BSC-OPS数据平台手册

1 测试环境

IP:**10.226.98.58**

内存信息

> MemTotal: 24522644 kB (24G)

内存	物理CPU	CPU内核	逻辑CPU	/var (/dev/mapper/rootvg-var)	/apps (/dev/mapper/appvg-applv)	
24G	3	6 (2x3)	6	25G	100G ((正在用的))	

2. OS version

-bash-4.2$ lsb_release -a
LSB Version: :core-4.1-amd64:core-4.1-noarch:cxx-4.1-amd64:cxx-4.1-noarch:desktop-4.1-amd64:desktop-4.1-noarch:languages-4.1-amd64:languages-4.1-noarch:printing-4.1-amd64:printing-4.1-noarch
Distributor ID: CentOS
Description: CentOS Linux release 7.9.2009 (Core)
Release: 7.9.2009
Codename: Core

```
docker-ce-20.10.7

sudo yum install docker-ce-20.10.7 docker-ce-cli-20.10.7 containerd.io

```

3. Tableau Desktop

Tableau Desktop is a visual exploration and analysis application, where users connect to data, create dashboards, and publish content to Tableau Server.You have been assigned a license key for Tableau Desktop. Below are resources to get started with Tableau Desktop.Visit Tableau @BSC for learning, support, and community resources for Tableau.

Install	**Complete these steps to install Tableau Desktop:Please make sure that your laptop is connected to VPNGo to Windows button->All Apps->Microsoft Endpoint Manager ->Software Center or search Software Center in your laptopSearch for tableau (upper right corner) under Application tabSelect Tableau Desktop 2020.2.4 and click Install**
Activate	Complete these steps to activate Tableau Desktop:Open **Tableau Desktop**Select **Help > Manage Product Keys** and then click **Activate****Assigned User:** Jia, Wen Wen.Jia@bsci.com **Assigned Key:** TDQU-B21B-95F0-0377-422B**Do not share or activate this license on more than one machine.**
Learning	The following training resources are available for Tableau Desktop:· Online Help· Tableau @BSC· Training Videos· Live Training
Support	You've got questions? We've got answers!Visit Tableau Support for details on how to submit a ticket

4 Docker 配置信息

4.1 Create CONTAINER

```
docker run -itd --net=bsc-br -p 8081:8081 -p 8080:8080 -p 8089:8089 -p 8443:8443 -p 9870:9870 -p 9868:9868 -p 9864:9864 -p 8088:8088 -p 10000:10000 --name bsc-ops-01 --hostname hadoop-master -p 10028:22 bsc-ops:3

docker run -itd --net=bsc-br --name hadoop-slave1 --hostname hadoop-slave1 -p 10027:22 bsc-ops:3
docker run -itd --net=bsc-br --name hadoop-slave2 --hostname hadoop-slave2 -p 10026:22 bsc-ops:3

scp -r /opt/module/azkaban/azkaban-exec hadoop-slave1:/opt/module/azkaban/
scp -r /opt/module/azkaban/azkaban-exec hadoop-slave2:/opt/module/azkaban/

```

4.2 Configure container

```
#1 删除旧数据
cd $HADOOP_HOME/data
rm -rf dfs  #删除旧数据
#2 初始化namenode
hdfs namenode -format

#3 启动所有的节点
start-all.sh
#4 查看进程
jps
3108 SecondaryNameNode
3752 Jps
2699 NameNode
3563 NodeManager
2847 DataNode

#5 启动HIVE Service2
```

```
hiveservices.sh start
hiveservices.sh status
```

4.3 Hadoop 空间不足

```
检查总体情况
 hadoop dfsadmin -report
检查每个目录
 hdfs dfs -du -h /
[root@hadoop-master hadoop]#  hdfs dfs -du -h /
2.4 G  2.4 G  /bsc
597    597    /system
1.2 G  6.7 G  /tmp
0      0      /user
如果tmp文件过大，可以删除用下面命令：
hdfs dfs -rm -f -r /tmp
hdfs dfs -mkdir /tmp

还可以给用户设置空间配额，避免每个用户占用的空间过大
  #设置配额
  hdfs dfsadmin -setSpaceQuota 2G /user/tom
  #清除配额
  hdfs dfsadmin -clrSpaceQuota /user/tom
  #察看配额
  hdfs dfs -count -q -v  /user/tom
```

5 Hadoop 平台常用命令

- ## start-all.sh

  启动hadoop 平台的name,data,yarn 节点服务

  ```
  [root@hadoop-master hadoop]# start-all.sh
  WARNING: HADOOP_SECURE_DN_USER has been replaced by HDFS_DATANODE_SECURE_USER. Using value of HADOOP_SECURE_DN_USER.
  Starting namenodes on [hadoop-master]
  Last login: Thu Jul  8 12:48:29 CST 2021 on pts/1
  Starting datanodes
  Last login: Thu Jul  8 12:49:17 CST 2021 on pts/1
  Starting secondary namenodes [hadoop-master]
  Last login: Thu Jul  8 12:49:19 CST 2021 on pts/1
  Starting resourcemanager
  Last login: Thu Jul  8 12:49:22 CST 2021 on pts/1
  Starting nodemanagers
  Last login: Thu Jul  8 12:49:26 CST 2021 on pts/1
  ```

- ## stop-all.sh

  停止namenodes，datanodes，secondary namenodes，resourcemanager，nodemanagers等节点服务

- **hiveservices.sh start**

  启动 Metastore 服务和HiveServer2 服务

- **hiveservices.sh stop**

  停止Metastore 服务和HiveServer2 服务

- **hiveservices.sh status**

  查看Metastore 服务和HiveServer2 服务状态

  ```
  [root@hadoop-master hadoop]# hiveservices.sh start
  Metastroe 服务已启动
  HiveServer2 服务已启动
  ```

- **Azkaban 启动与关闭**

  启动顺序不能改变：Executor服务---->web服务

  - 启动与关闭Executor服务

    ```
    #启动并激活Executor 服务
    cd /opt/module/azkaban/azkaban-exec
    bin/start-exec.sh
    #激活
    curl -G "hadoop-master:12321/executor?action=activate" && echo

    #停止Executor 服务
    bin/shutdown-exec.sh
    ```

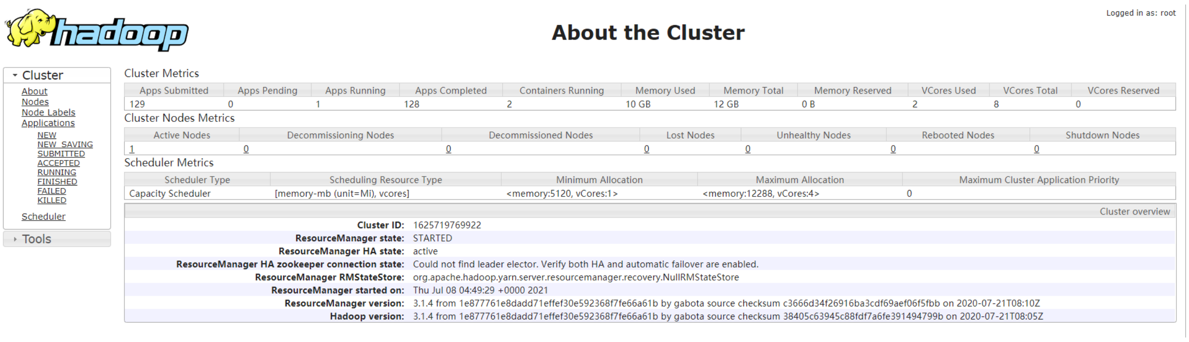
  - 启动与关闭web服务

    ```
    cd /opt/module/azkaban/azkaban-web
    bin/start-web.sh #启动
    bin/shutdown-web.sh #关闭
    ```

6 日常监控

6.1 Cluster运行状态

访问地址：http://10.226.98.58:8088/



About the Cluster

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved
129	0	1	128	2	10 GB	12 GB	0 B	2	8	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes	Shutdown Nodes
1	0	0	0	0	0	0

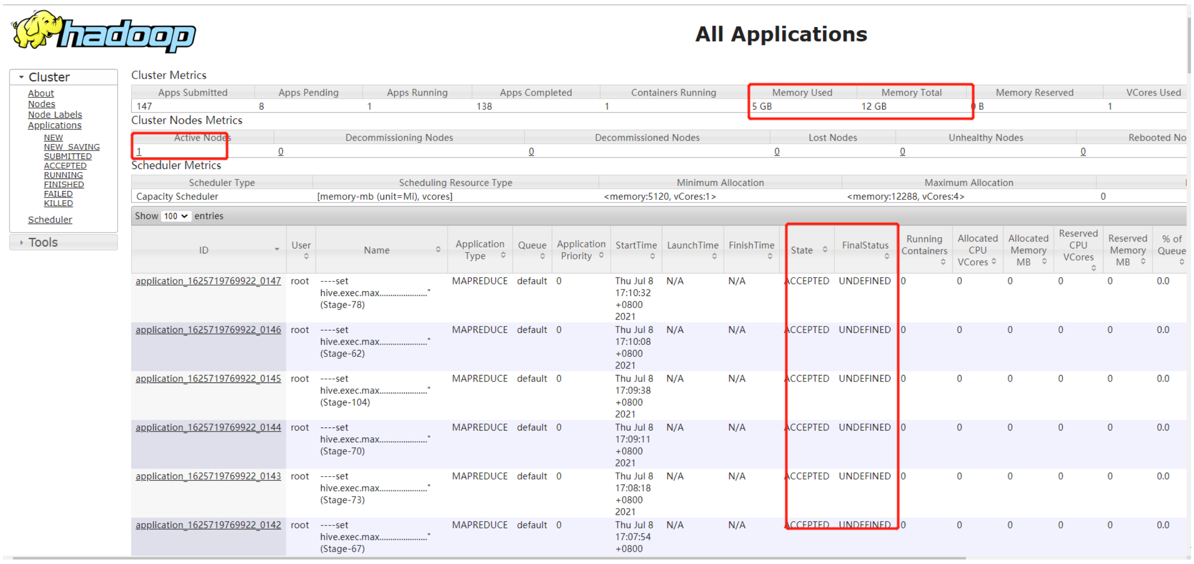
Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:5120, vCores:1>	<memory:12288, vCores:4>	0

Cluster overview

Cluster ID:	1625719769922
ResourceManager state:	STARTED
ResourceManager HA state:	active
ResourceManager HA zookeeper connection state:	Could not find leader elector. Verify both HA and automatic failover are enabled.
ResourceManager RMStateStore:	org.apache.hadoop.yarn.server.resourcemanager.recovery.NullRMStateStore
ResourceManager started on:	Thu Jul 08 04:49:29 +0000 2021
ResourceManager version:	3.1.4 from 1e877761e8dadd71effef30e592368f7fe66a61b by gabota source checksum c3666d34f26916ba3cdf69aef06f5fbb on 2020-07-21T08:10Z
Hadoop version:	3.1.4 from 1e877761e8dadd71effef30e592368f7fe66a61b by gabota source checksum 38405c63945c88fdf7a6fe391494799b on 2020-07-21T08:05Z

Applications



All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used
147	8	1	138	1	5 GB	12 GB	0 B	1

