

# A Study on the Viability of Hadoop Usage on the Umfort Cluster for the Processing and Storage of CReSIS Polar Data

JerNettie Burney, Glen Koch, Jean Bevins & Cedric Hall  
Mentor: Je'aime Powell  
Elizabeth City State University  
Elizabeth City, NC 27909

**Abstract**—The primary focus of this research was to explore the capabilities of Hadoop as a software package to process, store and manage CReSIS polar data in a clustered environment. The investigation involved Hadoop functionality and usage through reviewed publications. The team's research was aimed at determining if Hadoop was a viable software package to implement on the Elizabeth City State University (ECSU) Umfort computing cluster. Utilizing case studies processing, storage, management, and job distribution methods were compared. A final determination of the benefits of Hadoop for the storing and processing of data on the Umfort cluster was then made.

**Keywords**—Hadoop; NameNode; MapReduce; Hive; HBase; HDFS; Zookeeper; JobTracker

## I. INTRODUCTION

Hadoop is a set of open source technologies that support reliable and cost-efficient ways of processing large amounts of data. [1] The exponential growth of individual data footprints, as well as the amount of data generated by machines [2] called for a means to process said data. Hadoop deals with large amounts of data by splitting it into subsets and sending the data to multiple computer processor unit cores, better described as nodes. Multiple nodes used together process data at a much higher rate. Hadoop then reassembles the data into a single result set. Complex data operations shifted to clusters of computers known as clouds [3] in which software such as Hadoop orchestrates the merging of these processes. The Elizabeth City State University (ECSU) research team explored the Hadoop package to see how it might effectively aid in the processing of data sets associated with the Center for Remote Sensing of Ice Sheets (CReSIS) program in conjunction with the ECSU Umfort computing cluster. The research on the software package and associated ancillary structures was to find the origins, versions, adoption and implementations of Hadoop.

Hadoop originated from the open source web search engine, Apache Nutch. The project head and creator was Doug Cutting who utilized the name of his sons' stuffed elephant for the endeavor [4]. Hadoop has been adopted by over 100 different companies, and has a multitude of purposes. The uses ranged from processing millions of data streams, to processing pictures, to information extraction. Some of the more notable organizations that have used Hadoop are Facebook, the San Diego Super Computer

Center, eBay, and Google. What stood out about these organizations is that they have nothing in common as far as services or products. Facebook is a social website used for networking and socializing with friends, EBay is an internet auction website and Google is primarily a search engine website. Lastly, at the San Diego Super Computer Center Hadoop had been used as a grid for processing Light Detection and Ranging (LIDAR) data.

## II. HADOOP FUNCTIONALITY

Hadoop was broken down into different parts and functions and had a different purpose for each. Some of the key components of Hadoop included MapReduce, Zookeeper, HDFS, Hive, JobTracker, NameNode, and HBase. Each of the services worked together to make up the Hadoop software platform. Hadoop's adaptive functionalities allowed various organizations' needs to be met.

### A. MapReduce

MapReduce was a framework that processed large datasets using clusters or grids. It was broken down into two steps; the "Map" step and the "Reduce" step. The Map step was used to index jobs by sending code to each of the dozens of servers in a cluster allowing each server to operate on its own piece of the data. Results were then delivered back from the server as a unified whole. In summary MapReduce maps operations out to all servers and then reduces the results back into a single result set. [5]

### B. Hive

Hadoop uses large scale and fault tolerance capabilities (FTC). This was a result of sensor redundancy in a sensor network, for data storage and processing. Hive was used in Hadoop as a data warehouse infrastructure.

