SSH

ssh连接

```
ssh连接报错(不管是重启还是怎样):
java.net.ConnectException: Connection refused: connect

查看日志文件
/var/log/message
查看到其中一行
sshd: Missing privilege separation directory:/var/empty/sshd
sshd:丢失的权限分离目录:/var/empty/sshd
然后直接mkdir -p /var/empty/sshd
然后启动systemctl start sshd
```

Hadoop

hadoop的初始化元数据失败

```
#其中提示这个,因为该root用户没有权限
Underlying cause: java.sql.SQLException : Access denied for user 'root'@'node1' (using
password: YES)
通过以下命令，进行远程访问的授权
create user 'root'@'%' identified with mysql_native_password by 'root';
grant all privileges on *.* to 'root'@'%' with grant option;
flush privileges;

通过 ALTER USER 'root'@'localhost' IDENTIFIED BY 'root' PASSWORD EXPIRE NEVER;
命令修改加密规则，MySql8.0 版本 和 5.0 的加密规则不一样，而现在的可视化工具只支持旧的加密方式。
```

Could not locate executable null\bin\winutils.exe in the Hadoop binaries解决方式

```
18/07/21 10:18:24 ERROR util.Shell: Failed to locate the winutils binary in the hadoop binary path
java.io.IOException: Could not locate executable null\bin\winutils.exe in the Hadoop binaries.
        at org.apache.hadoop.util.Shell.getQualifiedBinPath(Shell.java:382)
        at org.apache.hadoop.util.Shell.getWinUtilsPath(Shell.java:397)
        at org.apache.hadoop.util.Shell.<clinit>(Shell.java:390)
        at org.apache.hadoop.util.StringUtils.<clinit>(StringUtils.java:80)
        at org.apache.hadoop.security.SecurityUtil.getAuthenticationMethod(SecurityUtil.java:611)
        at org.apache.hadoop.security.UserGroupInformation.initialize(UserGroupInformation.java:274)
        at org.apache.hadoop.security.UserGroupInformation.ensureInitialized(UserGroupInformation.java:262)
        at org.apache.hadoop.security.UserGroupInformation.loginUserFromSubject(UserGroupInformation.java:807)
        at org.apache.hadoop.security.UserGroupInformation.getLoginUser(UserGroupInformation.java:777)
```

2. 问题解决：

仔细查看报错是缺少winutils.exe程序。

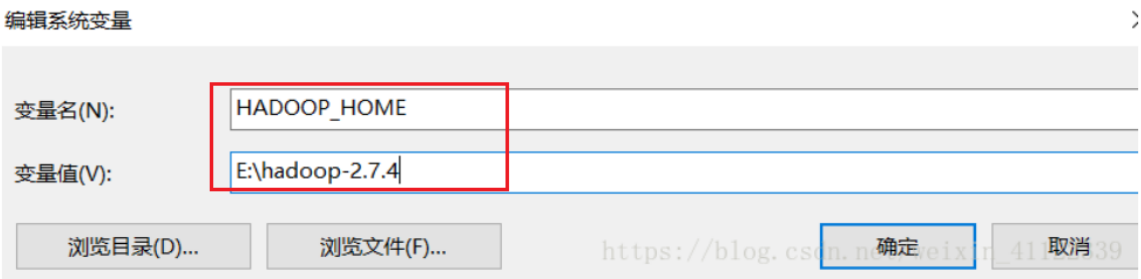
Hadoop都是运行在Linux系统下的，在windows下eclipse中运行mapreduce程序，要首先安装Windows下运行的支持插件（我的是hadoop2.7.4）

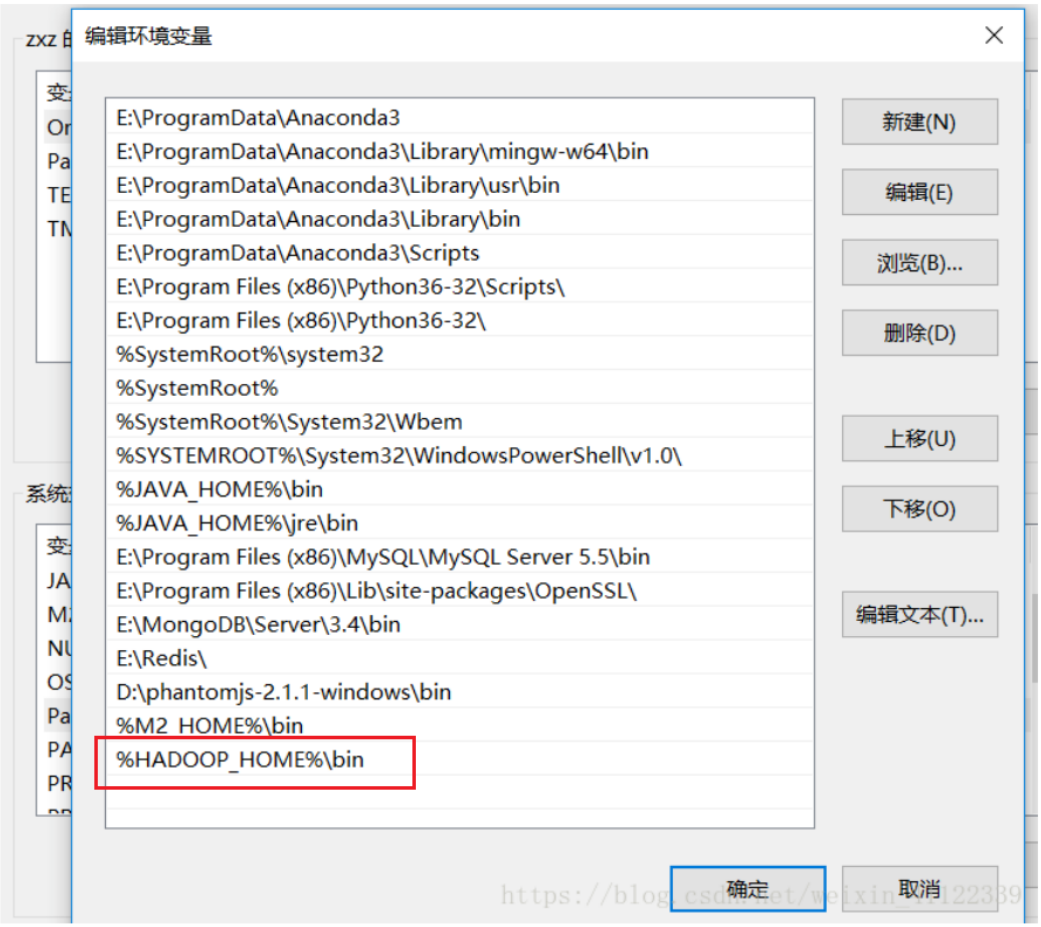
3. 安装并配置插件(我这里还是Linux版的hadoop安装包，我们只需要下载一个winutils.exe文件即可)

下载winutils.exe

https://github.com/steveloughran/winutils/tree/master/hadoop-2.7.1/bin

2.设置环境变量(第一个在环境变量下创建HADDOOP_HOME这个变量，第二个在path下添加最后一行的变量)：





3.至此重启电脑后，问题便可以解决了

```
下载 winutils.exe
创建文件夹，比如说
C:\winutils\bin
winutils.exe
里面复制
C:\winutils\bin
将环境变量设置
HADOOP_HOME
为
```

```
C:\winutils
```

