Due: 4/30/2011 No extension

PROJECT 2: MAPREDUCE PROGRAMMING MODEL AND HADOOP DISTRIBUTED SYSTEM

"In pioneer days they used oxen for heavy pulling, and when one ox couldn't budge a log, they didn't try to grow a larger ox. We shouldn't be trying for bigger computers, but for more systems of computers." Grace Hopper, http://womenshistory.about.com/od/quotes/a/grace_hopper.htm

Note: This is also the beginning quote in the book: Hadoop: A Definitive Guide, by Tom White. Second Edition, O'Reilly, 2011.

1. Purpose:

1. To understand the fundamentals of large scale programming using ***Mapreduce and Hadoop distributed file system.***
2. More importantly understand the significance of this approach for ***data-intensive*** computing and how it departs from the traditional methods.
3. Work with a simple single data-node Hadoop installation guidelines from Cloudera and understand its operation with an example.
4. Design and implement a ***mapreduce solution for a bioinformatics*** application of DNA/protein sequence alignment.
5. Understand the role *the* ***cloud infrastructure*** plays in enabling ***data-intensive computing***.

2. Preparation before lab:

1. Read and understand fundamentals of MapReduce programming model and the Hadoop distributed file system [1, 2].
2. Download a single data-node (and a name-node) Hadoop Cloudera [3]. This distribution is for Linux. So if you running Windows OS you will have to download a VMware virtual machine and install it before you install Hadoop. If you are running Linux you may directly install the Hadoop download.

3. Assignment:

The assignment is more about experimenting and learning rather than about design and implementation from the scratch.

3.1 Wordcount: First part of the assignment is to install the Cloudera/other Hadoop software. Understand the set up and configuration. Execute the *wordcount* program that comes with the software. First use the data that is provided in the Hadoop download. For a larger data choose a set of books available at Project Gutenberg [4]. Understand the Mapper and Reducer classes and study the tutorial provided at [5] or other sources available to you online.

3.2 Graph Algorithms: Now you have a good understanding of the basics. In this step you will move onto a more useful and practical problem. In this case, the problem you will solve is the shortest path algorithm by Dijkstra. A MapReduce (MR) solution for this is available in a presentation by J. Lin [6]. Design the map and reduce functions for this problem and implement it, and test it using the sample graph given in [6]. Then use a larger graph that will be given to you by the TAs and test your solution.

3.3 Bioinformatics Application: The third part involves using MapReduce programming model for an application area that has no dearth of problems. It is Bioinformatics, more specifically DNA and protein sequence analysis and alignment [7]. You are required to create a very large (about 1 GB) synthetic sequence and a synthetic short sequence (short read) and determine the alignment or occurrences of this short read in the larger sequence. We will discuss more about this during upcoming lectures.

4. Project Implementation Details and Steps:

1. Learn the foundational concepts in building parallel processing systems using MR.
2. Understand Hadoop Distributed File System (HDFS).
3. Implement the solutions for 3.2 and 3.3 by using a small data first to debug any programming problem before testing it with larger data set.

5. Project Deliverables:

1. Source code for the solutions to all three problem 3.1, 3.2, 3.3
2. Executable jars for the solutions to the three problems.
3. The data used for the sample runs (small data).
4. Keep the large data for the demos.

6. Submission Details:

Create a compressed deployable distribution of your project. Submit it online.
You should include all the details about the technology requirements to deploy and use your module and also a readme file providing clear instructions as to how to deploy and use your software.

7. Miscellaneous

- Start working on modules in an incremental fashion.
- Very important that you attend lectures to get more information on the project.
- You may discuss the problem with fellow students, however NO code sharing is allowed. You cannot email, you cannot file transfer and you cannot text or tweet.
- You may have to research on the material needed for the project, do not expect it to be given to you!
- Use best design and programming practices.

8. References:

[1] J. Dean and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. OSDI'04: Sixth Symposium on Operating System Design and Implementation, San Francisco, CA, December, 2004, 137-150, http://labs.google.com/papers/mapreduce.html, last viewed Mar 2011.
[2] Hadoop Distributed File System. http://hadoop.apache.org/hdfs/ last viewed Mar 2011.
[3] Cloudera Distribution of Hadoop http://www.cloudera.com/, last viewed Mar 2011.
[4] Project Gutenberg. http://www.gutenberg.org/wiki/Main_Page, last viewed Mar 2011.
[5] Yahoo MapReduce Tutorial, http://developer.yahoo.com/hadoop/tutorial/module4.html, last viewed Mar 2011.
[6]J. Lin, Graph Algorithms with MapReduce. www.umiacs.umd.edu/~jimmylin/cloud-2008-Fall/Session5.ppt, last viewed Mar 2011.
[7] Sequence Alignment. http://en.wikipedia.org/wiki/Sequence_alignment, last viewed Mar 2011.