ISSN 2348 – 9928
IJAICT Volume 2, Issue 2, February 2015 Doi:01.0401/ijaict.2014.02.08 Published Online 05 (2) 2015

DISTRIBUTED MINING ALGORITHM USING HADOOP ON LARGE DATA SET

Ms. E. Suganya
PG Scholar,
Computer Science and Engineering,
Nandha College of Technology,
Perundurai, Tamilnadu, India.

Mr. S. Thiruvenkatasamy
Assistant Professor,
Computer Science and Engineering,
Nandha College of Technology,
Perundurai, Tamilnadu, India.

Abstract—*Cloud computing, as we all know, is the buzzword today. The cloud offers infrastructure, platform and software as services. we discuss the implementation of cloud services at educational institutions and various opportunities and benefits of cloud services for the institutions. Openstack compute, a cloud fabric controller, serves as the base to implement the concept. The users need to register first to be able to use the cloud services. The usage amount of each and every user is monitored and billed accordingly. Web portals are also included to provide better GUI. The logic tier in the cloud integrates all the functionalities and resources provided to the users. This is a new benchmark towards enterprise applications that is effectively used to facilitate the students to avail its services and facilities. Finally, we present the suggested cloud infrastructure prototype for distributed campus.*

Keywords— Cloud Computing, Open Stack, Cloud Architecture, Cloud Components, IAAS

I. INTRODUCTION

Cloud computing refers to the application, development platforms and hardware delivered to the services over the internet by the cloud providers. It is one of the buzzwords in business and academic environment today. Some of the advantages of cloud computing include reduced implementation and reduced costs, increased mobility for a global workforce, flexible and scalable infrastructures, quick time to market, IT department transformation, greening of the data center, increased availability of high-performance applications to small/medium-sized businesses [1],[2]. Thus, cloud computing refers to providing the resources and capabilities of information technology via services offered by cloud providers. Building an IaaS cloud for the students to avail its facilities will require providing of several level features:

- Allowing application owners to register to cloud services, view their usage amount and bill.
- Allowing Developers / DevOps folks to create and store custom images for their applications.
- Allowing DevOps / Developers to launch, monitor and terminate instances.
- Allowing the Cloud operator to configure and operate the cloud infrastructure.

The above mentioned points are the very core components in providing IaaS. Now, assuming that these four top level features are correct, a conceptual architecture that looks something like this is designed.

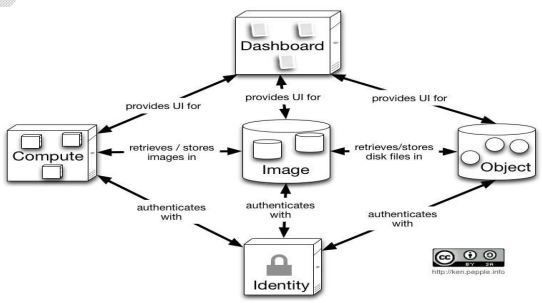

Fig 1 : Logical architecture

In this model, four sets of users (developers, devOps, Owners and Operators) who need to interact with the cloud are there, based on which the functionality needed for each are separated out. There onwards, a tiered approach to the architecture (Presentation, logic and resources) with two orthogonal areas namely integration and management is followed[3].

Corresponding Author: Ms. E. Suganya, Nandha College of Technology, Perundurai, Tamilnadu , India.

ISSN 2348 – 9928
IJAICT Volume 2, Issue 2, February 2015 Doi:01.0401/ijaict.2014.02.08 Published Online 05 (2) 2015

As with the presentation layers like in typical application architecture, components here interact with users to accept and present information. Here, web portals are added to provide graphical interfaces for non-developers and API endpoints for

developers. For more advanced architectures, load balancing, console proxies, security and naming services are also present.

II. CONCEPTUAL ARCHITECTURE

The intelligence and control functionality is given by the logic tier. The tier would house orchestration, scheduling, policy, image registry, and logging. There will be a need to integrate functions within the architecture. Assuming that most service providers will already have a customer identity and billing systems, this cloud architecture will need to integrate with these systems[4].

