International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Issue 3, Volume 2 (March 2015) www.ijirae.com

A Study on Digital Library using Hadoop Distributed File System

Piyush Puranik
Department of Computer Engineering, University of Pune

Umesh Kumar Toke
Department of Computer Engineering, University of Pune

Mohamadali Shaikh
Department of Computer Engineering, University of Pune

Mohit Raimule
Department of Computer Engineering, University of Pune

Chetan Patil
Department of Computer Engineering, University of Pune

Abstract— Most digital libraries implemented in organizations worldwide use server-client architecture. These libraries require a large infrastructure investment in server setups and maintenance. Servers are also very difficult to scale based on changing institutional requirements and needs. These drawbacks can be overcome by using a distributed database system. In this paper, we propose an implementation of digital library system using Hadoop Distributed File System. Hadoop is an upcoming technology being used widely for its scalability, parallel computation flexibility, and fault tolerance. This makes it an ideal platform for systems with zero downtime and low fault tolerance.
Keywords— Hadoop, HDFS, MapReduce, Digital Library, Replication

I. INTRODUCTION

A digital library is a system for storing massive amounts of data in a binary, digitally accessible format. With the rapid spurt of technology in this era, a lot of filing cabinet databases are switching over to a digital format. Although digital libraries are conceptually simple enough to comprehend and implement, actual implementation involves a large infrastructural cost and investment. Most of this cost is involved in fulfilling the hardware requirements of maintaining a fully scalable, and fault tolerant architecture. With the increase in number of users or growing demand, a library must allow scaling, and appropriate configuration updates.

Hadoop Distributed File System is a platform which allows easy scalability and solid fault tolerance at a very low implementation cost. It is possible to implement a Hadoop based system on multiple mainstream machines using MapReduce parallelism technique. Hadoop has already seen a rapid acceptance amongst multi-national corporations such as Facebook, Amazon, Yahoo, etc. These corporations have fully functioning Hadoop clusters catering to large amounts of data every day.

II. EXISTING SYSTEM

Most digital library systems today use a client-server architecture. The most significant drawback of such a system is scalability. Although servers are considered to be scalable, the cost required for scaling is quite high. This also includes maintaining the server on a regular basis. Another significant issue with server based systems is downtime and fault tolerance. Server systems consisting of a single server unit, have a higher tendency to fail in case of higher loads or bandwidth surges. Failed systems will again lead to downtimes while the server is in the process of rejuvenation. Although server systems provide support for RAID setups, these setups tend to be expensive, and are still not fully resistant to faults. There are systems which have also been implemented on cloud architecture using the PaaS (Platform as a Service) approach. This system suffers from scalability issues. PaaS systems are not easy to scale, and require a significantly higher infrastructure cost as compared to client-server systems. Most institutions and universities wouldn't go for cloud setups for this very reason.

III. PROPOSED SCHEME

In order to implement an efficient architecture for scalable digital library systems, we can use Hadoop as a reliable foundation. Hadoop provides a robust system which allows scaling and maintenance with zero downtime. Fault tolerance of such a system can be configured dynamically allowing more than 50% of the system to fail while maintaining full functionality.

Keeping these features in mind, we have proposed the use of Hadoop Distributed File System for maintaining a distributed database of files. All files which need to be uploaded or accessed will be present on a cluster of data-nodes which can be scaled as per requirements. An Apache Tomcat server will be used as a frontend to implement a web-based user interface. Web-based interface allows the system to be platform independent, catering to a wide range of users. The Apache Tomcat server will merely act as an interface between Hadoop and end-users. Using a web-interface also allows us to maintain security while accessing data. It allows multiple users to segregate and manage their data individually without any interference from other users.

In its current stage, the system can be implemented on an institution's intranet. All client machines with a web browser are supported. The interface will be designed using HTML5, CSS3, and some elements of jQuery. This will allow an interactive environment for users ensuring an intuitive user interface.

IV. MOTIVATION

The current slew of Indian institutional database/ library systems employ a rudimentary approach to data management. Data management is either done using a closed system designed for use by faculty, or else via a purchased PaaS cloud system. Google Apps are often used for this purpose.These systems do not offer the flexibility of searching through shared files or folders. It also does not provide local or intranet based solutions for institutions. Another limitation is the requirement of high-speed internet to upload large volumes of data. This requirement can be eliminated if we deploy the entire system on an internal network.

Our system was designed with the idea of using just the internal network of an institution. It will allow all members of the institution to upload, share and search through files on the network. It will also help in maintaining a repository of projects, notes and papers written by all students of the institution, including graduated students. This will help new students with their academic endeavors.

V. METHODOLOGY

Our main objective is to set up a Digital Library Engine on HDFS, thus creating an environment where users can upload their files, or retrieve the stored files from the system using a Search Engine provided with the system. Deployment of Digital File Library on HDFS provides higher scalability, reliability and speed data transfer or retrieval.

System Features
Availability
The system will be available 24/7 as it will be deployed over an intranet.

