CPU Utilization based Enhancing Resource Utilization And Minimizing Down Time During VM Migration

Jimmy D. Patel[1], Asst. Prof. Slesha Nayak[2]
[1]Student, Silver Oak College of Engineering & Technology, S.G.Highway, Gota Ahmedabad, India.
[2]Assistant Professor,Silver Oak College of Engineering & Technology, S.G.Highway, Gota Ahmedabad, India.

Abstract-*The Cloud Computing is enabling innovative and on demand services by allowing pay per use, location independency and device independency. In process of migration VM moves one physical machine to another. In live migration, the VMs are migrated without stopping their working. In offline migration, process is stopped till the VM can continue on target machine. In this we first present a live migration performance strategy. Live Task of migration and the needed properties of VM for monitoring the resources and optimal the fitness function evaluation. We Will reduce the operation cost, down time and also increases the resource utilization than migrate VM one server to another server base checking of prediction capacity. To solve the problem of the overload of virtual machine, virtual machine migration techniques used which maintain the load balance on the Physical Machine which is undergo unnecessary problems caused during the time of overload and also optimize the resource utilization and total down time.*

Keywords-Cloud Computing, CPU utilization, Fitness Function, virtual machine migration.

I. INTRODUCTION

cloud computing is a model for drawing by and large, on-request access to shared pool of configurable get ready assets (e.g., PC structures, servers, stockpiling, applications and organizations) that can be instantly provisioned and discharged with unimportant association exertion or ace focus affiliation. cloud computing will be registering administrations given over the web, whereby shared assets, programming and data are given to PCs and different gadgets on request. Cloud computing is the idea actualized to translate the day by day figuring issues, preferences of equipment programming and asset accessibility unhurried by computer users. A pool of abstracted, highly scalable, and managed compute infrastructure capable of hosting end customer applications and billed by consumption [2].

II. THEORATICAL BACKGROUND

Cloud computing is a popular trend in current computing, which people can easy access to computational resources, what's more, its very cheap.

There are administrations and models working behind the scene making the cloud computing rational and open to end clients. Taking after there are two models for cloud computing:

- Deployment Models
- Service Models

Deployment models describe the sort of access to the cloud, i.e., how the cloud is found? Cloud can have any of the four sorts: Public, Private, Hybrid, and Community.

A Public cloud is known as an "open cloud" when the organizations are rendered over a framework that is open for open use. Open cloud associations might be free.

Private cloud can't abstain from being cloud structure worked just for a solitary alliance, paying little regard to whether coordinated inside or by an outsider, and enabled either inside or remotely.

Hybrid cloud is a structure of at minimum two fogs (private, gathering or open) that stay unmistakable substances yet are bound together, offering the benefits of different course of action models.

In Community Cloud A social order cloud in taking care of is a total exertion in which foundation is shared between two or three relationship from a particular get-together with fundamental concerns (security, consistence, territory, and whatnot.), paying little regard to whether coordinated inside or by an outsider and supported inside or remotely.

Cloud computing is based on service models. These are categorized into three basic service models which are Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), Infrastructure-as–a-Service (IaaS)

In SAAS, The limit given to the purchaser is to utilize the supplier's applications running on a cloud framework. The applications are open from different client contraptions through either a thin customer interface, for example, a web program (e.g., online email), or a program interface.

In PAAS, The limit given to the customer is to send onto the cloud establishment client made or acquired applications made using programming dialects, libraries, administrations, and mechanical assemblies supported by the provider.

In IAAS, The limit given to the buyer is to preparing, stockpiling, organize, and other important enlisting resources where the client can pass on and run subjective programming, which can fuse working systems and applications.

Virtualization In Cloud Computing

Virtualization is the "creation of a virtual (rather than actual) version of something, such as a server, a desktop, a storage device, an operating system or network resources".

Virtualization is a structure, which licenses to share a solitary physical example of preference or an application among different clients and affiliations. It does by distributing a sensible name to a physical stockpiling and giving a pointer to that physical asset when asked [7].

