sqoop和dolphinscheduler任务调度和部署

sqoop

1.下载sqoop

```
#sqoop包
wget https://archive.apache.org/dist/sqoop/1.4.7/sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz
#jar包
wget https://repo1.maven.org/maven2/commons-lang/commons-lang/2.6/commons-lang-2.6.jar
#jar包
wget https://repo1.maven.org/maven2/org/apache/hive/hive-common/3.1.2/hive-common-3.1.2.jar
#jar包
wget https://repo1.maven.org/maven2/mysql/mysql-connector-java/5.1.49/mysql-connector-java-
5.1.49.jar

#解压
cd /export/software/
tar zxf ./sqoop-1.4.7.bin_hadoop-2.6.0.tar.gz -C /export/server/
```

2.创建软链接

```
cd /export/server/
in -s  sqoop-1.4.7.bin__hadoop-2.6.0 sqoop
```

3.配置环境变量

```
export SQOOP_HOME=/export/server/sqoop
export PATH=$SQOOP_HOME/bin:$PATH
```

生效

```
source /etc/profile
```

4.配置文件

```
cp /export/server/sqoop/conf/sqoop-env-template.sh /export/server/sqoop/conf/sqoop-env.sh
```

```
vim /export/server/sqoop/conf/sqoop-env.sh
添加修改
#Set path to where bin/hadoop is available
export HADOOP_COMMON_HOME=/export/server/hadoop

#Set path to where hadoop-*-core.jar is available
export HADOOP_MAPRED_HOME=/export/server/hadoop

#Set the path to where bin/hive is available
export HIVE_HOME=/export/server/hive
```

5.复制jar包

```
cp commons-lang-2.6.jar hive-common-3.1.2.jar mysql-connector-java-5.1.32.jar
/export/server/sqoop/lib/
```

6.测试

```
sqoop list-databases --connect jdbc:mysql://node1:3306 --username root --password 123456
等价于
sqoop-list-databases --connect jdbc:mysql://node1:3306/insurance --username root --password
123456
```

dolphinscheduler任务调度

安装

- 基础软件安装(必装项请自行安装)
    - PostgreSQL (8.2.15+) or MySQL (5.7系列) : 两者任选其一即可, 如MySQL则需要JDBC Driver 5.1.47+
    - JDK (1.8+) : 必装，请安装好后在/etc/profile下配置 JAVA_HOME 及 PATH 变量
    - ZooKeeper (3.4.6+) ： 必装
    - Hadoop (2.6+) or MinIO ： 选装，如果需要用到资源上传功能，可以选择上传到Hadoop or MinIO上
- 不要下载源码包，要下载二进制的安装包apache-dolphinscheduler-incubating-1.3.5-dolphinscheduler-bin.tar.gz。并上传到Linux上目录下/export/software/ 。

```
#解压
cd /export/pyworkspace/insurance_dev/5_software
```

```
tar zxvf apache-dolphinscheduler-incubating-1.3.5-dolphinscheduler-bin.tar.gz -C
/export/server/

#必须重命名，不要创建软链接
mv apache-dolphinscheduler-incubating-1.3.5-dolphinscheduler-bin dolphinscheduler_origin
```

- 创建数据库：
  - 进入mysql

    ```
    mysql -uroot -p
    ```

  - 输入

    ```
    CREATE DATABASE dolphinscheduler DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
    utf8_general_ci;

    GRANT ALL PRIVILEGES ON dolphinscheduler.* TO 'root'@'%' IDENTIFIED BY '123456';
    flush privileges;
    ```

- 修改 conf 目录下 datasource.properties 中的下列配置

  添加如下

  ```
  # mysql
  spring.datasource.driver-class-name=com.mysql.jdbc.Driver
  spring.datasource.url=jdbc:mysql://192.168.88.161:3306/dolphinscheduler?
  characterEncoding=UTF-8&allowMultiQueries=true
  spring.datasource.username=root
  spring.datasource.password=123456
  ```