As with any complex environment, a management tier is needed to operate the environment. This should include an API to access the cloud administration features as well as some forms of monitoring. It is likely that the monitoring functionality will take the form of integration into an existing tool. While monitoring and an API for our fictional provider is highlighted, in a more complete architecture one would see a vast array of operational support functions like provisioning and configure management.

Finally, since this is a compute cloud, actual compute, network and storage resources are needed to provide services to our customers. The tier provides these services, whether they be servers, network switches, network attached storage or any other resources.

2.1 Components of Openstack

In order to achieve the discussed IaaS cloud the openStack, an open source cloud computing platform developed by NASA, Rackspace & many other companies, is used. Openstack compute is a cloud fabric controller that is used to start up virtual instance for either a user or a group. It is also used to start up virtual instances for either a user or a group[5],[6]. It is also used to configure networking for each instance or project that contains multiple instances for a particular project.

It is a system to store objects in a massively scalable large capacity system with built-in redundancy and failover. Object Storage has a variety of applications, such as backing up or archiving data, serving graphics or videos, storing secondary or tertiary static data, developing new applications with data storage integration, storing data when predicting storage capacity is difficult, and creating the elasticity and flexibility of cloud-based storage for your web applications[7].

The OpenStack image service is a lookup and retrieval system for virtual machine images. It can be configured in three ways: using openstack object store to store images; using Amazons simple Storage Solution (S3) storage directly; or by using S3 storage with Object store as the intermediate for S3 access.

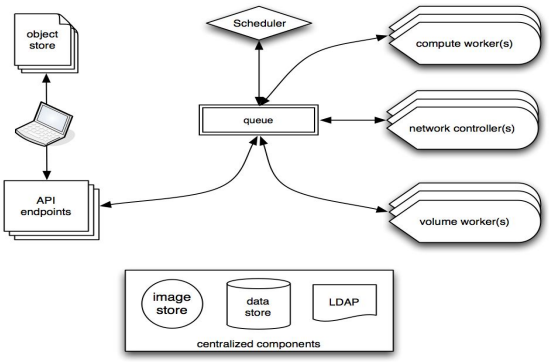

Fig 2 : Components of OpenStack

2.2 Architecture Overview

It is a software used to power your own Infrastructure as a service(IaaS) like amazon web services. It currently encompasses three main projects:

- NOVA (OpenStack compute) which provides virtual servers upon demand. This is similar to Rackspace cloud servers upon demand. This is similar to Rackspace cloud servers or Amazon EC2.
- Swift(OpenStack object Storage) which provides object/blob storage. This is roughly analogous to Rackspace Cloud Files or Amazon S3.
- Glance (OpenStack Image Service) which provides discovery, storage and retrieval of virtual machine images for OpenStack Nova.

III. COMPUTE LOGICAL ARCHITECTURE

There are several logical components of openstack compute architecture but majority of these components are custom-written python daemons of two varieties:

- WSGI applications to receive and mediate API calls
- Worker daemons to carry out orchestration tasks. However, there are two essential pieces of the logical

Corresponding Author: Ms. E. Suganya, Nandha College of Technology, Perundurai, Tamilnadu , India.

ISSN 2348 – 9928
IJAICT Volume 2, Issue 2, February 2015 Doi:01.0401/ijaict.2014.02.08 Published Online 05 (2) 2015

architecture which are neither custom written nor Python based: the messaging queue and the database. These two components facilitate the asynchronous orchestration of complex tasks through message passing and information sharing [8].

The above diagram can be summarized into following:

- End users (DevOps, Developers and even other OpenStack components) talk to nova api to interface with Openstack compute.
- OpenStack Compute Daemons exchange info through the queue (actions) and database (information) to carry out API requests.