MySQL

MySQL登录出现ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)

```
密码不对
```

Hive

Hive启动matestore出现：null, message from server: "Host 'node1' is not allowed to connect to this MySQL server"

```
#一般是MySQL不需要外部连接
mysql -u123456 -p
use mysql;
select user,host from user;
updata user set host="%" where user="root";
flush privileges;

centos修改mysql密码或者进入mysql后解决Access denied for user ''@'localhost' to database 'mysql错误
原因是MySQL的密码有问题

用mysql匿名用户可以进入数据库，但是看不见mysql数据库.

解决办法:
具体操作步骤:
关闭mysql:
# service mysqld stop
然后:
# mysqld_safe --skip-grant-tables
开启另一个终端并启动mysql:
# service mysqld start
mysql -u root
mysql> use mysql
mysql> UPDATE user SET Password=PASSWORD('root') WHERE user='root';
```

```
mysql> flush privileges;
mysql>\q

到这里密码已经修改成功，
mysql -u root -p
```

hive启动metastore出现 MetaException(message:Version information not found in metastore.)
在hive-site.xml添加

```
<property>
    <name>hive.metastore.schema.verification</name>
    <value>false</value>
</property>
```

hive启动metastore出现 MetaException(message:Required table missing : “DBS “ in Catalog “ “ Schema “ “. DataNucleus requires t
在hive-site.xml添加

```
<property>
    <name>datanucleus.schema.autoCreateAll</name>
    <value>true</value>
 </property>
```

hive启动metastore出现 MetaException(message:Error(s) were found while auto-creating/validating the datastore for classes. The errors are printed in the log, and are attached to this exception.)

```
是到mysql中的hive数据库里执行
alter database hive3 character set latin1;
改变hive元数据库的字符集
```

return code 137错误

Error while processing statement: FAILED: Execution Error, return code 137 from org.apache.hadoop.hive.ql.exec.mr.MapredLocalTask

```
错误的原因 ： 在执行 多表join的操作，HIVE会优化尝试 mapjoin，将小表的数据放置在内存中，但是内存不足无法放置，导致运行失败

解决方案：
    关闭掉mapjoin
    set hive.auto.convert.join= false;
```

执行select 出现 could not connect to hadoop02：10000报错

```
Could not connect to hadoop02:10000 (code THRIFTTRANSPORT): TTransportException('Could not connect to hadoop02:10000',)
```

```
原因：hiveserver2内存过小，导致无法执行，异常退出

解决方案：
  修改hiveserver2的java 堆栈大小

调整后 重启hive
```

Error in semantic analysis:DISTINCT on different columns notsupported with skew in data.

说明：当hive.groupby.skewindata=true时，hive不支持多列上的去重操作

hive 报错**ORC split generation failed with exception**:

java.lang.ArrayIndexOutOfBoundsException: 5

原因：hive版本是2.1.1，orc文件是由orcfilewriter用更大的版本写的，用的是spark高版本写的，所以hive低版本会查询失败，spark查询没问题

解决：1.临时：可以创建textFile文件格式先导入到textFile格式，

2.spark的orc版本过高，则只用hive创建和导入，spark只做查询

hive报错

Execution Error, return code 2 from org.apache.hadoop.hive.ql.exec.mr.MapRed

Error: java.io.IOException: java.lang.reflect.InvocationTargetException

Caused by: java.lang.ArrayIndexOutOfBoundsException: 7

本次遇到的问题是：hive版本是2.1.1，orc文件是由orcfilewriter用更大的版本写的，用的是spark高版本写的，所以hive低版本会查询失败，spark查询没问题

动态分区报错

```
Fatal error occurred when node tried to create too many dynamic partitions. The maximum number of dynamic partitions is controlled by hive.exec.max.dynamic.partitions and hive.exec.max.dynamic.partitions.pernode. Maximum was set to 100 partitions per node, number of dynamic partitions on this node: 101
```

```
Caused by: org.apache.hadoop.hive.ql.metadata.HiveFatalException: [Error 20004]: Fatal error occurred when node tried to create too many dy
namic partitions. The maximum number of dynamic partitions is controlled by hive.exec.max.dynamic.partitions and hive.exec.max.dynamic.part
itions.pernode. Maximum was set to 100 partitions per node, number of dynamic partitions on this node: 101
        at org.apache.hadoop.hive.ql.exec.FileSinkOperator.getDynOutPaths(FileSinkOperator.java:951)
        at org.apache.hadoop.hive.ql.exec.FileSinkOperator.process(FileSinkOperator.java:722)
        at org.apache.hadoop.hive.ql.exec.Operator.forward(Operator.java:882)
        at org.apache.hadoop.hive.ql.exec.SelectOperator.process(SelectOperator.java:95)
        at org.apache.hadoop.hive.ql.exec.mr.ExecReducer.reduce(ExecReducer.java:234)
        ... 7 more

2020-06-18 04:44:00,251 INFO [IPC Server handler 17 on 39454] org.apache.hadoop.mapred.TaskAttemptListenerImpl: Diagnostics report from att
empt_1591389362937_0111_r_000000_0: Error: java.lang.RuntimeException: org.apache.hadoop.hive.ql.metadata.HiveException: Hive Runtime Error
while processing row (tag=0) {"key":{"reducesinkkey0":"1042980"},"value":{"_col0":1042980,"_col1":1461818666,"_col2":"0","_col3":1,"_col
4":"INVALID_PUBLIC_OLD_CLUE","_col5":"false","_col6":"NETSERVICE","_col7":0,"_col8":229,"_col9":"12","_col10":"1","_col11":"0","_col12":
2,"_col13":"2016","_col14":"04","_col15":"28"}}
        at org.apache.hadoop.hive.ql.exec.mr.ExecReducer.reduce(ExecReducer.java:255)
        at org.apache.hadoop.mapred.ReduceTask.runOldReducer(ReduceTask.java:445)
```

解决:

```
set hive.exec.max.dynamic.partitions.pernode=10000; -- 最多可以几个分区
set hive.exec.max.dynamic.partitions=100000;
set hive.exec.max.created.files=150000;
```

Hive查询时，报错java.lang.OutOfMemoryError: Java heap space

问题：join时：属于JVM堆内存溢出了

解决方式1：关闭mapjoin

```
set hive.auto.convert.join = false;
```

解决方式2：调整内存

```
mapreduce不得超过yarn最大内存
--Task内存
mapreduce.map.java.opts=-Xmx6000m;
mapreduce.map.memory.mb=6096;
mapreduce.reduce.java.opts=-Xmx6000m;
mapreduce.reduce.memory.mb=6096;
--yarn内存-可以不设置
yarn.scheduler.maximum-allocation-mb=4096
yarn.scheduler.minimum-allocation-mb=1024
```

pycharm

pycharm启动spark出现Java gateway process exited before sending its port number

解决：把.bashrc 文件配置所需要的全局变量然后souce .bashrc

Python项目包含：

```
实现的代码包
```

```
    入口文件.py
    实体类包
    工具类包
config：配置文件目录（配置日志文件）
log：日志文件目录
resource：资源文件目录
```

