

LIVE VIRTUAL MACHINE MIGRATION TECHNIQUES – A SURVEY

Rakhi k Raj

PG Scholar

Department of Information Technology
Karunya University

Getzi Jeba Leelipushpam.P

Assistant Professor

Department of Information Technology
Karunya University

Abstract

Live virtual machine migration is the process of moving virtual machine from one physical host to other without disturbing the users. It is used in the case of proactive failure, load balancing, resource scheduling, and server consolidation. Effective migration of virtual machine required the movement of storage, memory, process states and network connectivity. During the process of migration, services are running in the virtual machines, hence it requires minimum total migration time, down time in order to avoid service degradation. In this paper we describe about various migration control techniques for effective live virtual machine migration to reduce both downtime and total migration time.

1. INTRODUCTION

Cloud computing is an emerging new paradigm for hosting services over the internet. Cloud computing offers infrastructure as a service, platform as a service and storage as a services to the cloud users. Cloud users are charged based on their service usage. The cloud computing services are available at anywhere, anytime, only we have to need internet connectivity. To improve the utilization of cloud resources we use virtual machines. Virtual machine is a software implementation of a computing environment in which operating system or program can be installed and run [1]. The virtual machine provides an isolated environment to an application, execution of an application running on one virtual machine is separated from the others.

The figure [1] is an indication of virtual machine diagram in which n number of virtual machines is working on the one physical host. Virtual machine monitor is a software implementation of a program it translates the request from the virtual machine in to physical host and allocates the CPU, memory, and hardware.

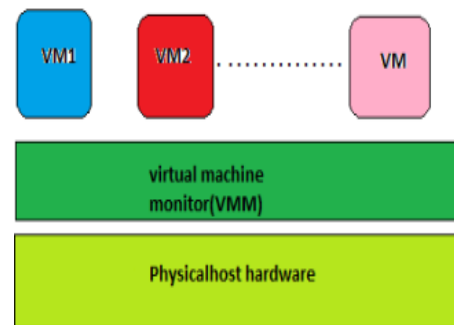


Fig 1: Virtualised machine diagram

The characteristics of virtual machine include 1) isolation 2) ease of testing 3) mobility. Isolation allow users to work with the applications independently even though they share a single computer's physical resources. Mobility means we can move virtual machine from one physical host to other in the case of failure, managing of energy power consumption etc. Ease of testing it provides the snapshot and rollback facility to users. Using this facility they can easily customise their application.

Live virtual machine migration

Live virtual machine migration is the process of moving virtual machine from one physical host to other host without disturbing others [2]. Live virtual machine migration is used in the following area of cloud computing 1) power management 2) server consolidation 3) proactive maintenance. Server consolidation is the process of aggregating number of VM in to a single physical server. The server consolidation is helpful in power consumption and energy of the data centers. Proactive maintenance deals live virtual machine can replace the failure virtual machine by moving in to new physical machine. Disaster recovery is other important feature of live migration. It also used for load balancing, in which work is shared among computers in order to

optimize the utilization of available CPU resources [3]. It is also used in the case of resource management.

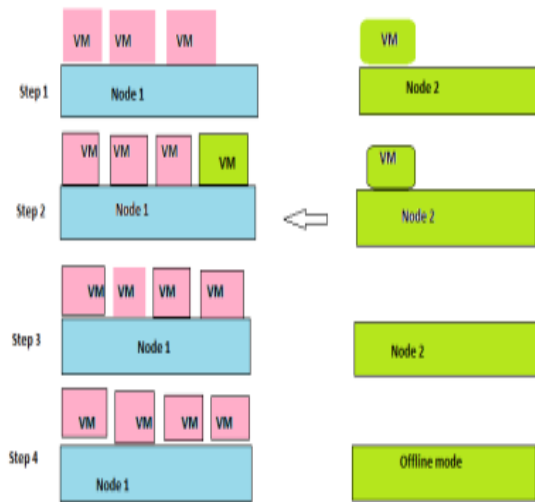


Fig 2: Live VM migration

The above diagram shows the live virtual machine migration. Initially three VM working on node 1 and one VM working on node2. Migrations allow the movement of VM from node 1 to node 2. Live virtual machine migration having mainly two performance metrics.

- 1) Total migration time: It is defined as the total time taken to migrate a virtual machine from its host machine to the target machine.
- 2) Down time: Down time is defined as the duration of time at which services are not available to the users.

The key challenges in the live virtual machine migrations are minimal service downtime and total migration time [4]. Migration techniques are trying to reduce both migration time and down time.

