ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(48), DOI: 10.17485/ijst/2016/v9i48/107961, December 2016* ISSN (Online) : 0974-5645

A Survey on Live Migration Techniques of Virtual Machines

S. Ambika, R. Ezhilarasie* and A. Umamakeswari

School of Computing, SASTRA University, Thanjavur - 613401, Tamil Nadu, India;
saamseeka@gmail.com,ezhil@cse.sastra.edu, aum@cse.sastra.edu

Abstract

Objectives: Cloud computing plays an imperative role in development of technology and makes the data available to the worldwide extent. Virtualization and virtual machines helps in enhancing the utility of the cloud computing to its maximum limit. **Method:** The method of replication of virtual machine in source physical machine to destination physical machine is known as migration of virtual machines. Live migration is a type of migration that deploys pre-copy, post copy, hybrid approach for transferring load. **Finding:** Load balancing is the foremost problems in data centres which can be handled by migration. This paper focuses on pre-copy approach and makes an investigation of the various pre-copy methods and its implementation. **Applications:** It helps in improving the utilization of data centres and load management with minimum downtime and no page faults.

Keywords: Cloud Computing, Live Migration, Load Balancing, Pre-copy Approach, Virtual Machines

1. Introduction

Emerging trends in computing makes the technology available worldwide. Its transformation makes everything accessible to anyone anywhere. Cloud computing is one among them. Cloud computing enables consumers to store and access the resources anywhere. Resources include networks, servers, storage, applications that can be accessed by everyone with the help of service providers which is illustrated in Figure 1.Virtualization[1] is the vital feature in cloud computing and one of the key ideas of data centre administration. It improves the utilization of the resources through virtual machine. A Data Centre (DC) generally conveys countless Machines (PMs) to boost space usage. The real point of preference of virtualization is the possibility of running a few working virtual machines (VM) on a solitary PM. The processing assets of DC are made accessible to the clients through VMs. Figure 2 demonstrates that VMs abstracts the hardware and appears as software with the help of hypervisor or virtual machine monitor. Migration is a technique to manage the resources used by the virtual machines. Applications residing on one physical machine can be moved to another physical machine for load balancing, energy saving is called migration of virtual machines. Application migration is the most complicated and challenging[2] task in migration. As there are 3 types of migration algorithms[3], Kernel memory, application states are the data for migration. Migration facilitate efficient resource utilization, load balancing, energy management. There are 2 techniques formigration of virtual machines which are Non live migration techniques[4] and Live migration. In Non-Live migration, Cold migration is the process of migrating guest OS and its application after shutting down the OS. Hot migration migrates the guest OS and its application while the OS is suspended. Live migration technique will help in migrating the virtual machine without suspending or interrupting guest OS. It contains 3 phases which are warm- up phase, iteration phase, stop and copy phase. These three phases are implemented using any one of the three approaches such as pre-copy approach, post copy approach and hybrid approach.

* *Author for correspondence*

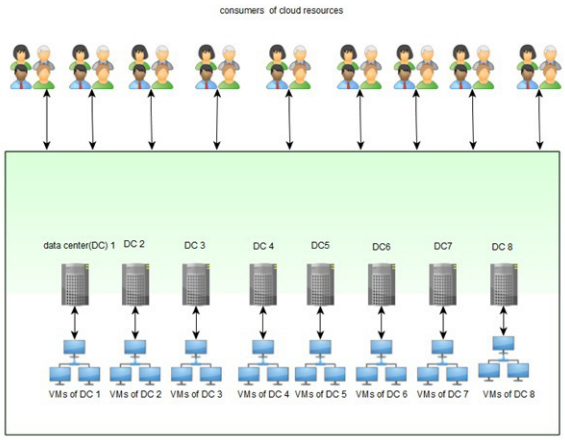

Figure 1. Structure of cloud computing.

