Snapshot Based Virtualization Mechanism for Cloud Computing

Prof.P.Pradeep Kumar[1], A.Rajashekar Reddy[2] and A.Rupa[3]

[1]**Head,Department of CSE,VIVEKANANDA INSTITUTE OF TECHNOLOGY&SCIENCE(N6)**
Karimnagar, Andhra Pradesh,505 001,India

[2] **Assistant Professor,VIVEKANANDA INSTITUTE OF TECHNOLOGY&SCIENCE(N6)**
Karimnagar, Andhra Pradesh,505 001,India

[3] **M.Tech,VIVEKANANDA INSTITUTE OF TECHNOLOGY&SCIENCE(N6)**
Karimnagar, Andhra Pradesh,505 001,India

Abstract

Virtualization in cloud computing has been the latest evolutionary technology in current applications of various industries and IT firms are adopting Cloud Technology. The concept of cloud computing was introduced long back. Since its inception there have been many number of new innovations implemented by different experts and researchers etc. Virtualization in cloud computing is very effective approach to gain different operational advantages in cloud computing. In this paper we have proposed the concept of virtualization using *Snapshot* based Mechanism, where the Memory virtualization and Storage virtualization are discussed in this paper.

Keywords: *Cloud Computing, Storage Virtualization, Memory Virtualization, Virtual data centers, Storage pools, Snapshot, Virtual private cloud.*

1. Introduction

The Cloud references a distributed collection of computing resources where the applications can reside anywhere on the accessible networks. In the cloud, a large pool of accessible virtualized resources such as hardware, development platforms, and ideally services, can be dynamically reconfigured to adjust to scalable load, with minimal management effort Cloud Computing actually covers more than just computing technology.

Cloud computing can be defined as a new approach of computing in which dynamically scalable and often virtualized resources are provided as a services over the Internet, With the cloud computing technology, users use a variety of devices, including PCs, laptops, smartphones, and PDAs to access programs, storage, and application-development platforms over the Internet, via services offered by cloud computing providers which provides cost savings, high availability, and easy scalability. It is a new model for providing business and IT services. Managing applications and distribution becomes a very steep task for IT departments. Installation mechanisms differ from application to application. Some programs require certain helper applications or frameworks, and these applications or frameworks may conflict with existing applications or new applications. Additionally, one-off applications exist for special users. The cloud computing-based high-performance computing center aims to solve the following problems:

- High-performance computing platform generated dynamically
- Virtualized computing resources
- High-performance computer management technology combined with tradition ones

High-performance computing requires a new way of resource supply, in which the platform should be dynamically generated according to the needs of every end user and application software. In the cloud computing-based high-performance computing center, the virtualization of physical resources can be realized through the Cloud platform virtualized resources can be used to establish high-performance computing platform and generate high-performance computing environment whose scale is larger than that of the actual physical resource so as to meet the requirements of customers. *Virtualization* allows the abstraction of computing, storage, and networking resources from underlying infrastructure. It shields the users from the knowledge of the underlying resources and thus reduces the required skill level to operate these applications. Data and information are two essential elements of an enterprise's Cloud transformation project. They hold the ultimate metrics for whether or not the project can be successful. Virtual instances were accessed in a thin-client model by green-screen terminals. This mode of access can be seen as a direct analog of the concept of virtualized instances in the cloud, although then a single machine was divided among users. In the cloud, it's potentially many thousands of machines. The scarcity of the computing resource in the past drove the virtualization of that resource so that it could be shared, whereas now, the desire to fully utilize physical compute resources is driving cloud virtualization. *Virtual data centers* provide the insulation that sets one organization's virtual resources apart from those of other organizations and from the underlying physical infrastructure. Software Virtualization Provides key benefits as

- **Client Deployments Become Easier:** copying a file to a workstation or linking to a file in a network share can install Virtual applications. Existing deployment methodology can be leveraged to automate this functionality.