III. VIRTUAL MACHINE MIGRATION

VMs allows to one example of a working framework alongside at least one applications running in a disconnected segment inside the system. There will be numerous VMs running on top of a solitary PM. When one physical host gets over-load, it might be required to proficiently exchange certain measure of its heap to another machine with insignificant impedance to the clients. This system of moving a VM from one physical host to an other host is named as development. As of now, to move a VM between two physical hosts, it was basic to shutdown the VM, dole out obliged points of interest for the new physical host, move the VM files and begin the VM in the new host.

VM is a product or working framework, which work like a different framework. The relocation is named as the way toward moving a VM beginning with one physical machine (PM) then onto the following physical machine (PM). When one physical machine gets over-burden, it might be required to move the information in to each other machine[1].

The performance of the VM migration is calculated by two metrics

- Total migration time: the time from the beginning of pre-migration work to the end of all migration work[1].
- Down time: the time during the VM service is unavailable.[1]

Types Of VM Migration

1.Cold Or Non-live Migration

In this type of migration, the VMs are migrated when they are not working.

2. Hot Or Live Migration:

In this type of migration, the VMs are migrated without stopping their working.

Live VM Migration:

Live migration migrate running VM from one host to another host. It reduce the downtime compare of the cold VM migration. Live migration is an amazingly capable instrument for bunch and cloud overseer. A director can move OS occasions with application so that the machine can be liberated for upkeep. Additionally, to enhance reasonability, OS occurrences might be adjusted crosswise over machines to ease the heap on over-burden hosts to play out the live migration of a VM, its runtime state must be exchanged from the source to the goal while VM as yet continue.

Live Migration is an advancement used for load adjusting and enhancement of VM improvement in server farms. With the help of live relocation, VMs can be traded to another center point without shutting down.

Live Migration is organized two phases – (i) Control is changed to the objective. (ii) Data Transferring (memory/plate) to the objective. Pre-copy In this, first Memory is traded and after this execution is traded. The pre-copy technique is used to trade the memory to objective center over different accentuations. Post-copy In this, First execution is traded and after this, memory is traded. Not in any way like pre-duplicate, in post duplicate the Virtual CPU and contraptions on objective center point is move in starting stride and starts the execution in second step.

Techniques Of VM Migration

1.Pre copy migration

This approach pages of memory are iteratively copied from the source machine to the objective have, all while constant the execution of the VM being moved [1].

Warm-up phase

In pre-copy memory migration, A Hypervisor commonly duplicates all are memory pages from source to objective while the VM is in the not to distant past running on the source. In the event that some memory pages change (progress toward getting the opportunity to be 'soiled') amidst this strategy, they will be re-duplicated until the rate of re-replicated pages is in any occasion page dirtying rate.

Stop-and-copy phase

After the warm-up stage, VM will be stopped on the central have, that staying dirty pages will be imitated to the goal, and the VM will be continued on the goal have. The time between ending the VM on the essential server and continuing with it on target is called "down-time", and ranges from a few milliseconds to seconds as indicated by the cross of memory and applications continuous on the VM. There are a couple of methodology to diminish live migration down-time.

2 . Post copy migration:

In post-copy approach every memory page is exchanged just once, which is the fundamental advantage over pre-copy approach. Post-copy VM development is started by suspending the VM at the source. With the VM suspended, an irrelevant subset of the execution condition of the VM (CPU state, registers and, on the other hand, non-pageable memory) is exchanged to the objective. The VM is then running at the objective.

IV. LITERATURE REVIEW

VM is a product or working framework, which work like a different framework. Live (VM) relocation empowers consistent development of an online server starting with one area then onto the next to accomplish failure recovery, load balancing, and system maintenance. Past single VM migration, a multi-level application includes a gathering of corresponded VMs and its live migration will require cautious planning of the relocations of the part VMs.

(1)“Efficient VM Migration in Cloud Computing”

In this paper, total migration time and total down time both are key parameters of live migration process. They consider proposed approach of modification of already existing optimized pre-copy approach will work better for high dirty page rate and low dirty page rate environment and we will further decrease page rate by compressing data by compression algorithm CBC (characteristic Based Compression) and gives efficient migration[1].