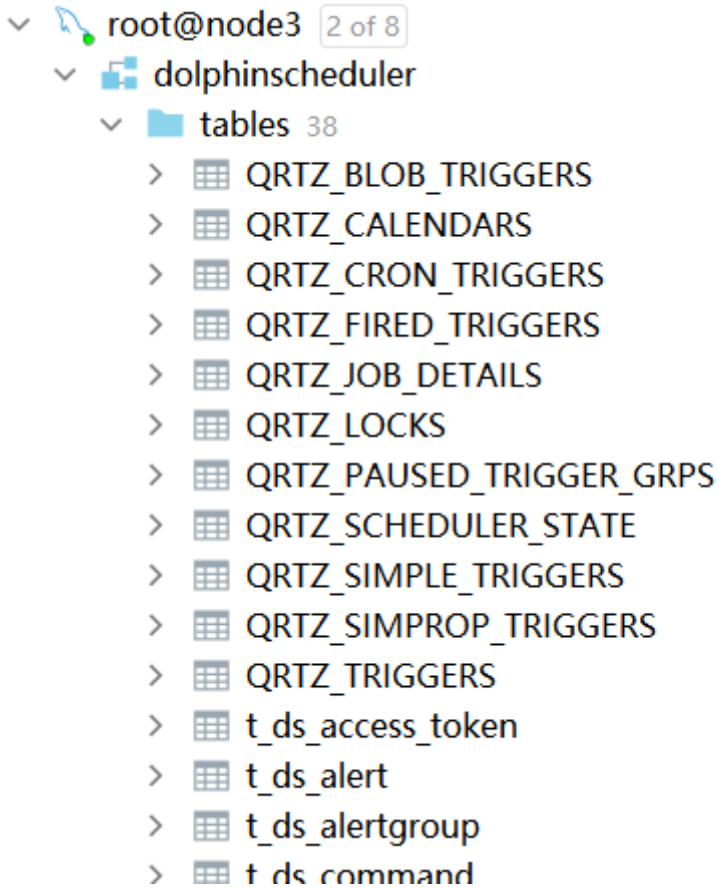
- 复制mysql的驱动mysql-connector-java-5.1.38.jar到dolphinscheduler_origin的lib目录下

  ```
  cp /export/server/hive/lib/mysql-connector-java-5.1.32.jar
  /export/server/dolphinscheduler_origin/
  ```

- 修改并保存完后，执行 script 目录下的创建表及导入基础数据脚本

  ```
  sh script/create-dolphinscheduler.sh
  ```

- 执行后的结果



- 修改 conf/env 目录下的 dolphinscheduler_env.sh 环境变量

```
export HADOOP_HOME=/export/server/hadoop
export HADOOP_CONF_DIR=/export/server/hadoop/etc/hadoop
export PYTHON_HOME=/usr/bin/python
export JAVA_HOME=/export/server/jdk1.8.0_241
export HIVE_HOME=/export/server/hive
export SQOOP_HOME=/export/server/sqoop
export
PATH=$HADOOP_HOME/bin:$PYTHON_HOME:$JAVA_HOME/bin:$HIVE_HOME/bin:$SQOOP_HOME/bin:$PATH
```

- 修改一键部署配置文件 conf/config/install_config.conf中的各参数
  - 还指定了要在哪些机器上一键安装dolphinscheduler，并且指定每个机器承担哪些角色。
  - ```
    #
    # Licensed to the Apache Software Foundation (ASF) under one or more
    # contributor license agreements.  See the NOTICE file distributed with
    # this work for additional information regarding copyright ownership.
    # The ASF licenses this file to You under the Apache License, Version 2.0
    # (the "License"); you may not use this file except in compliance with
    # the License.  You may obtain a copy of the License at
    #
    #     http://www.apache.org/licenses/LICENSE-2.0
    #
    # Unless required by applicable law or agreed to in writing, software
    # distributed under the License is distributed on an "AS IS" BASIS,
    ```
    ```

