Management Console for Open source Hypervisors and providing Cloud Support

Ajinkya Kshirsagar¹, Aamir Siddiqui², Neelhar Patil³, Harsh Shah⁴

*MAEER’s MIT College of Engineering, Pune, India
¹ajinkya@tesla.org.in
²aamir.siddiqui29@gmail.com
³neeharry@gmail.com
⁴shahmharsh@gmail.com

Abstract

Proprietary hypervisors exist for operating systems and enable connections with the Cloud to facilitate data transfer and connectivity. XEN, KVM are some of the hypervisors which have gained popularity among the developers over the years due to the flexible development options and community support that they entail. These hypervisors, being different in their specifications cannot interact with each other. Hence, the next logical step in their development is to create an interface through which, the admin can interact with different hypervisors present on the various virtual machines on the network. The management console will connect with individual hypervisors and then, create another virtual configuration from the resources available from these virtual machines through their respective hypervisors.

Keywords: ACM C2.4 Hypervisor, Cloud Computing, Virtualizations

1. INTRODUCTION

The concept of cloud computing and virtualization has gained much momentum and has become popular technology in IT field over the past few years. Infrastructure as a Service providers use virtualization to abstract their hardware and to create a dynamic data center. Virtualization enables the consolidation of virtual machines as well as the migration of them to other hosts during runtime. The usage of Hypervisors has seen a tremendous growth and has led to the development of their various flavors. Many hypervisors have been developed specific for some operating systems or for some particular tasks. The next logical step in the process of innovation is to create a unified hypervisor console that allows management of resources across the network and provides a abstract view to the end-client and a very detailed option preference to the admin user. This management console will provide the option to unite resources across platforms, bonding with their respective hypervisors and creating a virtual machine of desired configuration over the cloud according to the requirements of the admin.

Most of the organizations today require machines of configurations that alter on a very quick basis. Physically providing these machines and changing their hardware from time to time is a very inefficient process. Due to this, the concept of virtualization is implemented to create virtual machines of desired configurations without implicating costly hardware modifications. These Virtual machines are then deployed on the network so that they can be accessed through the various systems that are a part of it.

In this paper, a Management Console for open source hypervisors is proposed. This model is established using connectivity with the hypervisors and sharing of resources across the network. As a result, virtual configurations can be deployed across the network depending on the requirements and the specifications of the admin user.

2. RELATED WORK

2.1 Virtualization

Virtualization is the creation of a virtual (rather than actual) version of something, such as an operating system, a server, a storage device or network resources. Operating system virtualization is the use of software to allow a piece of hardware to run multiple operating system images at the same
time. The technology got its start on mainframes decades ago, allowing administrators to avoid wasting expensive processing power. In 2005, virtualization software was adopted faster than anyone imagined, including the experts. There are three areas of IT where virtualization is making head roads.[A] Network virtualization is a method of combining the available resources in a network by splitting up the available bandwidth into channels, each of which is independent from the others, and each of which can be assigned (or reassigned) to a particular server or device in real time. The idea is that virtualization disguises the true complexity of the network by separating it into manageable parts, much like your partitioned hard drive makes it easier to manage your files.[2] Storage virtualization is the pooling of physical storage from multiple network storage devices into what appears to be a single storage device that is managed from a central console. Storage virtualization is commonly used in storage area networks (SANs).

2.2 Hypervisors

A hypervisor, also called a virtual machine manager, is a program that allows multiple operating systems to share a single hardware host. Each operating system appears to have the host's processor, memory, and other resources all to itself. However, the hypervisor is actually controlling the host processor and resources, allocating what are needed to each operating system in turn and making sure that the guest operating systems (called virtual machines) cannot disrupt each other.

2.3 Cloud Computing

Cloud computing is a technology that uses the internet and central remote servers to maintain data and applications. Cloud computing allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. This technology allows for much more efficient computing by centralizing storage, memory, processing and bandwidth.

A simple example of cloud computing is Yahoo email, Gmail, or Hotmail etc. You do not need a software or a server to use them. All a consumer would need is just an internet connection and you can start sending emails. The server and email management software is all on the cloud (internet) and is totally managed by the cloud service provider Yeah, Google etc.

2.4 Libvirt

Libvirt is collection of software that provides a convenient way to manage virtual machines and other virtualization functionality, such as storage and network interface management. These software pieces include an API library, a daemon (libvirtd), and a command line utility (virsh). An primary goal of libvirt is to provide a single way to manage multiple different virtualization providers/hypervisors.

2.5 Eucalyptus

Eucalyptus is a software platform for the implementation of private cloud computing on computer clusters. There is an open-core enterprise edition and an open-source edition. Eucalyptus can use a variety of virtualization technologies including VMware, Xen and KVM hypervisors to implement the cloud abstractions it supports. Eucalyptus is an acronym for “Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems”.[7]

3. PROPOSED MODEL

3.1 Phases

The model will be implemented in 3 iterations or phases and each solution will act as the basis for expansion and deployment of the next phase. Each phase produces a functional prototype that has pre-defined objectives in very specific areas.

Phase 1: Create a management tool to manage different open source virtualization. Initially, the proposed plan for the project is to integrate and interact with just two popular hypervisors namely XEN and KVM.
Phase 2: Allow admin to create private cloud out of different virtualization solution. This solution will be customizable up to a certain degree that will allow the user to create applications characterized by a certain degree of flexibility and economy.

Phase 3: Provide plugin/api facility to integrate with any new virtualization player. This phase basically aims at creating an interface that will allow any perspective hypervisors to interact with the console and allow resource sharing, utilization and efficient organization of concerned hardware.

3.2 Modules

MCA (Management Console Agent) – This is the entity residing on each of the virtual machines that are located on the cloud and interact with the console server that provides it with execution details.

MCS (Management Console Server) – This is a web based entity that exhibits the management console, actually giving directions to execution to the virtual machines / hypervisors through connection with the MCAs.

DB - This is an online database that maintains a repository of the virtual machines that currently exist on the cloud and their current status along with their functional capabilities.

Client Machines/Users - These machines access the MCS through web based services and request status information and control mechanisms over the virtual machines present over the cloud to allow some semblance of control over the hypervisors present on each of the virtual machines.

XEN/KVM - Hypervisors

In this software, all the open source hypervisors (KVM and XEN) are brought under one platform that should find interest from the data administrators all over the world. The software has huge possibility of development with the additions of more and more hypervisors under one platform. Compared to single hypervisors, this software has an advantage as it provides a common view of all different types of virtual machines running on the network. The software is designed to expand to even paid hypervisors with a plugin/api facility to integrate with any new virtualization player.

5. FUTURE ENHANCEMENT

The project has immense scope for future work. Similar projects with marginal capabilities are already in research phases in leading virtualization giants like Vmware, Red Hat and SUSE. Cloud computing is already occupying prime importance in corporate fields and will continue to do so for the next decade.

The main challenge of this project that could be overcome in future would be to provide universal flexibility that will ensure uniform access to all the hypervisors and do so efficiently. The virtualization should also be taken care of in a more secure way so as to maintain system integrity and log.

The concept can be expanded to limitless domains and adopted in all fields. Corporate applications are particularly attractive since this will save the cost of customized hardware that companies need to invest in. Though presently in development, no open source version of this product is currently available. This product will be very beneficial to the developer’s community in creating better application base for future endeavors.

4. CONCLUSION

6. REFERENCES:


