Dynamic Management of Virtual machines for Server Consilidation in Data Centers

Ashwini K.S[1]. PG Student, Dept of CSE, CIT, Gubbi

Prof Shantala C.P[2] HOD, Dept of CSE CIT, Gubbi

Mr. Girish L[3] Asst. Professor, Dept of CSE CIT, Gubbi

Abstract — **Cloud computing technology which provides computing resource on demand over the internet as a service. To meet this requirement virtualized data centers are established all around the world. However, data centers utilize large amount of electrical power. Thus results in high operating costs and emits carbon dioxide to environment. Carbon dioxide emitted from cloud data centers cause global warming.**

This paper provides a model and techniques for the dynamic management virtual machines in data centers. The main purpose is to increase the utilization of computing resources and minimize energy consumption by switching off the idle nodes. Live migration technique, moving virtual machines between different physical servers, is mainly used to reduce the number of active compute nodes in cloud data centers with minimal downtime. In this paper, we provide an architecture and software framework of open stack neat, for dynamic virtual machine consolidation in open stack cloud deployment. Open stack is an open source cloud platform which provides infrastructure as a service solution. Open stack allows users to deploy virtual machines which manages different workloads.

Keywords — server consolidation, live migration, VM consolidation, virtualization.

I. INTRODUCTION

Cloud computing model makes use of virtualization to provide computing resources to the users over internet on demand and pay per use pricing model [1]. Cloud services allow users to use software and hardware those are set up and handled by third persons at distributed locations. It leverages the characteristics of on demand self service and rapid elasticity so that it enables customers to dynamically and efficiently manage their resource usage according to the present workload conditions. These characteristics of the cloud computing model allow the users not to invest high on infrastructure and thereby minimizing the time to market and provide the space for innovation.

Cloud computing resources are handed over to the users through three basic service models listed below:

1. Infrastructure as a service (IaaS): IaaS provides access to the computing resources in the form of virtual machines. Virtual machine brings the user an aspect of dedicated physical machine. The user is able to operate the system within a virtual machine and run the required software.
   Examples: Windows Azure, Google Compute Engine, Amazon EC2.
2. Platform as a Service (PaaS): PaaS provide access to the computing resources in the form of an application program interface. It is used by customer to develop and run their own applications. The user doesn't have the rights to access the system recourses. Allocation of resources to the application is done automatically by platform.
   Examples: Microsoft Azure, Google App Engine
3. Software as a Service (SaaS): SaaS provides software applications as a service to the users on subscription basis. Users don't have to bother about installation, setup and running the software application.
   Examples: Google Apps, Microsoft Office 365

In this project, we mainly work on infrastructure as a service platform. Other than computing service models, cloud computing services are also classified according to deployment models.

Virtualization [2] is a method of creating virtual of something like device or it may be resources, like server, storage, operating systems or computer network resources. It divides the computer resources in to different execution environments. By using virtualization concept as shown in Figure 1 we can create multiple virtual machines on same physical computer. Each virtual machine is capable of running different operating systems and its applications at the same time. Figure 2 gives the basic concept of virtualization which consists of physical machine, hypervisor and multiple virtual machines. Datacenters use virtualization technology due to popular benefits of consolidate multiple running servers to save energy and power, resource isolation, scale the infrastructure as the demand increases, increase the availability through dynamic resource provisioning and live migration

Single physical machine contains multiple operating systems where each works as separate physical machine. One physical host is capable of creating many virtual machines and each runs its own operating system and its applications. Virtual Machine [3] works exactly as like physical machine and has its own virtual processor, hard disk and network identity. The advantages of Virtual machines are as follows

ISSN: 2278 – 1323

1. Even though all the VMs use the underlying resources of single machine, they work completely unrelated to each other as like separate physical hosts. Example, if there are 3 virtual machines on single host and if 1 crashes the other 2 virtual machines are available.
2. A virtual machine is a virtual container that binds a complete set of virtual computing recourses, operating system and applications. It makes a virtual machine easy to port and manage.
3. Virtual machines are completely separate from their physical resources. Virtual machines can run different operating systems on the same single physical server