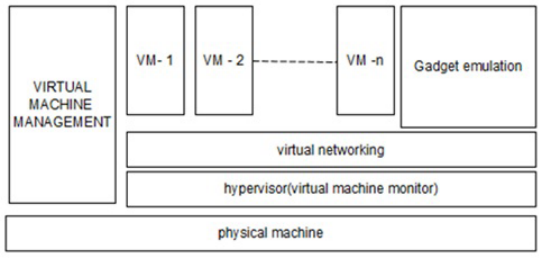

Figure 2. Structure of a physical machine.

2. Post-copy Approach

In this approach, warm up phase is carried out at first in which the destination physical machine is selected. In the next stop and copy phase, the virtual machine is migrated to the target host. Memory pages are not copied so page faults occur at the destination host. On the occurrence of the page faults, memory pages are duplicated at the respective target host which can be referred as the iteration phase.

3. Pre-copy Approach

Pre-copy approach[5] contains preparation phase, iterative synchronization phase, final synchronization phase, Resumption phase. In preparation phase,(warm-up phase) Live Migration Agent(LMA) on source virtual machine will inform destination live migration controller to reserve resources.In Iterative synchronization phase, LMA migrate the executive context to destination iteratively. In Final synchronization phase, LMA migrates the left over pages to the destination virtual machine. VM should be suspended at the source by the LMA and started at the destination. Then it migrates the remaining context. Speed of the dirty page migration plays a vital role in success of live migration. If the speed of dirty page generation is more than the speed of migration of memory pages then it will take long time for final synchronization. To overcome the performance degradation each virtual machine is allocated with a VCPU to control the dirty page generation at each round. The time required for final synchronization can reduced with help of the above method calledCPU scheduling. Synchronization manager[6] is necessary to manage the migration which monitors the entire pre-copy approach in all phases. In Resumption phase, VM is resumed at the target host and starts running

4. Methodologies for Pre-copy Migration

In[7] enhanced the conventional pre-copy approach. Modified pages are tracked by using a three bitmap approach. The First bitmap represents the memory pages modified at the previous iteration, the second bitmap represents the memory pages modified at present iteration and the third bitmap is used for denoting the number of memory pages sent at the previous iteration. This approach mitigates the needless transfer of memory pages at each round but cannot be applied to the alternative modified pages. Thus it will reduce the migration time but not the down time. Power consumption[8] is major side effects of the live migration. So the power efficient deployment in the data centres is ultimate challenge in migrating virtual machines. A cost effective model for migration will be based on the CPU utilization percentage. The important fact revealed in his work is, as the CPU utilization increases the migration quality decreases and power consumption increases only in the server system. The migration will not affect the destination physical machine. The power of the source machine decreases when the CPU utilization increases in the running virtual machines. But the same will not influence the destination physical machine. In proposed the actual arrangement of the memory in the virtual machines in Figure 3.Hypervisor

will contain configured memory which can be used by the virtual machines but it cannot be fully utilized by the virtual machines.In configured memory, only allocated part (allocated memory) will be accessed by the virtual machines where virtual memory will consume only a fraction of the allocated memory. From this fraction, we can calculate the dirtied pages and send it to the destination. For framing the memory construction of the physical storage device this framework will be very useful and this helps in manipulating the amount of memory to be moved to the destination. In[9], proposed Zephyr method to professionally migrate an alive database in a non-public transactional database structural design. Zephyr make use of stages such as demand drag and asynchronous drive of data, that necessitate negligible synchronization. It results in service availability of databases any time with a small number or no terminated communication. It also reduces the data transport overhead, offer ACID promise at the time of migration and make sure correctness in the existence of failures.

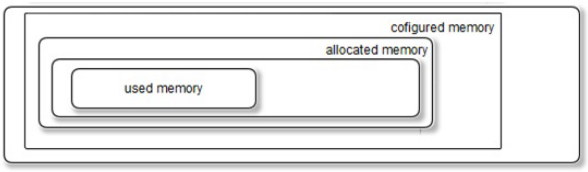

Figure 3. Organization of memory.

