DYNAMIC WORKLOAD PERFORMANCE OPTIMIZATION MODEL FOR MULTIPLE-TENANCY IN CLOUD SYSTEMS

Atwine Mugume Twinamatsiko, Ali Naser Abdulhussein Abdulhussein, Jugal Harshvadan Joshi, Arash Habibi Lashkari, Mohammad Sadeghi

Postgraduate Centre of Study (PGC), Limkokwing University of Creative Technology, Cyberjaya, Malaysia

Abstract - Cloud entails the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer. With this concept it's easy to reduce complexity for the clients as they don't have to handle installations and other things but just pay and to the provider a way to earn on a pay-as-use basis. Despite the concept being advantageous it's faced with complexities on the basis of handling dynamic workloads for multiple tenant systems. In this paper we propose a many-to-many entity relationship to handle the complexities of architectural design for the above mentioned types of systems.

Keywords: Cloud, Cloud Computing, Cloud System, Workload Performance, Multiple-Tenancy

1. Introduction

The cloud concept is a model of computing where clients are able to use remote resources based on certain principles of pay as you use (Aceto, G. et al., 2013). In this way the clients are excused of the burden of dealing with expenses and other issues that are involved in setting up new premises while using other provider's resources.

Cloud computing is an entirely internet-based approach where all the applications and files are hosted on a cloud which consists of thousands of computers interlinked together in a complex manner. Cloud computing incorporates concepts of parallel and distributed computing to provide shared resources; hardware, software and information to computers or other devices on demand. These are emerging distributed systems which follows a "pay as you use" model. The customer need not buy the software or computation platforms. With internet facility, the customer can use the computation power or software resources by paying money only for the duration he/she has used the resource. This forces the conventional software licensing policies to change and avoids spending of money for the facilities the customer does not use in a software package. (L.D., et al, P., 2013)

''Model for enabling convenient, on-demand network access to a shared pool of

configurable computing resources (networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.'' Cloud computing paradigm has some essential characteristics some of which include the following: (Aceto, G. et al., 2013)

On-demand self-service; it is paramount that anybody who needs to scale their resource preferences be able to do so on demand. That is customers utilizing the cloud services should be able to take on new resources with ease and automated (Parhizkar, B., et al, 2013).

Broad network access; there is need for connectivity to various resources over the internet so the importance of entire connectivity is really important. People should be able to access other services/resources while connected to the cloud services. Resource pooling; it's a good quality that a customer be able to take up as much of the resources as possible. If a need should arise where multiple resources are needed, the provider should be able to avail them (Parhizkar, B., et al, 2013).

Rapid elasticity; customers should be able to take on or leave some resources dependent on their needs and preferences with low difficulty. This is to say if someone wants to be availed more resources the cloud provider should be able to provide what is needed as soon as possible. Dependent on the location and deployment models cloud can be summarized in the following ways;

(i) Private cloud; this is a pool of resources that are owned by a particular entity, that is to say all the resources are used to serve certain purposes of the group.

(ii) Community Cloud; this is a pool of resources that is used by a group of entities to serve certain purposes; it is not accessible by the general public.

(iii) Public Cloud; these are resource pools that are accessible to the general public on a pay-as-use basis.

(iv) Hybrid Cloud Roles; this is a resource pool design that accommodates two types of cloud deployment models partly to serve a particular community and the other part to serve the general public.

Multiple roles can be supported by a Cloud developer, many of which can exist within a single organization: (i) Cloud Auditor; (ii) Cloud Service Provider; (iii) Cloud service carrier; (iv) Cloud Service Broker; (v) Cloud Service Consumer (Aceto, G. et al., 2013), (Garg, S.K., et al , R., 2013)

2. Related works

(L.D., et al, P., 2013) proposed a load balancing technique for cloud computing environments based on behavior of honey bee foraging strategy. This algorithm not only balances the load, but also takes into consideration the priorities of tasks that have been removed from heavily loaded Virtual Machines. The tasks removed from these VMs are treated as honey bees, which are the information updaters globally. This algorithm also considers the priorities of the tasks.

(Mei, Y. et al., 2013) argue that by exploiting optimizations for collocating different applications, performance improvement for cloud consumers can be as

high as 34 percent, and at the same time, the cloud providers can achieve over 40 percent performance gain by strategically collocating network I/O applications together.

Hamzeh et al (Member, S., 2012) have proposed an analytical technique based on an approximate Markov chain model for performance evaluation of a cloud computing center. Due to the nature of the cloud environment, we assumed general service time for requests as well as large number of servers, which makes our model flexible in terms of scalability and diversity of service time.

