Intelligent Scheduling System for Dynamic Resource Allocation in Cloud Computing

[I]Saranya.S, [II]Murugan.B.S

[I,II]Dept. of IT, Kalasalingam University, Krishnankoil, India

Abstract

Cloud computing allows business users to scale up and down their resource usage based on needs. In this system, intelligent agent uses virtualization technology to allot data centre resources dynamically based on the application demands. The concept of VM shift algorithm is to measure the unevenness in the multidimensional resource utilization of a server. The algorithm placed on the intelligent agents as a software system on the data centre. The intelligent agents are monitor allocation of resource from the data centre to the VM which is placed on the server. If any overload and under load will be identified, it will perform the avoidance through the usage of the algorithm. This algorithm is to shift the VM from one server to other server when the overload is detected. By using this algorithm, it will combine different workloads well and get improved overall utilization of server resources.

Keywords

Intelligent Agent, Cloud computing, Resource Allocation, Virtualization, Migration

I. Introduction

Cloud computing is an online based computing, resources, software, and data that provided to computers and other devices on demand, like an electricity grid. It encompasses any subscription based or pay by use service that, in real time, extends its existing capabilities. It conjointly reduces the cost of purchasing physical infrastructure by rental usage from the third party provider. In cloud there are three segments are available. They are application, platform, Infrastructure. In application on demand software services provided. It is having the lot of varieties on the usage of service. The pricing scheme is varied on the usage of the software to the end users. The platform segment is on the products to deploy the internet. Amazon, Google, which is the developed platform that allows the users to access the application. The infrastructure like the Google gears permits the users to develop the applications.

Resource allocation is component of assign the resources to the users, which is allotted dynamically using the virtual machine. A virtual machine is a proficient, isolated duplicate of real machine. It used through the virtual machine monitor (VMM). It used to allot the resources from the data centres to the physical machine to the physical machine.

This is terribly essential to allot the resource wants of VM. Due to workloads are growing and shrink based on the users. Therefore, the overload occurred on the data centre.

VM shift algorithm achieves the goal of overload rejection. This algorithm is to avoid the overload of server on the virtual machine. It set on the data centre.

Data centre is to allot the resources to the virtual machines. In data centre, the intelligent agent is located. The agent monitors the allocation of resources for avoiding the overload on the server. The work of intelligent agent is supervision for the aim of overload rejection. The intelligent agent uses the VM shift algorithm for the allocation and observation of the resources.

II. System Overview

In the system, overview contains the system design. It contains three elements. They are VM scheduler, data centre, and physical machines. The VM scheduler collects the hot spots and also the cold spots for the allocation. Data centre is employed to assign the resources to the physical machine. Intelligent agent can placed on the Data centre.

When the physical machines are, require the use of resources from the data centre. On that time, the intelligent agent checks the overload of the server.

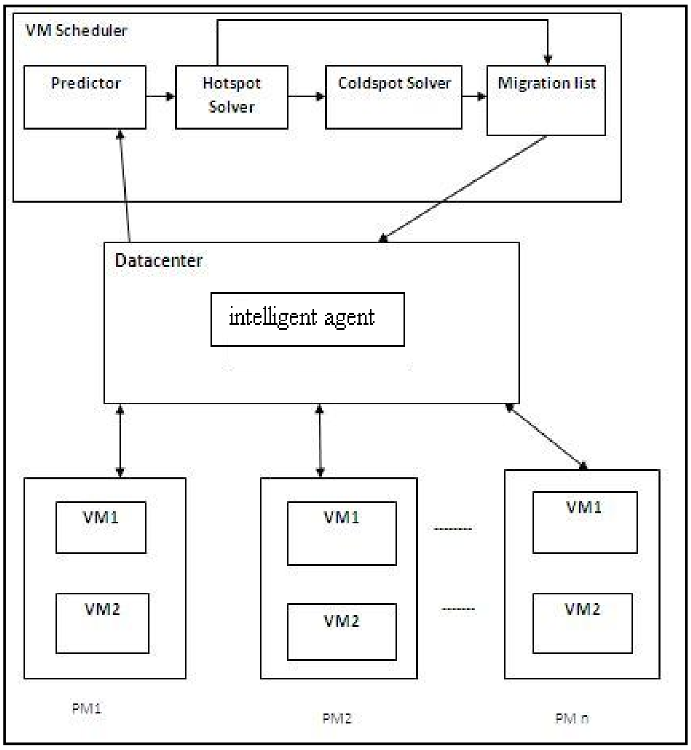


Fig. 1: System Architecture

The results are sending to the VM scheduler. The scheduler organizes the list of hot spots and cold spots.

