A Replication Based Data Management Approach for Data Intensive Cloud Systems

MEER AZMATH ALI[1], MD. ATEEQ UR RAHMAN[2]
[1]PG Scholar, Dept of CSE, SCET, Hyderabad, TS, India,
[2]Professor & HOD, Dept of CSE, SCET, Hyderabad, TS, India,

Abstract: Virtual machines migration to the destination server from the current host is required either because of hardware issues or high load causing the slow execution. Many solutions proposed to overcome the issues. Live Migration technique helps to overcome the slow execution speed. In this solution the Virtual machine performs the task on current host and the needed data moved to the destination. Precopy and Postcopy data transfer technique is implemented to run the Virtual Machine on other host. Data is send on demand as requested to avoid load issues. Various techniques are used to transfer data in Virtual machine migration. RDMA- Remote Direct Memory Access method for data transfer provides high speed data transfer and better transmission than I/O technique. Another technique NUMA (Non –Uniform memory access) technique which enable to have different memory access control which helps in increasing server utilization. Blade Server approach is implemented for better visibility on the resources and servers arise due to NUMA technique.

Index Terms: NUMA (Non –Uniform Memory Access), RDMA- Remote Direct Memory Access, CPU, VM.

I. INTRODUCTION

There are two kinds of virtualizations hardware virtualization and OS virtualization. In hardware virtualization, various virtual servers called as virtual machines are created; they play out the application execution on their virtual servers using hypervisor software to make this happen on real hardware. Increasing the quantity of virtual machines on a host server leads to better utilization of server assets yet at some point may cause overload and gradualness of execution. In such situations any particular virtual machine distinguished as moderate running one needs live migration to another host [1]. Virtual machine migration may also be required if any critical hardware segment is diagnosed as faulty. In this case, migration through closing down the virtual machine on the present server and restarting it on the destination have takes place. Live migration has a need to transfer data fasterly because the migration is gradual, for at some point the VM keeps running on the present host and data is transferred to destination have where it is moved eventually. There are two strategies to do live migration pre-duplicate where data having a place with the VM is transferred to the destination all in all and then VM is started on destination. In the event that data is large the migration is moderate. In postcopy 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 required in either case more in pre-duplicate an altered strategy of I/O transfer which is RDMA (remote direct memory access) is incorporated into this paper, which resembles DMA (coordinate memory access) however it is applicable when communication has to take place between two separate PCs, one providing the data and another getting it. DMA or RDMA are considerably faster than the conventional I/O strategies of 'Surveying' and 'Hinder driven I/O' as they don't include CPU [2]. Utilization of parallel transports as numerous transports instead of a solitary transport as regularly utilized as a part of RDMA is also included. Another method for better data communication is the NUMA (non – uniform memory access) here various PCs have diverse memory access times for accessing there local and different PCs memory which is named as remote access through some interconnection organize. Consolidation is to substantially increase the effective utilization of server assets which is advantageous however it may also increase the multifaceted nature of the configuration of data, applications, assets and servers that can be mistaking for a normal client to share with. To conquer this issue, we have made utilization of server virtualization approach or blade server.

In blade server, number of servers is placed in closeness with some basic assets. In case of OS virtualization, for actualizing virtual machines utilization of document framework virtualization to handle the issue of heterogeneity of record framework having a place with various visitor OS is incorporated into this paper. Rest of the paper is organized as:

- Reasons of VM migration
- Migration because of fault in hardware and cures.
- Live migration.
- RDMA and server consolidation as proposed for live migration
- Sharing of memory addressed with ccNUMA
- Record framework contrasts in source and destination have and their answer with document framework virtualization accomplished with unit (PrOcess Domain).