(2)“VM Migration Strategy in Cloud Computing”

In this paper, proposed framework architecture is equipped with two main components, Central controller and Local controller. The central controller deployed on the controller physical node along with the local controller. There is only one central controller for cluster. On the other hand data collector and distributer responsible for fetching data from the central database and distributes it to the relevant component in the framework and also saving central controller data on the central database[2].

(3)“POLVM: Parallel Optimized Live VM Migration”

In this paper, the parallel optimization problem for Live VM migration in cloud. They introduce a Adaptive Genetic Algorithm to characterize the service process in which VM should migrate to where in cloud and achieve load balance. The proposed system overloaded nodes has to be optimized and the resources in under loaded node can be redistributed amongst the other resource-starving nodes. After performing proper migration, the underutilized node can be turned off and this will in turn contribute towards more energy conservation. It allows business customers to scale up and down their resource usage based on needs. Different organization provides same service with different service charges and waiting time[3].

(4)“vHaul: Towards Optimal Scheduling of Live Multi-VM Migration for Multi Tier application”

In this paper, demonstrate that different migration strategies result in distinct performance impacts on a multi-tier application in dedicated data Centers. Using controlled experiments and queuing theory, we show the interdependence between different tiers of a multi-tier application causes this problem. Then we present a system, vHaul, which computes the optimal multi-VM migration scheme and improves the performance of multi-tier application during migration[4].

(5)“A novel agent based match making algorithm in multitenant architecture”

In this paper considers load balancing (task placement) in optimal way. For the efficient task placement, They are Calculating the fitness function with more number of parameters such as memory availability, Central processing

unit availability, i/o read availability, i/o write availability and bandwidth availability. They are using virtual machine live migration for load balancing which reduces the downtime. They considering network proximity between datacenters which reduces the response time. In this paper the author, study overloaded VM was Migrated to normal loaded VM by consist of different type of agents, which considers network proximity, live task migration and the needed properties of VM for optimal resource utilization. This algorithm reduces the system downtime, migration time by implementing live task migration and reduces the response time by considering the network proximity.

V. PROPOSED METHODOLOGY

Let us define the proposed work which make a migration process of VM in an evaluate fitness function for the resource utilization. Cloud Computing is most popular in the world. Now big enterprises and companies are migrating their work load on cloud. Workload is distributed on different location of data Center around the world. Sometimes it may happen that some data Center's VMs are in overload condition and some are in under loaded condition. So we are using migration strategy to balance load of VMs. Overloaded data Center will migrate whole work load on another data Center on predication basis. So whole machine will be migrate on another data Center. So it will minimize downtime.

Steps of Proposed Work

Step 1: Evaluation process for all data center should be done for physical machine which decide inside.
Step 2: For that fitness function calculate.
Step 3: The values at the fitness function at collectively collected as a data set.
Step 4: This data set is use for further preprocess to generate model.
Step 5: Predication model is applied to take precise decision.
Step 6: Based on the decisions migration call is taken care off.