It will organize the list from the vary of hotspots to the cold spots. By supported the result, intelligent agent is to assign the resource. The resource allotted to the server that has the minimum overload. Because of once the resource allotted to the utmost overloaded server, it will decrease the performance of the CPU. Therefore solely the resource is allotted to the minimum overload that is needed an equivalent resource.

In this method, the overload should be decreased on the server. The virtual machine is additionally shifted in this process for avoiding the overload. It is conjointly increases the performance of the physical machine. Therefore, all the physical machines are maintains the high performance.

III. VM Shift Algorithm

The process of VM shift algorithm is to checks the unevenness utilization of the server. This algorithm is placed on the intelligent agent. By exploitation, this algorithm the intelligent agent monitors the server. If any unevenness occurred on the server, it provides the resource to the opposite server desires an equivalent resource and VM shifted to the opposite machine by reducing the overload. Thus, it will increase the performance of the server.

A. Hot Spot Identification

The algorithm is to calculate the standing of the overload. The hot spot identification is that the method of characteristic the overload of the server. It termed as if the resource utilization, which is above the hot porch.

It indicates that the server is over loaded and therefore the VMs running on the server ought to be migrated away. The temperature of a hotspot reflects its overloaded degree. If a server is not a hotspot reflects the temperature is zero.

The warm threshold is that the level of resource utilization that is satisfies the hot threshold as same as warm threshold. For the server could be a hot spot if either its CPU usage is 90 percent. However the CPU normal usage is 80 percent.

B. Cold Spot Identification

The cold spot identification is that the method of identifying the below load of the server. It termed as if the resource utilization of the server is below the cold porch. It is referred to as because the cold spot.

It indicates that the server is below loaded the VMs are not running on the server. Therefore, the server is in idle state. The temperature of a cold threshold reflects its below loaded degree. For the server is cold spot if either its CPU usage is 20 percent. However, the CPU normal usage is 80 percent.

C. Hot and Cold Spot Alleviation

It sorts the list of the hot spots within the system in descending order (handle the hottest one). The goal of the algorithm is to eliminate the hot spots. In these approach, can realize the primary hottest spot within the list. Hot spot VMs are shift to the opposite server by reducing the overload of the server.

So first, got to realize the destination server to migrate the VM. If the VMs are shift, it will scale back the overload. The removal of the VM is reduces the load of the server most. Once shifting the VM, the destination server is additionally not overloaded. This sort of server chosen for the shifting. Otherwise, move to the next server for the shifting.

After performing the shifting, each the servers are not overloaded. Then solely move to succeeding hottest one to shift the VM. This can be continued for all the hottest spots. The cold spots are to be turned off for saving the energy.

IV. Related Work

A. Resource Allocation by Automated Control

The resource ought to be shared by the application using the virtualized data centres [7]. In this approach, service level objectives (SLOs) are difficult to satisfy workload. Because of resource patterns changed based on the user. So introduce the concept of the Auto control, which control all the resources. If workload changed, the resources usage also changed automatically [8]. Auto control is an integration of online model estimator and a novel multi-input, multi-output resource controller.

Auto control takes the multi-input from the resources. After checking the workload of the server, auto control will allot the resources to the application. This will also achieve the application SLOs. In this work, the RUBiS[7], TPC-W benchmarks used to detect the overload of the CPU and I/O by allocating the resource across multiple hosts. It also allocating the resource based on the priorities to provide the service separation. Automatic scaling [7] of web application was used for data centre environments. In MUSE, each server has the replication of all the web application running in the system [7], [9].

B. Resource Allocation by Dynamic Placement of Virtual Machine

The live migration [1] concept is used in this approach. A dynamic migration and consolidation algorithm [20] used on this approach. It uses the improvement of the static server consolidation [20] by decreasing the amount of the capacity and the rate of the service level agreement. Therefore, it is having the workload of 50% as compared to the static consolidation approaches.

Another output of this approach is that the rate of SLA [4] violations at the fixed capacity may be reduced by up to 20%. The result will be based across the variety of operating systems applications. In this approach the physical machine having the virtualized layer [17]. The virtualized servers have capability of maintaining five virtual machines.

