ISSN:0975-9646

Techniques for Reducing the Down Time during Virtual Machine Migration: Resource Failure & Live Migration Circumstances

Dr. Abhay Kothari,
Professor,
Acropolis Institute of Technology & Research, Indore

Ashish Agrawal,
Asst. Professor,
Acropolis Institute of Technology & Research, Indore

Abstract:Virtual Machines need migration from the present host to an another appropriate one due to error in the present hardware or presently loaded scenario causing slowness of execution. Most of the situations fall into second category leading to a solution called live migration. Here, in live migration the virtual machine runs for sometime on the present host while its required data gets transferred to the destination. The prevalent techniques under live migration are pre copy and post copy having their relative merits and demerits. In this paper, we are presenting our work in terms of proposing several techniques for better designs, namingly, Remote DMA, parallel buses, NUMA, server consolidation and blade server and in case of OS virtualization virtual file system implementation is proposed. In case of hardware failure, provisioning of redundant hardware component is included in this paper. All these techniques are there to enhance the data transfer and sharing capacities beyond what is offered by pre and post and other prevalent designed techniques so as to reduce the downtime.

INTRODUCTION:

There are two types of virtualizations hardware virtualization and OS virtualization. In hardware virtualization, multiple virtual servers called as virtual machines are created; they perform the application execution on their virtual servers utilizing hypervisor software to make this happen on real hardware. Increasing the number of virtual machines on a host server leads to better utilization of server resources but sometime may cause overload and slowness of execution. In such situations any particular virtual machine identified as slow running one needs live migration to another host [1]. Virtual machine migration may also be required if any critical hardware component is diagnosed as faulty. In this case, migration in the form of shutting down the virtual machine on the present server and restarting it on the destination host takes place. Live migration has a necessity to transfer data in a faster way because the migration is gradual, for sometime the VM runs on the present host and data is transferred to destination host where it is moved eventually. There are two techniques to do live migration pre-copy where data belonging to the VM is transferred to the destination as a whole and then VM is started on destination. If data is large the migration is slow. In post-copy the most necessary data to start the VM is transferred, VM is started on host and rest of the data is transferred on demand basis. For faster data transfer as needed in either case more in pre-copy a modified technique of I/O transfer which is RDMA (remote direct memory access) is included in this paper, which is like DMA (direct memory access) but it is applicable when communication has to take place between two separate computers, one supplying the data and another receiving it. DMA or RDMA are much faster then the conventional I/O techniques of 'Polling' and 'Interrupt driven I/O' as they don't involve CPU [2]. Use of parallel buses as multiple buses instead of a single bus as commonly used in RDMA is also included. Another technique for better data communication is the NUMA (non – uniform memory access) here different computers have different memory access times for accessing there local and other computers memory which is termed as remote access through some interconnection network. Consolidation is to substantially increase the efficient use of server resources which is advantageous but it may also increase the complexity of the configuration of data, applications, resources and servers that can be confusing for a normal user to share with. To overcome this problem, we have made use of server virtualization approach or blade server. In blade server, number of servers is placed in proximity with some common resources. In case of OS virtualization, for implementing virtual machines use of file system virtualization to handle the problem of heterogeneity of file system belonging to different guest OS is included in this paper.

Rest of the paper is organized as:

1. Reasons of VM migration
2. Migration due to fault in hardware and remedies.
3. Live migration.
4. RDMA and server consolidation as proposed for live migration
5. Sharing of memory addressed with ccNUMA
6. File system differences in source and destination host and their solution with file system virtualization accomplished with pod (PrOcess Domain).

Reasons of VM migration

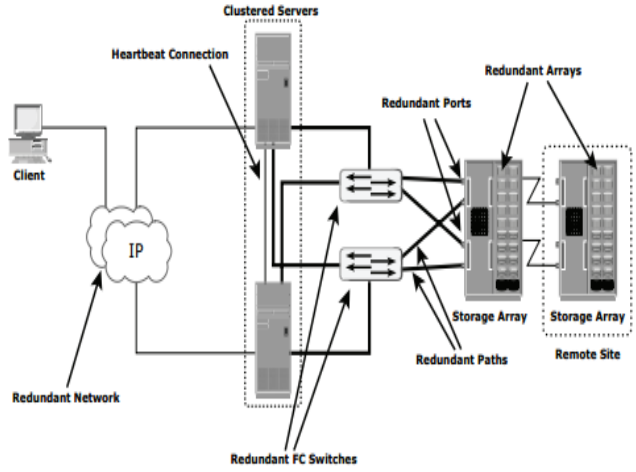
VM migration is majorly caused due to two main reasons- when the hardware components prominently CPU goes faulty, secondly when CPU is slowing down due to load and not likely to fulfill the service level agreement of response time or other parameters pertaining to limitations on processing time or volume of processing.

