# Performance Analyses for Massive Learning Grid Utilization through Virtual Warehousing


**Mr.G.Manoharan[1], Dr.P.Nirmala[2]**

[1]Ph.D Research Scholar, Manonmanium Sundararanar University , Tirunelveli.

[2]Associate Professor, Quaide Milleth College, Chennai.



## ABSTRACT

The Main objective of this proposed work is to create Web Learning site where the number of user can learn via any medium such as Video, Audio, Animation, Graphics etc., If the Number of user increase, the existing system will be very slow and data transfer will also be slow. This proposed work is to increase the speed of data transfer and also with less storage space compare to the existing process. This Process can be used via Grid Computing, Vertical process as Data Storage and Horizontal Process as Data Transfer to increase the speed and effective storage space. This Grid Computing, implemented in the heterogeneous network for Massive Open Online Course, for the large scale participant using the same media on a particular time due to the existing system will be very slow and data storage will not be effective too. To Avoid the Existing, the proposed work mainly focused on heterogeneity and task scheduling with different media depend on Sharable Content Object(SCO) for harmonizing massive large Object. Various methodologies like k-Mean Clustering, Clustering analysis used in Scheduling and Load balancing, Bee colonization for Tasking. This Proposed work will proved 85% to 90% of Accuracy and effective in Massive Open Online Courses.


***Keyword: Grid Computing, Sharable Content Object, Clustering, Scheduling.***

## 1.0 Introduction

Grid computing technology is widely adopted for deploying e-learning content in the web [Hsin-Chuan Ho et. al. 2004] as any e-learning environment would demand for large computing resources particularly for huge pool of simultaneous e-content users. Hence software and hardware are needed to be updated or renewed many times for this environment. Massive learning grid is a computational collaborative environment for effectively tackling large pools of e-learners of the web that is becoming vogue now-a-days. One of the fundamental issues in such grid environments is job scheduling, which is needed for achieving higher performance [Deepti Malhotra and Devanand 2012]. As grid environment is generally decentralized and it consists of heterogeneous systems, efficient scheduling technique would be very much needed for tapping appropriate resources for relevant e-content; say large or small or media intensive e-content. As large set of data and huge pool of users are involved in a dynamic environment of learning grids, virtual warehousing could be tried out for storing and retrieving data. Virtual warehousing is an important technology that is adopted in e-commerce. It provides flexibility in handling large data; but the main advantage of virtual warehouse is cost reduction. This is due to the fact that structuring of individual data set for many simultaneous users could be avoided, while allowing many such learners to pool from one data warehouse. The novelty of this paper is not on mere adaptation of data warehouse for storing e-content; but on the application of independent and Poison's probabilities for avoiding unnecessary trust computations on the clusters of the learning grid, thus economizing computational resources for several thousands of simultaneous e-users. The application of Poison's probability is justified as the proposed data warehousing framework that directs two servers simultaneously on the stored data through a pre processor. The paper elaborates two experimental procedures that determine empirical ratios of task/clusters, which will reduce the probability of encountering un-trusted clusters of the grid. Experiments were conducted on selective e-learning objects (prepared exclusively for experiments) by using GridSim 5.0. The paper also suggests splitting up of integrated long e-content into pedagogically suitable short objects on one hand and also splitting up media wise objects on the other hand. Results obtained from these proposed approaches are compared with the result of integrated e-content that computes trust values of clusters. Conclusions are drawn from these experiments which will be of immense use to learning grid researchers and designers.

## 2.0 Literature Support for Problem Identification and Methodology

Instructional documents could be kept in learning object repositories so that a search engine could rank them according to the similarity between the query and document [Wen-Chung Shih et. al.2009]. This suggestion would not only contribute to economizing queries, but also might avoid un-wanted uploading of long fully integrated e-content for all users, when all of whom might not require all the content all the time. Job scheduling through different methodologies in grid computing is posed with new challenges for better computing performances [Ng Wai Kent et al 2008]. These methodologies stipulate conditions for improved utilization of resources. Experimental methodology through simulation models that represent actual grid system is recommended which could be designed for the study of actual grid systems. Simulation studies are advantageous as they allow investigation of different situations in controlled conditions [Thanasis G. Papaioannou and George D. Stamoulis 2010]. A cluster participant of a grid service could only be able to offer the outcome of the collective efforts of groups of nodal participants rather than its individual efforts. Hidden information in grid systems have been identified and explained why they could not be handled satisfactorily by the existing cluster participants of grids.

