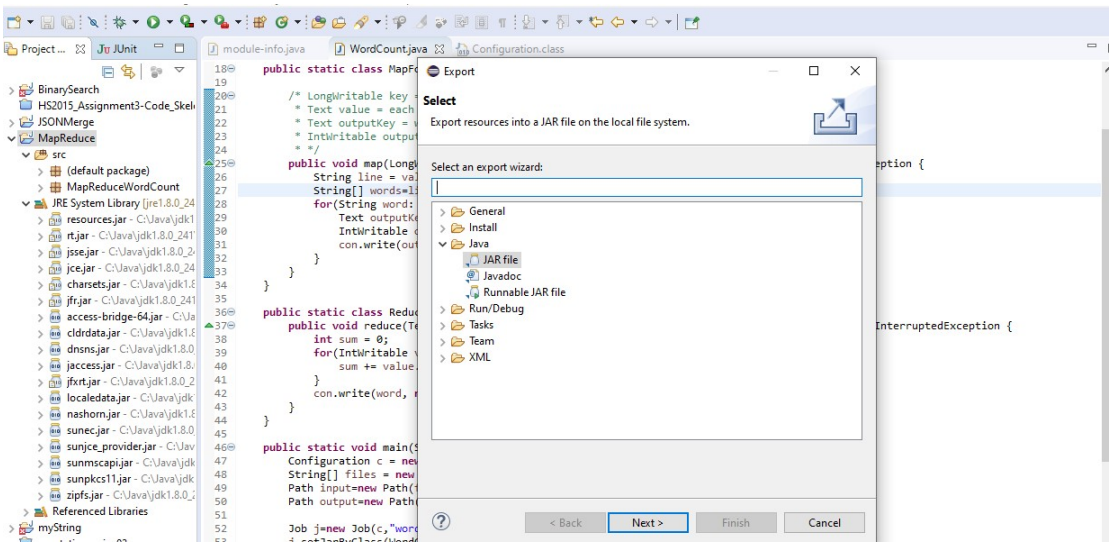
WordCount with MapReduce and Hadoop

Khai Nguyen
khainguyen@temple.edu
CIS 4517 Data Intensive and Cloud Computing

Data Intensive & Cloud Computing
Khai Nguyen
915552057

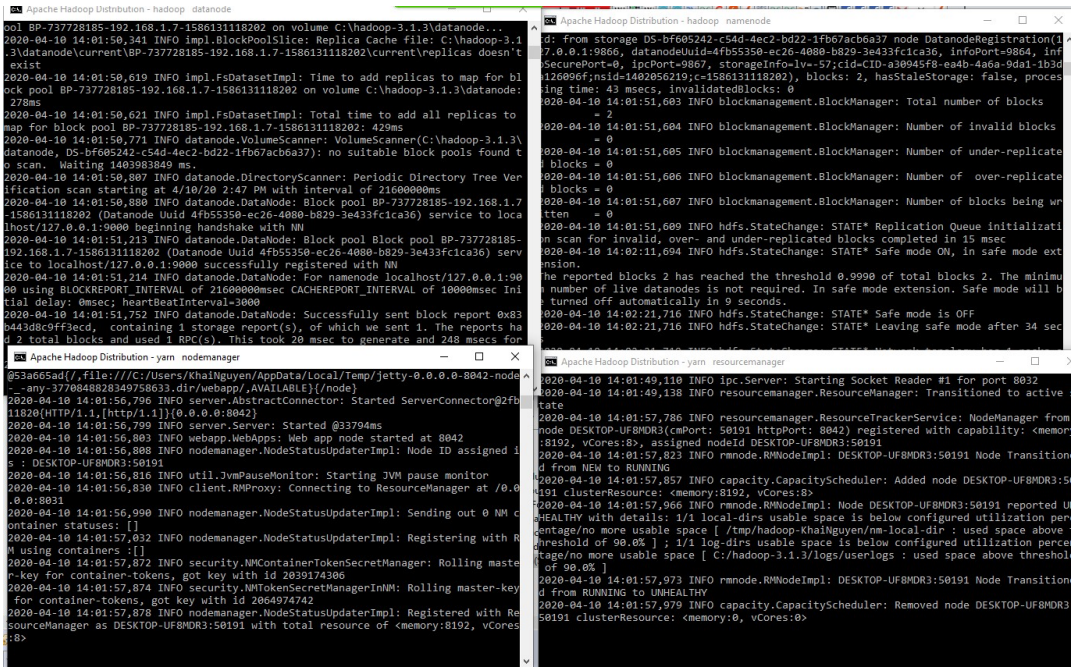
Problem 1 & 2: Word Count

To produce Project > Export > JAR File > *Next*




```
C:\hadoop-3.1.3\sbin>start-all.cmd
This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

C:\hadoop-3.1.3\sbin>
```



Data Intensive & Cloud Computing
Khai Nguyen
915552057

- Upload file to HDFS, store in /testKhaiNguyen

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -put Pride_and_Prejudice.txt /testKhaiNguyen

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -put Pride_and_Prejudice.txt /testKhaiNguyen
2020-04-10 15:21:38,753 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted
 = false

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>
```

- Check if file is upload by viewing content

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -cat /testKhaiNguyen/Pride_and_Prejudice.txt

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -cat /testKhaiNguyen/Pride_and_Prejudice.txt
```

- Run MapReduce

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop jar map_reduce.jar MapReduceWordCount.WordCount /testKhaiNguyen/Pride_and_Prejudice.txt MRDir3

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop jar map_reduce.jar MapReduceWordCount.WordCount /
testKhaiNguyen/Pride_and_Prejudice.txt MRDir3
2020-04-10 15:26:37,949 INFO impl.MetricsConfig: loaded properties from hadoop-metrics2.properties
2020-04-10 15:26:38,132 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-04-10 15:26:38,132 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2020-04-10 15:26:39,454 INFO input.FileInputFormat: Total input files to process : 1
2020-04-10 15:26:39,647 INFO mapreduce.JobSubmitter: number of splits:1
2020-04-10 15:26:40,023 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local160112882_0001
2020-04-10 15:26:40,026 INFO mapreduce.JobSubmitter: Executing with tokens: []