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted No
1	0	0	0	0	0

Scheduler Metrics

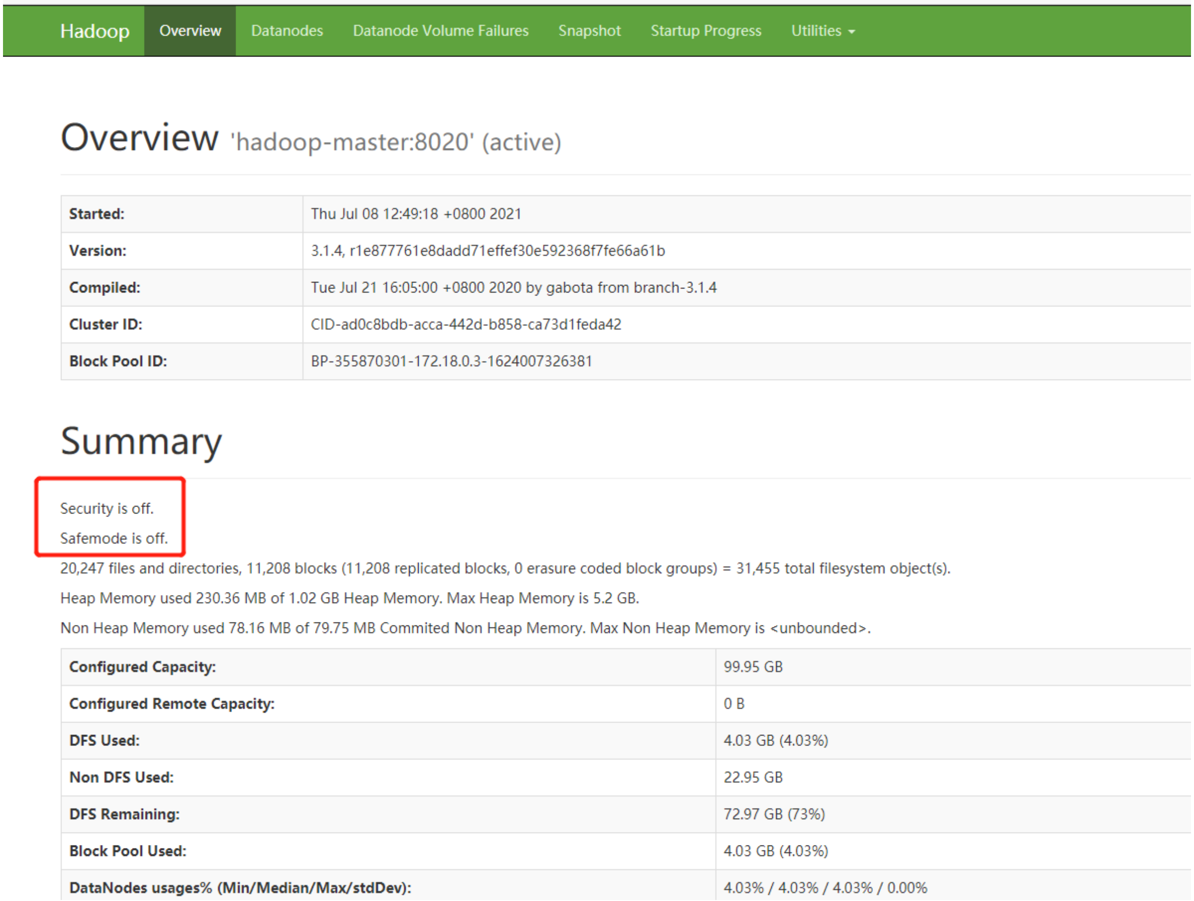
Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:5120, vCores:1>	<memory:12288, vCores:4>	0

Show 100 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores	Reserved Memory MB	% of Queue
application_1625719769922_0147	root	----set hive.exec.max......................" (Stage-78)	MAPREDUCE	default	0	Thu Jul 8 17:10:32 +0800 2021	N/A	N/A	ACCEPTED	UNDEFINED	0	0	0	0	0	0.0
application_1625719769922_0146	root	----set hive.exec.max......................" (Stage-62)	MAPREDUCE	default	0	Thu Jul 8 17:10:08 +0800 2021	N/A	N/A	ACCEPTED	UNDEFINED	0	0	0	0	0	0.0
application_1625719769922_0145	root	----set hive.exec.max......................" (Stage-104)	MAPREDUCE	default	0	Thu Jul 8 17:09:38 +0800 2021	N/A	N/A	ACCEPTED	UNDEFINED	0	0	0	0	0	0.0
application_1625719769922_0144	root	----set hive.exec.max......................" (Stage-70)	MAPREDUCE	default	0	Thu Jul 8 17:09:11 +0800 2021	N/A	N/A	ACCEPTED	UNDEFINED	0	0	0	0	0	0.0
application_1625719769922_0143	root	----set hive.exec.max......................" (Stage-73)	MAPREDUCE	default	0	Thu Jul 8 17:08:18 +0800 2021	N/A	N/A	ACCEPTED	UNDEFINED	0	0	0	0	0	0.0
application_1625719769922_0142	root	----set hive.exec.max......................" (Stage-67)	MAPREDUCE	default	0	Thu Jul 8 17:07:54 +0800	N/A	N/A	ACCEPTED	UNDEFINED	0	0	0	0	0	0.0

6.2 Namenode 状态

访问地址：http://10.226.98.58:9870/



Overview 'hadoop-master:8020' (active)

Started:	Thu Jul 08 12:49:18 +0800 2021
Version:	3.1.4, r1e877761e8dadd71effef30e592368f7fe66a61b
Compiled:	Tue Jul 21 16:05:00 +0800 2020 by gabota from branch-3.1.4
Cluster ID:	CID-ad0c8bdb-acca-442d-b858-ca73d1feda42
Block Pool ID:	BP-355870301-172.18.0.3-1624007326381

Summary

Security is off.

Safemode is off.

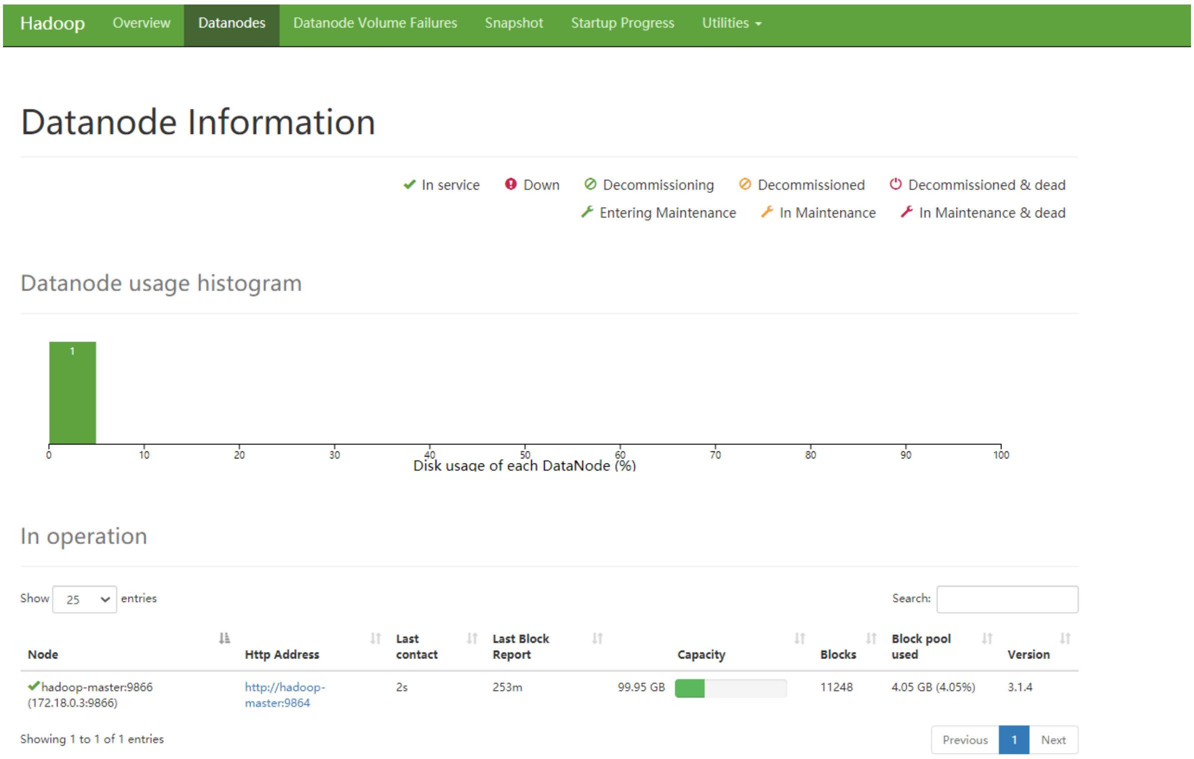
20,247 files and directories, 11,208 blocks (11,208 replicated blocks, 0 erasure coded block groups) = 31,455 total filesystem object(s).

Heap Memory used 230.36 MB of 1.02 GB Heap Memory. Max Heap Memory is 5.2 GB.

Non Heap Memory used 78.16 MB of 79.75 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	99.95 GB
Configured Remote Capacity:	0 B
DFS Used:	4.03 GB (4.03%)
Non DFS Used:	22.95 GB
DFS Remaining:	72.97 GB (73%)
Block Pool Used:	4.03 GB (4.03%)
DataNodes usages% (Min/Median/Max/stdDev):	4.03% / 4.03% / 4.03% / 0.00%

Datanodes



Datanode Information

✓ In service ● Down ⊘ Decommissioning ⊘ Decommissioned ⏻ Decommissioned & dead
Entering Maintenance In Maintenance In Maintenance & dead

Datanode usage histogram

In operation

Show 25 entries Search:

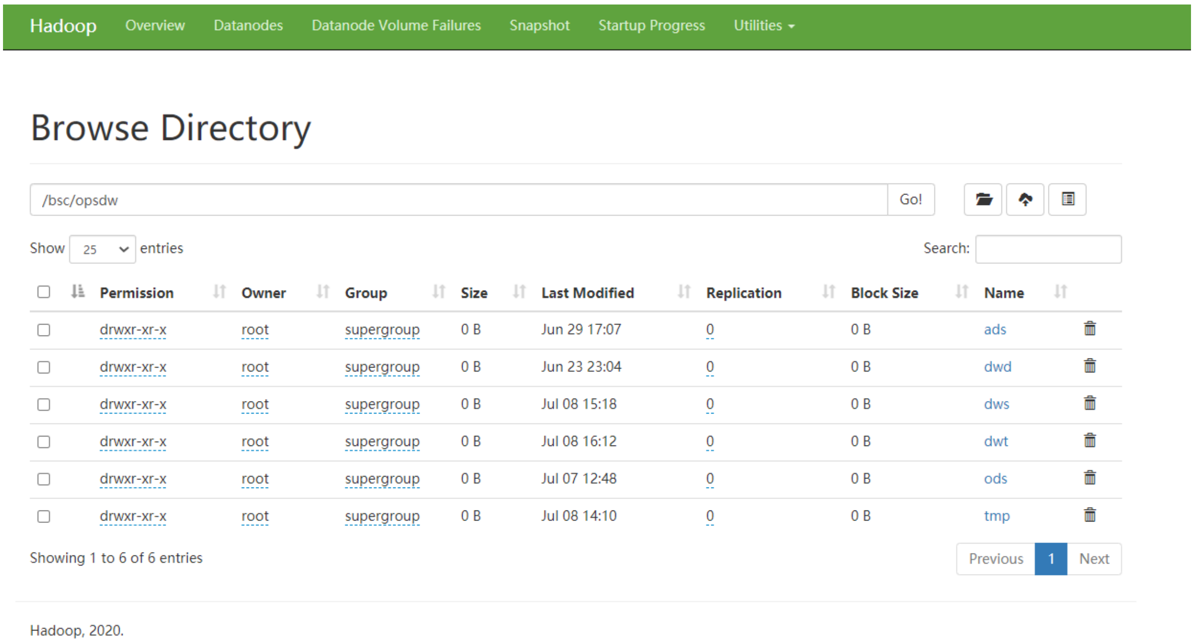
Node	Http Address	Last contact	Last Block Report	Capacity	Blocks	Block pool used	Version
✓hadoop-master:9866 (172.18.0.3:9866)	http://hadoop-master:9864	2s	253m	99.95 GB	11248	4.05 GB (4.05%)	3.1.4