II. REASONS OF VM MIGRATION

VM migration is majorly caused because of two main reasons when the hardware segments conspicuously CPU goes faulty, furthermore when CPU is backing off because of load and not liable to satisfy the administration level agreement of reaction time or different parameters pertaining to limitations on preparing time or volume of handling. At the point when hardware is mistaken or faulty How to distinguish the faulty state of hardware; The indications of a faulty CPU could not be right esult, distinctive outcome for same data and guideline when registered repeatedly. Hardware segments ought to be subjected to diagnostic schedules on an intermittent basis of the request of one or few direction execution time and in case any fault found in a particular hardware segment, remedial advances ought to be taken through moving the work to another provisioned redundant hardware segment of the same nature. This could be illustrated with a situation in information storage management scenario with following Fig.1[4]:

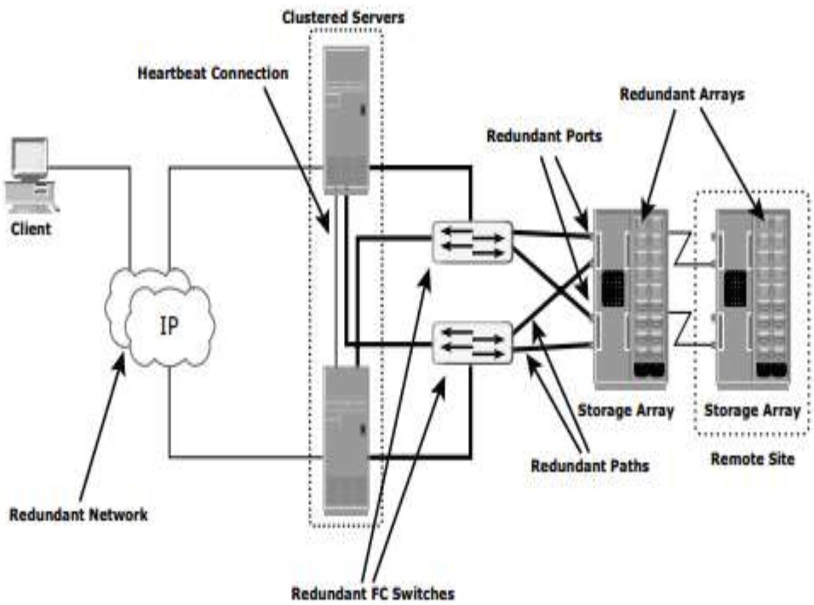

Fig.1. Proposed System Architecture.

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.

A. Live Migration

Live migration is prescribed when introduce hardware isn't probably going to serve the need of the virtual machine on which wanted application is running. This may be because of its utilization for different purposes. Each CPU and hardware asset has its capacity as far as assortment of employments or tasks it can handle offering advantages of different preparing. Instructions to recognize the utilization coefficient of CPU and other hardware assets; CPU can react slower is an indication of higher number of employments 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 processed value is greater than 1, its normal range is 0 to 1. What data should be transferred for virtual machine migration? Application forms status as process control square, enroll values, intermediate outcomes, Data records, question documents.

B. Memory Replication Amid Migration

In both of pre duplicate and post duplicate live migration procedures amid migration the data is available on the source as well as its duplicate on the destination. This is a duplication wonder which isn't desirable as it includes over necessity of memory. The pre duplicate and post duplicate are elaborated in the accompanying points. Factors adding to delay of data transfer: This is primarily because of adoption of moderate I/O strategies, for example, hinder driven I/O or Polling. Delay can also be caused because of volume of data and incorporation of single transport frameworks. Systems for chopping down this delay: When VM is migrated its down time is given as underneath:

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

As down time relies upon data transfer time there is an enormous need t chop it down. For this reason following methods can be utilized:

- Use of Direct Memory Access (DMA): In transferring data starting with one host then onto the next amid VM migration amongst surveying, hinder driven I/O and DMA, the last one is the best strategy. This surrenders the CPU from its transport controlling duty and data gets read or written in enormous squares specifically between I/O gadget and memory.
- Data pressure: This can be utilized for large volume of data transfer.
- Use of global memory: Not all data pertaining to the application running of VM is required on the other host immediately to keep itrunning on destination have. In the event that enormous data records are put away in global memory accessible to alternate hosts, they require not be transferred amid VM migration.