Hive was designed for a variety of purposes. It was used to enable easy data summarization, ad-hoc querying and analysis of large volumes of data. It provides a simple query language called Hive QL, which was based on structured query language (SQL). SQL was a database computer language designed for managing data. Hive enables users comfortable with SQL to create ad-hoc querying, summarization and data analysis. Hive QL also allows traditional map/reduce programmers to be able to use custom mappers and reducers to do a more sophisticated analysis of data that may or may not be supported by built-in capabilities of the language. Hive was not an online transaction

processor. It was best used for batch jobs over large sets data, for example, web logs.<sup>[6]</sup>

### C. Zookeeper

ZooKeeper was a service used to maintain configuration information, manage computer naming schemes, providedistributed synchronization, and provide group services. All of the varioustypes of services were used by distributed applications. Each time any of theservices were implemented, a significant amount of work went into fixing bugs and conditions set in the information.<sup>[7]</sup>

### D. HDFS

The Hadoop Distributed File System (HDFS) was a distributed storage system developedas a means to store various amounts of data . Though it had various similarities to distributed systems; there were differences that set it apart from other systems. One such difference was having designs to work and run onlow-cost hardware. It also had abilities to work on operations even when the system failed, to handle large set data and to stream access to data from the file system. HDFS was primarily written in the Java programming language. <sup>[8]</sup>

### E. NameNode

NameNode was the most essential piece of the HDFS file system. It kept a directory tree of all files in the file system, while tracking where the file data was stored in the cluster. NameNode; however, did not store data files itself. User applications communicated to NameNode when procedures such as locate, add, copy, move, or delete a file were required. The NameNode responds to successful requests by returning a list of applicable DataNode servers. NameNode was considered a single point of failure for a HDFS Cluster. When the NameNode failed, the file system went offline. There was an optional SecondaryNameNode that could be hosted on a separate machine that only created checkpoints of the namespace. It could do so by merging the edited files into a file system image file and would not provide any real redundancy.<sup>[9]</sup>

### F. JobTracker

JobTracker was another service within the Hadoop software package. It farmed out MapReduce tasks to specific nodes in the cluster. JobTracker worked by using a specific algorithm dependent on the required task. User applications submitted jobs and to the Job tracker, which in turn communicated with the NameNode to determine the location of the data. JobTracker then located TaskTracker nodes with availability near the data. Next JobTracker submitted work to chosen TaskTracker nodes, which were monitored. If nodes did not submit signals often, they were deemed to have failed and the work was rescheduled on a different TaskTracker node. TaskTracker would then notify JobTracker when the task failed. JobTracker then decided what process to follow. It could either resubmit the job elsewhere, mark a specific record as something to avoid, or it could blacklist the TaskTracker node as unreliable. When the work was completed, the JobTracker updated its status. The JobTracker could be a point of failure for the Hadoop MapReduce service. If it failed, all running jobs were halted.<sup>[10]</sup>

### G. HBase

Hadoop Base (HBase) was the Hadoop database. It was used when real-time read/write access to large data sets were needed. The goal of HBase was to host very large tables, with billions of rows by millions of columns. HBase was an open-source, distributed, versioned, column-oriented data store modeled after Google's Bigtable,adistributedstorage system for structured data. HBase included suitable base classes for backing Hadoop MapReduce jobs. In order to accomplish this, HBase used tables including cascading, Hive source modules. It also had query predicates to push down using a server side scan, optimizations for real time queries, web service support of XMLand binary data encoding options.<sup>[11]</sup>

## III. CASE STUDIES

Many institutions and companies utilized Hadoop for educational and production purposes. Google, Facebook, eBay and the San Diego Supercomputer Center are just a few of the many users of the Hadoop Platform. Case studies enabled research into how each of the companies utilized Hadoop and lessons that may be leveraged.

### A. Google

Google first created MapReduce, which was a programming model that expressed a simplified computation that was automatically parallelized and executed on a cluster. It was created to solve the complexity problem of distributing large data sets across multiple machines, as well as a source for handling computational failures. MapReduce involves a map operation to every rational record, creating a set including keys and pairs. A reduction operation then occurs to all values that split the same key. Also in MapReduce, re-execution becomes the main device for fault tolerance, being that the library must hold machine failures in a refined matter. Google, in 2007, took the initiative to use Hadoop, which was inspired by MapReduce, to support university courses that pertained to distributed computer programming.<sup>[12]</sup>

Google, along with IBM, through the University Initiative, created new software to develop methods of researching internet-scale applications for helping students understand the challenges involved in such problems. In this, Google and IBM decided to develop resources for massive parallel programs. Hadoop was used in the computing infrastructure for a cluster of running processors. Google mainly used Hadoop for large-scale distributed data processing. Hadoop's Distributed File System was calculated to dependably store large files across machines in a cluster, and was inspired by Google's file system. <sup>[13]</sup>