Large scale grid computational studies could be achieved through simulations. This could be done with simple declarative language through modeling [Rajkumar Buyya et al. 2002]. The modeling can also be achieved through parametric studies. Simulation can achieve seamless execution on global computational grids. A 10% of allowance for simulating exactly on actual conditions has been suggested. Examples with 1980, and 2970 seconds have been demonstrated. 10% allowance was taken for adjusting delays of standard schedulers. CPU running time of task execution could be an important component particularly on interactive applications based on grids [Yuanynan Zhang et al 2006]. A new method has been proposed to predict the running time of tasks in a grid. Prediction of running time of tasks is based on a novel CPU load prediction method and is calculated from predictions of CPU loads. Experiment through 39 load traces on a UNIX system has been conducted and conclusions have been drawn. It is thus evident that simulation studies could demonstrate 100s of seconds of processing time which could be on CPU loads.

Resources of grid may join or leave the environment at any instance of time, as grid is dynamic in nature [Shaik Naseera and K.V.Madhu Murthy 2010]. It is inefficient to select an appropriate cluster (resource) that often leaves the grid as the tasks may have to be rescheduled several times. In such cases, the overhead of load balancing strategies will be cost intensive. Therefore, selection of trustworthy resources is essential. Thus it is clear that trust computations may be needed, but at the same time it involves additional burden. It is also evident that simultaneous access to data is also probable.

## 3.0 Data warehousing and Learning Grid Framework

The proposed framework consists of two tiers (Figure 1.0) with two servers for each one respectively: i. The database server and ii. Online Analytical Processing Server (OLAP). The database server's job is to feed data (mining words supplied by the e-users) into the grid tasks and extract e-content (summary data) from the warehouse and supply them to the e-users. For want of space, the frontend query tools and the design of warehouse operations are not presented in this paper. As e-learning system should simultaneously engage content developers also apart from commentators/evaluators, the OLAP server operates simultaneously on the warehouse along with database server; but the former creates metadata for the warehouse. It can also create reports both for developers as well as for the grid tasks. The tasks for the grid might therefore run into several thousand for thousands of e-users.

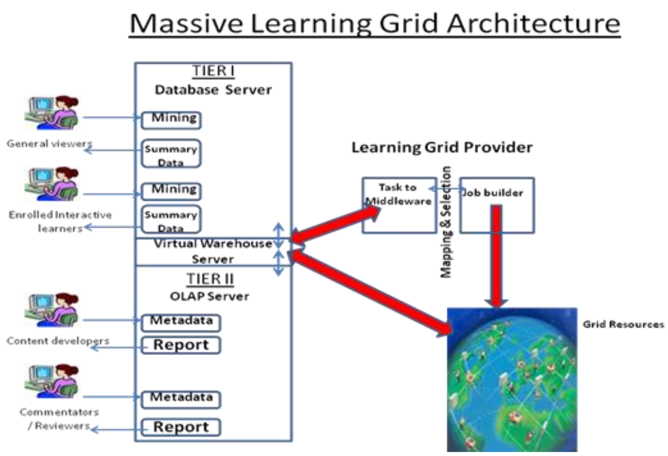



**Figure 1.0 Virtual Warehouse and Learning Grid Schema**

The grid provider's middleware takes these tasks that were supplied by the virtual warehouse's pre processor as input and then map them along with the path information (addresses) of the users. The middleware then maps these data to its job builder (right side in the Figure 1.0). The job builder sends these tasks and addresses to several clusters (participating in the grid) of the world, with instructions to directly send the processed tasks to different paths (users) supplied by the job builder. The entire procedure is conducted through GridSim 5.0 through the pre processor that was designed by the authors. The authors don't claim any originality in the architecture of this particular grid compartment, as it is a third party developed package (GridWay System Administrator's Guide – Globus - 2009).

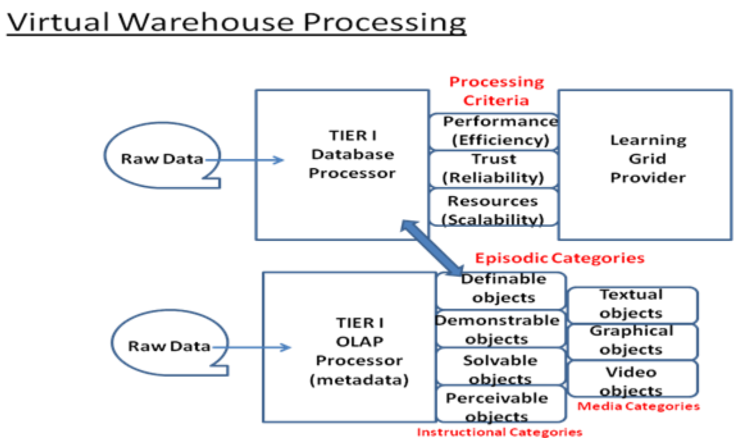
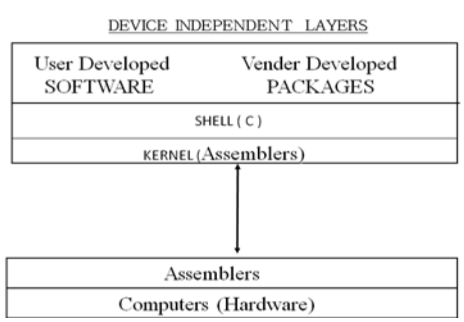