David C et al (Carrera, D. et al., 2012) present a technique that allows integrated management of heterogeneous workloads composed of transactional applications and long-running jobs, dynamically placing the workloads in such a way as to equalize their satisfaction. We use relative performance functions to make the satisfaction and performance of both workloads comparable.

(Wang, et al, 2011) discuss that, for tightly coupled CPU-intensive workloads, a key virtualization parameter for performance is the number of virtual machines: too many VMs will add significant inter- VM communication overhead and will limit the amount of virtual memory per VM; while too few VMs will require larger virtual memory per VM to obtain reasonable performance.

(Xu, H. et al., 2013) present a system that multiplexes virtual to physical resources adaptively based on the changing demand. We use the skewness metric to combine VMs with different resource characteristics appropriately so that the capacities of servers are well utilized. Our algorithm achieves both overload avoidance and green computing for systems with multi resource constraints.

(Bruneo, D., 2013), presented Anchor as a unifying fabric for resource management in the cloud, where policies are decoupled from the management mechanisms by the stable matching framework. We developed a new theory of job-machine stable matching with size heterogeneous jobs as the underlying mechanism to resolve conflict of interests between the operator and clients.

(Loraine Blaxter, 2001), presented a stochastic model to evaluate the performance of an IaaS cloud system. Several performance metrics have been defined, such as availability, utilization, and responsiveness, allowing to investigate the impact of different strategies on both provider and user point-of-views.

(Espadas, J. et al., 2013), say when large-scale applications are deployed over pay per use cloud high-performance infrastructures, cost-effective scalability is not achieved because idle processes and resources (CPU, memory) are unused but charged to application providers. Over and under provisioning of cloud resources are still unsolved issues. Even if peak loads can be successfully predicted, without an effective elasticity model, costly resources are wasted during nonpeak times (underutilization) or revenues from potential customers are lost after experiencing poor service (saturation). This work attempts to establish formal measurements for under and over provisioning of virtualized resources in cloud infrastructures, specifically for SaaS platform deployments and proposes a resource allocation model to deploy SaaS applications over cloud computing platforms by taking into account their

multitenancy, thus creating a cost-effective scalable environment.

(Espadas, J. et al., 2013), present a model to tackle over and underutilization when SaaS platforms are deployed over cloud computing infrastructures. This model contains three complementary approaches: (1) tenant-based isolation which encapsulates the execution of each tenant, (2) tenant-based load balancing which distributes requests according to the tenant information, and (3) a tenant-based VM instance allocation which determines the number of VM instances needed for certain workload, based on VM capacity and tenant context weight. After running all tests and simulations, the results were gathered and averages were calculated. In general, over and underutilization averages were reduced but only averages for underutilization were statistically improved.

(Ghosh, R. et al., 2013), think performance behaviors in such IaaS Clouds are affected by a large set of parameters, e.g., workload, system characteristics and management policies. Thus, traditional analytic models for such systems tend to be intractable. To overcome this difficulty, they propose a multi-level interacting stochastic sub-models approach where the overall model solution is obtained iteratively over individual sub-model solutions. By comparing with a single-level monolithic model, they demonstrate that their approach is scalable, tractable, and yet retains high fidelity. Since the dependencies among the sub-models are resolved via fixed-point iteration, they prove the existence of a solution. Results from their analysis show the impact of workload and system characteristics on two performance measures: mean response delay and job rejection probability.

Using interacting stochastic sub-models, (Ghosh, R. et al., 2013) propose a fast method suitable for analyzing the service quality of large sized IaaS Clouds. Results show that our approach enables the Cloud service providers to detect system bottlenecks. Moreover, optimization for a broad range of provider specific Cloud settings can be performed as the model allows exploring a large Cloud parameter space.

(Van den Bossche, 2013) tackle this problem by proposing a set of algorithms to cost-efficiently schedule the deadline constrained bag-of-tasks applications on both public cloud providers and private infrastructure. Their algorithms take into account both computational and data transfer costs as well as network bandwidth constraints. We evaluate their performance in a realistic setting with respect to cost savings, deadlines met and computational efficiency, and investigate the impact of errors in runtime estimates on these performance metrics.

