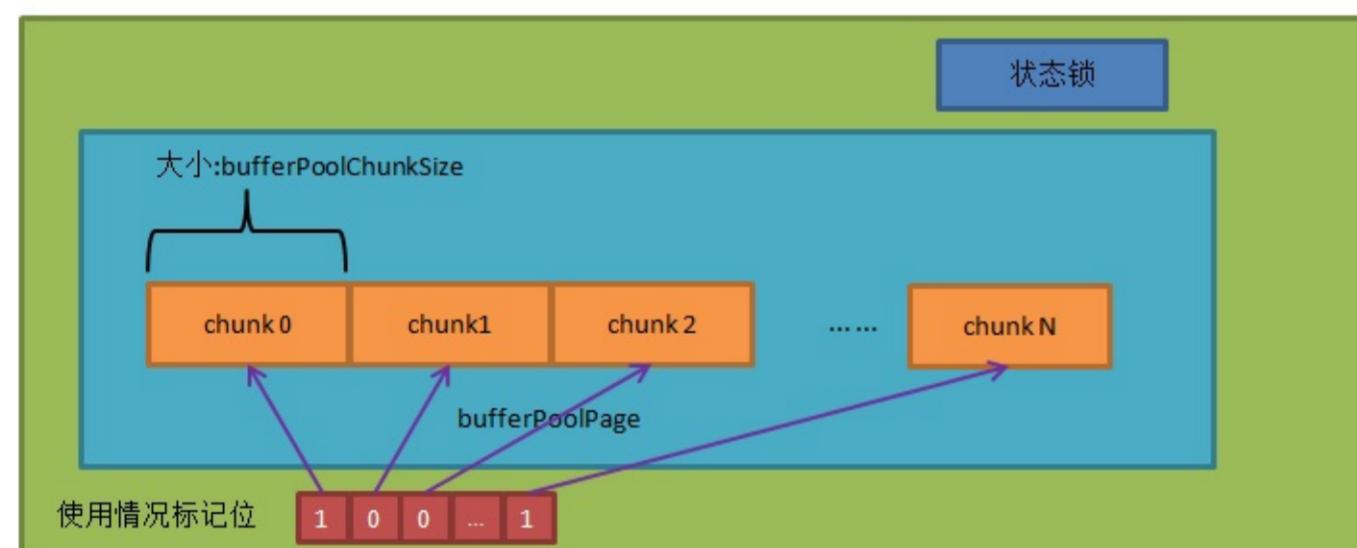
mappedFileSizebootstrap.cnf
```

### 2.7.2 DirectByteBufferPool

BufferPool



bufferPoolPage:



1.:

1.1

1.2 buffer

2.

bufferPoolPageNumberbufferPoolPageSize

3.

3.1

(bufferPoolChunkSize 4k,bufferPoolPageSize )

0

M \*bufferPoolChunkSize

3.2

N+1bufferPoolPageNumber-1(N)

M

ON

{bufferPoolPageSize }On-Heap

4.

4.1 On-Heap

clearGC

4.2 Off-Heap

### 2.7.3

dble session

- Join4Mjoin
- Order4M

- Other4M distinct group nestloop

.Heap4M

mappedFileSize

1:

2:DirectByteBufferPoolchunk

## 2.8 &

### 2.8.1

dble/

zookeeper

### 2.8.2

#### 2.8.2.1 cluster.conf

ZK

```

# clusterEnable=true
# zk
clusterMode=zk
# zk
clusterIP=10.186.19.aa:2281,10.186.60.bb:2281
#zkdble
rootPath=/dbe
#dbe
clusterId=cluster-1
# Ha
#needSyncHa=false
# binlog
#showBinlogStatusTimeout=60000
#
sequenceHandlerType=2
#
#sequenceStartTime=2010-11-04 09:42:54
#instanceIdZK
#sequenceInstanceByZk=true

```

#### 2.8.2.2 bootstrap.conf

instanceName instanceId 01023 0511

### 2.8.3

1.

init\_zk\_data.shZK,ZK

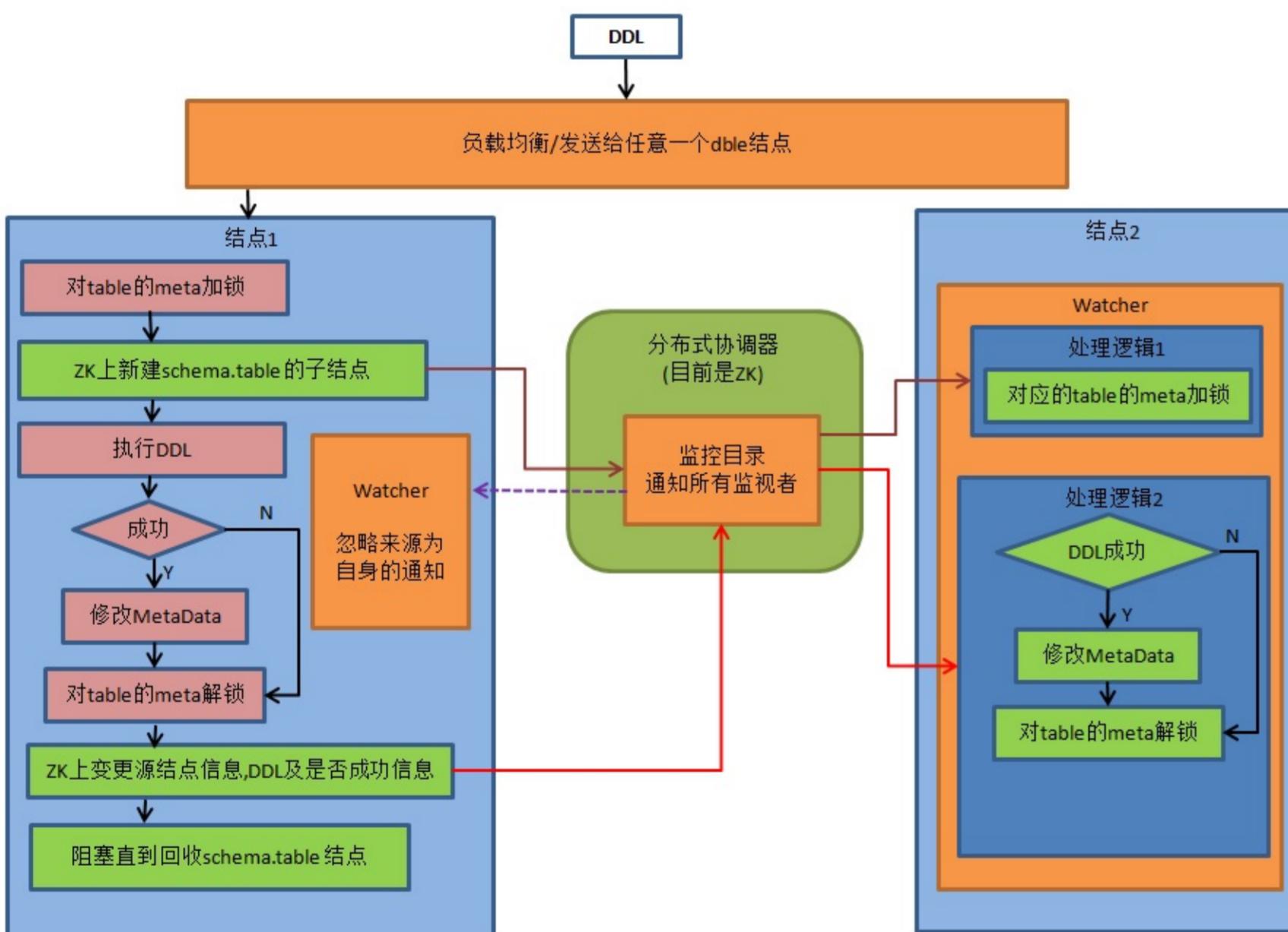
2.

ZK,ZK  
ZK

### 2.8.4

#### A.DDL

DDLZKZK



#### 1. DDL

DDL,

1. ddlschema.table

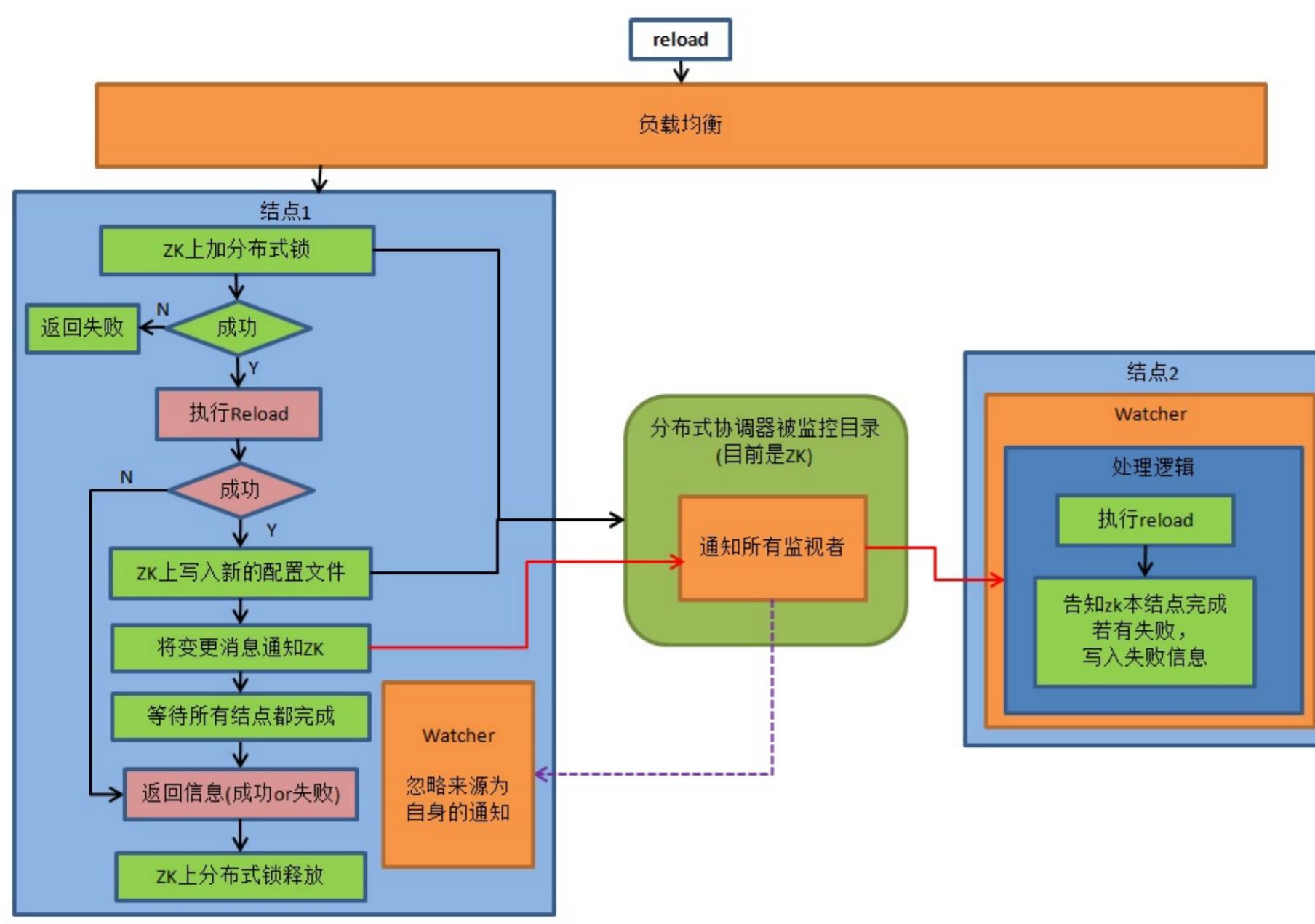
1. DDLSchemametaZKddlmetareload metadata

#### 1. meta

(meta,reload meta)

1. :view,viewview

**B.reload @@config/ reload @@config\_all/ rollback @@config**



db.xmlsharding.xml, userxml ,sequence\_conf.properties

sequence\_db\_conf.properties

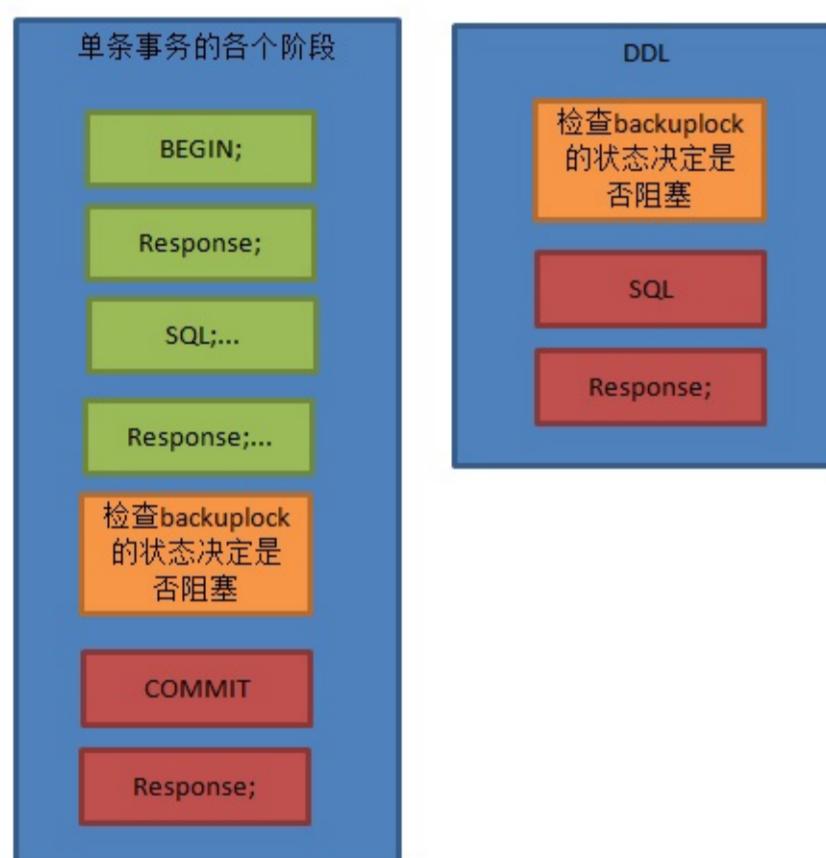
**C.binlog**

:

binlogshow master statusbinlog

```

show @@binlog.status session
sessionshow @@binlog.status
sessionsshow master status
  
```



```

:session1 tableA session2 tableA DDL
metaLock.session3 show @@binlog.status.
session1 session3, session2 session1, session3 session2.
session3 showBinlogStatusTimeout(60s)
  
```

```

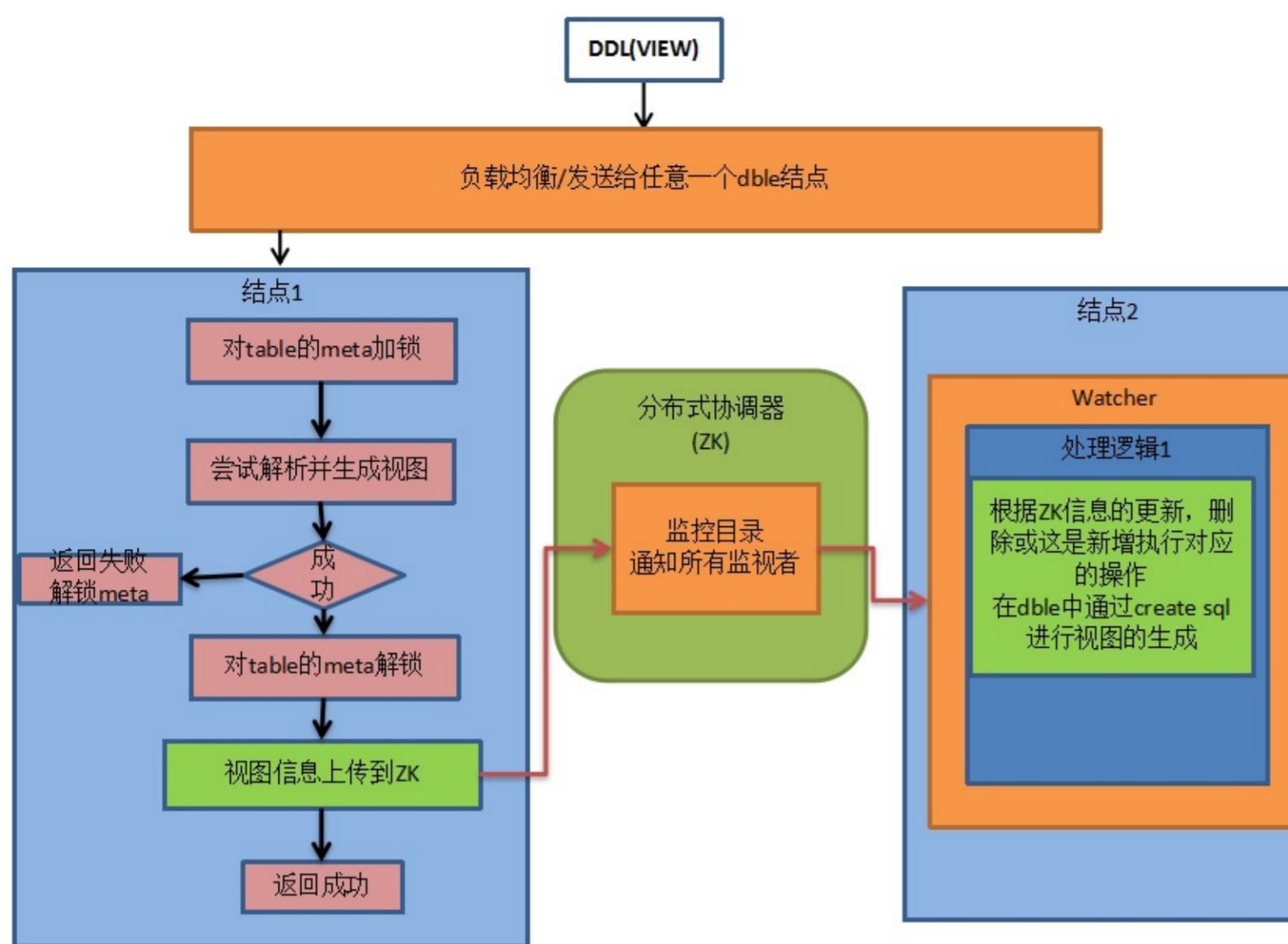
1.ZKzk
2.session
3./ZK
4.,show @@binlog.status
5.ZK
  
```

ZK

ZK/zk

**D.View**

zkviewdble



zkview key schema.table value {"serverId":"create\_Server\_id", "createViewSql": "view\_create\_sql"}

#### E.online

dbleonlineDDL

1.dble 2.dbleDDLBinlog

#### F.

needSyncHa

#### E.1 disable

- 1.
- 2.disable
- 3.
- 4.disable
- 5.
- 6.

#### E.2 enable

- 1.
- 2.enable
- 3.
- 4.
- 5.enable

#### E.3 switch

- 1.
- 2.switch
- 3.
- 4.
- 5.switch

#### G.

#### G.1 pause

1. 2. 3. 4. 5.

#### G.2 resume

1. 2. 3. 4.

#### 2.8.5 XA

XAzookeeper

#### 2.8.6 ZK

```

rootPath(cluster.cnf)
clusterId(cluster.cnf)
conf
  initiated
  status
  operator
    instanceName(keybootstrap.cnf:reloadid)
    sharding(sharding.xmljson)
    db(db.xmljson)
    user(user.xmljson)
    migration()
    pause
      instanceName(keybootstrap.cnf:)
    resume
      instanceName(keybootstrap.cnf:)
  sequences
  sequences
  sequences //zk
  incr_sequence//table_name
  common(sequence_conf.properties sequence_db_conf.properties)
binlog_pause
status()
instanceName(keybootstrap.cnf:)
lock
  syncMeta.lock()
  confInit.lock
  confChange.lock
  binlogStatus.lock
  ddl_lock/schema.table
  view_lock/schema.table
  dbGroup_locks/groupName
  pause_node.lock
online
  instanceName(keybootstrap.cnf)
ddl
  schema.tableddl
  instanceName(keybootstrap.cnf:)
  schema.table2
xalog
  node1
  node2
view
  schema:view

```

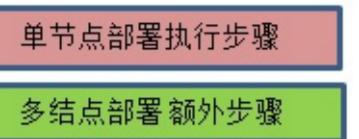
```
operator
schema.view:(update/delete,:createupdate)
instanceName(keybootstrap.cnf:)

dbGroups
dbGroup_status
groupName
dbGroup_response
instanceName(keybootstrap.cnf:)
```

### 2.8.7

twitter snowflake ZKinstanceID  
offset-step ZKStep

### 2.8.8



图例

## 2.9 Grpc

### 2.9.1

Dbleucoredbleucoregrpucore

### 2.9.2

#### myid.properties

				/
url	grpcurl	myid.properties ipAddress	grpcIP	
port		myid.properties port	grpc	
serverId	ID	\$ushard-id(ip1,ip2) ,\$ushard-id myid myid.properties myid		
componentId	ID	\$ushard-id myid.properties myid		
componentType		ushard		

## 2.10 meta

### Meta

- [2.10.1 Meta](#)
- [2.10.2 Meta](#)
- [2.10.3](#)
- [2.10.4 View Meta](#)

**2.10.1 Meta**

dble

**2.10.1.1**

```
dbleschematable1.2 schemal.xml  
show tables;  
show create table ...
```

**2.10.1.2**

```
DbleDble  
1 view  
2 ZKview  
view meta2.10.4 view meta
```

**2.10.1.3**

```
:  
• ;  
•  
•  
•  
•
```

## 2.10.2 Meta

dbledddll

- create table
- drop table
- alter table
- truncate table
- create index
- drop index

zookeeper(1.8 myid.properties)Meta:

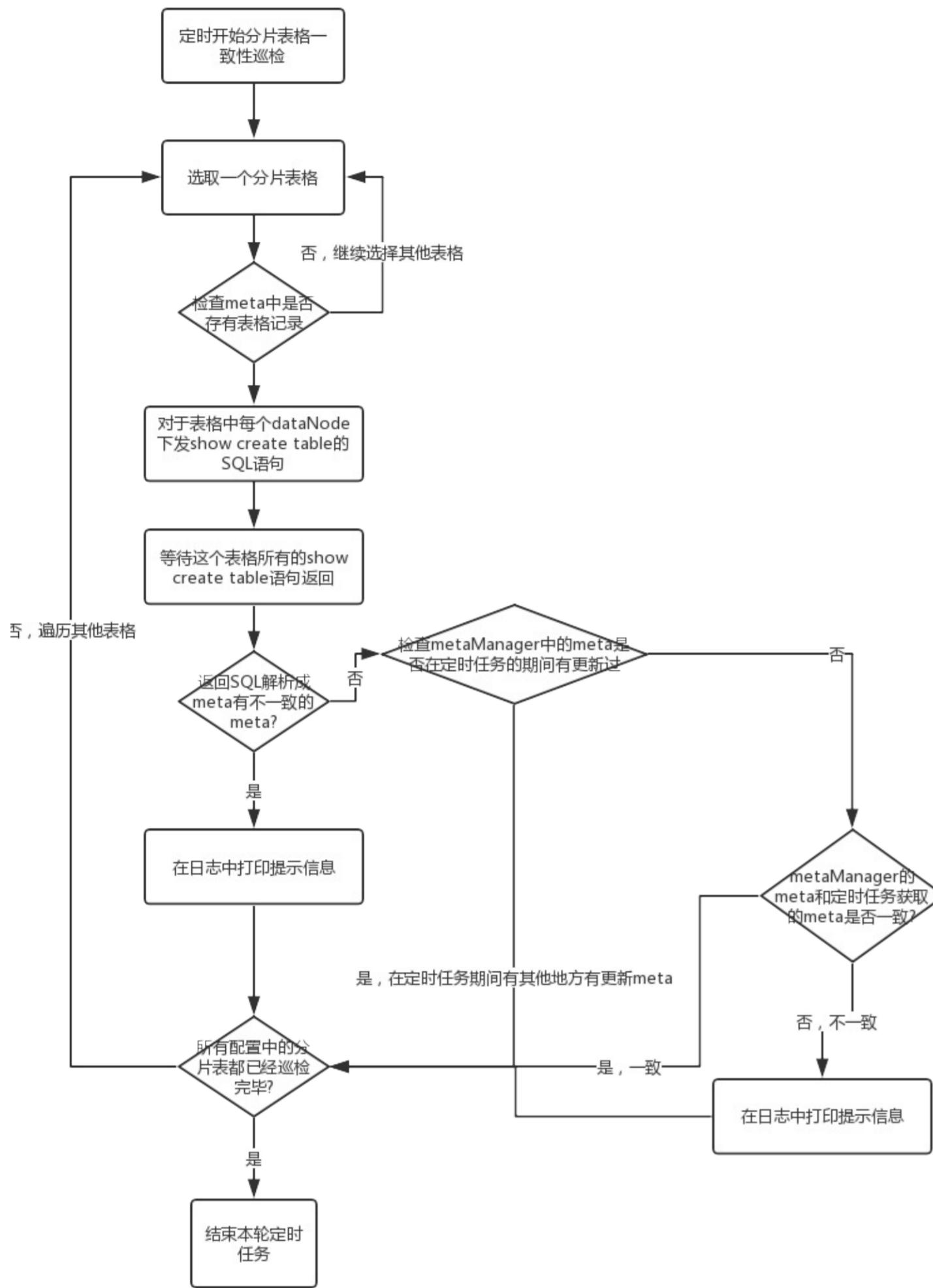
1. zookeeper
  - dble
2. zookeeper
  - :
  - a. dblezookeeper
  - b. dblezookepr
  - c. dble

**2.10.3**

DbleDbleDble

tableStructureCheckTaskcheckTableConsistencyPeriod30×60×100030checkTableConsistency0bootstrap.cnf

- schematable
- meta
- SQL “show create table”
- metaSet
- metameta
- metametameta



Dble,

## 2.10.4 view meta

### 2.10.4.1 view

- mysqlview
- dbleview

### 2.10.4.2 view meta

Dble 2.18.11.0 viewviewVIEWselectquerySQLview

MySQLView metaDbleviewDbleviewDbleDbleviewXASQL

- 
- ZK

DblemetaZKSQlmeta

Dble 2.19.10.0mysql viewmysql viewdblemysql mysql viewview schema

```
<schema name="schema2" sqlMaxLimit="100" shardingNode="dn5">
</schema>
```

viewshardingNode dn5 mysql viewdbleview metysql view dblereaddble

### 2.10.4.2 view meta

JSONbootstrap.cnfviewPersistenceConfBaseDirviewPersistenceConfBaseName./viewConf/viewJson,

```
[{
  "schema": "testdb",
  "list": [
    {
      "name": "view_test",
      "sql": "create view view_test as select * from a_test"
    },
    {
      "name": "vt2",
      "sql": "create or replace view vt2 as select * from suntest"
    },
    {
      "name": "suntest",
      "sql": "create view suntest as select * from sbtest"
    }
  ]
}]
```

### ZK K/V

Dbleviewkey ZK BASE\_PATH/viewkeyschema\_name:view\_nameviewjsoncreate sqlserverID

```
{
  "serverId": "10010",
  "createSql": "create view view_test as select * from a_test"
}
```

JSONvalue /.../view/testdb:view\_test

**2.11**

- [2.11.1](#)
- [2.11.2](#)
- [2.11.3](#)
- [2.11.4](#)
- [2.11.5 heartbeat](#)
- [2.11.6](#)

**2.11.1**

```
dble  
useSqlStat
```

**2.11.1.1**

```
reload @@query_cf=table&column;  
table column
```

```
reload @@query_cf;
```

**2.11.1.2**

```
show @@sql.condition;
```

**2.11.2**

useSqlStat

**2.11.2.1**

- 
- 
- 
- 

**2.11.2.2**

```
show @@sql.sum.table;
```

```
show @@sql.sum.table true;
```

**2.11.3**

useSqlStat

**2.11.3.1**

```
1.  
2. NSQLsqlNsqlRecordCount10SQLTsql  
:  
reload @@sqlslow=t;  
t  
3.  
4. 50  
5. 1000010select  
6. 1024sql  
7. MsqlMmaxResultSet
```

**2.11.3.2**

- show @@sql;
- show @@sql.high;
- show @@sql.large;
- show @@sql.resultset;
- show @@sql.slow;
- show @@sql.sum.user;

2.1

**2.11.3.3**

```
:  
  
reload @@user_stat;
```

**2.11.4**

dbe

1. initDB
2. query
3. stmtPrepare
4. stmtSendLongData
5. stmtReset
6. stmtExecute
7. stmtClose
8. ping
9. kill
10. quit
11. heartbeat
12. other

**2.11.4.1**

- show @@command;
- show @@command.count; [2.1](#)

**2.11.5 heartbeat**

heartbeatmysqlheartbeatmysqlheartbeat

**2.11.5.1**

heartbeatheartbeat

**2.11.5.2**

heartbeat

- show @@heartbeat;
- show @@heartbeat.detail where name=xxx; xxdbinstance
- show @@dbinstance.synstatus;
- show @@dbinstance.syndetail where name=xxx;xxdbinstance

2.1

**2.11.6**

dbe

**2.11.6.1**

- 
- 
- /

**2.11.6.2**

- show @@connection;
- show @@backend;
- show @@connection.sql;

2.1

## 2.12

```
2.20.04.0,
MySQL:
dblepython3dble,bootstrap.cnfsystemuseOuterHa false
dblesystemuseOuterHa true.
```

### 2.12.1

- dblebootstrap.cnfsystemuseOuterHa false
- python3"python"

### 2.12.2

```
dbledblepython3db.xmldbe
```

```
dble
```

```
show @@custom_mysql_ha
```

```
python
```

```
enable @@custom_mysql_ha
```

```
disable @@custom_mysql_ha
```

### 2.12.2

#### 2.12.2.1 linux

#### 2.12.2.2 reload

```
reload @@config python
1. disable @@custom_mysql_ha
2.
3. reload @@config
4. enable @@custom_mysql_ha
```

#### 2.12.2.3 python

```
custom_mysql_ha.py dblebin,python3,
```

#### 1Python3,python3

```
/usr/local/bin/python3 --version
/usr/local/bin/pip3 --version
```

#### 2mysqlclient

CentOS

```
yum install mysql-devel
```

or Ubuntu

```
apt-get install libmysqlclient-dev
```

```
pip3 install mysqlclient
```

#### 3six

```
pip3 install six
```

or

```
pip3 install six -i http://pypi.douban.com/simple --trusted-host pypi.douban.com
```

#### 4coloredlogs

```
pip3 install coloredlogs
```

or

```
pip3 install coloredlogs -i http://pypi.douban.com/simple --trusted-host pypi.douban.com
```

#### 5rsa

```
pip3 install rsa
```

or

```
pip3 install rsa -i http://pypi.douban.com/simple --trusted-host pypi.douban.com
```

## 2.13

### 2.13.1

- bootstrap.cnfprocessorCheckPeriod
- bootstrap.cnfidleTimeout
- 

### 2.13.2 SQL

- bootstrap.cnfprocessorCheckPeriod
- sqlExecuteTimeout
- DDL

## 2.14 ER

### 2.14.1 ER

join, nest loop  
join

销售单		
流水号	顾客ID	日期
201712010001	1	20171201
201712010002	4	20171201
201712020001	15	20171202
201712020002	4	20171202

销售详情单	
流水号	商品ID
201712010001	1
201712010001	2
201712010002	7
201712010002	31
201712020001	103
201712020001	2304
201712020002	27
201712020002	91

join

;

分片node1	
销售单(按日期拆分表1)	
销售详情单	
流水号	商品ID
201712010001	1
201712010002	2
201712010002	7
201712010002	31

分片node2	
销售单(按日期拆分表2)	
销售详情单	
流水号	商品ID
201712020001	103
201712020001	2304
201712020002	27
201712020002	91

ER

```
<shardingTable name="sales" shardingNode="dn1,dn2" function="sharding" shardingColumn="id">
<childTable name="sales_detail" joinKey="sales_detail_pos_num" parentKey="sales_pos_num"/>
</table>
```

### 2.14.2 ER

2mycatERER

```
<!--schema-->
<shardingTable name="tableA" shardingNode="dn1,dn2" function="hash_function" shardingColumn="id_a" />
<shardingTable name="tableB" shardingNode="dn1,dn2" function="hash_function" shardingColumn="id_b" />
<shardingTable name="tableC" shardingNode="dn2,dn1" function="hash_function" shardingColumn="id_c" />
<shardingTable name="tableD" shardingNode="dn3,dn4" function="hash_function" shardingColumn="id_a" />
<shardingTable name="tableE" shardingNode="dn1,dn2" function="hash_function" shardingColumn="id_a" />
<shardingTable name="tableF" shardingNode="dn1,dn2" function="enum_par" shardingColumn="id_a" />

<!--rfunction-->
<function name="enum_par">
  class="io.mycat.route.function.PartitionByFileMap"
  <property name="mapfile">partition-hash-int.txt</property>
</function>
<function name="hash_function" class="io.mycat.route.function.PartitionByLong">
  <property name="partitionCount">2</property>
  <property name="partitionLength">512</property>
</function>
```

functionER

table			function	
tableA	id_a	dn1,dn2	hash_function	1
tableB	id_b	dn1,dn2	hash_function	1
tableC	id_c	dn2,dn1	hash_function	2
tableD	id_a	dn3,dn4	hash_function	3
tableE	id_a	dn1,dn2	hash_function	1
tableF	id_a	dn1,dn2	enum_par	4

ER

PSschemaschemaERER.



## 2.16 cache

cache 1.6

### 2.16.1

```
[sql, ]  
KEYpool.SQLRouteCache  
VALUE cachefactoryname,
```

### 2.16.2 ER

```
[joinkey, ]  
KEYpool.ER_SQL2PARENTID  
VALUEEcachefactoryname,
```

## 2.17

### 2.17.1

SQLSQL

### 2.17.2

dbledble  
dbe: SQLdbeSQLSQLSQL  
: MySQL

### 2.17.3 dble

dbleEXPLAINdbe explain select id,accountno from account where userid=2;  
EXPLAINSQ

```
mysql> explain select * from sharding_two_node a inner join sharding_four_node b on a.id =b.id;
+-----+-----+-----+
| SHARDING_NODE | TYPE   | SQL/REF |
+-----+-----+-----+
| dn1.0          | BASE SQL | select `a`.`id`, `a`.`c_char`, `a`.`ts`, `a`.`si` from `sharding_two_node` `a` ORDER BY `a`.`id` ASC |
| dn2.0          | BASE SQL | select `a`.`id`, `a`.`c_char`, `a`.`ts`, `a`.`si` from `sharding_two_node` `a` ORDER BY `a`.`id` ASC |
| dn1.1          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| dn2.1          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| dn3.0          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| dn4.0          | BASE SQL | select `b`.`id`, `b`.`c_flag`, `b`.`c_decimal` from `sharding_four_node` `b` ORDER BY `b`.`id` ASC |
| merge.1        | MERGE   | dn1.0, dn2.0 |
| merge.2        | MERGE   | dn1.1, dn2.1, dn3.0, dn4.0 |
| join.1         | JOIN    | merge.1, merge.2 |
+-----+-----+-----+
9 rows in set (0.00 sec)
```