When hardware is erroneous or faulty

How to detect the faulty state of hardware;

The symptoms of a faulty CPU could be wrong result, different result for same data and instruction when computed repeatedly.

Hardware components should be subjected to diagnostic routines on a periodic basis of the order of one or few instruction execution time and in case any fault found in a particular hardware component, remedial steps should be taken in the form of moving the work to another provisioned redundant hardware component of the same nature. This could be illustrated with a situation in information storage management scenario with following diagram [4]:



In this diagram, if any of the networks goes faulty then other network can be started to continue the work. If the server in front of the user goes faulty the heartbeat network connection can work with the other server. Redundancy is cost oriented solution.

Live Migration:

Live migration is recommended when present hardware is not likely to serve the need of the virtual machine on which desired application is running. This may be due to its utilization for other purposes. Every CPU and hardware resource has its capacity in terms of multiplicity of jobs or tasks it can handle offering advantages of multiple processing.

How to detect the utilization coefficient of CPU and other hardware resources;

CPU can respond slower is an indication of higher number of jobs given than it can handle. Increased number page fault may indicate the same for memory.

Utilization Efficiency U = time per task * tasks arrival rate

U = 1 when it's computed value is greater than 1, its normal range is 0 to 1.

What data needs to be transferred for virtual machine migration?

Application processes status in the form of process control block, register values, intermediate results, Data files, object files.

Memory Replication during migration

In either of pre copy and post copy live migration techniques during migration the data is present on the source as well as its copy on the destination. This is a duplication phenomenon which is not desirable as it involves over requirement of memory.

The pre copy and post copy are elaborated in the following topics.

Factors contributing to delay of data transfer:

This is primarily due to adoption of slow I/O techniques such as interrupt driven I/O or Polling. Delay can also be caused due to volume of data and incorporation of single bus systems.

Techniques for cutting down this delay:

When VM is migrated its down time is given as below:

Down time = data transfer time + time required on destination host to start VM under migration

As down time depends on data transfer time there is an immense need t cut it down. For this purpose following techniques can be used:

1. Use of Direct Memory Access (DMA)
   In transferring data from one host to another during VM migration amongst polling, interrupt driven I/O and DMA, the last one is the best technique. This relinquishes the CPU from its bus controlling responsibility and data gets read or written in big blocks directly between I/O device and memory.
2. Data compression: This can be used for large volume of data transfer.
3. Use of global memory: Not all data pertaining to the application running of VM is needed on the other host immediately to keep it running on destination host. If big data files are stored in global memory accessible to the other hosts, they need not be transferred during VM migration.
4. Deploying high speed network
5. Data to be processed in next few instructions should only be considered for transferring.

Live Migration with RDMA and Server consolidation: Proposed Approach

Live migration means a running VM is moved from one physical host to another. Live migration allows you to move an entire running virtual machine from one physical server to another, with minimal downtime. To migrate a running VM across distinct physical hosts, its complete details have to be transferred from the source to the target host. The state of VM includes the information about permanent storage, the volatile storage, connected devices, and the internal state of virtual CPU.

The worldwide approach for live migration is pre-copy in which the contents of VM's memory are first transferred to the target host and then VM is restarted. To keep the downtime, the time during which the VM is not running, to a minimum, data is sent in several iterations (i.e., only the page that has been modified since the last round were sent) while the VM keeps running on the source host. Another approach is post-copy, in which only the VM's VCPU and device state is sent to the target host and restart them immediately. Memory pages accessed by VM are then fetched in parallel and on-demand while VM is running on the target host. [1]

Incase of large amount of data to be transferred use of parallel data buses is recommended. Here, sender will dispatch first few pages on the first bus and next few pages on parallel bus. At the receiving end data from the parallel bus will be buffered and memory will be allocated according to the paging scheme of the sender. This type of

arrangement will expedite the data transfer in pre-copy and virtual machine can be migrated relatively sooner.

RDMA:

Live migration provides 3 quality options to reduce the time required to live migrate a virtual machine. Either we can use memory compression [2], or we can choose Remote Direct Memory Access (RDMA) [3], or we can choose multichannel network adapters.