Showing 1 to 1 of 1 entries

Previous 1 Next

6.3 库文件状态

地址：[Browsing HDFS]()



Browse Directory

/bsc/opsdw

Show 25 entries Search:

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	drwxr-xr-x	root	supergroup	0 B	Jun 29 17:07	0	0 B	ads
□	drwxr-xr-x	root	supergroup	0 B	Jun 23 23:04	0	0 B	dwd
□	drwxr-xr-x	root	supergroup	0 B	Jul 08 15:18	0	0 B	dws
□	drwxr-xr-x	root	supergroup	0 B	Jul 08 16:12	0	0 B	dwt
□	drwxr-xr-x	root	supergroup	0 B	Jul 07 12:48	0	0 B	ods
□	drwxr-xr-x	root	supergroup	0 B	Jul 08 14:10	0	0 B	tmp

Showing 1 to 6 of 6 entries

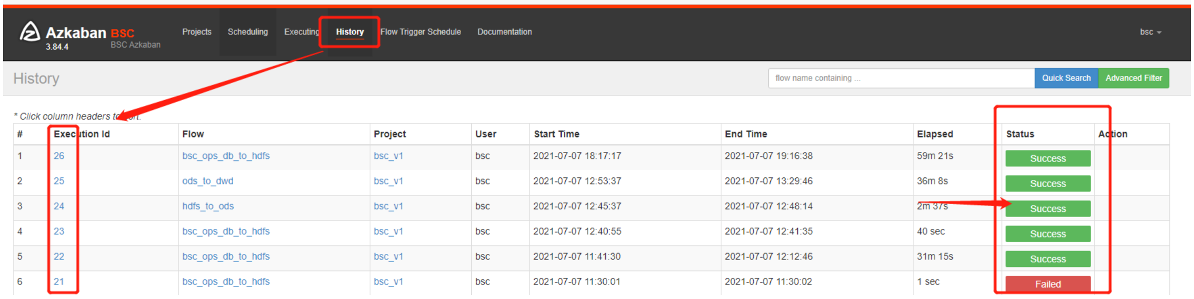
Previous 1 Next

Hadoop, 2020.

6.4 Azkaban 状态

访问地址：[Azkaban Web Client]()

历史执行记录

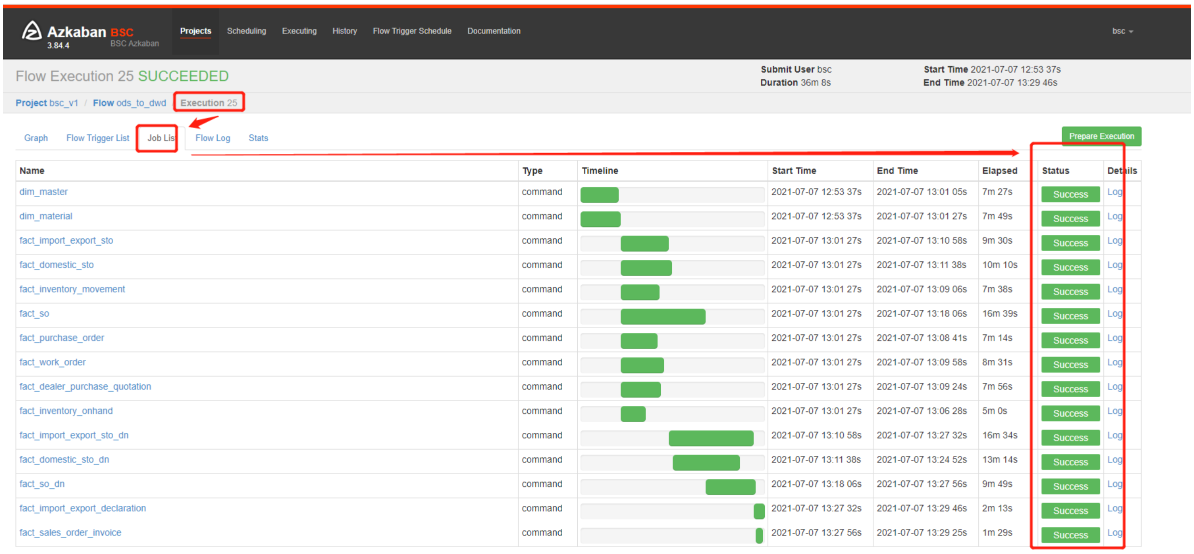


History

* Click column headers to sort.

#	Execution Id	Flow	Project	User	Start Time	End Time	Elapsed	Status	Action
1	26	bsc_ops_db_to_hdfs	bsc_v1	bsc	2021-07-07 18:17:17	2021-07-07 19:16:38	59m 21s	Success	
2	25	ods_to_dwd	bsc_v1	bsc	2021-07-07 12:53:37	2021-07-07 13:29:46	36m 8s	Success	
3	24	hdfs_to_ods	bsc_v1	bsc	2021-07-07 12:45:37	2021-07-07 12:48:14	2m 37s	Success	
4	23	bsc_ops_db_to_hdfs	bsc_v1	bsc	2021-07-07 12:40:55	2021-07-07 12:41:35	40 sec	Success	
5	22	bsc_ops_db_to_hdfs	bsc_v1	bsc	2021-07-07 11:41:30	2021-07-07 12:12:46	31m 15s	Success	
6	21	bsc_ops_db_to_hdfs	bsc_v1	bsc	2021-07-07 11:30:01	2021-07-07 11:30:02	1 sec	Failed	

Job 状态



Flow Execution 25 SUCCEEDED

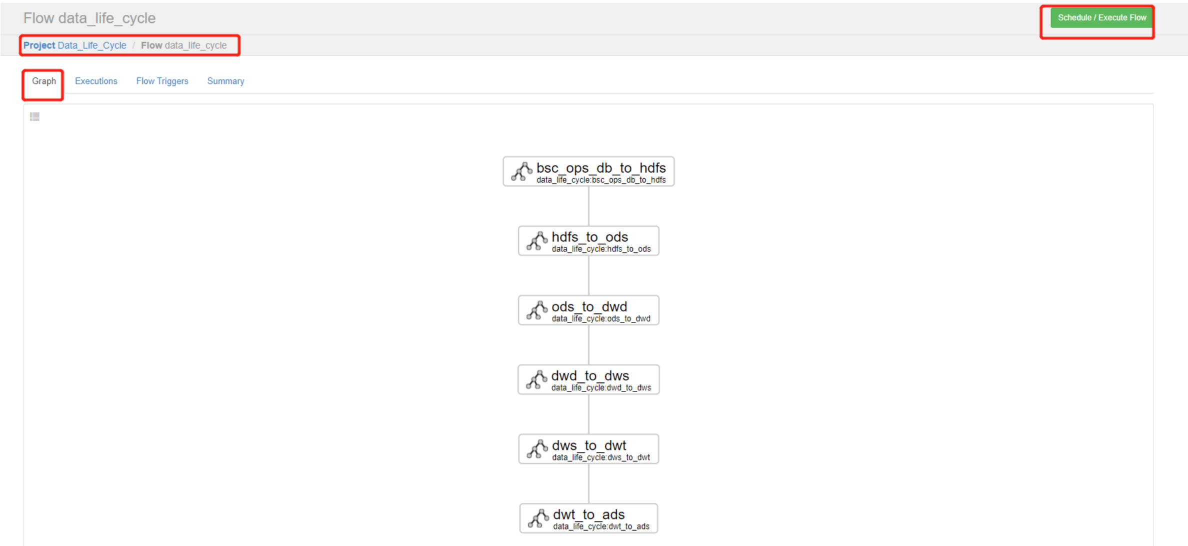
Submit User bsc
Duration 36m 8s

Start Time 2021-07-07 12:53 37s
End Time 2021-07-07 13:29 46s

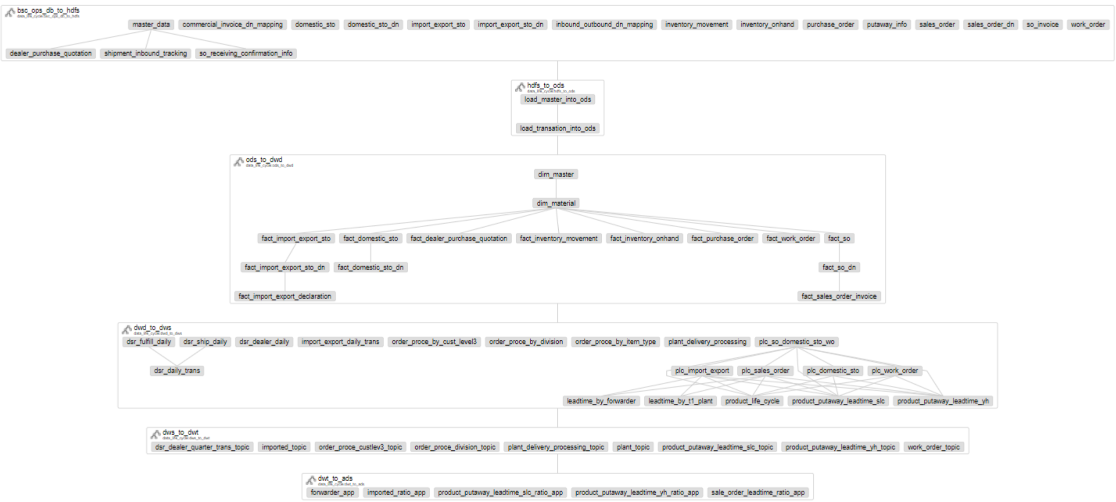
Project bsc_v1 / **Flow** ods_to_dwd / **Execution** 25