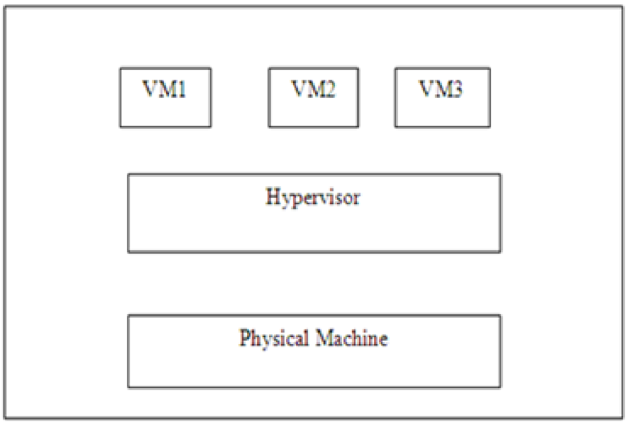

Figure 1: Virtualization

Live migration [4] is the process of moving running virtual machine from one physical node to another node without disconnecting the active services. If it is properly done, process occurs without any observable notifications from the view of end user. Live Migration allows provider to take virtual machine down for repair or upgrading without disrupting the services to end user. Figure 2 depicts the process of live migration.

When live migration is in process it is important to minimize both downtime and total migration time. Down time is period during which services are not available to the users because there is no executing instance of service. This duration of time is easily visible to users as disruption of service. Total migration time is total duration taken between initiation of migration and when the original virtual machine is discarded in source node. Memory migration of virtual machine occurs in three phases [5]:

1. Push Phase: In this phase the source virtual machine keep on running, at the same time some memory pages are moved across the network to new physical node. To maintain consistency, memory pages that are altered during this time must be moved again.
2. Stop and Copy Phase: In this phase the source virtual machine is terminated and rest of all memory pages are moved to the destination virtual machine at new node and new virtual machine is initiated. The duration between ending virtual machine on source and initiating it on new node is called down time. Down time of a virtual machine should be in milliseconds to seconds and it depends on memory size and application retain on virtual machine.
3. Pull Phase: In this phase a new virtual machine is started at destination node and when it tries to gain access a memory page that not yet been transferred, this memory page is faulted in from source virtual machine through the network. Virtual machine migration is started by shut down the virtual machine at the source node, essential execution state of the virtual machine is copied to the destination machine. The new virtual machine is then started at the at the destination node, even the complete memory pages are not yet been copied and remains at the source node itself. At the destination node, when virtual machine access memory pages that has not been copied to destination, it initiates page faults, which are captured at the destination and are taken away to the source node across network. Source node replay to the network page faults by transferring the faulted memory page

.

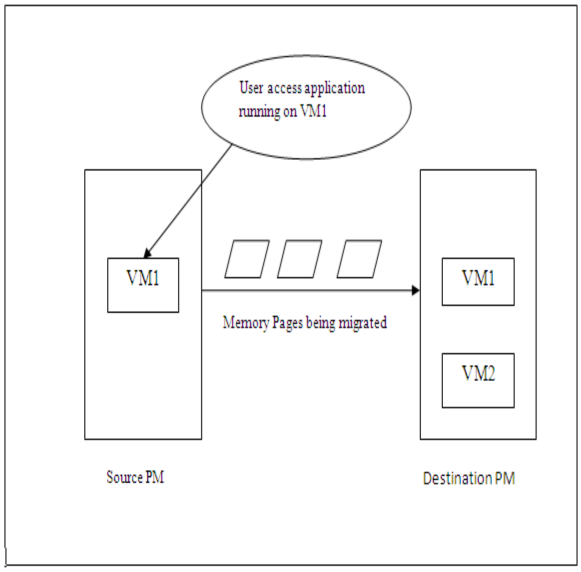

Figure 2: Live Migration

In this work, we propose an execution and architecture of OpenStack Neat. OpenStack Neat is a software extension to OpenStack that implements dynamic management of virtual machines in data centers using live migration. The main objective of this open source software is to consolidate virtual machines dynamically to increase the usage of computing resources and minimize the consumption of electrical energy by transferring the virtual machines using

ISSN: 2278 – 1323

live migration between different physical hosts with zero or minimal down time.

