

# Efficient Data Transformation in Multicasting Cluster Network in Grid Environment

Govindan Ganesan<sup>1</sup>  
Associate Professor,  
Department of Computer Science and  
Engineering,  
K.L.N.College of Information Technology,  
Madurai, Tamilnadu, India.

J. Suganthi<sup>2</sup>  
Professor, Department of Computer Science  
and Engineering,  
Hindusthan College of Engineering and  
Technology,  
Coimbatore, Tamilnadu, India.

## ABSTRACT

In the dawn of modern internet technology, Grid computing plays a vital role in efficient way of resources allocation merely focusing in self organization of resources. These ways of organizing Grid lets helps to increase the efficiency of Grid mechanism and further to incorporate a standalone architecture which works efficient in it terms. The working mechanism of Grids targets on parallel transaction ensures Multi transactions. In this paper we have proposed a new strategy called cluster mechanism which works efficient on multicasting. The Purpose of focusing multicasting is to implement self organization of resources and to implement efficient mechanism of resource allocation. In an extent, this paper addresses the need of utility computing since we have targeted on resource utility. This proposed method is well adapted on heterogeneous clusters which surpass multicasting. To add up with we are proposing a new model named as Cohesive Model (Grid Cohesive Model) which does multicasting based on sender and receiver initiative for cluster mechanism.

## Keywords

Cohesive grid, Dynamic Gridlet, Grid resource management, Cluster Organization

## 1. INTRODUCTION

A computational grid works as a network (Grid Network) to share the data based on the issued problem stack. Usually for solving a single problem more number of computer resources was incorporated to balance the load issued to the server. The server, decides on the number of nodes depends on the load. Once the data is been shared the Processing Element (PE) works independent or parallel with other PE to complete the given task. The number of PE were decided depends upon the job or task provided to the grid network.

In order to describe how the tasks is been allocated to PE, we have to focus on the Load of issued task allocated to grid. Now the constraints were focused on two main issues

- i) How the Load is balanced?
- ii) How the resource is well utilized?

### 1.1. Load Balancing

The Load is the amount of data given to a grid network for efficient and effective utilization of resources. The resource utilization addresses the resource management [1] which focuses of system wide performance. The Performance measure in grid depends upon the number of PE available and number of PE allocated for a task. Expected question to be answered to allocate the PE for a task, is based on Grid Network Model which is our proposed one.

### 1.2. Resource Management

The task of resource management is the key strategy for allocating the PE for its task. In order to allocated resource based on utility the factors have to be consider are Grid let Organization, Location, Number of Grid lets to be allocated. The commodities mentioned here in terms of resource allocation and its services were well managed by the proposed model. To explore the concept of resource management along with the factors mentioned above, which further focused on self organized resources management. The impact of self organized resource management, address the process of utility computing. The organization of utility computing works on the strategy of task sharing based on scheduling. To conclude the part of resource management further results in utility computing.

Apart from the two significant challenges in grid mentioned above, there are few other challenges explained further in this context.

**Heterogeneity:** It exists in both of computational and networks resources.

In terms of Heterogeneity it mainly focus on the bandwidth and communication protocols, further it emphasis on processors, resource capabilities memory size and covers operating systems, file systems, and all these result in cluster management.

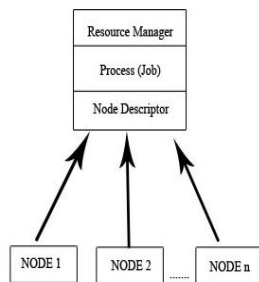
**Autonomy:** The sake of autonomy (self recognizing task) the grid resources, is viewed as an autonomous computational entity. It usually has its own scheduling policy, which complicates the task allocation problem. This autonomy is not alone required for grid performance, in spite it result in scheduling decision, which are independent of other sites according to its own performances.

**Resource selection and computation-data separation:** In traditional systems, executable codes of applications and input/output data are usually in the same site, or the input sources and output destinations are determined before the submission of an application. Thus the cost for data staging can be neglected or the cost is a constant determined before execution and load balancing algorithms need not consider it. But in a Grid the computation sites of an application are usually selected by the Grid scheduler according to resource status and some performance criterion. Additionally, the communication bandwidth of the underlying network is limited and shared by a host of background loads, so the communication cost cannot be neglected.

This situation brings about the computation-data separation problem: the advantage brought by selecting a computational resource that can provide low computational cost may be neutralized by its high access cost to the storage site.