**Figure 2.0 Processes on Virtual Warehouse for Learning Grid Inputs**

One of the major novelties demonstrated by this paper is the suggestion on splitting up of long integrated single lot of e-content into pedagogically convenient short objects that need to be designed through proven instructional models [Merrill, M.D, 2002]. These objects are represented in the forms of i. 'Definable': short form that contains wholly and well defined instructional notes (sample shown in Figure 4.0); ii. 'Demonstrable': elaborated descriptions on the operations and procedures of any subject/topic content with/without figures (sample shown in Figure 3.0); iii. 'Solvable': numerical/logical examples of problems of related to the e-subject; and iv. 'Perceivable': conceptual content with/without figures (sample shown in Figure 5.0). Yet again the paper also suggests to split up the whole e-content into media wise parts in separate objects namely i. Textual; ii. Graphical; and iii. Animation/videos. The idea behind these suggestions is to experimentally prove the reduction of storage space and reduce the un-necessary computation (process) time of the grid for manipulation of data, as the grid tasks would naturally become small. Besides, all simultaneous e-users may not require all the data of the entire integrated single lot e-content all the time. The above procedure is reflected clearly in the Tier II of the framework (Figure 2.0). It is clarified here that the adopted instructional principles involved in designing such e-content are not elaborated in this paper, but they are supported by authorized research documents.



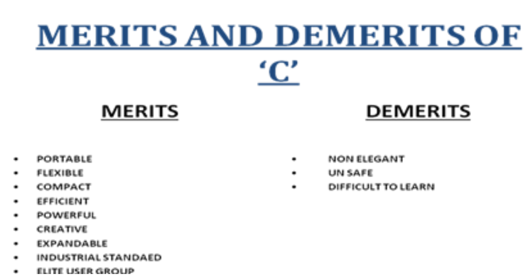




**Figure 3.0 Graphical / Demonstrable Object** **Figure 4.0 Textual / Definable Object**

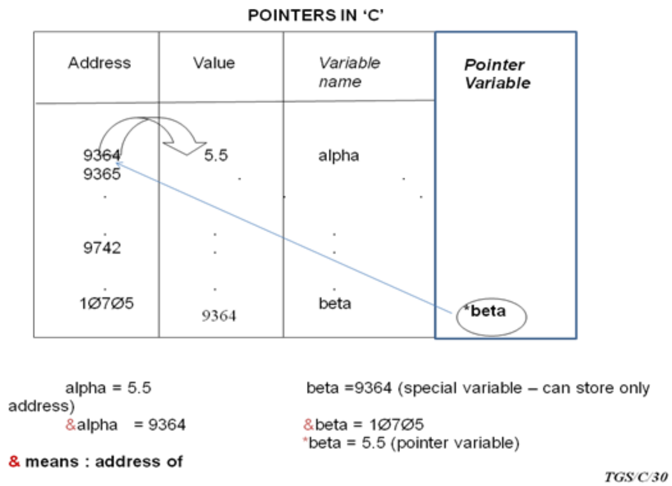



**Figure 5.0 Example of a Graphic Animation / Perceivable Object**

## 4.0 Object Data Samples and Experimental setup

The samples for the proposed experiments contain e-content objects that are in split up form of an integrated lot (both are used in the experiments). The file size of the 'definable' instruction object in storage is 188KB; and its size in memory is 186.6KB; processing time by the GridSim 5.0 would be around 2300 ms (excluding user retention time). Size of the 'demonstrable' instruction object in storage is 654KB; and its size in memory is 651KB; processing time by the GridSim 5.0 would be around 14300 ms. Size of the 'solvable' instruction object in storage is 362KB; and its size in memory is 660.7KB; processing time by the GridSim 5.0 is around 8600 ms. Size of 'perceivable' instruction object in storage is 160KB; and its size in memory is 158.8KB; processing time by the GridSim 5.0 would be around 1800 ms. Size of the same content integrated into a single document is about 1346KB; size the same in memory is 1340KB; processing time by the GridSim 5.0 would be around 28350 ms (excluding user retention time). With authorized research support [Kaladevi 2013], the average computational ratio of Definable : Demonstrable : Solvable : Perceivable has been empirically worked out to be about: 1.00 : 4.00 : 3.00 : 0.75 (i.e 11.5% : 46% : 34.5% : 8%), which is more or less matching with size and computational time. In a similar empirical study on media categories with authorized research support [Jagadeesan, B 2014], the average empirical computational ratio of Textual: Graphical: Animation is about 1.00 : 1.10 : 1.30 (i.e 28% : 33% : 39%) respectively.