II. Related Work

Entropy [6] was the primitive free open source implementation of virtual machine consolidation resource manager for clusters. It uses constraint programming to implement dynamic consolidation of active virtual machines in data centers. Entropy project also considers the overhead of migration in to account, it selects migrations that are performed efficiently with low performance overhead. Entropy focuses on assigning the virtual machine to feasible nodes and also concentrates on migration of virtual machines between these nodes.

Entropy is implemented on top of Xen hypervisor and it focus on two main purpose : (1) In this phase it describes viable, i.e. a set of virtual machines with in the cluster and assigns enough computing resources to each virtual machine, and (2) optimal, i.e. executes feasible migrations which reduces the number of active hosts in data centers.

It periodically executes a two phase approach. In first phase, a constraint programming which finds optimal nodes that is necessary to place all virtual machines and a sample configuration that makes use of these nodes. In second phase it computes another optimal configuration that minimizes the number of active nodes and also reduces the cost of virtual migrations. Scalability of entropy depends on the number of nodes and virtual machines. Entropy explicitly takes the migration plan cost and can minimize the number of active servers and duration of migration significantly. But Entropy optimizes the placement of virtual machine periodically. Entropy provides the optimal solution for placement of virtual machines by computing the global solution. Every aspects of virtual machine placement decision are made by central controller, by that it limits the scalability of system.

Sandpiper [7] main objective is to balance load in virtualized datacenters using live migration of virtual machines. This system monitors and determines the hotspots and maps physical resources to virtual resources and does the proper migration of virtual machines. The aim of the system is to detect host overload conditions know as hot spots and migrate the virtual machines from overloaded servers to less loaded servers. Sandpiper applies a black box approach i.e. application agnostic means virtual machines are monitored from outside without knowing what type of applications running inside the virtual machine. Another approach called gray box approach is application specific where virtual machines are observed closely to get the data related to applications running inside the virtual machine. Sandpiper is proposed and implemented to accomplish either black box technology or gray box technology or combination of both.

Sandpiper applies hot spot sensing algorithms that find out when to migrate virtual machines from one physical node to other, and hot spot mitigation algorithm that decides what and where to migrate and also allocates enough resources after migration. A hot spot is observed when the resource usage of host machine exceeds some specified threshold. The overloaded virtual machine is moved to less loaded host to minimize the memory hot spots.

But it will not consolidate the virtual machines inside the physical server hence it will not minimize the number of active hosts inside the data centers and cannot save energy. More number of active hosts uses immense amount of electrical power leads to high operating costs at datacenters

III. System design

The main aim of OpenStack Neat project is to give an open source extensible software framework for consolidations of virtual machines dynamically in data centers and is works on OpenStack Platform. This framework includes the various implementations of different decision making algorithms. The OpenStack Neat software is developed and implemented as a transparent service to OpenStack. To install OpenStack Neat no need to modify or configure OpenStack Neat installation. Figure 3 provides the implementation of OpenSatck Neat with OpenStack clouds.

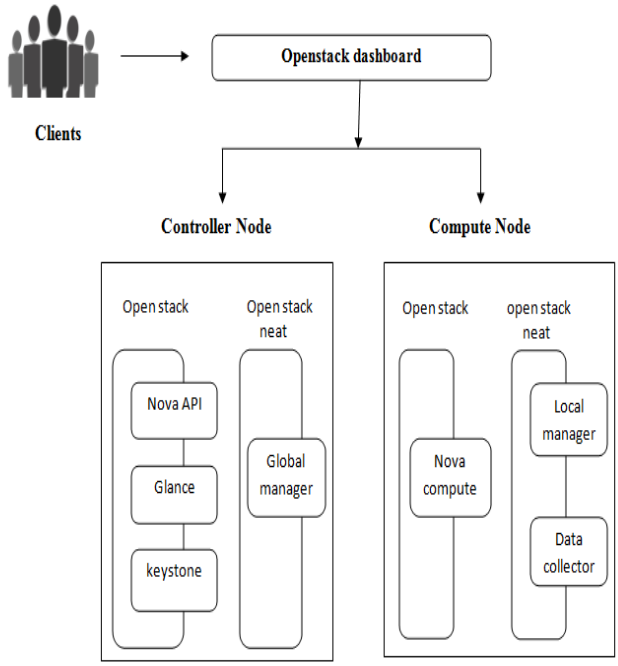

Figure 3: Deployment of OpenStack Neat

The design of OpenStack Neat framework has two major concerns:

1 It offers an open source extensible software framework for consolidation of virtual machines dynamically that employed to OpenStack clouds.

