Using Queueing Model to Analyze The Live Migration Process in Data Centers

Chuan Pham, Choong Seon Hong
Department of Computer Engineering, Kyung Hee University
1 Seocheon, Giheung, Yongin, Gyeonggi, 449-701 Korea
E-mail: {pchuan, cshong}@khu.ac.kr

Abstract

In recent years, cloud computing systems have grown with an explosion. And, in the future, they will play a quite important role in the development of Future Internet area. There are many researches that have been introduced to be a potential technology for the cloud computing technology. One of them, live migration is the hot topic, which is considered as a key management function in cloud computing. In this paper, we propose an analysis method to enhance efficiently the live migration of virtual machines by using queueing model.

1. INTRODUCTION

Virtualization technologies support more efficiently in cloud computing and they become one of the most notable features in this area. Today, many large companies such as Google, Yahoo, and Microsoft have a lot of services and commercial products which are developed in their private clouds. In the future, cloud computing will be a promising area for academia and business. The main technique for cloud computing is to hide the complexity of physical infrastructures which are operated underlying such as hardware or software and to make the available services which are easy to use by the end users. However, adapting to the customers' requests efficiently, data centers must deploy many operating services and functions to control the requests and virtual machines. The important function which has more consideration, is the migration solution.

In this paper, we explore the live migration process and analyze with the queueing model. As we may know, live migration describes the process of copying a resource of one physical machine to another physical machine, while the VM is still powered on [2]. In during the copying process, the resources are still being used by the services. Hence, in many researches, they often use the term "dirty pages" or "dirty blocks" [1,2] to mention the resources such as memory or storage are modified within the migration process. These dirty blocks must be recopied again to the destination host. However, one critical challenge is that the total number of dirty pages/blocks during migration is available that is not known before the migration completes [1]. To obtain this parameter, we use queueing model to analyze the total number of dirty blocks, an important variable, is used in migration control. Our main contributions are as follows:

- Formulating a queuing model M=M=1 for migration. Base on this model, we calculate the threshold to control the migration.
- Simulating the migration process and evaluating with the prediction threshold method [2].

The rest of the paper is organized as follows. The live migration are reviewed in Section 2. The system model and the analysis of the pre-copy process are presented in Section 3, 4. The simulation results are reported in Section 5 and the conclusion of the paper is given in Section 6.

2. LIVE MIGRATION OF VIRTUAL MACHINES

In this part, we re-describe again the migration process which is introduced in [2,3,4]. The full migration of a VM is affected by many aspects such as: the running state of the VM, the storage, and the connection of networks. In all stages below, we will briefly represented the resources, components and the process of live migration of the modern virtualization systems. To easy understand this process, we represent with three stages pre-migration, Iterative Pre-Copy and Hard copy.

- Stage 1: Pre-Migration. In this stage, VM on source host prepares the resources to move to another host. The migrating controller will preselect where the resources required to receive migration. This step will guarantee the safety migration process.
- Stage 2: Iterative Pre-Copy. During this stage, VM on the source host is still running. The memory/storage blocks are iteratively copied from the source to the target host. By “iterative”, it means that pre-copying occurs in

rounds, in which the pages to be transferred during round n are those that are modified during round n-1 [3]. As each round takes some time and in the mean time the VM is still running on the source host, some memory/storage pages may be dirtied and they must be resent in the additional rounds to ensure memory/storage consistency. This iterative pre-copy stage will be stopped by the stop condition. In traditional methods, they often follow thresholds: the number of iterative rounds, the total amount of memory that has already been transmitted, the number of dirty pages which still remain in the source host, or the limitation of transmission time [2].

- Stage 3: Hard copy. At this stage, this hypervisor suspends the VM on the source host to stop page dirtying and hard copies the remaining dirty pages to the destination host. After ensure all physical hardware consistency, the hypervisor on the target host resumes the VM [2].