## 2. GRID MODEL FOR UTILITY PROCESSING

The proposed model works for resource management using resource manager. The task of resource manager deals with resource sharing and its allocation. To proceed with its task, each grid node was allocated with its resources and checks whether all the grid nodes were equally balanced with the task. The process of doing this kind of impact is to ensure the grid works consistent and improve efficiently in terms of tasks allocation. In [2] focus on the model the grid model discussed in this work were elaborated in the figure 1. The task of the process scheduler is to allocate the task depends upon the number of nodes allocated to the grid source. The need for referring grid as grid source, since the task is centralized based on the grid nodes. To explain with the figure 1, it is assembled in 3 order of stack. The task of resource manager was discussed earlier along with utility computing.



**Figure 1: Grid Model for Utility Processing**

Grid computing makes use of software to distribute the task into number of processing element to as many as several thousand computers. Grid computing can be thought of as distributed and large-scale cluster computing.

Grid computing (along with Grid lets) seems to be a promising trend for three reasons:

- (1) Reducing time complexity by using the amount of Computer resources
- (2) Its' the way to reduce the complexity arise during problem solving phase.
- (3) Resource sharing, since it suggests that the resources of many computers can be cooperatively and perhaps synergistically harnessed and managed as a collaboration toward a common objective.

In [2] it focuses on utility sharing based on process scheduling.

The proposed model solves and comes out the hazards which work indifferent during the task in execution. The first reason were overcome using Resource manager and the second, third reason were overcome using Process scheduler and node descriptor. We believe that the proposed model works efficient for all the reason, proven to be the best aspect and practiced for a grid processor.

## 3. MULTICAST

The purpose of multicasting is to flourish the distributed mechanism of grid resource. In reference with the Fig 1, the grid processor controls the 'n' nodes based on the resource available. In this process of resource sharing, process scheduler plays a vital role in segregating the resource based on the grid processor. As a whole, the multicasting will be left unbalance without the initiation made by sender and receiver approach. To focus on this approaches based on the task evolved by the grid processor.

### 3.1. Sender-initiated approach

All the nodes were arranged in the aspect of spanning tree for transmitting (sending) the data. The important aspect of this kind an approach is to simulate the data transmission in an easy way. This Spanning tree paves the way for the network setup, further focus on data transmission. On the other hand it will decide upon which tree structure to be used for better transmission. This sender initiated approach works on these measures and provides efficient data transmission and balance one end (sender) of data transmission. This fact proven to be efficient since the sender initiated perfect so as the data transmission. The most often adapted techniques of modern day provide measures [3] on optimal data transmission [4], [5] of multicasting.

To discuss further with the disadvantage of this approach emphasis on implementing ubiquitously for monitoring system, and to provide accurate and stable multicast operations.

The bandwidth of network monitoring system will provide a tough ask for efficient data translation.

To avoid these parametric measures, we implement balance multicasting [6] to solve this scenario.

### 3.2. Receiver-initiated approach

The balanced multicasting based on peer-to-peer networking [7], [8] were the application were arranged in random mesh network and simulate dynamic data request with the neighboring nodes. This way, nodes were arranged dynamically and transmit the data in the mess network. To initiate this task it will work based on request-reply interaction between nodes automatically and adapt efficiently based on the bandwidth and it further reflects on heterogeneity.

### 3.3. Clustering nodes

The impact of cluster is to simulate the peer-to-peer system performance which uses biased neighbor selection based on cluster groups. The groups were selected based on threshold value using

$$\left[ r(x, C_x) \cdot \frac{|C_y|}{|C_x|} \right] \leq \left[ r(x, C_x + 1) \cdot \frac{|C_y|}{|C_x|} \right] \quad [1]$$

Node x then selects max ( $|C_x|, N$ ) nodes in  $C_x$  uniformly at random as its global peers and initiates connections with them.

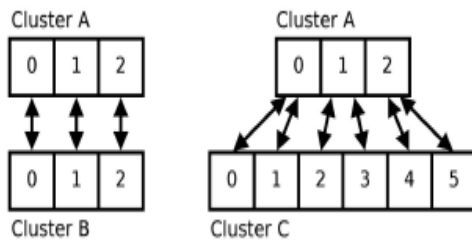


Fig 2.Example for peer selection

To initiate the collection of peers were further given by the formula

$$\left[ r(n, C) \cdot \frac{P}{|C|} \right] \leq i \leq \left[ r(n, C + 1) \cdot \frac{P}{|C|} \right] \quad [2]$$

The nodes' initiation can be changed at runtime, which stoles the work from the global peers and try to initiate more peers to work further. The work is organized in such a way that at least one node collective remotely. This will be initiated in an every cluster.

## 4. MULTICASTING AND LOAD BALANCING

In this work, Algorithm incorporates the work of Robbers [Multicasting algorithm] and combined with approach of Load Balancing Paradigm for efficient selection of Nodes in the Clusters to make the transmission.

