In 2010Jianfeng Zhan, Lei Wang, Weisong Shi, Shimin Gong and Xiutao Zang introduced Phoenix Cloud-Provisioning Resources for Heterogeneous Cloud Workloads [6]. It presented a RE conformity that state diverse RE necessities and make a novel system Phoenix Cloud to allow creating REs on require according to RE agreements. On behalf of two characteristic heterogeneous workloads: Web servicesand parallel batch jobs, proposed two corresponding resource provisioning solutions in two diverse Cloud scenarios. on behalf of three distinctive workload traces: SDSC BLUE, NASA iPSC and World Cup, experiments showed that: a) in the first Cloud scenario, when the throughput is almost same like that of a DCS, solution decreases the design size of cluster by about 40%; b) in the second Cloud scenario, solution decreases not only the total resource expenditure, but also the peak resource consumption maximally to 31% with respect to that of EC2 + Right Scale solution.

In 2011 Ajit B. Sharma proposed a Modeling and Synthesizing Task Placement Constraints in Google Compute Clusters [7] which addresses the concert impact of task residency constraints. Task residency constraints impact which resources tasks consume. Task placement constraints, such as distinctiveness individual by the Condor Class Ads method, offer a way to contract with machine heterogeneity and diverse software requirements in work out clusters. This understanding at Google suggests that task assignment constraints can have a huge impact on task scheduling delays. This paper is the first to expand a method that addresses the performance impact of task placement constraints. Here showed that in Google compute clusters, constraints can enhance average task scheduling delays by a factor of 2 to 6, which frequently revenue tens of minutes of additional task wait time. To recognize why, to begin a new metric, the Utilization Multiplier (UM) that extends the idea of resource utilization to comprise constraints. It showed that task scheduling delays enhance with UM for the responsibilities to learn. It as well shows how to describe and produce representative task constraints and machine properties, and how to include synthetic constraints and properties into presented performance benchmarks. Applying this come up to Google compute clusters, here find that these constraint characterizations accurately reproduce production performance characteristics.

In 2013 I. Solis Moreno proposed a paper on An Approach for Characterizing Workloads in Google Cloud to Derive Realistic Resource Utilization Models [8] which introduced analyzing behavioral patterns of workloads is critical to understanding Cloud computing environments. However, until now only a limited number of real-world Cloud data center trace logs have been available for analysis. This has led to a lack of methodologies to capture the diversity of patterns that exist in such datasets. This paper presents the first large-scale analysis of real-world Cloud data, using a recently released dataset that features traces from over 12,000 servers over the period of a month. Based on this analysis, we develop a novel approach for characterizing workloads that for the first time considers Cloud workload in the context of both user and task in order to derive a model to capture resource estimation and utilization patterns. The derived model assists in understanding the relationship between users and tasks within workload, and enables further work such as resource optimization, energy-efficiency improvements, and failure correlation. Additionally, it provides a mechanism to create patterns that randomly fluctuate based on realistic parameters. This is critical to emulating dynamic environments instead of statically replaying records in the trace log. Our approach is evaluated by contrasting the logged data against simulation experiments, and our results show that the derived model parameters correctly describe the operational environment within a 5% of error margin, confirming the great variability of patterns that exist in Cloud computing.

3. Existing System

In the existing system, the examination takes place with limited cloud traces from Google and yahoo to provide mechanisms to analyze and categorize workload patterns. In this system the main objective it to obtain coarse grain statistical data about jobs and tasks to classify them by duration. This characteristic limits the work's application to the study of timing problems. It makes it unsuitable to analyze the cloud computing issues related to resource usage patterns. In this system, group jobs with similar characteristics using clustering to analyze the resulting centroids. Unfortunately there is a lack of analyses to support the development of workload models that capture the inherent diversity of users and tasks, largely due to the limited availability of Cloud trace logs as well as the complexity in analyzing such systems.

Here they develop Cloud computing workload classifications based on task resource consumption patterns. The existing approach identifies workload characteristics, constructs the task classification, identifies the qualitative boundaries of each cluster and then reduces the number of clusters by merging adjacent clusters. This approach is useful to create the classification of tasks, but does not perform an analysis of the characteristics of the formed clusters in order to derive a detailed workload model. Finally, it is entirely focused on task modeling, neglecting user patterns.