安装PyHive、Oracle库

- step1：在Windows的用户家目录下创建pip.ini文件 自己创建即可
  - 例如：**C:\Users\Frank\pip\pip.ini**
  - 内容：指定pip安装从阿里云下载

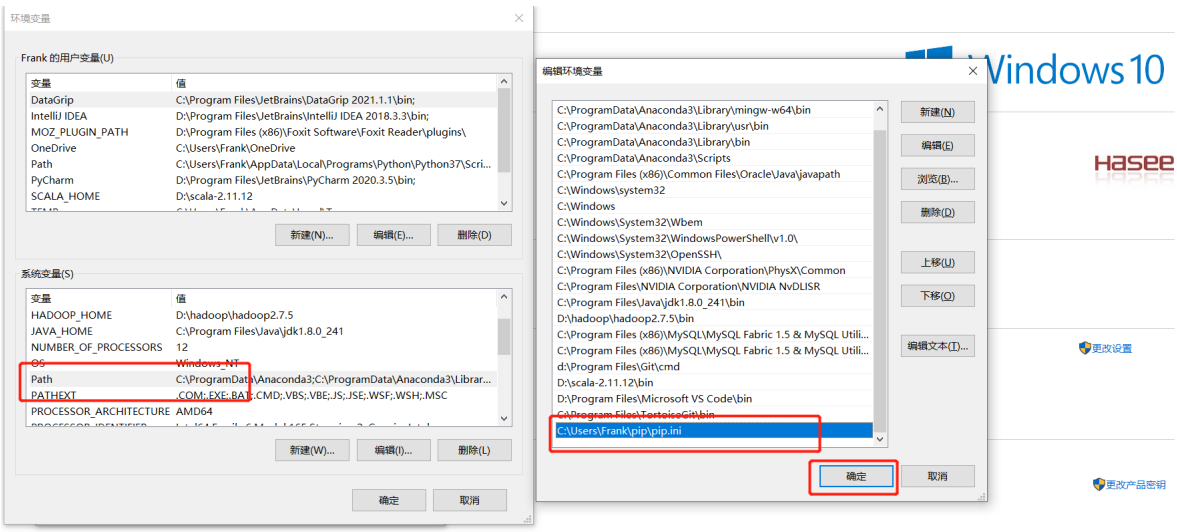
```
文件内容：
[global]

index-url=http://mirrors.aliyun.com/pypi/simple/

[install]

trusted-host=mirrors.aliyun.com
```

- step2：将文件添加到Windows的**Path环境变量**中



Oracle本地驱动目录

将提供的**instantclient_12_2**目录放入D盘的根目录下

instantclient-basic....zip
71.58MB

PyHive本地连接配置：将提供的CMU目录放入C盘的根目录下

CMU.rar
7.24KB

pycharm配置仓库为阿里源

http://mirrors.aliyun.com/pypi/simple=

一站式制造所需要的包

```
# 安装sasl包 -> 使用pycharm安装，会存在下载失败情况，因此提前下载好，对应python3.7版本
pip install sasl-0.2.1-cp37-cp37m-win_amd64.whl
# 安装thrift包
pip install thrift
# 安装thrift sasl包
pip install thrift-sasl
# 安装python操作oracle包
pip install cx-Oracle
# 安装python操作hive包，也可以操作sparksql
pip install pyhive
```

相关代码

OneMake30.zip
12.49MB

牛客网进阶sql第31题解题思路

```
    算出SQL未完成率然后排名去排名的50%以下的数据 用PERCENT_RANK() over() ：计算窗口下从第一个到最后一个的百分比 1-100%
    后面在过滤level 6和7
```

jps

如果jps不显示进程（这个文件夹每启动一个java进程就会有进程号）

```
rm -rf /tmp/hsperfdata_root 删除这个目录会
```

Java-MR

java编写MapReduce出现 (null) entry in command string: null

```
解决方法：
下载hadoop.dll文件，拷贝到c:\windows\system32目录中即可hadoop.dll
可以在github上下载：https://github.com/4ttty/winutils
各个版本的hadoop.dll好像是通用的。
```

海豚调度器

海豚调度器执行任务显示无权限

- 如果报错rent_actuary 没有EXECUTE权限，就vim /etc/passwd文件，将rent_actuary 的用户id改成0，即升级为root权限。

```
rent_actuary:x:0:0::/home/rent_actuary:/bin/bash
```

log4j警告:WARN Please initialize the log4j system properly

问题描述：

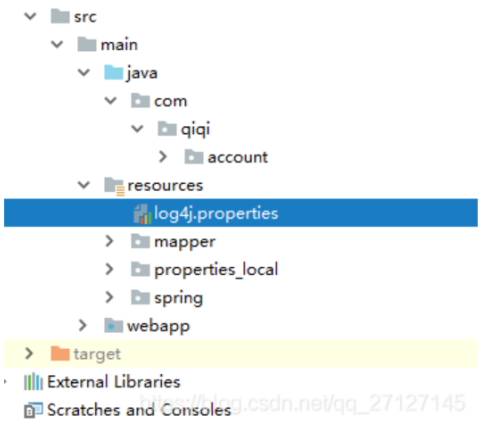
```
log4j:WARN No appenders could be found for logger (org.springframework.test.context.junit4.SpringJUnit4ClassRunner).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
```

出现这个警告的原因是没有配置log4j.properties文件或者该配置文件在项目目录中的位置不对。

对于非Maven项目，log4j.properties文件放在src根目录下。

但是对于Maven项目，，log4j. properties文件必须放在Resource文件下。

在Maven项目中，如下图：



文件内容为：log4j.properties

```
log4j.rootLogger=WARN, stdout
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
```

```
log4j.appender.stdout.layout.ConversionPattern=%d %p [%c] - %m%n
```

Maven

使用Maven构建项目时， 执行编译或者打包，报错误

was cached in the local repository, resolution will not be reattempted until the update interval of

问题原因 :Maven默认会使用本地缓存的库来编译工程，对于上次下载失败的库，maven会在~/.m2/repository/<group>/<artifact>/<version>/目录下创建xxx.lastUpdated文件，一旦这个文件存在，那么在直到下一次nexus更新之前都不会更新这个依赖库。

将其中的仓库添加 <updatePolicy>always</updatePolicy>来强制每次都更新依赖库。

更新settings.xml配置文件 添加

```
<repositories>
        <repository>
                <id>central</id>
                <url>http://central</url>
                <releases>
                        <enabled>true</enabled>
                        <updatePolicy>always</updatePolicy>
                </releases>
                <snapshots>
                        <enabled>true</enabled>
                        <updatePolicy>always</updatePolicy>
                </snapshots>
        </repository>
</repositories>
```

Failure to find org.glassfish:javax.el:pom:3.0.1-b06-SNAPSHOT in https://rep。。。。。。

