

更多资源v: 2938843375

online DDL 问题

更多资源v: 2938843375

目录

1 自制 online DDL - 触发器

2 自制 online DDL- binlog

基础知识回顾：replace into

```
CREATE TABLE `t` (  
  `id` int(11) NOT NULL,  
  `c` int(11) DEFAULT NULL,  
  `d` int(11) DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  unique KEY `c` (`c`)  
) ENGINE=InnoDB;
```

```
insert into t values(0,0,0),(5,5,5),  
(10,10,10),(15,15,15),(20,20,20),(25,25,25);
```

课堂练习：

replace into t values(10,15,11);
结果是什么？

```
root@devops_db 17:56: [test]> replace into t values(10,15,11);  
Query OK, 3 rows affected (0.02 sec)
```

基础知识回顾：trigger

--创建 insert 触发器

```
CREATE TRIGGER `ex_tri` AFTER INSERT ON `test`.`t`  
FOR EACH ROW  
  REPLACE INTO `test`.`t1` (id,c,d)  
  VALUES (NEW.`id`, NEW.c,NEW.d);
```

课堂练习：

```
begin  
replace into t values(7,7,7);  
t1 的结果是什么？
```

课堂练习：

```
replace into t values(10,15,11);  
t1 的结果是什么？
```

基础知识回顾：insert ... select

insert into t1 seelct * from t limit 2;

加哪些锁?

```
root@devops_db 19:42: [test]> begin;
Query OK, 0 rows affected (0.00 sec)

root@devops_db 19:42: [test]> insert into t1 select * from t limit 2;
Query OK, 2 rows affected (0.01 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

select
ENGINE_TRANSACTION_ID,THREAD_ID,OBJECT_NAME,index_name,OBJECT_INSTANCE_BEGI
N,LOCK_TYPE,LOCK_MODE,LOCK_STATUS,LOCK_DATA from
performance_schema.data_locks;

ENGINE_TRANSACTION_ID	THREAD_ID	OBJECT_NAME	index_name	OBJECT_INSTANCE_BEGIN	LOCK_TYPE	LOCK_MODE	LOCK_STATUS	LOCK_DATA
3895	119	t1	NULL	130040603931760	TABLE	IX	GRANTED	NULL
3895	119	t	NULL	130040603931664	TABLE	IS	GRANTED	NULL
3895	119	t	PRIMARY	130040603928688	RECORD	S	GRANTED	0
3895	119	t	PRIMARY	130040603928688	RECORD	S	GRANTED	5

基础知识回顾：RR 可重复读隔离级别

```
begin;  
select * from t1;
```

start transaction with consistent
snapshot

自制 online DDL V1

create table t1 like t;

alter table t1 add index(c);

insert into t1 select * from t;

交换表名 : rename table t to t_old, t1 to t;

问题1：锁表时间过长

解法：分段

问题2：insert 过程中有更新

解法：触发器

自制 online DDL V2

```
create table t1 like t;
```

```
alter table t1 add index(c);
```

1. 创建触发器

insert --> replace

delete --> delete

update --> delete+replace

2. insert into t1 select * from t limit 1,10000;

insert into t1 select * from t limit 10000,20000;

自制 online DDL V2

create table t1 like t;

alter table t1 add index(c);

1. 创建触发器

insert --> replace

delete --> delete

update --> delete+replace

2. insert into t1 select * from t limit M,N;

问题1：分页数据不准

解法：start transaction with consistent snapshot

问题2：锁等待问题

解法：log table

自制 online DDL V3

```
create table t1 like t;
```

```
alter table t1 add index(c);
```

1. 创建触发器

```
insert --> insert into log_table("replace...")
```

```
delete --> insert into log_table("delete...")
```

```
update --> insert("delete"); insert into("replace")
```

2. start ...