```
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#


# NOTICE :  If the following config has special characters in the variable `.*
[]^${}\+?|()@#&`, Please escape, for example, `[` escape to `\[`

# postgresql or mysql
#后端的数据库类型指定为MySQL
dbtype="mysql"

# db config
# db address and port
#数据库所在的主机名和端口
dbhost="192.168.88.161:3306"

# db username
#数据库的账户
username="root"

# database name
#数据库里面的库是哪个
dbname="dolphinscheduler"

# db passwprd
# NOTICE: if there are special characters, please use the \ to escape, for example,
`[` escape to `\[`
#数据库密码
password="123456"

# zk cluster
#zookeeper的集群
zkQuorum="192.168.88.161:2181,192.168.88.162:2181,192.168.88.163:2181"

# Note: the target installation path for dolphinscheduler, please not config as the
same as the current path (pwd)
#真正工作的目录,每台机器的所在目录，会自动分发创建
installPath="/export/server/dolphinscheduler"

# deployment user
# Note: the deployment user needs to have sudo privileges and permissions to operate
hdfs. If hdfs is enabled, the root directory needs to be created by itself
#部署的用户，必须有sudo有权限操作hdfs，
deployUser="root"

#告警配置
# alert config
# mail server host
#mailServerHost="smtp.exmail.qq.com"

# mail server port
# note: Different protocols and encryption methods correspond to different ports, when
SSL/TLS is enabled, make sure the port is correct.
#mailServerPort="25"

# sender
#mailSender="xxxxxxxxxx"
```

```
# user
#mailUser="xxxxxxxxxx"

# sender password
# note: The mail.passwd is email service authorization code, not the email login
password.
#mailPassword="xxxxxxxxxx"

# TLS mail protocol support
#starttlsEnable="true"

# SSL mail protocol support
# only one of TLS and SSL can be in the true state.
#sslEnable="false"

#note: sslTrust is the same as mailServerHost
#sslTrust="smtp.exmail.qq.com"


# resource storage type: HDFS,S3,NONE
#上传的文件存在哪里
resourceStorageType="HDFS"

# if resourceStorageType is HDFS, defaultFS write namenode address, HA you need to put
core-site.xml and hdfs-site.xml in the conf directory.
# if S3, write S3 address, HA, for example : s3a://dolphinscheduler,
# Note, s3 be sure to create the root directory /dolphinscheduler
#hdfs通信地址
defaultFS="hdfs://node1:8020"

# if resourceStorageType is S3, the following three configuration is required,
otherwise please ignore
#s3Endpoint="http://192.168.xx.xx:9010"
# s3AccessKey="xxxxxxxxxx"
# s3SecretKey="xxxxxxxxxx"

# if resourcemanager HA enable, please type the HA ips ; if resourcemanager is single,
make this value empty
#如果resourcemanager用了HA，指定HA的ip，有几台写几台
# yarnHaIps="192.168.xx.xx,192.168.xx.xx"

# if resourcemanager HA enable or not use resourcemanager, please skip this value
setting; If resourcemanager is single, you only need to replace yarnIp1 to actual
resourcemanager hostname.
#如果只有一台resourcemanager，就设置哪个
singleYarnIp="node1"

# resource store on HDFS/S3 path, resource file will store to this hadoop hdfs path,
self configuration, please make sure the directory exists on hdfs and have read write
permissions。/dolphinscheduler is recommended
#上传的文件存放目录
resourceUploadPath="/dolphinscheduler"

# who have permissions to create directory under HDFS/S3 root path
# Note: if kerberos is enabled, please config hdfsRootUser=
# hdfsRootUser="hdfs"

# kerberos config
# whether kerberos starts, if kerberos starts, following four items need to config,
otherwise please ignore
```