- **Added Security:** Many software virtualization providers include the ability to link to LDAP/Active Directory group membership to ensure that you are able to run the software. This ensures approved users are granted access, and the software cannot be run on a machine that does not have access to the LDAP/Active Directory domain. Plus, time bomb functions may exist that expire the software after a specified amount of time.

- **Ease of Management:** Managing updates becomes a much simpler task. Update one place; deploy the updated virtual application to the clients. If the update breaks something, just copy the original file back in place. Suddenly, it becomes possible to have a library of updated software for versioning and roll back functionality.

- **Software Migrations:** Moving users from one software platform to another takes much time and consideration for deploying, and impact on end user systems. By running a virtualized software environment, the migration can be as simple as replacing one file with another.

- **Conflict Mitigation with Existing Software:** Due to the fact that software is housed in virtualized containers, applications that do not play nicely with each other can co-exist on the same system. This is very useful for developers testing different software versions or running multiple versions of web browsers to verify application functionality.

Virtual applications collect those resources into separate manageable units. Cloud data centers represent the new generation of data centers that promote on-demand provisioning of computing resources and services, Datacenter monitoring. Event Capturing: Applications, OS, servers, network devices can generate formidable amount of events, which makes directly storing and searching these events infeasible. To address this issue, Bhatia et al. [2] proposed Chopstix, a tool that uses approximate data collection techniques to efficiently collect a rich set of system-wide data in large-scale production systems.

- Resource Consumption: Servers usually have limited resources available for monitoring. Assigning monitoring tasks and organizing monitoring overlays without considering this fact may lead to unreliable monitoring results.

2. Virtualization

Virtualized environments are easier to manage, allowing easier deployments of new software, less maintenance because there are less servers, and rapid replication and workload balance activities. Virtual machines use machine memory for two purposes: each virtual machine requires its own memory and the virtual machine monitor (VMM) requires some memory and a dynamic overhead memory for its code and data. The virtual and physical memory space is divided into blocks called pages. When physical memory is full, the data for virtual pages that are not present in physical memory are stored on disk

2.1 Memory Virtualization

Each virtual machine consumes memory based on its configured size, plus additional overhead memory for virtualization. The configured size is a construct maintained by the virtualization layer for the virtual machine. It is the amount of memory that is presented to the guest operating system, but it is independent of the amount of physical RAM that is allocated to the virtual machine, which depends on the resource settings (shares, reservation, limit), memory virtualization is seen as virtual memory, or swap, on servers and workstations. Conceptually, swap exists as a way to handle memory-full systems without having to halt, or even kill, processes. Swap is a portion of the local storage environment that is designated as memory to the host system. The host sees the local swap as additional addressable memory locations and does not delineate between RAM and swap. However, the swap file is addressed at the upper bounds of the memory addressing, so the physical memory will be consumed before the swap is consumed. Using swap imposes major performance degradation on the host system. Memory Virtualization provides some interesting memory related functions.

These functions include:

- The ability to share common memory pages across multiple virtual machines. This is great for when a host is running multiple copies of the same operating system. There is no need for multiple copies of the same pages to exist. Sharing the pages frees up memory to use elsewhere.
- The ability to snapshoot a memory state and revert back if the new state is not optimal.
- The ability to transmit the memory state across the network to another host in order to move virtual machine operations to the new host.
- Compress physical memory contents in order to save the physical host from utilizing swap.
- Releasing unused, but allocated, memory for other virtual machines to utilize.

2.2 Storage Virtualization

In Storage Virtualization mechanism, a strong link between the physical host and the locally installed storage devices exists. This paradigm is changing drastically, almost to the point that local storage is no longer needed. As technology progresses, more advanced storage devices are coming to the market that provide more functionality, and serve to obsolete local storage. Storage virtualization is a major component in storage best practices for servers, in the form of controllers and functional RAID levels. Operating systems and applications with raw device access prefer to write directly to the disks themselves. The controllers configure the local storage in RAID groups and present the storage to the operating system as a volume (or multiple volumes, depending on the configuration). The operating system issues storage commands to the volumes, thinking that it is writing directly to the disk.

