

# MapReduce, Pig, HCatalog and Oozie:

## Una guía práctica

Luis F. Rivera  
 Departamento Académico de Tecnologías de Información y Comunicaciones (TICs)  
 Universidad Icesi  
 Cali, Colombia  
 Email: lfrivera@icesi.edu.co

### Resumen

My abstract.

### Keywords

*map reduce, pig, hive, hcatalog, oozie.*

## I. INTRODUCTION

My introduction.

## II. OBJETIVOS DEL PROYECTO

My objectives.

## III. MAPREDUCE, PIG Y HIVE

En esta sección se presenta la comparación de los tiempos de ejecución de MapReduce, Apache Pig, y Apache Hive para el cálculo de la temperatura máxima registrada por año.

### III-A. Ejecución con MapReduce

A continuación se detallan los pasos necesarios para ejecutar el programa *MaxTemperature* en su versión MapReduce-Java.

1) *Preparación*: Inicialmente, es necesario compilar el código fuente del programa *MaxTemperature* en su versión Java. Para hacer esto, es necesario clonar el repositorio del libro *Hadoop: The Definitive Guide*<sup>1</sup>. Una vez hecho lo anterior, se debe proceder a compilar los archivos fuente necesarios mediante *Maven*. Finalmente, la ruta del archivo .jar compilado deberá establecerse en una variable de entorno llamada *HADOOP\_CLASSPATH*.

---

```

1 //Clonación del repositorio.
2 git clone https://github.com/tomwhite/hadoop-book.git
3
4 // Compilación del código fuente.
5 mvn package -DskipTests
6
7 // Definición del classpath de Hadoop.
8 export HADOOP_CLASSPATH=/home/sas6/Oozie-Pig-HCatalog-Demos/assets/hadoop-examples.jar

```

---

2) *Ejecución del programa MaxTemperature*: Una vez compilado el código fuente y definida la variable de entorno correspondiente, se procede a ejecutar el programa *MaxTemperature*.

---

```

1 hadoop MaxTemperature /user/hive/warehouse/weather_external/full_data.txt out_mr_300GB

```

---



---

<sup>1</sup>Repositorio provisto por Tom White en <https://github.com/tomwhite/hadoop-book>

3) *Seguimiento a la ejecución del programa*: Una vez iniciada la ejecución del programa, es posible monitorear el progreso del mismo por medio de la consola donde éste se ejecutó o por medio de la interfaz gráfica de YARN.

```
[root@grid102 Oozie-Pig-HCatalog-Demos]# hadoop MaxTemperature /user/hive/warehouse/weather_external/full_data.txt out_mr_300GB
17/07/22 14:23:45 INFO client.RMProxy: Connecting to ResourceManager at grid102.icesi.edu.co/192.168.161.43:8032
17/07/22 14:23:45 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and
execute your application with ToolRunner to remedy this.
17/07/22 14:23:46 INFO input.FileInputFormat: Total input paths to process : 1
17/07/22 14:23:47 INFO mapreduce.JobSubmitter: number of splits:2314
17/07/22 14:23:47 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1500744570838_0005
17/07/22 14:23:47 INFO impl.YarnClientImpl: Submitted application application_1500744570838_0005
17/07/22 14:23:47 INFO mapreduce.Job: The url to track the job: http://grid102.icesi.edu.co:8088/proxy/application_1500744570838_0005/
17/07/22 14:23:47 INFO mapreduce.Job: Running job: job_1500744570838_0005
17/07/22 14:23:52 INFO mapreduce.Job: Job job_1500744570838_0005 running in uber mode : false
17/07/22 14:23:52 INFO mapreduce.Job: map 0% reduce 0%
17/07/22 14:24:09 INFO mapreduce.Job: map 1% reduce 0%
17/07/22 14:24:11 INFO mapreduce.Job: map 2% reduce 0%
17/07/22 14:24:14 INFO mapreduce.Job: map 3% reduce 0%
17/07/22 14:24:21 INFO mapreduce.Job: map 4% reduce 0%
17/07/22 14:24:27 INFO mapreduce.Job: map 5% reduce 0%
```

Figura 1. Monitoreo de la ejecución del programa *MaxTemperature* en MapReduce por medio de la consola en donde éste se ejecutó.