Name	Type	Timeline	Start Time	End Time	Elapsed	Status	Details
dim_master	command		2021-07-07 12:53 37s	2021-07-07 13:01 05s	7m 27s	Success	Log
dim_material	command		2021-07-07 12:53 37s	2021-07-07 13:01 27s	7m 49s	Success	Log
fact_import_export_sto	command		2021-07-07 13:01 27s	2021-07-07 13:10 58s	9m 30s	Success	Log
fact_domestic_sto	command		2021-07-07 13:01 27s	2021-07-07 13:11 38s	10m 10s	Success	Log
fact_inventory_movement	command		2021-07-07 13:01 27s	2021-07-07 13:09 06s	7m 38s	Success	Log
fact_so	command		2021-07-07 13:01 27s	2021-07-07 13:18 06s	16m 39s	Success	Log
fact_purchase_order	command		2021-07-07 13:01 27s	2021-07-07 13:08 41s	7m 14s	Success	Log
fact_work_order	command		2021-07-07 13:01 27s	2021-07-07 13:09 58s	8m 31s	Success	Log
fact_dealer_purchase_quotation	command		2021-07-07 13:01 27s	2021-07-07 13:09 24s	7m 56s	Success	Log
fact_inventory_onhand	command		2021-07-07 13:01 27s	2021-07-07 13:06 28s	5m 0s	Success	Log
fact_import_export_sto_dn	command		2021-07-07 13:10 58s	2021-07-07 13:27 32s	16m 34s	Success	Log
fact_domestic_sto_dn	command		2021-07-07 13:11 38s	2021-07-07 13:24 52s	13m 14s	Success	Log
fact_so_dn	command		2021-07-07 13:18 06s	2021-07-07 13:27 56s	9m 49s	Success	Log
fact_import_export_declaration	command		2021-07-07 13:27 32s	2021-07-07 13:29 46s	2m 13s	Success	Log
fact_sales_order_invoice	command		2021-07-07 13:27 56s	2021-07-07 13:29 25s	1m 29s	Success	Log

Data Life Cyle


Flow data_life_cycle
Schedule / Execute Flow
Project Data_Life_Cycle / Flow data_life_cycle
Graph
Executions
Flow Triggers
Summary
bsc_ops_db_to_hdfs
data_life_cycle:bsc_ops_db_to_hdfs
hdfs_to_ods
data_life_cycle:hdfs_to_ods
ods_to_dwd
data_life_cycle:ods_to_dwd
dwd_to_dws
data_life_cycle:dwd_to_dws
dws_to_dwt
data_life_cycle:dws_to_dwt
dwt_to_ads
data_life_cycle:dwt_to_ads

扩展后查看各个节点运行情况



7 环境配置过程

环境配置基于Docker + Hadoop + Hive平台

7.1 Build SSH image

Dockerfile

```
FROM centos:7
MAINTAINER donnychen(donnych@wicrenet.com)

RUN yum install -y openssh-server sudo
RUN sed -i 's/UsePAM yes/UsePAM no/g' /etc/ssh/sshd_config
RUN yum  install -y openssh-clients

RUN echo "root:1qazxsw2" | chpasswd
RUN echo "root   ALL=(ALL)       ALL" >> /etc/sudoers
RUN ssh-keygen -t dsa -f /etc/ssh/ssh_host_dsa_key
RUN ssh-keygen -t rsa -f /etc/ssh/ssh_host_rsa_key

RUN mkdir /var/run/sshd
EXPOSE 22
CMD ["/usr/sbin/sshd", "-D"]
```

Docker build

```
docker build -t="bsc-ssh" . #镜像名
```

7.2 Install JDK & Hadoop & Hive

Dockerfile

```
# container
docker run -d -p 10023:22 bsc-ssh:latest /usr/sbin/sshd -D
```

Install JDK

```
tar -zxvf jdk-8u281-linux-x64.tar.gz -C /opt/module/

vim /etc/profile.d/bsc_env.sh

#JAVA_HOME
export JAVA_HOME=/opt/module/jdk18
export PATH=$PATH:$JAVA_HOME/bin

source /etc/profile.d/bsc_env.sh
```

Install Hadoop

```
tar -zxvf hadoop-3.1.4.tar.gz -C /opt/module
mv hadoop-3.1.4/ hadoop3

vim /etc/profile.d/bsc_env.sh

#HADOOP_HOME
export HADOOP_HOME=/opt/module/hadoop3
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin

source /etc/profile.d/bsc_env.sh
```

Install Hive

```
tar -zxvf apache-hive-3.1.2-bin.tar.gz -C /opt/module
mv apache-hive-3.1.2-bin/ hive3

vim /etc/profile.d/bsc_env.sh

#HIVE_HOME
export HIVE_HOME=/opt/module/hive3
export PATH=$PATH:$HIVE_HOME/bin

source /etc/profile.d/bsc_env.sh

```

install Sqoop

```
tar -zxvf sqoop-1.4.7.tar.gz -C /opt/module
mv sqoop-1.4.7/ sqoop
vim /etc/profile.d/bsc_env.sh
#SQOOP_HOME
export SQOOP_HOME=/opt/module/sqoop
export PATH=$PATH:$SQOOP_HOME/bin

source /etc/profile.d/bsc_env.sh

sqoop list-databases --connect jdbc:mysql://172.23.128.1:3306/ --username root --password 1234567
```

7.3 Configure cluster

- **core-site.xml**

```
cd /opt/module/hadoop3/etc/hadoop/
 vim core-site.xml

 <property>
        <name>fs.defaultFS</name>
        <value>hdfs://hadoop-master:8020</value>
    </property>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>/opt/module/hadoop3/data</value>
    </property>
    <property>
        <name>io.file.buffer.size</name>
        <value>131702</value>
    </property>
    <property>
        <name>hadoop.proxyuser.root.hosts</name>
        <value>*</value>
    </property>
    <property>
        <name>hadoop.proxyuser.root.groups</name>
        <value>*</value>
    </property>
    <property>
 <name>io.compression.codecs</name>
 <value>
 org.apache.hadoop.io.compress.GzipCodec,
 org.apache.hadoop.io.compress.DefaultCodec,
 org.apache.hadoop.io.compress.BZip2Codec,
 org.apache.hadoop.io.compress.SnappyCodec,
 com.hadoop.compression.lzo.LzoCodec,
 com.hadoop.compression.lzo.LzopCodec
 </value>
 </property>
 <property>
 <name>io.compression.codec.lzo.class</name>
 <value>com.hadoop.compression.lzo.LzoCodec</value>
 </property>
```

hdfs-site.xml

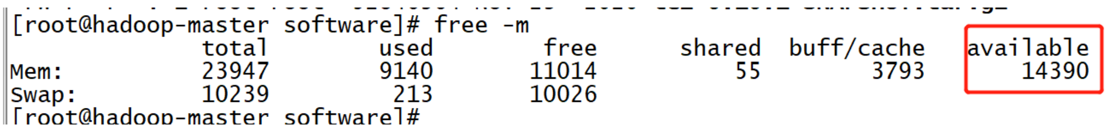
```
cd /opt/module/hadoop3/etc/hadoop
vim hdfs-site.xml
<property>
 <name>dfs.namenode.http-address</name>
 <value>hadoop-master:9870</value>
 </property>

 <property>
 <name>dfs.namenode.secondary.http-address</name>
 <value>hadoop-slave2:9868</value>
 </property>

 <property>
 <name>dfs.replication</name>
 <value>1</value>
 </property>

 <property>
   <name>dfs.webhdfs.enabled</name>
   <value>true</value>
 </property>

 <property>
    <name>dfs.client.use.datanode.hostname</name>
    <value>true</value>
    <description>Whether clients should use datanode hostnames when
      connecting to datanodes.
    </description>
</property>
```

yarn-site.xml

```
vim yarn-site.xml
 <property>
 <name>yarn.nodemanager.aux-services</name>
 <value>mapreduce_shuffle</value>
 </property>

 <property>
 <name>yarn.resourcemanager.hostname</name>
 <value>hadoop-slave1</value>
 </property>

 <property>
 <name>yarn.nodemanager.env-whitelist</name>
<value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CO
NF_DIR,CLASSPATH_PREPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_MAPRED_HOME</value>
 </property>

 <property>
 <name>yarn.scheduler.minimum-allocation-mb</name>
 <value>512</value>
 </property>
 <property>
 <name>yarn.scheduler.maximum-allocation-mb</name>
 <value>4096</value>
 </property>

 <property>
 <name>yarn.nodemanager.resource.memory-mb</name>
 <value>4096</value>
 </property>

 <property>
 <name>yarn.nodemanager.pmem-check-enabled</name>
 <value>false</value>
 </property>
```

```
<property>
<name>yarn.nodemanager.vmem-check-enabled</name>
<value>false</value>
</property>
```

内存分配时注意可以分配内存的80%：



```
[root@hadoop-master software]# free -m
              total        used        free      shared  buff/cache   available
Mem:          23947        9140       11014          55        3793       14390
Swap:         10239         213       10026
[root@hadoop-master software]#
```

mapred-site.xml

```
vim mapred-site.xml
  <property>
 <name>mapreduce.framework.name</name>
 <value>yarn</value>
 </property>

 <property>
 <name>mapreduce.jobhistory.address</name>
 <value>hadoop-master:10020</value>
</property>

 <property>
 <name>mapreduce.jobhistory.webapp.address</name>
 <value>hadoop-master:19888</value>
</property>
```

hive-site.xml

```
vim $HIVE_HOME/conf/hive-site.xml
<property>

<name>javax.jdo.option.ConnectionURL</name>

<value>jdbc:mysql://localhost:3306/hive_hdp?characterEncoding=UTF-8&createDatabaseIfNotExist=true</value>

<description>JDBC connect string for a JDBC metastore</description>

</property>
bscopsdw?characterEncoding=UTF-8&createDatabaseIfNotExist=true

<!-- 指定 hiveserver2 连接的 host -->
 <property>
 <name>hive.server2.thrift.bind.host</name>
 <value>hadoop-master</value>
 </property>
 <!-- 指定 hiveserver2 连接的端口号 -->
 <property>
 <name>hive.server2.thrift.port</name>
 <value>10000</value>
 </property
```

7.4 配置 workers

```
vim $HADOOP_HOME/etc/hadoop/workers
hadoop-master
```

7.5 NameNode格式化

```
hdfs namenode -format
```

7.6 启动HDFS

```
vi start-dfs.sh#第二行添加如下4句
vi stop-dfs.sh#第二行添加如下4句
HDFS_DATANODE_USER=root
HADOOP_SECURE_DN_USER=hdfs
HDFS_NAMENODE_USER=root
HDFS_SECONDARYNAMENODE_USER=root

vi start-yarn.sh#第二行添加如下3句
vi stop-yarn.sh#第二行添加如下3句
YARN_RESOURCEMANAGER_USER=root
HADOOP_SECURE_DN_USER=yarn
YARN_NODEMANAGER_USER=root

 start-dfs.sh
 [root@hadoop-master ~]# start-dfs.sh
WARNING: HADOOP_SECURE_DN_USER has been replaced by HDFS_DATANODE_SECURE_USER. Using value of HADOOP_SECURE_DN_USER.
Starting namenodes on [hadoop-master]
Last login: Wed Apr 14 14:52:31 UTC 2021 from gateway on pts/7
Starting datanodes
Last login: Wed Apr 14 14:53:04 UTC 2021 on pts/7
hadoop-slave1: WARNING: /opt/module/hadoop3/logs does not exist. Creating.
hadoop-slave2: WARNING: /opt/module/hadoop3/logs does not exist. Creating.
Starting secondary namenodes [hadoop-slave2]
Last login: Wed Apr 14 14:53:06 UTC 2021 on pts/7
[root@hadoop-master ~]#

[root@hadoop-slave1 sbin]# start-yarn.sh
Starting resourcemanager
Last login: Wed Apr 14 14:36:27 UTC 2021 from gateway on pts/7
Starting nodemanagers
Last login: Wed Apr 14 14:56:49 UTC 2021 on pts/7
hadoop-slave1: Warning: Permanently added 'hadoop-slave1,172.18.0.2' (RSA) to the list of known hosts.
hadoop-slave1: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
[root@hadoop-slave1 sbin]#