```
mysql> explain select id from single union all select b.si from sharding_four_node a inner join sharding_two_node b on a.id =b.id
+-----+-----+-----+
| SHARDING_NODE | TYPE   | SQL/REF |
+-----+-----+-----+
| dn1.0          | BASE SQL | select `single`.`id` from `single` |
| dn1.1          | BASE SQL | select `a`.`id` from `sharding_four_node` `a` ORDER BY `a`.`id` ASC |
| dn2.0          | BASE SQL | select `a`.`id` from `sharding_four_node` `a` ORDER BY `a`.`id` ASC |
| dn3.0          | BASE SQL | select `a`.`id` from `sharding_four_node` `a` ORDER BY `a`.`id` ASC |
| dn4.0          | BASE SQL | select `a`.`id` from `sharding_four_node` `a` ORDER BY `a`.`id` ASC |
| dn1.2          | BASE SQL | select `b`.`si`, `b`.`id` from `sharding_two_node` `b` ORDER BY `b`.`id` ASC |
| dn2.1          | BASE SQL | select `b`.`si`, `b`.`id` from `sharding_two_node` `b` ORDER BY `b`.`id` ASC |
| merge.2        | MERGE   | dn1.1, dn2.0, dn3.0, dn4.0 |
| merge.3        | MERGE   | dn1.2, dn2.1 |
| join.1         | JOIN    | merge.2, merge.3 |
| merge.1         | MERGE   | dn1.0 |
| union_all.1    | UNION_ALL | join.1, merge.1 |
+-----+-----+-----+
12 rows in set (0.01 sec)
```

### 2.17.4

EXPLAIN2  
explain2 shardingnode=dn1 sql=select id,accountno from account where userid=2;  
explain2sqlexplainshardingnodeexplain

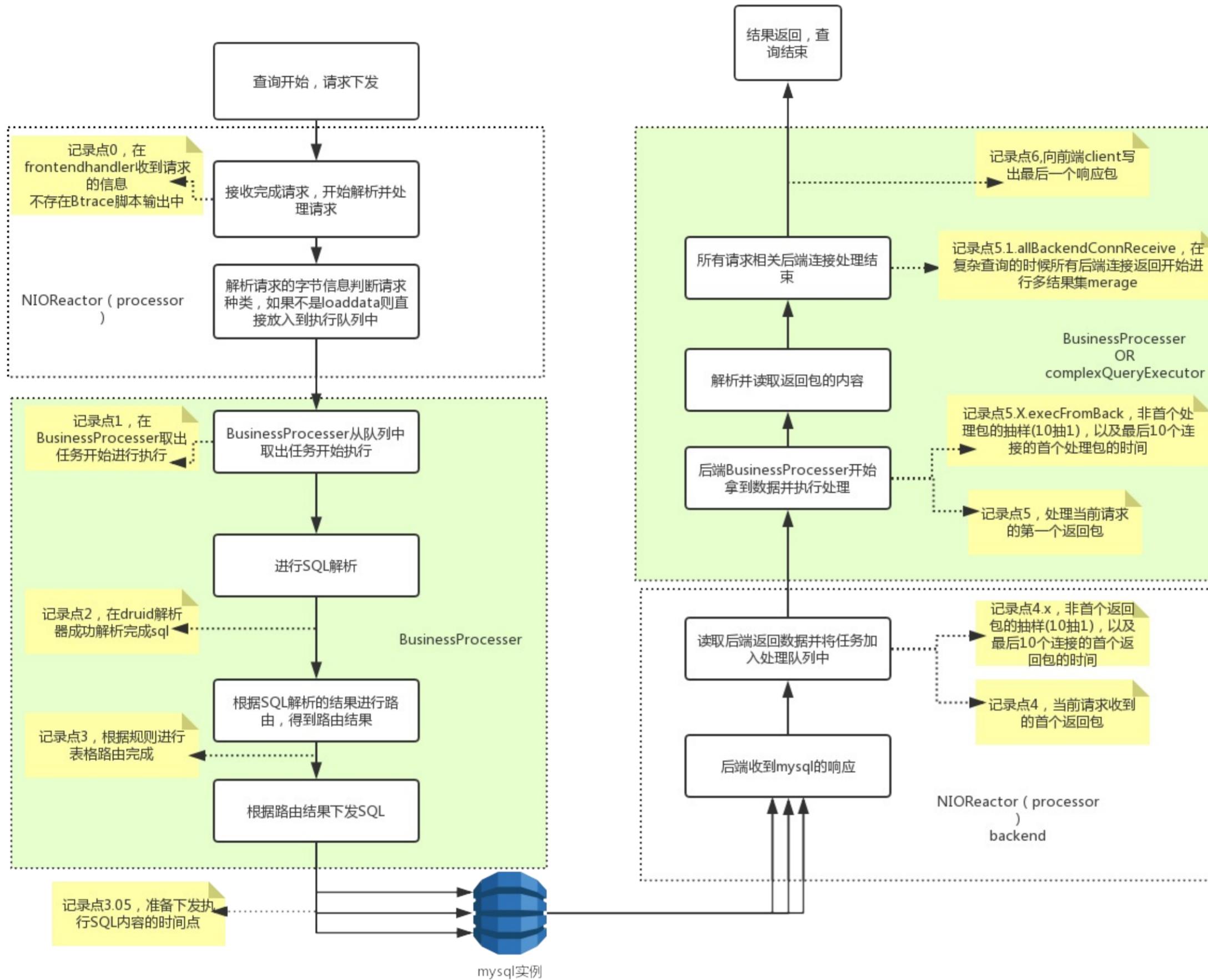
## 2.18

- Btrace()
- managershow @@thread\_used()

### 2.18.1 Btrace

#### 2.18.1.1

Dble, BTraceCostTime.java Btrace, Btrace v.1.3  
Btrace <https://github.com/btraceio/btrace>  
bootstrap.cnfuseCostTimeStat costSamplePercent useCostTimeStat = 1 costSamplePercent 1% btrace



bttraceable

profiling:	Block	Invocations	SelfTime.Total	SelfTime.Avg	SelfTime.Min	SelfTime.Max	WallTime.Total	WallTime.Avg	WallTime.Min	WallTime.Max
	request->1.startProcess	9073	-638142734	-70334	-1051058	493260	202952071	22368	10565	493260
	request->2.endParse	9073	234134936	25805	13206	523393	437087007	48174	23771	1016653
	request->3.endRoute	9073	404389553	44570	20123	121474	841476560	92745	43894	1075553
	request->4.resFromBack	9073	592398	65	-649019	1602901	4805691043	529669	261612	1602901
	request->5.startExecuteBackend	9073	-56808823	-6261	-1749483	2020297	5047581273	556329	350530	2020297
	request->6.response	9073	59150286	6519	3045	366620	5107315454	562913	353575	2386917

Block, Invocations, WallTime. , SelfTime. btrace

#### 2.18.1.2

- 0 : Odble
- 1.startProcess :
- 2.endParse : SQL
- 3.endRoute : SQL
- 3.05 readyToDeliver : SQL
- 4.resFromBack :
- 4.X.resFromBack : .4.3.resFromBack dble1
- 5.startExecuteBackend :
- 5.X.startExecuteBackend : 5.3.startExecuteBackend
- 5.1.allBackendConnReceive merge
- 6.response :

:BtraceSQLSQLSQL

#### 2.18.1.3

- 1-0bootstrap.cnfprocessorExecutor Odble01-0request->1.startProcess
- 4-3bootstrap.cnfbackendProcessors
- 5-4bootstrap.cnfbackendBusinessExecutor

### 2.18.2 Manager

#### 2.18.2.1

Dble 18.02.0managerbootstrap.cnfuseThreadUsageStat  
show @@thread\_useddble

THREAD_NAME	LAST_QUARTER_MIN	LAST_MINUTE	LAST_FIVE_MINUTE
backendBusinessExecutor2	0%	0%	0%
backendBusinessExecutor1	0%	0%	0%
backendBusinessExecutor0	0%	0%	0%
BusinessExecutor3	0%	0%	0%
\$_NIO_REACTOR_BACKEND-2	0%	0%	0%
BusinessExecutor1	0%	0%	0%
\$_NIO_REACTOR_BACKEND-3	0%	0%	0%
BusinessExecutor2	12%	3%	3%
\$_NIO_REACTOR_BACKEND-0	0%	0%	0%
\$_NIO_REACTOR_FRONT-0	0%	0%	0%
\$_NIO_REACTOR_BACKEND-1	0%	0%	0%
BusinessExecutor0	0%	0%	0%

12 rows in set (0.00 sec)

- BussinessExecutorX  
sqlmysqld
- backendBusinessExecutorX  
mysqlclient
- \$\_NIO\_REACTOR\_FRONT\_X  
,,BussinessExecutor
- \$\_NIO\_REACTOR\_BACKEND\_X  
mysqlbackendBusinessExecutor

### 2.18.2.2

80%

- processors \$\_NIO\_REACTOR\_FRONT\_X
- backendProcessors \$\_NIO\_REACTOR\_BACKEND\_X
- backendProcessorExecutor backendBusinessExecutorX
- processorExecutor BussinessExecutorX

**2.19 reload**

reload @@config\_all dbGroupreload

dbGroupdbGroupdbGroup

**2.19.1 reload @@config\_all**

:

**2.19.1.1 dbGroup**

,dbGroupschmea,

**2.19.1.2 dbGroup****2.19.1.3 dbGroup**

show backend

**2.19.2 reload @@config\_all -f**

:

**2.19.2.1 dbGroup**

,dbGroup

**2.19.2.2 dbGroup****2.19.2.3 dbGroup****2.19.3 reload @@config\_all -r**

dbGroup

show backend

**2.19.4 reload @@config\_all -s**

dbGroup

## 2.20

MySQLLdble server MySQL(MySQLmyisqldumpslowPerconapt-query-digest)  
SQL show @@connection.sql.status where FRONT\_ID=?;

### 2.20.1 bootstrap.cnf6

```
<!-- -->
-DenableSlowLog=1
<!-- -->
-DslowLogFile=./slowlogs
<!-- -->
-DslowLogBaseName=slow-query
<!-- -->
-DflushSlowLogPeriod=1
<!-- -->
-DflushSlowLogSize=1000
<!-- , -->
-DsqlSlowTime=100
```

### 2.20.2

```
enable @@slow_query_log; --
show @@slow_query_log; --
disable @@slow_query_log; --
show @@slow_query_log; --

show @@slow_query.time; --
reload @@slow_query.time=200; --

show @@slow_query.flushperiod; --
reload @@slow_query.flushperiod=2; --

show @@slow_query.flushsize; --
reload @@slow_query.flushsize=1100; --
```

### 2.20.3 MySQL mysqldumpslow Percona pt-query-digest

```
: 

/FAKE_PATH/mysql, Version: FAKE_VERSION. started with:
Tcp port: 3320 Unix socket: FAKE_SOCK
Time           Id  Command   Argument
# Time: 2018-08-23T17:40:10.149000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.132769  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000350  Prepare_Push: 0.116678  dn1_First_Result_Fetch: 0.013686  dn1_Last_Result_Fetch: 0.001422  Write_Client: 0.001995
SET timestamp=1535017210149;
select * from sharding_two_node where id =1;
# Time: 2018-08-23T17:40:10.200000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.035600  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000062  Prepare_Push: 0.006733  dn2_First_Result_Fetch: 0.012524  dn1_First_Result_Fetch: 0.010971  dn2_Last_Result_Fetch: 0.015368  dn1_Last_Result_Fetch: 0.005119  Write_Client: 0.017834
SET timestamp=1535017210200;
select * from sharding_two_node;
# Time: 2018-08-23T17:40:10.282000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.045337  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000166  Prepare_Push: 0.003941  dn1_First_Result_Fetch: 0.039652  dn1_Last_Result_Fetch: 0.000300  Write_Client: 0.001578
SET timestamp=1535017210282;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:10.315000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.031232  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.005467  Prepare_Push: 0.001989  dn2_First_Result_Fetch: 0.029240  dn2_Last_Result_Fetch: 0.001900  Write_Client: 0.003536
SET timestamp=1535017210315;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:10.432000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.116672  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.013625  Prepare_Push: 0.024767  dn2_First_Result_Fetch: 0.056395  dn1_First_Result_Fetch: 0.026420  dn2_Last_Result_Fetch: 0.000743  dn1_Last_Result_Fetch: 0.001700  Write_Client: 0.051861
SET timestamp=1535017210432;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:10.772000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.338569  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000082  Prepare_Push: 0.258365  dn1_0_First_Result_Fetch: 0.047494  dn1_0_Last_Result_Fetch: 0.029018  dn2_0_First_Result_Fetch: 0.042964  dn2_0_Last_Result_Fetch: 0.033525  Write_Client: 0.009385
SET timestamp=1535017210772;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:10.821000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.046745  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000059  Prepare_Push: 0.025401  dn1_0_First_Result_Fetch: 0.011755  dn1_0_Last_Result_Fetch: 0.001180  Generate_New_Query: 0.001706  dn1_1_First_Result_Fetch: 0.004224  dn1_1_Last_Result_Fetch: 0.001213
Write_Client: 0.001384
SET timestamp=1535017210821;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:12.061000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.036952  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.001111  Prepare_Push: 0.001132  dn1_First_Result_Fetch: 0.034266  dn1_Last_Result_Fetch: 0.000084  Write_Client: 0.000443
SET timestamp=1535017212061;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:12.091000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.028213  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000666  Prepare_Push: 0.001206  dn2_First_Result_Fetch: 0.025991  dn2_Last_Result_Fetch: 0.000101  Write_Client: 0.000349
SET timestamp=1535017212091;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:12.132000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.040365  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000223  Prepare_Push: 0.001172  dn2_First_Result_Fetch: 0.019852  dn1_First_Result_Fetch: 0.019810  dn2_Last_Result_Fetch: 0.000901  dn1_Last_Result_Fetch: 0.000780  Write_Client: 0.019160
SET timestamp=1535017212132;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:12.145000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.021296  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000115  Prepare_Push: 0.001403  dn1_0_First_Result_Fetch: 0.006714  dn1_0_Last_Result_Fetch: 0.002561  dn2_0_First_Result_Fetch: 0.006787  dn2_0_Last_Result_Fetch: 0.001806  Write_Client: 0.002280
SET timestamp=1535017212145;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:12.164000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.016979  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000224  Prepare_Push: 0.002236  dn1_0_First_Result_Fetch: 0.006678  dn1_0_Last_Result_Fetch: 0.000703  Generate_New_Query: 0.000866  dn1_1_First_Result_Fetch: 0.004532  dn1_1_Last_Result_Fetch: 0.000879
Write_Client: 0.001002
SET timestamp=1535017212146;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:13.134000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.010213  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000145  Prepare_Push: 0.001520  dn1_First_Result_Fetch: 0.007996  dn1_Last_Result_Fetch: 0.000201  Write_Client: 0.000551
SET timestamp=1535017213134;
select * from sharding_two_node;
# Time: 2018-08-23T17:40:13.153000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.014292  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000080  Prepare_Push: 0.002394  dn2_First_Result_Fetch: 0.008839  dn1_First_Result_Fetch: 0.008837  dn2_Last_Result_Fetch: 0.001424  dn1_Last_Result_Fetch: 0.002407  Write_Client: 0.002945
SET timestamp=1535017213212;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:13.240000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.027695  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000067  Prepare_Push: 0.000682  dn2_First_Result_Fetch: 0.026582  dn2_Last_Result_Fetch: 0.000078  Write_Client: 0.000364
SET timestamp=1535017213246;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:13.321000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.076093  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000231  Prepare_Push: 0.001334  dn2_First_Result_Fetch: 0.035072  dn1_First_Result_Fetch: 0.035074  dn2_Last_Result_Fetch: 0.018756  dn1_Last_Result_Fetch: 0.001263  Write_Client: 0.039457
SET timestamp=1535017213321;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:13.348000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.026278  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000335  Prepare_Push: 0.001249  dn1_0_First_Result_Fetch: 0.011028  dn1_0_Last_Result_Fetch: 0.009279  dn2_0_First_Result_Fetch: 0.019200  dn2_0_Last_Result_Fetch: 0.003441  Write_Client: 0.004600
SET timestamp=1535017213348;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:13.381000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.021952  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000662  Prepare_Push: 0.003189  dn1_0_First_Result_Fetch: 0.014453  dn1_0_Last_Result_Fetch: 0.001013  Generate_New_Query: 0.000911  dn1_1_First_Result_Fetch: 0.005703  dn1_1_Last_Result_Fetch: 0.001483
Write_Client: 0.002114
SET timestamp=1535017213381;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:14.163000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id:  2
# Query_time: 0.012540  Lock_time: 0.000000 Rows_sent: 0  Rows_examined: 0  Read_SQL: 0.000197  Prepare_Push: 0.001303  dn2_First_Result_Fetch: 0.006452  dn1_First_Result_Fetch: 0.007858  dn2_Last_Result_Fetch: 0.004065  dn1_Last_Result_Fetch: 0.002960  Write_Client: 0.004588
SET timestamp=1535017214163;
select * from sharding_two_node;
```

```

# Time: 2018-08-23T17:40:14.220000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.027587 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000051 Prepare_Push: 0.006744 dn1_First_Result_Fetch: 0.026441 dn1_Last_Result_Fetch: 0.000104 Write_Client: 0.000350
SET timestamp=1535017214220;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:14.253000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.031984 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000070 Prepare_Push: 0.001144 dn2_First_Result_Fetch: 0.030202 dn2_Last_Result_Fetch: 0.000182 Write_Client: 0.000568
SET timestamp=1535017214253;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:14.292000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.037327 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.0000272 Prepare_Push: 0.001316 dn2_First_Result_Fetch: 0.014299 dn1_First_Result_Fetch: 0.014331 dn2_Last_Result_Fetch: 0.000148 dn1_Last_Result_Fetch: 0.000753 Write_Client: 0.021440
SET timestamp=1535017214292;
insert into sharding_two_node values(15,'15','15),(519,'519','519);
# Time: 2018-08-23T17:40:14.303000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.010244 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000050 Prepare_Push: 0.001101 dn1_0_First_Result_Fetch: 0.004540 dn1_0_Last_Result_Fetch: 0.002781 dn2_0_First_Result_Fetch: 0.004708 dn2_0_Last_Result_Fetch: 0.002592 Write_Client: 0.002092
SET timestamp=1535017214303;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:14.327000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.020178 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000109 Prepare_Push: 0.002098 dn1_0_First_Result_Fetch: 0.006720 dn1_0_Last_Result_Fetch: 0.000748 Generate_New_Query: 0.001158 dn1_1_First_Result_Fetch: 0.0008043 dn1_1_Last_Result_Fetch: 0.001147 Write_Client: 0.001269
SET timestamp=1535017214327;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:15.254000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.010569 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000076 Prepare_Push: 0.001050 dn1_First_Result_Fetch: 0.008330 dn1_Last_Result_Fetch: 0.000146 Write_Client: 0.001113
SET timestamp=1535017215254;
select * from sharding_two_node where id =1;
# Time: 2018-08-23T17:40:15.312000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.024216 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000081 Prepare_Push: 0.001295 dn1_First_Result_Fetch: 0.021938 dn1_Last_Result_Fetch: 0.000422 Write_Client: 0.000902
SET timestamp=1535017215321;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:15.351000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.027796 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000666 Prepare_Push: 0.000760 dn2_First_Result_Fetch: 0.025984 dn2_Last_Result_Fetch: 0.000094 Write_Client: 0.000386
SET timestamp=1535017215351;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:15.392000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.039805 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000217 Prepare_Push: 0.000804 dn2_First_Result_Fetch: 0.017410 dn1_First_Result_Fetch: 0.017468 dn2_Last_Result_Fetch: 0.001490 dn1_Last_Result_Fetch: 0.001223 Write_Client: 0.021374
SET timestamp=1535017215392;
insert into sharding_two_node values(15,'15','15),(519,'519','519);
# Time: 2018-08-23T17:40:15.410000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.017384 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000152 Prepare_Push: 0.001183 dn1_0_First_Result_Fetch: 0.005037 dn1_0_Last_Result_Fetch: 0.007164 dn2_0_First_Result_Fetch: 0.008156 dn2_0_Last_Result_Fetch: 0.004962 Write_Client: 0.004043
SET timestamp=1535017215410;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:15.434000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.021341 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000318 Prepare_Push: 0.002764 dn1_0_First_Result_Fetch: 0.010897 dn1_0_Last_Result_Fetch: 0.000544 Generate_New_Query: 0.000798 dn1_1_First_Result_Fetch: 0.004506 dn1_1_Last_Result_Fetch: 0.000790 Write_Client: 0.000845
SET timestamp=1535017215434;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:16.322000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.030106 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000217 Prepare_Push: 0.001253 dn1_First_Result_Fetch: 0.028330 dn1_Last_Result_Fetch: 0.000086 Write_Client: 0.000306
SET timestamp=1535017216322;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:16.353000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.030005 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001306 Prepare_Push: 0.001004 dn2_First_Result_Fetch: 0.027242 dn2_Last_Result_Fetch: 0.000140 Write_Client: 0.000453
SET timestamp=1535017216353;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:16.403000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.049615 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001459 Prepare_Push: 0.000830 dn2_First_Result_Fetch: 0.024286 dn1_First_Result_Fetch: 0.025469 dn2_Last_Result_Fetch: 0.001726 dn1_Last_Result_Fetch: 0.000853 Write_Client: 0.023039
SET timestamp=1535017216403;
insert into sharding_two_node values(15,'15','15),(519,'519','519);
# Time: 2018-08-23T17:40:16.526000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.012102 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000337 Prepare_Push: 0.006889 dn1_0_First_Result_Fetch: 0.009370 dn1_0_Last_Result_Fetch: 0.002010 dn2_0_First_Result_Fetch: 0.009160 dn2_0_Last_Result_Fetch: 0.001779 Write_Client: 0.109753
SET timestamp=1535017216526;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:16.569000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.030306 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001534 Prepare_Push: 0.001759 dn1_0_First_Result_Fetch: 0.011846 dn1_0_Last_Result_Fetch: 0.001663 Generate_New_Query: 0.003223 dn1_1_First_Result_Fetch: 0.006428 dn1_1_Last_Result_Fetch: 0.002601 Write_Client: 0.002291
SET timestamp=1535017216560;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:17.325000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.017545 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000621 Prepare_Push: 0.002335 dn1_First_Result_Fetch: 0.008121 dn1_Last_Result_Fetch: 0.000277 Write_Client: 0.000857
SET timestamp=1535017217325;
select * from sharding_two_node where id =1;
# Time: 2018-08-23T17:40:17.390000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.026216 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000854 Prepare_Push: 0.000904 dn1_First_Result_Fetch: 0.024157 dn1_Last_Result_Fetch: 0.000081 Write_Client: 0.000301
SET timestamp=1535017217390;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:17.411000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.020095 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000699 Prepare_Push: 0.000711 dn2_First_Result_Fetch: 0.017634 dn2_Last_Result_Fetch: 0.000132 Write_Client: 0.001051
SET timestamp=1535017217411;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:17.491000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.078505 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001702 Prepare_Push: 0.000763 dn2_First_Result_Fetch: 0.018547 dn1_First_Result_Fetch: 0.018482 dn2_Last_Result_Fetch: 0.036637 dn1_Last_Result_Fetch: 0.000566 Write_Client: 0.057558
SET timestamp=1535017217491;
insert into sharding_two_node values(15,'15','15),(519,'519','519);
# Time: 2018-08-23T17:40:17.518000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.026112 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000686 Prepare_Push: 0.000872 dn1_0_First_Result_Fetch: 0.007054 dn1_0_Last_Result_Fetch: 0.001072 dn2_0_First_Result_Fetch: 0.005839 dn2_0_Last_Result_Fetch: 0.017248 Write_Client: 0.016586
SET timestamp=1535017217518;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:17.558000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.038199 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000966 Prepare_Push: 0.005189 dn1_0_First_Result_Fetch: 0.013137 dn1_0_Last_Result_Fetch: 0.001134 Generate_New_Query: 0.003973 dn1_1_First_Result_Fetch: 0.010228 dn1_1_Last_Result_Fetch: 0.003564 Write_Client: 0.002115
SET timestamp=1535017217558;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:18.353000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.019048 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.003008 Prepare_Push: 0.000844 dn2_First_Result_Fetch: 0.006415 dn1_First_Result_Fetch: 0.009082 dn2_Last_Result_Fetch: 0.000323 dn1_Last_Result_Fetch: 0.005902 Write_Client: 0.008781
SET timestamp=1535017218353;
select * from sharding_two_node;
# Time: 2018-08-23T17:40:18.410000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.025498 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000060 Prepare_Push: 0.000696 dn1_First_Result_Fetch: 0.024394 dn1_Last_Result_Fetch: 0.000084 Write_Client: 0.000348
SET timestamp=1535017218410;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:18.430000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.018794 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000047 Prepare_Push: 0.001301 dn2_First_Result_Fetch: 0.017073 dn2_Last_Result_Fetch: 0.000099 Write_Client: 0.000373
SET timestamp=1535017218430;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:18.471000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.039810 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000052 Prepare_Push: 0.000661 dn2_First_Result_Fetch: 0.019799 dn1_First_Result_Fetch: 0.019923 dn2_Last_Result_Fetch: 0.000698 dn1_Last_Result_Fetch: 0.000814 Write_Client: 0.019298
SET timestamp=1535017218471;
insert into sharding_two_node values(15,'15','15),(519,'519','519);
# Time: 2018-08-23T17:40:18.484000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.012214 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000047 Prepare_Push: 0.001782 dn1_0_First_Result_Fetch: 0.007109 dn1_0_Last_Result_Fetch: 0.001544 dn2_0_First_Result_Fetch: 0.005518 dn2_0_Last_Result_Fetch: 0.001470 Write_Client: 0.003568
SET timestamp=1535017218484;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:18.507000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.019695 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000046 Prepare_Push: 0.001448 dn1_0_First_Result_Fetch: 0.006244 dn1_0_Last_Result_Fetch: 0.000988 Generate_New_Query: 0.001564 dn1_1_First_Result_Fetch: 0.007080 dn1_1_Last_Result_Fetch: 0.001306 Write_Client: 0.001137
SET timestamp=1535017218507;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:19.351000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.020937 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000059 Prepare_Push: 0.000800 dn1_First_Result_Fetch: 0.019607 dn1_Last_Result_Fetch: 0.000169 Write_Client: 0.000472
SET timestamp=1535017219351;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:19.370000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.018011 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001184 Prepare_Push: 0.000583 dn2_First_Result_Fetch: 0.015894 dn2_Last_Result_Fetch: 0.000129 Write_Client: 0.000351
SET timestamp=1535017219370;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:19.412000Z

```

```

# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.041319 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000689 Prepare_Push: 0.000573 dn2_First_Result_Fetch: 0.017735 dn1_First_Result_Fetch: 0.017876 dn2_Last_Result_Fetch: 0.000601 dn1_Last_Result_Fetch: 0.000806 Write_Client: 0.022322
SET timestamp=1535017219412;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:19.423000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.010663 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000200 Prepare_Push: 0.001136 dn1_0_First_Result_Fetch: 0.006820 dn1_0_Last_Result_Fetch: 0.000694 dn2_0_First_Result_Fetch: 0.003519 dn2_0_Last_Result_Fetch: 0.003944 Write_Client: 0.001443
SET timestamp=1535017219423;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:19.450000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.027592 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000182 Prepare_Push: 0.012798 dn1_0_First_Result_Fetch: 0.005960 dn1_0_Last_Result_Fetch: 0.000530 Generate_New_Query: 0.000811 dn1_1_First_Result_Fetch: 0.005659 dn1_1_Last_Result_Fetch: 0.000926 Write_Client: 0.001101
SET timestamp=1535017219454;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:20.312000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.025903 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001470 Prepare_Push: 0.000887 dn1_First_Result_Fetch: 0.022114 dn1_Last_Result_Fetch: 0.000197 Write_Client: 0.001433
SET timestamp=1535017220312;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:20.342000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.028503 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.005643 Prepare_Push: 0.001172 dn2_First_Result_Fetch: 0.021342 dn2_Last_Result_Fetch: 0.000074 Write_Client: 0.000346
SET timestamp=1535017220342;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:20.381000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.037424 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000641 Prepare_Push: 0.000959 dn2_First_Result_Fetch: 0.015139 dn1_First_Result_Fetch: 0.015238 dn2_Last_Result_Fetch: 0.000795 dn1_Last_Result_Fetch: 0.000956 Write_Client: 0.020685
SET timestamp=1535017220381;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:20.408000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.016143 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000122 Prepare_Push: 0.001979 dn1_0_First_Result_Fetch: 0.004408 dn1_0_Last_Result_Fetch: 0.000484 Generate_New_Query: 0.000965 dn1_1_First_Result_Fetch: 0.006059 dn1_1_Last_Result_Fetch: 0.001553 Write_Client: 0.000755
SET timestamp=1535017220408;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:21.214000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.023376 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000073 Prepare_Push: 0.001306 dn1_First_Result_Fetch: 0.021694 dn1_Last_Result_Fetch: 0.000081 Write_Client: 0.000302
SET timestamp=1535017221214;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:21.241000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.025408 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000083 Prepare_Push: 0.001029 dn2_First_Result_Fetch: 0.023856 dn2_Last_Result_Fetch: 0.000122 Write_Client: 0.000440
SET timestamp=1535017221241;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:21.281000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.038482 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000087 Prepare_Push: 0.000871 dn2_First_Result_Fetch: 0.016690 dn1_First_Result_Fetch: 0.016708 dn2_Last_Result_Fetch: 0.000579 dn1_Last_Result_Fetch: 0.000891 Write_Client: 0.020835
SET timestamp=1535017221281;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:21.293000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.01657 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000615 Prepare_Push: 0.001320 dn1_0_First_Result_Fetch: 0.006906 dn1_0_Last_Result_Fetch: 0.001589 dn2_0_First_Result_Fetch: 0.005105 dn2_0_Last_Result_Fetch: 0.001548 Write_Client: 0.003341
SET timestamp=1535017221293;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:21.312000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.017169 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000635 Prepare_Push: 0.001669 dn1_0_First_Result_Fetch: 0.006997 dn1_0_Last_Result_Fetch: 0.000728 Generate_New_Query: 0.001037 dn1_1_First_Result_Fetch: 0.004816 dn1_1_Last_Result_Fetch: 0.000709 Write_Client: 0.000703
SET timestamp=1535017221312;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:22.150000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.026153 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000180 Prepare_Push: 0.000771 dn1_First_Result_Fetch: 0.024940 dn1_Last_Result_Fetch: 0.000061 Write_Client: 0.000261
SET timestamp=1535017222150;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:22.170000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.019181 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000180 Prepare_Push: 0.000642 dn2_First_Result_Fetch: 0.018060 dn2_Last_Result_Fetch: 0.000088 Write_Client: 0.000299
SET timestamp=1535017222170;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:22.220000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.049834 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000261 Prepare_Push: 0.000735 dn2_First_Result_Fetch: 0.019862 dn1_First_Result_Fetch: 0.019807 dn2_Last_Result_Fetch: 0.000418 dn1_Last_Result_Fetch: 0.000655 Write_Client: 0.029031
SET timestamp=1535017222220;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:22.240000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.019128 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000934 Prepare_Push: 0.002731 dn1_0_First_Result_Fetch: 0.013430 dn1_0_Last_Result_Fetch: 0.000521 dn2_0_First_Result_Fetch: 0.003296 dn2_0_Last_Result_Fetch: 0.001243 Write_Client: 0.011072
SET timestamp=1535017222240;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:22.270000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.028479 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.003233 Prepare_Push: 0.004986 dn1_0_First_Result_Fetch: 0.009870 dn1_0_Last_Result_Fetch: 0.001172 Generate_New_Query: 0.001590 dn1_1_First_Result_Fetch: 0.006060 dn1_1_Last_Result_Fetch: 0.000771 Write_Client: 0.000700
SET timestamp=1535017222270;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:23.097000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.053956 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000056 Prepare_Push: 0.001034 dn1_First_Result_Fetch: 0.052343 dn1_Last_Result_Fetch: 0.000174 Write_Client: 0.000523
SET timestamp=1535017223097;
select * from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:23.110000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.010839 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000499 Prepare_Push: 0.000680 dn2_First_Result_Fetch: 0.006349 dn1_First_Result_Fetch: 0.009082 dn2_Last_Result_Fetch: 0.000270 dn1_Last_Result_Fetch: 0.000333 Write_Client: 0.003311
SET timestamp=1535017223110;
select * from sharding_two_node;
# Time: 2018-08-23T17:40:23.181000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.027573 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000109 Prepare_Push: 0.000980 dn2_First_Result_Fetch: 0.026156 dn2_Last_Result_Fetch: 0.000086 Write_Client: 0.000328
SET timestamp=1535017223181;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:23.231000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.049380 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.002435 Prepare_Push: 0.000670 dn2_First_Result_Fetch: 0.025278 dn1_First_Result_Fetch: 0.025242 dn2_Last_Result_Fetch: 0.000392 dn1_Last_Result_Fetch: 0.000629 Write_Client: 0.021032
SET timestamp=1535017223231;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:23.268000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.025207 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000060 Prepare_Push: 0.001492 dn1_0_First_Result_Fetch: 0.007693 dn1_0_Last_Result_Fetch: 0.000752 Generate_New_Query: 0.001946 dn1_1_First_Result_Fetch: 0.008776 dn1_1_Last_Result_Fetch: 0.0005040
Write_Client: 0.001884
SET timestamp=1535017223268;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:24.121000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.039883 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000483 Prepare_Push: 0.000584 dn2_First_Result_Fetch: 0.017241 dn1_First_Result_Fetch: 0.017320 dn2_Last_Result_Fetch: 0.000603 dn1_Last_Result_Fetch: 0.000767 Write_Client: 0.021575
SET timestamp=1535017224182;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:24.141000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.019191 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000072 Prepare_Push: 0.000673 dn2_First_Result_Fetch: 0.017923 dn2_Last_Result_Fetch: 0.000092 Write_Client: 0.000522
SET timestamp=1535017224141;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:24.182000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.039883 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000483 Prepare_Push: 0.000584 dn2_First_Result_Fetch: 0.017241 dn1_First_Result_Fetch: 0.017320 dn2_Last_Result_Fetch: 0.000603 dn1_Last_Result_Fetch: 0.000767 Write_Client: 0.021575
SET timestamp=1535017224182;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:24.196000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.012406 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000073 Prepare_Push: 0.000958 dn1_0_First_Result_Fetch: 0.008102 dn1_0_Last_Result_Fetch: 0.001255 dn2_0_First_Result_Fetch: 0.007566 dn2_0_Last_Result_Fetch: 0.001772 Write_Client: 0.002300
SET timestamp=1535017224196;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:24.218000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.029579 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000106 Prepare_Push: 0.000882 dn1_First_Result_Fetch: 0.028241 dn1_Last_Result_Fetch: 0.000069 Write_Client: 0.000351
SET timestamp=1535017225093;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:25.121000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.027422 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001867 Prepare_Push: 0.001330 dn2_First_Result_Fetch: 0.023887 dn2_Last_Result_Fetch: 0.000102 Write_Client: 0.000338
SET timestamp=1535017225121;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:25.161000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.038859 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000050 Prepare_Push: 0.000753 dn2_First_Result_Fetch: 0.019091 dn1_First_Result_Fetch: 0.019189 dn2_Last_Result_Fetch: 0.000582 dn1_Last_Result_Fetch: 0.000560 Write_Client: 0.018965
SET timestamp=1535017225161;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:25.191000Z

```