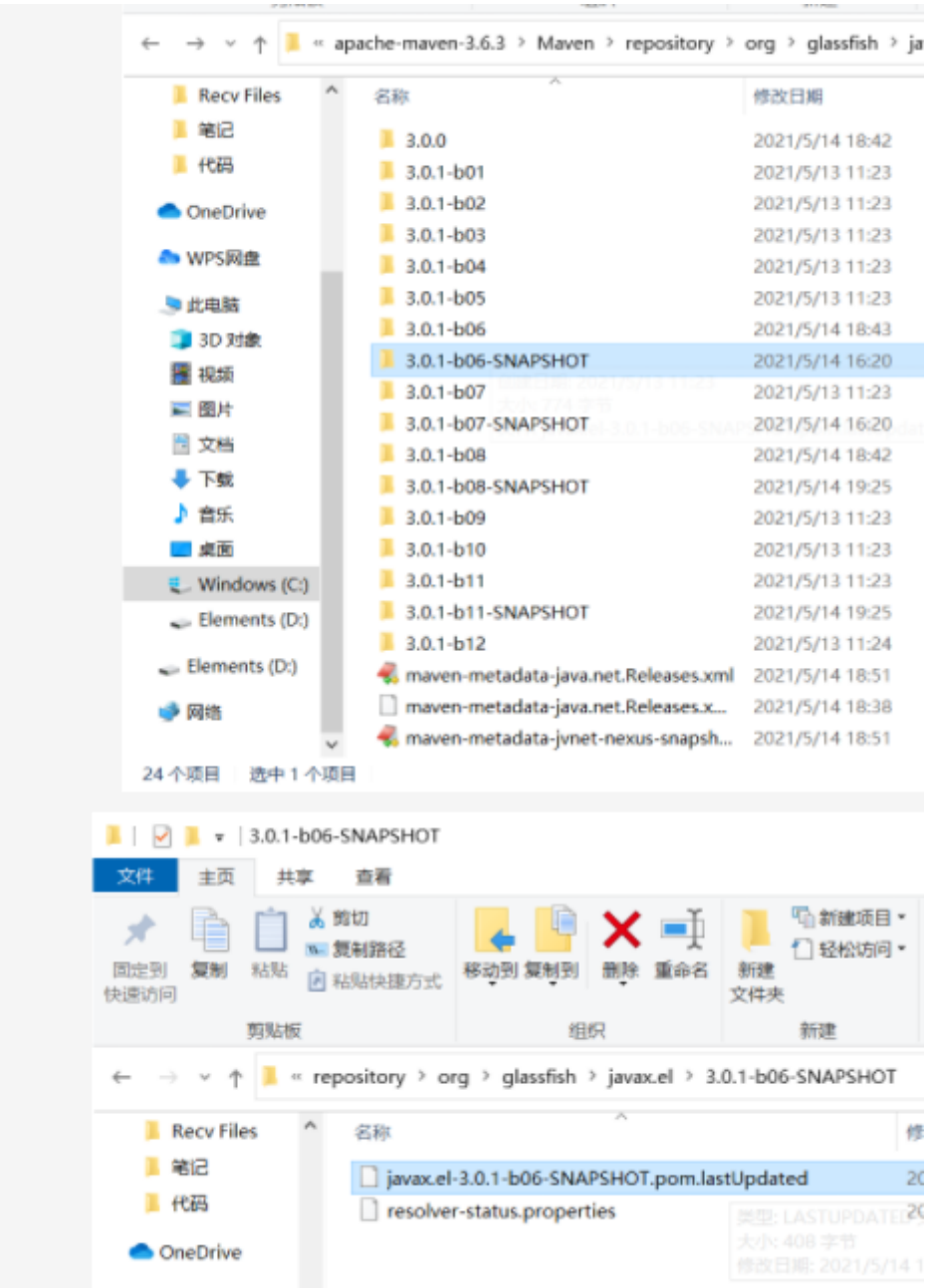
第一步：先通过pom.xml文件进行下载

```
<dependency>
            <groupId>org.glassfish</groupId>
            <artifactId>javax.el</artifactId>
            <version>3.0.1-b08</version>
        </dependency>
```

第二步：下载完成后找到Maven仓库目录的位置，找到 maven仓库目录\repository\org\glassfish\javax.el\，本人的仓库目录是：

```
C:\workspace\root\apache-maven-3.6.3\Maven\repository\org\glassfish\javax.el\
```

第三步：进入javax.el 目录下，找到后缀名为SNAPSHOT的所有文件夹，进入每个文件夹中，修改javax.el-3.0.1-b06-SNAPSHOT.pom.lastUpdated文件名称，去掉文件的.lastUpdated后缀即可



第四步：修改完成，重新使用idea进行打包，发现打包成功

HBase

提示没有那个文件

```
错误：找不到或无法加载主类 org.apache.hadoop.hbase.util.GetJavaProperty
错误：找不到或无法加载主类 org.apache.hadoop.hbase.util.HBaseConfTool
错误：找不到或无法加载主类 org.apache.hadoop.hbase.util.GetJavaProperty
错误：找不到或无法加载主类 org.apache.hadoop.hbase.zookeeper.ZKServerTool
running master, logging to /export/server/hbas/logs/hbase-root-master-node1.itcast.cn.out
nice: /export/server/hbas/bin/hbase: 没有那个文件或目录
cat: /export/server/hbas/conf/regionservers: 没有那个文件或目录
cat: /export/server/hbas/conf/regionservers: 没有那个文件或目录
```

解决

```
把hbase目录创建软连接指向hbas
ln -s /export/server/hbase /expoer/server/hbae
```

Flink

Cannot create Hadoop Security Module 启动flink发现：原因，缺少hadoop的jar包

- Flink Shaded Hadoop-jar：由于不清楚哪个所以都下载了

flink-shaded-hadoop-2-uber-2.8.3-10.0.jar	41.3 MB	JAR 文件
flink-shaded-hadoop-3-3.1.1.7.2.9.0-173-9.0.jar	37.6 MB	JAR 文件
flink-shaded-hadoop-3-3.1.1.7.0.3.0-79-7.0.jar	33 MB	JAR 文件

报错：from hdfs:/node1:8020/flink/ha/default/blob is not a valid DFS filename.

- 原因是配置文件的hdfs路径多了个 /

Application with id "appid'' doesn't exist in RM.