```

7.7 Hive 初始化

```
vim $HIVE_HOME/conf/hive-site.xml
schematool -initSchema -dbType mysql --verbose
jdbc:mysql://172.23.128.1:3306/bscopsdw?characterEncoding=UTF-8&amp;createDatabaseIfNotExist=true&amp;useSSL=false
```

7.8 hiveservices shell script

```
vim $HIVE_HOME/bin/hiveservices.sh

#!/bin/bash
HIVE_LOG_DIR=$HIVE_HOME/logs
if [ ! -d $HIVE_LOG_DIR ]
then
mkdir -p $HIVE_LOG_DIR
fi
#检查进程是否运行正常，参数 1 为进程名，参数 2 为进程端口
function check_process()
{
 pid=$(ps -ef 2>/dev/null | grep -v grep | grep -i $1 | awk '{print
$2}')
 ppid=$(netstat -nltp 2>/dev/null | grep $2 | awk '{print $7}' | cut -
d '/' -f 1)
 echo $pid
 [[ "$pid" =~ "$ppid" ]] && [ "$ppid" ] && return 0 || return 1
}
function hive_start()
{
 metapid=$(check_process HiveMetastore 9083)
 cmd="nohup hive --service metastore >$HIVE_LOG_DIR/metastore.log 2>&1
&"
 [ -z "$metapid" ] && eval $cmd || echo "Metastroe 服务已启动"
 server2pid=$(check_process HiveServer2 10000)
 cmd="nohup hiveserver2 >$HIVE_LOG_DIR/hiveServer2.log 2>&1 &"
 [ -z "$server2pid" ] && eval $cmd || echo "HiveServer2 服务已启动"
}
function hive_stop()
{
metapid=$(check_process HiveMetastore 9083)
 [ "$metapid" ] && kill $metapid || echo "Metastore 服务未启动"
 server2pid=$(check_process HiveServer2 10000)
 [ "$server2pid" ] && kill $server2pid || echo "HiveServer2 服务未启动"
}
case $1 in
"start")
 hive_start
 ;;
"stop")
 hive_stop
 ;;
"restart")
 hive_stop
 sleep 2
 hive_start
 ;;
```

```
"status")
 check_process HiveMetastore 9083 >/dev/null && echo "Metastore 服务运行
正常" || echo "Metastore 服务运行异常"
 check_process HiveServer2 10000 >/dev/null && echo "HiveServer2 服务运
行正常" || echo "HiveServer2 服务运行异常"
 ;;
*)
 echo Invalid Args!
 echo 'Usage: '$(basename $0)' start|stop|restart|status'
 ;;
esac
```

beeline测试连接：

```
chmod 777 $HIVE_HOME/bin/hiveservices.sh
hiveservices.sh start

beeline -u jdbc:hive2://hadoop-master:10000 -n root
```

7.9 相关软件列表

存放服务器地址： /opt/software

```
[root@hadoop-master software]# pwd
/opt/software
[root@hadoop-master software]# ll
total 1344448
-rw-r--r--. 1 root root 312850286 Nov 23  2020 apache-hive-3.1.2-bin.tar.gz
-rw-r--r--. 1 root root  62945274 Apr 20 12:57 apache-tez-0.9.2-bin.tar.gz
-rw-r--r--. 1 root root      6433 Mar 31 10:10 azkaban-db-3.84.4.tar.gz
-rw-r--r--. 1 root root  16175002 Apr  1 09:19 azkaban-exec-server-3.84.4.tar.gz
-rw-r--r--. 1 root root  20239974 Apr  1 09:19 azkaban-web-server-3.84.4.tar.gz
-rw-r--r--. 1 root root 348326890 Apr  7 23:10 hadoop-3.1.4.tar.gz
-rw-r--r--. 1 root root 143722924 Apr  7 22:31 jdk-8u281-linux-x64.tar.gz
-rw-r--r--. 1 root root   1006904 Apr 20  2020 mysql-connector-java-5.1.49.jar
-rw-r--r--. 1 root root 224453229 Apr  9 22:45 spark-3.0.0-bin-hadoop3.2.tgz
-rw-r--r--. 1 root root 156791324 Apr  9 22:45 spark-3.0.0-bin-without-hadoop.tgz
-rw-r--r--. 1 root root  17953604 Apr 19 14:00 sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz
-rw-r--r--. 1 root root   1152112 Apr 19 13:18 sqoop-1.4.7.tar.gz
-rw-r--r--. 1 root root  18214958 Nov 23  2020 tez-0.10.1-SNAPSHOT-minimal.tar.gz
-rw-r--r--. 1 root root  52846364 Nov 23  2020 tez-0.10.1-SNAPSHOT.tar.gz
```

Apache框架版本	
软件	版本
Hadoop	3.1.4
Hive	3.1.2
Sqoop	1.4.6
Java	1.8
azkaban	3.84.4

8 环境迁移流程

8.1 安装Docker & MySQL

安装过程请参考官方文档 [Get Started with Docker | Docker]

8.2 初始化Docker网络环境

```
docker network create --driver bridge bsc-br
```

8.3 导入镜像

```
cd '镜像tar文件所在目录'
docker load -i bsc-ops-dw.tar
```

8.4 启动容器及初始化

```
docker run -itd --net=bsc-br -p 9870:9870 -p 9868:9868 -p 9864:9864 -p 8088:8088 -p 10000:10000 -p 50070:50070 -p 10024:22  --name bsc-dev --hostname hadoop-master  bsc-ops:3

cd $HADOOP_HOME/data
rm -rf dfs  #删除旧数据
hdfs namenode -format #初始化namenode
start-all.sh #启动所有的节点
jps
[root@hadoop-master data]# jps
3108 SecondaryNameNode
3752 Jps
2699 NameNode
3563 NodeManager
2847 DataNode
```

8.5 修改MySql 连接地址

```
vim $HIVE_HOME/conf/hive-site.xml
# 修改mysql地址
jdbc:mysql://172.23.128.1:3306/bscopsdw?characterEncoding=UTF-8&amp;createDatabaseIfNotExist=true&amp;useSSL=false
```

8.6 初始化Hive 元数据库

```
schematool -initSchema -dbType mysql --verbose
```

8.7 启动 元数据服务和HiveServer2 服务

```
hiveservices.sh start
```

9 Shell Script

9.1 HDFS

ID	Name	Server Path	Comments
#1	bsc_ops_db_to_hdfs.sh	/bscflow/hdfs/bsc_ops_db_to_hdfs.sh	同步主数据和交易记录

9.2 ODS

ID	Name	Server Path	ODS Table	Comments
#1	hdfs_to_ods_master.sh	/bscflow/ods/	exchange_rate,IDD, calendar,location, plant,material,batch,customer,division	同步主数据相关ODS layer
#2	hdfs_to_ods_trans.sh	/bscflow/ods/	so,po, import sto, domatic sto, wo	同步相关业务数据至ODS layer, depends on:#1

9.3 DWD

Server Path: /bscflow/dwd/

ID	Name	DWD Table	Comments
#1	ods_to_dwd_master.sh	dwd_dim_plant, dwd_dim_locaiton, dwd_dim_batch, dwd_dim_calendar, dwd_dim_exchange_rate, dwd_dim_division, dwd_dim_customer	sync dimision table
#2	ods_to_dwd_dim_material.sh	dwd_dim_material	sync sku info, depends on #1
#3	ods_to_dwd_import_export_sto.sh	dwd_fact_import_export_sto	depends on #2
#4	ods_to_dwd_import_export_sto_dn.sh	dwd_fact_import_export_dn_detail	depends on #3
#5	ods_to_dwd_fact_import_export_declaration.sh	dwd_fact_import_export_declaration_info	depends on #4
#6	ods_to_dwd_fact_domestic_sto_info.sh	dwd_fact_domestic_sto_info	depends on #2
#7	ods_to_dwd_fact_domestic_sto_dn.sh	dwd_fact_domestic_sto_dn_info	depends on #6
#8	ods_to_dwd_fact_so.sh	dwd_fact_sales_order_info	depends on #2
#9	ods_to_dwd_fact_so_dn.sh	dwd_fact_sales_order_dn_info, dwd_fact_sales_order_dn_detail	depends on #8
#10	ods_to_dwd_fact_sales_order_invoice.sh	dwd_fact_sales_order_invoice	depends on #9
#11	ods_to_dwd_fact_work_order.sh	dwd_fact_work_order	depends on #2
#12	ods_to_dwd_fact_dealer_purchase_quotation.sh	dwd_fact_dealer_purchase_quotation	depends on #2
#13	ods_to_dwd_fact_inventory_movement.sh	dwd_fact_inventory_movement_trans	depends on #2
#14	ods_to_dwd_fact_inventory_onhand.sh	dwd_fact_inventory_onhand	depends on #2
#15	ods_to_dwd_fact_purchase_order.sh	dwd_fact_purchase_order_info	depends on #2

9.4 DWS

9.4.1 DSR

> Server Path: /bscflow/dws/

ID	Name	DWS Table	Comments
#1	dwd_to_dws_dsr_daily_trans.sh	dws_dsr_daily_trans	Depends On: #2, #3
#2	dwd_to_dws_dsr_fulfill_daily_trans.sh	dws_dsr_fulfill_daily_trans	
#3	dwd_to_dws_dsr_ship_daily_trans.sh	dws_dsr_ship_daily_trans	
#4	dwd_to_dws_dsr_dealer_daily_transation.sh	dws_dsr_dealer_daily_transation	