Their results quantify the additional gains in cost-efficiency that can be achieved by adopting an EDF approach on the private cloud. They demonstrate that further cost reductions are realized if cost is used as a discriminating factor for selecting outsourced applications. In addition, an EDF scheduling policy for the private cloud is shown to significantly increase robustness with respect to runtime estimation errors, at an additional cost in turnaround time. (Van den Bossche, 2013)

(Liang, A., et al, 2013), aim to balance power consumption and performance, this paper proposes an adaptive workload-driven dynamic power management policy for homogeneous clusters, which dynamically adjusts the power mode of computing nodes according to workload variation. The proposed policy combines the pre-wakeup method and the feedback mechanism to reduce performance degradation due to the wakeup delay.

An adaptive workload driven power management policy has been presented with an objective to improve energy efficiency

according to workload variations. Based on the timeout threshold dynamic power management, the proposed policy integrates the pre-wakeup method and feedback control. Computing nodes of the cluster dynamically adjust the power mode according to workload variation. Since sleeping nodes can be awakened in advance through workload forecast, the job latency would be reduced. Thus, performance during power management is improved. Feedback control is adopted for revising the workload model. (Liang, A., et al, 2013)

(Qi, H. & Gani, A., 2014), A provide and insight on the background and principle of MCC, characteristics, recent research work, and future research trends. A brief account on the background of MCC: from mobile computing to cloud computing; these include some of the factors that can influence functionality and basis of the cloud and need to be considered meticulously. Some of the analyzed factors include; Autonomy, virtualization, reliability, Usability and extensibility.

(Shiraz, M., et al., 2012), analyze the impact of VM deployment for application offloading in simulation environment using CloudSim they also investigate the heavyweight aspects of current offloading algorithms by qualitative analysis. After analysis they propose an optimal model for distributed application development and deployment for MCC. The deployment of such lightweight application framework results in substantial performance gains and enhancements in overall performance of distributed application deployment and processing in MCC.

3. Analysis of previous work

In this paper, (Wu, D. et al., 2012) develop a simple theoretic model to analyze two typical P2P models for VM image distribution, namely, isolated-image P2P distribution model and cross-image P2P distribution model. They compare their efficiency under different parameter settings and derive their corresponding optimal server bandwidth allocation strategies. In addition, they also propose a practical optimal server bandwidth provisioning algorithm for chunk-level cross image P2P distribution mechanism to further improve its efficiency.

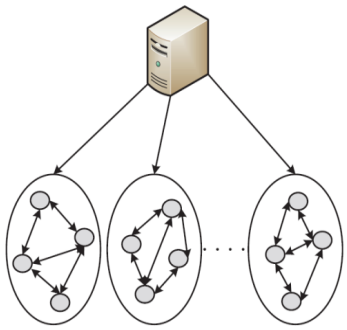

Figure 1: a simple theoretic model to analyze two typical P2P models for VM image distribution (Wu, D. et al., 2012)

Figure 1 demonstrates the contribution of (Wu, D. et al., 2012), a model for peer to peer relationship for virtual machines.

After the analysis done after several simulations some numerical statistics were compiled and expressed in the graphs below (Figure 2) for analysis which show the average distribution time under different request arrival rates (Left section) and the average distribution time of cross-image model when varying the fraction of common chunks (Right Section).

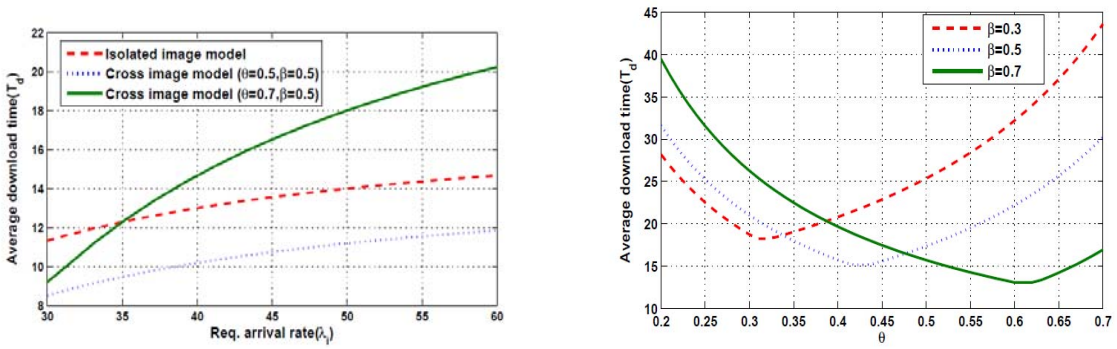

Figure 2: Average distribution time (Wu, D. et al., 2012)

Despite the conclusions of this paper some modeling could be done to enhance the performance for multiple tenant systems, under dynamic workloads over time.