In this system, it used Intra Cluster analysis algorithm. The cluster analysis and intra–cluster analysis do not contain sufficient detail to quantify the diversity of workload, instead presenting high–level observations. Furthermore, there is insufficient detail about the parameter distributions used; more detail is necessary in order for other researchers to simulate the workload obtained. Finally, the validation of the simulated model against that of the empirical data is based only on a visual match of the patterns from one single execution, and does not consider more rigorous statistical techniques. The following figure shows the existing system architecture:

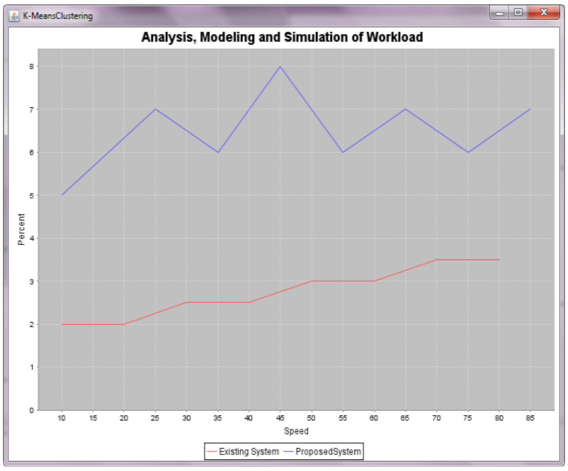

Figure 4: Speed of Existing and Proposed System

Figure 4 shows the graph obtained after simulation. It shows the comparison of speeds of existing and proposed system. From this graph, it can be seen that there is 3-4.5% increase in the speed of the proposed system.

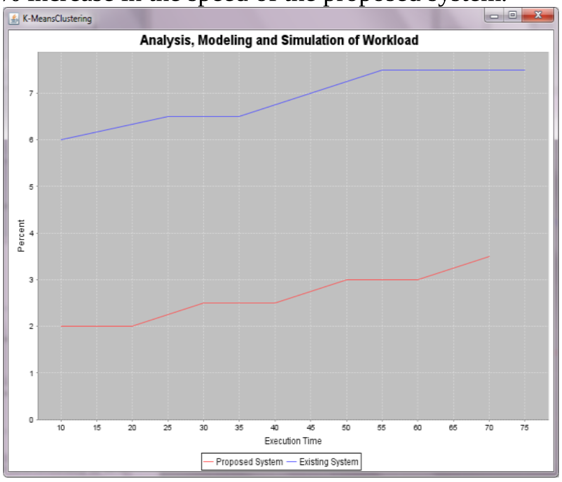

Figure 5: Execution Time of Existing and Proposed System

Figure 5 shows the graph obtained after simulation. It shows the comparison of execution times of existing and proposed system. From this graph, it can be seen that there is 3.5-4% decrease in the execution time of the proposed system. Therefore, this validates that the proposed system is better than the existing system.

8. Conclusion and Future Work

This paper presents an examination that quantifies the diversity of Cloud workloads and derives a workload model from a large-scale construction Cloud datacenter. The obtainable examination and model captures the distinctiveness and behavioral patterns of user and task variability across the entire system as well as different observational periods. The derivative model is implemented by means of the CloudSim construction and comprehensively validated during empirical comparison and statistical tests. From the explanation obtainable within this work and the outcome obtained from the simulations, a number of conclusions can be ended.

Future research includes extending the model to include tasks constraints based on server characteristics; this will allows us to analyze the impact of hardware heterogeneity on workload behavior. Other extensions include analyzing the workload from the jobs perspective specifically modeling the behavior and relationship of users and submitted jobs, accurately emulating and analyzing workload energy consumption and reliability enabling further research into energy-efficiency, resource optimization and failure-analysis in the Cloud environment. Finally, it is important to enable a collaboration link with the CloudSim group in order to integrate the proposed workload generator as an add-in of the current framework implementation allowing it to be made publicly available.

References

[1] Veeramallu, G. K. S. B. 2014. “Dynamically Allocating the Resources Using Virtual Machines.” (IJCSIT) *International Journal of Computer Science and Information Technologies.* 5(3):4646-4648
[2] CloudSim: A Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services https://code.google.com/p/cloudsim/
[3] El-Refaey, M. A. and Rizkaa, M. A. 2009. "Virtual Systems Workload Characterization: An Overview." *IEEE Intl. Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises.* 72-77.
[4] Kavulya, S., 2010. "An Analysis of Traces from a Production MapReduce Cluster." *IEEE/ACT int. Conf. Cluster, Cloud and Grid Computing.* 94-103.
[5] J. Zhan,., 2010. "PhoenixCloud: Provisioning Resources for Heterogeneous Workloads in Cloud Computing," *arXiv preprint arXiv*:1006.1401.
[6] B. Sharma., 2011. "Modeling and synthesizing task placement constraints in Google compute clusters." *Proc. ACM Symp. on Cloud Computing*. 1-14.
[7] I. Solis Moreno, 2013. "An Approach for Characterizing Workloads in Google Cloud to Derive Realistic Resource Utilization Models," *Int. Symp. on IEEE Service Oriented System Engineering.* 49-60.