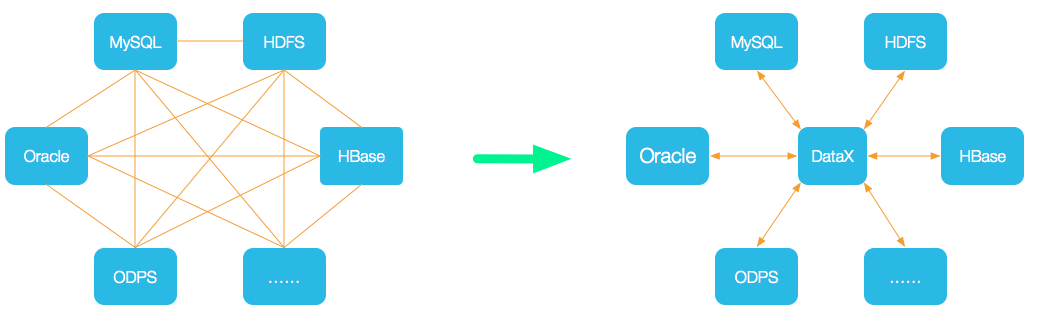
**关于 DataX**

DataX 是阿里巴巴集团内被广泛使用的离线数据同步工具/平台，实现包括 MySQL、Oracle、SqlServer、Postgre、HDFS、Hive、ADS、HBase、TableStore(OTS)、MaxCompute(ODPS)、DRDS 等各种异构数据源之间高效的数据同步功能。



如果想进一步了解 DataX ，请进一步查看 [DataX 详细介绍](https://github.com/alibaba/DataX/blob/master/introduction.md)。

**关于增量更新**

DataX 支持多种数据库的读写， json 格式配置文件很容易编写， 同步性能很好， 通常可以达到每秒钟 1 万条记录或者更高， 可以说是相当优秀的产品， 但是缺乏对增量更新的内置支持。

其实增量更新非常简单， 只要从目标数据库读取一个最大值的记录， 可能是 DateTime 或者 RowVersion 类型， 然后根据这个最大值对源数据库要同步的表进行过滤， 然后再进行同步即可。

由于 DataX 支持多种数据库的读写， 一种相对简单并且可靠的思路就是：

1. 利用 DataX 的 DataReader 去目标数据库读取一个最大值；
2. 将这个最大值用 TextFileWriter 写入到一个 CSV 文件；
3. 用 Shell 脚本来读取 CSV 文件， 并动态修改全部同步的配置文件；
4. 执行修改后的配置文件， 进行增量同步。

接下来就用 shell脚本来一步一步实现增量更新。

**增量更新的 shell 实现**

我的同步环境是从 SQLServer 同步到 PostgreSQL ， 部分配置如下：

{

"job": {

"content": [

{

"reader": {

"name": "sqlserverreader",

"parameter": {

"username": "...",

"password": "...",

"connection": [

{

"jdbcUrl": [

"jdbc:sqlserver://[source\_server];database=[source\_db]"

],

"querySql": [

"SELECT DataTime, PointID, DataValue FROM dbo.Minutedata WHERE 1=1"

]

}

]

}

},

"writer": {

"name": "postgresqlwriter",

"parameter": {

"username": "...",

"password": "...",

"connection": [

{

"jdbcUrl": "jdbc:postgresql://[target\_server]:5432/[target\_db]",

"table": [

"public.minute\_data"

]

}

],

"column": [

"data\_time",

"point\_id",

"data\_value"

],

"preSql": [

"TRUNCATE TABLE @table"

]

}

}

}

],

"setting": { }

}

}

更多的配置可以参考 [SqlServerReader 插件文档](https://github.com/alibaba/DataX/blob/master/sqlserverreader/doc/sqlserverreader.md)以及 [PostgresqlWriter 插件文档](https://github.com/alibaba/DataX/blob/master/postgresqlwriter/doc/postgresqlwriter.md)。

要实现增量更新， 首先要 PostgresqlReader 从目标数据库读取最大日期， 并用 TextFileWriter 写入到一个 csv 文件， 这一步我的配置如下所示：

{

"job": {

"content": [

{

"reader": {

"name": "postgresqlreader",

"parameter": {

"connection": [

{

"jdbcUrl": [

"jdbc:postgresql://[target\_server]:5432/[target\_db]"

],

"querySql": [

"SELECT max(data\_time) FROM public.minute\_data"

]

}

],

"password": "...",

"username": "..."

}

},

"writer": {

"name": "txtfilewriter",

"parameter": {

"dateFormat": "yyyy-MM-dd HH:mm:ss",

"fileName": "minute\_data\_max\_time\_result",

"fileFormat": "csv",

"path": "/scripts/",

"writeMode": "truncate"

}

}

}

],

"setting": { }

}

}

更多的配置可以看考 [PostgresqlDataReader 插件文档](https://github.com/alibaba/DataX/blob/master/postgresqlreader/doc/postgresqlreader.md)以及 [TextFileWriter 插件文档](https://github.com/alibaba/DataX/blob/master/txtfilewriter/doc/txtfilewriter.md)

有了这两个配置文件， 现在可以编写增量同步的 shell 文件， 内容如下：

#!/bin/bash

### every exit != 0 fails the script

set -e

# 获取目标数据库最大数据时间，并写入一个 csv 文件

docker run --interactive --tty --rm --network compose --volume $(pwd):/scripts \

beginor/datax:3.0 \

/scripts/minute\_data\_max\_time.json

if [ $? -ne 0 ]; then

echo "minute\_data\_sync.sh error, can not get max\_time from target db!"

exit 1

fi

# 找到 DataX 写入的文本文件，并将内容读取到一个变量中

RESULT\_FILE=`ls minute\_data\_max\_time\_result\_\*`

MAX\_TIME=`cat $RESULT\_FILE`

# 如果最大时间不为 null 的话， 修改全部同步的配置，进行增量更新；

if [ "$MAX\_TIME" != "null" ]; then

# 设置增量更新过滤条件

WHERE="DataTime > '$MAX\_TIME'"

sed "s/1=1/$WHERE/g" minute\_data.json > minute\_data\_tmp.json

# 将第 45 行的 truncate 语句删除；

sed '45d' minute\_data\_tmp.json > minute\_data\_inc.json

# 增量更新

docker run --interactive --tty --rm --network compose --volume $(pwd):/scripts \

beginor/datax:3.0 \

/scripts/minute\_data\_inc.json

# 删除临时文件

rm ./minute\_data\_tmp.json ./minute\_data\_inc.json

else

# 全部更新

docker run --interactive --tty --rm --network compose --volume $(pwd):/scripts \

beginor/datax:3.0 \

/scripts/minute\_data.json

fi

在上面的 shell 文件中， 使用我制作的 DataX docker 镜像， 使用命令 docker pull beginor/datax:3.0 即可获取该镜像， 当也可以修改这个 shell 脚本直接使用 datax 命令来执行。

**为什么用 shell 来实现**

因为 DataX 支持多种数据库的读写， 充分利用 DataX 读取各种数据库的能力， 减少了很多开发工作， 毕竟 DataX 的可靠性是很好的。