If any overload has been occurred on the physical machine, it will shift the virtual machine, whose elimination is to reduce the overload. This type of virtual machine should be shifted to another physical machine. It will increase the overall performance of the system. Map Reduce [11], [12] is a type of cloud service. Quincy [13] adopts min-cost flow model in task scheduling to maximize data locality. Delay Scheduling [19] algorithm gives the execution time for data locality.

C. Resource Allocation by Live Migration

This is a most widely used technique on the cloud computing environment [1],[2],[3]. Here the volume metric is used. It takes the volume of the pm and VM. It sorts the pm and VM based on the volume [1]. The migration decision can be taken based on the volume to size ratio. If the overload identified, it will select the VM of overloaded server and shifts to the physical machine. If the algorithm should select the wrong VM, it is having the failure of the migration [5] alleviation.

The harmony system applies the virtualization [8] technology. The extended vector product is also used to identify the imbalance of the server. These algorithms are used for an offline.

V. Experimental Results

The experiments are conducted using the virtual machine. The virtual machines have VM scheduler. The same parameters are in the simulation.

A. Algorithm Effectiveness

The effectiveness of an algorithm is in the overload and VM migration. Here the experiment consisting of five PMs and five VMs. First increases the CPU load of the two VMs on PM. It must be create an overload. When the load increases, the performance of the CPU is decreased. When the load decreases, the performance of the CPU is increased.

The load is increased in the server. The server is shift the virtual

machine to other under loaded server. The algorithm resolves the overload by migrating VM2 to PM2. It goes to the stable state.
After some time, when we decreases the load of all the VMs gradually. The server goes to under load. As the load goes up and down, algorithm will repeat the migration. The algorithm is compared to with and without load prediction.
When the load prediction is disabled, algorithm uses the last observed load in its decision-making. The load prediction radically reduces the average number of hot spots in the system.
The average numbers of APMs remain essentially the same with or without load prediction. It compares the average number of migrations per VM in each decision with and without load prediction. Here the performance of the heap memory is expressed in the following diagram. The performance will be taken by the time and the percentage. This graph shows the suddenly increase and decrease of the performance. The data's are moved sudden increase and decrease of the memory.

B. Impact of Live Migrations

The VM Live migration used its impact on application performance. Among the five migrations, we randomly take seven corresponding TPC-W sessions undergoing live migrations.

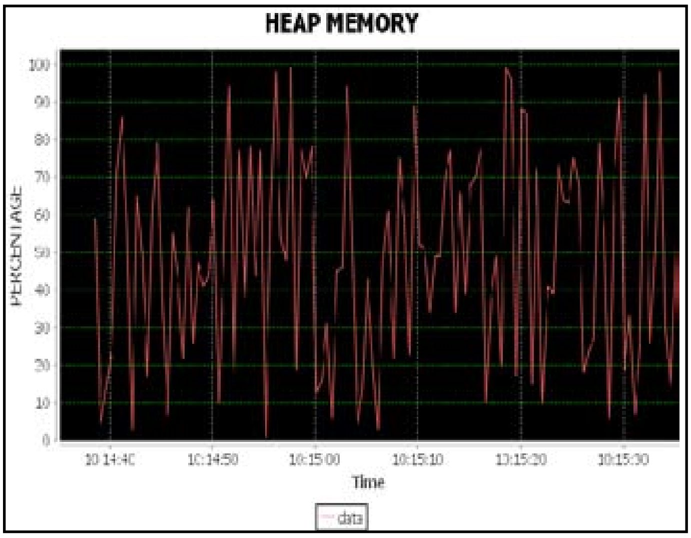


Fig. 2: Performance of memory

All the sessions run the "shopping mix" workload with lot browsers.
Fig. 3. shows the normalized web interactions per second (WIPS) for the 7 sessions. The only exception is session 3 whose degraded performance is caused by extremely busy server in the original experiment.

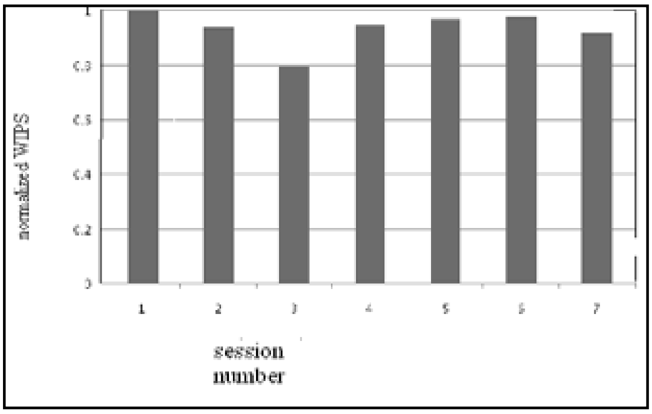


Fig. 3: Impact of Live Migration

As a target for comparison, we rerun the session with the same parameters but perform no migration and use the resulting performance as the baseline.