5. Performance, Energy, Cost Modeling in Live Migration

In[10]states that factors that influence the live migration size of VM memory, network bandwidth, dirtying rate will create some effect on performance of migration. Performance model is designed based on the migration parameters. On basis of performance model a linear model that estimate migration energy is designed. This model will help administrators for most excellent decision making on migration and also helps in developing optimal algorithm for migration. When the cost of traffic mechanisms is reduced the performance will not be improved[11]. For improving the performance, LRU and Splay tree algorithms are proposed by[12]. LRU depends on the time of the page last used. It replaces a page which is not used for a long time period of time when a new page is arrived. It uses stack to store and replacing. Recently used page will be in top of the stack when the stack is filled, the least recently used page which is at the bottom will be removed from the stack. Splay tree algorithm will arrange the pages in the form of tree where the least used page will be at the root. When a node is added to the tree then root node will be replaced. It is used for the prediction of the future use of memory pages. Prediction time is the performance blockage for this method.In[13] used a modified clock policy to eliminate the pages. This strategy proposed by in has a pre-processing stage. In this stage, each recently used memory pages are given a last used field. Pages which are not frequently used can be identified by this strategy and sent to the destination remaining pages are assigned to the working set[14] list. Pages of the Working set list will be migrated to the destination in the final synchronization phase. This approach will decrease the total migration time and downtime whereas maintaining the working set list is the will be an overhead. A cost model[15] is defined for live migration. Features that influence power consumption and performance are CPU, network bandwidth, memory. So based on these features the manipulation of the cost model can be done. The metrics for measuring the performance of the live migration are whole migration time and downtime. Overall migration time can be computed from the following terms such as time necessary for electing the target physical machine $TpMig$, time required for reserving resource $Trsv$,

Then total migration time will be as follows.

$Tmig = Tp\square ig + Trsv + TpCopy + TsCopy + Tsub + Tact$

$TdTime = TsCopy + Tsub + Tact$

$TpMig$ = time required for selecting free target node which has free space in advance

$Trsv$ = time required resources reserved for VM in future

Tpcopy = repeated copy of data

Tscopy = stop and copy state

Tsub = completing the submission of memory in target node

Tact = activating VM in target host

In live migration, initialization phase and resumption phase is the essential phase it consumes time as Initialization phase *Tini* = *Trsv* + *Tpmig*

Resumption phase *Tresum* = *tsub*+ *tact*

Thus the migration cost can be calculated phase by phase. Each phase influences the cost of live migration.

6. Algorithms for Live Migration

In[16] proposed two algorithms under assumptions such as virtual machine is a tiny workload that can be transmitted among hosts, multiple virtual machines cannot be transmitted in a single iteration, CPU load can be the only indicator. Objective of the algorithms are balancing the load in the system by node cooperation in the automated manner. Guest OS state is replicated in the destination host. Authors focused on the CPU state, memory content. Hotspot indicates the host which has CPU load more than the threshold. Author proposed Push and Pull strategies for automating the live migration. In Push strategy, overloaded PM will grab the role seller then it broadcast the specifications required for the migration to the other PMs (buyers). Buyers will reply with the available resources and usage of them to the seller. Seller decides on the best bid and executes migration. Pull strategy can be executed when the seller is underutilized. It will market the available resources among buyers, get hold of the perfect VM and attain stability. In[17], proposed the first technique for compressing the memory that mitigate the downtime and migration time. A character based algorithm is proposed for the live migration. MECOM uses memory compression technique for providing quick migration time and reducing downtime by compressing the pages of the memory. In this method pages are packed together in batches and recovered in destination host without loss. Compression technique takes word similarity as its parameter. Thus the compressed pages are transmitted over the destination and recovered. This memory compression technique works well in diversified networking environment with variable network bandwidth. In this technique, memory pages are compressed in source and can be decompressed at the destination host. Potential profit will be based on the compression proportion and compression speed. If the similarity between pages are elevated, then the amount of pages can be reduced to a maximum extent. In[18] proposed fast live migration mechanism with small I/O performance penalty. It uses storage area network to achieve quick live migration. Features of fast live migration mechanism are Increasing speed of migration by utilizing storage area network of the data centres to transport page cache. It partitions the page cache into two parts to achieve maximum utilization of the network.