Above options can support our private cloud infrastructure by:

- Increasing the efficiency of live migration when our hardware resources are constrained (memory compression).
- Increasing the scalability of live migration when our hardware resources are not constrained (multi-channel network adapter and RDMA)

In environments where hardware and networking resources are constrained, live migration delivers performance improvements for migrating virtual machines by compressing the memory data before sending it across the network. This utilizes spare CPU capacity available in the server running Hyper-V. Hyper-V closely monitors the CPU requirements of the virtual machine and only consumes an appropriate amount of CPU resources to quickly move virtual machines from one server to the next [2].

In environments where networking resources are not constrained, you can configure live migration to use multi-channel network adapters or RDMA-enabled network adapters, which reduces the time required to live migrate virtual machines. RDMA is able to perform a direct memory access from the memory of one computer into that of another without involving the operating system. This permits high-throughput, low-latency networking and delivers greater efficiency with live migration [3]

Server Consolidation:

Server consolidation is an approach to the efficient usage of computer server resources in order to reduce the total number of servers or server locations that an organization requires. According to Tony Iams, Senior Analyst at D.H. Brown Associates Inc. in Port Chester, NY, servers in many companies typically run at 15-20% of their capacity, which may not be a sustainable ratio in the current economic environment. Businesses are increasingly turning to server consolidation as one means of cutting unnecessary costs and maximizing return on investment (ROI) in the data center [11].

Server consolidation lets your organization [9]:

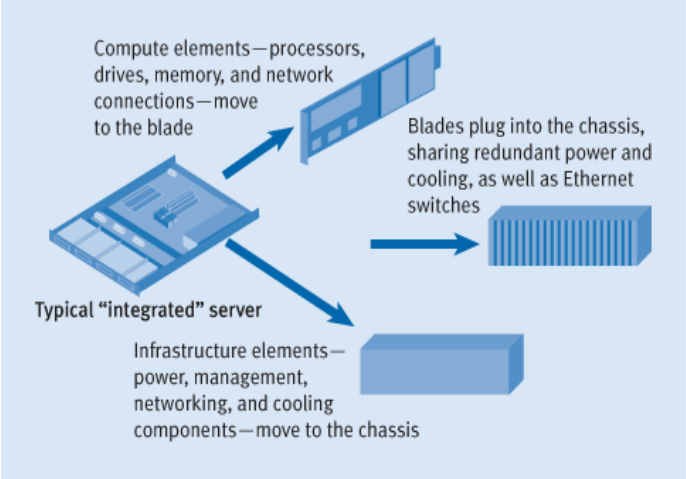
- Reduce hardware and operating costs by as much as 50 percent and energy costs by as much as 80 percent, saving more than $3,000 per year for each virtualized server workload.
- Reduce the time it takes to provision new servers by as much as 70 percent.
- Decrease downtime and improve reliability with business continuity and built-in disaster recovery.
- Deliver IT services on demand, independent of hardware, operating systems, applications or infrastructure providers.

Although consolidation substantially increases the efficient use of server resources but it may also increase the complex configuration of data, applications, resources and servers that can be confusing for a normal user to share with. To overcome this problem, we can use server virtualization approach or blade server.

Server Virtualization: Server virtualization is the masking of server resources, including various physical servers, operating system, processors. The administrator makes use of software to split one physical server into several servers to create a virtual environment.

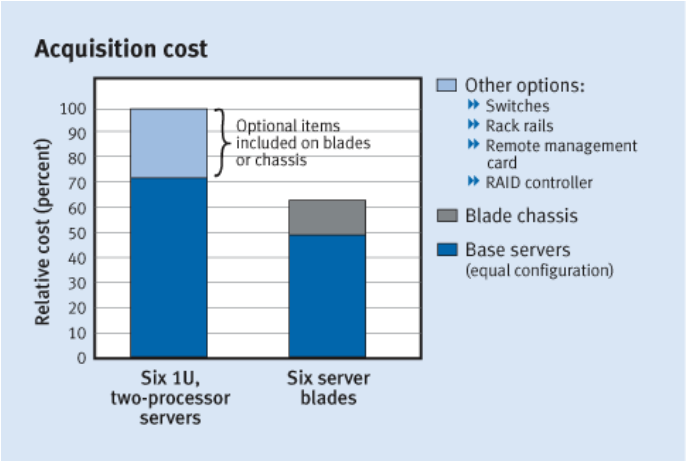
Server virtualization can be viewed as a part of an overall virtualization that includes storage virtualization, network virtualization, and memory virtualization and file management system.

Blade Server: According to Mike Roberts senior product manager at Dell systems, server blades architecture have the potential to increase server density, improve manageability, lower power consumption, and also enhance deployment and serviceability, all result in lower total cost of ownership.



Server blade is basically a server on a card-a single motherboard that contains a full computer system, including processors, memory, network connections and associated electronics.