C. Resource Balance

The goal of VM shift algorithm is to mix the workloads with different resource requirements together. So that the overall utilization of server capacity is improved. The algorithm handles a mix of CPU, memory, and network intensive workloads. The memory intensive applications are created by allocating memory on demand. The memory consumption is kept low for this.fig4. shows the number of hots spots. Load prediction significantly reduces the average number of hot spots in the system during a decision run. The number of VMs taken as X-axis and the number of hotspots taken as y-axis. the graph will be plotted between with and without prediction. Without prediction, the number of hot spots is increased. However, the algorithm reduces the hot spots with prediction.

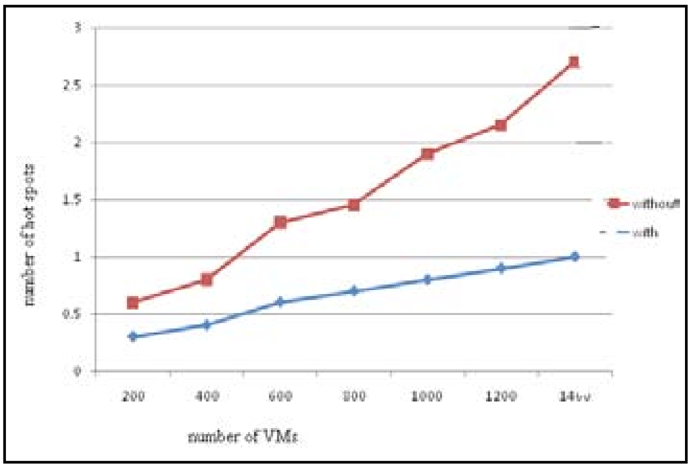


Fig. 4: Number of Hot Spots

Fig. 5. Shows the number of APMs is that the average numbers of APMs remain essentially the same with or without load prediction. Here the graph plotted between with and without prediction. This gives the same value of both with and without prediction.
Initially, keep the load of all VMs low and deploy all the CPU intensive VMs on PM2 and PM3. Then we increase the load on all VMs on gradually to make the underlying PMS hotspots. Fig. 6. Shows the number of migrations in the system with load prediction is smaller than that without prediction.

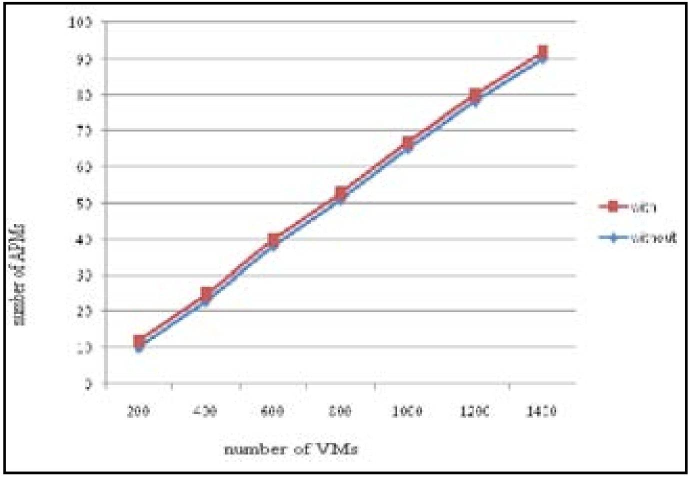


Fig. 5: Number of APMs

Here the graph plotted between with and without prediction. Here the number of migration increased using the algorithm.
So it reduces the overload on the server.