Proposed architecture

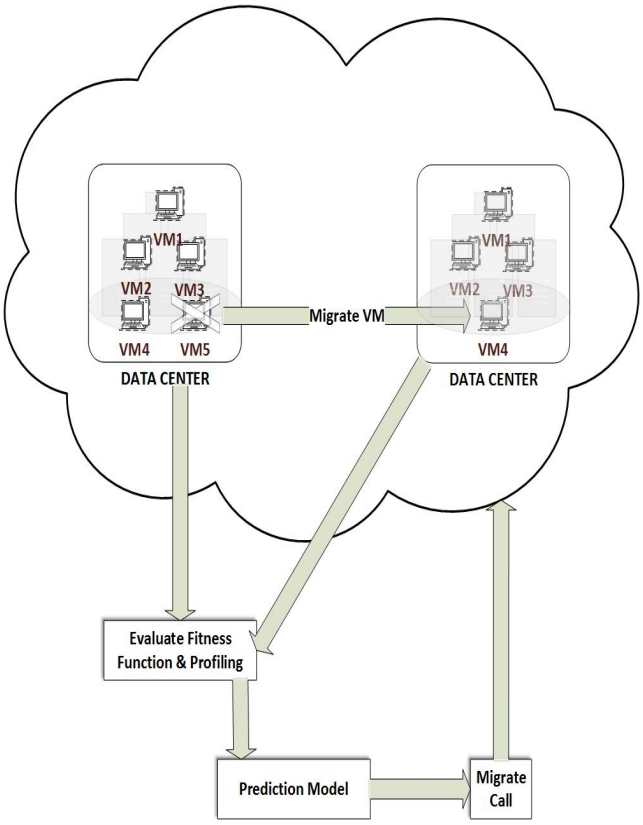

Figure 1:proposed architecture

Flowchart of proposed architecture

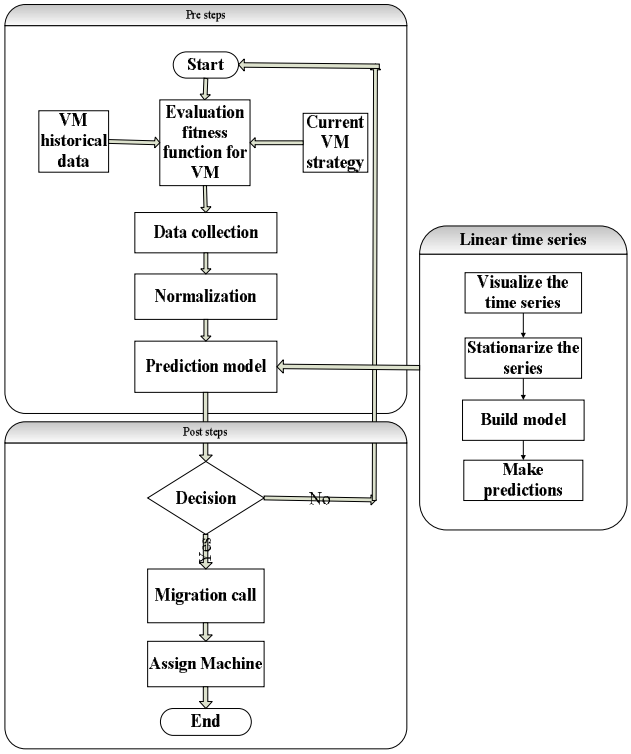

Figure 2: Proposed flowchart

VI. PROPOSED IMPLEMENTATION METHODOLOGY

Software Specification

Technology to be used: Amazon Web Service

AWS is a toolbox (library) for reproduction of Cloud registering situations. It gives fundamental classes to depicting server farms, VMs, applications, clients, computational assets, and arrangements for administration of various parts of the framework (e.g., planning and provisioning). Enables engineers to utilize JAVA programming dialect.

System Requirements

AWS can work on any 32 bit or 64 bit x86 architecture. It can work on any win‘ 10, win‘ 8, win‘ 7, win‘ service pack 2012 having 1 GB of minimum RAM, 1GB of free disk space,. No specific graphics card is required. It has been tested and ran on Sun's Java version 1.8.0 or newer with Eclipse. Older versions of Java are not compatible. If you have non-Sun Java version, such JDK, they may not be compatible. Apache tomcat 8.5.6 is for SQL Server.

Implementation Flow

Following steps describe the implementation flow for our work.