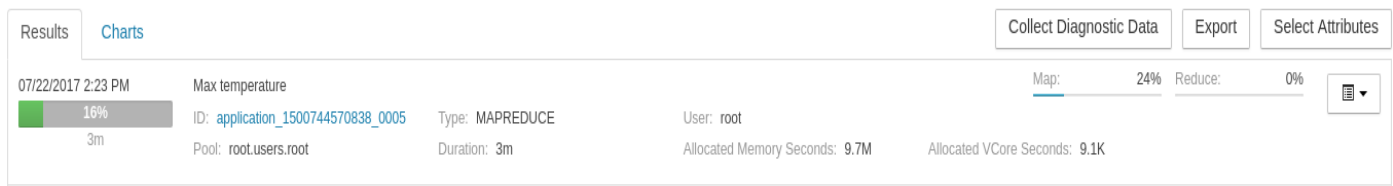


Figura 2. Monitoreo de la ejecución del programa *MaxTemperature* en MapReduce por medio de la interfaz de YARN.

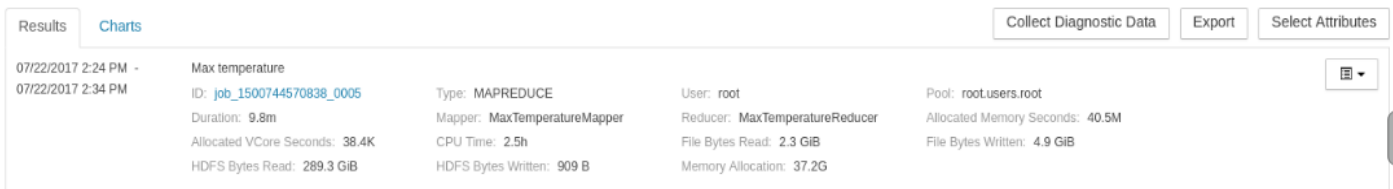


Figura 3. Ejecución finalizada.

### III-B. Ejecución con Hive

A continuación se detallan los pasos necesarios para ejecutar el programa *MaxTemperature* en su versión Apache Hive.

1) *Ejecución con la consola de Hive*: El siguiente script detalla la ejecución del programa *MaxTemperature* en su versión Apache Hive.

```
1 ADD jar /usr/lib/hive/lib/hive-contrib-1.1.0-cdh5.10.1.jar;
2 INSERT OVERWRITE DIRECTORY 'out_max_hive_300GB'
3 SELECT observation_date_year, MAX(air_temperature)
4 FROM weather_managed
5 WHERE air_temperature != 9999 AND at_quality_code IN (0,1,4,5,9)
6 GROUP BY observation_date_year;
```

2) *Seguimiento a la ejecución del programa*: El monitoreo de la ejecución del programa podrá realizarse a través de la interfaz gráfica de YARN, o por la información proporcionada por el Job History Server.

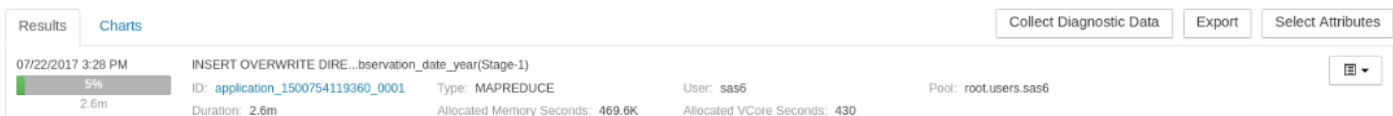


Figura 4. Monitoreo de la ejecución del programa *MaxTemperature* en Hive por medio de la interfaz de YARN.

Job Overview				
Job Name: INSERT OVERWRITE DIRE...bservation_date_year(Stage-1)				
User Name: sas6				
Queue: rootUsers.sas6				
State: SUCCEEDED				
Uberized: false				
Submitted: Sat Jul 22 15:28:56 COT 2017				
Started: Sat Jul 22 15:29:38 COT 2017				
Finished: Sat Jul 22 15:31:59 COT 2017				
Elapsed: 2mins, 20sec				
Diagnostics:				
Average Map Time 42sec				
Average Shuffle Time 2sec				
Average Merge Time 0sec				
Average Reduce Time 4sec				

ApplicationMaster	Attempt Number	Start Time	Node	Logs
1	Sat Jul 22 15:29:36 COT 2017		hgrid7.icesi.edu.co:8042	logs

Task Type	Total	Complete
Map	114	114
Reduce	457	457

Attempt Type	Failed	Killed	Successful
Maps	0	0	114
Reduces	0	0	457

Figura 5. Ejecución finalizada, vista desde el Job History Server.

### III-C. Ejecución con Pig

A continuación se detallan los pasos necesarios para ejecutar el programa *MaxTemperature* en su versión Apache Pig.