```
select * from t
```

```
insert into
```

3. lock table t read;

```
select alter_log from log_tables
```

```
apply alter_log
```

问题：锁表时间不可控

解法：递归增量

自制 online DDL V4

```
create table t1 like t;
```

```
alter table t1 add index(c);
```

1. 创建触发器

```
insert --> insert into log_table("replace...")
```

```
delete --> insert into log_table("delete...")
```

```
update --> insert("delete"); insert into("replace")
```

2. start ...

```
select * from t
```

```
insert into
```

3. start ...

```
select aid, alter_log from log_table ;
```

```
apply alter_log
```

```
delete from log_tables where id =$aid
```

4. repeat step 3

5. lock table t; repeat step 3;

问题：表上已经有trigger时

解法：换binlog方案

自制 online DDL V5

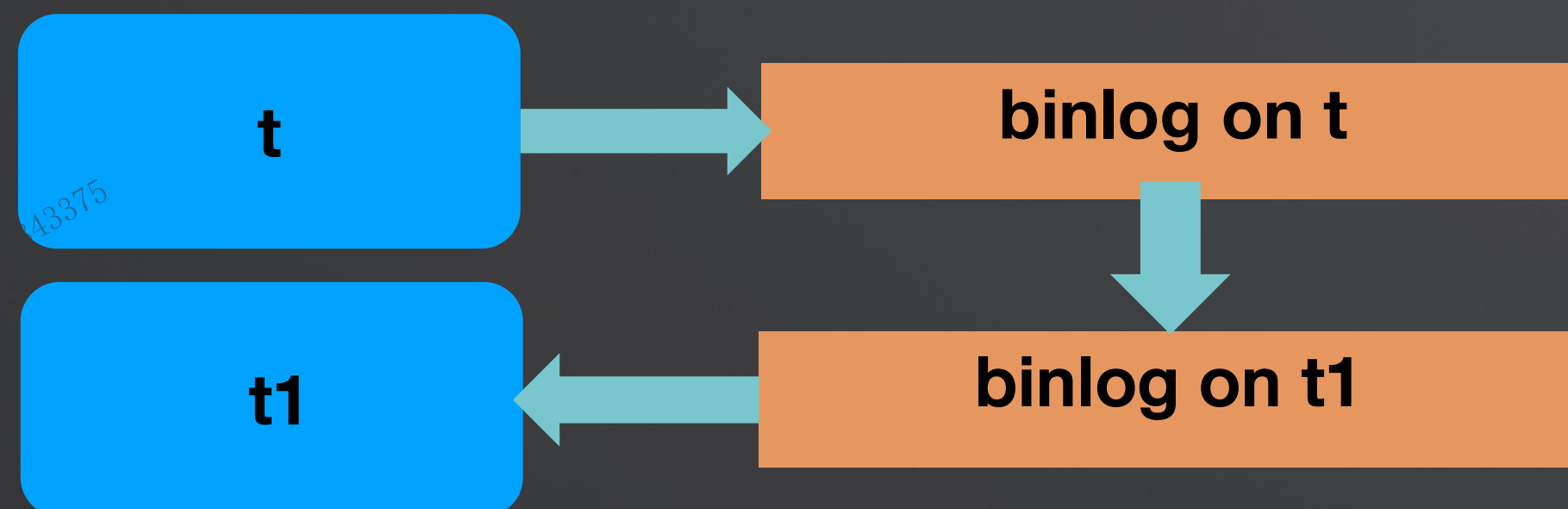
```
create table t1 like t;
```

```
alter table t1 add index(c);
```

1. start ...