Detailed study of the algorithms is as follows:

1. Virtual machine monitor receives command to execute migration in the source physical machine. Pages of the user processes are given Page Frame Number (PFN) of the cached pages and similar disks blocks.
2. Kernel will get these PFN of the cached pages and similar disk blocks.
3. User process will send PFN to the source and destination.
4. Memory transportation finished. SAN is utilized for the live migration.
5. Memory pages restored at the destination.
6. Other memory pages are copied using normal network
7. Once all the memory pages are copied and received all cached pages VM is started in destination host.

Thus the storage area network is utilized for the migration of the virtual machines. One bitmap (to skip 0 or to send 1) value will be allocated for each page in each iteration. Bitmap values of all the pages will be formed as a matrix. MAP-LEN[19] is used for denoting the accumulated sum of these bitmaps at every iteration. Many bitmaps are employed for deciding page transportation between source and destination is shown in Figure 4. Weight for a given page will be computed in association with the threshold value (2map-len-1). Pages having weight more than the threshold value will not be forwarded to the destination. This optimum threshold value alone makes the algorithm get success. The summary of different methods and models is shown in Table 1.

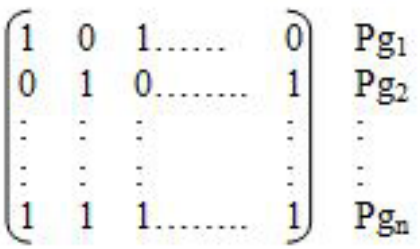

Figure 4. Transportation between source and destination.

Table 1. Summary of the live migration models

S.No	Method Proposed	Working	Drawback AndFuture Work
1	Fast live migration	Pages of the user processesare given page frame number (PFN) and transmitted from source to destinationusing storage area network	I/openalty will be the drawback of this approach which affects the QOS
2	Compression based live migration approach	Memory pages are compressedusing batch compression technique	Compressioncost present in the proposed method and requires huge disk space for holding memory data
3	Automated algorithms for live migration	Based on theadaptive threshold on CPU utilization migration occurs automatically	Multiple virtual machines cannot be transmitted, CPU load is the only threshold measure
4	Performance and energy modelling	Modelestimates the performance and migration costbased on the based on the migration parameters.	Future work of the Model should provide guidance on basis of the performance requirement and performance metrics
5	cost model for live migration	Compute the performance and cost of the migration of the each phase. Migration time, downtime, resumption time are calculated to determine performance.	Cost model can include the QoSand SLA parameters in future to help in decision for the migration.
6	MECOM A compression module	Memory pages will be compressed in batches and recovered at the destination host.	Compression cost will be the overhead
7	Working set strategy	Pages are classified time to time on the basis of the utilization. Unused pages are send over the destination first which helps in minimizing the migration time.	A pre- processing phase and maintaining a working set are the overhead
8	Zephyr method	Migrate a livedatabase by using on demand drag and asynchronous drive of data. This results in the service availability anytime with minimum or no terminated communication	Synchronization required between source and receiver nodes
9	Improved pre-copy approach	Pages are tracked using3 bitmap approach. It minimizes the unneeded transfer of the pages.	It is not applied to the alternative modified pages
10	Bitmap algorithm with help of matrix	When the pages having weight more than thethresholdvalue will not be forwarded to the destination host	Strength of the algorithm is only based on the threshold value.
11	LRU and Splay tree algorithm	Predicts the pages that will be used in future.	Prediction time is the bottleneck
12	Working set along with clock replacement[20] strategy	Determine the frequently used memory pages by using pre-processing stage.	Maintaining Working set with time

7. Conclusion

Live migration technique is used for balancing load in data centres without interrupting other running virtual machines. This paper illustrates the various pre-copy based live migration techniques which contributed in minimizing the migration time, downtime and increasing the performance. It also analyzes the algorithms that are proposed for overcoming the issues related to the pre-copy approach. Each proposed method will contain its own drawback and introduces an overhead. It is tough to judge the different proposed methods as each of them is implemented in various architecture and platforms. It is also hard to standardize the performance of the migration.