4. Proposed new model

The comparative architecture above is made in a one-to-many form of architecture and therefore may suffer poor characteristics such as Poor resource utilization and others. After simulation and comparison I have concluded on the many-to-many architecture based on the characteristics above that would look like the one below. This model came about after multiple simulations on the cloud Sim API, the related information is discussed below.

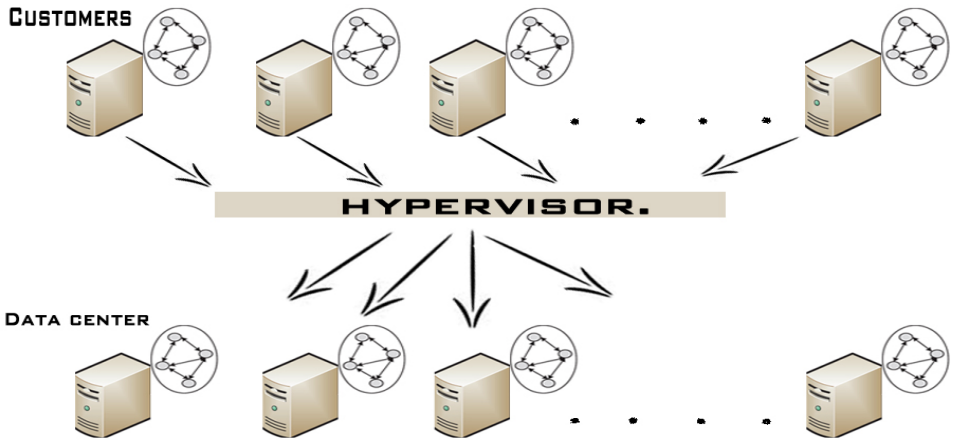

Figure 3: Representing the many-to-many architecture

In this architecture (Figure 3) the customers have different processes running with different needs, when forwarded to the hypervisor, the hypervisor allocates the processes to various hosts who are capable of handling multiple processes. The architecture is in the format of many to many as simulated from above.

4.1. Comparison table

The table above (Table 1) shows a comparison made between the different architectural models considered in the simulation tests. The notations shown in

the table represent the following: H represents high, P represents poor, G represents good, F represents fair as far as the numerical results and conclusions have been analyzed.

Mode	Cost	Resource Utilization	Power Consumption	Execution Time	Granularity
One-One	H	P	G	F	P
One-to-Many	H	P	G	F	F
Many-to-One	G	G	H	F	P
Many-to-Many	F	F	F	F	G

Table 1: An analysis demonstrating the capabilities of various factors

4.2. Simulating new model

The table below shows results after multiple simulations while considering various factors such as time, cost, and others for the many-to-many architecture. The table (Table 2) summarizes the factors and numerical representations of results achieved after simulation of the many-to-many architectural model.

NUMBER OF CUSTOMERS	BAND WIDTH	POWER CONSUMPTION (KW)	RESPONSE TIME (s)	RESOUCE UTILIZATION	COST (mu)	PE/ HOST	NO HOSTS	Time(m)
5	1	40	1750	34.25	60	5	7	29
10	1	46	1820	40	250	5	14	30
15	1	43.25	1800	37.5	382	5	17	30
20	1	40	1761.2	35	500	5	25	29
30	1	30.66	1832.5	26.6	636.5	5	36	31
50	1	30	1800	26.7	1250	5	60	30
80	1	28.6	1811	25	2000	5	90	30

Table 2: Numerical representation of various factors

The graph above (Figure 4) shows the changing tendency and unstable resource utilization in the architectures of cloud and multiple tenancy. It is a demonstration on how the resource utilization is effective in comparison to other architectures of cloud subjected to the same variable factors.

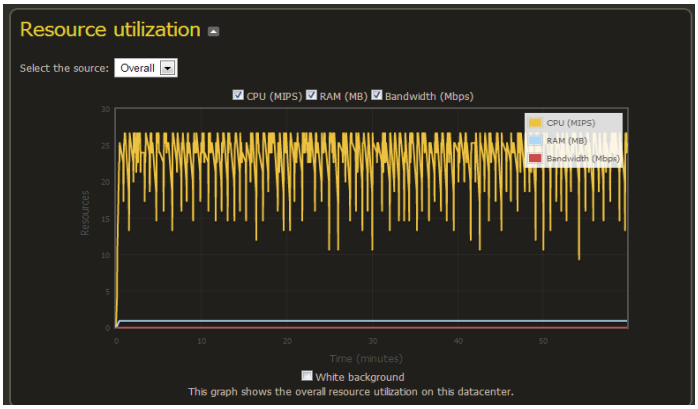

Figure 4: A graphical representation of the results

In the figure above (Figure 5) is a graphical demonstration of the factor power consumption if the many-to-many architectural model is applied to the concept of multiple tenancies for cloud systems.

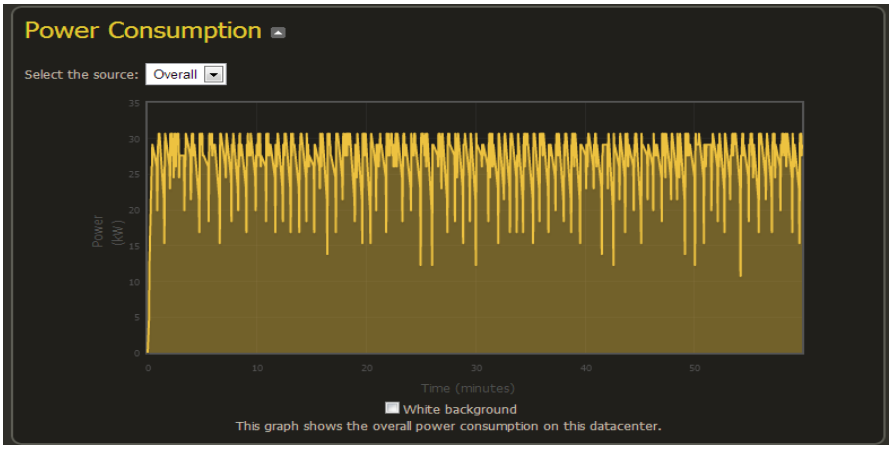

Figure 5: A graphical representation of the results

The figure above (Fig 6) is a graphical representation made after analysis of the factor; resource utilization for different architectures tested in the simulations. It is noted that the Many-to-many (M-M) architecture has the fair utilization of

resources as compared to many-to-one (M-1). As a final analysis it is obvious that in this factor the utilization favors the architecture for many to one as the best.

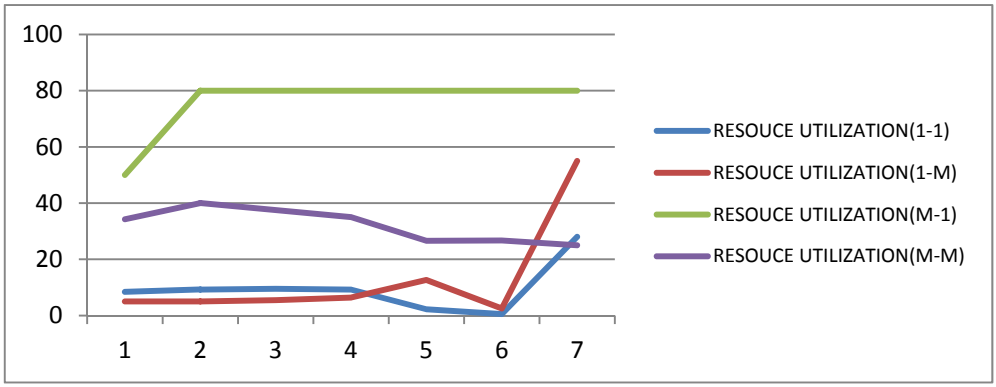

Figure 6: comparison of various models

4.3. Result and discussion

The results from comparing various factors such as, power consumption we realize a lot. From the graphs above we are able to see variations are the above factors. Such as below we look at the comparison in all models for resource utilization as demonstrated in the graph above.

RESOUCE UTILIZATION (1-1)	RESOUCE UTILIZATION (1-M)	RESOUCE UTILIZATION(M-1)	RESOUCE UTILIZATION(M-M)
8.4	5	50	34.25
9.25	5	80	40
9.52	5.45	80	37.5
9.23	6.4	80	35
2.25	12.65	80	26.6
0.5	2.5	80	26.7
28	55	80	25

Table 3: numerical representation for resource-utilization

From the table above, (Table 3) we are able to observe the optimal use of resources in respect to other factors; the many-to-many model has the optimal utilization of resources.
From the comparison table above we can see for a good model to run on optimizable workload the model of execution should be on a many-to-many basis. That is to say various customers with various types of needs can be attached to a data center with various hosts and according to the tests above there will be proper utilization of resources at fair cost for both the customer and the provider.