```

# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.016379 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000043 Prepare_Push: 0.001276 dn1_0_First_Result_Fetch: 0.007469 dn1_0_Last_Result_Fetch: 0.000678 Generate_New_Query: 0.001327 dn1_1_First_Result_Fetch: 0.003927 dn1_1_Last_Result_Fetch: 0.000787
Write_Client: 0.000893
SET timestamp=1535017225191;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
# Time: 2018-08-23T17:40:26.026000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.029878 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000162 Prepare_Push: 0.000916 dn1_First_Result_Fetch: 0.028497 dn1_Last_Result_Fetch: 0.000084 Write_Client: 0.000303
SET timestamp=1535017226026;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:26.051000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.024231 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001105 Prepare_Push: 0.000469 dn2_First_Result_Fetch: 0.022188 dn2_Last_Result_Fetch: 0.000100 Write_Client: 0.000470
SET timestamp=1535017226051;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:26.091000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.03762 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.001669 Prepare_Push: 0.001915 dn2_First_Result_Fetch: 0.018107 dn1_First_Result_Fetch: 0.018187 dn2_Last_Result_Fetch: 0.000633 dn1_Last_Result_Fetch: 0.000832 Write_Client: 0.018071
SET timestamp=1535017226091;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:26.105000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.012664 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000261 Prepare_Push: 0.000935 dn1_0_First_Result_Fetch: 0.007328 dn1_0_Last_Result_Fetch: 0.000733 dn2_0_First_Result_Fetch: 0.006229 dn2_0_Last_Result_Fetch: 0.002592 Write_Client: 0.003554
SET timestamp=1535017226105;
select count(*) from sharding_two_node;
# Time: 2018-08-23T17:40:26.134000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.028335 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000046 Prepare_Push: 0.003442 dn1_0_First_Result_Fetch: 0.009563 dn1_0_Last_Result_Fetch: 0.001069 Generate_New_Query: 0.001856 dn1_1_First_Result_Fetch: 0.010875 dn1_1_Last_Result_Fetch: 0.000798
Write_Client: 0.000712
SET timestamp=1535017226134;
select count(*) from sharding_two_node where id =1;
# Time: 2018-08-23T17:40:26.1874000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.014882 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000067 Prepare_Push: 0.001351 dn1_First_Result_Fetch: 0.013084 dn1_Last_Result_Fetch: 0.000137 Write_Client: 0.000381
SET timestamp=1535017226859;
select * from sharding_two_node where id =1;
# Time: 2018-08-23T17:40:26.1874000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.010509 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000066 Prepare_Push: 0.001761 dn2_First_Result_Fetch: 0.006921 dn1_First_Result_Fetch: 0.008256 dn2_Last_Result_Fetch: 0.000279 dn1_Last_Result_Fetch: 0.000211 Write_Client: 0.001761
SET timestamp=1535017226874;
select * from sharding_two_node;
# Time: 2018-08-23T17:40:26.1931000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.028690 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000237 Prepare_Push: 0.001126 dn1_First_Result_Fetch: 0.026194 dn1_Last_Result_Fetch: 0.000640 Write_Client: 0.001133
SET timestamp=1535017226931;
delete from sharding_two_node where id =15;
# Time: 2018-08-23T17:40:26.051000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.018818 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000342 Prepare_Push: 0.001671 dn2_First_Result_Fetch: 0.016482 dn2_Last_Result_Fetch: 0.000063 Write_Client: 0.000323
SET timestamp=1535017226951;
delete from sharding_two_node where id =519;
# Time: 2018-08-23T17:40:26.091000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.023174 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000342 Prepare_Push: 0.001671 dn2_First_Result_Fetch: 0.016482 dn2_Last_Result_Fetch: 0.000063 Write_Client: 0.000323
SET timestamp=1535017226951;
insert into sharding_two_node values(15,'15','15),(519,'519',519);
# Time: 2018-08-23T17:40:27.032000Z
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
# Query_time: 0.029495 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.000064 Prepare_Push: 0.001349 dn1_0_First_Result_Fetch: 0.005745 dn1_0_Last_Result_Fetch: 0.000632 Generate_New_Query: 0.001056 dn1_1_First_Result_Fetch: 0.018101 dn1_1_Last_Result_Fetch: 0.002282
Write_Client: 0.000863
SET timestamp=1535017227032;
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);

```

### 2.20.3.1 mysqldumpslow :

```

Reading mysql slow query log from /tmp/slow3.log
Count: 17 Time=0.05s (0s) Lock=0.00s (0s) Rows=0.0 (0), root[root]@[0:0:0:0:0:0:1]
  insert into sharding_two_node values(N,'S',N),(N,'S',N)

Count: 13 Time=0.05s (0s) Lock=0.00s (0s) Rows=0.0 (0), root[root]@[0:0:0:0:0:0:1]
  select count(*) from sharding_two_node

Count: 6 Time=0.04s (0s) Lock=0.00s (0s) Rows=0.0 (0), root[root]@[0:0:0:0:0:0:1]
  select * from sharding_two_node where id =N

Count: 33 Time=0.03s (0s) Lock=0.00s (0s) Rows=0.0 (0), root[root]@[0:0:0:0:0:0:1]
  delete from sharding_two_node where id =N

Count: 17 Time=0.03s (0s) Lock=0.00s (0s) Rows=0.0 (0), root[root]@[0:0:0:0:0:0:1]
  select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=N)

Count: 6 Time=0.02s (0s) Lock=0.00s (0s) Rows=0.0 (0), root[root]@[0:0:0:0:0:0:1]
  select * from sharding_two_node

```

### 2.20.3.2 pt-query-digest :

```

# 710ms user time, 70ms system time, 23.35M rss, 68.36M vsz
# Current date: Thu Aug 23 17:48:25 2018
# Hostname: 10-186-24-63
# Files: /tmp/slow_query4.log
# Overall: 92 total, 6 unique, 5.41 QPS, 0.18x concurrency _____
# Time range: 2018-08-23T17:40:10 to 2018-08-23T17:40:27
# Attribute          total    min     max   avg   95% stddev median
# ======          ======  ======  ======  ======  ======  ======  ======
# Exec time          3s    10ms   339ms   34ms   75ms   37ms   27ms
# Lock time          0      0      0      0      0      0      0
# Rows sent          0      0      0      0      0      0      0
# Rows examine       0      0      0      0      0      0      0
# Query size        4.91k   31     94    54.64   92.72   20.87   42.48
# Generate New      0.03   0.00   0.00   0.00   0.00   0.00   0.00
# Prepare Push       0.56   0.00   0.26   0.01   0.01   0.03   0.00
# Read SQL          0.07   0.00   0.01   0.00   0.00   0.00   0.00
# Write Client      0.70   0.00   0.11   0.01   0.02   0.01   0.00
# dn1_0 First       0.29   0.00   0.05   0.01   0.01   0.01   0.01
# dn1_0 Last R     0.07   0.00   0.03   0.00   0.01   0.01   0.00
# dn1_1 First       0.12   0.00   0.02   0.01   0.01   0.00   0.01
# dn1_1 Last R     0.03   0.00   0.01   0.00   0.00   0.00   0.00
# dn1 First Re      0.93   0.01   0.05   0.02   0.03   0.01   0.02
# dn1 Last Res      0.04   0.00   0.01   0.00   0.00   0.00   0.00
# dn2_0 First       0.13   0.00   0.04   0.01   0.02   0.01   0.01
# dn2_0 Last R     0.08   0.00   0.03   0.01   0.02   0.01   0.00
# dn2_1 First Re    0.80   0.01   0.06   0.02   0.03   0.01   0.02
# dn2_1 Last Res    0.09   0.00   0.04   0.00   0.01   0.01   0.00

# Profile
# Rank Query ID          Response time Calls R/Call V/M   I
# ====== ======          ====== ====== ====== ====== ====== =====
# === ====== ====== ====== ====== ====== ====== ====== ====== =====
# 1 0x13233F8ADA41C6E2D889AEE0C2B... 0.8815 28.1% 33 0.0267 0.00 DELETE sharding_two_node
# 2 0xF6BD1646E487184E8FD3B83912... 0.8525 27.1% 17 0.0501 0.01 INSERT sharding_two_node
# 3 0xB460B13C53699C853F7CBA6D20B... 0.6306 20.1% 13 0.0485 0.15 SELECT sharding_two_node
# 4 0x3FB41587E746A475282C1ED2606... 0.4335 13.8% 17 0.0255 0.00 SELECT sharding_two_node
# 5 0x64CD91DDFC4E1DD7A22E312C72... 0.2399 7.6% 6 0.0400 0.05 SELECT sharding_two_node
# MISC 0xMISC           0.1028 3.3% 6 0.0171 0.0 <1 ITEMS>

# Query 1: 2.06 QPS, 0.06x concurrency, ID 0x13233F8ADA41C6E2D889AEE0C2B6CB5 at byte 943
# Scores: V/M = 0.00
# Time range: 2018-08-23T17:40:10 to 2018-08-23T17:40:26
# Attribute          pct   total    min     max   avg   95% stddev median
# ======          ======  ======  ======  ======  ======  ======  ======
# Count            35     33
# Exec time        28    882ms   18ms   45ms   27ms   31ms   5ms   27ms
# Lock time        0      0      0      0      0      0      0      0
# Rows sent        0      0      0      0      0      0      0      0
# Rows examine     0      0      0      0      0      0      0      0
# Query size       27    1.37k   42     43    42.52   42.48   0.50   42.48
# Prepare Push     6     0.04   0.00   0.00   0.00   0.00   0.00   0.00
# Read SQL         35    0.03   0.00   0.01   0.00   0.00   0.00   0.00
# Write Client     2     0.02   0.00   0.00   0.00   0.00   0.00   0.00
# dn1_0 First      45    0.42   0.02   0.04   0.03   0.03   0.01   0.03
# dn1_0 Last Res   7     0.00   0.00   0.00   0.00   0.00   0.00   0.00
# dn1_1 First Re   47    0.38   0.02   0.03   0.02   0.03   0.00   0.02
# dn1_1 Last Res   3     0.00   0.00   0.00   0.00   0.00   0.00   0.00
# String:
# Hosts          0:0:0:0:0:0:0:0:1
# Users          root
# Query_time distribution
#   1us
#   10us
#   100us
#   1ms

```

```

# 10ms #####
# 100ms
#   1s
# 10s+
# Tables
#   SHOW TABLE STATUS LIKE 'sharding_two_node'\G
#   SHOW CREATE TABLE `sharding_two_node`\G
delete from sharding_two_node where id =15\G
# Converted for EXPLAIN
# EXPLAIN /*!50106 PARTITIONS*/
select * from sharding_two_node where id =15\G

# Query 2: 1.06 QPS, 0.05x concurrency, ID 0xF68D16B46E487184E8FD3BB3912A3658 at byte 1690
# Scores: V/M = 0.01
# Time range: 2018-08-23T17:40:10 to 2018-08-23T17:40:26
# Attribute   pct total  min   max   avg   95% stddev median
# ====== ===== ===== ===== ===== ===== ===== ===== =====
# Count      18    17
# Exec time   27  853ms  37ms 117ms  50ms  78ms  21ms  40ms
# Lock time    0     0     0     0     0     0     0     0     0
# Rows sent    0     0     0     0     0     0     0     0     0
# Rows examine  0     0     0     0     0     0     0     0     0
# Query size   21  1.06k   64    64   64    64    64    64
# Prepare Push  7   0.04   0.00   0.02   0.00   0.00   0.01   0.00
# Read SQL    32   0.02   0.00   0.01   0.00   0.00   0.00   0.00
# Write Client 63   0.45   0.02   0.06   0.03   0.05   0.01   0.02
# dnd First Re 37   0.35   0.01   0.04   0.02   0.03   0.00   0.02
# dnd Last Res 40   0.01   0.00   0.00   0.00   0.00   0.00   0.00
# dn2 First Re 46   0.37   0.01   0.06   0.02   0.03   0.01   0.02
# dn2 Last Res 72   0.07   0.00   0.04   0.00   0.02   0.01   0.00
# String:
# Hosts      0:0:0:0:0:0:1
# Users       root
# Query_time distribution
#   1us
# 10us
# 100us
#   1ms
# 10ms #####
# 100ms #####
#   1s
# 10s+
# Tables
#   SHOW TABLE STATUS LIKE 'sharding_two_node'\G
#   SHOW CREATE TABLE `sharding_two_node`\G
insert into sharding_two_node values(15,'15',15) /*... omitted ...*/\G

# Query 3: 0.81 QPS, 0.04x concurrency, ID 0xB46D813C53609C853F7CBA6D2DB4047C at byte 2152
# Scores: V/M = 0.15
# Time range: 2018-08-23T17:40:10 to 2018-08-23T17:40:26
# Attribute   pct total  min   max   avg   95% stddev median
# ====== ===== ===== ===== ===== ===== ===== ===== =====
# Count      14    13
# Exec time   20  631ms  10ms 339ms  49ms 116ms  84ms  12ms
# Lock time    0     0     0     0     0     0     0     0     0
# Rows sent    0     0     0     0     0     0     0     0     0
# Rows examine  0     0     0     0     0     0     0     0     0
# Query size   9   494    38    38   38    38    38    38
# Prepare Push 48   0.27   0.06   0.26   0.02   0.00   0.07   0.00
# Read SQL     5   0.00   0.00   0.00   0.00   0.00   0.00   0.00
# Write Client 24   0.17   0.00   0.11   0.01   0.02   0.03   0.00
# dnd 0 First  48   0.14   0.00   0.05   0.01   0.01   0.01   0.01
# dnd 0 Last R 80   0.06   0.00   0.03   0.00   0.01   0.01   0.00
# dn2 0 First 100   0.13   0.00   0.04   0.01   0.02   0.01   0.01
# dn2 0 Last R 100   0.08   0.00   0.03   0.01   0.02   0.01   0.00
# String:
# Hosts      0:0:0:0:0:0:1
# Users       root
# Query_time distribution
#   1us
# 10us
# 100us
#   1ms
# 10ms #####
# 100ms #####
#   1s
# 10s+
# Tables
#   SHOW TABLE STATUS LIKE 'sharding_two_node'\G
#   SHOW CREATE TABLE `sharding_two_node`\G
# EXPLAIN /*!50106 PARTITIONS*/
select count(*) from sharding_two_node\G

# Query 4: 1 QPS, 0.03x concurrency, ID 0x3FB41587E746A475282C1ED2606795FB at byte 2596
# Scores: V/M = 0.00
# Time range: 2018-08-23T17:40:10 to 2018-08-23T17:40:26
# Attribute   pct total  min   max   avg   95% stddev median
# ====== ===== ===== ===== ===== ===== ===== ===== =====
# Count      18    17
# Exec time   13  434ms  16ms 47ms  26ms  38ms   8ms  26ms
# Lock time    0     0     0     0     0     0     0     0     0
# Rows sent    0     0     0     0     0     0     0     0     0
# Rows examine  0     0     0     0     0     0     0     0     0
# Query size   31  1.56k   94    94   94    94    94    94
# Generate New 100   0.03   0.00   0.00   0.00   0.00   0.00   0.00
# Prepare Push 13   0.07   0.00   0.03   0.00   0.01   0.01   0.00
# Read SQL    12   0.01   0.00   0.00   0.00   0.00   0.00   0.00
# Write Client  2   0.02   0.00   0.00   0.00   0.00   0.00   0.00
# dnd 0 First  51   0.15   0.00   0.01   0.01   0.01   0.00   0.01
# dnd 0 Last R 19   0.01   0.00   0.00   0.00   0.00   0.00   0.00
# dn2 1 First 100   0.12   0.00   0.02   0.01   0.01   0.00   0.01
# dn2 1 Last R 100   0.03   0.00   0.01   0.00   0.00   0.00   0.00
# String:
# Hosts      0:0:0:0:0:0:1
# Users       root
# Query_time distribution
#   1us
# 10us
# 100us
#   1ms
# 10ms #####
# 100ms #####
#   1s
# 10s+
# Tables
#   SHOW TABLE STATUS LIKE 'sharding_two_node'\G
#   SHOW CREATE TABLE `sharding_two_node`\G
# EXPLAIN /*!50106 PARTITIONS*/
select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1)\G

# Query 5: 0.38 QPS, 0.01x concurrency, ID 0x04CDF910DFC4E1DD7A22E312C72C268D at byte 0
# Scores: V/M = 0.05
# Time range: 2018-08-23T17:40:10 to 2018-08-23T17:40:26
# Attribute   pct total  min   max   avg   95% stddev median
# ====== ===== ===== ===== ===== ===== ===== ===== =====
# Count      6     6
# Exec time   7   240ms  10ms 133ms  40ms 128ms  43ms  35ms
# Lock time    0     0     0     0     0     0     0     0     0
# Rows sent    0     0     0     0     0     0     0     0     0
# Rows examine  0     0     0     0     0     0     0     0     0
# Query size   5   258    43    43   43    43    43    43
# Prepare Push 22   0.12   0.00   0.12   0.02   0.12   0.04   0.00
# Read SQL    9   0.01   0.00   0.01   0.00   0.01   0.00   0.00
# Write Client  0   0.01   0.00   0.00   0.00   0.00   0.00   0.00
# dnd First Re 11   0.10   0.01   0.05   0.02   0.05   0.02   0.01
# dn2 Last Res 6   0.00   0.00   0.00   0.00   0.00   0.00   0.00
# String:
# Hosts      0:0:0:0:0:0:1
# Users       root
# Query_time distribution
#   1us
# 10us
# 100us
#   1ms
# 10ms #####
# 100ms #####
#   1s
# 10s+
# Tables
#   SHOW TABLE STATUS LIKE 'sharding_two_node'\G
#   SHOW CREATE TABLE `sharding_two_node`\G
# EXPLAIN /*!50106 PARTITIONS*/
select * from sharding_two_node where id =1\G

```

### 2.20.3.3

mysqldumpslow5.6mysqldumpslow'190428 10:28:16'Timedble'2019-04-28T10:28:16.515000Z'issue:

<https://github.com/actiontech/dble/issues/908>

### 2.20.4

#### 2.20.4.1 MySQL

MySQL

```
/usr/local/mysql5.7.11/bin/mysqld-debug, Version: 5.7.11-debug-log (MySQL Community Server - Debug (GPL)). started with:  
Tcp port: 3320 Unix socket: /tmp/mysql_3320.sock  
Time Id Command Argument  
# Time: 2018-05-15T10:53:23.798040Z  
# User@Host: action[action] @ [192.168.2.206] Id: 436  
# Query_time: 296.145816 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0  
use test;  
SET timestamp=1526381603;  
drop table sharding_two_node;  
# Time: 2018-05-15T11:32:25.549290Z  
# User@Host: action[action] @ [192.168.2.206] Id: 451  
# Query_time: 129.555883 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0  
use nosharding;  
SET timestamp=1526383945;  
drop table test4;  
# Time: 2018-05-15T11:32:25.550190Z  
# User@Host: action[action] @ [192.168.2.206] Id: 454  
# Query_time: 84.316518 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0  
SET timestamp=1526383945;  
insert into test4 values(1,'1');  
# Time: 2018-05-15T11:37:01.079214Z  
# User@Host: action[action] @ [192.168.2.206] Id: 483  
# Query_time: 49.571983 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0  
SET timestamp=1526384221;  
drop table test3;  
# Time: 2018-07-11T05:28:34.161405Z  
# User@Host: action[action] @ [192.168.2.206] Id: 16421  
# Query_time: 10.035706 Lock_time: 0.000000 Rows_sent: 1 Rows_examined: 0  
use test;  
SET timestamp=1531286914;  
insert into test4 values(1,'1');
```

MySQL

#### 1.1

/usr/local/mysql5.7.11/bin/mysqld-debug, Version: 5.7.11-debug-log (MySQL Community Server - Debug (GPL)). started with:

Tcp port: 3320 Unix socket: /tmp/mysql\_3320.sock

Time Id Command Argument

MySQLlog-short-formattimesession

#### 1.2 time

# Time: 2018-05-15T10:53:23.798040Z

#### 1.3 session

# User@Host: action[action] @ [192.168.2.206] Id: 436

#### 1.4

keyvalue

# Query\_time: 296.145816 Lock\_time: 0.000000 Rows\_sent: 0 Rows\_examined: 0

#### 1.5 Database changedschema

use nosharding;

#### 1.6 set(mysqlslowSET timestamp=)

SET timestamp=1526383945;

#### 1.6.1 last\_insert\_id

stmt\_depends\_on\_first\_successful\_insert\_id\_in\_prev\_stmt

last\_insert\_id=

#### 1.6.2 insert\_id

log-short-format

auto\_inc\_intervals\_in\_cur\_stmt\_for\_binlog.nb\_elements()

last\_id=

#### 1.6.3 timestamp=

#### 1.7

# administrator command

is\_command

#### 1.8 SQL

insert into test4 values(1,'1');

#### 2.20.4.2 dble

```
mysqldumpslowpt-query-digest dble
```

## 2.1

```
/FAKE_PATH/mysqld, Version: FAKE_VERSION. started with:  
Tcp port: 3320 Unix socket: FAKE_SOCK  
Time Id Command Argument
```

## 2.2 time

```
java80mysql
```

```
# Time: 2018-08-23T17:40:10.149000Z
```

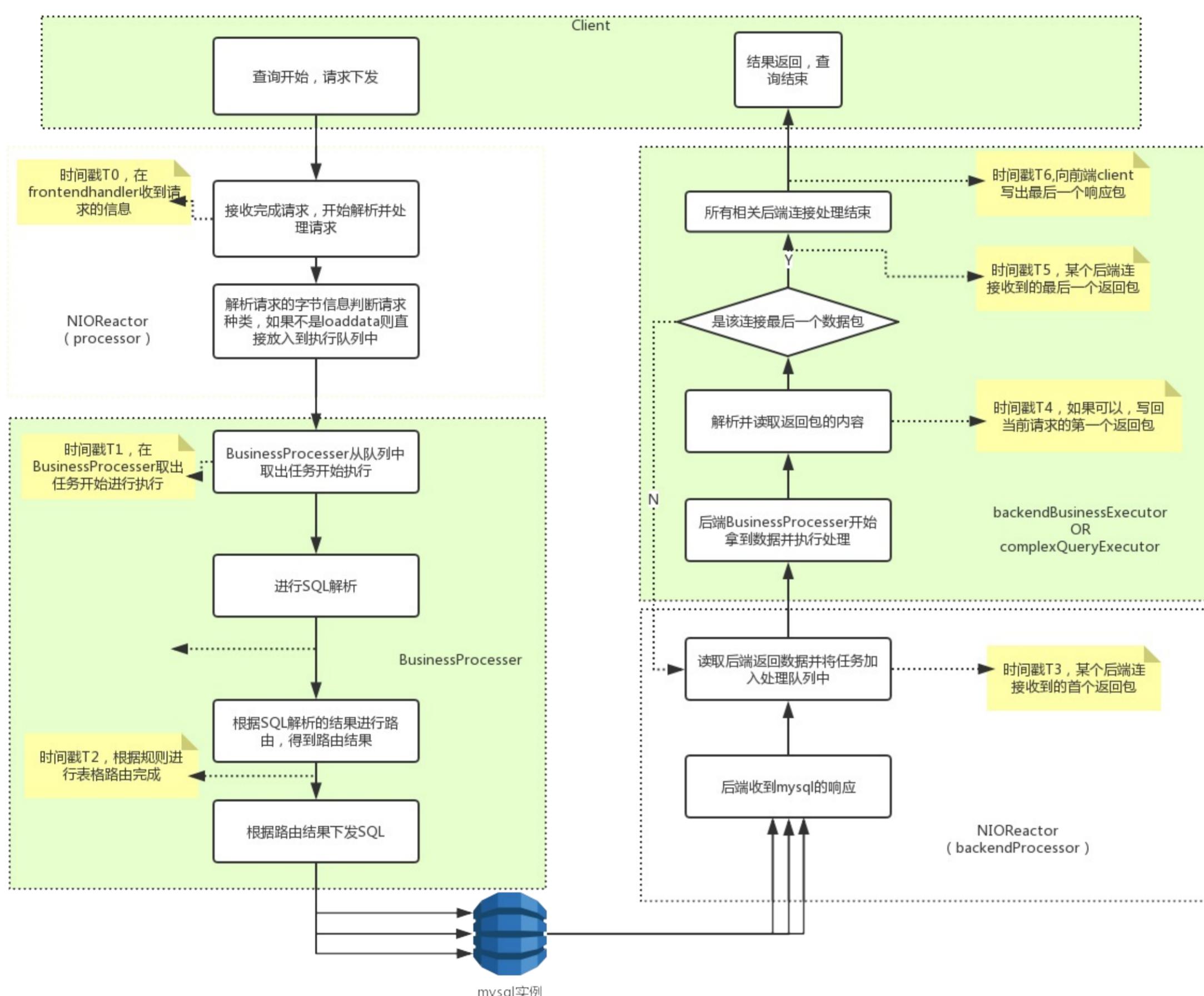
## 2.3 session

```
# User@Host: root[root] @ [0:0:0:0:0:0:1] Id: 2
```

## 2.4

keyvalue.

dble



## MySQL

```
Read_SQL: SQLSQL T1-T0
```

```
Prepare_Push/Read_SQL,T2-T1
```

```
`${shardingnodeName}_First_Result_Fetch shardingnodeNamePrepare_Push ,T3-T2 shardingnodeName
```

```
`${shardingnodeName}_Last_Result_Fetch shardingnodeName${shardingnodeName}__First_Result_Fetch,T5-T3
```

```
Write_Client,T6-T4
```

```
MySQLQuery_timedbleSQL,T6-T0  
Lock_timeRows_sentRows_examined0
```

```
# Query_time: 0.116672 Lock_time: 0.000000 Rows_sent: 0 Rows_examined: 0 Read_SQL: 0.013625 Prepare_Push: 0.024767 dn2_First_Result_Fetch: 0.056395 dn1_First_Result_Fetch: 0.026420 dn2_Last_Result_Fetch: 0.000743 dn1_Last_Result_Fetch: 0.001700 Write_Client: 0.051861
```

## 2.5 set(SET timestamp=)

```
SET timestamp=1535017210432;
```

## 2.6 SQL

```
insert into sharding_two_node values(15, '15', 15), (519, '519', 519);
```

## 2.21 SQLtrace

MySQL profiledbleSQLdbe session

### 1. trace

```
mysql> select @@trace;
+-----+
| @trace |
+-----+
| 0      |
+-----+
1 row in set (0.02 sec)
```

### 2. trace

```
mysql> set trace =1;
Query OK, 0 rows affected (0.09 sec)

mysql> select @@trace;
+-----+
| @trace |
+-----+
| 1      |
+-----+
1 row in set (0.00 sec)
```

### 3. trace

```
mysql> select * from sharding_two_node where id =1;
+-----+
| id | c_flag | c_decimal |
+-----+
| 1  | 1     | 1.0000 |
+-----+
1 row in set (0.02 sec)

mysql> show trace;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF
+-----+-----+-----+-----+-----+
| Read SQL  | 0.0       | 0.1085  | 0.1085   | -           | -
| Parse SQL | 0.1085   | 0.49607 | 0.38757  | -           | -
| Route Calculation | 0.49607 | 1.274142 | 0.778072 | -           | -
| Prepare to Push | 1.274142 | 1.569543 | 0.298401 | -           | -
| Execute SQL | 1.569543 | 18.711851 | 17.151308 | dn1         | select * from sharding_two_node where id =1
| Fetch result | 18.711851 | 18.711851 | 0.266362 | dn1         | select * from sharding_two_node where id =1
| Write to Client | 18.711851 | 19.276344 | 0.564493 | -           | -
| Over All   | 0.0       | 19.276344 | 19.276344 | -           | -
+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

### 4. trace

```
mysql> select * from sharding_two_node ;
+-----+
| id | c_flag | c_decimal |
+-----+
| 513 | 513   | 513.0000 |
| 514 | 514   | 514.0000 |
| 515 | 515   | 515.0000 |
| 516 | 516   | 516.0000 |
| 1  | 1     | 1.0000 |
| 2  | 2     | 2.0000 |
| 3  | 3     | 3.0000 |
| 4  | 4     | 4.0000 |
| 5  | 5     | 5.0000 |
| 7  | 7     | 7.0000 |
| 8  | 8     | 8.0000 |
| 9  | 9     | 9.0000 |
| 10 | 10    | 10.0000 |
| 11 | 11    | 11.0000 |
| 12 | 12    | 12.0000 |
+-----+
15 rows in set (0.01 sec)

mysql> show trace;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF
+-----+-----+-----+-----+-----+
| Read SQL  | 0.0       | 0.079175 | 0.079175 | -           | -
| Parse SQL | 0.079175 | 0.637315 | 0.55814   | -           | -
| Route Calculation | 0.637315 | 1.046389 | 0.409974 | -           | -
| Prepare to Push | 1.046389 | 1.465238 | 0.418849 | -           | -
| Execute SQL | 1.465238 | 8.141409 | 6.676171 | dn1         | SELECT * FROM sharding_two_node LIMIT 100
| Execute SQL | 1.465238 | 7.59109  | 6.125852  | dn2         | SELECT * FROM sharding_two_node LIMIT 100
| Fetch result | 8.141409 | 8.817824 | 0.676415 | dn1         | SELECT * FROM sharding_two_node LIMIT 100
| Fetch result | 7.59109  | 8.366118 | 0.775628  | dn2         | SELECT * FROM sharding_two_node LIMIT 100
| Write to Client | 7.59109  | 9.324157 | 1.733967 | -           | -
| Over All   | 0.0       | 9.324157 | 9.324157 | -           | -
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

### 5. SQLtrace,

```
mysql> insert into sharding_two_node values(15,'15',15),(519,'519',519);
Query OK, 2 rows affected (0.06 sec)

mysql> show trace;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF
+-----+-----+-----+-----+-----+
| Read SQL  | 0.0       | 0.131959 | 0.131959 | -           | -
| Parse SQL | 0.131959 | 0.601637 | 0.469678 | -           | -
| Route Calculation | 0.601637 | 0.825479 | 0.223842 | -           | -
| Prepare to Push | 0.825479 | 1.025374 | 0.199895 | -           | -
| Execute SQL | 1.025374 | 27.095675 | 26.070301 | dn1         | INSERT INTO sharding_two_node VALUES (15, '15', 15)
| Execute SQL | 1.025374 | 25.023911 | 23.998537 | dn2         | INSERT INTO sharding_two_node VALUES (519, '519', 519)
| Fetch result | 27.095675 | 27.405046 | 0.309371 | dn1         | INSERT INTO sharding_two_node VALUES (15, '15', 15)
| Fetch result | 25.023911 | 26.478398 | 0.454487 | dn2         | INSERT INTO sharding_two_node VALUES (519, '519', 519)
| Distributed Transaction Prepare | 27.405046 | 27.736411 | 0.331365 | -           | -
| Distributed Transaction Commit | 27.736411 | 57.426311 | 29.6899 | -           | -
| Write to Client | 25.023911 | 57.428266 | 32.404355 | -           | -
| Over All   | 0.0       | 57.428266 | 57.428266 | -           | -
+-----+-----+-----+-----+-----+
12 rows in set (0.00 sec)
```

### 6. trace

```
mysql> select count(*) from sharding_two_node;
+-----+
| COUNT(*) |
+-----+
| 20 |
+-----+
1 row in set (0.01 sec)

mysql> show trace;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF
+-----+-----+-----+-----+-----+
| Read SQL  | 0.0       | 0.08553  | 0.08553   | -           | -
| Parse SQL | 0.08553  | 0.56987  | 0.48434   | -           | -
| Try Route Calculation | 0.56987  | 0.71698  | 0.14711   | -           | -
| Try to Optimize | 0.71698  | 1.237487 | 0.520567  | -           | -
| Execute SQL | 1.237487 | 0.091029 | 7.853542  | dn1.0       | select COUNT(*) as `$COUNTS_rpda_0` from `sharding_two_node` LIMIT 100
| Fetch result | 0.091029 | 10.186782 | 1.095753  | dn1.0       | select COUNT(*) as `$COUNTS_rpda_0` from `sharding_two_node` LIMIT 100
| Execute SQL | 1.237487 | 8.348635 | 7.111148  | dn2.0       | select COUNT(*) as `$COUNTS_rpda_0` from `sharding_two_node` LIMIT 100
| Fetch result | 8.348635 | 9.342241 | 0.993666  | dn2.0       | select COUNT(*) as `$COUNTS_rpda_0` from `sharding_two_node` LIMIT 100
| MERGE     | 8.721543  | 10.289905 | 1.568362  | merge.1     | merge.1
| ORDERED_GROUP | 8.726919  | 10.424399 | 1.69739   | ordered_group.1 | merge.1
| LIMIT     | 9.020162  | 10.499574 | 1.479412  | limit.1     | ordered_group.1
| SHUFFLE_FIELD | 9.623584  | 10.501529 | 1.477945  | shuffle_field.1 | limit.1
| Write to Client | 9.072457  | 11.52055  | 2.448093  | -           | -
```

Over All	0.0	11.52055	11.52055	-	-
----------	-----	----------	----------	---	---

+-----+  
| COUNT(\*) |  
+-----+  
| 1 |  
+-----+

## 7. trace