Storage virtualization is becoming more and more present in various other forms like:

- *File servers:* The operating system is writing to a remote location with no need to understand how to write to the physical media.
- *pNFS:* A component of NFS v4.1, pNFS involves making a request for data over an NFS share. However, the data is stored in a large variety of disparate locations and medium. The requester has no idea where the data exists; that is handled by the NFS server.
- *DFS:* Similar in concept to pNFS, DFS, Distributed File System, creates a file system-like view of data. However, the composition of the file system is differing file shares on the network. The file system appears to be a single volume, but it is comprised of multiple locations.
- *WAN Accelerators:* Rather than send multiple copies of the same data over the WAN environment, WAN accelerators will cache data locally and present the re-requested blocks at LAN speed, while not impacting the WAN performance.
- *NAS and SAN:* Storage is presented over the Ethernet network to the operating system. NAS presents storage as file operations (like NFS and CIFS). SAN technologies present storage as block level storage (like iSCSI and Fibre Channel). SAN technologies receive operating instructions as if the storage was a locally attached device
- *Storage Pools:* Enterprise level storage devices can aggregate common storage devices, in the form of like disk types (speeds and capacity), to present an abstracted view of the storage environment for administrators to handle. The storage device handles which disks to place the data upon, versus the storage administrator deciding how to divide the available disks. This usually leads to higher reliability and performance as more disks are used.
- *Storage Tiering:* Utilizing the storage pool concept as a stepping stone, storage tiering analyzes most commonly used data and places it on the highest performing storage pool. The lowest used data is placed on the weakest performing storage pool. This operation is done automatically and without any interruption of service to the data consumer.

Benefits to storage virtualization include:

- Data is stored in more convenient locations away from the specific host. In the event of a host failure, the data is not necessarily compromised.
- The storage devices are able to perform advanced functions like anti-duplication, replication, thin provisioning, and disaster recovery functionality.

- By abstracting the storage level, IT operations can become more flexible in how storage is partitioned, provided, and protected.

3. Snapshot Mechanism

A memory snapshot represents the memory state of the profiled application at the moment it was captured. It contains information about all loaded classes, about all existing objects, and about references between objects. The key benefit of snapshot is to support atomic multi-word read operations that produce a consistent view on a large dataset. With snapshot, concurrent programs read a large dataset atomically and work with a consistent snapshot image of the dataset without synchronization code. A snapshot to read a multi-word dataset atomically and to process it concurrently in multithreading environment. The snapshot image is isolated from further memory updates,shared by multiple threads, and accessed with normal load/store instructions