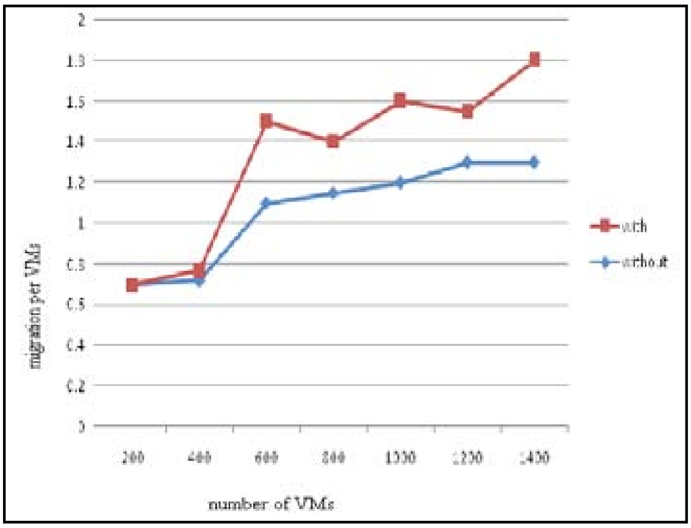
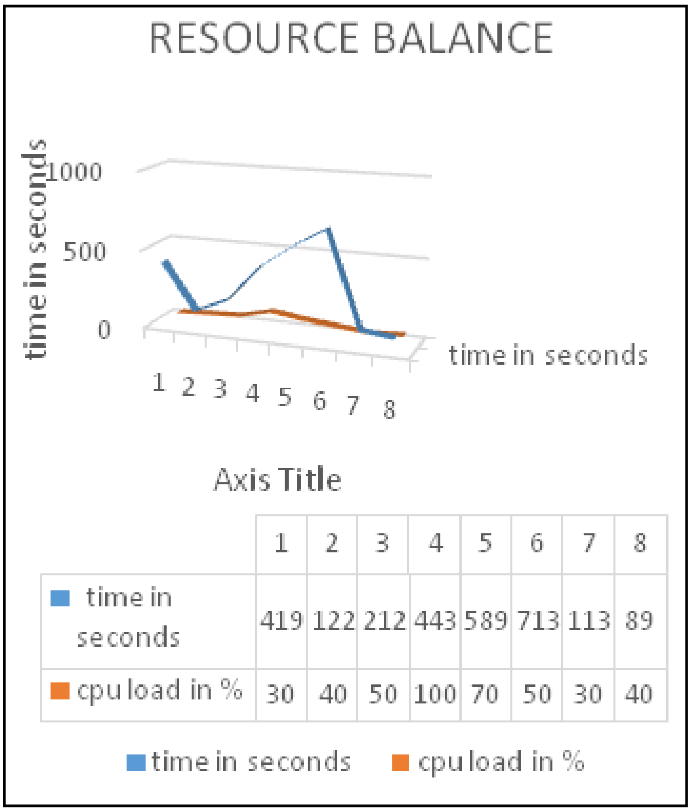


Fig. 6: Number of Migration

This algorithm used to migrate the one VM to other server for achieving the high performance.



	1	2	3	4	5	6	7	8
time in seconds	419	122	212	443	589	713	113	89
cpu load in %	30	40	50	100	70	50	30	40

Fig. 7: Resource balance

Fig. 7. shows that the resource balance of the CPU. Time has been taken in seconds and CPU load as taken as a %. The CPU load is gradually increased. After some seconds load is gradually decreased. So the load is decreased by migrating the VMs.

VI. Conclusion

The problem of dynamic resource allocation is a challenging task on the cloud environment. Hence the proposed method has described a VM shift algorithm, based on intelligent agent, which seems well adapted and efficient for avoiding the overload and increases the performance. By the intelligent agent helps the identification of the overloaded server and shift the virtual machine. It is used for avoid the overloading on the server. On the placement of Intelligent agent on the Data centre is allotted the resources to the servers. So it will avoid the overload and leads to the good performance. Further development of the work is control theoretic approach to find the overload of the server

VII. Future Work

In the intelligent agent is used to monitor and avoid the overload of the server. In the future work, control theoretic approach will find the overload of the server.

References

[1] *C. Clark, K. Fraser, S. Hand, J.G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live Migration of Virtual Machines," Proc. Symp. Networked Systems Design and Implementation (NSDI '05), May 2005.*

[2] *M. Nelson, B.-H. Lim, and G. Hutchins, "Fast Transparent Migration for Virtual Machines," Proc. USENIX Ann. Technical Conf., 2005.*

[3] *M. McNett, D. Gupta, A. Vahdat, and G.M. V an extoelker, "Usher: An Extensible Framework for Managing Clusters of Virtual Machines," Proc. Large Installation System Administration Conf. (LISA '07), Nov. 2007.*

[4] *N. Bobroff, A. Kochut, and K. Beaty, "Dynamic Placement of Virtual Machines for Managing SLA Violations," Proc. IFIP/IEEE Int'l Symp. Integrated Network Management (IM'07), 2007.*