1. Create an Amazon AWS Account.
2. Installation of eclipse and set the SDK configuration in eclipse.
3. Use the EC2 API from AWS account.
4. API based infrastructure operations.
5. Implement the algorithm with java SDK of EC2 API.
6. Data collection
7. Fetch the parameter which is use in algorithm
8. Generate the scenario
9. Results

4.4 Implementation Result

Step 1: create instance in AWS

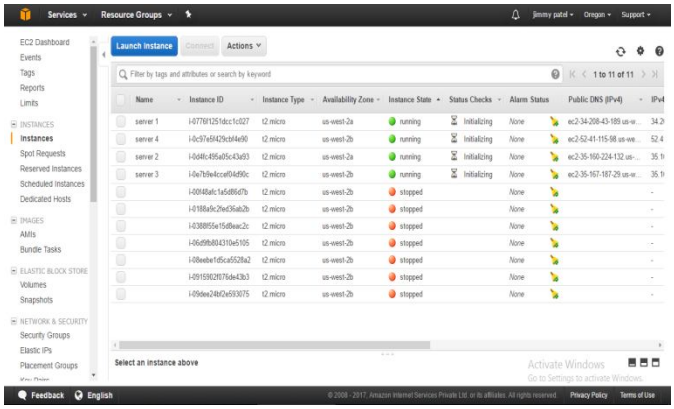

Figure 3: Instances in AWS

In this figure or step create the instances in AWS server. In Which instances created. four Instances are running state and another is stopped state. So these running state instances can be work as all mode of them.

Step 2: Algorithm start, fitness function calculated, normalize, and migrate or not

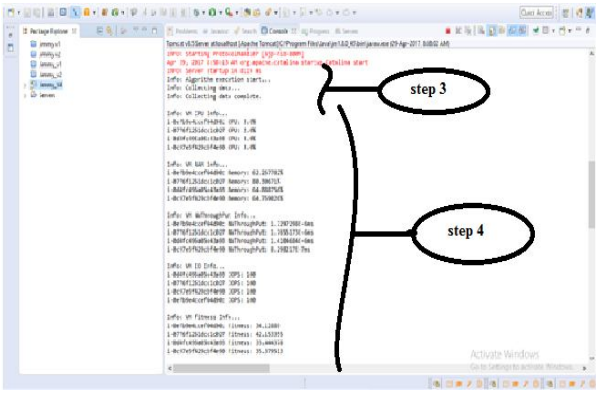

Figure 4: output of all steps

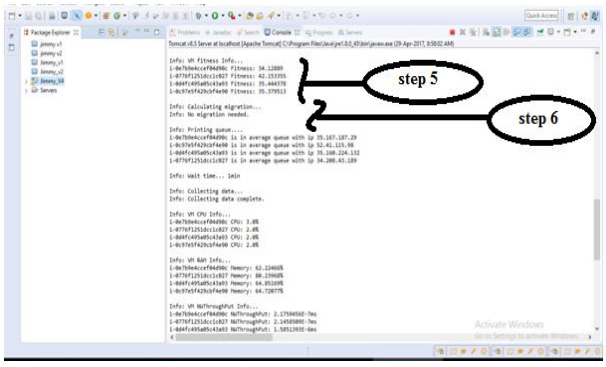

Figure 5: output of all steps

In this step algorithm execution start and all historical data collected after fitness function calculated for each parameters after find the average of fitness function. After all data normalize or printing in a queue with instance ip. After use of linear time series in prediction model make the decision

for VM Migration. When server are overloaded and underloaded of the system make the decision for migration.

Step 3: Algorithm start

```
Info: Algorithm execution start...
Info: Collecting data...
Info: Collecting data complete.
```

Figure 6: output of algorithm start

In this step Algorithm execution start and use of AWS instances data collecting

Step 4: Using all parameter CPU utilization

```
Info: VM CPU Info...
i-0e7b9e4ccef04d90c CPU: 3.0%
i-0776f1251dcc1c027 CPU: 2.0%
i-0d4fc495a05c43a93 CPU: 3.0%
i-0c97e5f429cbf4e90 CPU: 3.0%

Info: VM RAM Info...
i-0e7b9e4ccef04d90c Memory: 62.257782%
i-0776f1251dcc1c027 Memory: 80.30671%
i-0d4fc495a05c43a93 Memory: 64.888756%
i-0c97e5f429cbf4e90 Memory: 64.759026%

Info: VM NWThroughPut Info...
i-0e7b9e4ccef04d90c NWThroughPut: 1.7297298E-6ms
i-0776f1251dcc1c027 NWThroughPut: 1.7655173E-6ms
i-0d4fc495a05c43a93 NWThroughPut: 1.4104684E-6ms
i-0c97e5f429cbf4e90 NWThroughPut: 8.298217E-7ms

Info: VM IO Info...
i-0d4fc495a05c43a93 IOPS: 100
i-0776f1251dcc1c027 IOPS: 100
i-0c97e5f429cbf4e90 IOPS: 100
i-0e7b9e4ccef04d90c IOPS: 100
```

