International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 6, ISSN No. 2455-2143, Pages 77-80
Published Online April - May 2016 in IJEAST (http://www.ijeast.com)

LIVE VIRTUAL MACHINE MIGRATION IN CLOUD ENVIRONMENT

Kanchan Rani
M.Tech C.S.E (big data)
Pit Kapurthala (main campus PTU) India

Malti Rani
Assistant professor C.S.E.
Pit Kapurthala (main campus PTU) India

***Abstract*— Virtualization technology, is the powerful solution to the resource management of cloud computing. Now a days, cloud computing become very popular and the technology are rapidly used. It also allows the users to migrating their data or information or calculation from one virtual machine to another. In data centers live migration plays an important role. Live migration has been widely used in energy reduction, dynamic resizing and load balancing, to increase accessibility and H/W maintenance, which cause transferring massive amount of unnecessary memory pages resulting into raise in the total migration time and downtime. Hence to overcome this issue two approaches will be used which is pre-copy approach and post copy approach. By using these two approaches during live migration we may reduce the total migration time and also downtime of the data.**

Keywords— **Cloud computing, Live Migration, Virtualization**

I. INTRODUCTION

"Cloud computing is a parallel & distributed computing system consisting of collection & interconnected & virtualized". Clouds are large pool of simply useable & accessible virtualized resource (such as hardware, development platform & services).

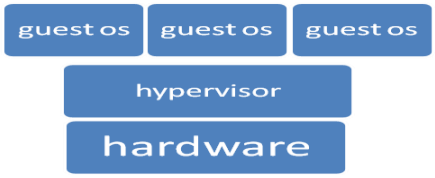

Fig.1. Virtual Machine

Virtualization is the component of cloud computing. It provides support for the creation of customizable, isolated and safe environment for the execution of applications. A virtual machine (VM) is an operating system that not only shows the behavior of a separate computer, although is also capable of performing jobs such as running applications and programs like a part of computer. A virtual machine, also known as a guest is created within another computing environment referred as a "host."

The rest of the paper is organized as follows. Background theory is explained in the section II. Related work is described in the section III. Proposed embedding are presented in section IV. Experimental results are shown in section V. Concluding remarks are given in section VI.

II. THEORY BACKGROUND

CLASSIFICATION OF VIRTUALIZATION

Virtualization covers a wide range of emulation technique that are applied to different areas of computing. A classification of these technique helps to better understand their characteristics and use.

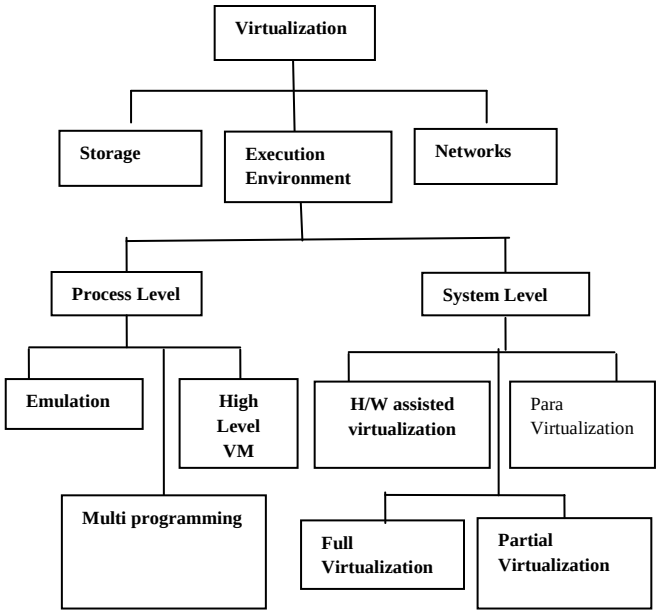

Fig.2. Classification of Virtualization

A. FULL VIRTUALIZATION: Full virtualization is a way in which a total installation of one machine is run on another. The result is a system in which all software supervision on the server is within a virtual machine. This sort of operation

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 6, ISSN No. 2455-2143, Pages 77-80
Published Online April - May 2016 in IJEAST (http://www.ijeast.com)

allows not only distinct applications to run, but also different operating system [6].

B. PARA VIRTUALIZATION: Para virtualization allows multiple operating systems to run only hardware device simultaneously by more efficiently using system resources, like as processors and memory. In full virtualization, the entire system is emulated (BIOS, drive, and so on), rather in Para virtualization, its management module operates with the operating system that has been adjusted to work in a virtual machine. Para virtualization typically runs better than the full virtualization model, simply because in a fully virtualized deployment, every element must be emulated [6].

