Institute of Advanced Engineering and Science

International Journal of Cloud Computing and Services Science (IJ-CLOSER)
Vol.3, No.3, June 2014, pp. 183~189
ISSN: 2089-3337

A Complete Theoretical Review on Virtual Machine Migration in Cloud Environment

Rajeev Kumar Gupta, R.K. Pateriya
Department of Computer Science & Engg , MANIT, Bhopal

Article Info

Article history:

Received Feb 20th, 2014
Revised Mar 25th, 2014
Accepted Apr 30th, 2014

Keyword:

Virtual Machine
Migration
VM Placement
Down Time,
Total Migration Time
Cloud Computing

ABSTRACT

Live migration of virtual machine is a key feature of virtualization. It allows the administrator to move the virtual machine from one physical machine to another physical machine. This technique is widely used for the fault tolerance, load balancing, server maintenance and resource consolidation. Down time and the total migration time are two critical issues in the virtual machine migration, both should be minimized as possible. So the application running on the virtual machine will be suspended for negligible time. Virtual machine migration problem consist of four distinct steps. First step is to select the host from where VM has to be migrated. After selecting the host next step is to select the VM which is migrated. Third step is to select the host where the migrated VM will be placed and the last step is to decide the method which is used to transfer the VM data. In this paper some exiting methodologies and their anomalies are explained for each of the above steps.

Corresponding Author:

First Author,
Department of Computer Science & Engg , MANIT, Bhopal
Email: rajeev_12276@yahoo.co.in

1. INTRODUCTION

Cloud computing is a rapidly growing area in the field of academic as well as industry. Cloud computing is the dynamic provisioning of IT capabilities (Network, Hardware, Software or services) as a service through the internet to the user on demand. According to NIST definition, cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources, these computing resources include networks, servers, storage, applications, and services. In cloud computing, the shared pool of computing resources can be rapidly provisioned and released [1]. Mainly Cloud computing is a business oriented model, which provide on demand computing resources. It has become famous in short time span because of its attractive features like easy to use, pay as use and accessibility of their services throughout the globe etc. It allows the user to access the data anywhere in the world via internet. Cloud computing is not a completely new concept it is a combination of grid computing, parallel computing, cluster computing, utility computing and distributed computing [2]. Cloud can be deployed in three ways i.e. Private, Public and Hybrid cloud [3,4]. Cloud services are broadly classified into three categories Software as a service (SaaS), Platform as a service (PaaS), and Infrastructure as a service (IaaS). SaaS provide the software as a service which allow users to access the software without installing them on the users machine such as Google doc. PaaS provide the complete platform to the developer for developing and deploying their application. Amazon EC2 and Microsoft Azure provide PaaS. IaaS is the delivery of computer infrastructure (Server, data center, network, virtualized environment) as a service to the client. These services are available to the client in a pay-as-you –go model. According to the Berkeley report [5], “Cloud computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even more attractive as a service’’. Cloud enables the user to focus on innovation and creating business value for their services instead of basic hardware and software infrastructures.

Virtualization is the key technology behind the cloud computing, which allow multiple OS to run simultaneously on a single physical machine. All computing resources are provided to the client through virtualization. It increases the resource utilization because same hardware can be used by multiple users. Virtualization is implemented through the hypervisor which is also called virtual machine manager (VMM). Hypervisor is a software layer which manages all virtual machine and separates the virtual hardware from the actual hardware. Each user has its own virtual machine, which is created on the basis of user requirement. Each virtual machine has its own resources and number of virtual machine can be run on a single physical machine. One of the important features of the virtualization is the live migration. Live migration is a process of transferring the virtual machine from one physical machine to another physical machine. Since load on the virtual machine can be changed dynamically, so there is a possibility when the current physical machine is unable to fulfill the resource requirement of the virtual machine. This problem can be avoided either by adding the extra resources to the physical machine or by migrating the virtual machine. Live migration of the virtual machine is useful in the case of server failure, server maintenance, load balancing, hot spot mitigation and server consolidation. Some of the well known approaches used to find the resource requirement of the virtual machine, select the target physical machine and transfer the virtual machine are explained in detail in the next section.

2. VIRUAL MACHINE MIGRATION

Transferring the entire virtual machine from one physical machine to another physical machine is called virtual machine migration. Downtime, total migration time, energy consumption and resource balancing are the important metrics which define the effectiveness of the migration process. Downtime is defined as the time for which service on the virtual machine are not available to the user. Total migration time is time duration between the migrations initiated at the source to the time when the migrated virtual machine gets a consistent state with the original state at the destination. Downtime and total migration time both should be minimized as possible.