Server blades are relatively inexpensive because each blade does not have a separate chassis and infrastructure like a traditional server. By leveraging power, cooling, management hardware, and cables over multiple systems, the per-server cost can be dramatically reduced, as shown below



When compared this architecture to other traditional rack servers, a blade server can handle any task or workload from client to cloud like server virtualization, Big data Applications, Web Page serving and caching, and also supports almost all of the operating systems available today.

In simple words, blades can be whatever we need them to be.

NUMA: Remedy to Memory sharing – problems

In this part we present some memory sharing problems such as replication of data and cache coherence.

Memory sharing problems:

1. Memory replication during migration:

The memory is subdivided into pages and pages into locations. The pages are fundamental unit of memory management and the locations are the fundamental unit of memory access. If we consider the virtual memory mechanisms, we can distinguish between virtual pages and physical pages. Virtual pages are in the memory location of some program or the operating system, whereas the physical pages are the actual memory in the clusters. Sharing may result more then one virtual page into 1 or more address spaces being mapped to the same physical page.

We concentrate on two major tools for the management of memory; replication and migration of virtual memory. Replication consists of making a copy of virtual page in another cluster and updating mappings that benefit from that copy. Migration consists of moving a virtual page from one cluster to another and updating all mappings to that page.

The online replication problem consists of determining when in a sequence of accesses of page should be replicated to other cluster without look ahead [10].

2. Cache Coherence:

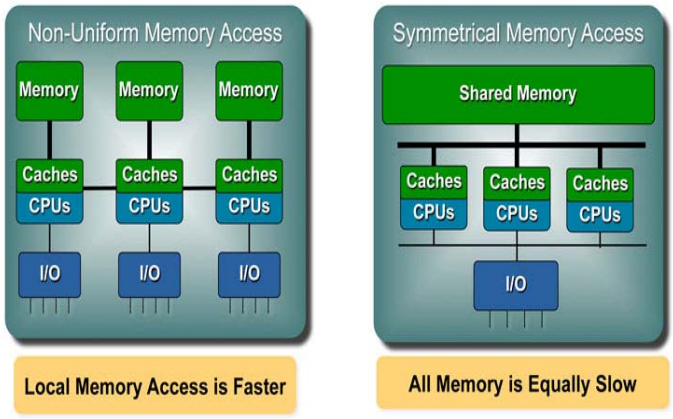
Multiprocessor systems with caches and shared memory space need to resolve the problem of keeping shared data coherent. This means that the most recently written data to a memory location from one processor needs to be visible to the other processors immediately. Cache Coherence means that all memory references from all processors will return the latest updated data from any cache in the system automatically.

Local and Remote Memory Sharing with NUMA architecture: A remedy

Given the limitations of bus-based multiprocessors, CC-NUMA is the scalable architecture of choice for shared-memory machines. This provides a facility for accessing local memory of the remote machine as well as local memory of the machine itself through some interconnection technique. This doesn't make use of copying. Since remote access involves more time this is recommended for limited data access keeping in view the down time factor.

CC-NUMA:

Non-Uniform Memory Access or Non-Uniform Memory Architecture (ccNUMA) is a computer memory design used in multiprocessors, where the memory access time depends on the memory location relative to a processor. Under ccNUMA, a processor can access its own local memory faster than non-local memory, that is, memory local to another processor or memory shared between processors. ccNUMA architectures logically follow in scaling from symmetric multiprocessing (SMP) architectures [12].



File system and remote file sharing through file system virtualization:

In all the situations of VM migration when there is a difference in OS for VM at source and destination hosts there is a need to handle differences in file systems belonging to the different OS, this problem is addressed here with file system virtualization technique.

Virtualization Infrastructure Requirements:

The key advantage of virtualization is greater utilization of physical server resources. For achieving this advantage, we can not down the services to the business at any cost.

To ensure that existing servers operate in a shared environment, detailed hardware inventory and performance utilization information must be obtained. At the completion of collection phase, the architect evaluates the results and provides documented recommendations on virtualization suitability across the server candidates [5].