2020-04-10 15:26:40,477 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2020-04-10 15:26:40,493 INFO mapreduce.Job: Running job: job_local160112882_0001
2020-04-10 15:26:40,496 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2020-04-10 15:26:40,535 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2020-04-10 15:26:40,536 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output director
y:false, ignore cleanup failures: false
2020-04-10 15:26:40,540 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2020-04-10 15:26:40,697 INFO mapred.LocalJobRunner: Waiting for map tasks
2020-04-10 15:26:40,702 INFO mapred.LocalJobRunner: Starting task: attempt_local160112882_0001_m_000000_0
2020-04-10 15:26:40,777 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2020-04-10 15:26:40,779 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output director
y:false, ignore cleanup failures: false
2020-04-10 15:26:40,832 INFO util.ProcfsBasedProcessTree: ProcfsBasedProcessTree currently is supported only on Linux.
2020-04-10 15:26:40,933 INFO mapred.Task:  Using ResourceCalculatorProcessTree : org.apache.hadoop.yarn.util.WindowsBasedProcessTr
ee@62465ba0
2020-04-10 15:26:40,969 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/testKhaiNguyen/Pride_and_Prejudice.txt:0+7250
```

- View files created.

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -ls /user/KhaiNguyen/MRDir3
Found 6 items
-rw-r--r--   1 KhaiNguyen supergroup          0 2020-04-10 15:26 /user/KhaiNguyen/MRDir3/_SUCCESS
-rw-r--r--   1 KhaiNguyen supergroup     143904 2020-04-10 15:26 /user/KhaiNguyen/MRDir3/part-r-00000
-rw-r--r--   1 KhaiNguyen supergroup     143250 2020-04-10 15:26 /user/KhaiNguyen/MRDir3/part-r-00001
-rw-r--r--   1 KhaiNguyen supergroup     140265 2020-04-10 15:26 /user/KhaiNguyen/MRDir3/part-r-00002
-rw-r--r--   1 KhaiNguyen supergroup     145175 2020-04-10 15:26 /user/KhaiNguyen/MRDir3/part-r-00003
-rw-r--r--   1 KhaiNguyen supergroup     143153 2020-04-10 15:26 /user/KhaiNguyen/MRDir3/part-r-00004
```

Data Intensive & Cloud Computing
Khai Nguyen
915552057

- Download files

hadoop fs -get /user/KhaiNguyen/MRDir3/part-r-* .