3. LITERATURE SURVEY

VM migration is the process, where VM are transferred from one physical host to another physical host. When, which and where are the three question that must be answered before starting the VM migration. Following four steps are involved in the complete virtual machine migration process.

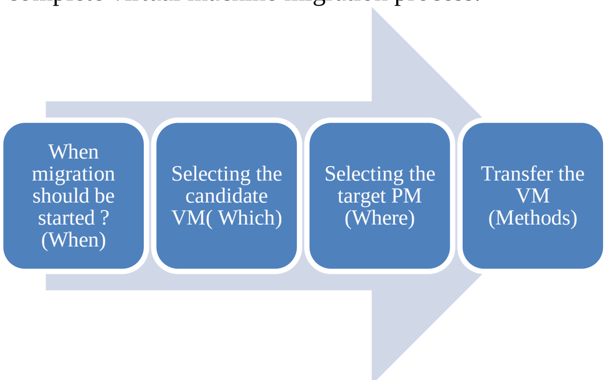

Figure 1. Steps involved in VM migration

3.1. When the migration process should be started?

Generally lower and upper threshold are used to decide when the virtual machine migration process should be started. A system is called overloaded, when the load on the system crosses the upper threshold. Similarly a system is called underloaded, when the load on the system is below to the lower threshold. Virtual machines are mostly migrated in three different situations.

i. To balance the load on the physical machine. Some VM are migrated from overloaded PM to underloaded PM.

ii. In the case of server consolidation, all the VM inside the underloaded PM are migrated to the other host, to increase the resource utilization and reduce the energy consumption.

iii. To mitigate the hot spot. Hot spot situation occurs when the physical machine does not have sufficient resources to fullfill the VM requirements.

L.Xu et al. [8], used 0.8 as a upper threshold and 0.2 as a lower threshold. To avoid the unnecessary migration due to the temporary peak load, they use the single exponential smoothing algorithm (SES). SES is a prediction technique, mostly used in the forecasting. Following equation were used to predict the future load on the system

$$P_{t+1} = P_t + \alpha\,(A_t - P_t)$$
$$P_{t+1} = \alpha\, A_t + (1 - \alpha)\, P_t$$

Where P_{t+1} is the predicted load on the system at t+1 time, P_t is the predicted load on the system at t time and α is the smoothing factor ($0< \alpha < 1$). Above equation is a recursive equation so

$$P_{t+1} = \sum_{i=0}^{t-1} \alpha\,(1-\alpha)^i A_t + (1-\alpha)^t P_t\,,\ t > 0$$

They considered only three resources CPU, memory and bandwidth, so resources can be represented as a cube, but what happened if more than three resources are considered. It was also assumd by the authors that the load smoothly changes with time which is practically impossible.

3.2. Selecting the candidate VM

Which VM is selected for the migration depends on the situations i.e. load balancing, server consolidation and hot spot mitigation. In the case of server consolidation all the VM available on the host have to be selected for the migration. VM selection in the case of hot spot and load balancing is a complicated task, because if light weight VM is chosen than, number of VM has to be migrated and if we choose weighted VM then it will increase the total migration time. Numbers of approaches have been used for selecting the VM.

T. Wood et al. [9] used a volume to size ratio (VSR) to select the VM. When the hot spot is detected one or more VM is moved from this PM to another PM. VM which is having highest VSR is selected as a candidate VM, where volume is corresponding to the load of the VM and size is the memory footprint of the VM.

$$\text{VSR} = \frac{sand\ volume}{Size\ of\ the\ VM}$$

Khanna et al. [10] proposed a method for the hot spot mitigation, they select the VM with least CPU utilization. Problem with this approach is that numbers of VM have to be migrated.

A. Beloglazov et al. [11], proposed a method for load balancing, they select the VM whose size is greater than or equal to the (host utilization-upper threshold). If K is the selected VM, then
Size of K >= host utilization −− upper threshold
If no such VM found then the number of VM will be migrated.

3.3. Method for choosing the target physical machine

Placing a VM on the appropriate host play a vital role in VM migration because resource utilization can be increased and power consumed by the data centers can be reduced by the efficient VM placement algorithm. After selecting the VM which has to be migrated, next step is to calculate the resource requirement of the virtual machine and then find the appropriate physical machine that can fulfill the individual resource requirement of the candidate virtual machine. Numbers of method are available to calculate the load on the system.
Y. Fang et al. [15], gives the equal weight to all the resources. They simply calculate the load on host, by adding the load of the VM available in the PM. Load on the i^{th} host are calculated by the following equation

$$HL_i = \sum_{j=1}^{n} VL_j$$

Where n is number of VM in i^{th} host.