3.1 Hardware-assisted Memory Snapshot

- Memory snapshot provides support for atomic multi-word read operations. A "snapshot" of m memory elements is created to provide a consistent view for p processors. Then, the processors are allowed to execute two operations: update to write a memory element in the snapshot and scan to read memory elements. The scan operation is an atomic read operation on the memory elements and produces a consistent copy of them at the moment the scan is executed. Memory snapshot solves the race problem easily without synchronization .A memory snapshot is taken (i.e., scan) on all nodes of the graph. The profiling thread reads the pointer from node C to node D in the snapshot image while the application thread modifies the pointer in the up-to-date image. Unfortunately, despite the great potential as a programming primitive for concurrent programming, memory snapshots have been implemented in software and have performance issues such as O(mp) update time or O(m) scan time, which prevent them from being adopted in a wide range of applications. We propose fast memory snapshot with hardware assistance for easy concurrent programming without synchronization code. It accelerates snapshot operations with hardware resources to provide O(1) update (as fast as a single memory write operation) and O(p) scan in O(m) space. The O(1) update time enables application threads to write to the up-to-date image without performance degradation. The O(p) scan time allows snapshots to be used for large datasets in performance-oriented programs. We use the cache as a buffer for snapshot data and additional bits per cache line for snapshot metadata. A consistent snapshot image is taken with inter process communication and maintained with cache coherence protocol support. Using hardware acceleration, memory snapshot can be applied to performance-oriented software system, including the following:
- *Fast Concurrent Checkpoint*: Process state is checkpointed for backward recovery with checkpoint in fault tolerance mechanisms [5] or for guest OS migration in virtual machines. The fast memory snapshot is used to create the backup memory image concurrently without slowing down the applications (due to O(1) update time). Logging threads run in parallel with application threads to log the check pointed image into disks.
- *Concurrent Garbage Collection*: Concurrent garbage collection typically incorporates sophisticated algorithms to deal with races between mutators and collectors and increases code management cost. With fast memory snapshots, concurrent garbage collection can be as simple as a stop-the world garbage collector by taking a snapshot of part of the heap and collecting garbage from the snapshot image concurrently.
- *Concurrent Memory Profiling*: As shown in the example in the previous section, concurrent memory profilers benefit from fast memory snapshot by traversing a consistent object reference graph in the snapshot image.
- *Concurrent Call-Path Profiling*: Just-In-Time (JIT) compilers in Java virtual machines and the C# runtime system
- Optimize the application code by finding hot execution paths with call-path profiling. With fast memory snapshot,
- Compilers take a snapshot of a thread stack periodically and analyze the stack in parallel with application threads.
- *Fast Copy-On-Write (COW)*: COW is used for shared virtual to physical page mappings in fast fork() and for disk block sharing between virtual machines

Memory forensics is growing concern. Insider attack and information leak have become a serious problem. VMM provides powerful facilities to access all states of guest VM: CPU context, memory and other block devices. Timely snapshot is important for evidence retrieval. If snapshot is taken after incident has happened, usually much of evidence has been lost. In this paper we propose an incident-driven memory snapshot for full-virtualized OS using interruptive debugging techniques. With proper modification of VMM, we can obtain memory snapshot just in time when the incident happens. We modify debug register handler for this purpose. When the incident has been occurred on guest OS, it is notified by changing debug register. An interruption generator is inserted into register handler of hypervisor. Then, host OS receives notification and takes snapshot. Snapshot does not increase with the size of the data set, whereas the same for a direct backup is proportional to the size of the data set. In some systems once the initial snapshot is taken of a data set, subsequent snapshots copy the changed data only, and use a system of pointers to reference the initial snapshot. This method of pointer-based snapshots consumes less disk capacity than if the data set was repeatedly cloned.

4. Figures

Figure 1 Cloud Computing Model

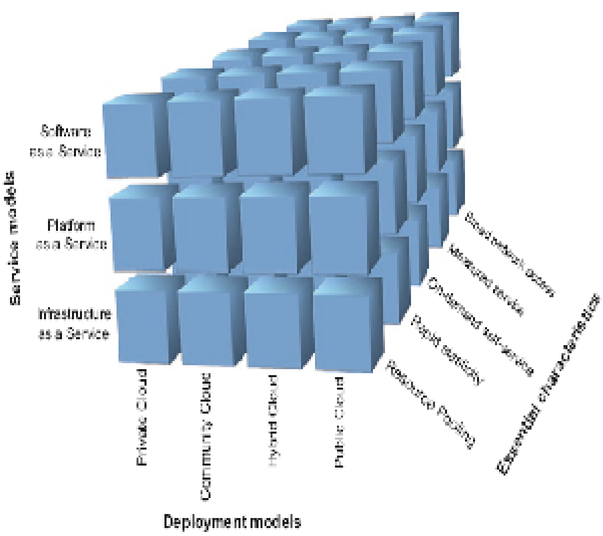
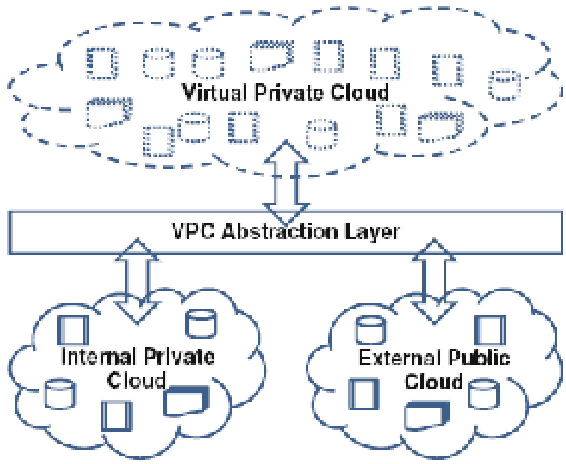