[5] *T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif, "Black-Box and Gray-Box Strategies for Virtual Machine Migration," Proc. Symp. Networked Systems Design and Implementation (NSDI '07),Apr. 2007.*

[6] *R. Nathuji and K. Schwan, "Virtualpower: Coordinated Power Management in Virtualized Enterprise Systems," Proc. ACM SIGOPS Symp. Operating Systems Principles (SOSP '07), 2007.*

[7] *C. Tang, M. Steinder, M. Spreitzer, and G. Pacifici, "A Scalable Application Placement Controller for Enterprise Data Centers," Proc. Int'l World Wide Web Conf. (WWW '07), May 2007.*

[8] *A. Singh, M. Korupolu, and D. Mohapatra, "Server-Storage Virtualization: Integration and Load Balancing in Data Centers," Proc. ACM/IEEE Conf. Supercomputing, 2008.*

[9] *G. Chen, H. Wenbo, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao, "Energy-Aware Server Provisioning and Load Dispatching for Connection-Intensive Internet Services," Proc. USENIX Symp.Networked Systems Design and Implementation (NSDI '08), Apr. 2008.*

[10] *L. Siegele, "Let It Rise: A Special Report on Corporate IT," The Economist, vol. 389, pp. 16, Oct. 2008.*

[11] *M. Zaharia, A. Konwinski, A.D. Joseph, R.H. Katz, and I. Stoica, "Improving MapReduce Performance in Heterogeneous Environments, "Proc. Symp. Operating Systems Design and Implementation (OSDI '08), 2008.*

[12] *T. Sandholm and K. Lai, "Mapreduce Optimization Using Regulated Dynamic Prioritization," Proc. Int'l Joint Conf. Measurement and Modeling of Computer Systems (SIGMETRICS '09), 2009.*

[13] *M. Isard, V. Prabhakaran, J. Currey, U. Wieder, K. Talwar,*

and A.Goldberg, "Quincy: Fair Scheduling for Distributed Computing Clusters," Proc. ACM Symp. Operating System Principles (SOSP '09), Oct. 2009.

[14] D. Meisner, B.T. Gold, and T.F. Wenisch, "Powernap: Eliminating Server Idle Power," Proc. Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS '09), 2009.

[15] Y. Agarwal, S. Hodges, R. Chandra, J. Scott, P. Bahl, and R. Gupta, "Somniloquy: Augmenting Network Interfaces to Reduce Pc Energy Usage," Proc. USENIX Symp. Networked Systems Design and Implementation (NSDI '09), 2009.

[16] M. Armbrust et al., "Above the Clouds: A Berkeley View of Cloud Computing," technical report, Univ. of California, Berkeley, Feb. 2009.

[17] T. Das, P. Padala, V.N. Padmanabhan, R. Ramjee, and K.G. Shin,"Litegreen: Saving Energy in Networked Desktops Using Virtualization, "Proc. USENIX Ann. Technical Conf., 2010.

[18] Y. Agarwal, S. Savage, and R. Gupta, "Sleepserver: A Software-Only Approach for Reducing the Energy Consumption of PCS within Enterprise Environments," Proc. USENIX Ann. Technical Conf., 2010.

[19] M. Zaharia, D. Borthakur, J. Sen Sarma, K. Elmeleegy, S. Shenker, and I. Stoica, "Delay Scheduling: A Simple Technique for Achieving Locality and Fairness in Cluster Scheduling," Proc.European Conf. Computer Systems (EuroSys '10), 2010.

[20] N. Bila, E.d. Lara, K. Joshi, H.A. Lagar-Cavilla, M. Hiltunen, and M. Satyanarayanan, "Jettison: Efficient Idle Desktop Consolidation with Partial VM Migration," Proc. ACM European Conf. Computer Systems (EuroSys '12), 2012.

Author Profile

Saranya.S received B.Tech (IT) Degree from Kanchi Pallavan Engineering College, Kanchipuram in 2012. Pursuing M.Tech (IT) in Kalasalinam University, Krishnankoil-626126. She was interested in the area of cloud computing.

B.S.Murugan received B.Tech degree in Information Technology and M.Tech in Information Technology in 2005 and 2009 respectively. He is working as Assistant Professor in the School of Information and Computing sciences, Kalasalingam University, India. He has published 2 papers in International conferences. His current research interests include Distributed Computing, Grid Computing and Cloud Computing. He is an active member of ISTE.