2. MIGRATION TECHNIQUES

All migration techniques are trying to reduce total migration time and down time. In the following section explained various migration techniques, which are used for reducing total migration time and down time. The broader type of VM Migration techniques are 1) Pre-copy 2) Post-copy. In pre-copy migration first transfer the memory contents to the target machine. After completing the memory transfer processor states are transferred to destination. In post copy memory data are transferred after the processor states transfer.

2.1 PRECOPY

In pre-copy approach bulk memory state is migrated to a target node even as the VM continues to execute at a source node [5], mean time they record the changes in memory data known as dirtied pages if a transmitted page is dirtied, it is re-sent to the target in the next round. This copying of dirtied pages continues until either a small number of pages or certain number of prefixed rounds is reached, whichever comes first. This indicates the end of the memory transfer phase and the beginning of service downtime. The virtual machine is then suspended at the source machine and its processor state plus any remaining dirty pages are sent to a target node. Finally, the virtual machine is restart and the copy at source is destroyed.

The migration performance can be calculated by using total migration time and total down time. The factors affecting the total migration are link bandwidth, migration overhead and page dirtied rate [10]. In pre copy approaches the source and target host are working on the same storage area network hence no need to transfer the storage. The network connectivity is changed by using ARP reply telling that IP address of the machine is moved in to new location and traffic flows are redirected to new location. The diagram [3] shows the steps in the pre copy approach

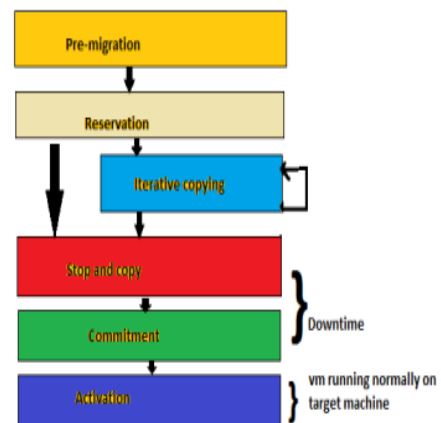


fig 3: Steps for precopy approach

Advantages of this approach are simple and it having very small reduced downtime .It also provides the fault management. One main disadvantage is it has more total migration time. It provide very difficult with quick migration over head.

2.2 POSTCOPY

Post copy migration includes the transfer of memory content after the process state is transferred to destination [8]. Concurrently memory pages that are faulted in to the memory not in the target machine and not yet pushed are demand-paged over the network from source machine. Post copy approach, memory pages are transferred at most once. For handling memory pages during migration it using dynamic self ballooning mechanism. For obtaining the memory pages from host it having four variants 1) demand-paging 2) active push 3) pre-paging 4) dynamic self ballooning (DSB). In pre-copy repeatedly dirtied pages could be resent multiple times where Demand paging ensures that each page is sent over the network only once. Similarly, active push phase are removed the residual dependencies from the source host as quickly as possible, compared to the non-deterministic copying iterations in pre-copy.

Advantage is that this one easy to implement and the pages transferred are once to avoid the duplication and using dynamic self ballooning to handle the free pages. One main disadvantages of this approach is downtime is more, which means that the service is not available for the users. If any kind of crashing occurs during the live VM migration it cannot be reconstructed.

3. MIGRATION CONTROL TECHNIQUE

The above approach is trying to reduce the total migration time and downtime but the effect is that total number of pages transfer is more. In order to reduce the total number of pages transfer during the migration, here explained four different techniques 1) Adaptive memory compression, 2) LRU and splay tree algorithm, 3) live virtual machine migration using CPU scheduling 4) check point recovery trace and replay.

3.1 ADAPTIVE MEMORY COMPRESSION

In pre copy approach it is difficult to provide fast migration in the case of low network overhead, its due to large number of pages are iteratively transfer from one host to target[6]. In order to avoid this condition we go for adaptive memory compression. This approach is used to optimize live VM migration based on the pre-copy approach. Here we are using the compression to provide the fast migration. In compression technique we are using the zero-aware characteristics-based compression algorithm for live VM migration.

Before transmission of data it is compressed and transfer to the target host. In target host compressed data is again decompressed. For doing the compression one step is necessary that is memory data characteristic analysis. In memory data analysis we first keep a dictionary of 16 words which is recently used. After that it finding the word

similarity of those pages. Using word similarity they find they decide the compression algorithm. The memory pages are classified with pages having zero bytes, pages having strong regularity and pages having weak regularity. The compression algorithm is helpful for fast moving of memory pages. This process is continued until all memory data transferred to destination.

The advantages of this approach is it have reduced number of pages transfer and total number of rounds, but one disadvantage is that it have more total migration time, additional one mechanism is needed in the pre processing phase. Here we use multithreaded technique to use parallel task.