- Deploying fast system
- Data to be prepared in next couple of directions should just be considered for transferring. Live Migration with RDMA and Server consolidation:

C. Proposed Approach

Live migration means a running VM is moved starting with one physical host then onto the next. Live migration allows you to move a whole running virtual machine starting with one physical server then onto the next, with minimal downtime. To migrate a running VM across particular physical hosts, its entire details have to be transferred from the source to the target have. The state of VM incorporates the information about permanent storage, the volatile storage, associated gadgets, and the internal state of virtual CPU. The overall approach for live migration is pre-duplicate in which the substance of VM's memory are first transferred to the target host and then VM is restarted. To keep the downtime, the time amid which the VM isn't running, to a base, data is sent in several iterations (i.e., just the page that has been altered since the last round were sent) while the VM continues running on the source have. Another approach is post-duplicate, in which just the VM's VCPU and gadget state is sent to the target have and restart them immediately. Memory pages accessed by VM are then gotten in parallel and on-demand while VM is running on the target have. [1] Incase of large amount of data to be transferred utilization of parallel data transports is suggested. Here, sender will dispatch initial couple of pages on the principal transport and next couple of pages on parallel transport. At the less than desirable end data from the parallel transport will be cradled and memory will be allocated according to the paging plan of the sender. This kind of arrangement will speed up the data transfer in pre-duplicate and virtual machine can be migrated relatively sooner.

1. RDMA: Live migration gives 3 quality choices to diminish the time required to live migrate a virtual machine. It is possible that we can utilize memory pressure [2], or we can pick Remote Direct Memory Access (RDMA) [3], or we can pick multichannel arrange adapters. Above choices can bolster our private cloud infrastructure by:

- Increasing the effectiveness of live migration when our hardware assets are constrained (memory pressure).
- Increasing the scalability of live migration when our hardware assets are not constrained (multi-channel organize adapter and RDMA)

In conditions where hardware and systems administration assets are constrained, live migration conveys performance enhancements for migrating virtual machines by compacting the memory data before sending it across the system. This uses spare CPU capacity available in the server running Hyper-V. Hyper-V nearly screens the CPU prerequisites of the virtual machine and just devours an appropriate amount of CPU assets to rapidly move virtual machines from one server to the next[2]. In situations where organizing assets are not constrained, you can design live migration to utilize multichannel arrange adapters or RDMA-enabled organize adapters, which lessens the time required to live migrate virtual machines. RDMA can play out an immediate memory access from the memory of one PC into that of another without including the operating framework. This licenses high-throughput, low-latency systems administration and conveys greater productivity with live migration [3] Server Consolidation: Server consolidation is an approach to the productive usage of PC server assets so as to diminish the total number of servers or server locations that an organization requires. According to Tony Iams, Senior Analyst at D.H. Dark colored Associates Inc. in Port Chester, NY, servers in many companies typically keep running at 15-20% of their capacity, which may not be a sustainable ratio in the current monetary condition. Organizations are increasingly swinging to server consolidation as one means of cutting unnecessary expenses and maximizing rate of return (ROI) in the data focus [11]. Server consolidation lets your organization [9]:

- Reduce hardware and operating expenses by as much as 50 percent and vitality costs by as much as 80 percent, saving more than $3,000 every year for each virtualized server workload.
- Reduce the time it takes to arrangement new servers by as much as 70 percent.
- Decrease downtime and enhance reliability with business coherence and implicit disaster recuperation.
- Deliver IT benefits on demand, autonomous of hardware, operating frameworks, applications or infrastructure suppliers.

Although consolidation substantially increases the proficient utilization of server assets however it may also increase the perplexing configuration of data, applications, assets and servers that can be mistaking for a normal client to share with. To conquer this issue, we can utilize server virtualization approach or blade server. Server Virtualization: Server virtualization is the masking of server assets, including various physical servers, operating framework, processors. The administrator makes utilization of software to part one physical server into several servers to create a virtual domain. Server virtualization can be seen as a part of an overall virtualization that incorporates storage virtualization, arrange virtualization, and memory virtualization and document management framework. Blade Server: According to Mike Roberts senior item manager at Dell frameworks, server blades architecture have the potential to increase server thickness, make strides manageability, bring down power utilization, and also enhance sending and serviceability, all outcome in bring down total cost of possession. Server blade is basically a server on a card-a solitary motherboard that contains a full PC framework, including processors, memory, organize associations and associated hardware. Server blades are relatively economical because each blade does not have a separate chassis and infrastructure like a traditional server. By leveraging power, cooling, management hardware, and cables over various frameworks, the per-server cost can be dramatically diminished, as appeared underneath When compared this architecture to other traditional rack servers, a blade server can handle any task or workload from customer to cloud like server virtualization, Big data Applications, Web Page serving and caching, and also bolsters almost all of the operating frameworks available today. In basic words, blades can be whatever we require them to be.