```
select * from t
```

```
insert into
```



自制 online DDL V6

```
create table t1 like t;
```

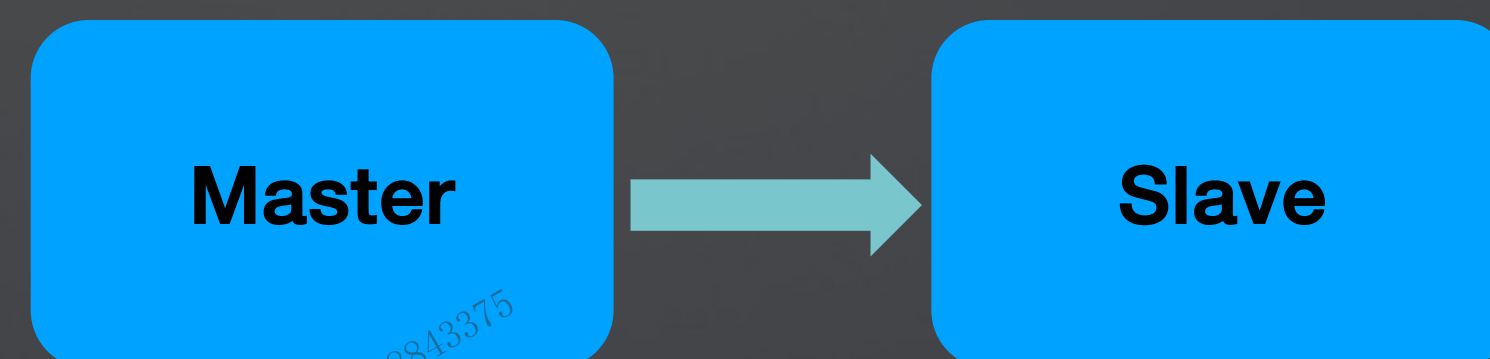
```
alter table t1 discard tablespace;
```

```
flush table t for export;
```

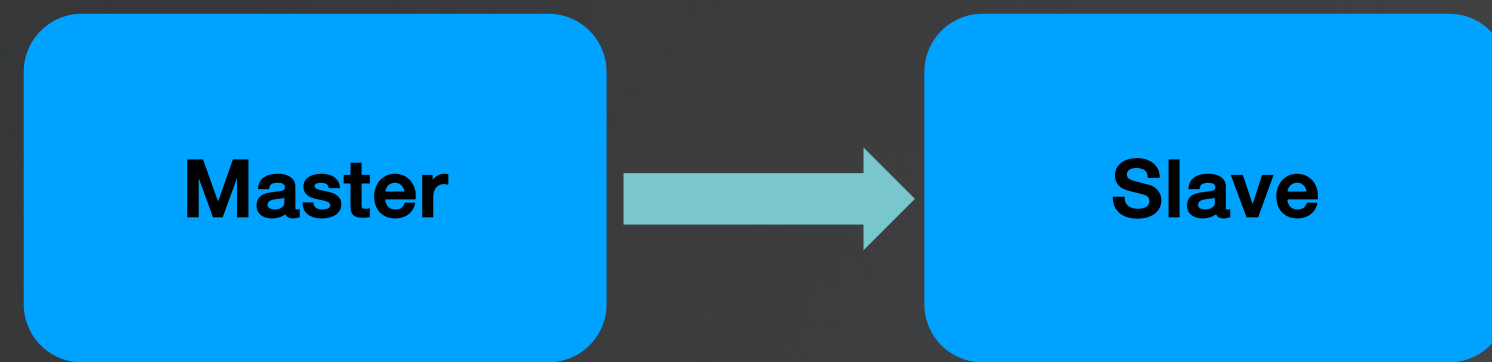
```
//cp t.ibd t1.ibd ; binlog
```

```
unlock tables
```

```
alter table t1 import tablespace;
```



N 个分表拆成 2N 怎么做 - 分区表



```
PARTITION BY list(shopid%32)
(  
  PARTITION p0 VALUES IN (0),  
  PARTITION p1 VALUES IN (1),  
  PARTITION p2 VALUES IN (2),  
  ...  
  PARTITION p3 VALUES IN (31)  
);
```

```
@slave  
set sql_log_bin =off; stop slave;  
create orders_new partition=64;  
insert into orders_new select from orders  
swapname of orders and orders_new  
start slave
```


更多资源v: 2938843375

Q&A

更多资源v: 2938843375

更多资源v: 2938843375

THANKS

更多资源v: 2938843375