```
mysql> select count(*) from sharding_two_node where id =(select id from sharding_two_node where id=1);
+-----+
| COUNT(*) |
+-----+
| 1 |
+-----+
1 row in set (0.03 sec)

mysql> show trace;
+-----+-----+-----+-----+-----+
| OPERATION | START(ms) | END(ms) | DURATION(ms) | SHARDING_NODE | SQL/REF
+-----+-----+-----+-----+-----+
| Read SQL | 0.0 | 0.063047 | 0.063047 | - | -
| Parse SQL | 0.063047 | 0.491182 | 0.428135 | - | -
| Try Route Calculation | 0.491182 | 0.799576 | 0.308394 | - | -
| Try to Optimize | 0.799576 | 2.347412 | 1.547836 | - | -
| Execute SQL | 2.347412 | 11.183608 | 8.836396 | dn1.0 | select `sharding_two_node`.`id` as `autoalias_scalar` from `sharding_two_node` where id = 1 LIMIT 2
| Fetch result | 11.183608 | 12.360691 | 1.176883 | dn1.0 | select `sharding_two_node`.`id` as `autoalias_scalar` from `sharding_two_node` where id = 1 LIMIT 2
| MERGE | 11.889546 | 12.436445 | 0.546899 | merge.1 | dn1.0
| LIMIT | 11.894923 | 12.483364 | 0.588441 | limit.1 | merge.1
| SHUFFLE_FIELD | 11.896389 | 12.48483 | 0.588441 | shuffle_field.1 | limit.1
| SCALAR_SUB_QUERY | 12.038123 | 12.485808 | 0.447685 | scalar_sub_query.1 | shuffle_field.1
| Generate New Query | 12.485808 | 13.824463 | 1.338655 | - | -
| Execute SQL | 13.824463 | 26.749647 | 12.925184 | dn1.1 | scalar_sub_query.1; select COUNT(*) as `$_COUNT$_rpda_0` from `sharding_two_node` where sharding_two_node.id = 1
| Fetch result | 26.685134 | 28.753476 | 2.068342 | dn1.1 | scalar_sub_query.1; select COUNT(*) as `$_COUNT$_rpda_0` from `sharding_two_node` where sharding_two_node.id = 1
| MERGE | 26.954918 | 29.091683 | 2.136765 | merge.2 | dn1.1
| ORDERED_GROUP | 26.977889 | 29.563316 | 2.585427 | ordered_group.1 | merge.2
| SHUFFLE_FIELD | 27.568285 | 29.567226 | 1.998941 | shuffle_field.2 | ordered_group.1
| Write to Client | 27.72517 | 30.014911 | 2.289741 | - | -
| Over All | 0.0 | 30.014911 | 30.014911 | - | -
+-----+-----+-----+-----+-----+
```

18 rows in set (0.01 sec)

## 2.22 KILL @@DDL\_LOCK

dble " There is other session is doing DDL " "xxx is doing DDL " reload

dble ZK DDL universe/dble/{cluster-id}/ddl/{schema.table} json status

1. status INIT table table metakey dble table table meta

2. DDL DDL DDL status SUCCESS FAILED status SUCCESS DDLDDLtable table meta

3. universe/dble/{cluster-id}/ddl/{schema.table}/{dble-id}:SUCCESS

ddl

- universe/dble/{cluster-id}/ddl/{schema.table}/{dble-id}

- universe/dble/{cluster-id}/online/

DDLuniverse/dble/{cluster-id}/ddl/{schema.table}/ online universe/dble/{cluster-id}/ddl/{schema.table}/ DDL ddl

table meta      universe/dble/{cluster-id}/ddl/{schema.table} kv

### kill ddl\_lock

ddlldl universe/dble/{cluster-id}/ddl/{schema.table} kv

1. kill

2.

## 2.23

- [2.23.1 MYSQL-HA](#)
- [2.23.2](#)
- [2.23.3](#)
- [2.23.4 HA](#)

### 2.23.1 MySQL-HA

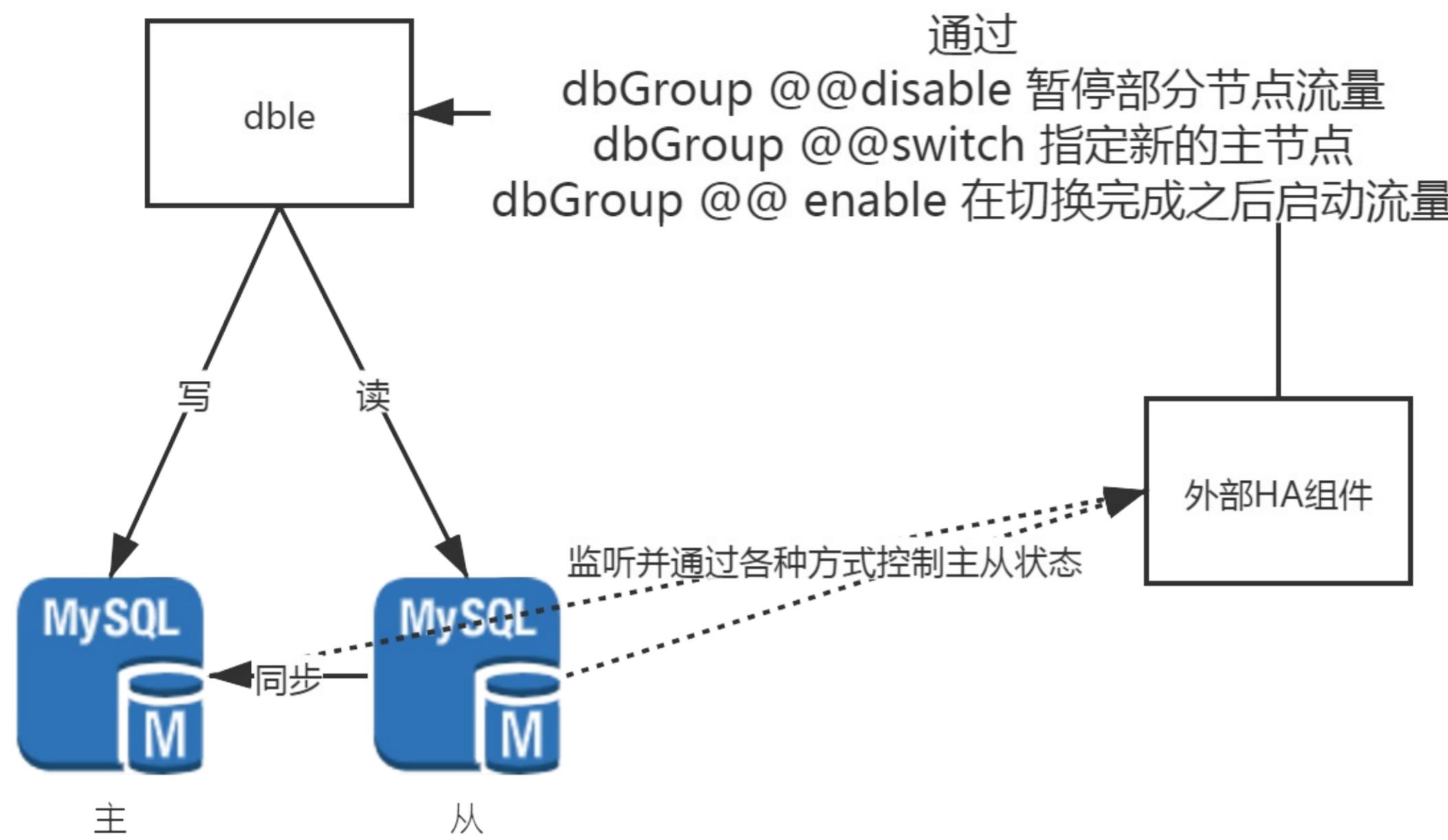
dble2.19.09.0

- HAdble
- HAMYSQL

- (dbGroup @@disable)
- (dbGroup @@enable)
- (dbGroup @@switch)

:

- (dbGroup @@events)

HA mysqlmysqlDBmysql  
zkzkdbleHA

HAdble

:

HA

## 2.23.2 dbGroup

### ha

dbleha

### bootstrap.cnf

```
-DuseOuterHa=true
```

:dble

### cluster.cnf

needSyncHa = true

- dbledbGroup
- bootstrap.cnf useOuterHatrue
- bootstrap.cnf useOuterHacluster.cnfneedSyncHatruebledbledble

dble

### dbleMySQL

dbInstance“disabled/enable”dbleMySQLMySQL

### dbGroup @@disable

dbGroup @@disable name = 'dbGroup\_name' [instance = 'instance\_name']

- dbGroup\_namedb.xmlldbGroupinstance\_namedbInstancename
- instance = '.'.dbGroupdbInstancedisable
- disabledmysql
- dbInstancedisableshow @@dbInstance
- dbledisable

### dbGroup @@enable

dbGroup @@enable name = 'dbGroup\_name' [instance = 'instance\_name']

- dbGroup\_namedb.xmlldbGroupinstance\_namedbInstancename
- instance = '.'.dbGroupdbInstanceenable

### dbGroup @@switch

dbGroup @@switch name = 'dbGroup\_name' master = 'instance\_name'

- dbGroup\_namedb.xmlldbGroupinstance\_namedbInstancename
- namemaster
- dbGroupprimarydbInstance
- dbInstancedisabledbInstancemainprimarydbInstancedbInstance

### 2.23.3

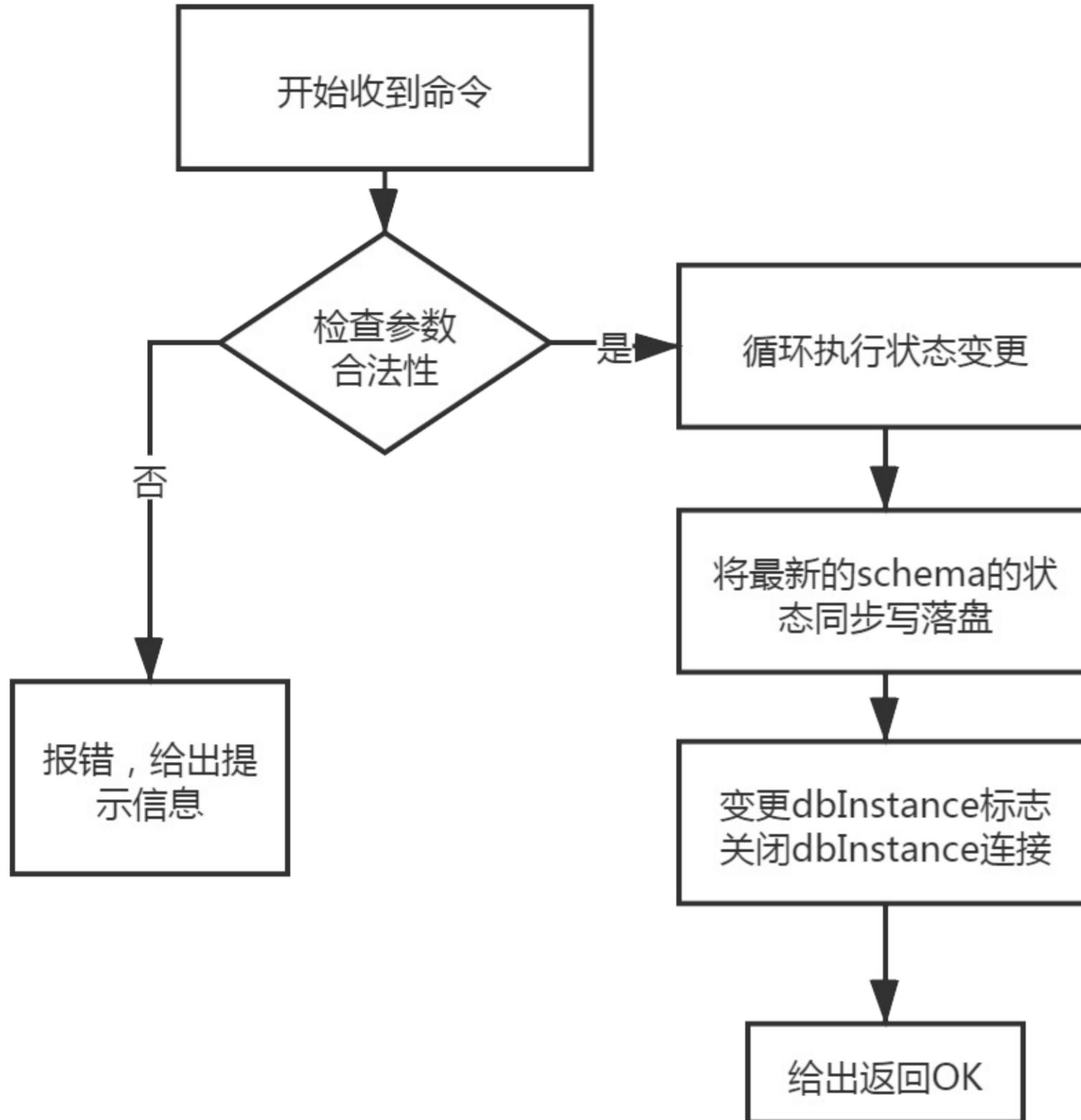
dbledbGroup

- disablezk
- enable
- switchprimary dbInstance

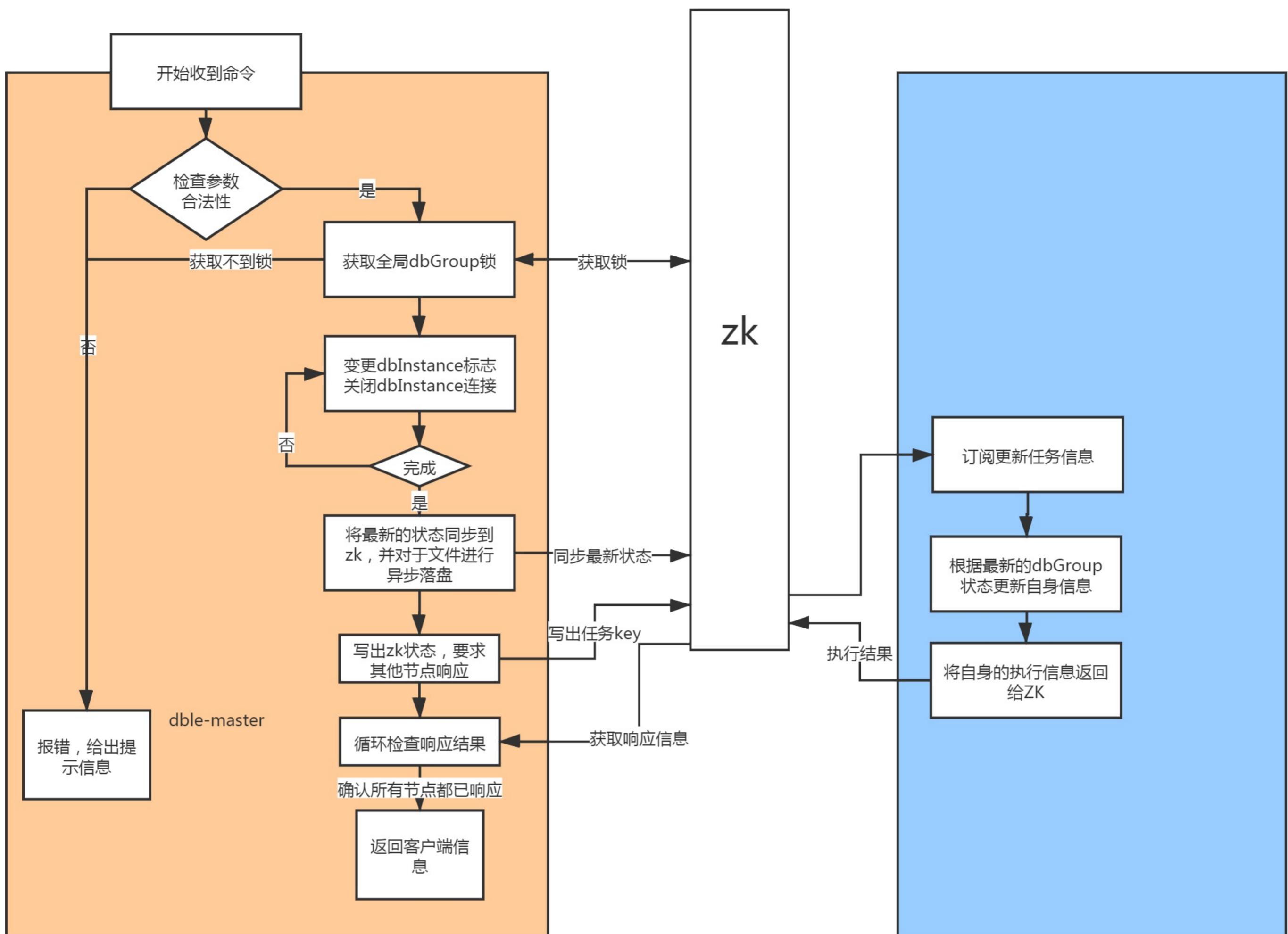
#### dbGroup @@disable

dbe

- 
- dbInstance
- 
- dbGroup
- OK

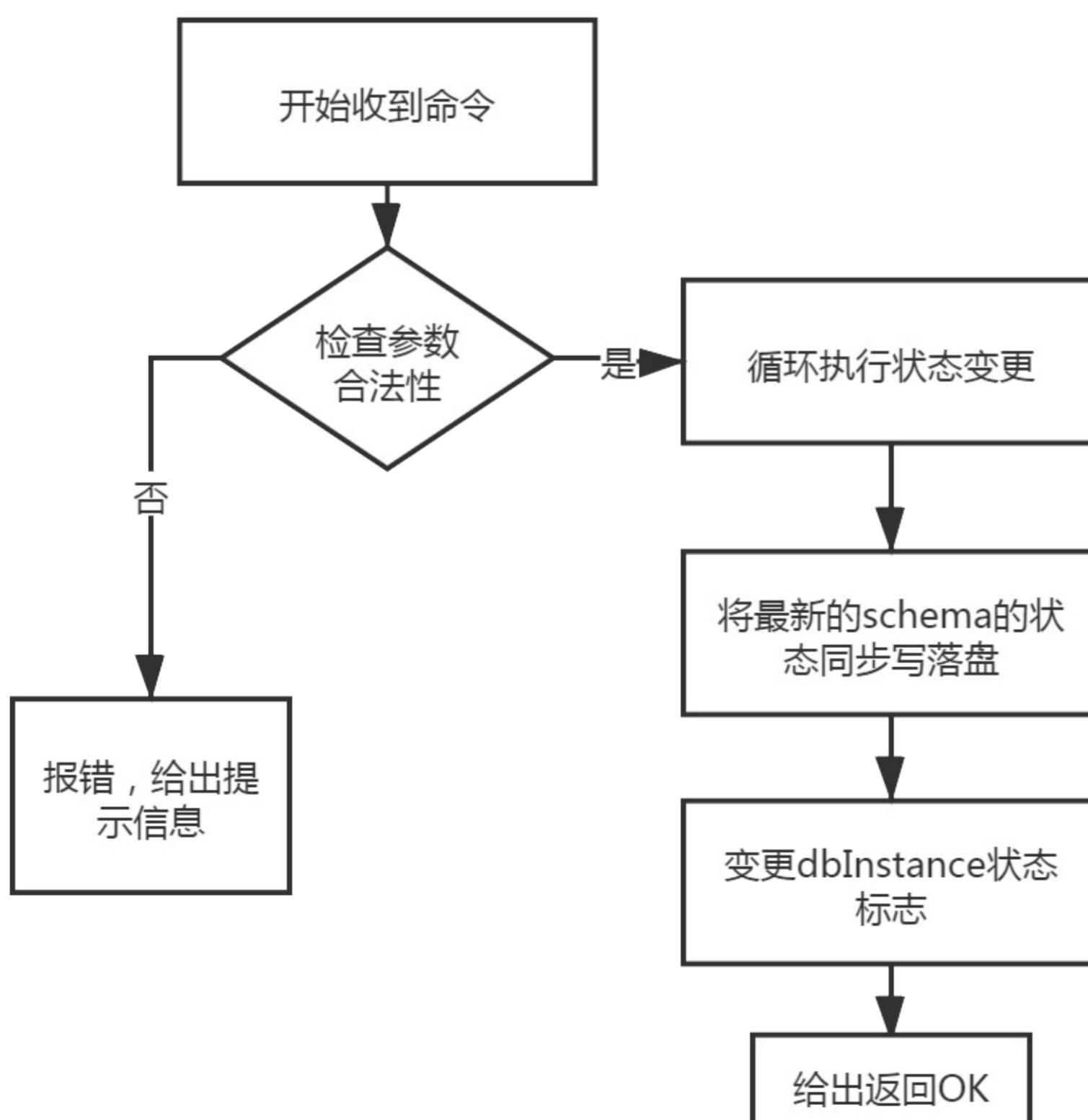


- 
- dbInstance
- 
- dbGroup
- dbGroupzk
- zkkey
- 
- dble
-

**dbGroup @@enable**

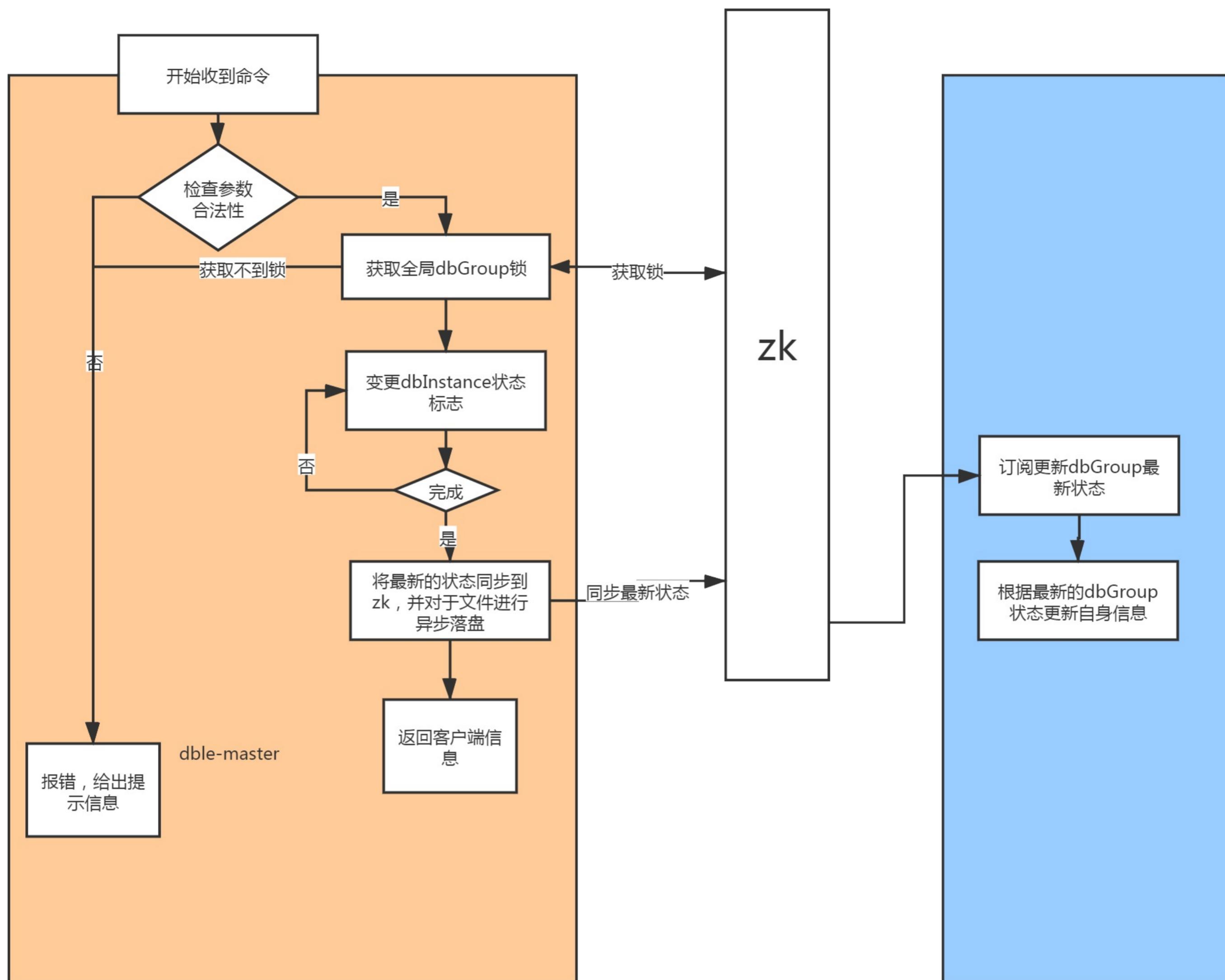
dble

- 
- dbInstance
- dbGroup
- OK



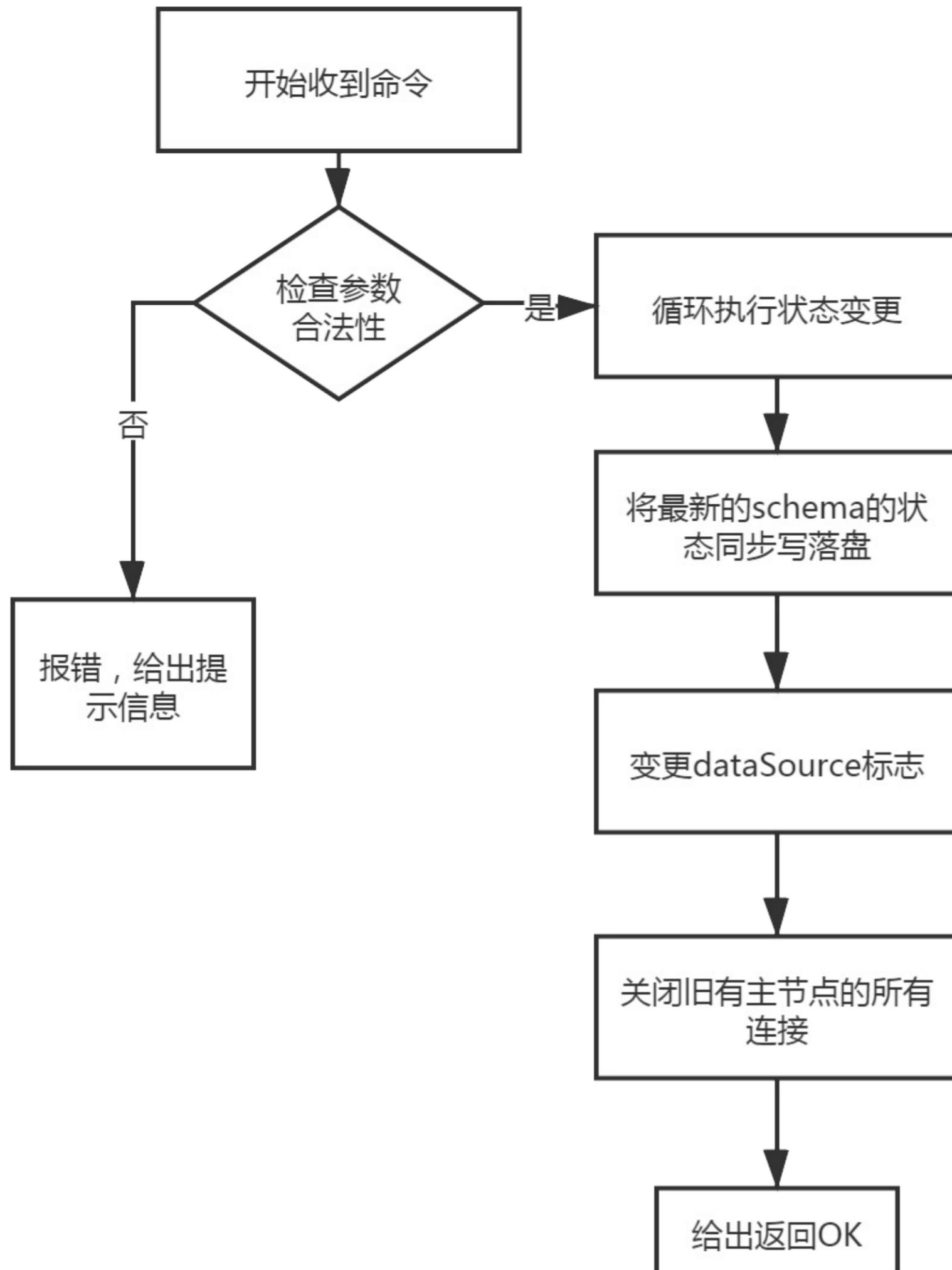
## ZK

- ddbInstance
- dbGroup
- dbGroupzk
- zkkey
- 
- dble
- 

**dbGroup @@switch**

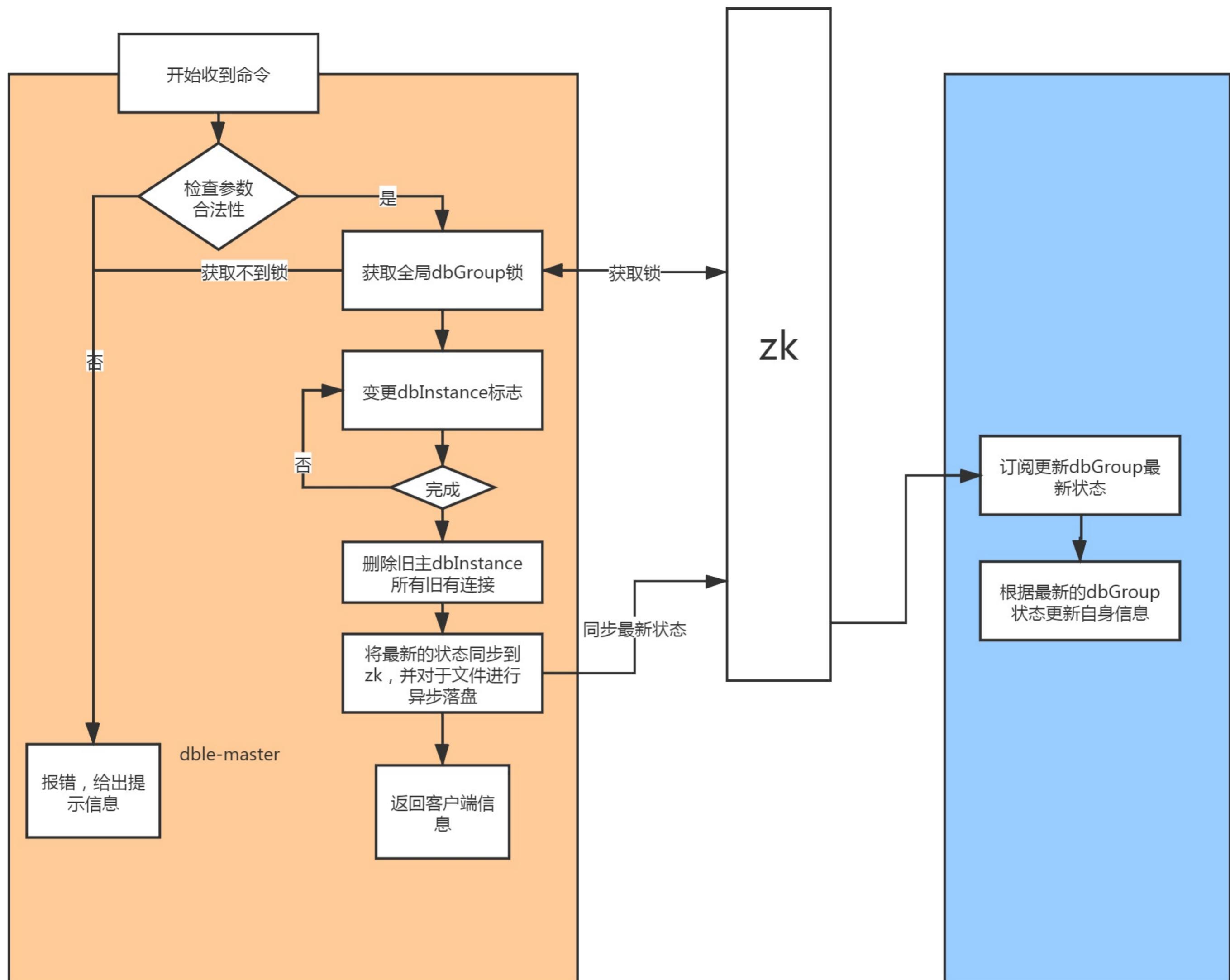
## dble

- dbInstance
- primary dbInstance
- dbGroup
- OK



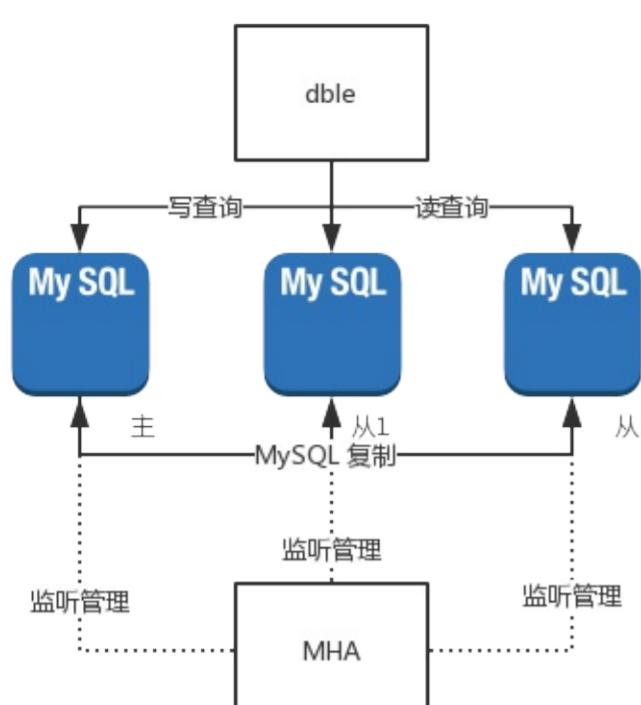
ZK

- 
- dbInstance
- 
- dbGroup
- dbGroupzk
- zkkey
- 
- dble
-

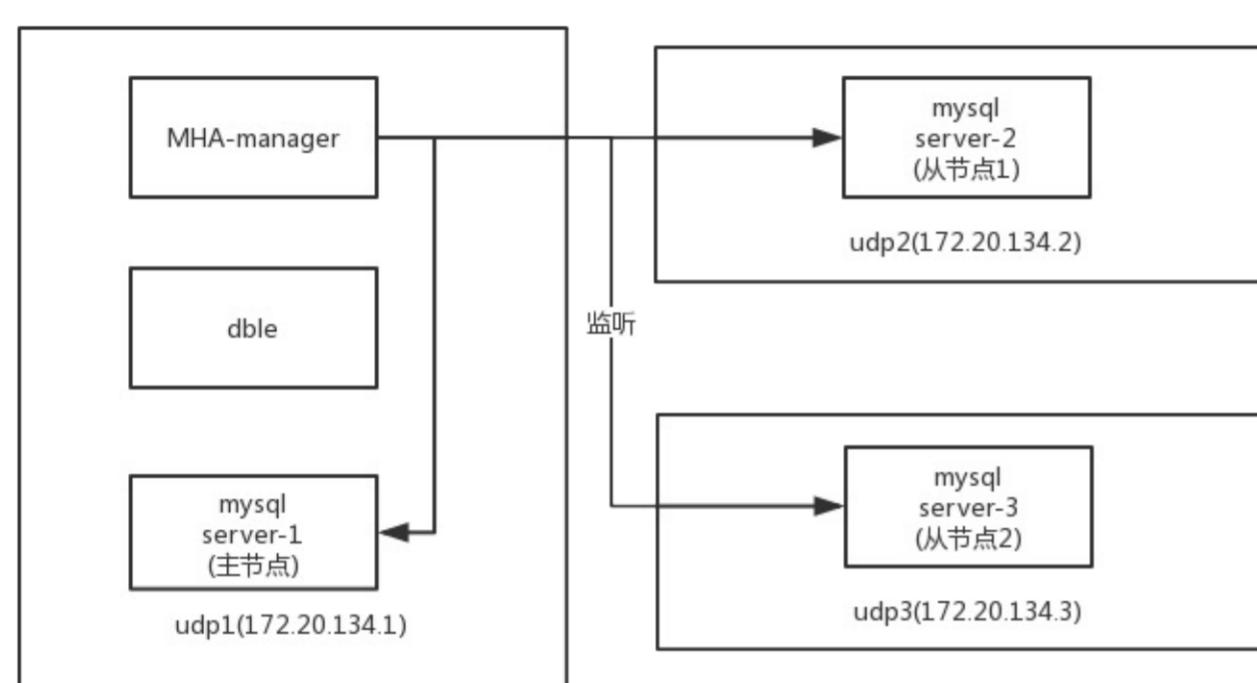


## 2.23.4 mha-dble

dbleMySQLdbGroupMHAMySQL



docker



- MySQL0
- MHAMySQL
- dbleMySQL
- kill MySQLMHA
- dbleMySQL

- dble2.19.09.0
- MHAdble

dockermysqlmysql()

## mha