2. NUMA: Remedy to Memory sharing – issues

In this part we show some memory sharing issues, for example, replication of data and cache rationality. Memory sharing issues:

Memory Replication Amid 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. In the event that we think about the virtual memory mechanisms, we can recognize virtual pages and physical pages. Virtual pages are in the memory location of some program or the operating framework, whereas the physical pages are the actual memory in the bunches. Sharing may come about more then one virtual page into at least 1 address spaces being mapped to the same physical page. We concentrate on two major devices for the management of memory; replication and migration of virtual memory. Replication comprises of making a duplicate of virtual page in another bunch and updating mappings that advantage from that duplicate. Migration comprises of moving a virtual page starting with one bunch then onto the next and updating all mappings to that page. The online replication issue comprises of deciding when in a succession of accesses of page ought to be replicated to other bunch without look ahead [10].

Cache Coherence: Multiprocessor frameworks with caches and shared memory space need to determine the issue of keeping shared data lucid. This means that the most as of late composed data to a memory location from one processor should be unmistakable to alternate processors immediately. Cache Coherence means that all memory references from all processors will restore the latest updated data from any cache in the framework automatically. Local and Remote Memory Sharing with NUMA architecture: A cure Given the limitations of transport based multiprocessors, CCNUMA is the scalable architecture of decision for shared memory machines. This gives a facility to accessing local memory of the remote machine as well as local memory of the machine itself through some interconnection strategy. This doesn't make utilization of duplicating. Since remote access includes additional time this is prescribed for constrained data access keeping in see the down time factor.

3. CC-NUMA: Non-Uniform Memory Access or Non-Uniform Memory Architecture (ccNUMA) is a PC memory configuration utilized as a part of multiprocessors, where the memory access time relies upon the memory location relative to a processor. Under ccNUMA, a processor can access its own particular 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]. Document framework and remote record sharing through document framework.

Virtualization: In all the situations of VM migration when there is a distinction in OS for VM at source and destination has there is a need to handle contrasts in document frameworks having a place with the diverse OS, this issue is addressed here with record framework virtualization procedure.

Virtualization Infrastructure Requirements: The key advantage of virtualization is greater utilization of physical server assets. For achieving this advantage, we can not down the administrations to the business at any cost. To guarantee that current servers operate in a shared domain, detailed hardware stock and performance utilization information must be obtained. At the consummation of gathering phase, the architect evaluates the outcomes and gives archived recommendations on virtualization suitability across the server candidates [5].

Record System Contrasts In OS Virtualization: Process migration is the ability to transfer a procedure starting with one machine then onto the next. It is a helpful facility in disseminated figuring conditions, especially as processing gadgets turn out to be more pervasive and Internet access turns out to be more omnipresent. The potential advantages of process migration, among others, are fault flexibility by migrating forms off of faulty hosts, data access locality by migrating forms nearer to the data, better framework reaction time by migrating forms nearer to clients, dynamic load balancing by migrating procedures to less loaded has, and enhanced administration availability and administration by migrating forms before have maintenance so applications can keep on running with minimal downtime. [7] Although process migration gives substantial potential advantages and many approaches have been thought about [8], achieving process migration functionality has been troublesome in practice. Toward this end, there are four important goals that should be met [7]:

- Given the large number of generally utilized legacy applications, applications ought to have the capacity to migrate and keep on operating accurately without modification, without requiring that they be composed utilizing exceptional languages or tool boxs, and without limiting their utilization of normal operating framework administrations.
- Migration should leverage the large existing installed base of product operating frameworks. It ought not necessitate utilization of new operating frameworks or substantial modifications to existing ones.
- Migration ought to maintain the freedom of autonomous machines.
- Migration ought to be fast and proficient. Overhead ought to be small for normal execution and migration.

