TP_SPARK_CORE_4

Axel Jacquin

January 17, 2017

Write a simple program that counts the number of JPG requests in a web log file. The name of the file should be passed in to the program as an argument.

This is the same task you did earlier in the "Getting Started With RDDs" exercise. The logic is the same, but this time you will need to set up the SparkContext object yourself.

Before to start the exercise, ensure to stop all Spark Shell sessions.

1 Write a Spark Application in Scala

A Maven project to get started has been provided: /exercises/projects/countjpgs.

1. Edit the file under src/main/scala/stubs/CountJPGs.scala.

2. First set up a SparkContext and name it sc.

3. In the body of the program, load the file passed in to the program, count the number of JPG requests, and display the count. You may wish to refer back to the TP-Spark-Core-1 exercise for the code to do this.

4. At the end of the program, be sure to stop the Spark Context.

5. From the ~/exercises/projects/countjpgs directory, run the mvn package command in order to build your Scala project using maven.

6. If the build is successful, it will generate a JAR file called countjpgs-1.0.jar under target directory. Run this program locally using spark-submit command. You must specify –class stubs.CountJPGs.

2 Run a Spark Application on Spark Standalone Cluster

Until now, we were working in local mode. We are going to start a Spark Standalone Cluster and run the jar on this cluster. The standalone cluster just

contains one Spark Master and one Spark Worker.

7. In a terminal window, start the Spark Master and Spark Worker daemons:

```
$ sudo service spark-master start
$ sudo service spark-worker start
```

8. View the Spark Standalone Cluster UI: Start Firefox on your VM and visit the Spark Master UI by using the provided bookmark or visiting `http://localhost:18080`

You should not see any applications in the Running Applications or Completed Applications areas because you have not run any applications on the cluster yet.

A real-world Spark cluster would have several workers configured. In this class we have just one, running locally, which is named by the date it started, the host it is running on, and the port it is listening on.

Click on the worker ID link to view the Spark Worker UI and note that there are no executors currently running on the node.

9. In the previous section, you ran your application locally, because you did not specify a master when starting it. Re-run the program, specifying the cluster master in order to run it on the cluster. Set the master to spark://localhost:7077.

10. Visit the Standalone Spark Master UI and confirm that the program is running on the cluster.

3 Configure a Spark Application

11. Re-run the CountJPGs program but this time specify a Application Name

12. Visit the Standalone Spark Master UI and confirm that the program is running under a name.

13. Create a new file named myspark.conf containing settings below :

spark.app.name My-Spark-App
spark.master spark://localhost:7077
spark.executor.memory 400M

14. Re-run the application using this property file and check this application