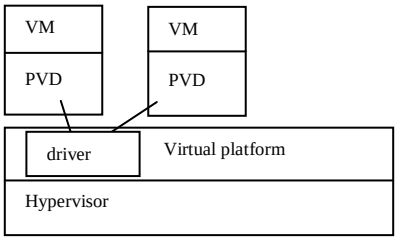

Fig.3. Para virtualization [7]

C. PARTIAL VIRTUALIZATION: Partial virtualization as the same indicates does not provide full emulation of the underlying hardware, it offers partial emulation. It does not allows the full execution of guest OS in fully isolation. Partial virtualization permits some application to execute transparently, not all the functions of OS can be supported. Address space virtualization is a good example of partial virtualization. It is utilized in time sharing system.

III. RELATED WORKS

Mehga r desai et.al. (2015) [1] - argued that migrate a data one virtual machine to another virtual machine. There exists many techniques that makes migration process very effective but still during migration, number of duplicate pages are transferred that increase total migration time and total downtime. Therefore there is need for approach which reduces transfer of duplicate pages. Total migration time and total downtime both are key parameter of live migration process. *Pradip D. patel et.al (2014) [3]* - presented a detailed survey of live migration of virtual machine in cloud computing. They also discussed different categories of migration techniques and performance metrics of live migration. They discussed a pre copy approach in which two phases are used warm up phase and stop and copy phase. By using these phase it reduced a total down time and migration timing between virtual machine. *Yi Zhong et.al. (2014) [5]* - described the optimization of memory state transfer in the iterative copy stage of the Pre- copy approach. The methods of optimizing memory pages, transfer for migrating virtual machine can be roughly into three categories: memory compression based methods, free memory pages elimination based methods and historical data of dirty page based methods. Approach uses historical dirty page of information periodically collected from the starting of migration at a permanent time interval to make the hot dirty pages recognition more accurate. j.*Arpurtharaj Johnson et.al. (2013) [2]* - discussed an improved time series based pre copy approach technique is proposed. With the time series prediction technique, frequently update dirty pages (high dirty pages) are identified more precisely, and transmit them in the end of iteration, in order to reduce unwanted dirty pages. Doing so it could reduce the total migration time. *Soramichi akiyama et.al. (2012) [4]* - proposed a technique called "memory reusing" that reduces the amount of transferred memory of live migration. When a VM migrates to destination host, the memory image of the VM is kept in the source host. After that VM migrates back to the original host later, the kept memory image will be reused i.e. memory pages are identical to the kept pages will not be transferred.

IV. PROPOSED METHODOLOGY

In this work, considering the drawbacks of the existing methodologies, post-copy approach has been implemented. Taking into consideration the memory and energy efficiency, steps have been taken into that direction. When we reduce the memory and energy consumption in the VMs migration process, first item that has been focused upon is the type of physical host because the VMs not performing heavy and critical computations are consolidated into a small number of shared servers as VMs that are active are distributed to dedicated servers to influence the performance. The main effort is to reduce the data and memory transfer in the migration.

In data migrations, the all data used to be transferred from source host A to the destination host B. The host B used to process that data and then send the complete data again back to the host A. So it was too memory and time consuming. It was enhanced by reducing the data to be sent back to the host A by using the memory reusing approach. Here, the only updated data was sent back to the host A rather than sending the complete data while only few pages have been updated. The data that have not been updated have already their memory image present on the source host. for this reason, it leads to better performance. Still there is scope to optimize it.

In the proposed method, the memory and time consuming has been reduced by taking a further step towards it. The memory starvation has been deal in the following way:

- A page that is updated on another host is no longer needed because the page cannot be reused.
- A memory page that has strong possibility of update in a near future does not need to be cached.

Pre-copy memory migration

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 6, ISSN No. 2455-2143, Pages 77-80
Published Online April - May 2016 in IJEAST (http://www.ijeast.com)

Warm-up phase

In pre-copy memory migration, the Hypervisor typically copies all the pages from Host A to Host B while the VM is still running on the source Host A. A few of the pages re-copied through this process until the re-copied pages is not less than dirty page.