To conquer limitations in past approaches to general purpose process migration we can give a thin virtualization layer over the operating framework that presents a PrOcess Domain (unit) abstraction. A case furnishes a gathering of procedures with a private namespace that gives the procedure aggregate the same virtualized perspective of the framework. This virtualized see associates virtual identifiers with operating framework assets, for example, process identifiers and system addresses. This decouples forms in a case from conditions on the host operating framework and from different procedures in the framework [7]. The main distinction between a case and a traditional operating framework condition is that each unit has its own private, virtual namespace. The idea of a private, virtual namespace is shockingly straightforward however has significant implications for supporting migratable registering situations. The namespace gives reliable, virtual asset names in place of host-subordinate asset names, for example, PIDs. To give modular help to numerous document frameworks, many OSs give a virtual record framework that backings a type of mediation known as document framework stacking [6]. Record framework virtualization is accomplished by creating a special catalog for each case that fills in as a staging area for the case's private document framework hierarchy. Storage necessities are limited by sharing read-just bits of the document framework among units, if applicable, through loopback mounting or organized record frameworks.

III. MODULE EXPLANATION

A. Client Authentication

On the off chance that you are the new Client going to access their page then they have to enlist first by giving necessary details. After fruitful finishing of join process, the customer has to login into the application by giving username and exact password. The customer has to give exact username and password which was given at the season of registration.

B. Data Request

The Client then login to the application and give raw data which is additionally transferred into binary stream and passed to the access control administrator for advance technique.

C. Access Control Administrator

The administrator confirms the got data from each customer login and Converts each data into graph see data after graph see the administrator analysis and separates each data utilizing k-anonymous Bi-target Graph Partitioning procedure. Where each graph data`s are separated from other customer graph data.

D. Graph Data Access

The administrator confirms the got data from each customer login and Converts each data into graph see data after graph see the administrator analysis and separates each data utilizing k-anonymous Bi-target Graph Partitioning procedure. Where each graph data`s are separated from other customer graph data.

E. Heuristic Analysis

The confirmed graph data are further send to social interaction site where Enabled data are displayed for advance heuristic analysis of graph data.

IV. SCREEN SHORTS

Results of this paper is as shown in bellow Figs. 2 to 10.

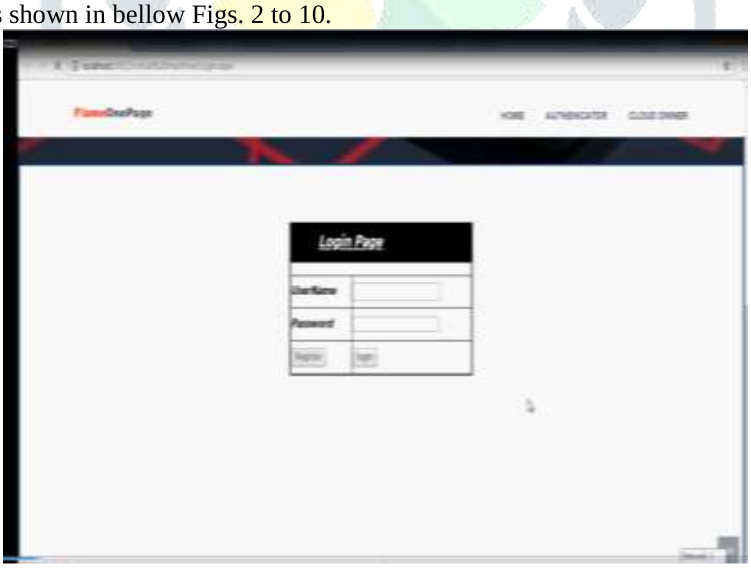

Fig.2. login page.