M.Mishra et al [12], gives the different weight to the each resources by using the weighting coefficient (W_i). Following equation is used to calculate the load on k^{th} host

$$HL_k = \sum_{i=1}^{n}(w_i) * \frac{\sum_{j=1}^{m}(RU_{ij})}{(Hcap_{ik})}$$

Where n is the type of resources i.e. cpu, memory, io etc, m is the number of VM in k^{th} host, RU_{ij} is the i^{th} type resource used by the j^{th} VM and $Hcap_{ik}$ is the i^{th} resource capacity of k^{th} host.
C. Clark et al. [7], proposed a method named as Sandpiper to decide where the VM has to be placed. It is a Xen based automated system which is used for detecting and mitigating the hot spot. Two algorithms are used by the sandpiper, One is a hot spot detection algorithm, which is used to detect the hot spot (when the virtual machine migrated) and another is the hot spot mitigation algorithm, which determine the virtual machine to be migrated and where to be migrated. Decision on the virtual machine migration are taken on the metric sand-volume, which is define as

$$\text{Sand-volume} = \frac{1}{1-cpu} * \frac{1}{1-net} * \frac{1}{1-mem}$$

Where cpu, net and mem are the normalized utilization of the cpu, network and memory respectively. If the multiple resource are heavily utilized it will increase the sand volume. It means system is overloaded. Total utilization volume of the physical machine is define as

Total utilization volume= cpu*net*mem

So the remaining volume of the physical machine, which is also known as exploitable volume is define as

Exploitable volume= (1-cpu)* (1-net)*(1-mem)

It means metrics used by the sandpiper is the inverse of the exploitable volume of the physical machine. To select the target virtual machine, physical server is arranged in decreasing order of their volume. After that virtual machine in each server are also arranged in decreasing order of their VSR. Virtual machine migration algorithm selects the virtual machine with highest VSR from the server (PM), having highest sand volume and then determine whether it can be hosted on the target physical machine which having least sand volume. This migration is possible only when the target physical machine is capable to fulfill each of the individual resource requirement of the virtual machine. If the sufficient resources are not available on the target physical machine then the migration algorithm select the next least loaded physical machine. This process continues till the correct match is not found for the candidate virtual machine. If no physical machine is able to satisfy the resource requirement of the highest VSR virtual machine then the VM with next highest VSR is considered.
This method seems good, but there is a possibility to select wrong target physical machine, because they are not using the shape of the resource utilization. 3D (CPU, network, memory) resource information are converted into 1D that is sand-volume. Since they used only sand-volume to select the target PM, so if two physical machine having the same Exploitable volume then they both are equally suitable for a target physical machine, but on the basis of shape of exploitable volume one may be better than the other. So sand-volume and the shape of the exploitable volume both metric should be used to select the target physical machine.

M.Mishra et al [12], proposed a vector method for placing a VM. In the VectorDot[12] method they are using dot product between the source utilization of physical machine and resource requirement of virtual machine to choose the target physical machine. In the first step they calculate the dot product between the normalized resource utilization vector $\overrightarrow{RUV}$ of physical machine and normalized resource requirement vector $\overrightarrow{RRV}$ of virtual machine and then choose the physical machine which having lowest dot product. They also use the angle between $\overrightarrow{RUV}$ of PM and $\overrightarrow{RRV}$ of VM to decide the target physical machine. Physical machine that's $\overrightarrow{RUV}$ makes the largest angle with $\overrightarrow{RRV}$ of VM is selected as a target physical machine.
For the proper utilization of the resources it is necessary that the virtual machine which required more CPU and less memory should be placed on the physical machine which has low CPU and more memory utilization. This method seems good, but there is a possibility to select wrong target physical machine. Following example illustrate wrong target physical machine selection.
Suppose

$\overrightarrow{RUV}$(PM1) = .4$\hat{\imath}$+.5$\hat{\jmath}$

$\overrightarrow{RUV}$(PM2) = .4$\hat{\imath}$+.2$\hat{\jmath}$

$\overrightarrow{RRV}$(VM)= .2$\hat{\imath}$+.1$\hat{\jmath}$

Where $\hat{\imath}$ and $\hat{\jmath}$ denote the unit vector along CPU and memory axis respectively. Dot product between $\overrightarrow{RUV}$(PM1) and $\overrightarrow{RRV}$(VM) is .13, and dot product between $\overrightarrow{RUV}$(PM2)) and $\overrightarrow{RRV}$(VM) is .10. So PM2 is selected as a target virtual machine. But PM1 is the better choice for the virtual machine placement, because virtual machine required more CPU then memory.