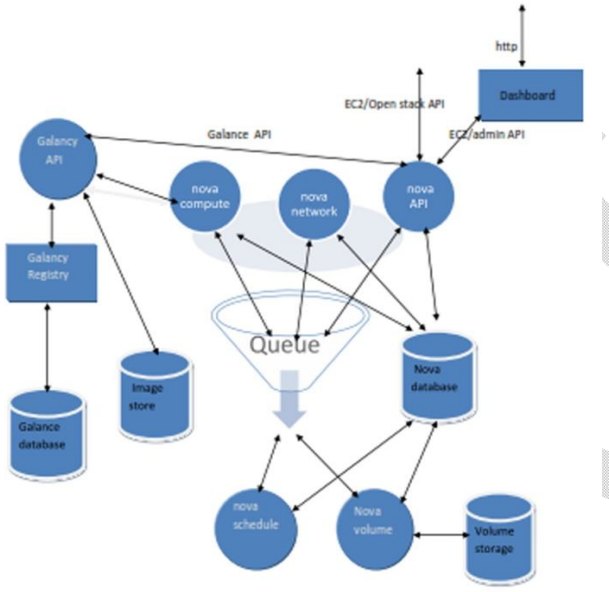

Fig 3 : Compute Logical Architecture

- OpenStack Glance is basically a completely separate infrastructure which OpenStack Compute interfaces through the Glance API.

IV. CONCEPTUAL MAPPING

Now that a conceptual architecture for a cloud provider has been seen and the logical architecture of OpenStack Nova been examined, it is easy to map the OpenStack components to the conceptual areas.

The largest gap in the functional coverage is logging and billing. At the moment, OpenStack Nova doesn't have a billing component that can mediate logging events, rate the logs and create/present bills. This could be fixed in a variety of ways: augmentations of the code, integration with commercial products or services or custom log parsing.

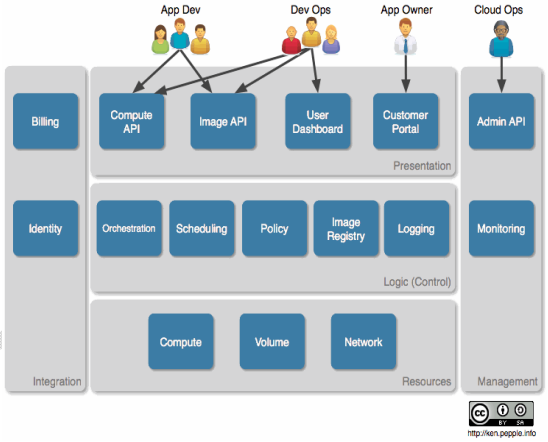

Fig 4 : Conceptual Mapping

Identity is also a point which will likely need to be augmented. Unless a stock LDAP for our identity system is being run, there is a need to integrate the solution with OpenStack Compute.

The customer portal will also be an integration point. While OpenStack Compute providers a user dashboard it doesn't provide an interface to allow application owners to sign up for service, track their bills and lodge trouble tickets.

Ideally, the admin API would replicate the all functionalities that can be done via the command line interface.Cloud monitoring and operations will be an important area of focus for the service provider. A key to any good operations approach is good tooling. While OpenStack Compute provides nova-instancemonitor, which tracks compute node utilization, we're really going to need a number of third party tools for monitoring.

Policy is an extremely important area but it is very provider-specific. Everything from quotas (which are supported) to quality of service (QoS) to privacy controls can fall under this area. An OpenStack Nova partial coverage here has been given but that might vary depending on the intricacies of the providers' needs. For the record, the catus release of open stack compute provides quotas for instances (number and cores used, volumes (size and number), floating IP addresses and metadata.

Scheduling within openstack compute is fairly rudimentary for larger installations today. The pluggable scheduler supports

chance (random host assignment), simple (least loaded) and zone (random nodes within an availability zone). As within most areas on this list, this will be greatly augumented in Diablo. In development, are distributed schedulers and schedulers, that understand heterogenous hosts (for support of GPUs and differing CPU architectures).

V. CLOUD'S BENEFITS

In data centers today, many computers suffer underutilization in computing power and networking bandwidth. For example, projects may need a large amount of computing capacity to complete a computation, but no longer need the computing power after completing the computation. One wants cloud computing when one wants a service that's available on-demand with the flexibility to bring it up or down through automation or with little intervention. The phrase cloud computing is often represented with a diagram that contains the services that afford computing power harnessed to get work done. Much like the electrical power that is received each day, cloud computing provides subscribers or users with access to a shared collection of computing resources: networks for transfer, servers for storage, and applications or services for completing tasks.