Figure 7: output of CPU utilization

In this step fitness function calculated. All are the parameter fetched , which are VM CPU, VM RAM, VM NW Throughput, VM Input Output Information calculated. After that all parameter evaluated in fitness function calculation can be done in below figure.

Step 5: average of fitness function calculated

```
Info: VM fitness Info...
i-0e7b9e4ccef04d90c Fitness: 34.12889
i-0776f1251dcc1c027 Fitness: 42.153355
i-0d4fc495a05c43a93 Fitness: 35.444378
i-0c97e5f429cbf4e90 Fitness: 35.379513
```

Figure 8: output of fitness function

In this step fitness function calculated. Using of the above all fitness parameter calculate the average of all them after that fitness function calculated

Step 6: normalize data in queue.

```
Info: Printing queue....
i-0e7b9e4ccef04d90c is in average queue with ip 35.167.187.29
i-0c97e5f429cbf4e90 is in average queue with ip 52.41.115.98
i-0d4fc495a05c43a93 is in average queue with ip 35.160.224.132
i-0776f1251dcc1c027 is in average queue with ip 34.208.43.189
```

Figure 9: output of normalize in queue

In proposed flowchart normalization step done in this output. All average of fitness function printing in queue and find there two instances ip in average queue in this figure. And find the waiting time for instances in AWS.

Step 7: Using prediction model make decision Migration or not

```
Info: Calculating migration...
Info: No migration needed.
```

Figure 10: output of migration

In this step in algorithm main concept work be done in this output. This output process use of the prediction model implementation for make the decision migration or not. In prediction model linear time series use and in this process visualize the series and Stationeries after that build of this model and make the decision. In prediction model is work on the past data for future. When server overloaded and under loaded of the system so it try for migration but use of prediction model inform to linearly time wise data for which one is free uses of the past data for migration and after that inform for migration. But this step system are not under loaded and not overloaded so print Migration is not needed.

Step 8: Virtual Machine Migration

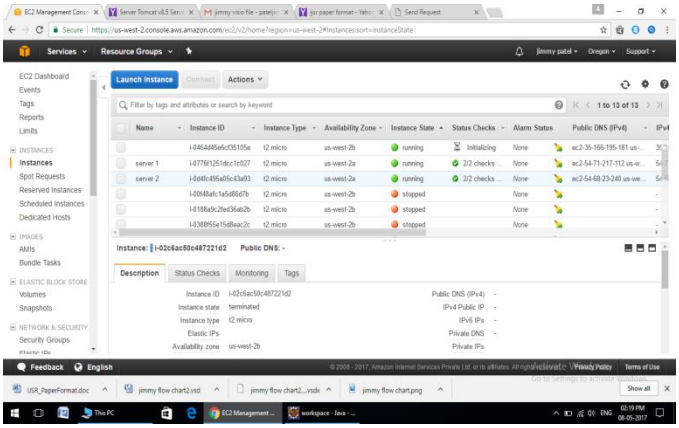

Figure 11:Migrate the machine

In this step VM Migration is done when system is underloaded or overloaded after that Migration done. In this output Virtual machine migrate in live AWS system.

