前提概要

由于项目上需要将a服务器数据同步至b服务器，一开始使用mysql主从复制，但是由于主从同步无法触发位于b服务器的触发器，只能放弃此方案。后来找到了datax可以实现数据同步同时也可以触发触发器，决定使用此方案。

datax准备

安装datax，python，jdk datax下载及安装地址

根据自身需求选择合适的writer和reader，我这里选择的是mysqlwriter和mysqlreader

下面是我使用的json文件，有两点需要注意

我在 where 使用了sql 语句 create\_time > FROM\_UNIXTIME(${create\_time}) and create\_time < FROM\_UNIXTIME(${end\_time}) ，其中FROM\_UNIXTIME()是mysql时间戳转换为时间格式的函数，${name}是datax提供的占位符后面会使用到

reader中连接字符串添加了useUnicode=true&characterEncoding=utf8 ，因为没有加这个导入到目标数据库中文乱码了，虽然我两边的数据库都是utf8mb4格式的



shell脚本准备

因为我有多张表，编写一个脚本

需要添加source /etc/profile ，因为在cron的系统环境和shell的环境不一样，会导致java commond not found错误 参考地址

$(date +%s) 为获取系统当前时间戳 ，$(($end\_time - 60)) 为算术表达式计算60前的时间戳

"-Dcreate\_time=$create\_time -Dend\_time=$end\_time"这里就是datax使用占位符的作用，可以将外部自定义参数传入

>>/home/gzjp/datax\_log/bus\_clue\_atta\_log.date +%Y%m%d2>&1 我这里把日期都放入每天的日志文件以免单一文件过大

最后由于我的同步的表格较少我直接使用 & 进行后台操作以免发生阻塞



crontab 定时任务准备

$ crontab -e

\*/1 \* \* \* \* /home/gzjp/jm\_db\_sync.sh >/dev/null 2>&1

我是定时每分钟跑一次脚本，注意一定要处理输入文件，因为cron会见执行情况通过mail发给用户，时间长系统会被塞爆

有意义的参考内容

https://helpcdn.aliyun.com/document\_detail/62149.html

---------------------

作者：quadimodo

来源：CSDN

原文：https://blog.csdn.net/quadimodo/article/details/82186788

版权声明：本文为博主原创文章，转载请附上博文链接！