9.4.2 Lead Time

ID	Name	DWS Table	Comments
#1	dwd_to_dws_plc_so_sto_wo_daily_trans_d835, dwd_to_dws_plc_so_sto_wo_daily_trans_d838	dws_so_sto_wo_daily_trans	
#2	dwd_to_dws_plc_wo_daily_trans.sh	dws_plc_wo_daily_trans	Depends On1⃞
#3	dwd_to_dws_plc_so_daily_trans.sh	dws_plc_so_daily_trans	Depends On1⃞
#4	dwd_to_dws_plc_import_export_daily_trans.sh	dws_plc_import_export_daily_trans	Depends On1⃞
#5	dwd_to_dws_plc_domestic_sto_daily_trans.sh	dws_plc_domestic_sto_daily_trans	Depends On1⃞
#6	dwd_to_dws_import_export_daily_trans.sh	dws_import_export_daily_trans	Depends on #1,4
#7	dwd_to_dws_product_putaway_leadtime_slc_daily_trans.sh	dws_product_putaway_leadtime_slc_daily_trans	Depends On1⃞#1,2,3,4,5
#8	dwd_to_dws_product_putaway_leadtime_yh_daily_trans.sh	dws_product_putaway_leadtime_yh_daily_trans	Depends On1⃞#1,2,3,4,5
#9	dwd_to_dws_order_proce_custlev3_daily_trans.sh	dws_order_proce_custlev3_daily_trans	
#10	dwd_to_dws_order_proce_division_daily_trans.sh	dws_order_proce_division_daily_trans	
#11	dwd_to_dws_order_proce_tob_daily_trans.sh	dws_order_proce_tob_daily_trans	
#12	dwd_to_dws_plant_delivery_processing_daily_trans.sh	dws_plant_delivery_processing_daily_trans	
#13	dwd_to_dws_t1_plant_trans.sh	dws_t1_plant_daily_transation	Depends on: #3, #4
#14	dwd_to_dws_forwarder_daily_trans.sh	dws_forwarder_daily_trans	Depends on: #3, #4
#15	dwd_to_dws_lifecycle_leadtime_slc_daily_trans.sh	dws_lifecycle_leadtime_SLC_daily_trans	Depends on: #2,#3, #4
#16	dwd_to_dws_lifecycle_leadtime_yh_daily_trans.sh	dws_lifecycle_leadtime_YH_daily_trans	Depends on: #2,#3, #4,#5
#17	dwd_to_dws_sale_order_leadtime_daily_trans.sh	dws_sale_order_leadtime_daily_trans	

9.5 DWT

> Server Path: /bscflow/dwt

9.5.1 DSR

ID	Name	DWT Table	Comments
#1	dws_to_dwt_dsr_dealer_quarter_trans.sh	dwt_dsr_dealer_quarter_trans	

9.5.1 LeadTime

ID	Name	DWT Table	Comments
#1	dws_to_dwt_forwarder_topic.sh	dwt_forwarder_topic	
#2	dws_to_dwt_imported_topic.sh	dwt_imported_topic	
#3	dws_to_dwt_order_proce_custlev3_topic.sh	dwt_order_proce_custlev3_topic	
#4	dws_to_dwt_order_proce_division_topic.sh	dwt_order_proce_division_topic	
#5	dws_to_dwt_order_proce_tob_topic.sh	dwt_order_proce_tob_topic	
#6	dws_to_dwt_plant_delivery_processing_topic.sh	dwt_plant_delivery_processing_topic	
#7	dws_to_dwt_plant_topic.sh	dwt_plant_topic	
#8	dws_dwt_product_putaway_leadtime_slc_topic.sh	dwt_product_putaway_leadtime_slc_topic	
#9	dws_to_dwt_product_putaway_leadtime_yh_topic.sh	dwt_product_putaway_leadtime_yh_topic	
#10	dws_to_dwt_sale_order_leadtime_topic.sh	dwt_sale_order_leadtime_topic	
#11	dws_to_dwt_lifecycle_slcyh_summarize_topic.sh	dwt_lifecycle_leadtime_slcyh_summarize_topic	depends on #12, #13
#12	dws_to_dwt_lifecycle_slc_summarize_topic.sh	dwt_lifecycle_leadtime_slc_summarize_topic	depends on #15
#13	dws_to_dwt_lifecycle_yh_summarize_topic.sh	dwt_lifecycle_leadtime_yh_summarize_topic	depends on #14
#14	dws_to_dwt_lifecycle_leadtime_yh_topic.sh	dwt_lifecycle_leadtime_YH_topic	
#15	dws_to_dwt_lifecycle_leadtime_slc_topic.sh	dwt_lifecycle_leadtime_SLC_topic	
#16	dws_to_dwt_lifecycle_leadtime_division_slcyh.sh	dwt_lifecycle_leadtime_division_slcyh_topic	

9.6 ADS

> Server Path: /bscflow/ads

ID	Name	ADS Table	DWT Table	DWS Table
#1	ads_imported_ratio.sh	ads_imported_ratio		
#2	ads_product_putaway_leadtime_slc_ratio.sh	ads_product_putaway_leadtime_slc_ratio		
#3	ads_product_putaway_leadtime_yh_ratio.sh	ads_product_putaway_leadtime_yh_ratio		
#4	ads_sale_order_leadtime_ratio.sh	ads_sale_order_leadtime_ratio		
#5	ads_lifecycle_leadtime_slcyh_ratio.sh	ads_lifecycle_leadtime_slcyh_ratio		

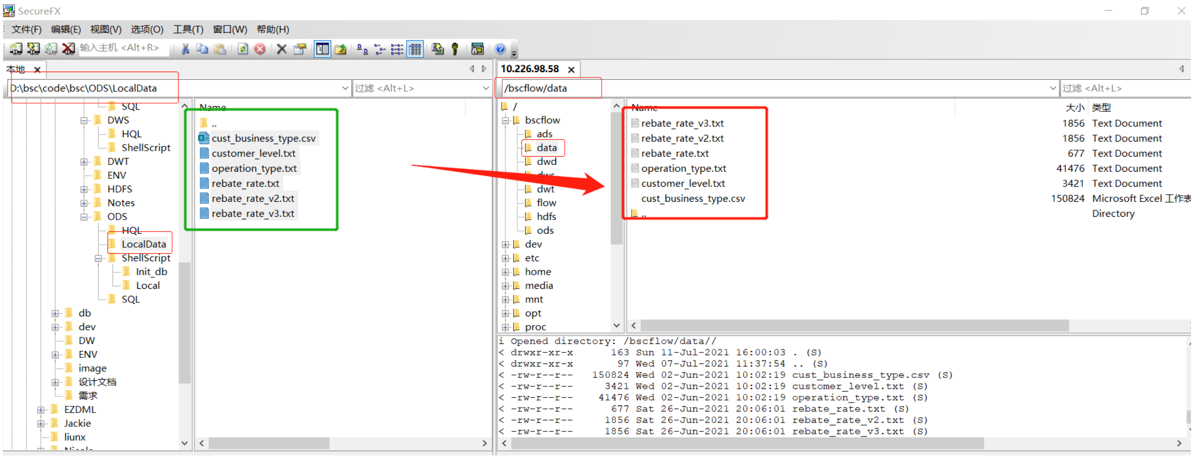
9.7 LocalData

> Server Path:/bscflow/ods/load_local_data_to_ods_dwd.sh
>
> /bscflow/ods/wo_qr_local_to_ods.sh
>
> /bscflow/dwd/ods_to_dwd_wo_qr.sh

Work Oder QR code:

a) Upload data file to server

from bsc\ODS\LocalData to /bscflow/data



D:\bsc\code\bsc\ODS\LocalData

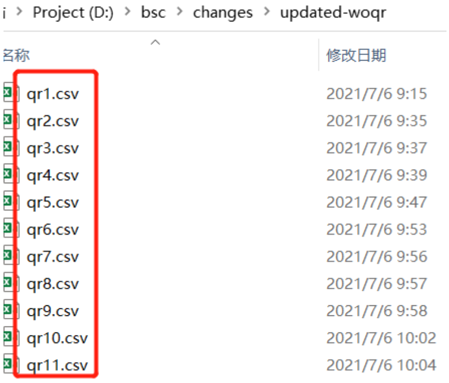
Name
..
cust_business_type.csv
customer_level.txt
operation_type.txt
rebate_rate.txt
rebate_rate_v2.txt
rebate_rate_v3.txt

/bscflow/data

Name	大小	类型
rebate_rate_v3.txt	1856	Text Document
rebate_rate_v2.txt	1856	Text Document
rebate_rate.txt	677	Text Document
operation_type.txt	41476	Text Document
customer_level.txt	3421	Text Document
cust_business_type.csv	150824	Microsoft Excel 工作表
..		Directory

```
i Opened directory: /bscflow/data//
< drwxr-xr-x      163 Sun 11-Jul-2021 16:00:03 . (S)
< drwxr-xr-x       97 Wed 07-Jul-2021 11:37:54 .. (S)
< -rw-r--r--   150824 Wed 02-Jun-2021 10:02:19 cust_business_type.csv (S)
< -rw-r--r--     3421 Wed 02-Jun-2021 10:02:19 customer_level.txt (S)
< -rw-r--r--    41476 Wed 02-Jun-2021 10:02:19 operation_type.txt (S)
< -rw-r--r--      677 Sat 26-Jun-2021 20:06:01 rebate_rate.txt (S)
< -rw-r--r--     1856 Sat 26-Jun-2021 20:06:01 rebate_rate_v2.txt (S)
< -rw-r--r--     1856 Sat 26-Jun-2021 20:06:01 rebate_rate_v3.txt (S)
```

b) 修改文件的编号，使其连续。

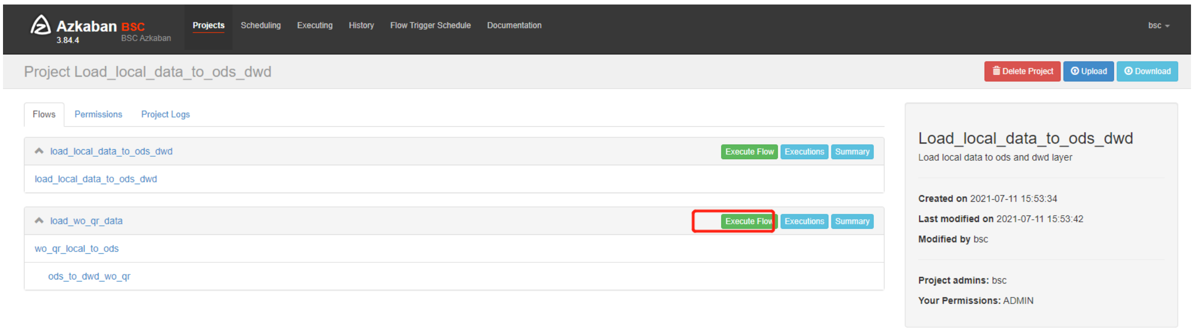


i › Project (D:) › bsc › changes › updated-woqr

名称	修改日期
qr1.csv	2021/7/6 9:15
qr2.csv	2021/7/6 9:35
qr3.csv	2021/7/6 9:37
qr4.csv	2021/7/6 9:39
qr5.csv	2021/7/6 9:47
qr6.csv	2021/7/6 9:53
qr7.csv	2021/7/6 9:56
qr8.csv	2021/7/6 9:57
qr9.csv	2021/7/6 9:58
qr10.csv	2021/7/6 10:02
qr11.csv	2021/7/6 10:04

c) 执行Job

选择 Load wo_qr_data flow, 点击Execute Flow



Project Load_local_data_to_ods_dwd

load_local_data_to_ods_dwd

load_local_data_to_ods_dwd

load_wo_qr_data

wo_qr_local_to_ods

ods_to_dwd_wo_qr

Load_local_data_to_ods_dwd

Load local data to ods and dwd layer

Created on 2021-07-11 15:53:34

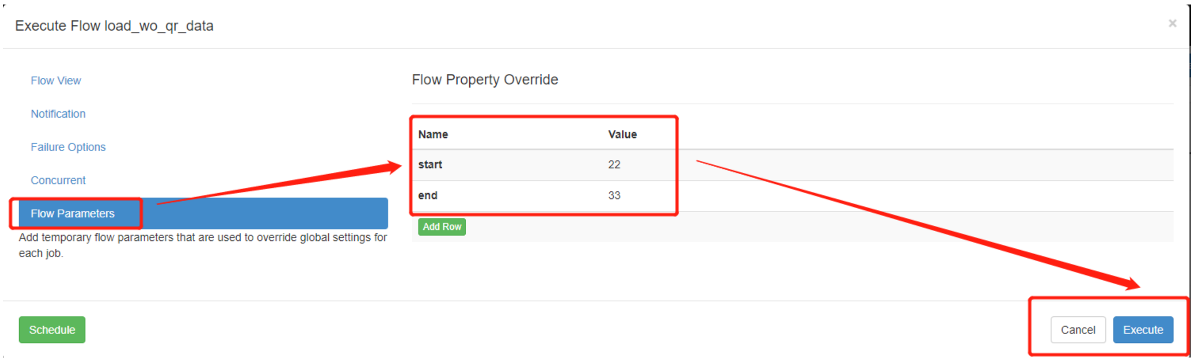
Last modified on 2021-07-11 15:53:42

Modified by bsc

Project admins: bsc

Your Permissions: ADMIN

```
sh /bscflow/ods/wo_qr_local_to_ods.sh ${start} ${end}
```



Execute Flow load_wo_qr_data

Add temporary flow parameters that are used to override global settings for each job.