8. References

1. Shetty J, Anala MR, Shobha G. A survey on techniques of secure live migration of virtual machine. International Journal of Computer Applications. 2012 Feb; 39(12):34–9.
2. Guay WL, Reinemo S-A, Johnsen BD, Yen C-H, Skeie T,Lysne O, Tørudbakken O. Early experiences with live mi-

gration of SR-IOV enabled InfiniBand. Journal of Parallel and Distributed Computing. 2015 Apr; 78:39–52.
3. Singh A, Juneja D, Malhotra M. Autonomous agent based load balancing algorithm in cloud computing. Procedia Computer Science. 2015; 45:832–41.
4. Divyambika R, Umamakeswari A. Protection of virtual machines during live migration in cloud environment. Indian Journal of Science and Technology. 2015; 8(S9):333–9.
5. Mallu L, Ezhilarasie R. Live migration of virtual machines in cloud environment: A survey. Indian Journal of Science and Technology. 2015 May; 8(S9):326–32.
6. Lu T, Stuart M, Tang K, He X. Clique migration: Affinity grouping of virtual machines for inter-cloud livemigration. IEEE Computer Society; 2014. p.216–25.
7. Live virtual machine migration based on improved pre-copy approach[Internet]. [cited 2010 Jul 16]. Available from:http://ieeexplore.ieee.org/document/5552416/.
8. Huang Q, Gao F, Wang R, Zhengwei. Power consumption of virtual machine migration in clouds. 2011 Third International Conference on Communications and Mobile Computing;2011. p.122–5.
9. Elmore AJ, Das S, Agrawal D, El Abbadi A.Zephyr: Live migration in shared nothing databases for elastic cloud platforms. SIGMOD '11 Proceedings of the 2011 ACM SIGMOD International Conference on Management of data; 2011. p. 301–12
10. Liu H, Xu C-Z, Jin H, Gong J, Liao X. Performance and energy modeling for live migration of virtual machines. Proceedings of the 20thInternational Symposium on High Performance Distributed Computing; 2011. p.171–82
11. Qiu L, Yang R, Zhang Y. **On selfish routing in internet-like environments**. Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications SIGCOMM '03; 2013. p.151–62.
12. Zaw EP, Thein NL. Improved live VM migration using LRU and Splay tree algorithm.International Journal for Computer Science and Telecommunication. 2012 Mar; 3(3):1–7.
13. Optimized live migration using NRU and modified clock policy[Internet]. [cited 2015 Apr 16]. Available from: http://ieeexplore.ieee.org/document/7156325/?reload=true&arnumber=7156325.
14. Hines MR, Gopalan K. Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning. Proceedings of the 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (ACM); 2009. p.51–60.
15. Costs of virtual machine live migration: A survey[Internet]. [cited 2012 Jun 24]. Available from: http://ieeexplore.ieee.org/document/6274069/.
16. Forsman M, Glad A,Lundberg L, Ilie D. Algorithms for automated live migration of virtual machine. Journal of Systems and Software. 2015 Mar;101:110–26.
17. Jin H, Deng L, Wu S, Shi X, Chen H, Pan X. MECOM: Live migration of virtual machines by adaptively compressing memory pages. Future Generation Computer Systems. 2014 Sep; 38:23–35.
18. Akiyama S, Hirofuchi T. Fast live migration with small I/O performance penalty by exploiting SAN in parallel. Cloud '14 IEEE Conference on Cloud Computing;2014. p.40–7.
19. Live memory migration with matrix bitmap algorithm[Internet]. [cited 2010 Aug 16]. Available from: http://ieeexplore.ieee.org/document/5607439/.
20. Performance enhancement in live migration for cloud computing environments[Internet]. [cited 2013 Feb 21]. Available from: http://ieeexplore.ieee.org/document/6508339/.