实验环境

mysql版本：5.6.24

[percona-toolkit](https://www.percona.com/downloads/percona-toolkit/3.0.2/binary/redhat/6/x86_64/percona-toolkit-3.0.2-1.el6.x86_64.rpm)版本:3.0.2

主库：172.16.40.105:3306

从库：172.16.40.106:3306

安装步骤:

1、下载并安装

#wget <https://www.percona.com/downloads/percona-toolkit/3.0.2/binary/redhat/6/x86_64/percona-toolkit-3.0.2-1.el6.x86_64.rpm>

#yum localinstall [percona-toolkit-3.0.2-1.el6.x86\_64.rpm](https://www.percona.com/downloads/percona-toolkit/3.0.2/binary/redhat/6/x86_64/percona-toolkit-3.0.2-1.el6.x86_64.rpm)

由于数据库binlog\_format为ROW模式，所以只能使用dsn跨数据中心检测

2、配置

在主库上创建dsns表：

create database percona;

CREATE TABLE `percona`.`dsns` ( `id` int(11) NOT NULL AUTO\_INCREMENT, `parent\_id` int(11) DEFAULT NULL, `dsn` varchar(255) NOT NULL, PRIMARY KEY (`id`) );

为了减少不必要的麻烦，确保你的 checksum@'xxx' 用户能同时登陆主库和从库

GRANT SELECT, PROCESS, SUPER, LOCK TABLES, REPLICATION SLAVE, REPLICATION CLIENT ON \*.\* TO 'checksum'@'%' IDENTIFIED BY PASSWORD '\*29E0800F3786835ADC73A063CDA6F31BD55143AB';

use percona;

插入从库信息:

insert into dsns(dsn) values('h=172.16.40.106,P=3306,u=checksum,p=jjjr@2015');

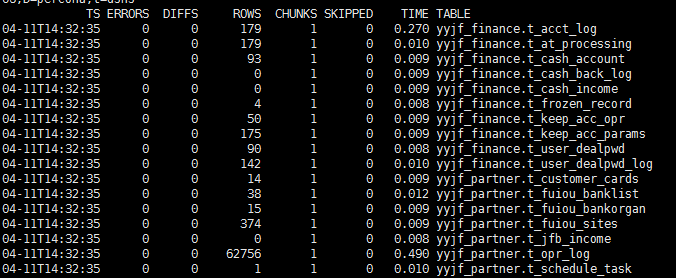
执行校验，只校验yyjf相关库

#pt-table-checksum h=172.16.40.105,u=checksum,p='jjjr@2015',P=3306 --replicate=percona.checksums --nocheck-replication-filters --no-check-binlog-format --databases-regex=yyjf\* --recursion-method dsn=h=172.16.40.105,u=checksum,p='jjjr@2015',P=3306,D=percona,t=dsns

参数解释：

* --replicate= 指定 checksum 计算结果存到哪个库表里，如果没有指定，默认是 percona.checksums
* --no-check-replication-filters 表示不需要检查 Master 配置里是否指定了 Filter。 默认会检查，如果配置了 Filter，如 replicate\_do\_db,replicate-wild-ignore-table,binlog\_ignore\_db 等，在从库checksum就与遇到表不存在而报错退出，所以官方默认是yes（--check-replication-filters）但我们实际在检测中时指定--databases=，所以就不存在这个问题，干脆不检测
* --nocheck-binlog-format：不检测日志格式。这个选项对于 ROW 模式的复制很重要，因为pt-table-checksum会在 Master和Slave 上设置binlog\_format=STATEMENT（确保从库也会执行 checksum SQL），MySQL限制从库是无法设置的，所以假如行复制从库，再作为主库复制出新从库时（A->B->C），B的checksums数据将无法传输。（没验证）
* --recursion-method：发现从库的方式。pt-table-checksum 默认可以在主库的 processlist 中找到从库复制进程，从而识别出有哪些从库，但如果使用是非标准3306端口，会导致找不到从库信息。最终极的办法是dsn，dsn指定的是某个表（如 percona.dsns ），表行记录是改主库的（多个）从库的连接信息。

执行结果：



* TS ：完成检查的时间戳。
* ERRORS ：检查时候发生错误和警告的数量。
* DIFFS ：不一致的chunk数量。当指定 --no-replicate-check 即检查完但不立即输出结果时，会一直为0；当指定 --replicate-check-only 即不检查只从checksums表中计算crc32，且只显示不一致的信息（毕竟输出的大部分应该是一致的，容易造成干扰）。
* ROWS ：比对的表行数。
* CHUNKS ：被划分到表中的块的数目。
* SKIPPED ：由于错误或警告或过大，则跳过块的数目。
* TIME ：执行的时间。
* TABLE ：被检查的表名

查询sql语句即可看到是否有主从不一致发生:

select \* from percona.checksums where master\_cnt <> this\_cnt OR master\_crc <> this\_crc OR ISNULL(master\_crc) <> ISNULL(this\_crc) \G;

常见问题：

1. Diffs cannot be detected because no slaves were found不能自动找到从库，确认processlist或host或dsns方式用对了。
2. Cannot connect to h=slave1.*\**.com,p=...,u=percona\_user

可以在pt-table-checksum命令前加PTDEBUG=1来看详细的执行过程，如端口、用户名、权限错误。

1. Waiting for the --replicate table to replicate to XXX问题出在 percona.checksums 表在从库不存在，根本原因是没有从主库同步过来，所以看一下从库是否延迟严重。
2. Pausing because Threads\_running=25

反复打印出类似上面停止检查的信息。这是因为当前数据库正在运行的线程数大于默认25，pt-table-checksum 为了减少对库的压力暂停检查了。等数据库压力过了就好了，或者也可以直接 Ctrl+C 终端，下一次加上--resume继续执行，或者加大--max-load=值。

1. 字符集问题