Flow Property Override

Name	Value
start	22
end	33

d） 设置Flow parameters：start, end

单击Execute

cust_level, cust_type, operation_type, rebate_rate:

```
sh /bscflow/ods/load_local_data_to_ods_dwd.sh ${data_type} ${date_file}
```

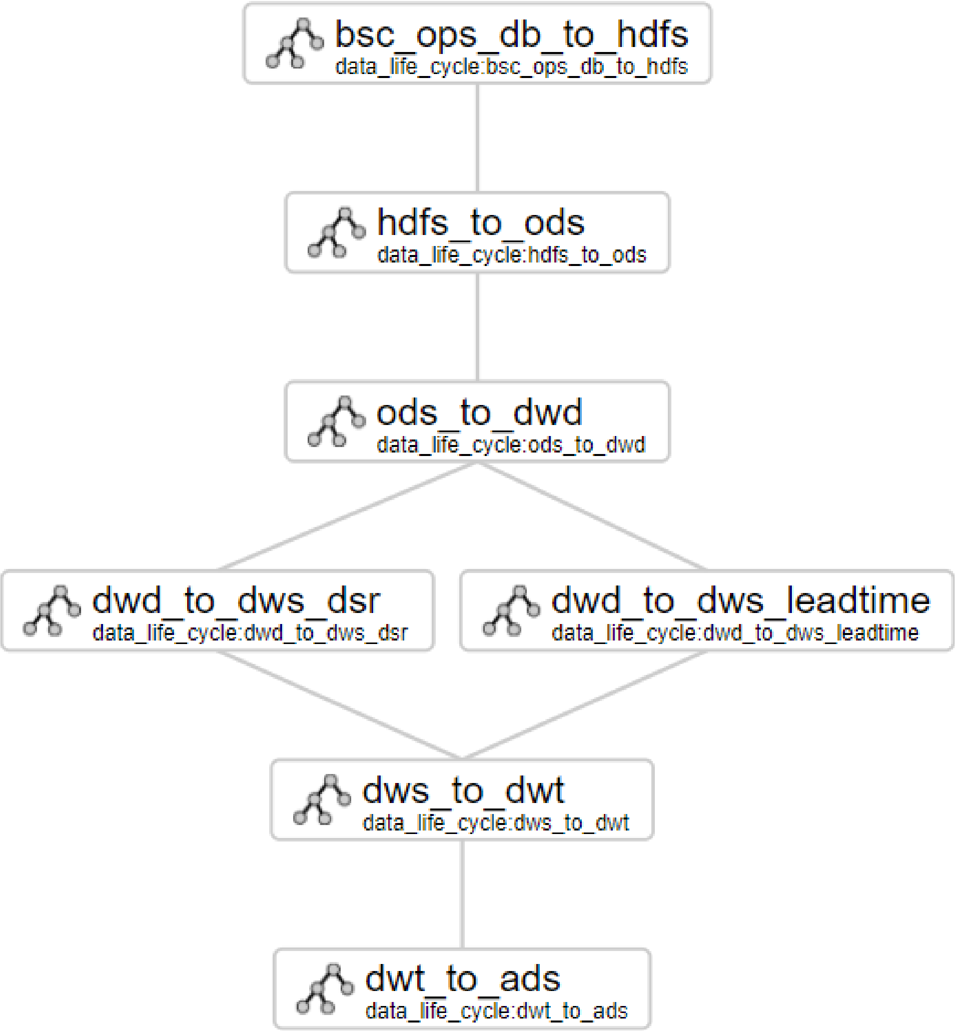
需要设置两个参数:

data_type: 取值范围cust_level|cust_type|operation_type|rebate_rate

data_file: 文件路径 例如：/bscflow/data/customer_level.txt

10 Azkaban

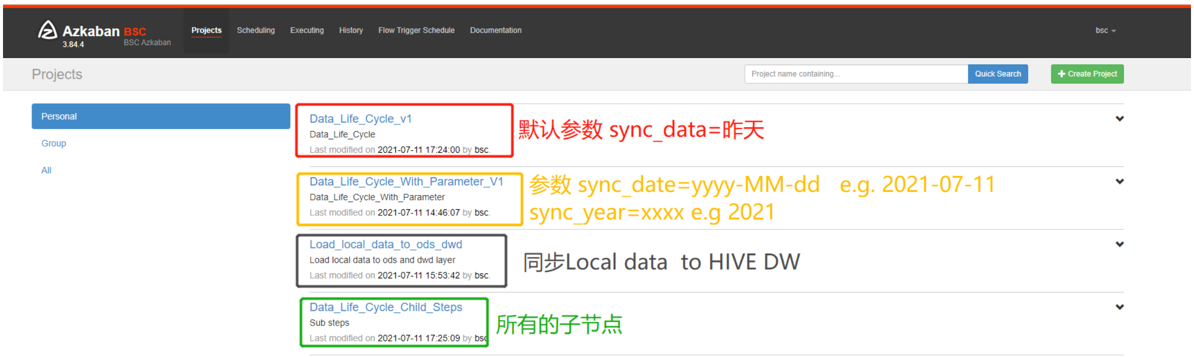
调度流程图如下:


bsc_ops_db_to_hdfs
data_life_cycle:bsc_ops_db_to_hdfs
hdfs_to_ods
data_life_cycle:hdfs_to_ods
ods_to_dwd
data_life_cycle:ods_to_dwd
dwd_to_dws_dsr
data_life_cycle:dwd_to_dws_dsr
dwd_to_dws_leadtime
data_life_cycle:dwd_to_dws_leadtime
dws_to_dwt
data_life_cycle:dws_to_dwt
dwt_to_ads
data_life_cycle:dwt_to_ads

10.1 手工调度过程

Step 1: Go to [Azkaban Web Client]()

Step 2: 选择对应的Project：



Data_Life_Cycle_v1
Data_Life_Cycle
Last modified on 2021-07-11 17:24:00 by bsc.

默认参数 sync_data=昨天

Data_Life_Cycle_With_Parameter_V1
Data_Life_Cycle_With_Parameter
Last modified on 2021-07-11 14:46:07 by bsc.

参数 sync_date=yyyy-MM-dd e.g. 2021-07-11
sync_year=xxxx e.g 2021

Load_local_data_to_ods_dwd
Load local data to ods and dwd layer
Last modified on 2021-07-11 15:53:42 by bsc.

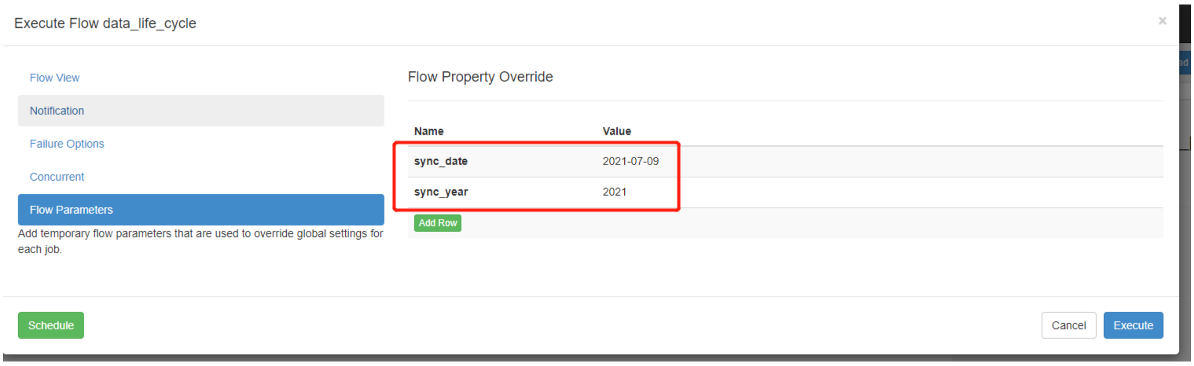
同步Local data to HIVE DW

Data_Life_Cycle_Child_Steps
Sub steps
Last modified on 2021-07-11 17:25:09 by bsc

所有的子节点

Step 3:根据Job特点设置参数

以参数job为例：



Execute Flow data_life_cycle

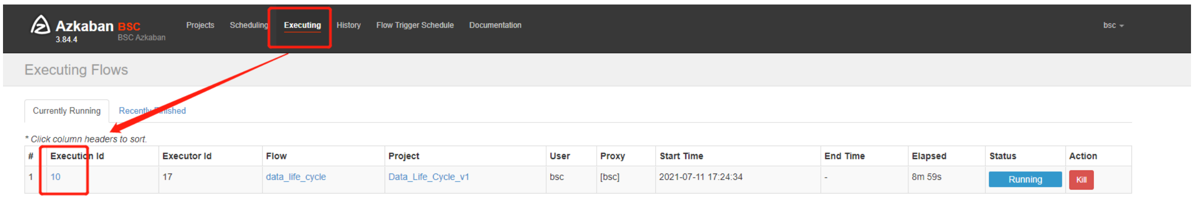
Flow Property Override

Name	Value
sync_date	2021-07-09
sync_year	2021

Add temporary flow parameters that are used to override global settings for each job.