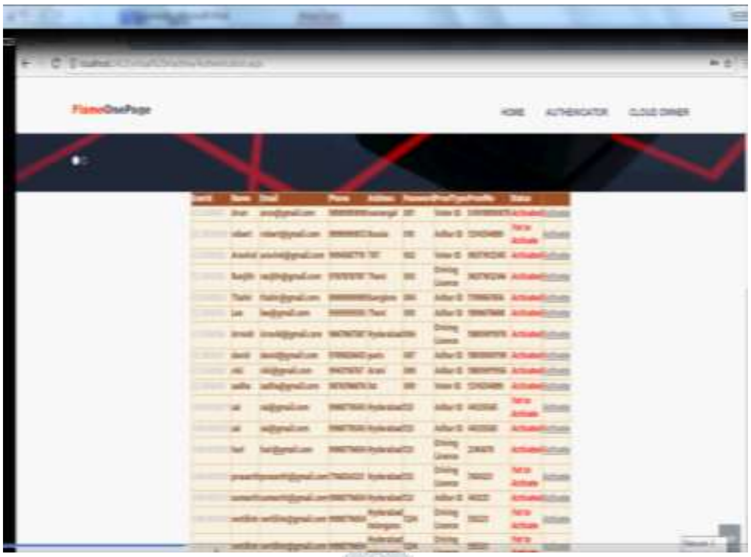

Fig.3. Authendicator.

Fig.4. alert message.

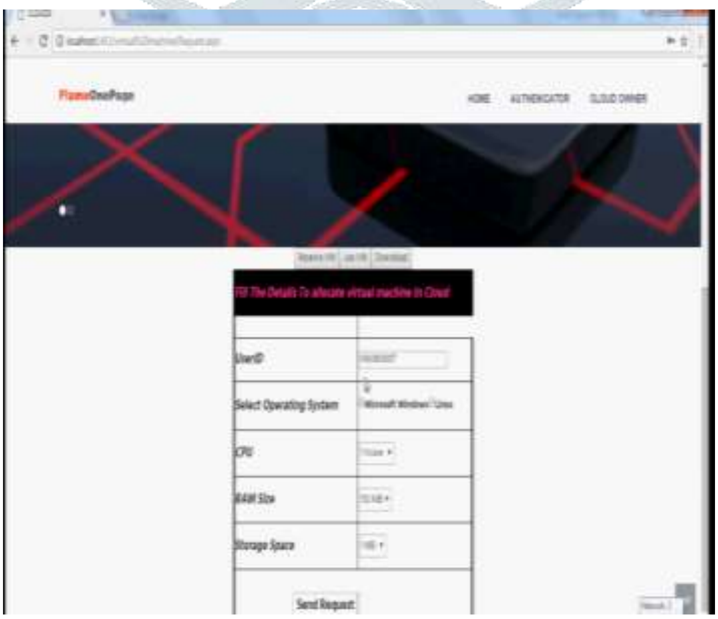

Fig.5. resource virtual machine.

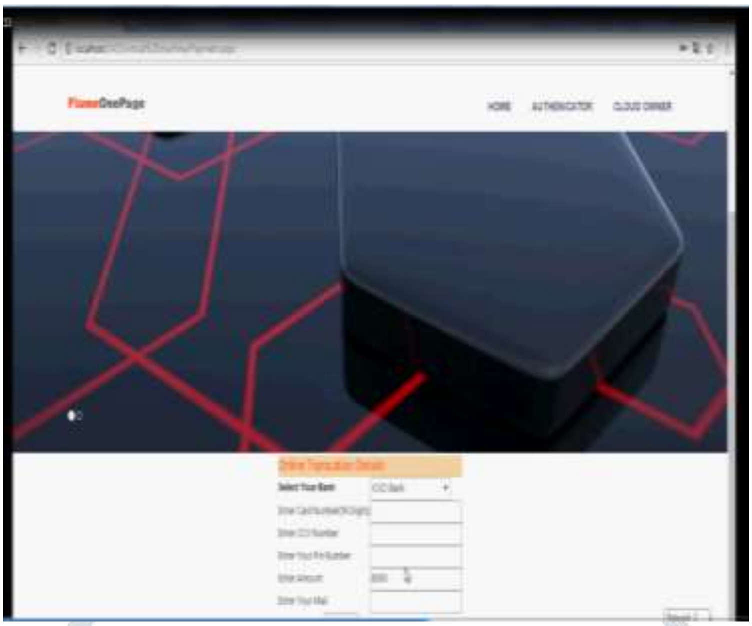

Fig.6. online payments.

