Warning: The script contains `rm` command, please test the script fully in test environment before deploying to production.

I. Introduction of the Project

This project is intended to provide sample shell scripts for deploying Cassandra cluster in a Slrum computing cluster that does not provide an interactive command line for computing nodes. It also provide an achievement of loading data using csv.

The recommended environment is Ubuntu or Centos.

There may be some bugs due to different enviroment.

1.1 Structure of the Project :

–CassandraDeployer

—-Cassandra

———initial.sh

———initialAll.sh

———start.sh

———startAll.sh

———run.sh

———loader.cql

1.2 Explanation of Each Part :

- **initial.sh**

This script is responsible for initializing Cassandra on a single computing node, configuring the cassandra.yaml file, and outputting a txt file to the shared folder hostlist according to the current node name. For example, if the initial.sh script is run on xcnd20, it will output a xcnd20.txt to `$HOME/hostlist`, containing the IP address of xcnd20. Maintaining the names and IP addresses of each node in /hostlist facilitates the sequential startup of nodes.

- **initialAll.sh**

This script is responsible for submitting the initial.sh script to five slurm compute nodes.

- **start.sh**

This script is responsible for starting a Cassandra instance on a single node. Additionally, it checks the `$HOME/hostlist file in the shared folder of slurm to ensure that the nodes start in sequence.`

- **startAll.sh**

This script is responsible for submitting **startAll.sh** to five slurm compute nodes.

- **run.sh**

This script calls **initialAll.sh** and **startAll.sh**, which in turn call **initial.sh** and **start.sh** respectively.

- **loader.cql**

Includes table structure and table building statements.

II. Deployment

2.1 Pull the project files to your linux home directory

You should first download the jdk archive within the `$HOME/java` folder. Cassandra requires JDK version 11, and since downloading the JDK requires logging into the Apache website, it cannot be downloaded directly via script, instead, you can use scp or vscode, etc.

The expected structure is to have two folders under the home directory:

–$HOME/

—-Cassandra

———initial.sh

———initialAll.sh

———start.sh

———startAll.sh

———run.sh

———loader.cql

—–java

———jdk-11.0.19_linux-x64_bin.tar.gz

2.2 Set environment variables

2.2.1 Modify the slurm task output file path specified in initialAll.sh, the default value is as follows:

```
#SBATCH --output=/home/Cassandra/initial_output.txt
#SBATCH --error=/home/Cassandra/initial_error.txt
```

Note: You need to replace `"/home` with your **absolute** `$HOME` path. for example, if your `$HOME` path is /home/abc, you need to ensure the sbatch option is “output=/home/abc/Cassandra/initial_output.txt” rather than “output=`$HOME`/Cassandra/initial_output.txt”

2.2.2 Modify the output file path specified in startAll.sh, the default value is as follows:

```
sbatch --nodes=1 --ntasks=1 --cpus-per-task=4 --nodelist=${host} --mem=32G --
time=10:00:00 -p long --output=/home/Cassandra/${host}.txt ~/Cassandra/start.sh
```

Note: You need to replace `"/home"` with your **absolute** `$HOME` path.

2.2.3 Modify each csv file paths specified in loader.cql, the default value is as follows:

```
COPY R1(R1_ID)
FROM '< path to R1.csv>' WITH DELIMITER=',' AND HEADER=FALSE AND CHUNKSIZE=1000 AND
NUMPROCESSES=15;
```

2.3 Confirm idle servers and Modify the nodes in initialAll.sh if necessary

Enter the `sinfo` command to view the currently idle servers in the medium partition. Confirm that the servers specified in the initialAll.sh script (*`xcnc[27-31]`*) are idle in the long partition. If they are idle, proceed to 2.4. If they are not idle, you need to modify line 10 in initialAll.sh, which specifies which servers the initial.sh will be submitted to execute:

```
SBATCH --nodelist=node[27-31]
```

Note: modify *`node[27-31]`* to five other idle servers.

2.4 Run run.sh

Ensure the nodes specified in initialAll.sh are idle. Then, run the following command:

`bash run.sh`

III. Checking the Output Files

check the name of the last node :

`ls $HOME/Cassandra/hostlist`

The output file of the last node that responsible for loading data (assuming the last node's name is xcnc31) :

`$HOME/Cassandra/xcnc31.txt`

Open it to check the progress of deployment.