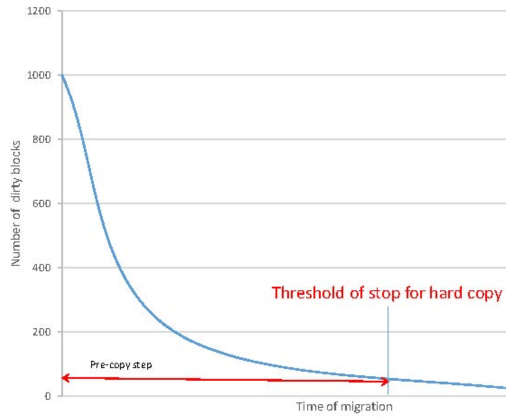

Figure 1: Number of dirty blocks in live migration

The matter of concern in this process to enhance efficiently migrating time is the threshold in stage 2. In this work, we base on the queuing model to analyze this the total number of dirty pages. This parameter is already mentioned by [1]. In that paper, the authors defined the dirty set which is calculated by the previous writing. But in the huge of VMs in datacenters, the calculation of this method will be significantly severe. With difference point of view, we consider live migration process like a queueing model. In that, we represent the process and analyze the stable in our system to the live migration policy.

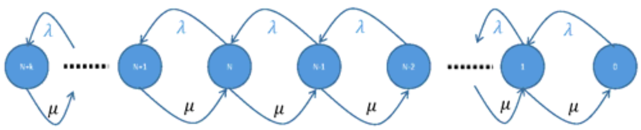

Figure 2: The transition diagram

3. SYSTEM MODEL

The pre-copy migration model is widely used in KVM system and XEN system [1]. According to these system, in the virtual disk migration, all the writing operations from services or applications will be logged. The dirty blocks generated during the migration are again logged and retransmitted. This iterative is performed until the number of dirty blocks falls below a threshold, and then the memory migration begins [1]. However, it is too difficult to calculate this threshold accurately. Hence, there are many researches proposed the prediction threshold to identify the number of dirty iteration. In this our work, we model and analyze this problem by queuing theory.

Observing the migration process, in general, we assume that at the current time t, the migration VM has N dirty blocks. The blocks of VM will be dirty with an inter arrival time followed the exponential distribution. Based on the moving block from source VM to destination VM, each block can be served and finished with an exponential distribution. We denote that the dirty arrival rate is λ and μ is the rate of moving. From the Figure 1, if the time of pre-copy stage goes to the infinitive, the threshold of stop will closed to the average number dirty blocks in the system.

During the migration process, the transition of total number of dirty blocks is change from N + k to 0, with k is the new dirty blocks. By the Markov mode, Figure 2 represents the transition state of migration process [5].

4. STEADY STATE ANALYSIS

The transition diagram used to calculate the probability of all states in the system in steady state.

$$\begin{cases} \lambda P_0 = \mu P_1 \\ (\lambda+\mu) P_1 = \lambda P_1 + \mu P_2 \\ \vdots \\ (\lambda+\mu) P_N = \lambda P_{(N-1)} + \mu P_{N+1} \\ \vdots \\ \mu P_{N+k} = \lambda P_{N+k-1} \\ P_0 + P_1 + P_2 + \ldots + P_{N+k} = 1 \end{cases} \quad (1)$$

From (1), we derive the probability of each state:

$$\begin{cases} P_1 = \frac{\lambda P_0}{\mu} \\ P_2 = \frac{\lambda}{\mu} P_1 = \left(\frac{\lambda}{\mu}\right)^2 P_0 \\ \vdots \end{cases} \quad (2)$$

In general,

$$P_i = \left(\frac{\lambda}{\mu}\right)^i P_0,\ i \geq 0$$

From (1,2), we denote $\rho = \frac{\lambda}{\mu}$ and derive P_0 and P_i.

