Volume 2, Issue 12, December 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Designs of Virtual file system in Cloud Environment

T.Gopikiran[1], B.Rajkumar[2], S.Satyanarayana[3], K. Ashok chakravarthi[4]
[1,2,4] M.Tech(CSE)Students,
[3] Assoc. Professor of CSE,
KL University,
Guntur, Andhra Pradesh, India

Abstract: In this paper emphasis the virtual file system for storage of virtual image. The virtual machines raises the new challenges in the design and development of IaaS . The cloud middleware interacts with deployment and snapshot ting on clouds in a Amazon EC2 fashion. In this virtualization is also a technology based on cloud service providers and able to provide the data from data centers to users. The deploying the VM instances and to gather the number of users. To more challenge is to snapshot multiple images and persist them supervision tasks like stop VM`s temporarily and resume when required. This paper proposed new virtual file system for optimization of multi-deployment and multi-snapshot ting effectively.

Keywords: AmazonEC2, virtual file system, multi-deployment, multi-snapshot ting, virtual machine.

1. INTRODUCTION

In this decade, Cloud computing is an increasingly popular paradigm for accessing computing resources. Infrastructure as a Service (IaaS) is the delivery of hardware (server, storage and network), and associated software (operating systems virtualization technology, file system), as a service. The foundation of any cloud computing stack begins with its infrastructure. IaaS cloud computing is the delivery of computing on demand as a shared service, avoiding the cost of investing in, operating and maintaining the hardware. In order to be fully functional, the infrastructure must be reliable and flexible to allow easy implementation and operation of applications running within the cloud. Leasing of computation time can be achieved by enabling users to deploy virtual machines (VMs) on the resources of the datacenter. The on-demand nature of IaaS is grave to make such leases attractive, it enables the users to shrink their resources according to their computational uses. This rising model fallout in new challenges relating to design and development of IaaS system. One along with regularly resulting pattern in the operation of IaaS is the necessity for Deploying a huge number of VMs on most of the nodes relative to a datacenter at the same instant of time. Starting from set of VM images earlier and stored in importunate fashion. For example this pattern occurs when the user wants to deploy a virtual clusters that executes in a distributed application or a group of environments for supporting a workflow. So, we refer this pattern as multi-deployment.

Once the virtual machine instances are being run, a similar type of challenge is applies to snapshot ting the deployment. The majority of the VM images which were changed locally need to be transferred in a concurrent manner for making storage stable with the reason to capture the VM state for later use(i.e for instance in check pointing or online migration to another cluster or cloud). This pattern is referred to as multi-snapshot ting. The technique of conventional snapshot ting rely on VM image file formate to store in only incremental differences in a new file which rely on the original VM image similar to backing file.

2. Cloud Infrastructure

In recent years, infrastructure-as-a-service (IaaS) cloud computing [14] has emerged as an attractive alternative to the acquisition and management of physical resources. The on demand provisioning it supports allows users to elastically expand and contract the resources available to them based on an immediate need a pattern that enables rapid turnaround when dealing with emergencies, working toward deadlines, or growing an institutional resource base. This pattern makes it convenient for institutions to configure private clouds that allow their users a seamless or near-seamless transition to community or commercial clouds supporting compatible virtual machine (VM) images and cloud interfaces. Such private clouds are typically configured by using open source IaaS implementations. To ensure on-demand availability, a provider needs to overprovision: keep a large proportion of nodes idle so that they can be used to satisfy an on-demand request, which could come at any time. The need to keep all these nodes idle leads to low utilization. The only way to improve it is to keep fewer nodes idle. But this means potentially rejecting a higher proportion of request to a point at which a provider no longer provides on-demand computing. This situation is particularly hard to accept in the world of scientific computing where the use of batch schedulers typically ensures high utilization [6] and thus much better resource amortization. Thus, potential low utilization constitutes a significant potential obstacle to the adoption of cloud computing in the scientific world. We propose a cloud infrastructure that combines on demand allocation of resources with opportunistic provisioning of cycles from idle cloud nodes to other processes, such as HTC, by deploying backfill VMs. In such an infrastructure, backfill VMs are deployed on idle cloud nodes and can be configured to perform any desired function. A backfill VM is terminated when the resource is needed to satisfy an on demand request. If we can ensure that the computation occurring in backfill VMs is resilient to such sudden termination, the time that would otherwise be idle can be profitably spent. Furthermore, cycles via backfill VMs can be provided to users at a lower cost than on-demand VMs because of the cloud provider's ability to terminate the instances when needed. Thus, for users who work with HTC resources and possibly expect such behavior already, backfill VMs would provide a less expensive option when moving their workloads to the cloud. Overall, this design achieves two goals: for cloud providers, it offers a path to higher utilized clouds; for cloud users, it offers another type of resource lease, potentially cheaper than on-demand, non perceptible resource.