5. Conclusion

Cloud computing technology has evolved from simple grid computing architectures and parallel computing concepts to what is today a pay-as-use system. With this way of approaching technology people are able to use resources and reduced cost which is very advantageous both to the client and provider. In this paper we have considered the behavior of model architectures based on various factors. We wanted to find a way by which dynamic workload can be managed optimally for multiple tenancy architectures of cloud systems. In order to do the analysis we used the CloudSim simulator to test the functionality of the architectures we used. We conclude after many simulations and trials that the many-to-many is more advantageous in terms of proper functionality if various factors are to be considered for optimal and dynamic load administration in data centers.

Acknowledgement

The special thank goes to our helpful supervisor Dr. Arash Habibi Lashkari from Postgraduate school for his unrivaled supervision and guidance in our dissertation and project.

References

Aceto, G., Botta, A., De Donato, W., & Pescapè, A. (2013), Cloud monitoring: A survey. Computer Networks, 57(9), 2093-2115

Bruneo, D., 2013, A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems, IEEE Transactions on Parallel and Distributed Systems, pp.1–1

Carrera, D., Steinder, M., Whalley, I., Torres, J., & Ayguade, E. ,2012, Autonomic placement of mixed batch and transactional workloads. Parallel and Distributed Systems, IEEE Transactions on, 23(2), 219-231

Espadas, J., Molina, A., Jiménez, G., Molina, M., Ramírez, R., & Concha, D. , 2013, A tenant-based resource allocation model for scaling Software-as-a-Service applications over cloud computing infrastructures. Future Generation Computer Systems, 29(1), 273-286

Garg, S.K., Versteeg, S. & Buyya, R., 2013, A framework for ranking of cloud computing services, Future Generation Computer Systems, 29(4), pp.1012–1023

Ghosh, R. Ghosh, Rahul Longo, Francesco Naik, Vijay K, Trivedi, Kishor S, 2013, Modeling and performance analysis of large scale IaaS Clouds. Future Generation Computer Systems, 29(5), pp.1216–1234

L.D., D.B. & Venkata Krishna, P., 2013, Honey bee behavior inspired load balancing of tasks in cloud computing environments. Applied Soft Computing, 13(5), pp.2292–2303

L.D., D.B. & Venkata Krishna, P., 2013, Honey bee behavior inspired load balancing of tasks in cloud computing environments. Applied Soft Computing, 13(5), pp.2292–2303

Liang, A., Xiao, L. & Ruan, L., 2013, Adaptive workload driven dynamic power management for high performance

computing clusters, Computers & Electrical Engineering, 39(7), pp.2357–2368

Loraine Blaxter, Christina Hughes and Malcolm Tight., 2001, How to research, third edition, Open University Press

Mei, Yiduo Liu, Ling Member, Senior Pu, Xing., 2013, Performance Analysis of Network I / O Workloads in Virtualized Data Centers , 6(1), pp.48–63

Member, S., 2012, Performance Analysis of Cloud Computing Centers Using M = G = m = m þ r Queuing Systems. , 23(5), pp.936–943

Parhizkar, B., Abdulhussein, A. N. A., Joshi, J. H., & Twinamatsiko, A. M., 2013, A Common Factors Analysis on cloud computing models

Qi, H. & Gani, A., 2014, Research on Mobile Cloud Computing : Review, Trend and Perspectives. , pp.195–202

Shiraz, M., Gani, A., & Khokhar, R.H., 2012, Towards Lightweight Distributed Applications for Mobile Cloud Computing. *IEEE International Conference*, (1), pp.89–93

Van den Bossche, R., Vanmechelen, K. & Broeckhove, J., 2013, Online cost-efficient scheduling of deadline-constrained workloads on hybrid clouds, Future Generation Computer Systems, 29(4), pp.973–985

Wang, Q. & Varela, C. a., 2011. Impact of Cloud Computing Virtualization Strategies on Workloads' Performance, 2011 Fourth IEEE International Conference on Utility and Cloud Computing, pp.130–137

Wu, Di Zeng, Yupeng He, Jian Liang, Yi Wen, Yonggang., 2012, 2012 IEEE 4th International Conference on Cloud Computing Technology and Science On P2P Mechanisms for VM Image Distribution in Cloud Data Centers : Modeling, Analysis and Improvement. , pp.50–57

Xu, Hong Member, Student Li, Baochun Member, Senior, 2013, Anchor : A Versatile and Efficient Framework for Resource Management in the Cloud. , 24(6), pp.1066–1076