3.2 LRU AND SPY TREE ALGORITHM

The above described approach number of pages transferred is more in order to avoid the number of pages and iterative rounds we go for LRU and Splay tree algorithm[7]. In LRU and Splay tree algorithm it consists of stacks and counters, top of stack contain the last recently used pages. Based on the algorithm they are finding the working set.

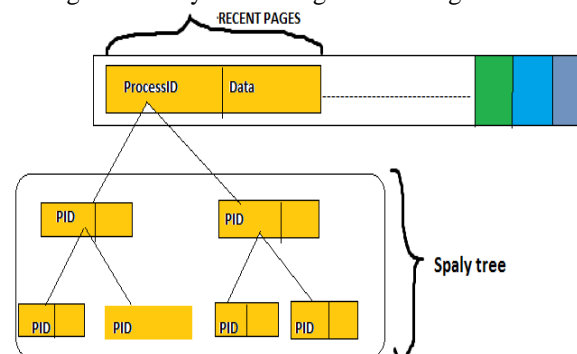


Fig:4 Working set prediction

If a new process is fall with process ID first check whether it in the LRU cache or not, if it is not in the LRU cache they replaced in the LRU cache and constructing the splay tree for that. If it is in the LRU cache then replace in the top of the cache. if LRU cache is full then they replace the last one in the cache. In the next step they transfer the memory pages to the destination and re execute in the destination host. It consist of mainly 3 steps 1)pre processing, 2)push phase, 3)stop and copy phase.

During pre-processing phase it will calculates the working set prediction algorithm and calculate recently used memory pages. Then they are transferring the page other than last recently used in first step in the iteration in push phase. Then they are iteratively transferring the memory pages that are dirtied. Stop and copy phase stop source machine and resume at the target.

Advantages are that it has reduced migration time because less number of pages is transferred during the migration. But the performance is depends

on the number of modified pages if the page modified more than the migration time is less.

3.3 CHECK POINT RECOVERY/TRACE AND REPLAY

The all approach described above are neither or nor having both total migration time and down time [9]. Here it is described an approach which having both reduced time and total migration time which utilising the effectiveness of small size of logs.

Check point is the process of store the state of a system .Replay is used for re-execution of the past system. It will helpful for the debugging, proactive failure. CR/TR motion is having both reduced down time and total migration time.

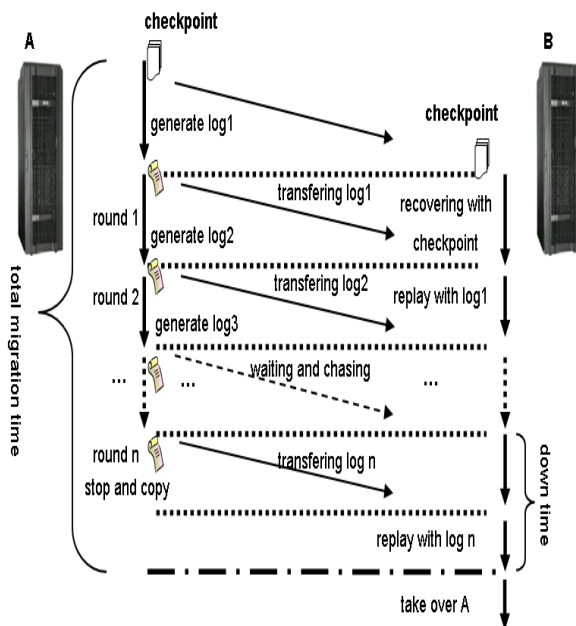


Fig: 5 process diagram in check point recovery trace and replay

The trace daemon is continuously monitor the discrete event happens in the system and generate the logs. In the first step of migration they transfer the checkpoint to the target host machine. After that they logs to the destination location. In source it iteratively generates the logs it transfers in to target. The process is continues up to the completely transfer the logs, at that time process is suspend at the source and resume at the target.

3.4 LIVE MIGRATION USING CPU SCHEDULING

In this approach it is trying to reduce the number of pages transfer to the target host by cutting down the CPU performance. They are trying to decrease the speed of CPU hence number of dirtied pages is very less. In the first step they assign the VM whole memory number of round set as zero. Then calculates the count transferring time , scheduled the CPU time to execute memory.

One advantage is that it has reduced number of pages transfer .disadvantage is service degradation of performance.