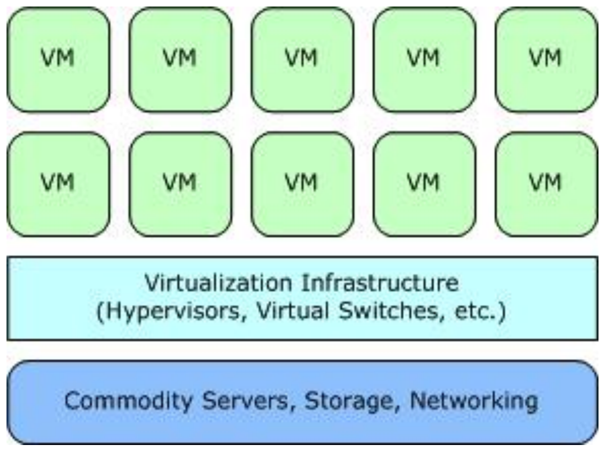

Fig: Cloud infracture with VM

3. Virtualization

Virtualization, in its broadest sense is the emulation of one or more workstations/servers, within a single physical computer. In other words, it is the emulation of hardware within a software platform. This type of virtualization is sometimes referred to as full virtualization and it allows one physical computer to share its resources across a multitude of environments. This means that a single computer can essentially take the role of multiple computers. Virtualization is not only limited to the simulation of entire machines. In fact there are many different kinds of virtualization. One of these is in use by almost all modern machines today and is referred to as virtual memory. Although the physical locations of data may be scattered across a computers RAM and Hard Drive, the process of virtual memory makes it appear that the data is stored contiguously and in order. RAID (Redundant Array of Independent Disks) is also a form of virtualization along with disk partitioning, processor virtualization and many more virtualization techniques. Whenever a business needs to expand its number of workstations or servers, it is often a lengthy and costly process. An organization first has to make room for the physical location of the machines. The new machines then have to be ordered in, setup, etc. This is a time consuming process and wastes a business's resources both directly and indirectly. Virtual machines can be easily setup. There are no additional hardware costs, no need for extra physical space and no need to wait around wasting time. Virtual machine management software also makes it easier for administrators to setup virtual machines and control access to particular resources, etc.

4 . Virtual File System

A **virtual file system (VFS)** or **virtual file system switch** is an abstraction layer on top of a more concrete file system. The purpose of a VFS is to allow client applications to access different types of concrete file systems in a uniform way. A VFS can, for example, be used to access local and network storage devices transparently without the client application noticing the difference. It can be used to bridge the differences in Windows, Mac OS and Unix file systems, so that applications can access files on local file systems of those types without having to know what type of file system they are accessing. A VFS specifies an interface (or a "contract") between the kernel and a concrete file system. Therefore, it is easy to add support for new file system types to the kernel simply by fulfilling the contract. The terms of the contract might change incompatibly from release to release, which would require that concrete file system support be recompiled, and possibly modified before recompilation, to allow it to work with a new release of the operating system; or the supplier of the operating system might make only backward-compatible changes to the contract, so that concrete file system support built for a given release of the operating system would work with future versions of the operating system.

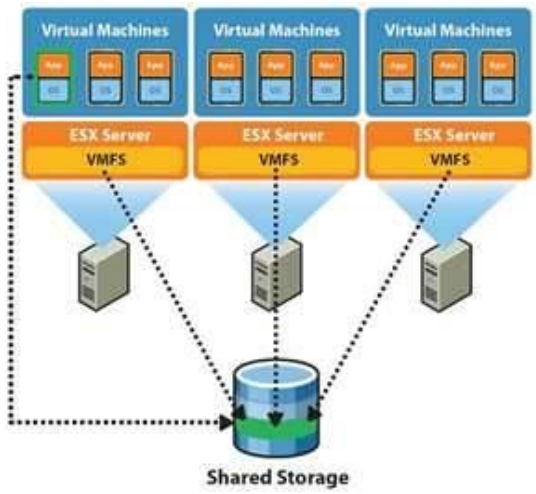

Fig: virtual file system

5. Deployment of a multi-cloud virtual cluster

The distributed cluster test bed used in this work deployed of top of a multi-cloud infrastructure. This kind of multi-cloud deployment involves several challenges, related to the lack of a cloud interface standard; the distribution and management of the service master images; and the interconnection links between the service components. A brief discussion of these issues and the main design decisions adopted in this work to face up these challenges are included in the Appendix A of the supplemental material.

Our experimental test bed starts from a virtual cluster deployed in our local data center, with a queuing system managed by Sun Grid Engine (SGE) software, and consisting of a cluster front-end (SGE master) and a fixed number of virtual worker nodes (four nodes in this setup). This cluster can be scaled-out by deploying new virtual worker nodes on remote clouds. The cloud providers considered in this work are Amazon EC2 (Europe and USA zones) and Elastic Hosts.

6. Virtual machine

Software can be used to implement a virtual machine that runs inside a physical machine. The virtual machine can simulate real hardware or it can be a machine that has no physical counterpart. The Java Virtual Machine is an example for the final. Just like a physical machine, the purpose of a virtual machine is to run code compile for it. If an operating system runs inside the virtual machine, this operating system is called a guest operating system and the operating system of the physical machine hosting the virtual machine is called a host operating system.

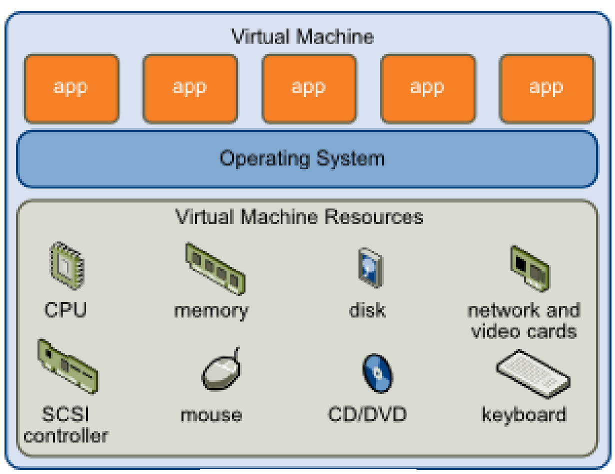

Fig: virtual machine

7. Virtual machine deployment problem

The benefits of the cloud computing paradigm are flexibility. This means that resources can be provisioned quickly and efficiently. If the process of deploying a virtual machine on a remote physical host takes a long time, some of this flexibility has been lost. The problem is not just a matter of having to wait for the virtual machine to become operational, since the deployment process itself takes up resources. Network resources are used to transfer data that is necessary to boot the virtual machine and disk space is taken up on the physical hosts to store that data. While the deployment is in progress, this resource usage does not produce any direct benefit. In other words, the resources are not used to work on the task for which the virtual machine is being deployed.

8. Related work

In previous days transferring the image data and making that data available to virtual machine. That evolution can be done in both general terms and cloud environment. This specific environment is a relatively small cloud system with 72 physical machines. To find the best method of deployment given the size of the system and the performance of the network that is used for the deployments. The basic transfer methods we use are BitTorrent and multicast.

8.1 Bit torrent:

BitTorrent is a protocol for rapid distribution of files over local or wide area networks. It leverages the upload capacity of downloading hosts to speed up transfers and to scale to a large number of hosts. Files that are to be served are divided into multiple pieces. These pieces are then served to a number of receivers from the initial source of the file. Once a receiver has finished downloading a given piece, it can start uploading it to the other receivers. This increases the total upload capacity of the system. BitTorrent uses separate tracker software to allow receiving hosts to find other receiving hosts. The tracker is a server that has information about which hosts are downloading which files. More recently, distributed ways of tracking other downloading hosts have been added, making the tracker an optional component. One example is distributed hash tables. Metadata about the files to be shared is contained in a so called torrent file. This file contains hashes for the pieces of the files. These can be used to check the integrity of pieces before distributing them to other hosts. It also contains information about the tracker. Previous research on the scalability of BitTorrent has shown that it scales well to a very large number of simultaneous peers. It works well in very heterogeneous environments. For example, Izal et al. studied one publicly available torrent that was downloaded by a wide variety of users using many different connection types and found out that the users were mostly able to utilize the bandwidth available to them effectively.

8.2 Multi cast:

Multicast is a routing scheme in which data is sent only once by the sender and is then forwarded to a group of receivers. Unlike in broadcasting, packets that are multicast are only forwarded to hosts as necessary and not to all possible hosts. Any transfer protocol that is used for file transfer must provide consistency and correct ordering of data. As multicast is typically a best effort protocol, these functions need to be implemented separately. There are various methods for providing consistent multicast, and the best way depends on that situation in which multicast is being used. The scalability of any multicast _le transfer tool depends on the reliability mechanism it uses. One very scalable reliability mechanism is to send redundant data along with the _les to be sent. Packet losses can then be recovered by using the redundant data. This scheme scales similarly to unreliable transmission, as it is almost exactly the same case except for the slightly increased amount of transmitted data. However, if more packets are lost than can be recovered using the redundant data, there is no way for the receivers to notify the sender that retransmission is necessary, and the data is irrecoverably lost.