$$\sum_{i=1}^{N+k} P_i = 1$$
$$P_0\left(1+\rho+\rho^2+\ldots+\rho^{N+k}\right)=1$$
$$P_0\sum_{i=0}^{\infty}\rho^i = P_0\sum_{i=0}^{\infty}\left(\frac{\lambda}{\mu}\right)^i \approx P_0\left[\frac{1}{1-\frac{\lambda}{\mu}}\right]=1 \quad (3)$$
$$P_0 = 1-\frac{\lambda}{\mu},$$
$$P_i = \left(1-\frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^i, i \geq 0$$

The expected of dirty block in the system can be solved as following:

$$E\left[\text{Number of dirty blocks in the system}\right] = \sum_{i=0}^{\infty} iP_i$$
$$= \sum_{i=0}^{\infty}(1-\rho)\rho^i = (1-\rho)\sum_{i=0}^{\infty} i\rho^i =$$
$$= (1-\rho)\rho\sum_{i=0}^{\infty} i\rho^{i-1} = (1-\rho)\rho\sum_{i=0}^{\infty}\frac{\partial\rho^i}{\partial\rho} \quad (4)$$
$$= (1-\rho)\rho\frac{\partial}{\partial\rho}\sum_{i=0}^{\infty}\rho^i$$
$$= \frac{\rho}{1-\rho}$$

Applying the Little' law, we calculate the average waiting time in the system:

$$W = \frac{1}{\mu-\lambda} \quad (5)$$

From (5), we can obtain the expected time for migration N blocks. This value is the important value to find out the duration time of migration.

5. SIMULATION AND NUMERICAL RESULTS

In this part, we represent the simulation result of our analysis to find out the threshold of stop pre-copy process in migration a VM. We create the stop threshold with 70% the expected time which is analyzed and calculated. This value is chosen after running iteratively many times. The simulation model is operated with 20 VMs and each VMs is created randomly total number of blocks. The size of each block in this model is defined by 1MB. In addition, we define the dirty rate 10MB=s and the moving rate 100MB=s. Without monitoring the migration process, our model can easy capture the threshold to stop pre-copy process that is closed to the threshold in the traditional method. The Figure 3 shows the values of this threshold of 100 VMs and compare to the traditional methods.

6. CONCLUSION

In this paper, we propose an analysis model for live migration on VM. The proposed approach is used queueing M/M/1 to model the live migration process and find the stop threshold in pre-copy stage. Without monitoring the writing process of VM, our model can easy capture the threshold based on queueing model.

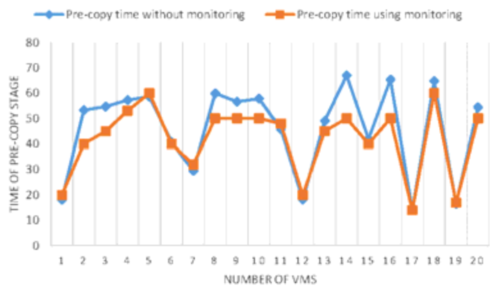

Figure 3: The duration of pre-copy stage.

This proposed approach has a significant effect to reduce the cost of migration in data center with a huge VMs. The analysis and simulation show that our proposed system is adaptable and can be applied in data centers.

Acknowledgment

This research was supported by the MSIP (Ministry of Science, ICT&Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2014(H0301-14-1020)) supervised by the NIPA (National IT Industry Promotion Agency). *Dr. CS Hong is the corresponding author

REFERENCES

[1] J. Zheng, T. Ng, K. Sripanidkulchai, and Z. Liu, "Pacer: A progress management system for live virtual machine migration in cloud computing,"2013.

[2] A. Strunk, "Costs of virtual machine live migration: A survey," in Services (SERVICES), 2012 IEEE Eighth World Congress on. IEEE, 2012, pp. 323–329.

[3] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2. USENIX Association, 2005, pp. 273–286.

[4] 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 ACM international symposium on High performance distributed computing. ACM, 2009, pp. 101–110.

[5] V. Vinothina, D. R. Shridaran, and D. P. Ganpathi, "A survey on resource allocation strategies in cloud computing," International Journal of Advanced Computer Science and Applications, vol. 3, no. 6, pp. 97–104, 2012.