4. COMPARISON

In the pre-copy approach it have reduced total down time. After completing the transfer of all pages, then only they suspend the process at the source machine and resume at the target machine. But they are iteratively transferring the modified memory pages hence it have more total migration time. When pre copy is compared with all other live migration using compression, LRU and splay tree, it is reduced the number of pages transfer, but it have more total migration time compared with pre copy. In the post copy approach it first transfers the memory pages first, for that they suspend the process at source and resume on the target machine. All the process have take place during the resume phase hence down time is more. Post copy has the less total migration time. Hence we can say that no approach having the reduced total migration time, total down time, and reduced number of pages transferred. Check point recovery trace and replay having both reduced total migration time and down time. But one of the disadvantage is that the migration having reduced total migration and down time only when the log generation rate is less than the log replay rate.

A comparison is provided based on the analysis of different migration techniques. Comparison table is explained based on the experiment conducted on the application SPECWEB 2005. It is a web server benchmark involving at least 2 or more physical hosts, In which tests are take place in guest VM, in which six different clients are working on the VM.

Parameters	Pre copy	Post copy	Adaptive memory compression	LRU and splay tree	Check point recovery trace and replay	Migration using CPU scheduling
Pages transferred	180000-200000 bytes	120000-140000 bytes	600 MB	500 -600 MB	Logs are transferred.	100000 – 140000 bytes
Total downtime	600 ms	800 ms	600 ms	580 ms	620 ms	288 ms
Total migration time	10 sec	8 sec	100 sec	18 sec	8 sec	-

Table 1: comparison between different approaches

CONCLUSION

Live virtual machine migration is movement of virtual machine from physical host to other. Live virtual machine migration is helpful in server consolidation, power management, resource scheduling. There are several migration techniques are used for migrate from one host to other pre copy, post copy, adaptive compression, LRU and splay tree, check point recovery trace and replay method. The pre copy approach is one of the best approach it having both reduced total migration time and down time. Post copy having reduced total migration time but compared to others down time is more, due to this, implementing virtual machine migration with post copy is a tedious task. Adaptive compression approaches down time is same as that of pre copy but one fact is that huge increase in the total migration time. Check point recovery trace and replay having both reduced total migration time and down time but it applicable only with the log transfer rate is less than log replay rate. Live virtual machine migration using CPU scheduling it reducing the speed of CPU total number of pages are transferred is less but more down time. From the above we can conclude that no approach having reduced total migration time and down time.

References

- [1] Bhaskar Prasad Rimal, Eunni Choi, Ian Lumb “a taxonomy and Survey of Cloud Computing Systems” 2Fifth International Joint Conference on INC, IMS and IDC 2009.
- [2] NELSON, M., Lim, and Greg Hutchins” Fast transparent migration for virtual machines”. In Unix, Anaheim, Ca(2005), pp 25-25.
- [3] tiago c.ferreto, marco A.S Netto, Rodrigo “Server consolidation with migration control for virtualized data centers” by future generation computer system 27(2011)1027-1034
- [4] R. Nathuji and K. Schwan , “Virtual power: Coordinated power management in virtualized enterprise systems,” in *Proceedings of the 21st ACM Symposium on Operating Systems Principles (SOSP’07)*, 2007, pp. 265–278.
- [5] Clarck, C., Fraser, k., Hand, S., Hansen, J., Jule, E., Limpach, C., Pratti, I., And warfield, A.” Live migration of virtual machines”. In *Network System Design and Implementation* (2005).
- [6] H. Jin, L. Deng, S. Wu, X. Shi, and X. Pan.” Live VM migration with adaptive memory compression”. In *Proceedings of the 2009 IEEE International Conference on Cluster Computing (Cluster 2009)*, 2009.
- [7] Ei PhyuZaw “improved Live Virtual migration using LRU and splay tree algorithm” International

journal of computer science and telecommunications.

[8] M. R. Hines and K. Gopalan, "Post-copy based live VM migration using adaptive pre-paging and dynamic selfballooning", in Proceedings of the ACM/Usenix international conference on Virtual execution environments (VEE'09), 2009, pp. 51–

[9] H. Liu, H. Jin, X. Liao, L. Hu, and C. Yu, "Live migration of virtual machine based on full system trace and replay," in *Proceedings of the 18th International Symposium on High Performance Distributed Computing (HPDC'09)*, 2009, pp. 101–110.

[10] Sherif Akoush, Ripduman Sohan, Andrew Rice, Andrew W. Moore and Andy Hopper "Predicting the Performance of Virtual Machine Migration"

[11] Hai Jin a,n, Wei Gao a, Song Wua,n, Xuanhua Shi a, Xiaoxin Wub, Fan Zhou " Optimizing the live migration of virtual machine by CPU scheduling" *Journal of Network and Computer Applications* 34 (2011) 1088–1096