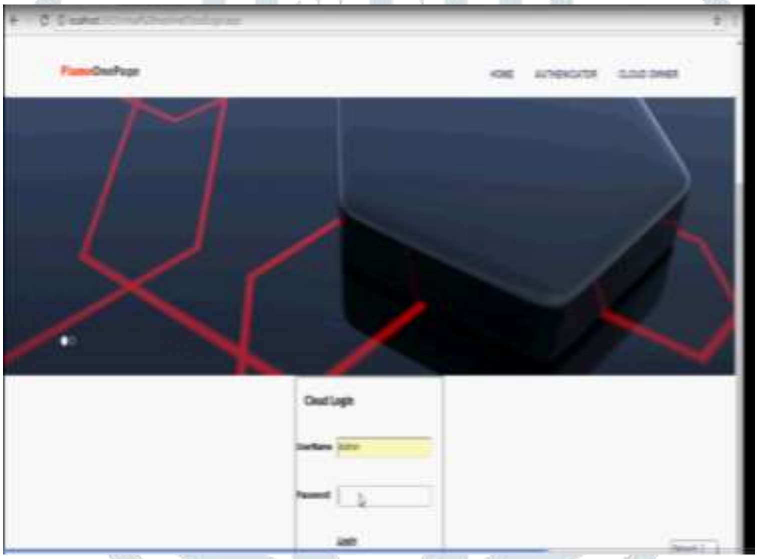

Fig.7. cloud login.

V. CONCLUSION

This work provides the reduction time of migration of VM with the help of techniques like RDMA that helps in faster data transfer from host to destination server. This solution helps to solve problems that were caused by traditional techniques like polling I/O and driven I/O. Parallel bus system will further help in increasing the performance by minimize the execution down time. Blade server technique with appropriate shared resources is also discussed in the live migration as part of proposed project. Architectural designed approach cc-NUMA helps in overcoming the problems like memory replication and cache coherence. Live migration of Heterogeneous Operating system platforms cause issues at disk level is addressed in this paper by replacing with file system based on pod.

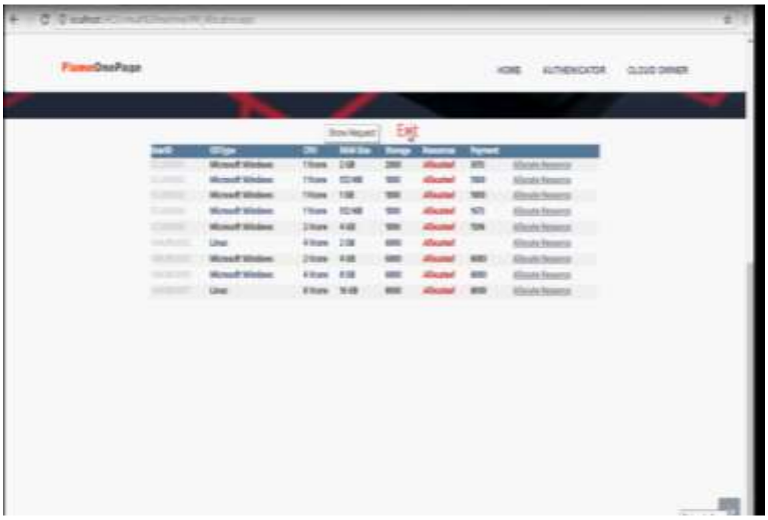

Fig.8. cloud owner allocation.