```

● dockersh
/usr/sbin/sshd -D

● ,
ssh-keygen -t rsa
ssh-copy-id -i ~/.ssh/id_rsa.pub root@other1
ssh-copy-id -i ~/.ssh/id_rsa.pub root@other2
ssh-copy-id -i ~/.ssh/id_rsa.pub root@self

● /usr/bin/mysqlbinlog/usr/bin/mysqlyummysq
● MHA
mkdir /etc/masterha/app1 -p
mkdir /var/log/masterha/app2 -p
mkdir /var/log/masterha/app1 -p

● mysql
grant all on *.* to root@'%' identified by '123456' with grant option;
grant replication slave on *.* to repl@'%' identified by 'repl';

● MHA rpm
https://github.com/yoshinorim/mha4mysql-manager/wiki/Downloads
MHA Manager 0.56 rpm RHEL6
MHA Node 0.56 rpm RHEL6
rpm
yum localinstall rpmyum
NodeManager
rpm
yumrpmyum

```

mhapperlhamhamaster\_ip\_failover\_script  
[https://github.com/yoshinorim/mha4mysql-manager/wiki/Parameters#master\\_ip\\_failover\\_script](https://github.com/yoshinorim/mha4mysql-manager/wiki/Parameters#master_ip_failover_script)

- HAmaster\_ip\_failoverstatus
- MySQL mastermaster\_ip\_failoverstopmaster
- masterread\_only=0master\_ip\_failoverstart

mhadble

- MySQL masterdbGroupdisablestopdbledbGroup @@disable
- MySQL masterstartdbGroup @@@switchmasterdbGroup

master\_ip\_failover()

```
#!/usr/bin/env perl

# Copyright (C) 2011 DeNA Co.,Ltd.
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc.,
# 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

## Note: This is a sample script and is not complete. Modify the script based on your environment.

use strict;
use warnings FATAL => 'all';

use Getopt::Long;
use MHA::DBHelper;

my (
    $command,           $ssh_user,           $orig_master_host,
    $orig_master_ip,   $orig_master_port,   $new_master_host,
    $new_master_ip,    $new_master_port,    $new_master_user,
    $new_master_password
);
GetOptions(
    'command=s'         => \$command,
    'ssh_user=s'        => \$ssh_user,
    'orig_master_host=s'=> \$orig_master_host,
    'orig_master_ip=s'  => \$orig_master_ip,
    'orig_master_port=i'=> \$orig_master_port,
    'new_master_host=s'  => \$new_master_host,
    'new_master_ip=s'   => \$new_master_ip,
    'new_master_port=i'=> \$new_master_port,
    'new_master_user=s' => \$new_master_user,
    'new_master_password=s'=> \$new_master_password,
);
exit &main();

sub main {
    if ( $command eq "stop" || $command eq "stopssh" ) {

        # $orig_master_host, $orig_master_ip, $orig_master_port are passed.
        # If you manage master ip address at global catalog database,
        # invalidate orig_master_ip here.
        my $exit_code = 1;
        eval {

            # dbGroup @@disable name = "dbGroup1" instance="$orig_master_host"
            # disable
            $orig_master_host =~ tr/././;
            system "mysql -P9066 -u man1 -p654321 -h 172.20.134.1 -e \"dbGroup @@@disable name = 'dbGroup1' instance='".$orig_master_host."\"";
            $exit_code = 0;
        };
        if ($@) {
            warn "Got Error: $@\n";
            exit $exit_code;
        }
        exit $exit_code;
    }
    elsif ( $command eq "start" ) {

        # all arguments are passed.
        # If you manage master ip address at global catalog database,
        # activate new_master_ip here.
        # You can also grant write access (create user, set read_only=0, etc) here.
        my $exit_code = 10;
        eval {
            my $new_master_handler = new MHA::DBHelper();

            # args: hostname, port, user, password, raise_error_or_not
            $new_master_handler->connect( $new_master_ip, $new_master_port,
                                         $new_master_user, $new_master_password, 1 );

            ## Set read_only=0 on the new master
            $new_master_handler->disable_log_bin_local();
            print "Set read_only=0 on the new master.\n";
            $new_master_handler->disable_read_only();

            ## Creating an app user on the new master
            print "Creating app user on the new master..\n";
            $new_master_handler->enable_log_bin_local();
            $new_master_handler->disconnect();

            ## try to switch the dbGroup master into new master
            ## dbGroup switchnew_master_host
            $new_master_host =~ tr/././;
            system "mysql -P9066 -u man1 -p654321 -h 172.20.134.1 -e \"dbGroup @@@switch name = 'dbGroup1' master='".$new_master_host."\"";

            $exit_code = 0;
        };
        if ($@) {
            warn $@;
            # If you want to continue failover, exit 10.
            exit $exit_code;
        }
        exit $exit_code;
    }
    elsif ( $command eq "status" ) {
        # test for start command
        exit 0;
    }
    else {
        &usage();
        exit 1;
    }
}

sub usage {
    print
"Usage: master_ip_failover --command=start|stop|stopssh|status --orig_master_host=host --orig_master_ip=ip --orig_master_port=port --new_master_host=host --new_master_ip=ip --new_master_port=port\n";
}

sub master_ip_failover {
    my ($command, $orig_master_host, $orig_master_ip, $orig_master_port, $new_master_host, $new_master_ip, $new_master_port) = @_;
    my $exit_code = 0;
    eval {
        # disable
        $orig_master_host =~ tr/././;
        system "mysql -P9066 -u man1 -p654321 -h 172.20.134.1 -e \"dbGroup @@@disable name = 'dbGroup1' instance='".$orig_master_host."\"";
        $exit_code = 0;
    };
    if ($@) {
        warn "Got Error: $@\n";
        exit $exit_code;
    }
    eval {
        # enable
        $new_master_host =~ tr/././;
        system "mysql -P9066 -u man1 -p654321 -h 172.20.134.1 -e \"dbGroup @@@enable name = 'dbGroup1' instance='".$new_master_host."\"";
        $exit_code = 0;
    };
    if ($@) {
        warn "Got Error: $@\n";
        exit $exit_code;
    }
}
```

/etc/masterha/app1  
MHA app1.conf/etc/masterha/app1

```
#mha manager
manager_workdir = /var/log/masterha/app1
manager_log = /var/log/masterha/app1/app1.log
remote_workdir = /var/log/masterha/app2
master_ip_failover_script=/etc/masterha/app1/master_ip_failover
# master_ip_online_change_script=/etc/masterha/app1/master_ip_online_change
```

```
# MySQL
user=root
password=123456
# ssh
ssh_user=root

#
repl_user=repl
repl_password=repl

# ()
ping_interval=1
manager_log=/var/log/masterha/app1/manager.log

[server1]
hostname=172.20.134.1
master_binlog_dir = /opt/3306/
port=3306

[server2]
# mysql
hostname=172.20.134.2
master_binlog_dir = /opt/3306/
candidate_master=1
check_repl_delay=0
port=3306

[server3]
# mysql
hostname=172.20.134.3
master_binlog_dir = /opt/3306/
candidate_master=1
check_repl_delay=0
port=3306
```

## MHA

```
nohup masterha_manager --conf=/etc/masterha/app1/app1.conf >> /var/log/masterha/app1/manager.log 2>&1 &
```

## dble

```
dblerelease2.19.0.0conf
```

```
mv cluster_template.cnf cluster.cnf
mv bootstrap_template.cnf bootstrap.cnf
mv db_template.xml db.xml
mv user_template.xml user.xml
mv sharding_template.xml sharding.xml
```

## db.xml

```
db.xml
```

```
<dbGroup name="dbGroup1" delayThreshold="100">
<heartbeat>show slave status</heartbeat>
<dbInstance name="172_20_134_1" url="172.20.134.1:3306" password="123456" user="root" disabled="false" id="udp-1" primary="true" />
<dbInstance name="172_20_134_2" url="172.20.134.2:3306" password="123456" user="root" disabled="false" id="udp-3" />
<dbInstance name="172_20_134_3" url="172.20.134.3:3306" password="123456" user="root" disabled="false" id="udp-2" />
</dbGroup>
```

## dbleman1dble

```
MySQL [(none)]> show @@dbinstance;
+-----+-----+-----+-----+-----+-----+-----+
| DB_GROUP | NAME   | HOST    | PORT | W/R | ACTIVE | IDLE | SIZE | EXECUTE | READ_LOAD | WRITE_LOAD | DISABLED |
+-----+-----+-----+-----+-----+-----+-----+
| dbGroup1 | 172_20_134_1 | 172.20.134.1 | 3306 | W   | 1 | 0 | 1000 | 1 | 0 | 0 | false |
| dbGroup1 | 172_20_134_3 | 172.20.134.3 | 3306 | R   | 1 | 0 | 1000 | 0 | 0 | 0 | false |
| dbGroup1 | 172_20_134_2 | 172.20.134.2 | 3306 | R   | 1 | 0 | 1000 | 0 | 0 | 0 | false |
+-----+-----+-----+-----+-----+-----+-----+
```

172.20.134.1"

- ps172.20.134.1mysqld
- kill -9 172.20.134.1mysqld
- dble

```
MySQL [(none)]> show @@dbinstance;
+-----+-----+-----+-----+-----+-----+-----+
| DB_GROUP | NAME   | HOST    | PORT | W/R | ACTIVE | IDLE | SIZE | EXECUTE | READ_LOAD | WRITE_LOAD | DISABLED |
+-----+-----+-----+-----+-----+-----+-----+
| dbGroup1 | 172_20_134_2 | 172.20.134.2 | 3306 | W   | 1 | 0 | 1000 | 0 | 0 | 0 | false |
| dbGroup1 | 172_20_134_3 | 172.20.134.3 | 3306 | R   | 1 | 0 | 1000 | 0 | 0 | 0 | false |
| dbGroup1 | 172_20_134_1 | 172.20.134.1 | 3306 | R   | 0 | 0 | 1000 | 0 | 0 | 0 | true |
+-----+-----+-----+-----+-----+-----+-----+
```

- dble,172\_20\_134\_2

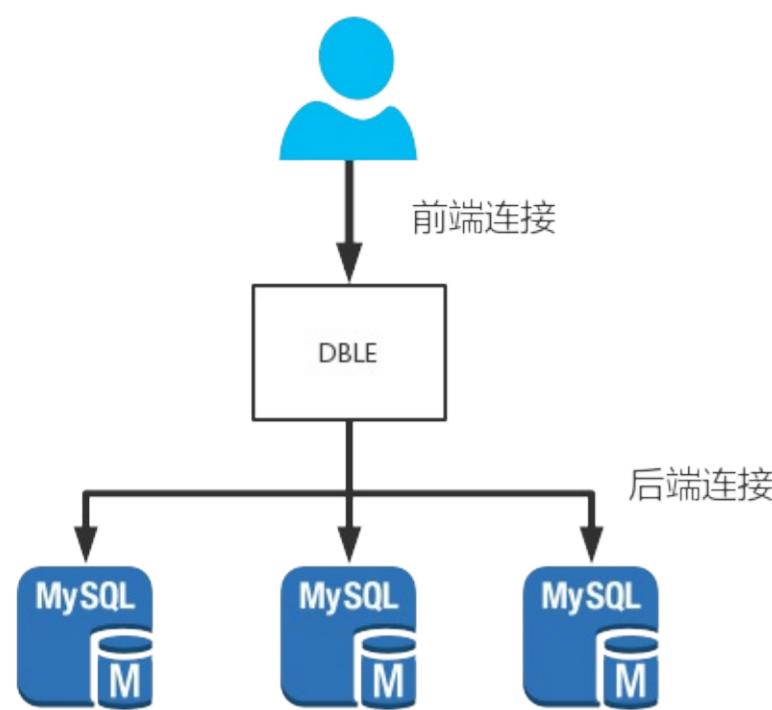
```
<dbGroup name="dbGroup1" delayThreshold="100">
<heartbeat>show slave status</heartbeat>
<dbInstance name="172_20_134_1" url="172.20.134.1:3306" password="123456" user="root" disabled="false" id="udp-1" />
<dbInstance name="172_20_134_2" url="172.20.134.2:3306" password="123456" user="root" disabled="false" id="udp-3" />
<dbInstance name="172_20_134_3" url="172.20.134.3:3306" password="123456" user="root" disabled="false" id="udp-2" primary="true" />
</dbGroup>
```

- 172.20.134.3mysql172.20.134.2dble

```
MySQL [(none)]> show slave status\G
***** 1. row *****
Slave_IO_State: Waiting for master to send event
  Master_Host: 172.20.134.2
  Master_User: repl
  Master_Port: 3306
  Connect_Retry: 60
  Master_Log_File: mysql-bin.000001
  Read_Master_Log_Pos: 2113
  Relay_Log_File: udp3-relay-bin.000002
  Relay_Log_Pos: 320
  Relay_Master_Log_File: mysql-bin.000001
  Slave_IO_Running: Yes
  Slave_SQL_Running: Yes
.....
```

**2.24 /**

dbledblemyql



TCPdbledble dble

bootstrap.cnf

- sqlExecuteTimeout()
- idleTimeout
- processorCheckPeriod
  
- dbleprocessorCheckPeriod
- - sqlExecuteTimeout
  - DDLxa
- - xacommit/rollback
  - idleTimeout
  
- DDL,XAsqlExecuteTimeout
- XAidleTimeoutloaddata

**SQL**

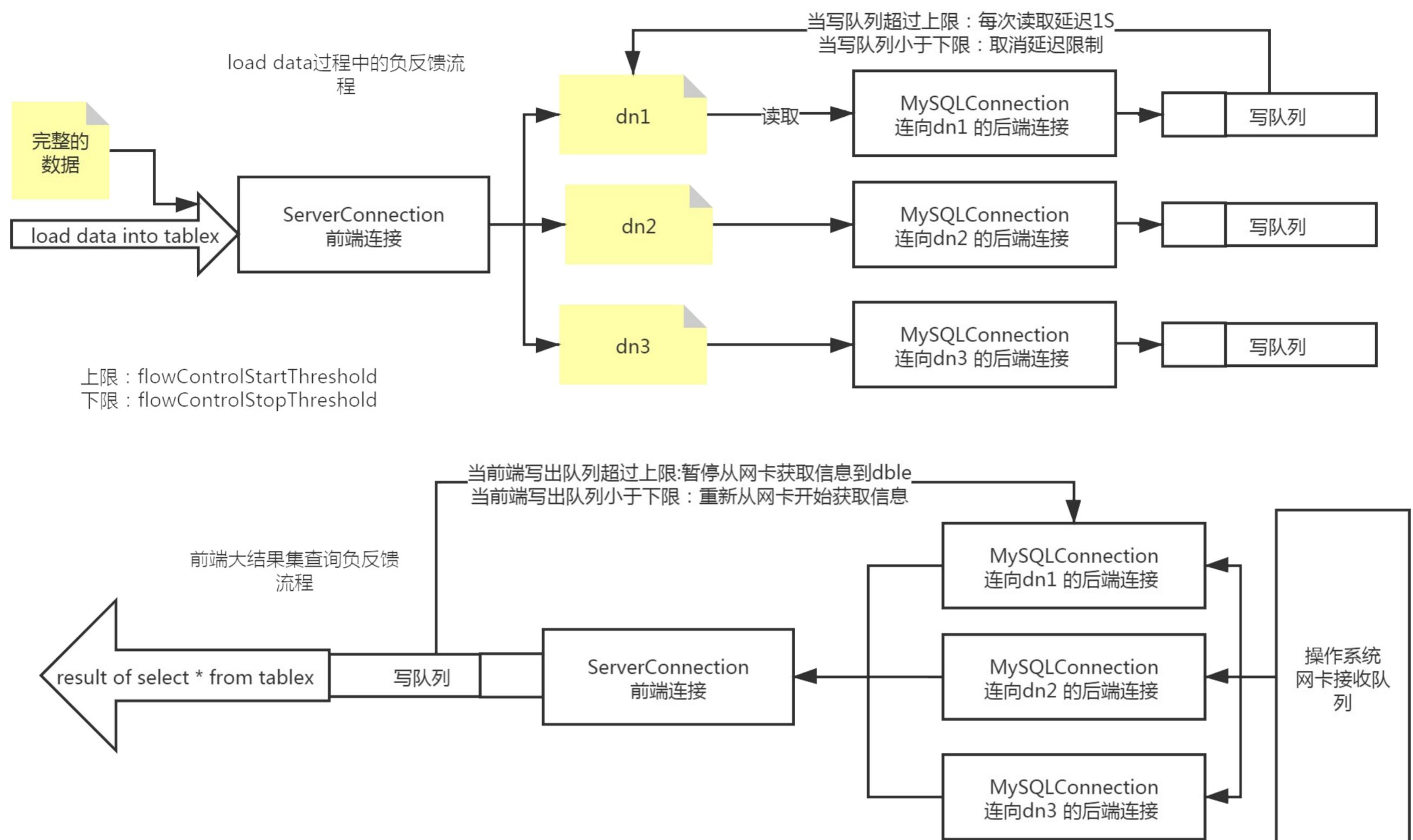
- 
- SQL
- load dataidleTimeoutidleTimeout
- processorCheckPeriodSQL

## 2.25 dble

dbleloaddbdbleOOM  
2.20.04@ssxlulu/load issue

dble

- load data
- Query



bootstrap.cnf

bootstrap.cnf

```
enableFlowControl()  
flowControlStartThreshold()  
flowControlStopThreshold()
```

- flow\_control @@show
- flow\_control @@set [enableFlowControl = true/false] [flowControlStart = ?] [flowControlEnd = ?]
- flow\_control @@list

## 2.26 /client\_found\_rows

### 2.26.1 &

DBLEdbclient\_found\_rows

#### 2.26.1.1 client\_found\_rows

handshakeclient\_found\_rowsDML(found rows)(affect rows)

#### 2.26.1.2 client\_found\_rows

MySQLclient\_found\_rowaffect rows  
JDBCclient\_found\_rowsfound rows

#### 2.26.1.3 JDBCuseAffectedRowsclient\_found\_row

useAffectedRows=true client\_found\_rows  
useAffectedRows=false() client\_found\_rows

### 2.26.2

#### 2.26.2.1 bootstrap.cnf

```
#client_found_rows
-DcapClientFoundRows=false
```

#### 2.26.2.2

```
show @@cap_client_found_rows; -- client_found_row 0- 1-
disable @@cap_client_found_rows; -- client_found_row
enable @@cap_client_found_rows; -- client_found_row
```

dblemysql() insert

### 2.26.3

step1. dble(client\_found\_rows)

step2. client\_found\_rows

```
mysql -uman1 -h192.xx.xx.xx -P9066 -p654321
enable @@cap_client_found_rows;
```

step3. dbleclient\_found\_rows

```
mysql -uroot -h192.xx.xx.xx -P8066 -p123456
ERROR 1045 (HY000): The client requested CLIENT_FOUND_ROWS capabilities does not match, in the manager use show @@cap_client_found_rows check latest status.
```

### 3.

- 3.1 DDL
  - 3.1.1 DDL&Table Syntax
  - 3.1.2 DDL&View Syntax
  - 3.1.3 DDL&Index Syntax
  - 3.1.4 DDL
  - 3.1.5 DDL&Database Syntax
  - 3.1.6 ONLINE DDL
- 3.2 DML
  - 3.2.1 INSERT
  - 3.2.2 REPLACE
  - 3.2.3 DELETE
  - 3.2.4 UPDATE
  - 3.2.5 SELECT
  - 3.2.6 SELECT JOIN syntax
  - 3.2.7 SELECT UNION Syntax
  - 3.2.8 SELECT Subquery Syntax
  - 3.2.9 LOAD DATA
  - 3.2.10
- 3.3 Prepared SQL Syntax
- 3.4 Transactional and Locking Statements
  - 3.4.1 Lock&unlock
  - 3.4.2 XA
  - 3.4.3
  - 3.4.4 SET TRANSACTION Syntax
- 3.5 DAL
  - 3.5.1 SET
  - 3.5.2 SHOW
  - 3.5.3 KILL
- 3.6
- 3.7 Utility Statements
- 3.8 Hint
- 3.9
- 3.10 (alpha)
- 3.11

### 3.1 DDL

DDL

DDLDM LDDL

- 3.1.1 DDL&Table Syntax
- 3.1.2 DDL&View Syntax
- 3.1.3 DDL&Index Syntax
- 3.1.4 DDL

### 3.1.1 TABLE DDL

#### 3.1.1.1 CREATE TABLE Syntax

```

CREATE TABLE [IF NOT EXISTS] tbl_name
  (create_definition,...)
  [table_options]
  [partition_options]

create_definition:
  col_name column_definition

column_definition:
  data_type [NOT NULL | NULL] [DEFAULT default_value]
  [AUTO_INCREMENT] [UNIQUE [KEY] | [PRIMARY] KEY]
  [COMMENT 'string']

data_type:
  BIT[(length)]
  | TINYINT[(length)] [UNSIGNED] [ZEROFILL]
  | SMALLINT[(length)] [UNSIGNED] [ZEROFILL]
  | MEDIUMINT[(length)] [UNSIGNED] [ZEROFILL]
  | INT[(length)] [UNSIGNED] [ZEROFILL]
  | INTEGER[(length)] [UNSIGNED] [ZEROFILL]
  | BIGINT[(length)] [UNSIGNED] [ZEROFILL]
  | REAL[(length,decimals)] [UNSIGNED] [ZEROFILL]
  | DOUBLE[(length,decimals)] [UNSIGNED] [ZEROFILL]
  | FLOAT[(length,decimals)] [UNSIGNED] [ZEROFILL]
  | DECIMAL[(length[,decimals])] [UNSIGNED] [ZEROFILL]
  | NUMERIC[(length[,decimals])] [UNSIGNED] [ZEROFILL]
  | DATE
  | TIME[(fsp)]
  | TIMESTAMP[(fsp)]
  | DATETIME[(fsp)]
  | YEAR
  | CHAR[(length)]
  | VARCHAR(length)
  | BINARY[(length)]
  | VARBINARY(length)
  | TINYBLOB
  | BLOB
  | MEDIUMBLOB
  | LONGBLOB
  | TINYTEXT
  | TEXT
  | MEDIUMTEXT
  | LONGTEXT
  | ENUM(value1,value2,value3,...)

table_options:
  table_option [[,] table_option] ...

table_option:
  ENGINE [=] engine_name
  | [DEFAULT] CHARACTER SET [=] charset_name
  | CHECKSUM [=] {0 | 1}
  | [DEFAULT] COLLATE [=] collation_name
  | COMMENT [=] 'string'
  | CONNECTION [=] 'connect_string'
  | KEY_BLOCK_SIZE [=] value
  | MAX_ROWS [=] value
  | MIN_ROWS [=] value
  | PASSWORD [=] 'string'
  | ROW_FORMAT [=] {DEFAULT|DYNAMIC|FIXED|COMPRESSED|REDUNDANT|COMPACT}
  | STATS_AUTO_RECALC [=] {DEFAULT|0|1}
  | STATS_PERSISTENT [=] {DEFAULT|0|1}
  partition_options:
    {[LINEAR] HASH(expr)
     | PARTITION BY [linear] KEY (column_list)
     | RANGE(expr) | COLUMNS(column_list)
     | LIST(expr) | COLUMNS(column_list)
    }
  [(partition_definition [, partition_definition] ...)]

```

- engine\_name“InnoDB”

- CREATE TABLE `test` (`id` enum('1','2','3') DEFAULT '1')id'l'dbleissue

<https://github.com/actiontech/dble/issues/816>

```

create table if not exists test(
  id bigint primary key AUTO_INCREMENT,
  col1 int not null default 5,
  col2 int null COMMENT 'info for col1',
  col3 varchar(20) not null,
  col4 varchar(20) unique key
);

create table test(
  id int primary key,
  col_bit      BIT(1),
  col_tinyint  TINYINT(2) UNSIGNED ZEROFILL,
  col_smallint  SMALLINT(3) UNSIGNED ZEROFILL,
  col_mediumint MEDIUMINT(4) UNSIGNED ZEROFILL,
  col_int      INT(5) UNSIGNED ZEROFILL,
  col_integer  INTEGER(6) UNSIGNED ZEROFILL,
  col_bigint   BIGINT(7) UNSIGNED ZEROFILL,
  col_real     REAL(8,1) UNSIGNED ZEROFILL,
  col_double   DOUBLE(9,2) UNSIGNED ZEROFILL,
  col_float    FLOAT(10,3) UNSIGNED ZEROFILL,
  col_decimal  DECIMAL(11,4) UNSIGNED ZEROFILL,
  col_numeric   NUMERIC(12,5) UNSIGNED ZEROFILL,
  col_date     DATE,
  col_time     TIME(3),
  col_timestamp TIMESTAMP(4),
  col_datetime DATETIME(5),
  col_year     YEAR,
  col_char     CHAR(10),
  col_varchar   VARCHAR(20),
  col_binary   BINARY(30),
  col_varbinary VARBINARY(40),
  col_tinyblob TINYBLOB,
  col_blob      BLOB,
  col_mediumblob MEDIUMBLOB,
  col_longblob  LONGBLOB,
  col_tinytext   TINYTEXT,
  col_text      TEXT,
  col_mediumtext MEDIUMTEXT,
  col_longtext   LONGTEXT,
  col_enum      ENUM('a','b','c')
);

```

```

create table test(
  id int primary key,
  col1 varchar(20)
)ENGINE = innodb
AVG_ROW_LENGTH = 20
DEFAULT CHARACTER SET = utf8
CHECKSUM = 1
DEFAULT COLLATE = utf8_general_ci
COMMENT = 'info of table test'
CONNECTION = '111111'
DELAY_KEY_WRITE = 1
INSERT_METHOD = LAST
KEY_BLOCK_SIZE = 65536
MAX_ROWS = 3
MIN_ROWS = 2
PACK_KEYS = 1
ROW_FORMAT = DEFAULT;

```

#### 3.1.1.2 ALTER TABLE Syntax

```
ALTER [IGNORE] TABLE tbl_name
```

```
[alter_specification [, alter_specification] ...]

alter_specification:
| ADD [COLUMN] col_name column_definition
| [FIRST | AFTER col_name]
| ADD [COLUMN] (col_name column_definition,...)
| ADD {INDEX | KEY} [index_name]
| CHANGE [COLUMN] old_col_name new_col_name column_definition
| [FIRST|AFTER col_name]
| MODIFY [COLUMN] col_name column_definition
| [FIRST | AFTER col_name]
| DROP [COLUMN] col_name
| DROP {INDEX | KEY} index_name
| ADD {INDEX|KEY} [index_name] (index_col_name,...)
| DROP {INDEX|KEY} index_name
| ADD PRIMARY KEY (index_col_name,...)
| DROP PRIMARY KEY
| ALTER [COLUMN] col_name
{SET DEFAULT {literal | (expr)} | DROP DEFAULT}
| COMMENT [=] 'string'
```

```
alter table test add column col5 int not null default 1 first,add column col6 int after col4;
alter table test change column col1 col1_new int after col3;
alter table test modify column col1_new varchar(20) after id;
alter table test drop column col6;
alter table test add key idx_col4(col4);
alter table test add index idx_col4(col4);
alter table test drop key idx_col4;
alter table test drop index idx_col4;
alter table test drop primary key;
alter table test add primary key (id);
alter table test alter column col set default 0;
alter table test alter column col drop default;
alter table test comment = 'string';
```

### 3.1.1.3 DROP TABLE Syntax

```
DROP TABLE [IF EXISTS]
tbl_name [, tbl_name] ...
[RESTRICT | CASCADE]
```

```
drop table if exists test cascade;
drop table test restrict;
```

### 3.1.1.4 TRUNCATE TABLE Syntax

```
TRUNCATE [TABLE] tbl_name
```

```
truncate table test;
```

### 3.1.2 VIEW DDL

Syntax

**create view :**

```
CREATE [OR REPLACE] VIEW  
view_name [(column_list)]  
AS select_statement
```

**alter view :**

```
ALTER VIEW  
view_name [(column_list)]  
AS select_statement
```

**drop view:**

```
DROP VIEW [IF EXISTS] view_name [, view_name]
```

**show create view :**

```
SHOW CREATE VIEW view_name;
```

### 3.1.3 INDEX DDL

#### 3.1.3.1 CREATE INDEX Syntax

```
CREATE [UNIQUE|FULLTEXT] INDEX index_name  
[index_type]  
ON tbl_name (index_col_name,...)  
  
index_col_name:  
col_name [(length)] [ASC | DESC]  
  
index_type:  
USING {BTREE | HASH}
```

```
create unique index idx1 using btree on test(col1);  
create index idx2 using hash on test(col2);  
create fulltext index idx3 on test(col4);  
create fulltext index idx4 on test(col4(10));
```

#### 3.1.3.2 DROP INDEX Syntax

```
DROP INDEX index_name ON tbl_name
```

```
drop index idx1 on test;
```

**3.1.4 DDL**

DDLdbmysqlDDL(eg. ALTER EVENT)  
dbmysqlDDL

```
/*!dbe:sql=select ... from tbx where id=M*/ ddl statement
```

tbx idM

```
MySQL [TESTDB]> /*!dbe:sql=select * from a_test where id=2*/CREATE PROCEDURE account_count()  
BEGIN  SELECT 'Number of accounts:', COUNT(*) FROM mysql.user;  
END//
```

### 3.1.5 DATABASE DDL

#### 3.1.5.1 CREATE DATABASE Syntax

```
CREATE {DATABASE | SCHEMA} [IF NOT EXISTS] db_name  
[create_specification] ...  
  