This problem can be eliminated by taking the dot product between the $\overrightarrow{RRV}$(VM) and unit vector along $\widehat{RUV}$(PM). The unit vector of $\widehat{RUV}$(PM1) and $\widehat{RUV}$(PM2) given as

$$\widehat{RUV}(\text{PM1}) = \frac{.4\hat{\imath}+.5\hat{\jmath}}{\sqrt{.16+.25}} = \frac{.4\hat{\imath}+.5\hat{\jmath}}{.64}$$

$$=.62\hat{\imath} +.78\hat{\jmath}$$

$$\widehat{RUV}(\text{PM2}) = \frac{4\hat{\imath}+.2\hat{\jmath}}{\sqrt{.16+.04}} = \frac{4\hat{\imath}+.2\hat{\jmath}}{.45}$$

$$=.89\hat{\imath} + .44\hat{\jmath}$$

Dot product given as

$$\widehat{RUV}(\text{PM1})*\overrightarrow{RRV}(\text{VM}) = .62*.2 + .78*.1 = .202$$

$$\widehat{RUV}(\text{PM2})*\overrightarrow{RRV}(\text{VM}) = .89*.2 + .44*.1 = .222$$

So PM1 chosen as a target physical machine because its having a lowest dot product.

3.4. Method for transferring the Virtual machine

After choosing the target physical machine, next step is to transfer the VM from source PM to the selected target PM. In this step entire VM (memory and CPU state) has to be transfer from one physical machine to another machine. Three methods are available to transfer the VM [6].

i. Stop and copy- In this approach stop the virtual machine on the source PM, copy all memory pages and CPU states to the destination PM, and then resume the VM on the destination PM.

This approach gives the best total migration time than the other migration algorithms, but increases the downtime.

ii. Post-copy migration- In this approach stop the virtual machine on the source, instead of transferring all memory pages and CPU states as in the Stop and copy approach, transfer the minimum memory pages and CPU states that required initializing the virtual machine on the destination PM. Remaining memory pages and CPU states copied through the demand paging. It comes under the live migration.

This approach reduces down time but increase total migration time

iii. Pre-copy migration- It's an iterative method of the live migration. In the first round all memory pages and CPU states are transferred to the destination physical machine. Since virtual machine still run on the source virtual machine, so during the transmission some memory pages are modified. These modified pages are called dirty page. Set of dirty pages is called writable working set (WWS). Set of dirty pages in the previous round are transferred into the next round. This process is repeated until the size of WWS is reduced to the predefine maximum allowable window size say 1MB or predefine maximum number of iteration, then stop the virtual machine on the source PM and transfer all memory pages and CPU states to the destination physical machine, then start the virtual machine on the destination PM.

Pre-copy approach is more reliable then the post copy because in the pre copy approach source node keep the updated copy of all memory pages and CPU states, so in the case of destination failure virtual machine can be recovered. This approach is useful only if the dirty page transfer rate is larger than the dirty page growth rate. So many methods have been proposed for the VM migration. Main aim in all exiting method is to reduce the down time and total migration time.

H. Jin et al [13], used compression based pre copy approach. In this approach, they are using pre copy method for transfer of data from one physical machine to another physical machine. At the source, data is compressed and compressed data is send to the destination in each iteration. Data is decompressed at the destination.

Data is compressed on the basis of memory page characteristics. To count the similar word (Fully and partially) and zero byte they are using 16 word dictionary, which keep the 16 recently seen word. On the basis of memory data characteristics pages are classified into three type

I. Page having large number of zero byte then the non zero byte
II. Page with high similarities
III. Page with low similarities

To count the number of zero byte whole page is scanned and then sparse matrix is used to keep the information about the offset and the value of non-zero byte. In the second case where page have high similarities, an algorithm with low compression ratio is required. In the last case when page have low similarities, an algorithm with high compression ratio is required.

Memory compression does not change the speed, it only reduces the transfer time by compressing the data. There are two main problems with this approach

I. Compression and decompression overhead are introduced by this algorithm
II. In the pre copy approach a single page is sent repeatatively. That means repeatative compression/decompression, which is wastage of memory and bandwidth.

H. Jin et al [14] , proposed an approach based on trace and replay. Method discussed above transfer the dirty page to migrate the virtual machine while this method [14] transfer the log file instead of dirty pages. These log files record the non deterministic system event which is used by the target virtual machine to reach the desire CPU state. Non deterministic event may be a time or external input. Target virtual machine replayed with the received log file to recover the state.

