

EXP 2: Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

AIM:

To run a basic Word Count MapReduce program.

PROCEDURE:

Step 1: Create Data File

1. Log in with your Hadoop user.
2. Create a file named `word_count_data.txt`.
3. Populate the file with the text data you wish to analyze.

Step 2: Mapper Logic

1. Create a file named `mapper.py`.
2. Write the logic to read input, split lines into words, and output each word with a count.

Step 3: Reducer Logic

1. Create a file named `reducer.py`.
2. Write the logic to aggregate the occurrences of each word and generate the final count.

Step 4: Prepare Hadoop Environment

1. Start Hadoop daemons by running the necessary command.
2. Create a directory in HDFS to store your data.

Step 5: Upload Data to HDFS

1. Copy your `word_count_data.txt` file from the local file system to HDFS.

Step 6: Make Python Files Executable

1. Grant executable permissions to the `mapper.py` and `reducer.py` files.

Step 7: Run Word Count with Hadoop Streaming

1. Download the Hadoop Streaming JAR file.
2. Run the Word Count program by specifying the input data, output directory, and the mapper and reducer files.

Step 8: Check Output

1. Check the output of the Word Count program in the specified HDFS output directory.

Commands:

C:\hadoop\sbin> **start-all.cmd**

C:\hadoop\sbin> **jps**

C:\hadoop\sbin> **cd /**

C:\> **cd hadoop**

C:\hadoop> **hadoop fs -mkdir input**

C:\hadoop> **hadoop fs -put**

C:/Users/monik/Documents/wordcount/data.txt /input1

C:\hadoop> **hadoop jar C:\hadoop\share\hadoop\tools\lib\hadoop-streaming-3.3.6.jar -input /user/input/inpfile.txt -output /user/output -mapper " C:\Users\monik\Documents\wordcount\mapper.py" -reducer " C:\Users\monik\Documents\wordcount\reducer.py"**

OUTPUT:

```
C:\>hadoop fs -put C:\Navya\sem7\WordCount\data.txt /user

C:\>hadoop jar C:\hadoop\share\hadoop\tools\lib\hadoop-streaming-3.3.6.jar ^ -file C:\Navya\sem7\WordCount\mapper.py ^ -file C:\Navya\sem7\WordCount\reducer.py ^ -input /user/data.txt ^ -output /user/output ^ -mapper "python C:\Navya\sem7\WordCount\mapper.py" ^ -reducer "python C:\Navya\sem7\WordCount\reducer.py"
2024-08-19 16:24:38,677 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [C:\Navya\sem7\WordCount\mapper.py, C:\Navya\sem7\WordCount\reducer.py, /C:/Users/navya/AppData/Local/Temp/hadoop-unjar3595406824116402259/] [] C:\Users\navya\AppData\Local\Temp\streamjob5050291353604589221.jar tmpDir=null
2024-08-19 16:24:39,495 INFO client.DefaultHadoopFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-08-19 16:24:39,689 INFO client.DefaultHadoopFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-08-19 16:24:48,592 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/navya/.staging/job_1724064228341_0001
2024-08-19 16:24:48,899 INFO mapred.FileInputFormat: Total input files to process : 1
2024-08-19 16:24:48,969 INFO mapreduce.JobSubmitter: number of splits:2
2024-08-19 16:24:49,121 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1724064228341_0001
2024-08-19 16:24:49,121 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-08-19 16:24:49,272 INFO conf.Configuration: resource-types.xml not found
2024-08-19 16:24:49,272 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2024-08-19 16:24:49,699 INFO impl.YarnClientImpl: Submitted application application_1724064228341_0001
2024-08-19 16:24:49,740 INFO mapreduce.Job: The url to track the job: http://LAPTOP-H3TCD98P:8080/proxy/application_1724064228341_0001/
2024-08-19 16:24:49,742 INFO mapreduce.Job: Running job: job_1724064228341_0001
2024-08-19 16:25:03,229 INFO mapreduce.Job: Job job_1724064228341_0001 running in uber mode : false
2024-08-19 16:25:03,232 INFO mapreduce.Job: map 100% reduce 0%
2024-08-19 16:25:09,349 INFO mapreduce.Job: map 100% reduce 100%
2024-08-19 16:25:09,355 INFO mapreduce.Job: Job job_1724064228341_0001 completed successfully
2024-08-19 16:25:09,432 INFO mapreduce.Job: Counters: 54
    File System Counters
      FILE: Number of bytes read=113
      FILE: Number of bytes written=843702
      FILE: Number of read operations=0

Microsoft Windows [Version 10.0.22621.3880]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd C:\hadoop\sbin

C:\hadoop\sbin>start-all.cmd
This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

C:\hadoop\sbin>jps
3152 ResourceManager
42196 Jps
45876 DataNode
42396 NodeManager
48044 NameNode

C:\hadoop\sbin>hadoop fs -mkdir /user
mkdir: Cannot create directory /user. Name node is in safe mode.

C:\hadoop\sbin>cd ..

C:\hadoop>cd ..

C:\>hadoop fs -mkdir /user

C:\>hadoop fs -put C:\Navya\sem7\WordCount\data.txt /user
put: 'C:/Navya/sem7': No such file or directory
put: '7/WordCount/data.txt': No such file or directory

C:\>hadoop fs -put C:\Navya\sem7\WordCount\data.txt /user
```

```
Virtual memory (bytes) snapshot=1206562816
Total committed heap usage (bytes)=662175744
Peak Map Physical memory (bytes)=342204416
Peak Map Virtual memory (bytes)=401985536
Peak Reduce Physical memory (bytes)=241143808
Peak Reduce Virtual memory (bytes)=405164032

Shuffle Errors
  BAD_ID=0
  CONNECTION=0
  IO_ERROR=0
  WRONG_LENGTH=0
  WRONG_MAP=0
  WRONG_REDUCE=0

File Input Format Counters
  Bytes Read=105
File Output Format Counters
  Bytes Written=42
2024-08-19 16:25:09,432 INFO streaming.StreamJob: Output directory: /user/output
```

File information - part-00000

Download

Head the file (first 32K)

Tail the file (last 32K)

Block information

Block 0

Block ID: 1073741845

Block Pool ID: BP-2005220528-192.168.56.1-1723478856842

Generation Stamp: 1021

Size: 42

Availability:

- LAPTOP-H3TCD9BP

File contents

bye 1

day 1

good 2

hello 4

hi 3

morning 1

Close