原因：flinkSQL客户端连接不上yarn

解决：启动yarn-session.sh

Tortoise图标不显示解决

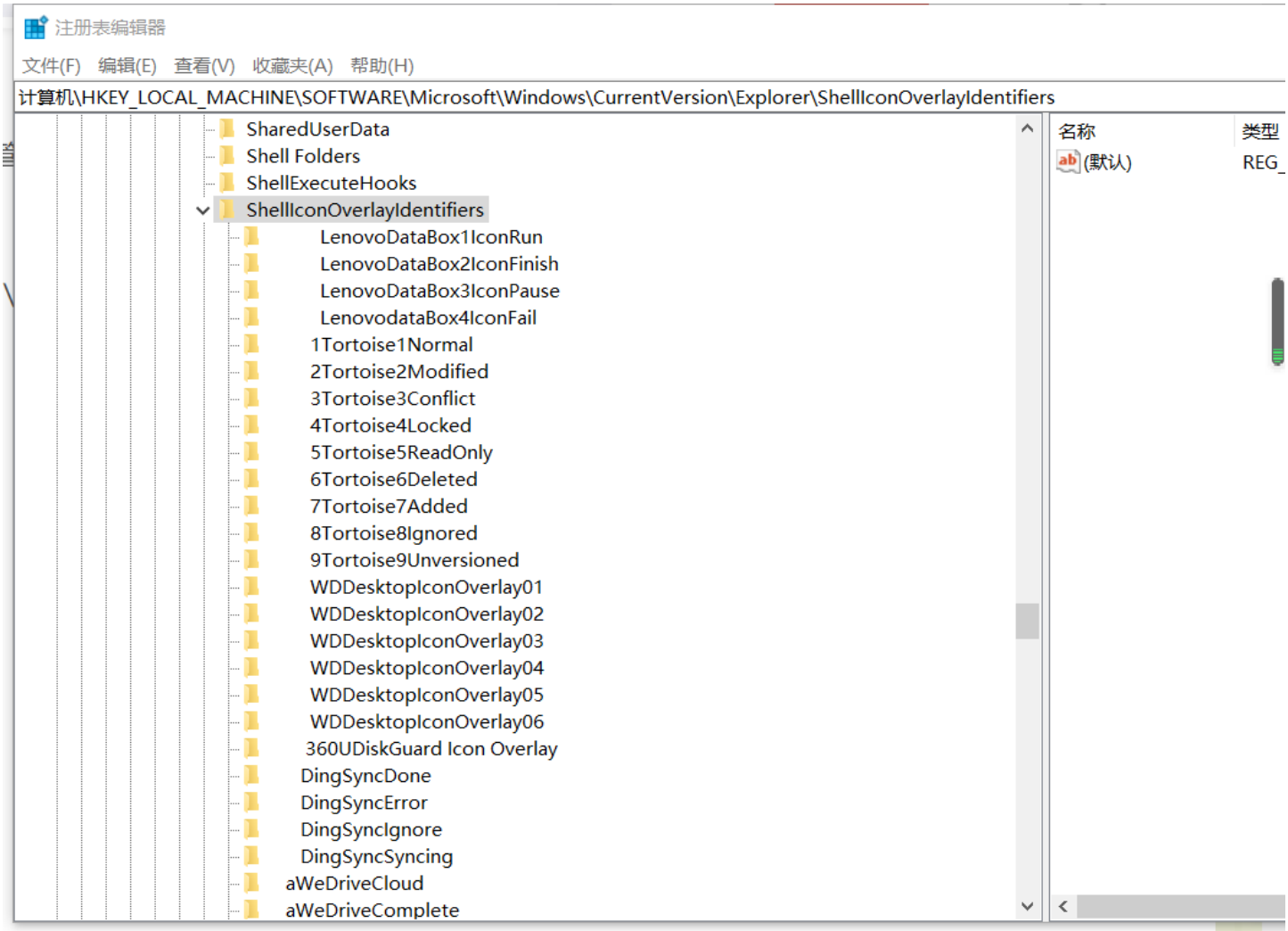
1，win + R 输入regedit 打开注册表管理器

2，找到以下位置：

计算机

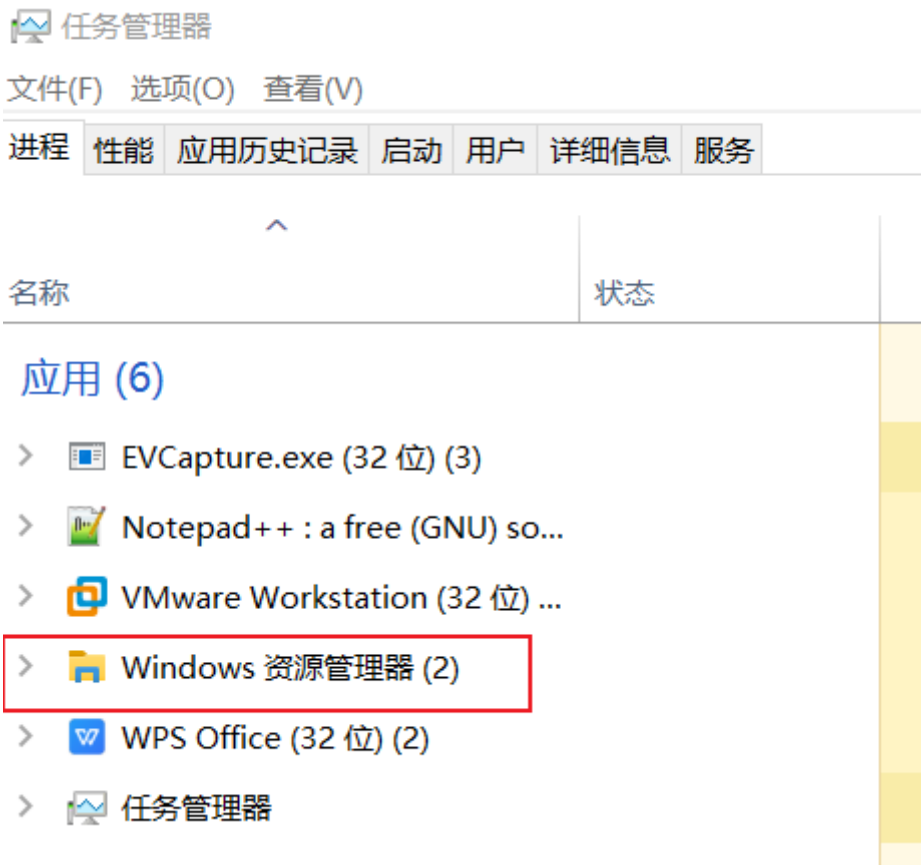
\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ShellIconOverlayIdentifiers

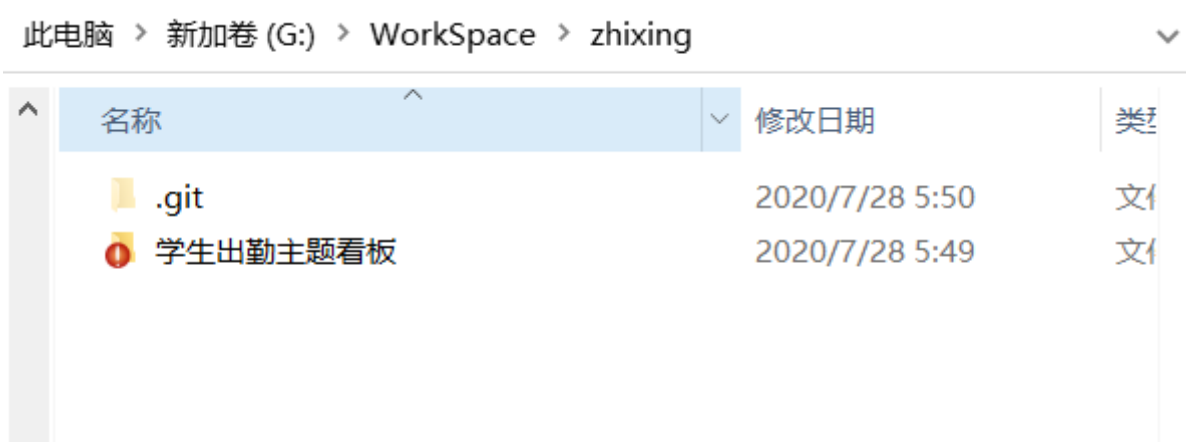
3，可以看到：



4，我这里是已经修改好了，原有的顺序是OneDrive1-7在前面（因为OneDrive前有空格，所以排序在最前），1-9开头的Tortoise在后面，所以没显示。我把所有的Tortoise前重命名也加上了空格，然后退出重开注册表，就能看到排序在先了。

