Part 2: Optimize Data Technology Platform

Set up HDFS Block Replica and YARN computation resources

```
$nano /opt/hadoop-3.1.4/etc/hadoop/yarn-site.xml
  <property>
    <name>yarn.nodemanager.resource.memory-mb</name>
    <value>4096</value>
  </property>
  <property>
    <name>yarn.nodemanager.resource.cpu-vcores</name>
    <value>4</value>
  </property>

  <property>
    <name>yarn.scheduler.minimum-allocation-mb</name>
    <value>384</value>
  </property>
  <property>
    <name>yarn.scheduler.maximum-allocation-mb</name>
    <value>896</value>
  </property>
  <property>
    <name>yarn.scheduler.minimum-allocation-vcores</name>
    <value>1</value>
  </property>
  <property>
    <name>yarn.scheduler.maximum-allocation-vcores</name>
    <value>2</value>
  </property>

$nano /opt/hadoop-3.1.4/etc/hadoop/hdfs-site.xml
  <property>
    <name>dfs.replication</name>
    <value>3</value>
  </property>

$nano /opt/hadoop-3.1.4/etc/hadoop/mapred-site.xml
   <property>
      <name>yarn.app.mapreduce.am.resource.mb</name>
      <value>896</value>
   </property>
   <property>
      <name>yarn.app.mapreduce.am.command-opts</name>
      <value>-Xmx672m</value>
   </property>

   <property>
      <name>mapreduce.reduce.memory.mb</name>
      <value>896</value>
```

```
   </property>
   <property>
      <name>mapreduce.reduce.java.opts</name>
      <value>-Xmx672m</value>
   </property>
   <property>
      <name>mapreduce.map.memory.mb</name>
      <value>896</value>
   </property>
   <property>
      <name>mapreduce.map.java.opts</name>
      <value>-Xmx672m</value>
   </property>


### Teragen test file

$yarn  jar /opt/hadoop-3.1.4/share/hadoop/mapreduce/hadoop-mapreduce-
examples-3.1.4.jar teragen -Dmapred.map.tasks=1 12  /tmp/12row
$hdfs dfs -get /tmp/12row/part-m-00000  teragen-12row

### MapReduce test file
$hdfs dfs -rm -r /tmp/16m &>/dev/null; yarn  jar /opt/hadoop-3.1.4/
share/hadoop/mapreduce/hadoop-mapreduce-examples-3.1.4.jar teragen -
Dmapreduce.job.maps=1 167770  /tmp/16m
$hdfs dfs -rm -r /tmp/320m &>/dev/null; yarn  jar /opt/hadoop-3.1.4/
share/hadoop/mapreduce/hadoop-mapreduce-examples-3.1.4.jar teragen -
Dmapreduce.job.maps=1 3355400  /tmp/320m


### Terasort test file

$hdfs dfs -rm -r /tmp/sortest/ &>/dev/null; yarn  jar  /opt/
hadoop-3.1.4/share/hadoop/mapreduce/hadoop-mapreduce-
examples-3.1.4.jar terasort -Dmapred.reduce.tasks=1  /tmp/16m  /tmp/
sortest/
$hdfs dfs -rm -r /tmp/sortest/ &>/dev/null; yarn  jar  /opt/
hadoop-3.1.4/share/hadoop/mapreduce/hadoop-mapreduce-
examples-3.1.4.jar terasort -Dmapred.reduce.tasks=1  /tmp/320m  /tmp/
sortest/

$hdfs dfs -rm -r /tmp/terasort-report/ &>/dev/null; yarn  jar  /opt/
hadoop-3.1.4/share/hadoop/mapreduce/hadoop-mapreduce-
examples-3.1.4.jar teravalidate -Dmapred.reduce.tasks=1 /tmp/
sortest/  /tmp/terasort-report
$pig -e 'cat /tmp/terasort-report' 2>/dev/null


### Check MapReduce calculation resources
```

```
$time hadoop jar /opt/hadoop-3.1.4/share/hadoop/mapreduce/hadoop-
mapreduce-examples-3.1.4.jar pi 12 100000
```