File System differences in OS virtualization:

Process migration is the ability to transfer a process from one machine to another. It is a useful facility in distributed computing environments, especially as computing devices become more pervasive and Internet access becomes more ubiquitous. The potential benefits of process migration, among others, are fault resilience by migrating processes off of faulty hosts, data access locality by migrating processes closer to the data, better system response time by migrating processes closer to users, dynamic load balancing by migrating processes to less loaded hosts, and improved service availability and administration by migrating processes before host maintenance so that applications can continue to run with minimal downtime. [7]

Although process migration provides substantial potential benefits and many approaches have been considered [8], achieving process migration functionality has been difficult in practice. Toward this end, there are four important goals that need to be met [7]:

i) given the large number of widely used legacy applications, applications should be able to migrate and continue to operate correctly without

modification, without requiring that they be written using uncommon languages or toolkits, and without restricting their use of common operating system services.

ii) Migration should leverage the large existing installed base of commodity operating systems. It should not necessitate use of new operating systems or substantial modifications to existing ones.

iii) Migration should maintain the independence of independent machines.

iv) Migration should be fast and efficient. Overhead should be small for normal execution and migration.

To overcome limitations in previous approaches to general-purpose process migration we can provide a thin virtualization layer on top of the operating system that introduces a PrOcess Domain (pod) abstraction. A pod provides a group of processes with a private namespace that presents the process group with the same virtualized view of the system. This virtualized view associates virtual identifiers with operating system resources such as process identifiers and network addresses. This decouples processes in a pod from dependencies on the host operating system and from other processes in the system [7].

The main difference between a pod and a traditional operating system environment is that each pod has its own private, virtual namespace. The idea of a private, virtual namespace is surprisingly simple but has significant implications for supporting migratable computing environments. The namespace provides consistent, virtual resource names in place of host-dependent resource names such as PIDs.

To provide modular support for multiple file systems, many OSs provide a virtual file system framework that supports a form of interposition known as file system stacking [6]. File system virtualization is accomplished by creating a special directory per pod that serves as a staging area for the pod's private file system hierarchy. Storage requirements are minimized by sharing read-only portions of the file system among pods, if applicable, through loopback mounting or networked file systems.

CONCLUSION:

Our work addresses the crucial issue of down time reduction during VM migration with several architectural based techniques presented in the paper prominently, RDMA which offers services towards faster and direct data transfer between host and destination server as against many of the prevalent and past solutions for the same which incorporated traditional I/O techniques of polling and interrupt driven I/O. It's advantageous towards reducing down time can be enhanced through mentioned parallel bus system this will cater to present needs where data volumes are large and processing are response oriented. Server consolidation with enhanced benefits using blade server concept which is built with proximate shared resources in multiplicity offering space saving then distant multiple host system is a part of our proposal for live migration along with RDMA. Secondly, cc-NUMA is deployed as an architectural solution to two dominant problems - memory replication and cache coherence; this is a relatively new approach for VM migration systems, although it's an established technique. A very important aspect of OS virtualization in which a problem of heterogeneous OS platforms for the migrating VM, which probably cause problem at file system level is addressed here in our work with virtual file system based on pod. Talking all about issues of live migration the other circumstance of resource failure making VM migration unavoidable is alternatively addressed through redundancy.

REFERENCES:

[1] Changyeon Jo, Erik Gustafsson, Jeongseok Son, and Bernhard Egger, "Efficient Live Migration of Virtual Machines Using Shared Storage", VEE'13, March 16–17, 2013, Houston, Texas, USA.

[2] H. Jin, L. Deng, S. Wu, X. Shi, and X. Pan, "Live virtual machine migration with adaptive, memory compression," inCluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on. Ieee, 2009, pp. 1–10.

[3] High Performance Virtual Machine Migration based on RDMA a case study by Wan Huang 89-439-6702

[4] Information Storage & Management, Storing, Managing and protecting digital information by G. Somasundaram and Alok Shrivastava, EMC education Services

[5] Cloud Computing by Dr. Kumar Saurabh, Wiley Publication.

[6] E. Zadok. FiST: A System for Stackable File System Code Generation. PhD thesis, Computer Science Department, Columbia University, May 2001.

[7] The Design and Implementation of Zap: A System for Migrating Computing Environments

[8] D. Milojicic, F. Douglis, and R. Wheeler, Mobility: Processes, Computers, and Agents, Addison Wesley Longman, February 1999

[9] See more at: http://www.vmware.com/in/consolidation/overview.html#sthash.egq8ryoZ.dpuf

[10] http://books.google.co.in/books?id=YgXP-sumHu4C&pg=PA74&lpg=PA74&dq=memory+replication+during+migration&source=bl&ots=hmYEzJEEiy&sig=3X6SMSJHwksCe76GmZEBgNTkfCk&hl=en&sa=X&ei=x02RU9WdKILc8AX0lYDICw&ved=0CFsQ6AEwBw#v=onepage&q=memory%20replication%20during%20migration&f=false on page no. 75

[11] http://searchdatacenter.techtarget.com/definition/server-consolidation

[12] http://www.numascale.com/ccnuma.html