create_specification:  
    [DEFAULT] CHARACTER SET [=] charset_name  
    | [DEFAULT] COLLATE [=] collation_name  
    | DEFAULT ENCRYPTION [=] {'Y' | 'N'}
```

- schemashema.xml
- create\_specification
-

### 3.1.6 ONLINE DDL

#### 3.1.6.1

1. 3.20.04.0dbledddbtleSQLdddbleshow create table dble
2. SQLdbledbldbledbnullsqlmysqlonlineDDL

#### 3.1.6.2 ONLINE DDL

MySQL 8.0online ddldbleonline ddl

- Index Operations
- Primary Key Operations
- Column Operations
- Generated Column Operations
- Foreign Key Operations
- Table Operations
- Tablespace Operations
- Partitioning Operations

1			CREATE INDEX <i>name</i> ON <i>table</i> ( <i>col_list</i> );ALTER TABLE <i>tbl_name</i> ADD INDEX <i>name</i> ( <i>col_list</i> );		
2			DROP INDEX <i>name</i> ON <i>table</i> ;ALTER TABLE <i>tbl_name</i> DROP INDEX <i>name</i> ;		
3			ALTER TABLE <i>tbl_name</i> RENAME INDEX <i>old_index_name</i> TO <i>new_index_name</i> , ALGORITHM=INPLACE, LOCK=NONE;	mysql 5.6 online ddl	
4	FULLTEXT		CREATE FULLTEXT INDEX <i>name</i> ON <i>table</i> ( <i>column</i> );	mysql	
5	SPATIAL		CREATE TABLE <i>geom</i> ( <i>g</i> GEOMETRY NOT NULL); ALTER TABLE <i>geom</i> ADD SPATIAL INDEX( <i>g</i> ), ALGORITHM=INPLACE, LOCK=SHARED;	mysql	
6			ALTER TABLE <i>tbl_name</i> DROP INDEX <i>i1</i> , ADD INDEX <i>i1</i> ( <i>key_part</i> ...) USING BTREE, ALGORITHM=INSTANT;		
7			ALTER TABLE <i>tbl_name</i> ADD PRIMARY KEY ( <i>column</i> ), ALGORITHM=INPLACE, LOCK=NONE;		
8			ALTER TABLE <i>tbl_name</i> DROP PRIMARY KEY, ALGORITHM=COPY;	COPYdml	
9			ALTER TABLE <i>tbl_name</i> DROP PRIMARY KEY, ADD PRIMARY KEY ( <i>column</i> ), ALGORITHM=INPLACE, LOCK=NONE;		
10			ALTER TABLE <i>tbl_name</i> ADD COLUMN <i>column_name</i> <i>column_definition</i> , ALGORITHM=INSTANT;	dble	
11			ALTER TABLE <i>tbl_name</i> DROP COLUMN <i>column_name</i> , ALGORITHM=INPLACE, LOCK=NONE;	dble	
12			ALTER TABLE <i>tbl</i> CHANGE <i>old_col_name</i> <i>new_col_name</i> <i>data_type</i> , ALGORITHM=INPLACE, LOCK=NONE;	dble	
13			ALTER TABLE <i>tbl_name</i> MODIFY COLUMN <i>col_name</i> <i>column_definition</i> FIRST, ALGORITHM=INPLACE, LOCK=NONE;		
14			ALTER TABLE <i>tbl_name</i> CHANGE <i>c1</i> <i>c1</i> BIGINT, ALGORITHM=COPY;	dml	
15	VARCHAR		ALTER TABLE <i>tbl_name</i> CHANGE COLUMN <i>c1</i> <i>c1</i> VARCHAR(255), ALGORITHM=INPLACE, LOCK=NONE;	dble	
16			ALTER TABLE <i>tbl_name</i> ALTER COLUMN <i>col</i> SET DEFAULT <i>literal</i> , ALGORITHM=INSTANT;		
17			ALTER TABLE <i>tbl</i> ALTER COLUMN <i>col</i> DROP DEFAULT, ALGORITHM=INSTANT;		
18			ALTER TABLE <i>table</i> AUTO_INCREMENT= <i>next_value</i> , ALGORITHM=INPLACE, LOCK=NONE;		
19	NULL		ALTER TABLE <i>tbl_name</i> MODIFY COLUMN <i>column_name</i> <i>data_type</i> NULL, ALGORITHM=INPLACE, LOCK=NONE;	dble	
20	NULL		ALTER TABLE <i>tbl_name</i> MODIFY COLUMN <i>column_name</i> <i>data_type</i> NOT NULL, ALGORITHM=INPLACE, LOCK=NONE;	dble	
21	ENUM SET		CREATE TABLE <i>t1</i> ( <i>c1</i> ENUM('a', 'b', 'c')); ALTER TABLE <i>t1</i> MODIFY COLUMN <i>c1</i> ENUM('a', 'b', 'c', 'd'), ALGORITHM=INSTANT;		
22			ALTER TABLE <i>tbl1</i> ADD CONSTRAINT <i>fk_name</i> FOREIGN KEY ( <i>col1</i> ) REFERENCES <i>tbl2</i> ( <i>col2</i> ) <i>referential_actions</i> ;	dble	
23			ALTER TABLE <i>tbl</i> DROP FOREIGN KEY <i>fk_name</i> ;		
24	Generated	Generated	ALTER TABLE <i>t1</i> ADD COLUMN ( <i>c2</i> INT GENERATED ALWAYS AS ( <i>c1</i> + 1) STORED), ALGORITHM=COPY;		
25		Generated	ALTER TABLE <i>t1</i> MODIFY COLUMN <i>c2</i> INT GENERATED ALWAYS AS ( <i>c1</i> + 1) STORED FIRST, ALGORITHM=COPY;		
26		Generated	ALTER TABLE <i>t1</i> DROP COLUMN <i>c2</i> , ALGORITHM=INPLACE, LOCK=NONE;		
27	VIRTUAL		ALTER TABLE <i>t1</i> ADD COLUMN ( <i>c2</i> INT GENERATED ALWAYS AS ( <i>c1</i> + 1) VIRTUAL), ALGORITHM=INSTANT;		
28		VIRTUAL	ALTER TABLE <i>t1</i> MODIFY COLUMN <i>c2</i> INT GENERATED ALWAYS AS ( <i>c1</i> + 1) VIRTUAL FIRST, ALGORITHM=COPY;		
29		VIRTUAL	ALTER TABLE <i>t1</i> DROP COLUMN <i>c2</i> , ALGORITHM=INSTANT;		
30				MySQL	
31				dble	

#### 3.1.6.3

1. dble druidonline ddl "ALGORITHM=INPLACE, LOCK=NONE" <https://github.com/alibaba/druid/issues/3750>

## 3.2 DML

### DML

- 3.2.1 INSERT
- 3.2.2 REPLACE
- 3.2.3 DELETE
- 3.2.4 UPDATE
- 3.2.5 SELECT
- 3.2.6 SELECT JOIN syntax
- 3.2.7 SELECT UNION Syntax
- 3.2.8 SELECT Subquery Syntax
- 3.2.9 LOAD DATA
- 3.2.10

### 3.2.1 INSERT

#### 3.2.1.1 Syntax

```
INSERT [INTO] tbl_name
[(col_name,...)]
{VALUES | VALUE} ({expr }...)(...),...
[ ON DUPLICATE KEY UPDATE
  col_name=expr
  [, col_name=expr] ... ]
OR
INSERT [INTO] tbl_name
SET col_name=(expr | DEFAULT), ...
[ ON DUPLICATE KEY UPDATE
  col_name=expr [, col_name=expr] ... ]
```

#### 3.2.1.2

```
insert into test (col1,col3) values(1,'cust1'),(2,'cust2');
insert into test (col1,col3) values(default,'cust3');
insert into test set col1=4,col3='cust4';
insert into test set col1=default,col3='cust5';
insert into test (col1,col3) values(default,cast(now() as char));
```

#### 3.2.1.3

- ERROW
- dble
- CREATE TABLE `test` (`id` enum('1','2','3') DEFAULT '1')id'1'dbleissue <https://github.com/actiontech/dble/issues/816>
- insert/replace... select dbleSQL
  - 
  - 
  - select

### 3.2.2 REPLACE

#### 3.2.2.1 Syntax

REPLACE

[INTO] tbl\_name [(col\_name [, col\_name] ...)]

{VALUES | VALUE} {value\_list} [, (value\_list)] ...

OR

REPLACE

[INTO] tbl\_name SET assignment\_list

#### 3.2.2.2

```
REPLACE INTO test VALUES (1, 'Old', '2014-08-20 18:47:00');  
REPLACE INTO test set id = 1, type= 'Old',create_date = '2014-08-20 18:47:00';
```

#### 3.2.2.3

- replace
- replaceIDIDID
- insert/replacedbleSQL
  - 
  - 
  - select

### 3.2.3 DELETE

#### 3.2.3.1 Syntax

DELETE [IGNORE]

FROM tbl\_name [WHERE where\_condition]

#### 3.2.3.2

```
delete from test where id>5;
```

#### 3.2.3.3

- Deletewhere\_condition
- Join DELETE
- DELETEDELETE
  - update/deletewhere where
  - update/deletewherewhere

### 3.2.4 UPDATE

#### 3.2.4.1 Syntax

UPDATE **table\_reference**

SET **col\_name1= {expr1} [, col\_name2= {expr2}] ...**

[WHERE **where\_condition**]

#### 3.2.4.2

```
UPDATE test SET VALUE =1 where id=5;
```

#### 3.2.4.3

- UPDATEwhere\_condition
- Join UPDATE
- updateDELETE
  - update/deletewhere where
  - update/deletewherewhere

### 3.2.5 SELECT

#### 3.2.5.1 Syntax

```
SELECT  
[ALL | DISTINCT | DISTINCTROW ]  
select_expr  
[, select_expr ...]  
[FROM table_references [WHERE where_condition]  
[GROUP BY {col_name | expr | position} [ASC | DESC], ...]  
[HAVING where_condition] [ORDER BY {col_name | expr | position} [ASC | DESC], ...]  
[LIMIT {[offset,] row_count | row_count OFFSET offset}]
```

#### 3.2.5.2

```
select id,col1,col3 from test where id=3;  
select distinct col1,col3 from test where id>=3;  
select count(*),max(id),col1 from test group by col1 desc having(count(*)>1) order by col1 desc;  
select id,col1,col3 from test order by id limit 2 offset 2;  
select id,col1,col3 from test order by id limit 2,2;  
select 1+1,'test',id,col1*1.1,now() from test limit 3;  
select current_date,current_timestamp;
```

### 3.2.6 JOIN Syntax:

table\_references:

```
table_reference [, table_reference] ...
```

table\_reference:

```
table_factor | join_table
```

table\_factor:

```
tbl_name [(AS) alias]  
| table_subquery [AS] alias  
| ( table_references )
```

join\_table:

```
table_reference [(INNER | CROSS) JOIN table_factor [join_condition]  
| table_reference STRAIGHT_JOIN table_factor  
| table_reference STRAIGHT_JOIN table_factor ON conditional_expr  
| table_reference {LEFT|RIGHT} [OUTER] JOIN table_reference join_condition  
| table_reference NATURAL [{LEFT|RIGHT} [OUTER]] JOIN table_factor
```

join\_condition:

```
ON conditional_expr  
| USING (column_list)
```

### 3.2.7 UNION Syntax:

```
SELECT ...  
UNION [ALL | DISTINCT] SELECT ...  
[UNION [ALL | DISTINCT] SELECT ...]
```

### 3.2.8 Subquery

#### 3.2.8.1 The Subquery as Scalar Operand

For example :

```
SELECT (SELECT s2 FROM t1);
SELECT (SELECT s1 FROM t2) FROM t1;
SELECT UPPER((SELECT s1 FROM t1)) FROM t2;
```

#### 3.2.8.2 Comparisons Using Subqueries

The most common use of a subquery is in the form:

```
non_subquery_operand comparison_operator (subquery)
```

Where comparison\_operator is one of these operators:

```
= > < >= <= <> != <=>
```

MySQL also permits this construct:

```
non_subquery_operand LIKE (subquery)
```

#### 3.2.8.3 Subqueries with ANY, IN, or SOME

Syntax:

```
operand comparison_operator ANY (subquery)
operand IN (subquery)
operand comparison_operator SOME (subquery)
```

Where comparison\_operator is one of these operators:

```
= > < >= <= <> !=
```

#### 3.2.8.4 Subqueries with ALL

Syntax:

```
operand comparison_operator ALL (subquery)
```

#### 3.2.8.5 Subqueries with EXISTS or NOT EXISTS

For example:

```
SELECT column1 FROM t1 WHERE EXISTS (SELECT * FROM t2);
```

Not support Correlated Subqueries for now.

#### 3.2.8.6 Derived Tables (Subqueries in the FROM Clause)

```
SELECT ... FROM (subquery) [AS] tbl_name ...
```

### 3.2.9 LOAD DATA

#### 3.2.9.1 Syntax

```
LOAD DATA
[LOCAL]
INFILE 'file_name' INTO TABLE tbl_name
[CHARACTER SET charset_name]
[ {FIELDS | COLUMNS}
[TERMINATED BY 'string']
[[OPTIONALLY] ENCLOSED BY 'char']
[ESCAPED BY 'char'] ]
[LINES [STARTING BY 'string']
[TERMINATED BY 'string']]
```

#### 3.2.9.2

```
load data infile 'data.txt' into table test_table CHARACTER SET 'utf8' FIELDS TERMINATED by ',';
```

#### 3.2.9.3

```
dbleMySQL,maxRowSizeToFile(bootstrap.cnf) load data local infile
local_infileload data.
#1085
```

#### 3.2.9.4

- BUGdbleCHARACTER SET charset\_name
- mysqldbleload data
- ENCLOSURE BYBUG
- loaddata65535bootstrap.cnfmaxCharsPerColumn
- load data,,,dble.issue: <https://github.com/actiontech/dble/issues/770>
- load data,druid,,druid.issue: <https://github.com/actiontech/dble/issues/1248>
- load data: <https://github.com/actiontech/dble/issues/1507>
- load dataissue <https://github.com/actiontech/dble/issues/1761>

**3.2.10 DML****DO Syntax****HANDLER Syntax****LOAD XML Syntax**

### 3.3 PREPARE SQL Syntax

#### 3.3.1 PREPARE Syntax

```
PREPARE stmt_name FROM preparable_stmt
```

```
prepare stmt1 from "select * from a_test where id=?";
```

#### 3.3.2 EXECUTE Syntax

```
EXECUTE stmt_name  
[USING @var_name [, @var_name] ...]
```

```
SET @a = 1;  
EXECUTE stmt1 USING @a;
```

#### 3.3.3 DEALLOCATE PREPARE Syntax

```
{DEALLOCATE | DROP} PREPARE stmt_name
```

```
DROP PREPARE stmt1;
```

## 3.4 Transactional and Locking Statements

Transactional and Locking Statements

- [3.4.1 Lock&unlock](#)
- [3.4.2 XA](#)
- [3.4.3](#)
- [3.4.4 SET TRANSACTION Syntax](#)

### 3.4.1 Lock&unlock

#### 3.4.1.1 Syntax

```
LOCK TABLES tbl_name [[AS] alias] lock_type
```

lock\_type: READ | WRITE

UNLOCK TABLES

#### 3.4.1.2

```
lock tables test_table read;  
unlock tables;
```

#### 3.4.1.3

1. session
- 2.
3. xa lock table dblexa

### 3.4.2 XA

#### 3.4.2.1 Syntax

XA

set xa = {0|1}

START TRANSACTION;

BEGIN

SET autocommit = {0 | 1}

COMMIT

ROLLBACK

#### 3.4.2.2

1. DbleSQLBEGIN(druid)

2. lock table xa

### 3.4.3

#### 3.4.3.1 Syntax

```
START TRANSACTION;  
BEGIN  
SET autocommit = {0 | 1}  
  
COMMIT
```

```
ROLLBACK
```

#### 3.4.3.2

- DbleSQLBEGIN(druid)
- 2PC(xa)commit,XA

**3.4.4 SET TRANSACTION Syntax**

SET SESSION TRANSACTION ISOLATION LEVEL level

**level:**REPEATABLE READ  
| READ COMMITTED  
| READ UNCOMMITTED  
| SERIALIZABLE

SET @@SESSION.TX\_ISOLATION = 'level\_str'

**level\_str:** REPEATABLE-READ  
| READ-COMMITTED  
| READ-UNCOMMITTED  
| SERIALIZABLE

session

### 3.4.5 SAVEPOINT, ROLLBACK TO SAVEPOINT, and RELEASE SAVEPOINT Syntax

#### 3.4.5.1 Syntax

SAVEPOINT *identifier*

ROLLBACK [WORK] TO [SAVEPOINT] *identifier*

RELEASE SAVEPOINT *identifier*

#### 3.4.5.1

```
# start transaction
set autocommit = 0;

# savepoint
savepoint s0;
insert into test value(1);
savepoint s1;
insert into test value(2);
savepoint s2;
insert into test value(3);

# rollback to
rollback to s0

# release
release savepoint s0
```

#### 3.4.5.2

1. mysql.savepoint,savepoint.dblesavepoint,.
2. ROLLBACK TO [SAVEPOINT] *identifier* work .

### 3.5 DAL

DAL

- 3.5.1 SET
- 3.5.2 SHOW
- 3.5.3 KILL

**3.5.1.1 XA**

```
set xa=value

value:
  0
  | off
  | false
  | 1
  | on
  | true
```

```
set xa=1
```

XA

**3.5.1.2 AUTOCOMMIT**

```
set autocommit=value

value:
  0
  | off
  | false
  | 1
  | on
  | true
```

```
set autocommit=1
```

AUTOCOMMIT

**3.5.1.3 NAMES**

```
SET NAMES {'charset_name' [COLLATE 'collation_name'] | DEFAULT}
```

```
set names utf8;
set names utf8 collate utf8_general_ci;
set names default;
```

**3.5.1.4 CHARSET**

```
SET {CHARACTER SET | CHARSET}
{'charset_name' | DEFAULT}
```

```
set CHARACTER SET utf8;
```

**3.5.1.5 COLLATION\_CONNECTION/CHAESET\_SET\_X**

```
SET COLLATION_CONNECTION='collation_name'
SET CHARSET_SET_CLIENT='charset_name'
SET CHARSET_SET_RESULTS='charset_name' 'charset_name' NULL
SET CHARSET_SET_CONNECTION='charset_name'
```

```
set collation_connection=utf8_general_ci;
set CHARSET_SET_CLIENT=utf8
set CHARSET_SET_RESULTS=utf8;
set CHARSET_SET_CONNECTION=utf8;
```

**3.5.1.6 TRANSACTION ACCESS MODE**

```
SET SESSION { TX_READ_ONLY | TRANSACTION_READ_ONLY}=value

value:
  0
  | off
  | false
  | 1
  | on
  | true
```

```
set session @@tx_read_only=1;
```

**3.5.1.7 TRANSACTION ISOLATION LEVEL**

```
SET SESSION {TRANSACTION_ISOLATION | TX_ISOLATION}=level

level:
  READ-UNCOMMITTED | READ-COMMITTED | REPEATABLE-READ | SERIALIZABLE
```

```
SET SESSION TX_ISOLATION=READ-COMMITTED;
```

**3.5.1.8 USER/SYSTEM VARIABLE**

```
SET variable_assignment[, variable_assignment ] ...

variable_assignment:
  @user_var_name = expr
  | SESSION system_var_name = expr
  | system_var_name = expr
  | @@system_var_name = expr
  | @@session.system_var_name = expr
```

1.  
2.

```

audit_log_current_session
audit_log_filter_id
auto_increment_increment
auto_increment_offset
autocommit
big_tables
binlog_direct_non_transactional_updates
binlog_error_action
binlog_format
binlog_row_image
binlog_rows_query_log_events
binlogging_impossible_mode
block_encryption_mode
bulk_insert_buffer_size
character_set_client
character_set_connection
character_set_database
character_set_filesystem
character_set_results
character_set_server
collation_connection
collation_database
collation_server
completion_type
debug
debug_sync
default_storage_engine
default_tmp_storage_engine
default_week_format
disconnect_on_expired_password
div_precision_increment
end_markers_in_json
eq_range_index_dive_limit
error_count
explicit_defaults_for_timestamp
external_user
foreign_key_checks
group_concat_max_len
gtid_next
gtid_owned
identity
innodb_create_intrinsic
innodb_ft_user_stopword_table
innodb_lock_wait_timeout
innodb_optimize_point_storage
innodb_strict_mode
innodb_support_xa
innodb_table_locks
innodb_tmpdir
insert_id
interactive_timeout
join_buffer_size
keep_files_on_create
last_insert_id
lc_messages
lc_time_names
lock_wait_timeout
long_query_time
low_priority_updates
max_allowed_packet
max_delayed_threads
max_error_count
max_execution_time
max_heap_table_size
max_insert_delayed_threads
max_join_size
max_length_for_sort_data
max_seeks_for_key
max_sort_length
max_sp_recursion_depth
max_statement_time
max_user_connections
min_examined_row_limit
myisam_repair_threads
myisam_sort_buffer_size
myisam_stats_method
ndb-allow-copying-alter-table
ndb_autoincrement_prefetch_sz
ndb-blob-read-batch-bytes
ndb-blob-write-batch-bytes
ndb_deferred_constraints
ndb_force_send
ndb_fully_replicated
ndb_index_stat_enable
ndb_index_stat_option
ndb_join_pushdown
ndb_log_bin
ndb_log_bin
ndb_table_no_logging
ndb_table_temporary
ndb_use_copying_alter_table
ndb_use_exact_count
ndb_use_transactions
ndbinfo_max_bytes
ndbinfo_max_rows
ndbinfo_show_hidden
ndbinfo_table_prefix
net_buffer_length
net_read_timeout
net_retry_count
net_write_timeout
new
old_alter_table
old_passwords
optimizer_prune_level
optimizer_search_depth
optimizer_switch
optimizer_trace
optimizer_trace_features
optimizer_trace_limit
optimizer_trace_max_mem_size
optimizer_trace_offset
parser_max_mem_size
preload_buffer_size
profiling
profiling_history_size
proxy_user
pseudo_slave_mode
pseudo_thread_id
query_alloc_block_size
query_cache_type
query_cache_wlock_invalidate
query_prealloc_size
rand_seed1
rand_seed2
range_alloc_block_size
range_optimizer_max_mem_size
rbr_exec_mode
read_buffer_size
read_rnd_buffer_size
session_track_gtids
session_track_schema
session_track_state_change
session_track_system_variables
show_old_temporals
sort_buffer_size
sql_auto_is_null
sql_big_selects
sql_buffer_result
sql_log_bin
sql_log_off
sql_mode
sql_notes
sql_quote_show_create
sql_safe_updates
sql_select_limit
sql_warnings
storage_engine
thread_pool_high_priority_connection

```

```
thread_pool_prio_kickup_timer  
time_zone  
timestamp  
tmp_table_size  
transaction_alloc_block_size  
transaction_allow_batching  
transaction_prealloc_size  
transaction_write_set_extraction  
tx_isolation  
tx_read_only  
unique_checks  
updatable_views_with_limit  
version_tokens_session  
version_tokens_session_number  
wait_timeout  
warning_count
```

```
set @a=20  
SET SESSION sql_mode = 'TRADITIONAL';  
SET sql_mode = 'TRADITIONAL';
```

1. insert\_id sql\_auto\_is\_null insert\_id issue <https://github.com/actiontech/dble/issues/1252>.

### 3.5.1.9 TRACE

```
SQL,      show trace  
select @@trace      SQLtrace
```

```
set trace=value
```

```
value:  
  0  
  | off  
  | false  
  | 1  
  | on  
  | true
```

```
set trace=1
```

### 3.5.2 SHOW

#### 3.5.2.1 dbleSHOW

- SHOW DATABASES  
sharding.xml schema
- SHOW CREATE DATABASE [IF NOT EXISTS] schema  
sharding.xml schemadble
- SHOW [FULL|ALL] TABLES [FROM db\_name] [LIKE 'pattern'] WHERE expr  
schmeaschematables  
schemaschematables
- SHOW ALL TABLES [FROM db\_name] [LIKE 'pattern'] WHERE expr  
dbleSHOW FULL TABLES Table\_type SHARDING TABLEsharding table GLOBAL TABLE [6.Differenrece\\_from\\_SQL\\_Server.md](#)
- SHOW [FULL] {COLUMNS|FIELDS} FROM tbl\_name [{FROM|IN} db\_name] [LIKE 'pattern'] WHERE expr  
schemaschema
- SHOW { INDEX | INDEXES | KEYS } {FROM | IN} tbl\_name [ {FROM | IN} db\_name ] [ WHERE expr ]  
schemaschema
- SHOW CREATE TABLE tbl\_name  
schemaschema
- SHOW [GLOBAL | SESSION] VARIABLES [LIKE 'pattern'] WHERE expr  
global
- SHOW CREATE VIEW view\_name  
dbview
- SHOW CHARSET  
show character set
- SHOW TABLE STATUS [{FROM | IN} db\_name] [LIKE 'pattern'] WHERE expr  
SQLyognameshow tables
- SHOW TRACE  
trace [SQLtrace](#)

explain

```
show databases
show full tables
show columns from a_test;
show index from a_test;
show create table a_test;
show variables;
show charset;
```

#### 3.5.2.2 dbleSHOW

dbleSHOWSHOWmysql

```
SHOW CHARACTER SET;
SHOW CHARACTER SET like 'utf8';
SHOW CHARACTER SET where maxlen=2;
```

### 3.5.3 KILL

#### 3.5.3.1 KILL conn\_id

conn\_id idshow @@connection

##### 3.5.3.1.1

```
kill 1;
```

##### 3.5.3.1.2

- KillOK
- KillXA
- MYSQL KILL processlist\_id

#### 3.5.3.2 KILL query conn\_id

conn\_id idshow @@connection

##### 3.5.3.2.1

```
kill query 1;
```

##### 3.5.3.2.3

- dble kill query
- mysql

##### 3.5.3.2.2

- ddl
- dml

## 3.6

### 3.6.1 Syntax

#### Create procedure

```
/Hint/ CREATE [DEFINER = { user | CURRENT_USER }]
PROCEDURE sp_name ([proc_parameter[...]])
[characteristic ...] routine_body
/Hint/ CREATE
[DEFINER = { user | CURRENT_USER }]
FUNCTION sp_name ([func_parameter[...]])
RETURNS type [characteristic ...] routine_body
```

#### drop procedure

```
/Hint/ DROP {PROCEDURE | FUNCTION} [IF EXISTS] sp_name
```

#### call procedure

```
[/Hint/] CALL sp_name([parameter[...]])
[/Hint/] CALL sp_name()
```

## 3.6.2

```
/*!dbe:sql=select 1 from account */drop procedure if exists proc_arc;
/*!dbe:sql=select 1 from account */create procedure proc_arc(userid1 int)
begin
    insert into account_arc select * from account where userid=userid1;
    update account set arc_flag=true,arc_time=now() where userid=userid1;
end;
/*!dbe:sql=select 1 from account */call proc_arc(1);
```

## 3.6.3

- dbleMySQL
- MySQL
- dble

## 3.7 Utility Statements

### 3.7.1 USE

```
USE db_name
```

```
use TESTDB;
```

### 3.7.2 EXPLAIN

```
EXPLAIN explainable_stmt  
SELECT statement  
| DELETE statement  
| INSERT statement  
| REPLACE statement  
| UPDATE statement
```

1. INSERT

```
explain SELECT select * from a_test where id=1;
```

2. dbleEXPLAIN DESC

### 3.7.3 EXPLAIN2

```
EXPLAIN2 shardingNode=node_name sql$sql_stmt
```

```
explain2 shardingNode=dn2 sql=select * from a_test where id=1;
```

### 3.7.4 DESC

```
{DESCRIBE | DESC} tbl_name [col_name | wild]
```

```
DESC a_test id;
```

:dbleEXPLAIN DESC

## 3.8 Hint

### 3.8.1 -Syntax

```
/* { ! | #}dble: {sql=SELECT select_expr FROM table_references WHERE where_condition
|shardingNode=shardingNode_name
|db_type={slave|master}}
*/
ordinary_sql
```

### 3.8.2 -Syntax

```
/* { ! | #}dble: {db_type={slave|master}}
|db_instance_url={ip:port}}
*/
ordinary_sql
/* master */ ordinary_sql
/* uproxy_dest: ip:port */ ordinary_sql
```

• sql /\* master \*/ /\* uproxy\_dest: ip:port \*/ sql

### 3.8.3

```
/*!dbe:sql=select 1 from sbtest */ call p_show_time();
/*!dbe:shardingNode=dn1*/ update sbtest set name = 'test';
/*!dbe:db_type=master*/ select count(*) from sbtest;
/*!dbe:db_instance_url=127.0.0.1:3307*/ select count(*) from sbtest;
/*#dbe:sql=select 1 from sbtest */ call p_show_time();
/*#dbe:shardingNode=dn1*/ update sbtest set name = 'test';
/*#dbe:db_type=master*/ select count(*) from sbtest;
/*#dbe:db_instance_url=127.0.0.1:3307*/ select count(*) from sbtest;
select /* master */ * from sbtest;
show processlist /* uproxy_dest: 127.0.0.1:3307 */
```

### 3.9

- Compound-Statement Syntax
- Replication Statements
- DDL
  - databasealter databasedrop databasecreate database schemaok
  - create tabletable optionDATA DIRECTORYALGORITHM table optionalter table
  - ALTER TABLE ... LOCK ...
  - ALTER TABLE ... ORDER BY ...
  - create table ... like ...create table ... select ...
  - 
  - 
  - 
  - 
  -
- DML
  - INSERT... VALUES(expr)expr
  - INSERT DELAYED...
  - INSERT... SELECT...
  - INSERT
  - HANDLER
  - UPDATE
  - 
  - DELETE ... ORDER BY ... LIMIT ...
  - DELETE
  - DELETE/UPDATE ...LIMIT
  - DO
  - - select ... use/ignore index ...
    - select ... group by ... with rollup
    - select ... for update | lock in share mode
    - select ... into outfile ...
    - Row Subqueries
    - select ... union [all] select ... order by ... (select ...) union [all] (select ...) order by ...
    - sessionset @rowid=0;select @rowid:=@rowid+1,id from user;
    - 
    - ANALYZE/CHECK/CHECKSUM/OPTIMIZE/REPAIR TABLE
    - INSTALL/UNINSTALL PLUGIN
    - BINLOG
    - CACHE INDEX/ LOAD INDEX INTO CACHE
    - FLUSH TABLES [WITH READ LOCK]FLUSHFLUSH TABLE
    - RESET
    - SHOWSHOW PROFILESSHOW ERRORS

## 3.10 (alpha)

### 3.10.0

1. SQLMySQL.
2. 2.18.09.0 bug
- 3.
- 4.

### 3.10.1 Operators

Name	Description	Support
AND, &&	Logical AND	Y
=	Assign a value (as part of a SET statement, or as part of the SET clause in an UPDATE statement)	Y
:=	Assign a value	N
BETWEEN ... AND ...	Check whether a value is within a range of values	Y
BINARY	Cast a string to a binary string	N
&	Bitwise AND	Y
~	Bitwise inversion	Y
	Bitwise OR	Y
^	Bitwise XOR	Y
CASE	Case operator	Y
DIV	Integer division	Y
/	Division operator	Y
=	Equal operator	Y
<=>	NULL-safe equal to operator	Y
>	Greater than operator	Y
>=	Greater than or equal operator	Y
IS	Test a value against a boolean	Y
IS NOT	Test a value against a boolean	Y
IS NOT NULL	NOT NULL value test	Y
IS NULL	NULL value test	Y
->	Return value from JSON column after evaluating path; equivalent to JSON_EXTRACT().	N
->>	Return value from JSON column after evaluating path and unquoting the result; equivalent to JSON_UNQUOTE(JSON_EXTRACT()).	N
<<	Left shift	Y
<	Less than operator	Y
<=	Less than or equal operator	Y
LIKE	Simple pattern matching	Y
-	Minus operator	Y
%, MOD	Modulo operator	Y
NOT, !	Negates value	Y
NOT BETWEEN ... AND ...	Check whether a value is not within a range of values	Y
!=, <>	Not equal operator	Y
NOT LIKE	Negation of simple pattern matching	Y
NOT REGEXP	Negation of REGEXP	Y
, OR	Logical OR	Y
+	Addition operator	Y
REGEXP	Whether string matches regular expression	Y
>>	Right shift	Y
RLIKE	Whether string matches regular expression	N
SOUNDS LIKE	Compare sounds	N
*	Multiplication operator	N
-	Change the sign of the argument	Y
XOR	Logical XOR	Y
COALESCE()	Return the first non-NULL argument	Y
GREATEST()	Return the largest argument	Y
IN()	Check whether a value is within a set of values	Y
INTERVAL()	Return the index of the argument that is less than the first argument	Y
ISNULL()	Test whether the argument is NULL	Y
LEAST()	Return the smallest argument	Y
STRCMP()	Compare two strings	Y

### 3.10.2 Control Flow Functions

Name	Description	Support
CASE	Case operator	Y
IF()	If/else construct	Y
IFNULL()	Null if/else construct	Y
NULLIF()	Return NULL if expr1 = expr2	Y

### 3.10.3 String Functions

Name	Description	Support
ASCII()	Return numeric value of left-most character	Y
BIN()	Return a string containing binary representation of a number	N
BIT_LENGTH()	Return length of argument in bits	Y
CHAR()	Return the character for each integer passed	Y
CHAR_LENGTH()	Return number of characters in argument	Y
CHARACTER_LENGTH()	Synonym for CHAR_LENGTH()	Y
CONCAT()	Return concatenated string	Y
CONCAT_WS()	Return concatenate with separator	Y
ELT()	Return string at index number	Y
EXPORT_SET()	Return a string such that for every bit set in the value bits, you get an on string and for every unset bit, you get an off string	N
FIELD()	Return the index (position) of the first argument in the subsequent arguments	Y
FIND_IN_SET()	Return the index position of the first argument within the second argument	Y

FORMAT()	Return a number formatted to specified number of decimal places	Y
FROM_BASE64()	Decode base64 encoded string and return result	N
HEX()	Return a hexadecimal representation of a decimal or string value	Y
INSERT()	Insert a substring at the specified position up to the specified number of characters	Y
INSTR()	Return the index of the first occurrence of substring	Y
LCASE()	Synonym for LOWER()	Y
LEFT()	Return the leftmost number of characters as specified	Y
LENGTH()	Return the length of a string in bytes	Y
LIKE	Simple pattern matching	Y
LOAD_FILE()	Load the named file	N
LOCATE()	Return the position of the first occurrence of substring	Y
LOWER()	Return the argument in lowercase	Y
LPAD()	Return the string argument, left-padded with the specified string	Y
LTRIM()	Remove leading spaces	Y
MAKE_SET()	Return a set of comma-separated strings that have the corresponding bit in bits set	Y
MATCH	Perform full-text search	N
MID()	Return a substring starting from the specified position	N
NOT LIKE	Negation of simple pattern matching	Y
NOT REGEXP	Negation of REGEXP	Y
OCT()	Return a string containing octal representation of a number	N
OCTET_LENGTH()	Synonym for LENGTH()	N
ORD()	Return character code for leftmost character of the argument	Y
POSITION()	Synonym for LOCATE()	N
QUOTE()	Escape the argument for use in an SQL statement	Y
REGEXP	Whether string matches regular expression	Y
REPEAT()	Repeat a string the specified number of times	Y
REPLACE()	Replace occurrences of a specified string	Y
REVERSE()	Reverse the characters in a string	Y
RIGHT()	Return the specified rightmost number of characters	Y
RLIKE	Whether string matches regular expression	N
RPAD()	Append string the specified number of times	Y
RTRIM()	Remove trailing spaces	Y
SOUNDEX()	Return a soundex string	Y
SOUNDS LIKE	Compare sounds	Y
SPACE()	Return a string of the specified number of spaces	Y
STRCMP()	Compare two strings	Y
SUBSTRO	Return the substring as specified	Y
SUBSTRING()	Return the substring as specified	Y
SUBSTRING_INDEX()	Return a substring from a string before the specified number of occurrences of the delimiter	Y
TO_BASE64()	Return the argument converted to a base-64 string	N
TRIM()	Remove leading and trailing spaces	Y
UCASE()	Synonym for UPPER()	Y
UNHEX()	Return a string containing hex representation of a number	Y
UPPER()	Convert to uppercase	Y
WEIGHT_STRING()	Return the weight string for a string	N

### 3.10.4 Numeric Functions and Operators

Name	Description	Support
ABS()	Return the absolute value	Y
ACOS()	Return the arc cosine	Y
ASIN()	Return the arc sine	Y
ATAN()	Return the arc tangent	Y
ATAN2(), ATAN()	Return the arc tangent of the two arguments	Y
CEIL()	Return the smallest integer value not less than the argument	Y
CEILING()	Return the smallest integer value not less than the argument	Y
CONV()	Convert numbers between different number bases	Y
COS()	Return the cosine	Y
COT()	Return the cotangent	Y
CRC32()	Compute a cyclic redundancy check value	Y
DEGREES()	Convert radians to degrees	Y
DIV	Integer division	Y
/	Division operator	Y
EXP()	Raise to the power of	Y
FLOOR()	Return the largest integer value not greater than the argument	Y
LN()	Return the natural logarithm of the argument	Y
LOG()	Return the natural logarithm of the first argument	Y
LOG10()	Return the base-10 logarithm of the argument	Y
LOG2()	Return the base-2 logarithm of the argument	Y
-	Minus operator	Y
MOD()	Return the remainder	Y
%, MOD	Modulo operator	Y
PI()	Return the value of pi	Y
+	Addition operator	Y
POW()	Return the argument raised to the specified power	Y
POWER()	Return the argument raised to the specified power	Y
RADIANS()	Return argument converted to radians	Y
RAND()	Return a random floating-point value	Y
ROUND()	Round the argument	Y
SIGN()	Return the sign of the argument	Y
SIN()	Return the sine of the argument	Y
SQRT()	Return the square root of the argument	Y

TAN()	Return the tangent of the argument	Y
*	Multiplication operator	Y
TRUNCATE()	Truncate to specified number of decimal places	Y
-	Change the sign of the argument	Y

### 3.10.5 Date and Time Functions

Name	Description	Support
ADDDATE()	Add time values (intervals) to a date value	Y
ADDTIME()	Add time	Y
CONVERT_TZ()	Convert from one time zone to another	N
CURDATE()	Return the current date	Y
CURRENT_DATE()	Synonyms for CURDATE()	Y
CURRENT_TIME()	Synonyms for CURTIME()	Y
CURRENT_TIMESTAMP()	Synonyms for NOW()	Y
CURTIME()	Return the current time	Y
DATE()	Extract the date part of a date or datetime expression	Y
DATE_ADD()	Add time values (intervals) to a date value	Y
DATE_FORMAT()	Format date as specified	Y
DATE_SUB()	Subtract a time value (interval) from a date	Y
DATEDIFF()	Subtract two dates	Y
DAY()	Synonym for DAYOFMONTH()	N
DAYNAME()	Return the name of the weekday	Y
DAYOFMONTH()	Return the day of the month (0-31)	Y
DAYOFWEEK()	Return the weekday index of the argument	Y
DAYOFYEAR()	Return the day of the year (1-366)	Y
EXTRACT()	Extract part of a date	Y
FROM_DAYS()	Convert a day number to a date	Y
FROM_UNIXTIME()	Format Unix timestamp as a date	Y
GET_FORMAT()	Return a date format string	Y
HOUR()	Extract the hour	Y
LAST_DAY	Return the last day of the month for the argument	N
LOCALTIME()	Synonym for NOW()	Y
LOCALTIMESTAMP()	Synonym for NOW()	Y
MAKEDATE()	Create a date from the year and day of year	Y
MAKETIME()	Create time from hour, minute, second	Y
MICROSECOND()	Return the microseconds from argument	Y
MINUTE()	Return the minute from the argument	Y
MONTH()	Return the month from the date passed	Y
MONTHNAME()	Return the name of the month	Y
NOW()	Return the current date and time	Y
PERIOD_ADD()	Add a period to a year-month	Y
PERIOD_DIFF()	Return the number of months between periods	Y
QUARTER()	Return the quarter from a date argument	Y
SEC_TO_TIME()	Converts seconds to 'HH:MM:SS' format	Y
SECOND()	Return the second (0-59)	Y
STR_TO_DATE()	Convert a string to a date	Y
SUBDATE()	Synonym for DATE_SUB() when invoked with three arguments	Y
SUBTIME()	Subtract times	Y
SYSDATE()	Return the time at which the function executes	Y
TIME()	Extract the time portion of the expression passed	Y
TIME_FORMAT()	Format as time	Y
TIME_TO_SEC()	Return the argument converted to seconds	Y
TIMEDIFF()	Subtract time	Y
TIMESTAMP()	With a single argument, this function returns the date or datetime expression; with two arguments, the sum of the arguments	N
TIMESTAMPADD()	Add an interval to a datetime expression	Y
TIMESTAMPDIFF()	Subtract an interval from a datetime expression	Y
TO_DAYS()	Return the date argument converted to days	Y
TO_SECONDS()	Return the date or datetime argument converted to seconds since Year 0	Y
UNIX_TIMESTAMP()	Return a Unix timestamp	Y
UTC_DATE()	Return the current UTC date	Y
UTC_TIME()	Return the current UTC time	Y
UTC_TIMESTAMP()	Return the current UTC date and time	Y
WEEK()	Return the week number	Y
WEEKDAY()	Return the weekday index	Y
WEEKOFYEAR()	Return the calendar week of the date (1-53)	Y
YEAR()	Return the year	Y
YEARWEEK()	Return the year and week	Y
CURRENT_DATE	Synonyms for CURDATE()	N
CURRENT_TIME	Synonyms for CURTIME()	N
CURRENT_TIMESTAMP	Synonyms for NOW()	N
LOCALTIME	Synonym for NOW()	N
LOCALTIMESTAMP()	Synonym for NOW()	N

### 3.10.6 Cast Functions and Operators

Name	Description	Support
BINARY	Cast a string to a binary string	N
CAST()	Cast a value as a certain type	Y
CONVERT()	Cast a value as a certain type	Y

#### 3.10.6.1 CAST

BINARY  
 CHAR(N) [charset\_info] charset\_info  
 JSON  
 SIGNED [INTEGER] INTEGER(druid)  
 UNSIGNED [INTEGER] INTEGER (druid)

### 3.10.6.2 CONVERT

BINARY  
 CHAR(N) [charset\_info] charset\_info  
 JSON  
 SIGNED [INTEGER] INTEGER(druid)  
 UNSIGNED [INTEGER] INTEGER (druid)

### 3.10.7 Bit Functions and Operators

Name	Description	Support
BIT_COUNT()	Return the number of bits that are set	Y
&	Bitwise AND	Y
~	Bitwise inversion	Y
	Bitwise OR	Y
^	Bitwise XOR	Y
<<	Left shift	Y
>>	Right shift	Y

### 3.10.8 Aggregate (GROUP BY) Functions

Name	Description	Support
AVG()	Return the average value of the argument	Y
BIT_AND()	Return bitwise AND	Y
BIT_OR()	Return bitwise OR	Y
BIT_XOR()	Return bitwise XOR	Y
COUNT()	Return a count of the number of rows returned	Y
COUNT(DISTINCT)	Return the count of a number of different values	Y
GROUP_CONCAT()	Return a concatenated string	Y
JSON_ARRAYAGG()	Return result set as a single JSON array	N
JSON_OBJECTAGG()	Return result set as a single JSON object	N
MAX()	Return the maximum value	Y
MIN()	Return the minimum value	Y
STD()	Return the population standard deviation	Y
STDDEV()	Return the population standard deviation	Y
STDDEV_POP()	Return the population standard deviation	Y
STDDEV_SAMP()	Return the sample standard deviation	Y
SUM()	Return the sum	Y
VAR_POP()	Return the population standard variance	Y
VAR_SAMP()	Return the sample variance	Y
VARIANCE()	Return the population standard variance	Y

: STD VARIANCE  
 AVG\SUMdbleMySQL  
 MySQLMySQL,  
 for data type float, dble and mysql may get different results

### 3.10.9 Full-Text Search Functions

not supported

### 3.10.10 XML Functions

not supported

### 3.10.11 Encryption and Compression Functions

not supported

### 3.10.12 Information Functions

not supported

### 3.10.13 Spatial Analysis Functions

not supported

### 3.10.14 JSON Functions

not supported

### 3.10.15 Functions Used with Global Transaction IDs

not supported

### 3.10.16 MySQL Enterprise Encryption Functions

not supported

### 3.10.17 Miscellaneous Functions

not supported

[MySQL5.7](#)

1. workbench  
2.dbeaver  
3. mysqldump  
4. navicat  
5. mysqlsource load data

1. mysqldump mysqldump dble