重启电脑或重启资源管理器，再回到git本地仓库文件夹，熟悉的文件图标出现了





5，说明：

Windows 内部是按图标名称的字母顺序来优先显示的，以前OneDrive1-7是在最前面的，它们的命名是“OneDrive1”，名称前加了个空格，所以排在最先。

sqoop

导出失败

```
Export job failed! 导出失败
        at org.apache.sqoop.mapreduce.ExportJobBase.runExport(ExportJobBase.java:444)
        at org.apache.sqoop.manager.SqlManager.exportTable(SqlManager.java:930)
        at org.apache.sqoop.tool.ExportTool.exportTable(ExportTool.java:93)
        at org.apache.sqoop.tool.ExportTool.run(ExportTool.java:112)
        at org.apache.sqoop.Sqoop.run(Sqoop.java:146)
        at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:76)
        at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:182)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:233)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:242)
        at org.apache.sqoop.Sqoop.main(Sqoop.java:251)
21/02/24 15:10:35 INFO hive.metastore: Closed a connection to metastore, current connections: 1
```

说明：

对于sqoop而言，需要将导入导出命令翻译为MR进行执行的，之后sqoop只需要捕获MR最终执行结果即可，如果成功，标记为成功，如果失败，就会告知导入或者导出失败了

具体失败的原因，需要查询MR的日志，从MR日志才能判断出为何而失败

如何查看MR的日志呢？jobHistory

Retired Jobs

Show 20 entries Search:

Submit Time	Start Time	Finish Time	Job ID	Name	User	Queue	State	Maps Total	Maps Completed	Reduces Total	Reduces Completed	Elapsed Time
2021.02.24 15:10:14 CST	2021.02.24 15:10:23 CST	2021.02.24 15:10:34 CST	job_1614126502521_0025	itcast_visit.jar	root	root.users.root	FAILED	0	0	0	0	00hrs, 00mins, 10sec
2021.02.24 15:04:29 CST	2021.02.24 15:04:37 CST	2021.02.24 15:04:57 CST	job_1614126502521_0024	-- 统计每年,	hue	root.users.hue	SUCCEEDED	1	1	1	1	00hrs, 00mins, 20sec
2021.02.24	2021.02.24	2021.02.24	job_1614126502521_0023	-- 统计每年,	hue	root.users.hue	SUCCEEDED	1	1	1	1	00hrs, 00mins,

Job Overview

Job Name: itcast_visit.jar
User Name: root
Queue: root.users.root
State: FAILED
Uberized: false
Submitted: Wed Feb 24 15:10:14 CST 2021
Started: Wed Feb 24 15:10:23 CST 2021
Finished: Wed Feb 24 15:10:34 CST 2021
Elapsed: 10sec
Diagnostics: Task failed task_1614126502521_0025_m_000000 Job failed as tasks failed. failedMaps:1 failedReduces:0 killedMaps:0 killedReduces: 0

ApplicationMaster

Attempt Number	Start Time	Node	Logs
1	Wed Feb 24 15:10:18 CST 2021	hadoop02:8042	logs