In this method system state of the source virtual machine at the current instant are saved into the image, which is called the checkpoint. In the first round checkpoint is transferred from the source physical machine to the target physical machine, during this transmission virtual machine continually run on the source host. In the subsequent round log files are transferred, generated dring the previous round. Finally when the size of the log file is less than the threshold, stop the source virtual machine and transfer the last log file to the target physical machine. Target physical machine replay with the received log file and takeover the source virtual machine services

This approach reduces downtime, total migration time and network bandwidth consumption, because the size of the log file is much less than the size of page. This algorithm seems good but it is a useless if

I. Log replay rate is slower than the log growth rate
II. Log growth rate is higher than the log transfer rate

Because of the resource availability, all events can be immediately replayed. So log replay rate is mostly faster than the log growth rate. Generally log transfer rate is higher than the log growth rate but still there is a possibility when the log growth rate is higher than the log transfer rate especially when the high OS intensive workload is running on the virtual machine.

4. CONCLUSION

Virtual machine migration is the process to move the VM from one host to another host. Live migration is the migration without suspension of the VM. Virtual machine migration is required in the case of load balancing; sever consolidation and hot spot mitigation. The main aim of Virtual machine migration is to increase Resource utilization and to reduce power consumption. Some of the migration approaches found in existing literature are discussed here with their drawbacks and anomalies. In this study it is obsereved that placement of VM is one of the critical task to perform because it affects energy consumption as well as resource utilization. Energy consumption can be reduced by minimizing the number of active servers which can be achieved only by adapting proper VM placement approach.

REFERENCES

[1] Peter Mell,Timothy Grance, "The NIST Defination of Cloud Computing" by National Institute of Standards and Technology - Computer Security Resource Center, 2011.
[2] M. Armbrust et al., “A view of cloud computing”, Communications of the ACM, pp. 50–58, 2009
[3] S. Wind “Open source cloud computing management platform”, IEEE conference on open system pp. 175-179,2011
[4] V. Sarathy, P. Narayan and R. Mikkilineni, “ Next generation cloud computing architecture”, 19th IEEE international workshop on enabling technology: infrastructure for collaborative enterprise ,pp.48-53,2012.
[5] M. Armbrust et. al. Above the Clouds: A Berkeley View of Cloud Computing. Technical Report No. UCB/EECS-2009-28, University of California at Berkley, USA, Feb. 10, 2009.
[6] D.Perez Botero, “A brief tutorial on live migration of virtual machine from a security perspective”, 2011
[7] C. Clark, K. Fraser, s. Hand and J.C. Warfield, “Live migration of virtual machine”, proceeding of the 2nd conference on symposium on network system design and implementation, 2007.
[8] L. Xu at el.,”Smart DRS: A Strategy of Dynamic Resource Scheduling in Cloud Data Center”, proceeding of the IEEE International Conference on cluster Computing Workshop, pp. 120-127, 202
[9] T. Wood et al., “Black-Box and Gray-Box strategies for virtual machine migration”, proc. 4th conference on symposium on network system design and implementation, 2007.
[10] Gunjan Khanna et al., “Application performance management in virtualized server environments”, Network Operations and Management Symposium NOMS 10th IEEE/IFIP, pp. 373-381, 2006.
[11] A. Beloglazov et al. “Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing”, Elsevier journal of Future Generation Computer Systems, pp. 755–768, 2012.
[12] M.Mishra et al.,” On theory of virtual machine placement: Anomalies in existing methodologies and their migration using a novel vector based approach”, proceeding of the IEEE international conference on cloud computing, pp. 275-282, 2011.
[13] H. Jin et al., Live virtual machine migration with adaptive memory compression” proceeding of the IEEE international conference on cluster computing, pp. 1-10, 2012.
[14] H. Jin et al., “Live migration of virtual machine based on full system trace and replay”, proceeding of the 18th ACM 2009.
[15] Y. Fane et al., “A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing”,in the proc. Of the international conference on Web information systems and mining (Springer and ACM), pp. 271-271, 2010.

BIOGRAPHY OF AUTHORS

Rajeev Kumar Gupta is a PhD candidate in the Department of Computer Science and Engineering at Maulana Azad National Institute of Technology, Bhopal, India. He has completed his Bachelor degrees in CSE at SATI, Vidisha and Master degree inCSE at MANIT, Bhopal. His research area is Cloud Computing, Distributed computing and Gried computing.

Dr. R. K. Pateriya is an Associate Professor in the Department of Computer Science and Engineering, at Maulana Azad National Institute of Technology, Bhopal, India. He is a member of the IEEE. His current research interests include Cloud and Distributed Computing, E-Commerce, Security etc. He has authored and published over 100 research Papers.