9. Conclusions

As cloud computing becomes popular, efficiency of deployment is very extensively. So, the multi-deployment of images can be deploy by using virtual file system, virtual machine also propagation to compute the nodes and snapshotting . The performance of these virtual file system effects the usability of the benefits offered by cloud computing. This paper introduces a technique virtual file system in place of virtual machine and integrate with cloud middleware to efficiently handle cloud deployment. Based on these techinique we are explore the multi-deployment effectively. The bit torrent and multi cast also possible to deploy the images in that cloud environment. High performance computing application also used to deploy in the real world.

REFERENCES

[1] Amazon elastic block storage(ebs) http://aws.amazon.com/ebs/.

[2] Infrastructure as a service(IaaS)http://klucloudseminar.weebly.com/iaas.html

[3] RistoLaurikainen, Master's Thesis"Improvingthe eciency of deploying virtual machines in a cloud environment" january 2012.

[4] Bogdan Nicolae,John Bresnahan,Kate Keahey," Going back and forth: efficient multideployment and multisnapshotting on clouds"The 20th international ACM symposium on high-performance parallel and distributed computing. 2011

[5] Syeda Farthath Begum, Dr.Kahalid Mohiuddin and Ashique Rasool Mohammad," Going back and forth: efficient multideployment and multisnapshotting on clouds"Global journals Inc july 2012.

[6] Virtual machine in Wikipedia. http://en.wikipedia.org/wiki/Virtual_machine.

[7] S.Satyanarayana, T.Gopikiran, B.Rajkumar. "Cloud Business Intelligence" international journal of advanced and innovative research (ijair) volume 1 issue 6 PP(43-52) ,2012.

[8] S.Satyanarayana, "Design on Security of Hybrid Cloud Business Intelligence System" international journal of computational mathematical ideas(ijcmi), VOL 4-NO3-PP 1019-1024 (2012).

ACKNOWLEDGEMNT

This work was supported by our research guide Dr.S.Satyanarayana is contributed of my own work I thank them for my development in research work and supported by KL university.

Authors Profile:

1.TadiboyinaGopikiran studying M.Tech (CSE) in KLUniversity.Published one international journal paper and attended presented research papers in 2 international conferences and member of IACSIT(Singapore), SDIWC.My research on cloud computing, Business Intelligence and software engineering.

2.**BatchuRajkumar** studying M.Tech(CSE) in KLUniversity.Published one international journal paper and attended presented research papers in 2 international conferences and member of SDIWC. My research on cloud computing, Business Intelligence and software engineering.

3.**Dr.S.Satyanarayana** is Distinguished scientist, Professor , inventor, author, and Business leader Born in India. He received his M.Tech(CSE) & PhD (Computational Mathematics) & Ph.D(Computer Science & Engineering) from Acharya Nagarjuna University & Capitol University and CMJ University. Dr.S.Satyanarayana early scientific work was mainly Graph Theory, Software Engineering, Cloud Computing Cryptography, SAS, Business Intelligence and Artificial Neural Networks. He Published More than 30 Research Papers & 3 Books in International Journals& Publications. He is a member of ,IAENG(Hong Kong),IACSIT(Singapore), Chief Technical Advisory Board of IJSCE, IJRTE, Member of IAENG Society of Artificial Intelligence, Bioinformatics, Chemical Engineering, Computer Science, Data Mining, Electrical Engineering, Internet Computing and Web Services, Imaging Engineering, Industrial Engineering, Information System Engineering, Mechanical Engineering, Operations Research, Scientific Computing, Software Engineering, Wireless Networks, Editor in chief of IJCMI,IJASEI, IJAII,IJEI,IJCCTI,Associate Editor of Research Journal of Applied Sciences, Engineering and Technology, IJSNDC,IJSIEC,Board Member of IJCTT, IJCIIS,IJMTT,IJCOT,IJEAT, IJCST,IJART, IJARCSEE and senior member of Universal Association of Computer and Electronics Engineers London, IACSIT(Singapore),presently he is working Assoc.professor of CSE in KLUniversity.

4. Kolla Ashok Chakravarthi studying M.Tech(CSE) in KLUniversity.Published one paper on International journal of computer science information and technology(IJCSIT) and member of SDIWC.My research on cloud networks.