Figure 2 Virtual Private Cloud Implementation Profile



5. Conclusions

Virtualization overall, irrespective of the type, helps improve scalability and resource utilization. In most cases, the main benefit to IT professionals is the ease of management, as virtualization helps to centralize administrative tasks, whether they involve day-to-day updates or large scale deployments and migrations. The Snapshot virtualization provides hardware virtualization, Software virtualizations, Data Virtualization as a part of Storage virtualization which includes all the required resources in the cloud service, it requires vast hardware based and Data based in depth Analysis to prove the significance and reliability of the proposed mechanism.

.

References

[1] Bowers, K. D., Juels, A., & Oprea, A. (November 2009). HAIL: A high-availability and integrity layer for cloud storage. *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS), ACM, Chicago, IL, USA*, 187–198.

[2] Cohen, R. (May 2008). *Virtual private cloud*, *Elastic vapor: Life in the cloud.* Retrieved in January 2010, from http://www.elasticvapor.com/2008/05/virtual-private-cloud-vpc.html.

[3] Chen, H., Chen, J., Mao, W., & Yan, F. (June 2007). Daonity – Grid security from two levels of virtualization. *Elsevier Journal of Information Security Technical Report, 12*(3), 123–138.

[4] Garfinkel, T., Pfaff, B., Chow, J., Rosenblum, M., & Boneh, D. (December 2003). Terra: A virtual machine-based platform for trusted computing. *ACM SIGOPS Operating Systems Review,37*(5), 193–206.

[5] Jothy RosenbeRg- The Cloud at your Service.

[6] Bill hill- Intro to Virtualization: Hardware, Software, Memory, Storage, Data and Network Virtualization Defined.

[7] Jaewoong Chung -Fast Memory Snapshot for Concurrent Programming without Synchronization.

[8] http://www.yourkit.com/docs/80/help/memory_snapshots.jsp

Professor P. Pradeep Kumar, having overall 18+ years academic & Industry work Experience in reputed organizations.Presently working as Prof & Head of the department, Computer Science & Engineering at Vivekananda Institute of Technology & Science, Karimnagar, AP, India

He holds a bachelor degree in Computer Science & Engineering from Nagpur University,Maharastra and Masters degree in Computer Science & Engineering from Osmania University, Hyderabad,AP, as a faculty/ trainer , he progressed to the Prof & Head of the department of Computer Science & Engineering. He specializes in Programming Languages C/C++, RDBMS, Which he taught for more than 12 years out of a rich teaching experience of more than 18+ years, along with Computer Science & Engineering subjects for UG/PG Engineering Courses. His research interest in the areas of Image Processing, Distributed Databases, Cloud Computing. He has more than 12+ publications in National and International Journals and Conferences to his credit. IT Director - Cerebrum Software Solutions Pvt Ltd, Hyderabad. Member Advisory Board- Syndo Health Pvt Ltd,Hyderabad He is a leading Education Consultant in Regional News Paper.

A.Rajashekar Reddy is working as Assistant Professor in VITS,Karimnagar . He has hands on experience in developing various applications in Microsoft .NET framework. His interest area is Distributed Computing especially Cloud computing.

A.Rupa is pursuing M.Tech in VITS,Karimnagar under the guidance of first author i.e., Prof. P. Pradeep Kumar.