Task Type	Total	Complete
Map	1	1
Reduce	0	0

Attempt Type	Failed	Killed	Successful
Maps	1	0	0
Reduces	0	0	0

Log Type: syslog
Log Upload Time: 星期三 二月 24 15:10:41 +0800 2021
Log Length: 98215
Showing 4096 bytes of 98215 total. Click here for the full log. 查看更加详细的日志

```
t/job_1614126502521_0025.summary_tmp to hdfs://hadoop01:8020/user/history/done_intermediate/root/job_1614126502521_0025.summary
2021-02-24 15:10:34,270 INFO [Thread-71] org.apache.hadoop.mapreduce.jobhistory.JobHistoryEventHandler: Moved tmp to done: hdfs://hadoop01:8020/user/history/done_intermediate/root/job_1614126502521_0025_conf.xml_tmp to hdfs://hadoop01:8020/user/history/done_intermediate/root/job_1614126
2021-02-24 15:10:34,272 INFO [Thread-71] org.apache.hadoop.mapreduce.jobhistory.JobHistoryEventHandler: Moved tmp to done: hdfs://hadoop01:8020/user/history/done_intermediate/root/job_1614126502521_0025-1614150614281-root-itcast_visit.jar-1614150634099-0-0-FAILED-root.users.root-1614150
2021-02-24 15:10:34,272 INFO [Thread-71] org.apache.hadoop.mapreduce.jobhistory.JobHistoryEventHandler: Stopped JobHistoryEventHandler. super.stop()
2021-02-24 15:10:34,273 INFO [Thread-71] org.apache.hadoop.mapreduce.v2.app.launcher.ContainerLauncherImpl: KILLING attempt_1614126502521_0025_m_000000_0
2021-02-24 15:10:34,292 INFO [AsyncDispatcher event handler] org.apache.hadoop.mapreduce.v2.app.job.impl.TaskAttemptImpl: attempt_1614126502521_0025_m_000000_0 transitioned from state FAIL_FINISHING_CONTAINER to FAILED, event type is TA_CONTAINER_CLEANED and nodeId=hadoop03:8041
2021-02-24 15:10:34,296 INFO [Thread-71] org.apache.hadoop.mapreduce.v2.app.rm.RMCommunicator: Setting job diagnostics to Task failed task_1614126502521_0025_m_000000
Job failed as tasks failed. failedMaps:1 failedReduces:0 killedMaps:0 killedReduces: 0

2021-02-24 15:10:34,295 INFO [Thread-71] org.apache.hadoop.mapreduce.v2.app.rm.RMCommunicator: History url is http://hadoop01:19888/jobhistory/job/job_1614126502521_0025
2021-02-24 15:10:34,302 INFO [Thread-71] org.apache.hadoop.mapreduce.v2.app.rm.RMCommunicator: Waiting for application to be successfully unregistered.
2021-02-24 15:10:35,304 INFO [Thread-71] org.apache.hadoop.mapreduce.v2.app.rm.RMContainerAllocator: Final Stats: PendingReds:0 ScheduledMaps:0 ScheduledReds:0 AssignedMaps:1 AssignedReds:0 CompletedMaps:0 CompletedReds:0 ContAlloc:1 ContRel:0 HostLocal:1 RackLocal:0
2021-02-24 15:10:35,305 INFO [Thread-71] org.apache.hadoop.mapreduce.v2.app.MRAppMaster: Deleting staging directory hdfs://hadoop01:8020 /user/root/.staging/job_1614126502521_0025
2021-02-24 15:10:35,309 INFO [Thread-71] org.apache.hadoop.ipc.Server: Stopping server on 39464
2021-02-24 15:10:35,312 INFO [IPC Server Responder] org.apache.hadoop.ipc.Server: Stopping IPC Server Responder
2021-02-24 15:10:35,312 INFO [TaskHeartbeatHandler PingChecker] org.apache.hadoop.mapreduce.v2.app.TaskHeartbeatHandler: TaskHeartbeatHandler thread interrupted
2021-02-24 15:10:35,312 INFO [IPC Server listener on 39464] org.apache.hadoop.ipc.Server: Stopping IPC Server listener on 39464
2021-02-24 15:10:35,312 INFO [Ping Checker] org.apache.hadoop.yarn.util.AbstractLivelinessMonitor: TaskAttemptFinishingMonitor thread interrupted
2021-02-24 15:10:40,313 INFO [Thread-71] org.apache.hadoop.ipc.Server: Stopping server on 40071
2021-02-24 15:10:40,314 INFO [IPC Server Responder] org.apache.hadoop.ipc.Server: Stopping IPC Server Responder
2021-02-24 15:10:40,314 INFO [IPC Server listener on 40071] org.apache.hadoop.ipc.Server: Stopping IPC Server listener on 40071
2021-02-24 15:10:40,317 INFO [Thread-71] org.eclipse.jetty.server.handler.ContextHandler: Stopped o.e.j.w.WebAppContext@697d6ded{/,null,UNAVAILABLE}{/mapreduce}
2021-02-24 15:10:40,320 INFO [Thread-71] org.eclipse.jetty.server.AbstractConnector: Stopped ServerConnector@251c4280{HTTP/1.1,[http/1.1]}{0.0.0.0:0}
2021-02-24 15:10:40,321 INFO [Thread-71] org.eclipse.jetty.server.handler.ContextHandler: Stopped o.e.j.s.ServletContextHandler@45bf6f39{/static,jar:file:/yarn/nm/filecache/10/3.0.0-cdh6.2.1-mr-framework.tar.gz/hadoop-yarn-common-3.0.0-cdh6.2.1.jar!/webapps/static,UNAVAILABLE}
```

```
2021-02-24 15:10:04,001 ERROR [IPC Server handler 3 on 59404] org.apache.hadoop.mapred.TaskAttemptListenerImpl: Task: attempt_1614125052521_0025_m_000000_0 - exited : java.io.IOException: com.mysql.jdbc.MysqlDataTruncation: Data truncation: Data too long for column 'from_url' at row 3
        at org.apache.sqoop.mapreduce.AsyncSqlRecordWriter.write(AsyncSqlRecordWriter.java:233)
        at org.apache.sqoop.mapreduce.AsyncSqlRecordWriter.write(AsyncSqlRecordWriter.java:46)
        at org.apache.hadoop.mapred.MapTask$NewDirectOutputCollector.write(MapTask.java:670)
        at org.apache.hadoop.mapreduce.task.TaskInputOutputContextImpl.write(TaskInputOutputContextImpl.java:89)
        at org.apache.hadoop.mapreduce.lib.map.WrappedMapper$Context.write(WrappedMapper.java:112)
        at org.apache.sqoop.mapreduce.hcat.SqoopHCatExportMapper.map(SqoopHCatExportMapper.java:56)
        at org.apache.sqoop.mapreduce.hcat.SqoopHCatExportMapper.map(SqoopHCatExportMapper.java:35)
        at org.apache.hadoop.mapreduce.Mapper.run(Mapper.java:146)
        at org.apache.sqoop.mapreduce.AutoProgressMapper.run(AutoProgressMapper.java:64)
        at org.apache.hadoop.mapred.MapTask.runNewMapper(MapTask.java:799)
        at org.apache.hadoop.mapred.MapTask.run(MapTask.java:347)
        at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:174)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1875)
        at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:168)
Caused by: com.mysql.jdbc.MysqlDataTruncation: Data truncation: Data too long for column 'from_url' at row 3   在 from_url 字段上, 数据太长了
        at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3968)
        at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3906)
        at com.mysql.jdbc.MysqlIO.sendCommand(MysqlIO.java:2524)
        at com.mysql.jdbc.MysqlIO.sqlQueryDirect(MysqlIO.java:2677)
        at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2549)
        at com.mysql.jdbc.PreparedStatement.executeInternal(PreparedStatement.java:1861)
        at com.mysql.jdbc.PreparedStatement.execute(PreparedStatement.java:1192)
        at org.apache.sqoop.mapreduce.AsyncSqlOutputFormat$AsyncSqlExecThread.run(AsyncSqlOutputFormat.java:233)
```

```
找到了，发现  from_url上数据太长，导致无法存储数据

思考如何解决呢？  将字段的允许变得更长即可 ，修改字段长度
```

sqoop导出 报错**ORC split generation failed with exception**:

java.lang.ArrayIndexOutOfBoundsException: 5

原因：hive版本是2.1.1，orc文件是由orcfilewriter用更大的版本写的

解决：临时：可以创建textFile文件格式先导入到textFile格式，再用sqoop导出即可

sqoop导出报错ERROR tool.ExportTool: Error during export: Export job failed!

原因：可能是字段类型不一样，可能是字段长度放不下

```
说明：
    对于sqoop而言，需要将导入导出命令翻译为MR进行执行的，之后sqoop只需要捕获MR最终执行结果即可，如果成功，标记为成功，如果失败，就会告知导入或者导出失败了
    具体失败的原因，需要查询MR的日志，从MR日志才能判断出为何而失败

19888查看历史日志
```

```
Error: java.io.IOException: com.mysql.jdbc.MysqlDataTruncation: Data truncation: Data too long for column 'from_url' at row 1
```

- qoop采集完成后导致HDFS数据与Oracle数据量不符
- **原因**
  - sqoop以文本格式导入数据时，默认的换行符是特殊字符
    - Sqoop遇到特殊字段就作为一行（字段值里面出现）
  - Oracle中的数据列中如果出现了\n、\r、\t等特殊字符，就会被划分为多行

解决

- 方案一：删除或者替换数据中的换行符
  - --hive-drop-import-delims：删除换行符

    - --hive-delims-replacement char：替换换行符
    - 不建议使用：侵入了原始数据
- 方案二：使用特殊文件格式：AVRO格式

spark

移动数据不如移动计算：把task发送给和数据一起的节点
用idea一定要把相对应的jar包导入到pom文件中

SQL语法如何实现分区调整： /*+repartition(1) */ ： select /*+repartition(1) */ ...

