Cloudera Streaming Analytics 1.7.0

Job lifecycle

Date published: 2019-12-17
Date modified: 2022-05-05

https://docs.cloudera.com/

Legal Notice

Contents

Running a Flink job

After developing your application, you can submit your Flink job in YARN per-job or session mode. To submit the Flink job, you need to run the Flink client in the command line including security parameters and other configurations with the run command.

About this task

Submitting a job means uploading the job's JAR and related dependencies to the Flink cluster and initiating the job execution.

The Flink jobs you submit to the cluster are running on YARN. Submitting a job means that the JAR file of the Flink application is uploaded to the cluster with the related dependencies. and the job execution is initiated. You have the following mode in which you can run your Flink jobs:

- Per-job mode

  Per-job mode means that you run the Flink job in a dedaction YARN application. In this case each submitted Flink job has its own Flink cluster in YARN, with its own Job Manager and Task Managers. When you run Flink jobs in per-job mode, every job submission creates a new cluster. As the cluster deployment has to be created with every submission, the execution of the job can take up time.
- Session mode

  Session mode means that you run multiple Flink jobs in the same YARN sessions. In this case every Flink job shares the cluster, the allocated resources, the Job Manager and Task Managers. When you run Flink jobs in session mode, the submitted jobs are created in one cluster and are long-lived. The execution time is shorter than in per-job mode, however you need to consider that in a session mode a cluster failure affects every Flink job, and recreation from a savepoint can take up time.

You can set how to run your Flink job with the execution.target setting in the Flink configuration file. By default, execution.target is set to yarn-per-job, but you can change it to yarn-session. Alternatively, you can add the corresponding arguments to the flink run command when submitting the Flink job.

Before you begin

- You have installed and configured the Flink service on your CDP Private Cloud Base cluster.

  For more information, see the Adding Flink as a service documentation.
- You have HDFS Gateway, Flink and YARN Gateway roles assigned to the host you are using for Flink submission.

  For more information, see the Cloudera Manager documentation.
- You have uploaded the Flink application JAR file and job properties file to the Flink cluster.

Procedure

1. Connect to the cluster using ssh where you want to run the Flink application.

   ```
   ssh root@<your_host_name>
   ```

   Note: You are prompted to provide your password to the cluster.

2. Submit the Flink job using the flink run command.

   Per-job mode

   ```
   flink run \
     -d \
   ```

```
  -m yarn-cluster \
  -ynm <job_name_in_yarn> \
  -p <job_parallelism> \
  -ys <slots_per_task_manager> \
  -ytm <memory_per_container_in_mb> \
  <job_jar_file> <job_parameters> …
```

Session mode

a. Start a Flink session cluster.

```
flink-yarn-session \
  -d \
  -nm <job_name_in_yarn> \
  -s <slots_per_task_manager> \
  -tm <memory_per_container_in_mb>
```

The flink-yarn-session command outputs the ID of the corresponding YARN application. You need to add the YARN application ID to the flink run command.

```
YARN ApplicationID: application_1616633166424_0024
```

b. Submit the Flink job.

```
flink run \
  -d \
  -m yarn-cluster \
  -e yarn-session \
  -yid <application_id> \
  -p <parallelism> \
  <job_jar_file> <job_parameters>
```

Note:

To run a Flink job, your HDFS Home Directory has to exist. If it does not exist, you receive an error message similar to:

```
Permission denied: user=$USER_NAME, access=WRITE, inode="/user"
```

Related Information

Setting up your HDFS Home directory

Simple Tutorial: Running the application from IntelliJ

Simple Tutorial: Running the application on a Cloudera cluster

Stateful Tutorial: Deploy and monitor the application

Using Flink CLI

You can use the Flink command line interface to operate, configure and maintain your Flink applications.

The Flink CLI works without requiring the user to always specify the YARN application ID when submitting commands to Flink jobs. Instead, the jobs are identified uniquely on the YARN cluster by their job IDs.

The following improvements are implemented for Flink CLI:

- flink list: This command lists all the jobs on the YARN cluster by default, instead of listing the jobs of a single Flink cluster.
- flink savepoint <jobId> and flink cancel <jobId>: The savepoint and cancel commands, along with the other single job commands, no longer require the -yId parameter, and work if you provide only the ID of the job.

- flink run: You do not need to specify -m yarn-cluster, as it is included in the run command by default.

Enabling savepoints for Flink applications

Beside checkpointing, you are also able to create a savepoint of your executed Flink jobs. Savepoints are not automatically created, so you need to trigger them in case of upgrade or maintenance. You can also resume your applications from savepoint.

You can set the default savepoint directory in flink-conf.yaml under state.savepoints.dir property.

The following command lines can be used to maintain savepoints:

Trigger savepoint	$ bin/flink savepoint -yid <*yarnAppID*> <*jobId*> [targetDirectory]
Stop job with savepoint	$ bin/flink stop -yid <*yarnAppID*> <*jobId*>
Resume from savepoint	$ bin/flink run -s <*savepointPath*> [runArgs]
Deleting savepoint	$ bin/flink savepoint -d <*savepointPath*>