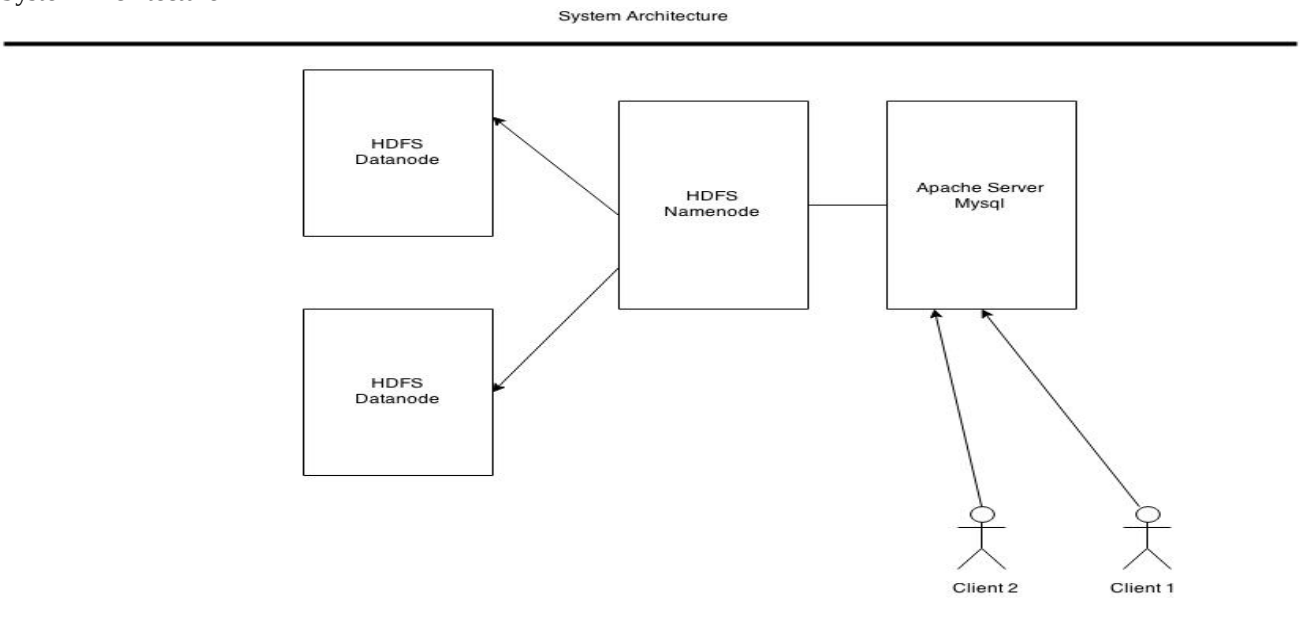
Flexibility
System is highly flexible, since a web user interface is being provided. The user can login from any computer and work from anywhere within an organisation's campus.

Interoperability
Ability of HDFS system to work with other systems is unmatchable. It works very well operating within its own namenode and datanode.

Reliability
HDFS itself is reliable.

Robustness
The version hadoop which has been used has a secondary namenode which acts as primary namenode when the namenode fails to work. Hence, providing higher fault tolerance.

System Architecture



User Interface

The System provides a web interface with the use of languages like HTML5, CSS3, AJAX, Jquery, Javascript at the front end.

Digital Library Engine's user interface is intuitively designed to allow all users to use the system with a short and gradual learning curve.

VI. CONCLUSION

Deploying Digital library on HDFS can be easily configured on commodity hardware, hence cutting down the cost of buying expensive servers or contracting a cloud service provider for a PaaS setup. Deploying the entire system on an intranet eliminates intrusions from outside the network. Using HDFS as a base, we can eliminate issues arising from lack of server scalability, and high infrastructure cost. Downtimes can be cut down significantly by the high availability features present in the system. Scalability and lower downtimes translate into a more robust and reliable system, which can serve a large user base without any failures. By using a web interface, the need for platform dependent applications is eliminated, allowing users to access the system on any machine within an organisation's campus.This also eliminates the user's need to understand the functionality involved in the back end. Although this model deems to resolve most of the basic infrastructural issues of an organisation, it also assumes that the data processed within these organisations is of greater volume. Keeping this requirement in mind, we can safely conclude, that this model is a successful implementation of the Hadoop Distributed File System.

REFERENCES

[1] Farag Azzedin, "Towards scalable HDFS architecture", IEEE, 2013

[2]Kala Karun. A, Chitharanjan. K, "A review on hadoop — HDFS infrastructure extensions",IEEE 2013

[3] Anam Alam, "Hadoop Architecture and Its Issues", IEEE, 2014

[4]Weiming Lu, Liangju Zheng, Jian Shao, Baogang Wei, Yueting Zhuang, "Digital Library Engine: Adapting Digital Library for Cloud Computing", IEEE, 2013

[5] Tom White, "Hadoop The definitive guide", O'Reilly