1) *Definición del script en PigLatin:* El siguiente script contiene el código utilizado para ejecutar el programa *MaxTemperature* en su versión PigLatin. En las dos primeras líneas del script se detalla el uso de una UDF (*User defined function*), provista por Tom White, para la lectura de registros a partir de la definición de rangos de lectura para sus atributos. El contenido del script fue guardado en un archivo llamado *max-temp.pig*.

```

1 REGISTER $load_loc;
2 records = LOAD '$in_sl' USING com.hadoopbook.pig.CutLoadFunc('16-19,88-92,93-93') AS (year:int,
   temperature:int, quality:int);
3 filtered_records = FILTER records BY temperature != 9999 AND com.hadoopbook.pig.IsGoodQuality(
   quality);
4 grouped_records = GROUP filtered_records BY year;
5 max_temp = FOREACH grouped_records GENERATE group,MAX(filtered_records.temperature);
6 STORE max_temp INTO '$out_max';

```

2) *Definición de los parámetros del script:* Una vez definido el script en PigLatin, se procede a definir en un nuevo archivo los parámetros necesarios para la correcta ejecución del script. Los parámetros mencionados fueron guardados en un archivo llamado *max.param*.

```

1 # Load function location.
2 load_loc=/home/sas6/Oozie-Pig-HCatalog-Demos/assets/pig-examples.jar
3 # Input.
4 in_sl=/user/hive/warehouse/weather_external/full_data.txt
5 # Output.
6 out_max=out_max_pig

```

3) *Ejecución con Grunt en modo batch:* A continuación se detalla el comando utilizado para ejecutar el programa *MaxTemperature* mediante el modo *batch* de Grunt.

```

1 pig -param_file /home/sas6/Oozie-Pig-HCatalog-Demos/scripts/pig/300GB/max.param /home/sas6/Oozie-
   Pig-HCatalog-Demos/src/pig/max-temp.pig

```

4) *Seguimiento a la ejecución del programa:* El monitoreo de la ejecución del programa podrá realizarse a través de Grunt, por medio de la interfaz gráfica de YARN, o por la información proporcionada por el Job History Server.

```
2017-07-22 12:31:02,505 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Processing aliases
filtered_records,grouped_records,max_temp,records
2017-07-22 12:31:02,505 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - detailed locations
: M: records[2,10],records[-1,-1],filtered_records[3,19],max_temp[5,11],grouped_records[4,18] C: max_temp[5,11],grouped_records[4,18] R:
max_temp[5,11]
2017-07-22 12:31:02,590 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 0% complete
2017-07-22 12:32:09,727 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 4% complete
```

Figura 6. Monitoreo de la ejecución del programa *MaxTemperature* en Pig por medio de la consola Grunt desde donde se ejecutó.

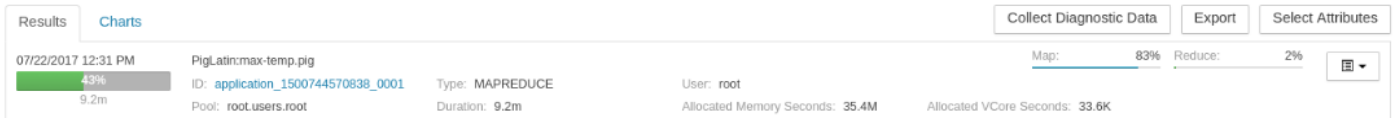


Figura 7. Monitoreo de la ejecución del programa *MaxTemperature* en Pig por medio de la interfaz de YARN.

Job Overview				
Job Name: PigLatin:max-temp.pig				
User Name: root				
Queue: root.users.root				
State: SUCCEEDED				
Uberized: false				
Submitted: Sat Jul 22 12:31:01 COT 2017				
Started: Sat Jul 22 12:31:24 COT 2017				
Finished: Sat Jul 22 12:42:36 COT 2017				
Elapsed: 11mins, 11sec				
Diagnostics:				
Average Map Time: 15sec				
Average Shuffle Time: 24sec				
Average Merge Time: 0sec				
Average Reduce Time: 0sec				

ApplicationMaster				
Attempt Number	Start Time	Node	Logs	
1	Sat Jul 22 12:31:21 COT 2017	hgrid7.icesi.edu.co:8042	logs	

Task Type		Total	Complete	
Map	2314	311	2314	311
	311		311	
Attempt Type		Failed	Killed	Successful
Maps	0	0	2314	
Reduces	0	0	311	

Figura 8. Ejecución finalizada, vista desde el Job History Server.

IV. PIG AND HCATALOG

Pig and HCatalog.

V. OOZIE

Oozie.

VI. RESULTS

Results.

VII. CONCLUSIONS

Conclusions.

VIII. FUTURE WORK

Future work.

REFERENCIAS

[1] H. Kopka and P. W. Daly, *A Guide to L<sup>A</sup>T<sub>E</sub>X*, 3rd ed. Harlow, England: Addison-Wesley, 1999.