```
# kerberosStartUp="false"
# kdc krb5 config file path
# krb5ConfPath="$installPath/conf/krb5.conf"
# keytab username
# keytabUserName="hdfs-mycluster@ESZ.COM"
# username keytab path
# keytabPath="$installPath/conf/hdfs.headless.keytab"


# api server port
#web的端口
apiServerPort="12345"


# install hosts
# Note: install the scheduled hostname list. If it is pseudo-distributed, just write a
pseudo-distributed hostname
#安装到哪几台机器，需要被调度的列表
ips="node1,node2,node3"

# ssh port, default 22
# Note: if ssh port is not default, modify here
#怎么连 ssh连22端口
sshPort="22"

# run master machine
# Note: list of hosts hostname for deploying master
#哪些机器作为materserver
masters="node1,node2"

# run worker machine
# note: need to write the worker group name of each worker, the default value is
"default"
#哪些机器作为workerserver node1有最全的hadoop组件
workers="node1"

# run alert machine
# note: list of machine hostnames for deploying alert server
#告警服务，发邮件发短信的服务，启动哪台
alertServer="node3"

# run api machine
# note: list of machine hostnames for deploying api server
#微服务的前端（也可以作为高可用）作为node1机器访问
apiServers="node1"
```

- 修改dolphinscheduler/conf/application-api.properties，指定web前端的端口和域名目录。（什么都不用变）

  ```
  #
  # Licensed to the Apache Software Foundation (ASF) under one or more
  # contributor license agreements.  See the NOTICE file distributed with
  # this work for additional information regarding copyright ownership.
  # The ASF licenses this file to You under the Apache License, Version 2.0
  # (the "License"); you may not use this file except in compliance with
  # the License.  You may obtain a copy of the License at
  #
  ```

```
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#

# server.port=12345
server.port=12345

# session config
server.servlet.session.timeout=7200

server.servlet.context-path=/dolphinscheduler/

# file size limit for upload
spring.servlet.multipart.max-file-size=1024MB
spring.servlet.multipart.max-request-size=1024MB

# enable response compression
server.compression.enabled=true
server.compression.mime-
types=text/html,text/xml,text/plain,text/css,text/javascript,application/javascrip
t,application/json,application/xml

# post content
server.jetty.max-http-post-size=5000000

spring.messages.encoding=UTF-8

#i18n classpath folder , file prefix messages,  if have many files, use ","
seperator
spring.messages.basename=i18n/messages

# Authentication types (supported types: PASSWORD)
security.authentication.type=PASSWORD
```

- 在三台机器上都启动Zookeeper集群

```
/export/server/zookeeper/bin/zkServer.sh start
```

- 一键部署，他会自动将当前机器的软件分发到其他机器上。不用再手动的scp。并且启动每台机器的对应角色进程。

```
sh /export/server/dolphinscheduler_origin/install.sh
```

```
Last login: Mon Dec  6 10:42:41 2021
(base) [root@node1 ~]# jps
21825 MasterServer
3586 SparkSubmit
21922 LoggerServer
3430 RunJar
17734 RunJar
2891 ResourceManager
22189 Jps
2223 NameNode
21874 WorkerServer
3059 NodeManager
21975 ApiApplicationServer
17913 RunJar
2395 DataNode
20222 QuorumPeerMain
(base) [root@node1 ~]#
```

```
(base) [root@node2 ~]# jps
1697 DataNode
8834 Jps
1833 SecondaryNameNode
1932 NodeManager
8701 MasterServer
8014 QuorumPeerMain
```