### B. Facebook

Facebook holds terabytes of disks managed by Hadoop/Hive. Facebook's Hadoop/Hive system consumed 15 terabytes each day of new data. Facebook had 610 Hadoop nodes, running in a solitary cluster. The site also had two major clusters; an 1100 machine cluster with 8800 cores and 12PB raw storage, and a 300 machine cluster

with 2400 cores and approximately three petabytes of raw storage. Facebook uses java and streaming, as well as a data framework using Hive features. Like Facebook, Umfort used various nodes and clusters to manage data. Using Hadoop would permit Polar Grid to administer data easily, and work as a tool for managing greater storage pools. [13]

### C. Ebay

Ebay used Hadoop for image analysis. The cluster was used for production work as well. Ebay used Hadoop's Fair Scheduler to manage distributions, classify job pools, assign weights, limit coexisting jobs for users and teams, set timeouts and delayed preparation. NameNode was the master server whose primary use consisted of managing the HDFS. Zookeeper was the distributed lock coordinator that provided consistency for HBase. JobTracker was used for coordination of jobs and tasks. HBaseMaster was the root storage for HBase and smoothed the process of coordination with blocks of storage. Ebay's use of Hadoop gave a breakdown of their system. Polar Grid doesn't work to differently with its data. Umfort would be able to make the processing of jobs and tasks easier with the setup of the Hadoop platform. Tracking of the jobs was once manually done. This contributed to longer wall times consuming processes. In every project consistency was a plus, doing studies using the ZooKeeper approach would enable Umfort to do more precise research. [14]

### D. San Diego Super Computer Center

Another case study is the investigation of MapReduce technology for a local gridding algorithm for the generation of Digital Elevation Models, also known as DEM, by the San Diego Supercomputer Center. They use Light Detection and Ranging (LIDAR) for measurements of evaluated grid cells. Data paralleling is how the implementation is used with Hadoop MapReduce. Points are inputted and assigned to corresponding grid cells or local bins. Next, the reduce phase shows matching elevations for each grid cell, which are computed from the local bins. The abridged outputs are merged and sorted, and the DEM is then generated in the Arc ASCII grid format. Similar to DEM, PolarGrid investigates the urgent problems in glacial melting using grid evaluations. In the Hadoop-based DEM implementation, inputted points or locations are allocated to corresponding grid cells, showing the use of the parallel method. [15]

### E. Umfort

Over all, Hadoop MapReduce was an encoding model and software structure for writing applications that quickly develop enormous amounts of data in parallel form on huge clusters of computer nodes. Umfort uses a vast amount of data for research that requires computational nodes and storage. Hadoop has proven reliable in distributed computing, and scalability. Like many large companies and institutions, Hadoop could be a great asset to future work and projects. For example, node failures are routinely covered by the framework. Having these failures managed through the system instead of by the user, cuts down time spent by manual repair or reconfiguration.

## IV. CONCLUSION

The Polar Grid team conducted research as to what Hadoop was, what kind of services were available and how other companies and organizations have used Hadoop. After the research, the team found that Hadoop is set of open source technologies that support reliable and cost-efficient ways of dealing with large amounts of data. The Hadoop software comes with user customizable packages such as NameNode, Hive, JobTracker, ZooKeeper, HBase, HDFS, and MapReduce, just to name a few.

The team found that a number of corporations and organizations used Hadoop extensively. Companies like Facebook, Ebay, and Google have different uses for the services. Though some of the corporations may have had the same needs for Hadoop, there were signs of how they could customize the package for a particular use. The team conducted this research as a means to discovering some of these services and how they can be beneficial to the teams' future operations in working with CReSIS data on the Umfort cluster. With the Google case study, the team found that Google used HDFS for the storage of large amount of data on a cluster. Combined with MapReduce, processing data that is being stored could be conducted for the polar data on Umfort. The Facebook case study showed a way that the team could use Hadoop for the administering of nodes and to have increased storage pools for polar data. The Ebay case study gave the team a new idea of how to do image analysis of polar media that CReSIS gathered and stored on the Umfort cluster. Closer studies of the imagery gathered by CReSIS would be more convenient using the Hadoop HDFS. San Diego Super Computer Center's (SDSC) work with LIDAR data was very similar to work that CReSIS does. Because of expansive amount of data that was collected by both organizations over a short period of time, the data must be constantly processed. MapReduce was highly used by SDSC to process this data on their cluster. With these similarities in objectives, the team thought it would be best for Umfort to use MapReduce for the processing of the data received by CReSIS as opposed to the current use of a proprietary MATLAB script.