These are the salient features of a cloud: On-demand self – service: Users can provision servers and networks with little human intervention of the Cloud Models. Resources of the Cloud must be the IAAS model or service model. Thus models rented by the Service Level Agreement of the consumers.

- ***Network access***

Any computing capabilities are available over the network. Many different devices are allowed access through standardized mechanisms[9].

- ***Resource pooling***

Multiple users can access clouds that serve other consumers according to demand.

- ***Elasticity***

Provisioning is rapid and scales out or in based on need[10].

- ***Metered or measured service***

Just like utilities that are paid for by the hour, clouds should optimize resource use and control it for the level of service or type of servers such as storage or processing. Cloud computing

offers different service models depending on the capabilities a consumer may require.

- ***SaaS***

Provides the consumer the ability to use the software in a cloud environment, such as web-based email for example.

- ***PaaS***

Platform as a Service Provides the consumer the ability to deploy applications through a programming language or tools supported by the cloud platform provider. An example of platform as a service is an Eclipse/Java programming platform provided with no downloads required[11],[12].

- ***IaaS***

Infrastructure as a service Provides infrastructure such as computer instances, network connections, and storage so that people can run any software or operating system[13].

When terms such as public cloud or private cloud are used, they refer to the deployment model for the cloud. A private cloud operates for a single organization, but can be managed on-premise or off-premise. A public cloud has an infrastructure that is available to the general public or a large industry group and is likely owned by a cloud services company. The NIST also defines community cloud as shared by several organizations supporting a specific community with shared concerns. Cloud can also be describes as hybrid[14]. A hybrid cloud can be a deployment model, as a composition of both public and private clouds, or a hybrid model for cloud computing may involve both virtual and physical servers.

Cloud computing can help with large scale computing needs or can lead consolidation efforts by virtualizing servers to make more use of existing hardware and potentially release old hardware from service. People also use cloud computing for collaboration because of its high availability through networked computers. Productivity suits for word processing, number crunching, and email communications and more also available through cloud computing[15]. Cloud computing also avails additional storage to the cloud user, avoiding the need for additional hard drives on each users' desktop and enabling access to huge data storage capacity online in the cloud.

- ***Hadoop Distributed File System***

Apache Hadoop is an open-source software framework that supports data-intensive distributed applications licensed under the Apache v2 license. It supports parallel running of applications on large clusters of commodity hardware. Hadoop derives from Google's MapReduce and Google File System (GFS) papers. Hadoop implements a computational paradigm named MapReduce where the application is divided into many

small fragments of work, each of which can execute or re-execute on any node in the cluster. In addition, it provides a distributed file system that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both map reduce and the distributed file system are designed so that node failures are automatically handled by the framework.

- ***Distributed Mining in Map Reduce***

The main objective of data mining is to discover knowledge from large databases. The discovered knowledge helps in decision making. Some of the basic data mining topics are association rule mining, sequential pattern mining, clustering and classification. The most important and popular topic in data mining is frequent pattern generation. The basic concept of frequent pattern mining is to discover patterns from database that are more frequent than the specific threshold. As per association rule is defined as X => Y, where X and Y are the set of items. There are two main primitive algorithms in frequent mining are Apriori algorithm (Agarwal & Srikanth 1994) and Frequent Pattern Growth approach (Han et al 2004).

The Apriori like method suffers from two main problems. One is main memory has to be large to hold all candidate item sets. Second it scans the database multiple times. So inorder to overcome this Han (2004) published his research work introducing a new structure for storing data Frequent Pattern tree (FP-tree) where transactions are stored in tree structure in a compressed format. Then using FP growth algorithm, frequent item sets has been discovered from databases.But increase in database size and intensity affects the computation efficiency and takes long time to execute. Also demand for memory was the main concern in case of single task computing because to mine large databases more memory is required to handle large number of item sets especially when the threshold is low.