Step 4: 点击执行

Step 5: Go to Executing

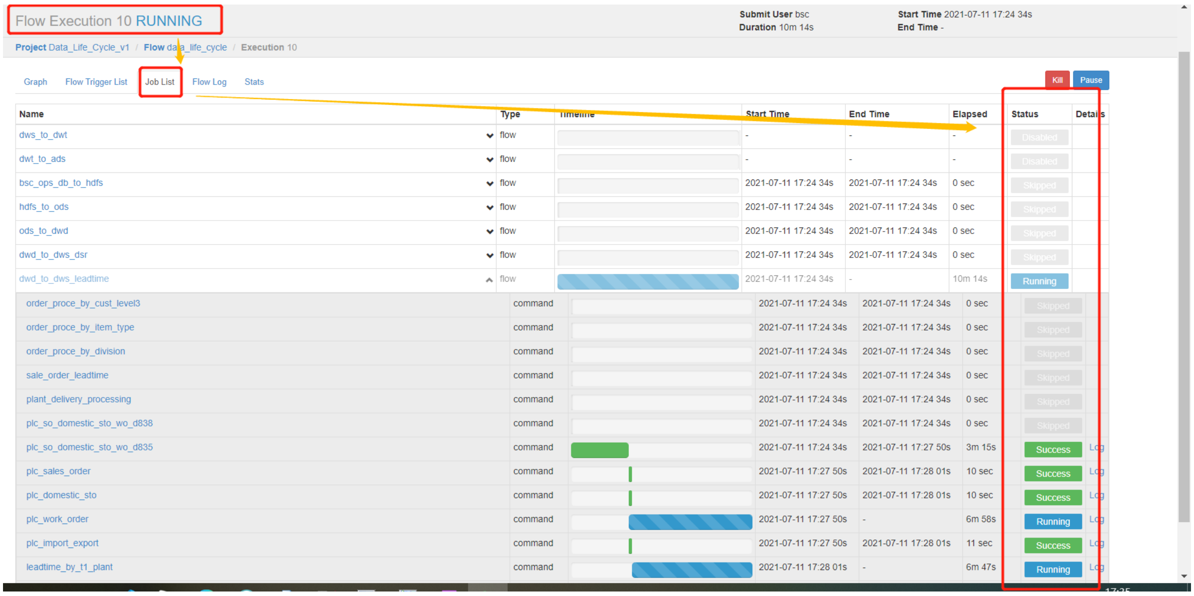


* Click column headers to sort.

#	Execution Id	Executor Id	Flow	Project	User	Proxy	Start Time	End Time	Elapsed	Status	Action
1	10	17	data_life_cycle	Data_Life_Cycle_v1	bsc	[bsc]	2021-07-11 17:24:34	-	8m 59s	Running	Kill

Step 6: 点击 Execution Id -> Job List

查看子每个子job 状态



Flow Execution 10 RUNNING

Submit User bsc
Duration 10m 14s

Start Time 2021-07-11 17:24 34s
End Time -

Project Data_Life_Cycle_v1 / **Flow** data_life_cycle / **Execution** 10

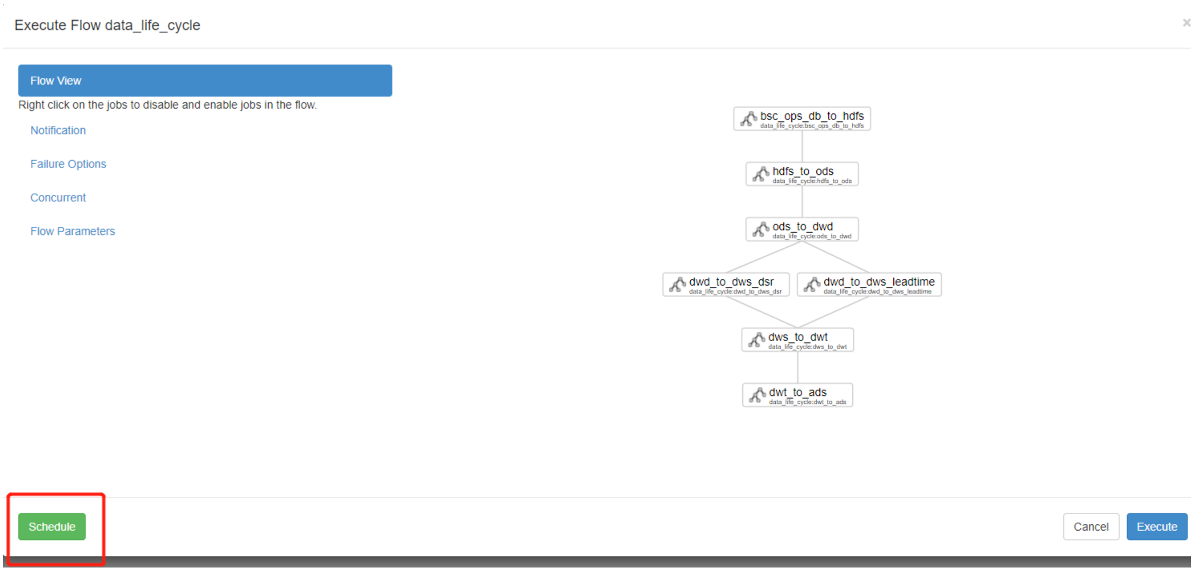
Name	Type	Timeline	Start Time	End Time	Elapsed	Status	Details
dws_to_dwt	flow		-	-	-	Disabled	
dwt_to_ads	flow		-	-	-	Disabled	
bsc_ops_db_to_hdfs	flow		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
hdfs_to_ods	flow		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
ods_to_dwd	flow		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
dwd_to_dws_dsr	flow		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
dwd_to_dws_leadtime	flow		2021-07-11 17:24 34s	-	10m 14s	Running	
order_proce_by_cust_level3	command		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
order_proce_by_item_type	command		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
order_proce_by_division	command		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
sale_order_leadtime	command		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
plant_delivery_processing	command		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
plc_so_domestic_sto_wo_d838	command		2021-07-11 17:24 34s	2021-07-11 17:24 34s	0 sec	Skipped	
plc_so_domestic_sto_wo_d835	command		2021-07-11 17:24 34s	2021-07-11 17:27 50s	3m 15s	Success	Log
plc_sales_order	command		2021-07-11 17:27 50s	2021-07-11 17:28 01s	10 sec	Success	Log
plc_domestic_sto	command		2021-07-11 17:27 50s	2021-07-11 17:28 01s	10 sec	Success	Log
plc_work_order	command		2021-07-11 17:27 50s	-	6m 58s	Running	Log
plc_import_export	command		2021-07-11 17:27 50s	2021-07-11 17:28 01s	11 sec	Success	Log
leadtime_by_t1_plant	command		2021-07-11 17:28 01s	-	6m 47s	Running	Log

Step 7： 点击 Log

查看Job的 执行过程，是否存在异常情况

10.2 定时调度

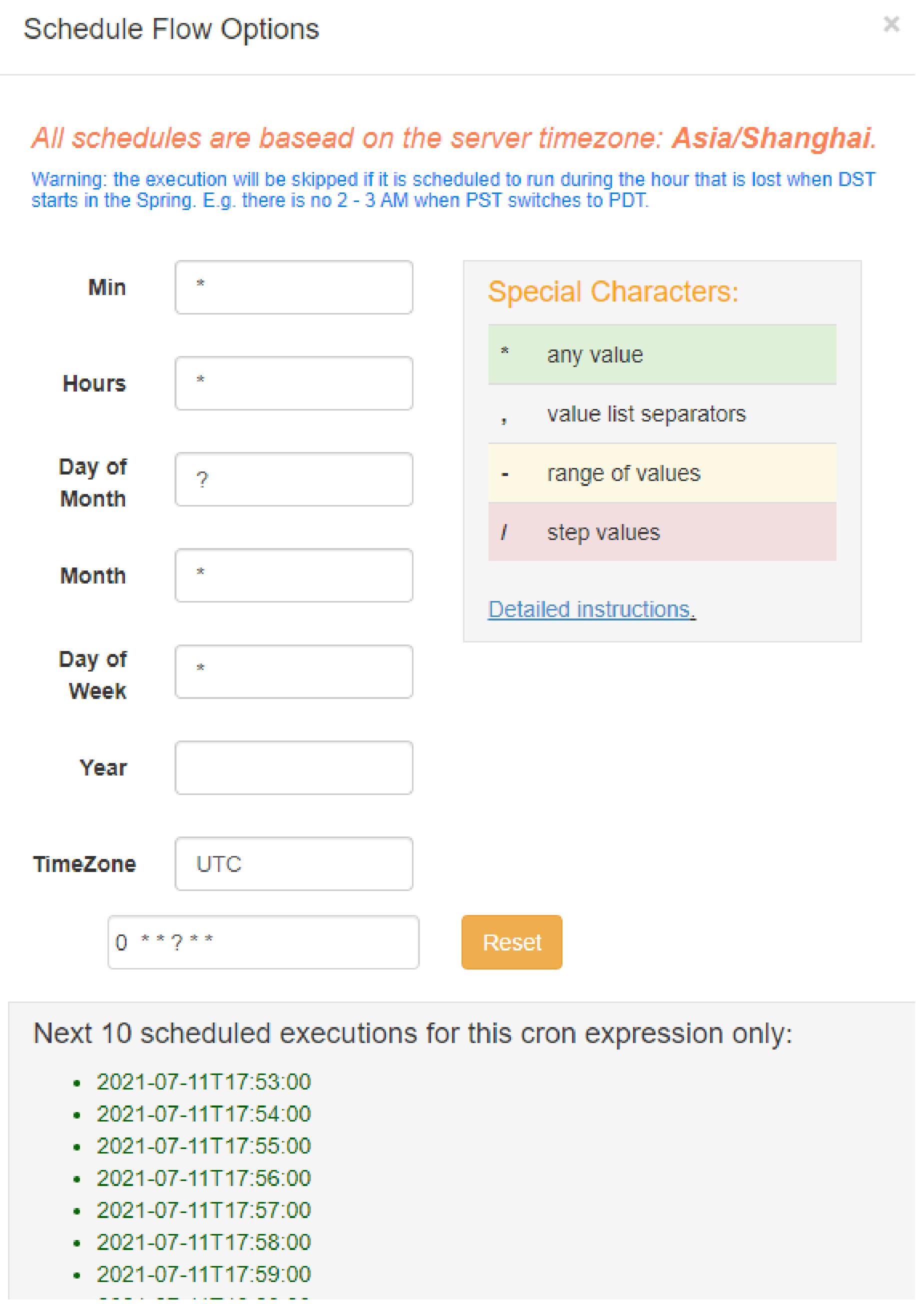
Step 1： 选择Project->Execute flow



Execute Flow data_life_cycle

Right click on the jobs to disable and enable jobs in the flow.

Step 2： 单击 Schedule



Schedule Flow Options

*All schedules are basead on the server timezone: **Asia/Shanghai**.*

Warning: the execution will be skipped if it is scheduled to run during the hour that is lost when DST starts in the Spring. E.g. there is no 2 - 3 AM when PST switches to PDT.

Min	*
Hours	*
Day of Month	?
Month	*
Day of Week	*
Year	
TimeZone	UTC

0 * * ? * *

Special Characters:

*	any value
,	value list separators
-	range of values
/	step values

Detailed instructions.

Next 10 scheduled executions for this cron expression only:

- 2021-07-11T17:53:00
- 2021-07-11T17:54:00
- 2021-07-11T17:55:00
- 2021-07-11T17:56:00
- 2021-07-11T17:57:00
- 2021-07-11T17:58:00
- 2021-07-11T17:59:00

指定参数：

Field Name	Mandatory	Allowed Values	Allowed Special Characters
Seconds	YES	0-59	, - * /
Minutes	YES	0-59	, - * /
Hours	YES	0-23	, - * /
Day of month	YES	1-31	, - * ? / L W
Month	YES	1-12 or JAN-DEC	, - * /
Day of week	YES	1-7 or SUN-SAT	, - * ? / L #
Year	NO	empty, 1970-2099	, - * /

Demo:

0 0 9 ? * * Fire at 9:00am every day

10.3 历史记录

Step 1： Go to Job History