HADOOP AND RDBMS HOOK UP

Fall 2010 CSE 633: Parallel Algorithms
Professor: Russ Miller
Student: Li Xiao
xli23@buffalo.edu
2010-12-09
Project Idea Recap

- Business Drivers Changing IT – Really huge data is waiting for processing

- Hadoop is not as efficient as parallels RDBMS

- parallels RDBMS do not scale as well as Hadoop

- Try to make a combination of the two Systems

- Compare the efficiency between hadoop and hadoopDB
The way to do it

- We use simple but telling “word count” application as an example

- We implement the map and reduce function using Map/Reduce framework in java.

- We implement the map and reduce function using Oracle table function

- We using open source project ”FUSE” as middle ware to mount HDFS as normal Operation System

- We configure different number of nodes involved as workers
My Test Environment Settings

• Oracle Database 11g R2 as the DBMS.

• Test data set, I use a program to generate several big text.

• Hadoop, standard version Hadoop available at http://hadoop.apache.org/

• I use three laptops and Vmware to simulate a virtual network to run the test.

• Later, I put the test on U2 Cluster.
My Laptop network topology

CPU: Due core 2.0 GHZ  Ram: 2GB
Network Connection: 1G
Vmware Settings

• The basic unit virtual machine configuration is: 500 MHz CPU and 256M RAM

• Why do it this way, because we need to simulate 8 virtual machines on each laptop at last based on the limited hardware resource.

• The NameNode and JobTracker is on the same laptop

• I test the time needed for calculation with 2, 4, 8, 16 virtual nodes. (on U2 I tested with 2, 4, 8, 16, 32)
The laptop network with everything setup

Oracle installed
NameNode
Jobtracker

FUSE
Worker
Oracle table function for map and reduce

- The detail code for the two function, you may refer to reference

- Here is the snapshot of using the table function

```
SQL> select word, count(*)
2   from <
3   select value(map_result).word word
4   from table<oracle_map_reduce.mapper(cursor(<select a from documents)
5   >>> map_result)
5   group by <word>;

+-----+--------+
| WORD | COUNT(*)|
|------+---------|
| ghi  | 2       |
| abc  | 1       |
| def  | 2       |
| jkl  | 1       |

SQL> _
```
Result

- The comparison between Map/Reduce framework and Oracle processing engine.

- **Green**: Oracle with HDFS, **Red**: Hadoop
Result

- The comparison between Map/Reduce framework using my laptop network and U2
Conclusion

• Using Hadoop File System feeding file to Oracle processing engine is faster than Using Map/Reduce framework in this situation.

• As more nodes get involved in processing, the processing time decreases in general, but with file size around 1G, from 16 nodes, the communication consumption will be dominant.
Future Work

• I am still having some problems with configuring FUSE on U2. I will move on to find a solution to this.

• Try to find a real industry application which could fit into this Hadoop and Oracle environment.
Source Code

Thank You

Thanks!