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -get /user/KhaiNguyen/MRDir3/part-r-* .
2020-04-10 16:18:10,009 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
```

- Sort

hadoop fs -cat /user/KhaiNguyen/MRDir3/part-r-* | sort > ./outputFiles/combined.txt

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>hadoop fs -cat /user/KhaiNguyen/MRDir3/part-r-* | sort > ./outputFiles/combined.txt
2020-04-10 16:21:20,698 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
```

- View number of words OR number of lines since each line is designated for 1 word

find /c /v "" "./outputFiles/combined.txt"

Problem 1:

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>find /c /v "" "./outputFiles_regex/combined.txt"

---------- ./OUTPUTFILES_REGEX/COMBINED.TXT: 6738

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>
```

Problem 2:

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>find /c /v "" "./outputFiles/combined.txt"

---------- ./OUTPUTFILES/COMBINED.TXT: 6582

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project>
```

- Display last 5 lines.

Problem 1:

Since there are 6738 lines, we can display the rest of the files starting at line 6733 to get the last 5 lines.

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project\outputFiles_regex>more +6733 combined.txt
YOURSELF        50
YOURSELVES      2
YOUTH   9
YOUTHS  1
ZIP     3

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project\outputFiles_regex>
```

Problem 2:

Since there are 6582 lines, we can display the rest of the files starting at line 6577 to get the last 5 lines.

Data Intensive & Cloud Computing
Khai Nguyen
915552057

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project\outputFiles>more +6577 combined.txt
YOURSELF        50
YOURSELVES      2
YOUTH   9
YOUTHS  1
ZIP     3

C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project\outputFiles>
```

- Display first line

more combined.txt

Problem 1:

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project\outputFiles_regex>more combined.txt
        25436
#       1
$       2
%       1
[EBOOK  1
[ILLUSTRATION   4
]       2
^{ST}   1
_       2
_]      1
_ACCIDENT_      1
_ADVANTAGES_    1
_AFFECT_        1
_ALL_   4
_AM_    1
_ANOTHER        1
_ANY_   1
_ANYBODY        1
_APPEARANCE_    3
_ARE_   2
_AS     1
_BE_    1
```

Problem 2:

```
C:\Users\KhaiNguyen\Documents\CS_4517\Project\Project_4\MapReduce_Project\outputFiles>more combined.txt
        4585
A       2003
ABATEMENT       1
ABHORRENCE      6
ABHORRENT       1
ABIDE   2
ABIDING 1
ABILITIES       6
ABLE    54
ABLUTION        1
ABODE   8
ABOMINABLE      6
ABOMINABLY      4
ABOMINATE       2
ABOUND  1
ABOUT   131
ABOUTS  1
ABOVE   21
ABROAD  4
ABRUPT  1
ABRUPTLY        2
ABRUPTNESS      2
ABSENCE 27
ABSENT  4
```

CIS 4517 Data Intensive & Cloud Computing
Khai Nguyen
915552057

Homework 3
(15 points) Problem 3: The file sample.dat has **two** blocks A and B, explain in HDFS how this file is written to a Hadoop cluster with **one** namenode and **three** datanodes in a default configuration.

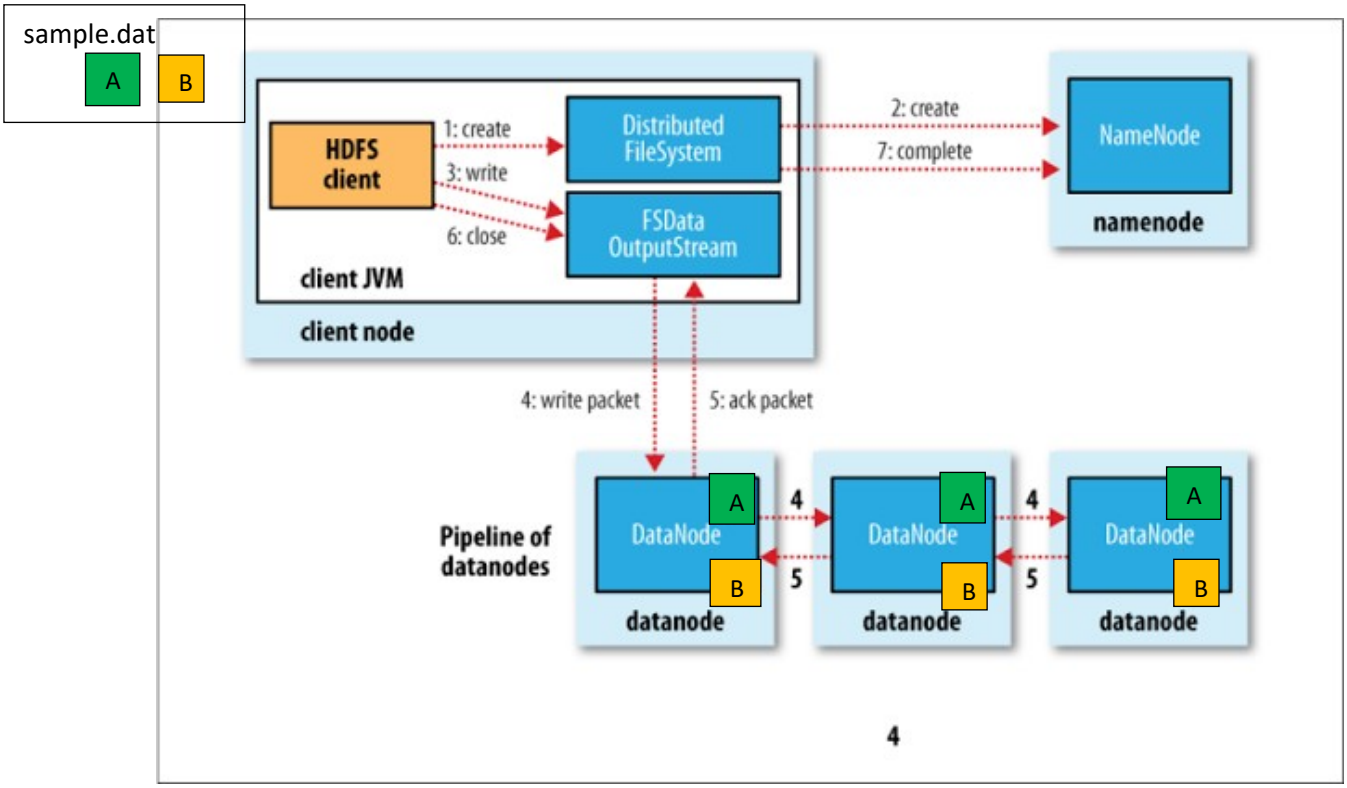


Image from Lecture slides

The client node will notify the namenode about the storing of file sample.data, saying how many blocks the file has. Given that each block needs to be replicated by 3 (default), the client node will write block A, and B on to the 3 datanode sequentially.

When *write* hits the final datanode, an *ack* message will be returned to the previously written nodes recursively, then to the clientnode. At the final step, the client node will notify the namenode to update the metadata of the nodes and the blocks contained.

CIS 4517 Data Intensive & Cloud Computing
Khai Nguyen
915552057

(10 points) Problem 5: In Quorum Consensus algorithm, Reads go to a read quorum of size R and writes go to a write quorum of size W. For a group of **5 replicas**, explain and compare the following three possibilities:

1) R=5 and W=1 • Improves *writes* at the expense of *reads,* since *write*s can be performed at any one replica. • Bad choice, since *writes* can be performed at 1 replica that later fails, leading to data loss.	2) R=1 and W=5 • Improves *reads* at the expense of *writes,* we can read from any replica. • Bad for *writes*. If one in the 5 replicas is down, *writes* have to wait until that replica recovers to read W=5
3) R=3 and W=3 • A good compromise, increasing the cost of reads and providing a reasonable availability of *writes* (W>1 at least)	