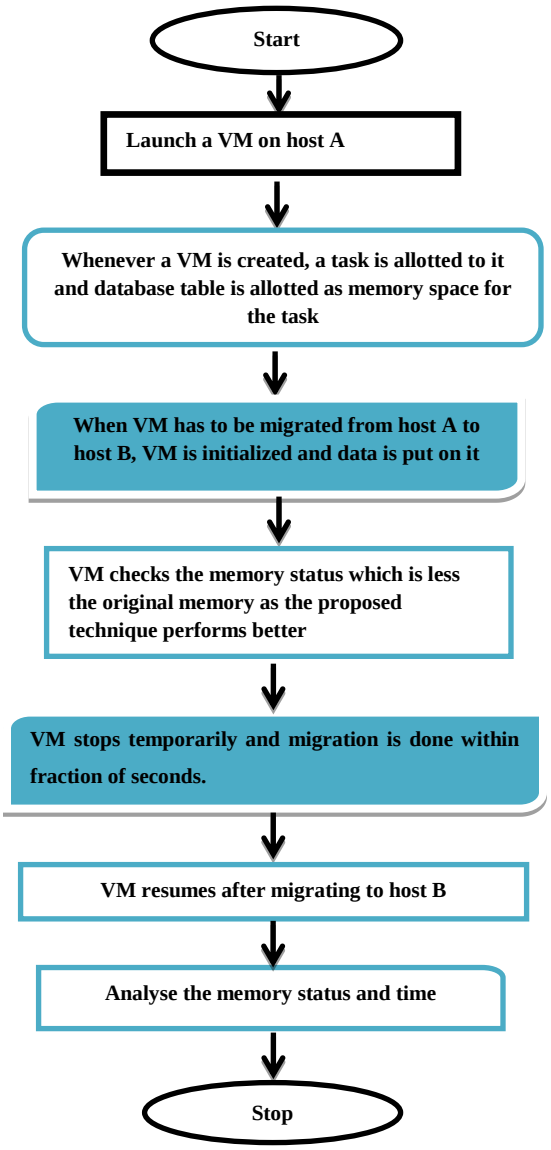

Fig.4. Proposed Methodology

Stop-and-copy phase
Later than the warm-up phase, the VM will be stopped on the original host, the remaining dirty pages will be copied to the target, and the VM will be resumed on the target host. The time between stopping the VM on the original host and resuming it on destination is called "down-time", and ranges from a milliseconds to seconds according to the range of memory and applications running on the VM. There are several techniques to reduce migration down-time, such as using probability density function of memory change

Post-copy memory migration
When VM migration is starting in post copy approach, it suspends the VM at the source. When the VM suspended, a small subset of the execution state of the VM (CPU registers and non-page able memory) is transferred to the destination. At the destination VM is resumed while most of the memory state of the VM still resides at the source. When the VM tries to access pages at the destination, which is not still transferred, it generates page-faults. These page faults are trapped at the destination and redirected towards the source over the network. Such faults are known as network faults. The source host responds to the network-fault by sending the faulted page. Since each page faults of the running VM is redirected towards the source, this method can degrade performance of applications running inside the VM. still, pure demand-paging accompanied with this method such as pre-paging can reduce this impact by a great extent.

V. EXPERIMENTAL SET UP

Table - 1 Experimental result

No of VM	Memory Buffer(Host A)	memory migrating	Changes in the pages back (host A)	Time Consumed
5	587 MB	587 MB	121 MB	33922 ms
10	1143 MB	1143 MB	226 MB	34398 ms
15	1874 MB	1874 MB	Unable to migrate 15 VM from host A to B due to memory leakage	
15	1874 MB	1874 MB	It received 13 VM 1135 MB	35320 ms

In this section memory migrate from host A to host B where memory leakage problem has occurring and it unable to migrate 15 VM from host A to Host B due to memory leakage. Table 1 shown this memory leakage problem.

VI. CONCLUSION

Total migration time and total downtime both are key parameters of live migration process. Our proposed approach is to remove the memory leakage problem in pre-copy approach we will further improve the memory leakage problem by using post copy approach and gives efficient migration In future, the proposed work will be implemented in cloud sim for different scenario.

VII. REFERENCES

[1] Megha r deasi hiren b.patel *"Efficent Virtual Machine Migration In Cloud Computing"* Communication Systems and Network Technologies (CSNT), 2015 Fifth International Conference on, IEEE, pp. 1015-1019.

[2] J.Arputharaj Johnson*"Optimization of Migration Downtime of Virtual Machines in Cloud",* Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on, IEEE, pp.1-4.

[3] Pradip D. patel , Miren karamta , M.D. Bhavsar and M.B.Potdar *"live virtual machine migration technique in cloud computing: A survey"* intenational journal of computer application (0975-8887) colume 86- No 16, January (2014), Foundation of computer application.

[4] Soramichi Akiyama, Takahiro Hirofuchi , Ryousei Takano and Shinichi Honiden *"A memory reusing mechanism for dynamic VM consolidation "* Cloud Computing (CLOUD), 5th International Conference on, IEEE, 2012, pp. 606-613.

[5] Yi Zhong, Jian Xu, Qianmu Li, Hong Zhang, Fengyu Liu *"memory state transfer optimization for pre-copy based live VM migration"* Advanced Research and Technology in Industry Applications (WARTIA), IEEE Workshop on, IEEE, 2014 pp. 290-293.

[6] Cloud Computing: A Practical Approach https://books.google.co.in/books?id=6wJ3XNHzGoIC&pg=PA10&lpg=PA10&dq=%22t#v=onepage&q=%22t&f=false

[7] Patidar, Deepika and Patheja, PS and Waoo, Akhilesh A " *An Efficient Approach for Cloud Computing based on Hierarchical Secure Paravirtualization System Resource Model"* International Journal of Applied Engineering Research, vol 7, no. 11(2012).