报错Unable to fetch table web_chat_ems. Invalid method name: 'get_table_req' **-- 用3.1版本的spark（内置hive版本默认是**2.3.7**）去整合CHD里的hive2.1.1版本，导致版本不兼容，看不到表，能看到数据库查询表就报错**
解决：把这两个配置写到 spark/conf/spark-defaults.conf 文件里

```
spark.sql.hive.metastore.version=2.1.1
spark.sql.hive.metastore.jars=/opt/cloudera/parcels/CDH/lib/hive/lib/*
```

1.错误：没有开启Cross Join

```
Exception in thread "main" org.apache.spark.sql.AnalysisException: Detected implicit cartesian product for INNER join between logical plans.Use the CROSS JOIN syntax to allow cartesian products between these relations

Spark2.x默认不允许执行笛卡尔积，除非显示申明cross join或者开启属性
set spark.sql.crossJoin.enabled=true
```

2.错误：Unable to move source

```
Error: org.apache.spark.sql.AnalysisException: org.apache.hadoop.hive.ql.metadata.HiveException: Unable to move source hdfs://hadoop.bigdata.cn:9000/data/dw/dws/one_make/dim_warehouse/.hive-staging_hive_2020-12-23_04-26-01_363_5663538019799519260-16/-ext-10000/part-00000-63069107-6405-4e31-a55a-6bdeefcd7d9b-c000 to destination hdfs://hadoop.bigdata.cn:9000/data/dw/dws/one_make/dim_warehouse/dt=20210101/part-00000-63069107-6405-4e31-a55a-6bdeefcd7d9b-c000; (state=,code=0)

重启SparkSQL的ThriftServer，与MetaStore构建新的会话连接
```

什么时候需要调节Executor的堆外内存大小?
当出现一下异常时：
shuffle file cannot find，executor lost、task lost，out of memory
https://www.cnblogs.com/colorchild/p/12175328.html

Spark执行任务时出现java.lang.OutOfMemoryError: GC overhead limit exceeded和java.lang.OutOfMemoryError: java heap space原因和解决方法?
答：原因：加载了太多资源到内存，本地的性能也不好，gc时间消耗的较多
解决方法：
1）增加参数，-XX:-UseGCOverheadLimit，关闭这个特性，同时增加heap大小，-Xmx1024m
2）下面这个两个参数调大点
export SPARK_EXECUTOR_MEMORY=6000M
export SPARK_DRIVER_MEMORY=7000M

IDEA

structureStreaming kafka报错： failed to find data source kafka
问题：缺少spark-sql-kafka-0-10_...依赖
解决：1.在idea添加

```
<dependency>
    <groupId>org.apache.spark</groupId>
    <artifactId>spark-sql-kafka-0-10_2.12</artifactId>
    <version>3.1.2</version>
</dependency>
```

2.如果还出现这种情况 在liunx添加相对应的jar包
如果是yarn上则上传对spark的jar包

```
hdfs dfs -put spark-sql-kafka-0-10_2.12-3.1.2.jar /spark/jars
```

如果是spark本地，则添加到spark目录下的jars目录里

yarn

问题1：程序已提交YARN，但是无法运行，报错：Application is added to the scheduler and is not activated. User’s AM resource limit exceeded.

```
yarn.scheduler.capacity.maximum-am-resource-percent=0.8
```

配置文件：${HADOOP_HOME}/etc/hadoop/capacity-scheduler.xml

- 属性功能：指定队列最大可使用的资源容量大小百分比，默认为0.2，指定越大，AM能使用的资源越多

问题2：程序提交，运行失败，报错：无法申请Container

```
yarn.scheduler.minimum-allocation-mb=512
```

- 配置文件：${HADOOP_HOME}/etc/hadoop/yarn-site.xml
- 属性功能：指定AM为每个Container申请的最小内存，默认为1G，申请不足1G，默认分配1G，值过大，会导致资源不足，程序失败，该值越小，能够运行的程序就越多

问题3：怎么提高YARN集群的并发度?

```
YARN资源配置
yarn.nodemanager.resource.cpu-vcores=8 --yarn的最大cpu核心
yarn.nodemanager.resource.memory-mb=8192 --yarn的最大内存

Container资源
yarn.scheduler.minimum-allocation-vcores=1
yarn.scheduler.maximum-allocation-vcores=32
yarn.scheduler.minimum-allocation-mb=1024
yarn.scheduler.maximum-allocation-mb=8192

MR Task资源 （2-4G之间）
mapreduce.map.cpu.vcores=1
mapreduce.map.memory.mb=1024
mapreduce.reduce.cpu.vcores=1
mapreduce.reduce.memory.mb=1024

Spark Executor资源
--driver-memory  #分配给Driver的内存，默认分配1GB
--driver-cores   #分配给Driver运行的CPU核数，默认分配1核
--executor-memory #分配给每个Executor的内存数，默认为1G，所有集群模式都通用的选项
--executor-cores  #分配给每个Executor的核心数，YARN集合和Standalone集群通用的选项
--total-executor-cores NUM  #Standalone模式下用于指定所有Executor所用的总CPU核数
--num-executors NUM #YARN模式下用于指定Executor的个数，默认启动2个
```

- 程序提交成功，但是不运行而且不报错，什么问题，怎么解决?

    - 资源问题：APPMaster就没有启动
    - 环境问题
        - NodeManager进程问题：进程存在，但不工作
        - 机器资源不足导致YARN或者HDFS服务停止：磁盘超过90%，所有服务不再工作
        - 解决：实现监控告警：80%，邮件告警
- YARN中程序运行失败的原因遇到过哪些？
    - 代码逻辑问题
    - 资源问题：Container
        - Application / Driver：管理进程
        - MapTask和ReduceTask / Executor：执行进程
    - 解决问题：配置进程给定更多的资源

Kafka

生产者序列化报错org.apache.kafka.common.serialization.ByteArraySerializer is not an instance of org.apache.kafka.common.serialization.Deserializer

原代码：

```
val producerConfigs = new util.HashMap[String, AnyRef]
ConsumerConfig.KEY_DESERIALIZER_CLASS_CONFIG
"org.apache.kafka.common.serialization.StringDeserializer",
ConsumerConfig.VALUE_DESERIALIZER_CLASS_CONFIG ->
"org.apache.kafka.common.serialization.StringDeserializer"
val produce = new KafkaProducer[String, String](producerConfigs)
```

解决：

```
方法1.
val producerConfigs = new util.HashMap[String, AnyRef]
producerConfigs.put(ProducerConfig.KEY_SERIALIZER_CLASS_CONFIG,"org.apache.k
afka.common.serialization.StringSerializer")
producerConfigs.put(ProducerConfig.VALUE_SERIALIZER_CLASS_CONFIG,"org.apache
.kafka.common.serialization.StringSerializer")
方法2.
val produce = new KafkaProducer[String, String](producerConfigs,new
StringSerializer(),new StringSerializer())
```

其他问题

1.分桶后，用insert into 插入表 比如10个桶，会分10个reduce 这时插入失败了每个reduce都失败了【设置了reduce的java的内存大小和开启了hive.optimize.sort.dynamic.partition=true（只会生成一个reduce）】

按照访问咨询看板中增加内存的设置进行配置：

1. 提高Yarn的NodeManager内存配置

修改参数yarn.nodemanager.resource.memory-mb。

2. 提高MR的内存配置

修改参数mapreduce.map.java.opts、mapreduce.reduce.java.opts、mapreduce.map.memory.mb、mapreduce.reduce.memory.mb。

2.如果分桶后产生的文件过多，后续会一个桶产生一个map，可以把分桶关了，只做普通的join即可

```
--分桶
set hive.enforce.bucketing=false;
set hive.enforce.sorting=false;
set hive.optimize.bucketmapjoin = false;
set hive.auto.convert.sortmerge.join=false;
set hive.auto.convert.sortmerge.join.noconditionaltask=false;
```