```
./mysqldump -h127.0.0.1 -utest -P3306 -p11111 --default-character-set=utf8mb4 --master-data=2 --single-transaction --set-gtid-purged=off --hex-blob --databases schemai > export.sql
```

1.  
2. dbledble

**4**

- [4.1](#)
- [4.2](#)
- [4.3](#)
- [4.4 \(Prepared Statements\)](#)
- [4.5](#)

**4.1**

- 
- 16M
- ,bootstrap.cnf
-

## 4.2

### 4.2.1 Authentication Plugin

a.(mysql\_native\_password 8.0caching\_sha2\_password)  
 b.mysql\_native\_password  
 c.caching\_sha2\_password

### 4.2.2 Capabilities

```
dble
dbcdblCLIENT_FOUND_ROWS
dblIGNORE_SPACE select @@sql_mode IGNORE_SPACE
MySQL issue-972
IGNORE_SPACEsessionsql_mode
dbe
```

CLIENT_LONG_PASSWORD	1	Use the improved version of Old Password Authentication.Assumed to be set since 4.1.1.	Y	Y	
CLIENT_FOUND_ROWS	2	Send found rows instead of affected rows in EOF_Packet	Y	Y	
CLIENT_LONG_FLAG	4	Get all column flags.	Y	Y	
CONNECT_WITH_DB	8	Database (schema) name can be specified on connect in Handshake Response Packet.	Y	Y	
CLIENT_NO_SCHEMA	16	Don't allow database.table.column.	N	N	
CLIENT_COMPRESS	32	Compression protocol supported		bootstrap.cnfuseCompression	
CLIENT_ODBC	64	Special handling of ODBC behavior.No special behavior since 3.22.	Y	Y	
CLIENT_LOCAL_FILES	128	Can use LOAD DATA LOCAL.	Y	Y	
CLIENT_IGNORE_SPACE	256	Ignore spaces before '('.	Y	Y	
CLIENT_PROTOCOL_41	512	New 4.1 protocol	Y	Y	
CLIENT_INTERACTIVE	1024	This is an interactive client.	Y	Y	
CLIENT_SSL	2048	Use SSL encryption for the session	N	N	
CLIENT_IGNORE_SIGPIPE	4096	Client only flag.Not used.	Y	Y	
CLIENT_TRANSACTIONS	8192	Client knows about transactions	Y	Y	
CLIENT_RESERVED	16384	DEPRECATED:Old flag for 4.1 protocol.	N	N	
CLIENT_RESERVED2	32768	DEPRECATED:Old flag for 4.1 authentication.	Y	Y	
CLIENT_MULTI_STATEMENTS	65536	Enable/disable multi-stmt support	Y	Y	
CLIENT_MULTI_RESULTS	131072	Enable/disable multi-results	Y	Y	
CLIENT_PS_MULTI_RESULTS	262144	Multi-results and OUT parameters in PS-protocol	N	N	
CLIENT_PLUGIN_AUTH	524288	Client supports plugin authentication.	N	Y	
CLIENT_CONNECT_ATTRS	1048576	Client supports connection attributes.	N	N	
CLIENT_PLUGIN_AUTH_LENENC_CLIENT_DATA	2097152	Enable authentication response packet to be larger than 255 bytes.	N	N	
CLIENT_CAN_HANDLE_EXPIRED_PASSWORDS	4194304	Don't close the connection for a user account with expired password.	N	N	
CLIENT_SESSION_TRACK	8388608	Capable of handling server state change information.	N	N	
CLIENT_DEPRECATED_EOF	16777216	Client no longer needs EOF_Packet and will use OK_Packet instead.	N	N	
CLIENT_SSL_VERIFY_SERVER_CERT	1UL << 30	Verify server certificate	N	N	
CLIENT_REMEMBER_OPTIONS	1UL << 31	Don't reset the options after an unsuccessful connect.	N	N	

:

[https://dev.mysql.com/doc/dev/mysql-server/8.0.13/group\\_group\\_cs\\_capabilities\\_flags.html](https://dev.mysql.com/doc/dev/mysql-server/8.0.13/group_group_cs_capabilities_flags.html)

## 4.3

### 4.3.1 Supported

- COM\_INIT\_DB  
Specifies the default schema for the connection.
- COM\_PING  
Sends a packet containing one byte to check that the connection is active.
- COM\_QUERY  
Sends the server an SQL statement to be executed immediately. Support Multi-Statement.
- COM\_QUIT  
Client tells the server that the connection should be terminated.
- COM\_SET\_OPTION  
Enables or disables server option.
- COM\_CHANGE\_USER  
Resets the connection and re-authenticates with the given credentials.
- COM\_RESET\_CONNECTION  
Resets a connection without re-authentication.
  - (rollback & unlock)
  - 
  - 
  - 
  - prepare
  - 0
  - LAST\_INSERT\_ID
- COM\_FIELD\_LIST  
MySQL Doc said that it is deprecated from 5.7.11 . But some tools are still use it, like OGG or MariaDB client.

#### 4.3.1.1 Multi-Statement

- Supported
  - DML:select/insert/update/replace/delete
  - DDL
  - OTHER
    - BEGIN;
    - COMMIT;
    - LOCK TABLE
    - UNLOCK TABLES
    - START
    - KILL
    - USE
    - ROLLBACK
    - MYSQL\_CMD\_COMMENT
    - MYSQL\_COMMENT
    - SELECT VERSION\_COMMENT (SELECT @@VERSION\_COMMENT)
    - SELECT DATABASE select database()
    - SELECT USERselect user()
    - SELECT VERSION (select version())
    - SELECT SESSION\_INCREMENT(select @@session.auto\_increment\_increment)
    - SELECT SESSION\_ISOLATION(select @@session.tx\_isolation)
    - SELECT LAST\_INSERT\_ID(select last\_insert\_id() as id )
    - SELECT IDENTITY(select @@identity)
    - SELECT SESSION\_TX\_READ\_ONLYselect @@session.tx\_read\_only
- Not Supported
  - EXPLAIN
  - EXPLAIN2
  - DESCRIBE
  - SET
  - SHOW DATABASES/TABLES/TABLE\_STATUS/COLUMNS/INDEX/CREATE\_TABLE/VARIABLES/CREATE\_VIEW/CHARSET
  - HELP
  - LOAD\_DATA\_INFILE\_SQL
  - CREATE\_VIEW
  - REPLACE\_VIEW
  - ALTER\_VIEW
  - DROP\_VIEW

#### 4.3.2 Not Supported

- COM\_DEBUG  
Forces the server to dump debug information to stdout
- COM\_STATISTICS  
Get internal server statistics.
- COM\_CREATE\_DB
- COM\_DROP\_DB

#### 4.3.3 Internal

- COM\_SLEEP  
Used inside the server only.
- COM\_CONNECT an internal command in the server.
- COM\_TIME an internal command in the server.
- COM\_DAEMON an internal command in the server.
- COM\_DELAYED\_INSERT an internal command in the server.

#### 4.3.4 Deprecated

- COM\_PROCESS\_INFO  
Deprecated from 5.7.11.
- COM\_PROCESS\_KILL  
Deprecated from 5.7.11.
- COM\_SHUTDOWN  
Deprecated from 5.7.9.
- COM\_REFRESH  
Deprecated from 5.7.11.

## 4.4 (Prepared Statements)

### 4.4.1 Supported

- COM\_STMT\_CLOSE  
Closes a previously prepared statement.
- COM\_STMT\_EXECUTE  
Executes a previously prepared statement.
- COM\_STMT\_RESET  
Resets a prepared statement on client and server to state after preparing.
- COM\_STMT\_SEND\_LONG\_DATA  
When data for a specific column is big, it can be sent separately.

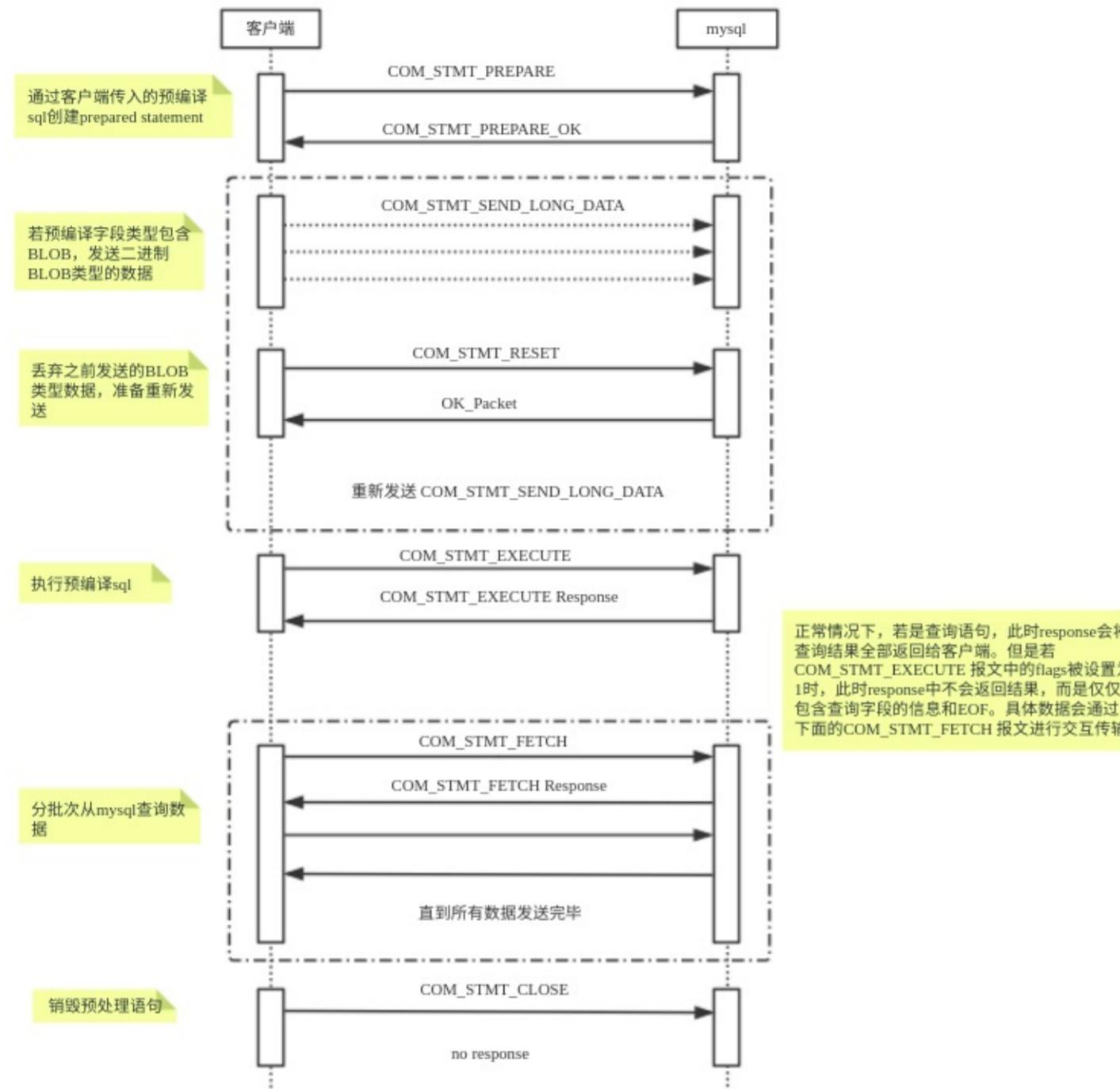
### 4.4.2 Faked

- COM\_STMT\_PREPARE  
Prepares a statement on the server  
COM\_STMT\_PREPARE response is faked. Parameters and numbers are faked.

### 4.4.3 Not Supported

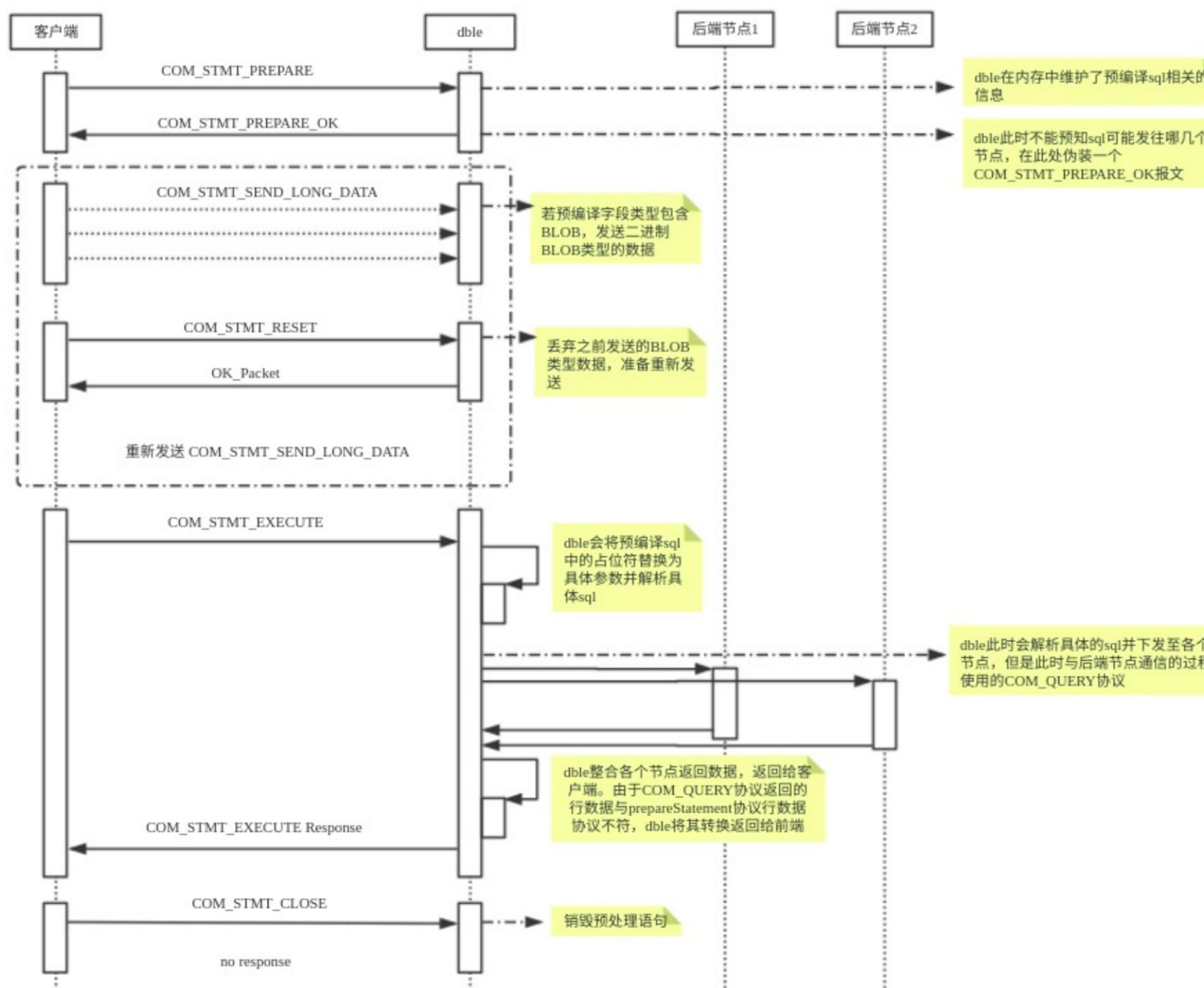
- COM\_STMT\_FETCH  
Fetches rows from a prepared statement

### 4.4.4 MySQL Flow

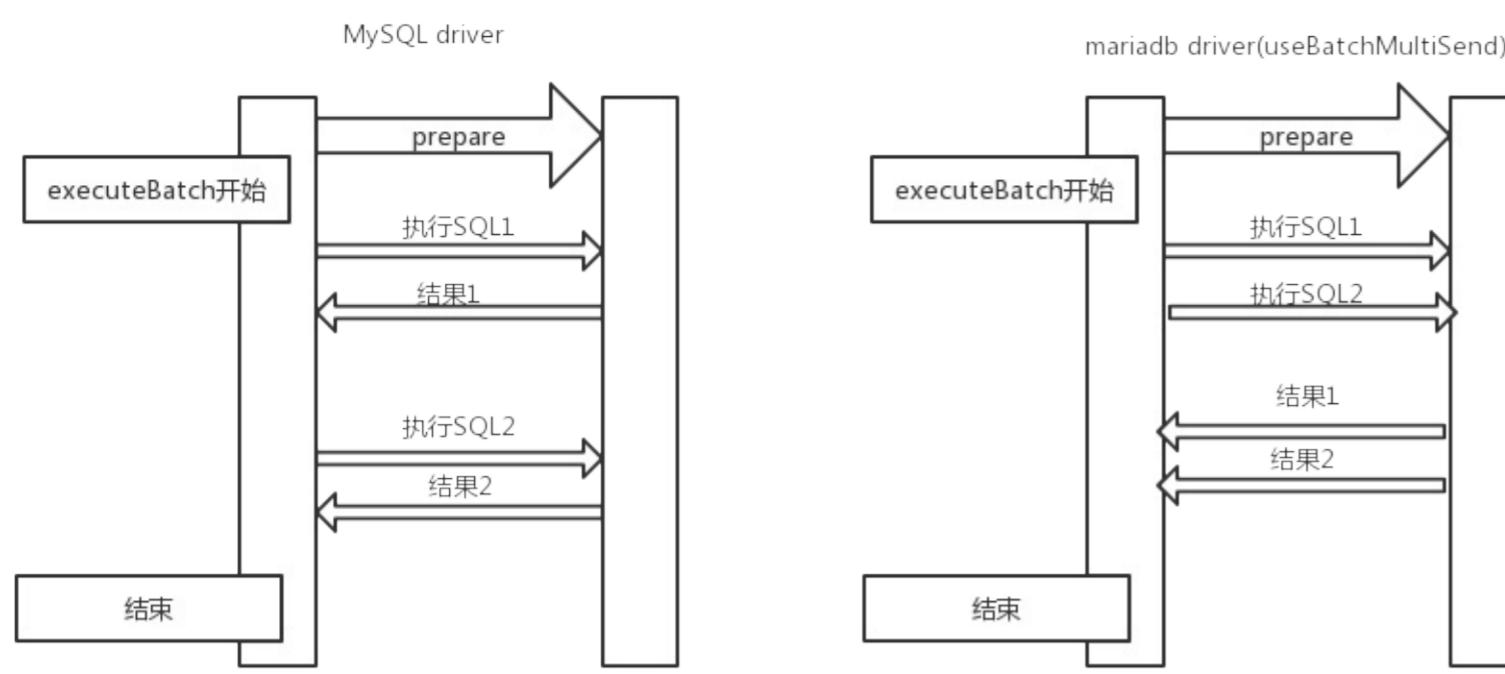


- urluseCursorFetch=true&server
- jdbcfetchSize>0&dbcfetchSize>0&fetchPrepareStatement

### 4.4.5 Dble Flow



mariadb-java-client 1.5.0 db executeBatch driver useBatchMultiSend  
mariadb-java-client 1.5.0 useBatchMultiSend executeBatch driver db



<https://mariadb.com/kb/en/library/option-batchmultisend-description/>

## 4.5

- EOF\_Packet
- ERR\_Packet
- OK\_Packet
- LOCAL\_INFILE Packet
- PACKET\_LOCAL\_INFILE
- PACKET\_RESULTSET



## 5.1 druid

1. INSERT ... VALUE ... INSERT ... VALUES ...VALUE[\$]

druidbugVALUES[\$]druid

druidissue2218

dbleissue [dble\\_issue\\_379](#).

2. SHOW ALL TABLES dble

3.

[dble\\_issue\\_788](#)

## 5.2

```

1. ,parentkey
   issue : https://github.com/actiontech/dble/issues/12

2. JDBCrewiteBatchedStatements=true
   :
   insert : insertinsert.. values().com_query dble

3. JDBCuseServerPrepStmts=true
   : dbleBinary Protocol Text Protocol

4. lock/unlock
   issue : https://github.com/actiontech/dble/issues/38

5. schema.xml3.20.07.0
   : schema.xml
   issue : https://github.com/actiontech/dble/issues/70

6. dbletabledble, dble

7. /
   : sql
   issue : https://github.com/actiontech/dble/issues/85

8.
   : dropjavatcp_keepalive
   issue : https://github.com/actiontech/dble/issues/87

9. /
   :
   issue : https://github.com/actiontech/dble/issues/100

10. order by lock in share mode/for update , lock clause is ignored
    :
    issue : https://github.com/actiontech/dble/issues/127

11. _charset_name 'string' _charset_name+b'val'
    issue : https://github.com/actiontech/dble/issues/262
    issue : https://github.com/actiontech/dble/issues/267

12. set sql_select_limit
    issue : https://github.com/actiontech/dble/issues/331

13. sEndDatedefault node
    :
    issue : https://github.com/actiontech/dble/issues/357

14. selece @@sql_mode IGNORE_SPACE
    : 4.2
    issue : https://github.com/actiontech/dble/issues/364

15. replace ... into
    replace replaceIDIDID

16. kill sessionok

17. 2.19.01rule/schema/server.xmlversionBUG2.19.01zkversion

18. mysql set global local_infile = 0 ,dble load data
    : dbleload data load data local infile ... mysql local_infile
    issue : https://github.com/actiontech/dble/issues/1111

19. set @@sql_auto_is_null=on; : set @@sql_auto_is_null=on null,dble
    issue : https://github.com/actiontech/dble/issues/978

20. mariadb-java-client 1.5.0useBatchMultiSend mariadb-java-client 1.5.0prepare issue https://github.com/actiontech/dble/issues/1244

21. explain . issue https://github.com/actiontech/dble/issues/1449

```

## 6. MySQL Server

[MySQL Serverbug](#)[MySQL](#)

- [6.1](#)
- [6.2 INSERT](#)
- [6.3 "show all tables"](#)
- [6.4 message](#)
- [6.5 information\\_schema](#)

## 6.1 MySQL

:

MySQL:

```
[test_yhq]>select * from char_columns_4;
+----+-----+
| id | c_char |
+----+-----+
| 1 | xx   |
| 4 | z    |
+----+-----+
2 rows in set (0.02 sec)
[test_yhq]>begin;
Query OK, 0 rows affected (0.01 sec)

[test_yhq]>insert into char_columns_4 values(1,'yy');
ERROR 1062 (23000): Duplicate entry '1' for key 'PRIMARY'
[test_yhq]>insert into char_columns_4 values(2,'yy');
Query OK, 1 row affected (0.00 sec)

[test_yhq]>commit;
Query OK, 0 rows affected (0.02 sec)
```

dble:

```
[testdb]>select * from sharding_four_node order by id;
+----+-----+
| id | c_flag | c_decimal |
+----+-----+
| 1 | 1_1   | 1.0000 |
| 2 | 2     | 2.0000 |
| 3 | 3     | 3.0000 |
+----+-----+
3 rows in set (0.28 sec)

begin;
Query OK, 0 rows affected (0.01 sec)

[testdb]>insert into sharding_four_node values(1,'1',1.0);
ERROR 1062 (23000): Duplicate entry '1' for key 'PRIMARY'
[testdb]>insert into sharding_four_node values(13,'13',13.0);
ERROR 1003 (HY000): Transaction error, need to rollback.Reason:[ errNo:1062 Duplicate entry '1' for key 'PRIMARY']
[testdb]>commit;
ERROR 1003 (HY000): Transaction error, need to rollback.Reason:[ errNo:1062 Duplicate entry '1' for key 'PRIMARY']
```

## 6.2 INSERTdb MySQL

:

MySQL:

```
desc mysql_autoinc;
+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra      |
+-----+-----+-----+-----+
| c_char | char(255) | YES  |     | NULL    | auto_increment |
| id    | bigint(20) | NO   | PRI | NULL    | auto_increment |
+-----+-----+-----+-----+
2 rows in set (0.02 sec)
```

```
[test_yhq]>insert into mysql_autoinc values('1',1);
Query OK, 1 row affected (0.01 sec)
```

dble

```
desc sharding.four_node_autoinc;
+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra      |
+-----+-----+-----+-----+
| c_char | char(255) | YES  |     | NULL    | auto_increment |
| id    | bigint(20) | NO   | PRI | NULL    | auto_increment |
+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

```
[testdb]>insert into sharding.four_node_autoinc values('2',2);
ERROR 1064 (HY000): In insert Syntax, you can't set value for Autoincrement column!
```

### 6.3 ADD "show all tables"

The optional ALL modifier causes SHOW TABLES to display a second output column with values of BASE TABLE for a table ,VIEW for a view, SHARDING TABLE for a sharding table and GLOBAL TABLE for a global table.