```
(base) [root@node3 ~]# jps
1845 DataNode
1751 QuorumPeerMain
1977 NodeManager
2585 Jps
2508 AlertServer
```

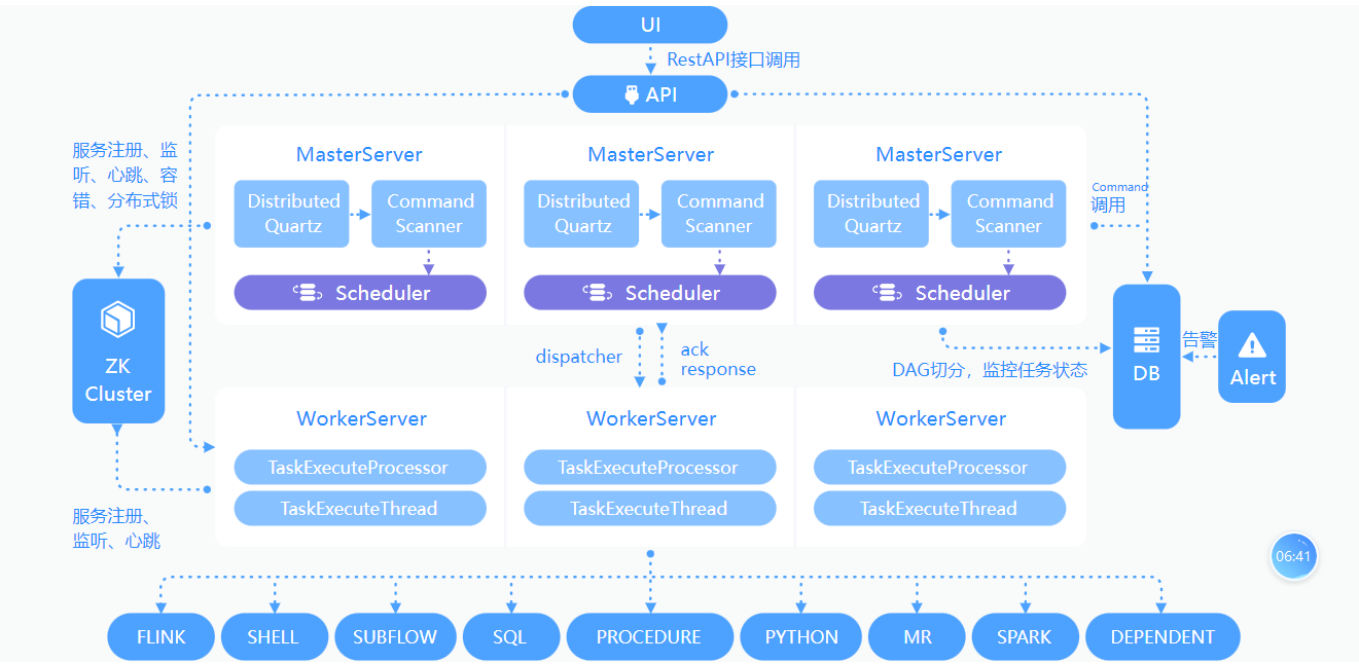
注意

- zookeeper必须要有leader

dolphinscheduler任务调度

> 大体步骤：
> -》下载
> -》MySQL数据库初始化
> -》一堆配置
> -》特别是一件部署的配置文件
> -》启动3台zookeeper
> -》执行一键分发脚本（同时自动会在各自机器启动对应的进程）
> -》做测试
> -》用admin用户登录，只创建子用户，队列。admin不能调度应用程序。
> -》再退出登录，使用子用户登录，子用户才能调度应用程序
> -》用张三用户调度sqoop脚本

特性



高可靠性

- 去中心化的多Master和多Worker，自身支持HA功能，采用任务队列来避免过载，不会造成机器卡死。

简单易用

- DAG监控界面，所有流程定义都是可视化，通过拖拽任务完成定制DAG，通过API方式与第三方系统集成，一键部署

丰富的使用场景

- 支持多租户，支持暂停恢复操作. 紧密贴合大数据生态，提供Spark, Hive, M/R, Python, Sub_process, Shell等近20种任务类型

高扩展性

- 支持自定义任务类型，调度器使用分布式调度，调度能力随集群线性增长，Master和Worker支持动态上下线