These assumptions of empirical findings are specifically suited for the particular case study demonstrated by this paper, and any attempt made by other researchers should be based on their own case study findings, as instructions and learning are subjective. In the above study, the output from GridSim 5.0 that provided the total resource processing time in ms includes the following: process request time + process response time + process creation time + time taken to save processed tasks to host [Selvi, 2012]. The time taken for retention of the instructional content by the users is not considered. With these samples and inputs for the proposed experimental setup are summed up in Table 1.0.

| Specification | Trials & range |
|---|---|
| Tasks for massive users (Experiment 1). | 10, 50, 100, 200 and 400 |
| Clusters (resources) requested for creation of tasks for massive users (Experiment 2). | 100 – 1000 in steps of 100 |
| Massive clusters that would be grouped into resources. | Varies; and will be decided through first experiments |
| Parameters for selection of clusters (Experiment 1) | Trust; performance; redundancy removal |

**Table 1.0 Experimental Setup for Grid Resources**

## 6.0 Conclusion

It is clearly demonstrated that splitting up of integrated modules into objects (tasks) according to instructional and media categories will save both data storage as well as grid computational time substantially. The redundancy rate is decreased when Poison's probability values are adopted on trusted grid clusters. It is demonstrated that harmonization of efficiency and storage optimization can be achieved in massive learning grids through virtual warehouse approach.

## References

[1] Ng Wai Keat, Ang Tan Fong, Ling Teck Chaw and Liew Chee Sun "Scheduling Framework for Bandwidth-Aware Job Grouping-Based Scheduling in Grid computing", Malaysian Journal of Computer Science, Vol.19 No.2, pp. 117-126, 2006.

[2] Thanasis G. Papaioannou and George D. Stamoulis, " Reputation based estimation of individual performance in collaborative and competitive grids", Future Generation Computer Systems, Vol.26, Issue 8, pp. 1327-1335, 2010.

[3] Rajkumar Buyya, David Abramson, Jonathan Giddy and Heinz Stockinger, "Economic models for resource management and scheduling in Grid Computing", Concurrency and Computation: Practice and Experience, Vol. 14, pp. 1507-1542, DOI: 10.1002/cpe.690. 2002.

[4] Yuanynan Zhang, Wei Sun and Yasushi Inoguchi, "CPU Load Predictions on the Computational Grids", Proceedings of Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID 2006). Singapore, May16-19.DOI.ieeecomputersociety.org/10.1109/CCGRID.2006.27, 2006.

[5] Deepti Malhotra and Devanand, "Framework for Job Scheduling in Grid Environment", International Journal of Computer Applications, Volume 38, Number 7, pp. 42-44, DOI- 10.5120/4703-6861,2012.

[6] Shaik Naseera, and Madhu Murthy K.V, "Trust Based Load Distribution in a Data Grid Environment", International Journal on Computer Science and Engineering, Vol. 02, No.01S, pp. 24-29, 2010.

[7] Wen-Chung Shih, Chao-Tung Yang and Shian-Shyong Tseng, (2009), “Ontology-based Content Organization and Retrieval for SCORM-compliant Teaching Materials in Data Grids”, Future Generation Computer Systems, No. 25, pp. 687-694.

[8] Hsin-Chuan Ho, Chao-Tung Yang and Chi-Chung Chang, (2004), “Building and E-Learning Platform by Access Grid and Data Grid Technologies”, Proceedings of IEEE International Conference on e-Technology, e-Commerce and e-Service, Taipei, Taiwan, 2004, pp:452-455.

[9] Merrill M.D., (2002). “First Principles of Instruction”, Englewood Cliffs, NJ: Educational Technology Publications.

[10] Ani Brown Mary, N & Saravanan K, 2013, “Performance Factors of Cloud Computing Data Centres Using M/G/1/GD Model Queuing Systems”, International Journal of Grid Computing & Applications (IJGCA) Vol. 4, No. 1, March 2013.

[11] Selvi, K, 2012, “An Enhanced Gridway Metascheduler for Hetrogeneous Grid Using Trust and Reputation”, Ph.D dissertation, Anna University, Chennai, India, 2012.

[12] Kaladevi, 2013, “Load Balance Optimization in Learning Grids Through Semantic Matching for E-Contents”, Ph.D dissertation, Anna University, Chennai, India, 2013.

[13] Jagadeesan, B, 2014, “Domain Dependent Vertical Scalability for Development Efforts on E-Contents”, Ph.D dissertation, Anna University, Chennai, India, 2014.