:

```
[testdb]>show all tables;
+-----+-----+
| Tables in testdb | Table_type |
+-----+-----+
| global_four_node | GLOBAL TABLE |
| global_four_node_autoinc | GLOBAL TABLE |
| global_two_node | GLOBAL TABLE |
| sbtest1 | SHARDING TABLE |
| sharding_four_node | SHARDING TABLE |
| sharding_four_node2 | SHARDING TABLE |
| sharding_four_node_autoinc | SHARDING TABLE |
| sharding_two_node | SHARDING TABLE |
| single | SHARDING TABLE |
| customer | BASE TABLE |
| district | BASE TABLE |
+-----+-----+
11 rows in set (0.02 sec)
```

## 6.4 message

:

MySQL:

```
mysql> insert into sharding_two_node values(9,'9',9.0),(10,'10',10.0);
Query OK, 2 rows affected (0.24 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

dble:

```
mysql> insert into sharding_two_node values(11,'11',11.0),(12,'12',12.0);
Query OK, 2 rows affected (0.49 sec)
```

## 6.5 information\_schema

Navicat Premium 12 dbleNavicat Premium 12 information\_schemamysql  
dbe driver

### Navicat Premium12

1. SELECT SCHEMA\_NAME, DEFAULT\_CHARACTER\_SET\_NAME, DEFAULT\_COLLATION\_NAME FROM information\_schema.SCHEMATA;

```
mysql schemecharacter set collationNavicat Premium 12
dblescheme SchemeConfig scheme scheme character set collation
1dbleschemeshardingNodeshardingNode character set collation character set collation ,MySQL
2dbe SCHEM ATA
```

### driver

0

```
1. SELECT TABLE_SCHEMA, TABLE_NAME, TABLE_TYPE
   FROM information_schema.TABLES WHERE TABLE_SCHEMA = 'testdb'
   ORDER BY TABLE_SCHEMA, TABLE_TYPE
2. SELECT TABLE_SCHEMA, TABLE_NAME, COLUMN_NAME, COLUMN_TYPE
   FROM information_schema.COLUMNS
  WHERE TABLE_SCHEMA = 'testdb'
  ORDER BY TABLE_SCHEMA, TABLE_NAME
3. SELECT DISTINCT ROUTINE_SCHEMA, ROUTINE_NAME, PARAMS.PARAMETER
   FROM information_schema.ROUTINES LEFT JOIN
  (SELECT SPECIFIC_SCHEMA, SPECIFIC_NAME,
  GROUP_CONCAT(CONCAT(DATA_TYPE, ' ', PARAMETER_NAME) ORDER BY ORDINAL_POSITION SEPARATOR ', ') PARAMETER, ROUTINE_TYPE
   FROM information_schema.PARAMETERS GROUP BY SPECIFIC_SCHEMA, SPECIFIC_NAME, ROUTINE_TYPE
  )PARAMS
  ON ROUTINES.ROUTINE_SCHEMA = PARAMS.SPECIFIC_SCHEMA AND
  ROUTINES.ROUTINE_NAME = PARAMS.SPECIFIC_NAME AND
  ROUTINES.ROUTINE_TYPE = PARAMS.ROUTINE_TYPE
  WHERE ROUTINE_SCHEMA = 'testdb' ORDER BY ROUTINE_SCHEMA
4. SELECT TABLE_NAME, CHECK_OPTION, IS_UPDATABLE, SECURITY_TYPE, DEFINER
   FROM information_schema.VIEWS
  WHERE TABLE_SCHEMA = 'testdb' ORDER BY TABLE_NAME ASC
5. SELECT * FROM information_schema.ROUTINES
  WHERE ROUTINE_SCHEMA = 'testdb' ORDER BY ROUTINE_NAME
6. SELECT EVENT_CATALOG, EVENT_SCHEMA, EVENT_NAME, DEFINER, TIME_ZONE,
  EVENT_DEFINITION, EVENT_BODY, EVENT_TYPE, SQL_MODE, STATUS, EXECUTE_AT,
  INTERVAL_VALUE, INTERVAL_FIELD, STARTS, ENDS, ON_COMPLETION, CREATED,
  LAST_ALTERED, LAST_EXECUTED, ORIGINATOR, CHARACTER_SET_CLIENT,
  COLLATION_CONNECTION, DATABASE_COLLATION, EVENT_COMMENT
   FROM information_schema.EVENTS WHERE EVENT_SCHEMA = 'testdb'
   ORDER BY EVENT_NAME ASC
7. SELECT COUNT(*) FROM information_schema.TABLES
  WHERE TABLE_SCHEMA = 'testdb' UNION
SELECT COUNT(*)
FROM information_schema.COLUMNS
  WHERE TABLE_SCHEMA = 'testdb' UNION
SELECT COUNT(*) FROM information_schema.ROUTINES WHERE ROUTINE_SCHEMA = 'testdb'
```



## 7.1 SQL

- SQLSQLSQL

1

```
dbledbleSQLSQL SQLdbeSQLSQLSQL dbleMySQL dbleEXPLAINdble  
explain select id,accountno from account where userid=2;
```

EXPLAINSQL EXPLAIN2

```
explain2 shardingNode=dn1 sql=select id,accountno from account where userid=2;
```

explain2sqlexplainshardingNodeexplain

## 2SQL

SQL:

- SQL
- 
- INININ
- SQLDISTINCTGROUP BYORDER BYSQL
- 

3

- Join
- 
- 

- 
- JoinJoinJoin
- Join
- JoinJoin
- limit a,b
- GROUP
- 
- Join

## 7.2 dbleDemo

### ibatis

ibatisdbleMySQL JDBC

```
jdbc.driverClass=com.mysql.jdbc.Driver
jdbc.jdbcUrl=jdbc:mysql://127.0.0.1:8066/TESTDB?useUnicode=true&characterEncoding=utf-8
jdbc.user=root
jdbc.password=123456
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="com.mapper.UserMapper">
<insert id="saveUser" parameterType="com.bean.User">
    insert into user(id,name,phone,birthday)
    values (0,#{name},#{phone},#{birthday})
    <selectKey keyProperty="id" order="after" resultType="int">
        select last_insert_id() as id
    </selectKey>
</insert>
<delete id="deleteUserById" parameterType="java.lang.String">
    delete from user where id=#{id}
</delete>
<update id="updateUser" parameterType="com.bean.User">
    update user set name=#{name},phone=#{phone},birthday=#{birthday} where id=#{id}
</update>
<update id="updateUsers">
    /*!dble:sql=select * from user;*/update users set usercount=(select count(*) from user),ts=now()
</update>
<select id="getUserById" parameterType="java.lang.String" resultType="com.bean.User">
    select * from user where id=#{id}
</select>
<select id="getUsers" resultType="com.bean.User">
    select * from user
</select>
```

select last\_insert\_id() as idID updateUsers dbleibatis##

### hibernate

hibernatedbleMySQL hibernate.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
"-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
<session-factory>
<property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
<property name="hibernate.connection.url">jdbc:mysql://192.168.58.51:8066/testdb?useUnicode=true&characterEncoding=utf-8</property>
<property name="hibernate.connection.username">root</property>
<property name="hibernate.connection.password">123456</property>
<property name="hibernate.dialect">org.hibernate.dialect.MySQLInnoDBDialect</property>
<property name="hibernate.format_sql">true</property>
<property name="hibernate.hbm2ddl.auto">update</property>
<mapping resource="com/actiontech/test/News.hbm.xml"/>
</session-factory>
</hibernate-configuration>
```

News.hbm.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
"-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
<class name="com.actiontech.test.News" table="news_table">
    <id name="id" type="java.lang.Integer">
        <column name="id" />
    </id>
    <property name="title" type="java.lang.String">
        <column name="title" />
    </property>
    <property name="content" type="java.lang.String">
        <column name="content" />
    </property>
</class>
</hibernate-mapping>
```

News.java

```
package com.actiontech.test;
public class News {
    private Integer id;
    private String title;
    private String content;
    public Integer getId() {
        return id;
    }
    public void setId(Integer id) {
        this.id = id;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
}
```

public String getContent() { return content; } public void setContent(String content) { this.content = content; } &lt;/pre&gt; NewsManager.java

```
package com.actiontech.test;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
```

```

import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;
public class NewsManager {
    public static void main(String[] args)
        throws Exception {
        Configuration config = new Configuration().configure();
        SessionFactory factory = config.buildSessionFactory();
        Session session = factory.openSession();
        Transaction transaction = session.beginTransaction();
        News news = new News();
        news.setId(10);
        news.setTitle("dble");
        news.setContent("Hibernate dble");
        session.save(news);
        transaction.commit();
        session.close();
        factory.close();
    }
}

```

dbleHibernateHibernateSQLSQL

**JDBC**

JDBCdbleMySQL

```

package com.actiontech.test;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Properties;
import java.util.concurrent.CountDownLatch;
import java.util.concurrent.atomic.AtomicLong;
public class SingleMixEngine {
    public static void main(String[] args) throws Exception {
        Class.forName("com.mysql.jdbc.Driver");
        Properties props = new Properties();
        props.setProperty("user", "root");
        props.setProperty("password", "123456");
        SingleMixEngine engine = new SingleMixEngine();
        engine.execute(props, "jdbc:mysql://192.168.58.51:8066/testdb");
    }
    final AtomicLong tmAL = new AtomicLong();
    final String tableName="news_table";
    public void execute(Properties props, String url) {
        CountDownLatch cd1 = new CountDownLatch(1);
        long start = System.currentTimeMillis();
        for (int i = 0; i < 1; i++) {
            TestThread insertThread = new TestThread(props,cd1, url);
            Thread t = new Thread(insertThread);
            t.start();
            System.out.println("Test start");
        }
        try {
            cd1.await();
            long end = System.currentTimeMillis();
            System.out.println("Test end, total cost:" + (end-start) + "ms");
        } catch (Exception e) {
        }
    }
    class TestThread implements Runnable {
        Properties props;
        private CountDownLatch countDownLatch;
        String url;
        public TestThread(Properties props,CountDownLatch cd1, String url) {
            this.props = props;
            this.countDownLatch = cd1;
            this.url = url;
        }
        public void run() {
            Connection connection = null;
            PreparedStatement ps = null;
            Statement st = null;
            try {
                connection = DriverManager.getConnection(url,props);
                connection.setAutoCommit(true);
                st = connection.createStatement();
                String dropSql = "drop table if exists " + tableName;
                System.out.println("Execute SQL:\n\t"+dropSql);
                st.execute(dropSql);

                String createSql = "create table " + tableName + "(id int,title varchar(20),content varchar(50))";
                System.out.println("Execute SQL:\n\t"+createSql);
                st.execute(createSql);

                String insertSql = "insert into " + tableName + " (id,title,content) values(?, ?, ?)";
                System.out.println("Prepared SQL:\n\t"+insertSql);
                ps = connection.prepareStatement(insertSql);
                for (int i = 1; i <= 3; i++) {
                    ps.setInt(1,i);
                    ps.setString(2, ""+i);
                    ps.setString(3, ""+i+"");
                    ps.execute();
                    System.out.println("Insert data:\t"+i+"\t"+i+"\t"+i);
                }

                String querySQL = "select * from " + tableName + " order by id";
                System.out.println("Execute SQL:\n\t"+querySQL);
                ResultSet rs = st.executeQuery(querySQL);
                int colcount = rs.getMetaData().getColumnCount();
                System.out.println("Current Data:");
                while(rs.next()){
                    for(int i=1;i<=colcount;i++){
                        System.out.print("\t"+rs.getString(i));
                    }
                    System.out.println();
                }

                String updateSql = "update " + tableName + " set title='test1' where id=1";
                System.out.println("Execute SQL:\n\t"+updateSql);
                st.execute(updateSql);
                rs = st.executeQuery(querySQL);
                System.out.println("Current Data:");
                while(rs.next()){
                    for(int i=1;i<=colcount;i++){
                        System.out.print("\t"+rs.getString(i));
                    }
                    System.out.println();
                }

                String deleteSql = "delete from " + tableName + " where id=2";
                System.out.println("Execute SQL:\n\t"+deleteSql);
                st.execute(deleteSql);
                rs = st.executeQuery(querySQL);
                System.out.println("Current Data:");
                while(rs.next()){
                    for(int i=1;i<=colcount;i++){
                        System.out.print("\t"+rs.getString(i));
                    }
                    System.out.println();
                }

                String createIndexSql = "create index idx_1 on " + tableName + "(title)";
                System.out.println("Execute SQL:\n\t"+createIndexSql);
                st.execute(createIndexSql);

                String dropIndexSql = "drop index idx_1 on " + tableName;
                System.out.println("Execute SQL:\n\t"+dropIndexSql);
                st.execute(dropIndexSql);
            } catch (Exception e) {
                System.out.println(new java.util.Date().toString());
                e.printStackTrace();
            }
        }
    }
}

```

```
    } finally {
        if (ps != null)
            try {
                ps.close();
            } catch (SQLException e1) {
                e1.printStackTrace();
            }
        if (connection != null)
            try {
                connection.close();
            } catch (SQLException e1) {
                e1.printStackTrace();
            }
        this.countDownLatch.countDown();
    }
}
```

### 7.3

- 
- MySQL
- 
- DDLDDL

**8**

- [8.1](#)
- [8.2 MySQL-offset-step](#)

**8.1**

tbidid

**1cluster.conf**

```
sequenceHandlerType=2
sequenceStartTime=2010-10-01 09:42:54
...
```

**2bootstrap.conf**

```
instanceId=1
...

```

**3user.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE dble:user SYSTEM "user.dtd">
<dbl: user xmlns:dbl="http://dble.cloud/">
  <managerUser name="test" password="test"/>
  <shardingUser name="abc" password="abc" schemas="myschema" maxCon="1000000">
    </shardingUser>
</dbl: user>
```

**4db.xml**

```
<?xml version="1.0"?>
<!DOCTYPE dbl:db SYSTEM "db.dtd">
<dbl: db xmlns:dbl="http://dble.cloud/">
  <dbGroup name="host_1" rwSplitMode="0" delayThreshold="100">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM1" url="172.100.10.101:3306" user="test1" password="test1" maxCon="1000" minCon="1000" primary="true" />
  </dbGroup>
  <dbGroup name="host_2" rwSplitMode="0" delayThreshold="100">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM2" url="172.100.10.102:3306" user="test1" password="test1" maxCon="1000" minCon="1000" primary="true" />
  </dbGroup>
</dbl: db>
```

**5sharding.xml**

```
<?xml version="1.0"?>
<!DOCTYPE dbl:sharding SYSTEM "sharding.dtd">
<dbl: sharding xmlns:dbl="http://dble.cloud/">
  <schema name="myschema" shardingNode="dn1">
    <shardingTable name="sbtest1" shardingNode="dn1,dn2" function="mod" shardingColumn="id" incrementColumn="id" />
  </schema>
  <shardingNode name="dn1" dbGroup="host_1" database="dble"/>
  <shardingNode name="dn2" dbGroup="host_2" database="dble"/>
  <function name="mod" class="Hash">
    <property name="partitionCount">2</property>
    <property name="partitionLength">1</property>
  </function>
</dbl: sharding>
```

**6**

```
``mysql mysql -utest -p111111 -h127.0.0.1 -P8066 -Dmyschema mysql> drop table if exists sbtest1; Query OK, 0 rows affected (0.05 sec) mysql> create table sbtest1(id bigint(20), k int unsigned not null default '0', primary key(id)); Query OK, 0 rows affected (0.05 sec)
mysql> insert into sbtest1 values(2); Query OK, 1 row affected (0.11 sec)
mysql> select * from sbtest1;
bigint
```

## 8.2 MySQL-offset-step

sbtest1idMySQL-offset-stepid

### 1cluster.conf

```
sequenceHandlerType=1
...

```

### 2user.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE dble:user SYSTEM "user.dtd">
<dble:user xmlns:dble="http://dble.cloud/" >
  <managerUser name="test" password="test"/>
  <shardingUser name="abc" password="abc" schemas="myschema" maxCon="1000000">
    </shardingUser>
</dble:user>
```

### 3db.xml

```
<?xml version="1.0"?>
<!DOCTYPE dble:db SYSTEM "db.dtd">
<dble:db xmlns:dble="http://dble.cloud/" >
  <dbGroup name="host_1" rwSplitMode="0" delayThreshold="100">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM1" url="172.100.10.101:3306" user="test1" password="test1" maxCon="1000" minCon="1000" primary="true" />
  </dbGroup>
  <dbGroup name="host_2" rwSplitMode="0" delayThreshold="100">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM2" url="172.100.10.102:3306" user="test1" password="test1" maxCon="1000" minCon="1000" primary="true" />
  </dbGroup>
</dble:db>
```

### 4sharding.xml

```
<?xml version="1.0"?>
<!DOCTYPE dble:sharding SYSTEM "sharding.dtd">
<dble:sharding xmlns:dble="http://dble.cloud/" >
  <schema name="myschema" shardingNode="dn1">
    <shardingTable name="sbtest1" shardingNode="dn1" database="dble" function="mod" shardingColumn="id" incrementColumn="id" />
  </schema>
  <shardingNode name="dn1" dbGroup="host_1" database="dble"/>
  <shardingNode name="dn2" dbGroup="host_2" database="dble"/>

  <function name="mod" class="Hash">
    <property name="partitionCount">2</property>
    <property name="partitionLength">1</property>
  </function>
</dble:sharding>
```

### 5sequence\_db\_conf.properties

```
#sequence stored in shardingNode
'myschema'.`sbtest1`=dn1
```

myschema,sbtest1,dn1sharding.xml

dn1host\_1/dbledbleconf/dbseq.sql()

```
mysql -h172.100.10.101 -utest1 -ptest1 -Ddble
mysql>source conf/dbseq.sql
```

### sqlDBLE\_SEQUENCE

```
mysql -h172.100.10.101 -utest1 -ptest1 -Ddble
mysql>INSERT INTO DBLE_SEQUENCE VALUES ('`mytest`.`sbtest1`', 16, 1);
```

### DBLE\_SEQUENCE

- namesequence\_db\_conf.properties
- current\_value
- increment1

### 6

dble

```
mysql -utest -p111111 -h127.0.0.1 -P8066 -Dmyschema
mysql> drop table if exists sbtest1;
Query OK, 0 rows affected (0.05 sec)
mysql> create table sbtest1(id int, k int unsigned not null default '0', primary key(id));
Query OK, 0 rows affected (0.05 sec)

mysql> insert into sbtest1 values(2);
Query OK, 1 row affected (0.11 sec)

mysql> select * from sbtest1;
+----+---+
| id | k |
+----+---+
| 17 | 2 |
+----+---+
1 row in set (0.01 sec)
```

sqlDBLE\_SEQUENCEcurrent\_value16insert17

- sequence\_db\_conf.properties

```
'myschema'.`sbtest1`=dn1
```

- sequence\_db\_conf.propertiesdn1dbseq.sql

## 9 sysbenchdble

- [9.1](#)
- [9.2 dble](#)
- [9.3 sysbench](#)

## 9.1

- Sysbench version: 1.0
- Dble version: 5.6.23-dble-2.19.11.0-2d7c4911b7a4fecaa9eb0299f49c32ec11e97c42-20200228124218
- MySQL version: 5.7.25

- sysbench172.20.134.1
- dble172.20.134.2
- 3mysql172.20.134.3172.20.134.4172.20.134.5

## 9.2 dble

2.18

### 1. bootstrap.cnf

```
-Dprocessors=10 -DbbackendProcessors=10 -DprocessorExecutor=8 -DbbackendProcessorExecutor=6 -DsqlExecuteTimeout=3000000
```

### 2.user.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE dble:User SYSTEM "user.dtd">
<db:User xmlns:db="http://dble.cloud/" version="4.0">
  <managerUser name="root" password="111111" />
  <shardingUser name="test" password="111111" schemas="sbtest" maxCon="1000000">
    </shardingUser>
</db:User>
```

### 3.db.xml

```
<?xml version="1.0"?>
<!DOCTYPE db:db SYSTEM "db.dtd">
<db:db xmlns:db="http://dble.cloud/" version="4.0">
  <dbGroup name="host_1" rwSplitMode="0" delayThreshold="-1">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM1" url="172.20.134.3:3306" uuser="test1" password="test1" maxCon="1000" minCon="100" primary="true"/>
  </dbGroup>
  <dbGroup name="host_2" rwSplitMode="0" delayThreshold="-1">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM1" url="172.20.134.4:3306" uuser="test1" password="test1" maxCon="1000" minCon="100" primary="true"/>
  </dbGroup>
  <dbGroup name="host_3" rwSplitMode="0" delayThreshold="-1">
    <heartbeat>select USER()</heartbeat>
    <dbInstance name="hostM1" url="172.20.134.5:3306" uuser="test1" password="test1" maxCon="1000" minCon="100" primary="true"/>
  </dbGroup>
</db:db>
```

### 4.sharding.xml

```
<?xml version="1.0"?>
<!DOCTYPE dble:sharding SYSTEM "sharding.dtd">
<db:sharding xmlns:db="http://dble.cloud/" version="4.0">
  <schema name="sbtest">
    <shardingTable name="sbtest1" shardingNode="dn$1-9" rule="hash-sysbench" shardingColumn="id" />
  </schema>
  <shardingNode name="dn$1-3" dbGroup="host_1" database="dbledb$1-3" />
  <shardingNode name="dn$4-6" dbGroup="host_2" database="dbledb$4-6" />
  <shardingNode name="dn$7-9" dbGroup="host_3" database="dbledb$7-9" />
  <function name="hash-sysbench" class="Hash">
    <property name="partitionCount">9</property>
    <property name="partitionLength">1</property>
  </function>
</db:sharding>
```

### 5.mysql

- 172.20.134.3:3306 dbledb1, dbledb2, dbledb3
- 172.20.134.4:3306 dbledb4, dbledb5, dbledb6
- 172.20.134.5:3306 dbledb7, dbledb8, dbledb9

### 9.3 sysbench

how about a [sysbench-testing-quick-start](#); issue

sysbench

```
FATAL: mysql_drv_query() returned error 1062 (Duplicate entry '49823' for key 'PRIMARY') for query 'INSERT INTO sbtest1 (id, k, c, pad) VALUES (49823, 58210, '27111667985-11552069038-79242882109-05602914209-02374993639-32242662584-65155028223-08319627673-44873060047-22215118936', '07405724915-3279961660-96659146042-59717172693-66753749407')'
```

```
/usr/share/sysbench/oltp_read_write.lua --mysql-db=sbtest --mysql-host=172.20.134.2 --mysql-port=8066 --mysql-user=test1 --mysql-password=test1 --auto_inc=off --tables=1 --table-size=100000 --threads=4 --time=30 --report-interval=1 --max-requests=0 --percentile=95 --db-ps-mode=disable -skip-trx=on cleanup
```

```
/usr/share/sysbench/oltp_read_write.lua --mysql-db=sbtest --mysql-host=172.20.134.2 --mysql-port=8066 --mysql-user=test1 --mysql-password=test1 --auto_inc=off --tables=1 --table-size=100000 --threads=4 --time=30 --report-interval=1 --max-requests=0 --percentile=95 --db-ps-mode=disable -skip-trx=on prepare
```

```
/usr/share/sysbench/oltp_read_write.lua --mysql-db=sbtest --mysql-host=172.20.134.2 --mysql-port=8066 --mysql-user=test1 --mysql-password=test1 --auto_inc=off --tables=1 --table-size=100000 --threads=4 --time=30 --report-interval=1 --max-requests=0 --percentile=95 --db-ps-mode=disable -skip-trx=on run
```

- [A.1 ErrorCode](#)
- [A.2](#)
- [A.3](#)

- Max Connections
- Out Of Memory Error
- The Problem Of Hint
- NestLoop Parameters Lead To Temptable Exception
- Can't Get Variables From ShardingNode
- Port Already In Use 1984
- Sharding Column Cannot Be Null

## dble-MaxConnections

### Issue

- [Err] 3009 - java.io.IOException: the max activeConnections size can not be max than maxconnections.

### Resolution

1. dble
2. maxConmaxCon

		/	
maxCon		1024	,maxcon maxConmanager

### Root Cause

### Relevant Content

1. maxWait

maxActive

2. connectionProperties

connectTimeout socketTimeout connectTimeout TCP socketTimeout

socket socket DB

3. maxActive

maxWait 0

## dble-OutOfMemoryError

### Setting

load data17G4K  
load data

### Issue

- INFO | jvm 1 | 2019/06/28 14:55:37 | Exception in thread "backendBusinessExecutor17" java.lang.OutOfMemoryError: GC overhead limit exceeded.

### Resolution

- maxRowSizeToFile

maxCharsPerColumn		65535	
maxRowSizeToFile		10000	load dataOOM

- wrapper.confXmx

    ||| OOM

### Root Cause

- maxRowSizeToFile
- Xmx

    ||| On-Heap JVM Xms ,Xmx jvm

### Relevant Content

load data

load dataOOM

JVM

- dble=0.6 (.)
- Xmx = 0.4 dble
- MaxDirectMemorySize = 0.6 \* dble

- JVM-Xms1/64
- JVM-Xmx1/4
- 40%JVM-Xmx70%JVM-Xms

## dble-TheProblemOfHint

### Setting

- mysql> /\*dble:sql=select 1 from rp\_cre\_data\_mobile\_track\_cmcc / call update\_track();
- sql
- 

### Issue

- mysql> show @@processlist;

Front_Id	shardingNode	BconnID	user	Front_Host	db	Command	Time	State	Info
33	NULL	NULL	root		NULL	NULL	0	updating	NULL
34	NULL	NULL	root		NULL	NULL	0	updating	NULL
35	NULL	NULL	root		NULL	NULL	0	updating	NULL
41	dn9	9372	root		db9	Query	0	updating	NULL
42	NULL	NULL	root		NULL	NULL	0	updating	NULL
43	NULL	NULL	root		NULL	NULL	0	updating	NULL
30	NULL	NULL	admin		NULL	NULL	0	updating	NULL

### Resolution

1. /\*dble:type=..../
  2. mysql client -c
- ```
mysql --help
-c, --comments
Preserve comments. Send comments to the server. The default is --skip-comments (discard comments), enable with --comments.
```

### Root Cause

1. /\*dble:type=..../ mysql
2. -c skip

### Relevant Content

#### hint

- 1.
2. dbleinsert...select...

#### Hint

##### Hint

1. /\*!dble:type=....\*/
2. /\*#dble:type=....\*/
3. /\* \*/()

type4shardingNodedb\_typesql\_db\_instance\_url  
type [https://actiontech.github.io/dble-docs-cn/2.Function/2.04\\_hint.html](https://actiontech.github.io/dble-docs-cn/2.Function/2.04_hint.html)

#### Hint

- selectSQLdelete/update/delete/update/insert SQL
- SQL
- hintDDLreload @@metadata
- hintsession
- dbleMySQL, MySQL#1169
- SQLSQL select id from tab\_a where id='10000'
-

## dble-NestLoop Parameters Lead To Temptable Exception

### Setting

- NestLoopNestLoop4
- joinstudentclass
- studentidnameclass\_nameid
- classidclass\_nameteacher\_name
- SELECT class.teacher\_name FROM student LEFT JOIN class on student.class\_name=class.class\_name WHERE student.name=""

### Issue

- com.actiontech.dble.plan.common.exception.TempTableException: temptable too much rows,[rows size is 5].

### Resolution

- NestLoop

|                  |          |      |              |
|------------------|----------|------|--------------|
|                  |          | /    |              |
| useJoinStrategy  | nestLoop |      | joinwhereSQL |
| nestLoopConnSize |          | 4    |              |
| nestLoopRowsSize |          | 2000 |              |

- NestLoop
  - <property name="useJoinStrategy">false</property>
- 

### Root Cause

- NestLoopNestLoopNestLoop
- NestLoop

- joinNestLoop join
  - wherewhere
  - wherewhereNestLoop
  - NestLoop
- SQL
    - SELECT class.teacher\_name FROM student LEFT JOIN class on student.class\_name=class.class\_name WHERE student.name=""

dbleNestLoop

1. SQLstudentclass
2. wherestudentNestLoopstudent
3. SQLstudentNestLoop

### Relevant Content

#### MySQLNestLoop

1. NestLoop:

- Nested Loop
- Nested LoopJoinJoin Nested Loop
- Nested Loop“”——

2. NestLoop

|          |  |         |  |  |
|----------|--|---------|--|--|
|          |  |         |  |  |
| NestLoop |  | CPU/I/O |  |  |

## dble-Can't get variables from shardingNode

### Setting

- db.xml

```
<dbGroup name="localhost1" rwSplitMode="0" delayThreshold="100">
<heartbeat>show slave status</heartbeat>
<dbInstance host="hostM1" url="localhost:3306" user="root" password="nE7jA%5m" maxCon="1000" minCon="10" primary="true" > </dbInstance>
</dbGroup>
<dbGroup name="localhost2" rwSplitMode="0" delayThreshold="100">
<heartbeat>show slave status</heartbeat>
<dbInstance host="hostM2" url="localhost:3306" user="root" password="nE7jA%5m" maxCon="1000" minCon="10" primary="true"> </dbInstance> </dbGroup>
```

### Issue

- dble

```
Running dble-server...
wrapper | --> Wrapper Started as Console
wrapper | Launching a JVM...
jvm 1 | Wrapper (Version 3.2.3)
http://wrapper.tanukisoftware.org
jvm 1 | Copyright 1999-2006 Tanuki Software, Inc. All Rights Reserved.
jvm 1 |
jvm 1 | java.io.IOException:Can't get variables from shardingNode ...
wrapper | <-- Wrapper Stopped
```

### Resolution

1. mysqlmysql5.1mysql,
2. db.xmlrootmysql
3. root
4. mysqlshow variables
5. dbGroupdble

### Root Cause

- db.xml mysqlshow variables
- mysql 5.7

### Relevant Content

1. mysql5.7
2. mysqlmysqlMysql
3.
  - o mysql> show databases;
  - o ERROR 1820 (HY000): You must reset your password using ALTER USER statement before executing this statement.
  - o mysql > set password = password('xxxxxx');

## dble-Port already in use:1984

### Issue

- wrapper.log>Error1

```
STATUS | wrapper | 2019/07/23 16:37:06 | --> Wrapper Started as Daemon
STATUS | wrapper | 2019/07/23 16:37:06 | Launching a JVM...
INFO | jvm 1 | 2019/07/23 16:37:06 | OpenJDK 64-Bit Server VM warning: ignoring option MaxPermSize=64M; support was removed in 8.0
INFO | jvm 1 | 2019/07/23 16:37:07 | : java.rmi.server.ExportException: Port already in use: 1984; nested exception is:
INFO | jvm 1 | 2019/07/23 16:37:07 | java.net.BindException: Address already in use (Bind failed)
INFO | jvm 1 | 2019/07/23 16:37:07 | sun.management.AgentConfigurationError: java.rmi.server.ExportException: Port already in use: 1984;
```

- wrapper.log>Error2

```
STATUS | wrapper | 2019/07/26 16:12:48 | --> Wrapper Started as Daemon
STATUS | wrapper | 2019/07/26 16:12:49 | Launching a JVM...
INFO | jvm 1 | 2019/07/26 16:12:49 | Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
INFO | jvm 1 | 2019/07/26 16:12:49 | Copyright 1999-2006 Tanuki Software, Inc. All Rights Reserved.
INFO | jvm 1 | 2019/07/26 16:12:49 |
INFO | jvm 1 | 2019/07/26 16:12:51 | java.net.BindException: Address already in use
INFO | jvm 1 | 2019/07/26 16:12:51 | at sun.nio.ch.Net.bind0(Native Method)
INFO | jvm 1 | 2019/07/26 16:12:51 | at sun.nio.ch.Net.bind(Net.java:433)
INFO | jvm 1 | 2019/07/26 16:12:51 | at sun.nio.ch.Net.bind(Net.java:425)
INFO | jvm 1 | 2019/07/26 16:12:51 | at sun.nio.ch.ServerSocketChannelImpl.bind(ServerSocketChannelImpl.java:223)
INFO | jvm 1 | 2019/07/26 16:12:51 | at com.actiontech.dble.net.NIOAcceptor.<init>(NIOAcceptor.java:46)
```

### Resolution

- Error1  
wrapper.conf1984  
-Dcom.sun.management.jmxremote.port=1984
- Error2  
netstat -nap pid80669066  
kill -9 pid
- dble

### Root Cause

- jmxjavadble
- dblejvmjmx
- jmxconsolejvmjvm

### Relevant Content

1. JVM  
JVM JavaJava  
JavaJavaC++
2. JMX  
JMX Java Management ExtensionsJava  
JPAMJS
3. Jconsole  
JconsoleJDKJVM.java

## double-Sharding column can't be null

### Setting

- sharding.xml

```
<sharingTable shadingColumn="number" ... >
...
<function name="rangeLong" class="NumberRange">
<property name="mapFile">partition.txt</property>
<property name="defaultNode">0</property>
</function>
```

- create table account (id int(10),number int(10) not null,name varchar(20) not null);
- insert into account (id,number,name) values (1,NULL,'aaa');

### Issue

ERROR 1064 (HY000): Sharding column can't be null when the table in MySQL column is not null

### Resolution

- numbernameNULL
- numberNULL;
- ALTER TABLE account MODIFY number VARCHAR (20);
- blacklistalterTableAllow;

```
<blacklist name="bk1">
<property name="alterTableAllow">true</property>
</blacklist>
```

sharding-by-rangedbleERalter

```
<sharingTable shadingColumn="id" ... >
```

- alter

### Root Cause

- MySQLInsert
 

ERROR 1048 (23000): Column 'number' cannot be null
- descnumbername

Field	Type	Null	Key	Default	Extra
id	int(10)	YES		NULL	
number	int(10)	NO		NULL	
name	varchar(20)	NO		NULL	

- 
- How To Use Explain To Resolve The Distribution Rules Of Group Gy
  - Hash And ConsistentHashing And Jumpstringhash

## dble-How To Use Explain To Resolve The Distribution Rules Of Group Gy

### Questions

group by

### Conclusions

- 1. explainsql
  2. dble explainmycatSQL

### For Example

1.

sharding.xml

```
<shardingTable name="eee" shardingNode="dn1,dn2" function="hashLong" shardingColumn="id"/>
...
<function name="hashLong" class="Hash">
  <property name="partitionCount">2</property>
  <property name="partitionLength">128</property>
</function>
```

1. dble client eee mysql> select \* from eee;

| id | name | -- | -- | 1 | 2 | 3 | 4 | 5 | 130 | 131 | 132 | 133 | 134 |

mysql> select name,count(name) from eee group by name;

name	COUNT(name)
	4
	4
	1
	1

1. explainsql

SHARDING_NODE	TYPE	SQL/REF
dn1_0	BASE SQL	select eee . name ,COUNT(name) as _\$COUNT\$_rpda_0 from eee GROUP BY eee . name ASC
dn2_0	BASE SQL	select eee . name ,COUNT(name) as _\$COUNT\$_rpda_0 from eee GROUP BY eee . name ASC
merge_1	MERGE	dn1_0; dn2_0
aggregate_1	AGGREGATE	merge_1
shuffle_field_1	SHUFFLE_FIELD	aggregate_1

### Instructions

explain

1. dblesqlshardingnode
2. shardingnodemerge
3. mergegroup by
4. SHUFFLE\_FIELD
  - SHUFFLE\_FIELD

### Relevant Content

dble

1. dbleapp mysql
- 2.

- /
- dblejoingroup by
- <https://opensource.actionsky.com/dble-lesson-one/>

dble

1. IO MySQLLsql
2. MySQL
3. MySQL

PSdble

## dble-Hash And ConsistentHashing And Jumpstringhash

### Questions

- dblehashhash
- hash

### Conclusions

- dblehashhash
- hashhash

### Instructions

#### dble-hash

- 
- sharding.xml

```
<function name="hashLong" class="hash">
<property name="partitionCount">1,2</property>
<property name="partitionLength">10,20</property>
</function>
```

- 

```
<function name="hashLong" class="hash">
<property name="partitionCount">4</property>
<property name="partitionLength">10</property>
</function>
```

partitionCount  
partitionLength  
MC1L1 + ... + CnLn

dble-hash

Count=2,Length=2 -> [0,2](2,4) -> =4  
key1,2,3,4,5,6,7,8

node1	node2
1,4,5,8	2,3,6,7

Count=3,Length=2 -> [0,2](2,4)(4,6) -> =6 key1,2,3,4,5,6,7,8

node1	node2	node3
1,6,7	2,3,8	4,5

count

### summary

node

### Consistent Hashing

NodeANode BNode CNode DABCDANode ABNode BCNode CDNode D:

A —> NodeA  
B —> NodeB  
C —> NodeC  
D —> NodeD

Node CABDCNode D Node X:

A —> NodeA  
B —> NodeB  
—> NodeX  
C —> NodeC  
D —> NodeD

ABDCNode X

Hash node

“”

NodeANode B “Node A#1”“Node A#2”“Node A#3”“Node B#1”“Node B#2”“Node B#3”  
“Node A#1”“Node A#2”“Node A#3”Node A,“Node B#1”“Node B#2”“Node B#3”Node B

Node A#1 —> NodeA  
Node A#2 —> NodeA  
Node A#3 —> NodeA  
Node B#1 —> NodeB  
Node B#2 —> NodeB  
Node B#3 —> NodeB

“”

### summary

hashnode

### Jumpstringhash

- 
- sharding.xml

```
<function name="jumpHash" class="jumpStringHash">
<property name="partitionCount">2</property>
<property name="hashSlice">0:2</property>
</function>
```

partitionCount hashSlice

GoogleA Fast, Minimal Memory, Consistent Hash Algorithmhash1/n

- 0,1,2,3
- $(0,1,1)/(n+1), 1/2, (2,1)/(n+1), 1/3, (3,1)/(n+1), 1/4$
- $n/(n+1)$  1/(n+1) n+1
- indexmaxa0,1,2,33

0,1,2,30,1,2,3,4

## summary

### Jump stringhashhash

- Jump stringhashhash
- Jump stringhashhash
- Jump stringhashhashble-hash
- dble-hashJump stringhashhash

- 
- ToBeContinued2