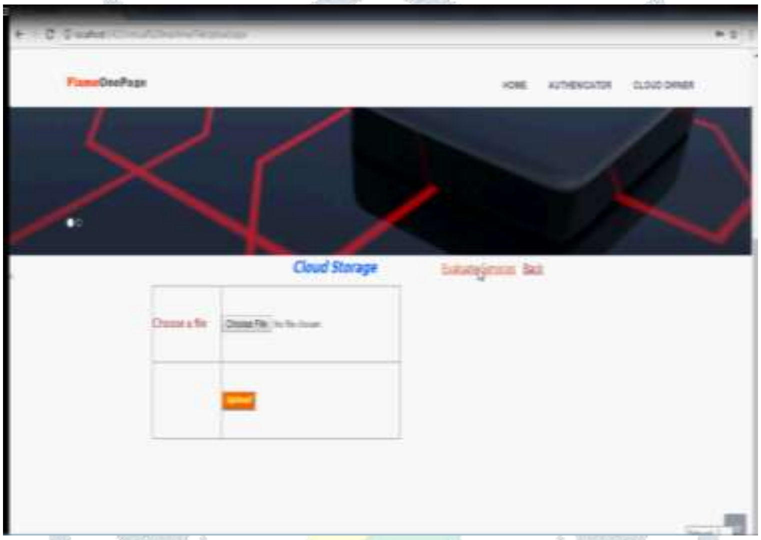

Fig.9. evaluate virtual machine.

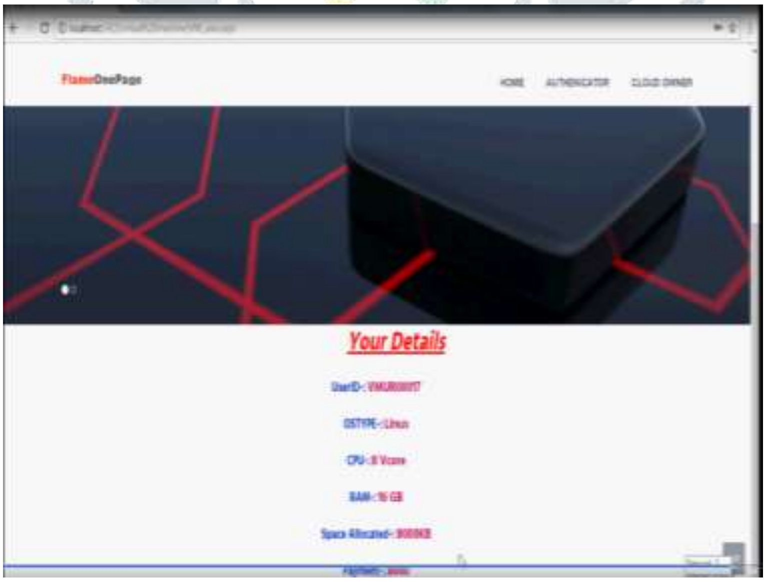

Fig.10. vm user details.

VI. REFERENCES

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

[2] H. Jin, L. Deng, S. Wu, X. Shi, and X. Pan, "Live virtual machinemigration with adaptive, memory compression," inClusterComputing and Workshops, 2009. CLUSTER'09.IEEEInternational Conference on.Ieee, 2009, pp. 1–10.

[3] High Performance Virtual Machine Migration based on RDMA acase study by Wan Huang 89-439-6702.
[4] Information Storage & Management, Storing, Managing andprotecting digital information by G. Somasundaram and AlokShrivastava, EMC education Services.
[5] Cloud Computing by Dr. Kumar Saurabh, Wiley Publication.
[6] E. Zadok. FiST: A System for Stackable File System CodeGeneration. PhD thesis, Computer Science Department, ColumbiaUniversity, May 2001.
[7] The Design and Implementation of Zap: A System for MigratingComputing Environments.
[8] D. Milojicic, F. Douglis, and R. Wheeler, Mobility: Processes,Computers, and Agents, Addison Wesley Longman, February 1999.

Author’s Details:

Mr. MEER AZMATH ALI has completed his Bachelor of Engineering from MuffakhamJah College of Engineering and technology, Hyderabad, OSMANIA University. Presently, he is pursuing his Masters in Computer Science Engineering from Shadan college of Engineering and technology, Hyderabad, TS. India.

MR MD ATEEQ UR RAHMAN received his B.E Degree from P.D.A College of Engineering, Gulbarga, Karnataka, India in 2000. In 2004, He obtained M.Tech degree in Computer Science & Engineering from Visvesvaraya Technological University, Hyderbad, India. He is currently pursuing Ph.D from Jawaharlal Nehru Technological University, Hyderabad, India. Presently he is working as Associate Professor in Computer Science & Engineering Dept, S.C.E.T Hyderabad. His areas of interest include Spatial Databases, Spatial Data Mining, Remote Sensing, Image Processing and Networks protocols etc.