Step 9: Save the data

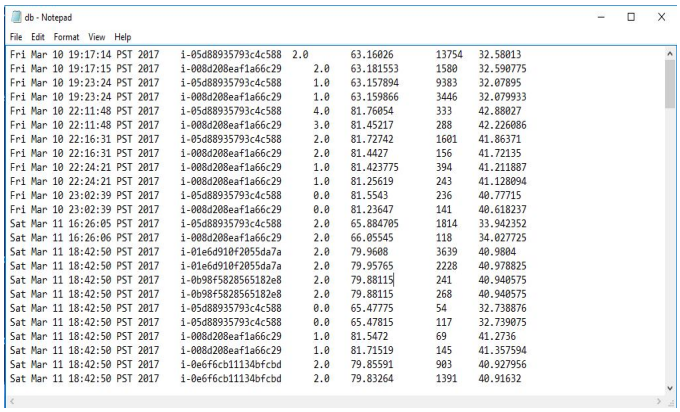

Figure 12: output of saved data

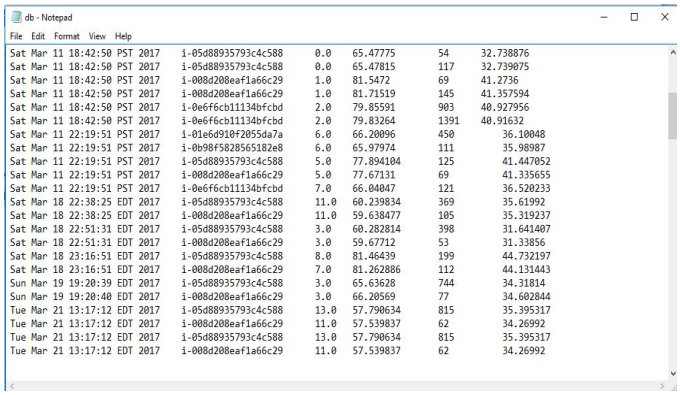

Figure 13: output of saved data

In this step accesses data can be save in db.log file.

VII. CONCLUSION

In cloud computing, Virtualization means more than one VM in single physical machine. Virtualization is for the most part valuable to deal with workload adjusting between physical machine in datacenter when accessible assets are insufficient for VMs. Based on this, the performance of VM migration is done in CPU utilization. It decrease total migration time and down time using prediction techniques. It increase the response time and work done in 0% down time.

VIII. FUTURE SCOPE

We will create web service architecture and move for a B2B customer for further to accept the performance testing and expectance testing. To implement the working of algorithm on Amazon web service(AWS) and analyze the performance and check the response time.

REFERENCES

[1] S. Liyanage, S.Khaddaj, "VM Migration Strategy in Cloud Computing.", School Of Computing And Information Systems, Kingstone University, Kingston Upon Thames, United Kingdom,IEEE, 2015

[2] Hiren B. Patel, "Efficient VM Migration in Cloud Computing" Sankalchand College of engineering Visnagar, Gujarat, india, IEEE, 2015

[3] Jayasree P, Minakshisharma, "POLVM: Parallel Optmized Live VM Migration" Sri Sai college of engineering and technology, Punjab, india, IEEE, 2014

[4] Hui Lu, Cong Xu, Cheng Cheng, RamanaKompella, DongyanXu, " vHaul: Towards optimal scheduling of live Multi-VM migration for Multi-tier appications," Department of computer science, purude university, IEEE, 2015.

[5] M. Arunfera, P.vanathi, M.sarumathi, I.ashabanu, V.umayal"A Novel Agent Based Match Making Algorithm In Multitenant Architecture," Thaiagarjar college of engineering, madurai,india, IEEE, 2015.

[6] Hatem Ibn-Khedher, Emad Abd-Elrahman and Hossom Afifi, "OMAC: Optimal Migration Algorithm for Virtual CDN", Institute Mines-Telecom,Telecom Sudparis,Saclay, France, IEEE,2016

[7] Jun chen, Yun chuan Qin, Yu Ye, Zhuo Tang, "A Live Migration Algorithm for VM in a Cloud Computing Environment", College of Information Science and Engineering Hunan University, Changsha, China, IEEE,2015

[8] Raabiatul Addawiyah Mat Razali, Ruhani Ab Rahman, Norliza Zaini, Mustaffa Samad, " VM Migration Implementation in Load Balancing for Cloud Computing", Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia, IEEE, 2014