- ***Map Reduce Programming Model***

A Map Reduce program is composed of a map() procedure that performs filtering and sorting such as sorting students by first name into queues, one queue for each name and a reduce() procedure that performs a summary operation such as counting the number of students in each queue, yielding name frequencies. The Map Reduce system orchestrates by marshalling the distributed servers, running the various tasks in parallel, managing all communications and data transfers between the various parts of the system, providing for redundancy and fault tolerance, and overall management of the whole process. Map Reduce is a programming model and an associated implementation for processing and generating large data sets. Users specify a map function that processes a key/value pair to generate a set of intermediate key/value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key.

VI. CONCLUSION

Cloud Computing paradigm is a new approach to produce a solution for old problems. This approach offers many benefits to enterprises, industries and educational institutions at large. Most of the research in literature focused on benefits, opportunities, advantages, disadvantages, risks and configuration of Cloud computing for enterprises. Use of Cloud Computing in educational institutions has many benefits such as accessing the file storages, e-mails, databases, educational resources, research applications and tools anywhere for faculty, administrators, staff, students and other users. The main goal of suggested prototype is; managing effectively the technological needs of the institutions such as delivery of software, providing of development platform, storage of data, and computing.

References

[1] Khmelevsky, Y., & Voytenko, V. (2010). Cloud Computing Infrastructure Prototype for University Education and Research. Proceedings of the 15th Western Canadian Conference on Computing Education. Kelowna, Canada: ACM.

[2] Katzan, H. (2010). The Education Value Of Cloud Computing. Contemporary Issues In Education Research , 3 (2), 37-42.

[3] Behrend, T. S., Wiebe, E. N., London, J. E., & Johnson, E. C. (2011). Cloud computing adoption and usage in community colleges. Behaviour & Information Technology , 30 (2), 231–240.

[4] Vouk, M. A. (2008). Cloud Computing – Issues, Research and Implementations. Journal of Computing and Information Technology, 4, 235–246.

[5] IBM Press. (2009, November 4). IBM Press Room. Retrieved March 21,2011,fromIBM:http://www03.ibm.com/press/us/en/pressrelease/28749.wss

[6] Rittinghouse, J. W., & Ransome, J. F. (2010). Cloud Computing Implementation, Management, and Security. New York: Taylor and Francis Group.

[7] Mell, P., & Grance, T. (2009, 7 10). The NIST Definition of Cloud Computing. Retrieved 2 11, 2011, from NIST Information Technology Laboratory:http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf

[8] Foster, I., Zhao, Y., Raicu, I., & Lu, S. (2008). Cloud Computing and Grid Computing 360-Degree Compared. Grid Computing Environments Workshop (pp. 1 - 10). Austin: GCE.

[9] Sultan, N. (2010). Cloud computing for education: A new dawn? International Journal of Information Management , 30, 109–116.

[10] Surgient, D. M. (2009, April 9). The five defining characteristics of cloud computing. Retrieved March 15, 2011, from ZDNet: http://www.zdnet.com/news/the-five-defining-characteristics-of cloud co mputing / 287001

[11] Spínola, M. (2009, September 6). The Five Characteristics of Cloud Computing. Retrieved March 17, 2011, from Cloud Computing Journal: http://cloudcomputing.sys-con.com/node/1087426
[12] Murley, D. (2009). Law Libraries in the Cloud. Law Library Journal , 101:2 (15), 249-254.
[13] Gruman, G., & Knorr, E. (2008, April 7). What Cloud Computing Really Means. Retrieved March11, 2011, from InfoWorld: http://www.infoworld.com/d/cloud-computing/what-cloud-computing-reallymeans-031
[14] Sasikala, S., & Prema, S. (2010). Massive Centralized Cloud Computing (MCCC) Exploration in Higher Education. Advances in Computational Sciences and Technology, 3 (2), 111–118.
[15] IBM. (n.d.). Cloud Computing. Retrieved March 20, 2011, from IBMAcademiInitiative:https://www.ibm.com/developerworks/university/cloud