ISSN: 2278 – 1323

2 It also provides a framework for carrying research on consolidation of virtual machines dynamically.

The proposed approach is divided into three sub problems:

1. Determining if a physical host is considered as under loaded so that all the virtual machines are transferred from it, and host should be switched off.
2. Choosing the virtual machines to migrate.
3. Selected virtual machines are placed on other active physical hosts.

OpenStack Neat framework is framed of a number of components and data collectors. Some components are implemented on the compute node and some on controller node, which can possibly have many replicas. The system design is composed of main three modules:

1. Global Manager: This component implemented on the controller node in data center and takes global management decisions. It will make the decisions such as creating virtual machine instances at physical hosts and initiating live migration of virtual machines.
2. Local Manager: This component is implemented on all compute nodes and takes local decisions for virtual machine selection. It executes the algorithm to decide whether host is overloaded or under loaded and also chooses which virtual machine to migrate to other physical host.
3. Data Collector: This component is implemented on all compute nodes and is capable of collecting the information about resource utilization by virtual machines and hypervisors running on each compute node and stores the information in local database and also sent to the central database maintained at the controller node.

IV CONCLUSION

In this paper we have presented design of open source implementation of OpenStack Neat, a framework for dynamic management of virtual machines for server consolidation in IaaS clouds. Here the problem is divided in to 3 sub problems: host underload detection, virtual machine selection and virtual machine migration. The framework proposed is extensible and is transparent to the OpenStack installation and does not require any modification in OpenStack configuration. The results conclude that OpenStack Neat is able to minimize the energy consumption and also improve the utilization of resources at compute nodes. The performance overhead of the framework is very negligible and it brings significant energy savings with confined impact on the performance of application.

REFRENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G.Lee, D. Patterson, A. Rabkin, I. Stoica, and others, "A view of cloud computing,"Communications of the ACM, vol. 53, pp. 50–58, 2010.

[2] Gulshan Soni, Mala Kalra. "Comparative study of live virtual machine migration techniques in cloud", vol. 84-No 14. December 2013.

[3] http://searchservervirtualization.techtarget.com/definition/virtual-machine

[4] C Clark, KFraser, S Hand, J Hansen, and E Jul., "Live migration of virtual machines", Proceedings of the 2nd ACM/USENIX Symposium on Networked Systems Design and Implementation (NSDI), pages: 273-286, 2005

[5] Ashima Agarwal , Shangruff Raina " Live Migration of Virtual Machines in Cloud" International Journal of Scientific and Research Publications, Volume 2, Issue 6, June 2012.

[6] Hermenier F, Lorca X, Menaud J, Muller G, Lawall J. " Entropy: a consolidation manager for clusters" Proceedings of the 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments, Washington, DC,USA, 2009; 41-50.

[7] Wood T, Shenoy P, Venkataramani A, Yousif M." Black-box and gray-box strategies for virtual machine migration". Proceedings of the 4th USENIX Symposium on Networked Systems Design & Implementation, Cambridge,MA, USA, 2007; 229–242.

[8] Anton Beloglazov, Rajkumar Buyya. " OpenStack Neat: A Framework for Dynamic Consolidation of Virtual Machines in OpenStack Clouds – A Blueprint".

[9] Omar Sefraoui, Mohammed Aissaoui and Mohsine Eleuldj." OpenStack: Toward an Open-Source Solution for Cloud Computing", Volume 55 - No. 03, October 2012

[10] Sidharth Sachdev, Amit Mahajan. "Deployment of Private Cloud using Open Stack: An Open Source Cloud Computing Solution for Small and Mid-sized Organizations", Volume 3, Issue 10, October 2013.

ISSN: 2278 – 1323