On the cluster, xCAT was used to manage the nodes; Linux ext3 over Network File System (NFS) was being used for storage; to distribute jobs, TORQUE was utilized; and to process the data, MATLAB was the primary software. The team saw that Hadoop covers all of these areas using its service packages. In addition, a Hadoop cluster would be able to provide services for other research projects that Umfort currently cannot. Therefore, the team highly recommends using Hadoop as opposed to using different software packages that only covers one area and isn't a consistent product. Since Hadoop's services are created under the same umbrella, it would diminish software conflicts.

## V. FUTURE WORK

The team recommended that future teams would install Hadoop on a test cluster, such as Madogo which is an eight-

node cluster located on the campus of Elizabeth City State University; or on a few nodes of the Umfort cluster. The suggested services are MapReduce, Zookeeper, NameNode, TaskKeeper, and HDFS. This would be an experiment that would confirm Hadoop's benefits in a test environment. After the installation of Hadoop has been executed; next, the physical comparisons of the different services installed on Umfort versus the Hadoop packages would take place.

Further testing of Hadoop is suggested with the possibility of taking the research project done by the summer 2009 PolarGrid research group and expanding the project using MapReduce instead of MATLAB. The team also proposed the possibility of converting CReSIS polar data into a Geographical Information System (GIS) database using Hadoop.

## VI. REFERENCES

- [1] *So What is Hadoop?* (2010). Retrieved February 2011, from Atbrox: <http://atbrox.com/2010/02/17/hadoop/>
- [2] T. White, "Hadoop, the Definitive Guide," O'Reilly Media, May 2009, p. 2.
- [3] T. White, "Hadoop, the Definitive Guide," O'Reilly Media, May 2009, p. 9.
- [4] *The Two Flavors of Google.* (2007). Retrieved February 2011, from Bloomberg Businessweek: [http://www.businessweek.com/magazine/content/07\\_52/b4064000281756.htm](http://www.businessweek.com/magazine/content/07_52/b4064000281756.htm)
- [5] *How Map and Reduce Operations are Actually Carried Out.* (2009). Retrieved February 2011, from Hadoop Wiki: <http://wiki.apache.org/hadoop/HadoopMapReduce>
- [6] *Hive/Tutorial.*(2011). Retrived February 2011, from Hadoop Wiki: <http://wiki.apache.org/hadoop/Hive/Tutorial>
- [7] *Zookeeper Overview.* (2010). Retrieved February 2011, from Hadoop Wiki: <http://wiki.apache.org/hadoop/ZooKeeper/ProjectDescription>
- [8]
- [9] *NameNode.* (2009). Retrieved February 2011, from Hadoop Wiki: <http://wiki.apache.org/hadoop/NameNode>
- [10] *JobTracker.* (2010). Retrieved February 2011, from Hadoop Wiki: <http://wiki.apache.org/hadoop/JobTracker>
- [11] *This is Apache HBase.* (2011). Retrieved February 2011, from The Apache Software Foundation: <http://hbase.apache.org/>
- [12] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", *Google, Inc.*, 2004.
- [13] *Google and IBM Announce University Initiative to Address Internet-Scale Computing Challenges.* (2007). Retrieved February 2011, from IBM: <http://www-03.ibm.com/press/us/en/pressrelease/22414.wss>
- [14] *Facebook, Hadoop, and Hive.* (2009). Retrieved February 2011, from DBMS2-A Monash Research Publication: <http://www.dbms2.com/2009/05/11/facebook-hadoop-and-hive/>
- [15] *PoweredBy.*(2011). Retrived February 2011, from Hadoop Wiki: <http://wiki.apache.org/hadoop/PoweredBy>