Step 1: Initiate the Process with the 'n' number of nodes

Step 2: Select the Nodes for Multicasting

Step 3: Make Clusters

Step 3: Implement Multicast algorithm for the clusters

Step 4: Selected node then implemented with Load Balancing strategy to make Receiver Initiative in that sense which emphasis on the number of nodes considered for the transmission.

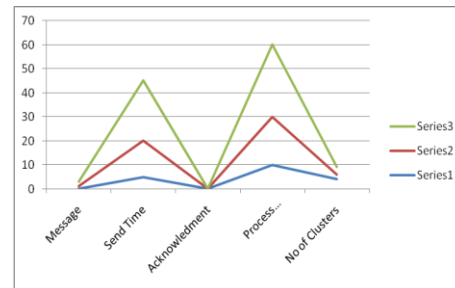
## 5. CONCLUSION AND FUTURE ENHANCEMENT

This project currently supports one-to-many application (Broadcasting). It can be extended to support many-many application that is all participants send as well as receive data. More precisely, in the many-many case, each participant is equally likely to act as a sender for a randomly chosen packet. The future impact of this concept deals with Data Services to be incorporated with Multicasting.

This can be extended in cloud computing to implement SAS (Software as Services).

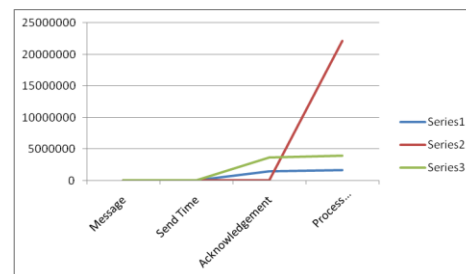
### Results

The results were evaluated with GridSim 5.0 and its latest updated version.



**Result 1**

The results shows the graph based on send time, acknowledgment, and its process completion time. The Maximum data used are 1GB, 2 GB, and 3 GB with 16 peers approximatly.



**Result 2**

The proposed results shows the graph based on send time, acknowledgment, and its process completion time. The Maximum clusters used are 500, 350, 350 with total of 1152 processing element.

## **About Author(s)**

### **Author 1**

Govindan Ganesan received B.E. degree in Computer Science and Engineering from Madurai Kamaraj University, Madurai in 1994, M.S. degree in Information Technology from Bharathidasan University, Trichy in 2002, and M.E.degree in Computer Science and Engineering from Anna University, Chennai in 2006. He has three years experience in Industry and fourteen years teaching experience in engineering colleges. He has been an Associate Professor in the Department of Computer Science and Engineering, K.L.N.College of Information Technology, Madurai, Tamilnadu, India.

### **Author 2**

Dr. J. Suganthi received B.E degree in Computer Science and Engineering from Madurai Kamaraj University, Madurai, M.E Computer Science and Engineering from Bharathiar University, Coimbatore, and PhD from Anna University, Chennai. She has eight years of experience in industry and eleven years of teaching experience in engineering colleges. She has been a Professor in the Department of Computer Science and Engineering, Hindusthan College of Engineering and Technology, Coimbatore, Tamilnadu, India.

## **6. REFERENCES**

- [1] I. Foster, C. Kesselman (Eds.), *The Grid (2nd Edition): Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 2004.
- [2] Hong-qing Gao, Ying Xing, *An Economic Model for Multi-Resource Transaction in Grid Environment*, 2009.
- [3] D. Kosti\_c, A. Rodriguez, J. Albrecht, and A. Vahdat, "Bullet: High Bandwidth Data Dissemination Using an Overlay Mesh," *Proc.19th ACM Symp. Operating System Principles (SOSP-19)*, Oct. 2003.
- [4] B. Cohen, "Incentives Build Robustness in BitTorrent," *Proc. First Workshop Economics of Peer-to-Peer Systems*, June 2003.
- [5] V. Pai, K. Kumar, K. Tamilmani, V. Sambamurthy, and A. Mohr, "Chainsaw: Eliminating Trees from Overlay Multicast," *Proc.Fourth Int'l Workshop Peer-to-Peer Systems (IPTPS '05)*, Feb. 2005.
- [6] J. Pouwelse, P. Garbacki, J.W.A. Bakker, J. Yang, A. Iosup, D. Epema, M. Reinders, M. van Steen, and H. Sips, "Tribler: A Social- Based Peer-to-Peer System," *Proc. Fifth Int'l Workshop Peer-to-Peer Systems (IPTPS '06)*, Feb. 2006.
- [7] M. den Burger and T. Kielmann, "MOB: Zero-Configuration High-Throughput Multicasting for Grid Applications," *Proc. 16<sup>th</sup> IEEE Int'l Symp. High Performance Distributed Computing (HPDC '07)*, June 2007
- [8] C. Gkantsidis and P. Rodriguez, "Network Coding for Large Scale Content Distribution," *Proc. IEEE INFOCOM, Mar. 2005.of Grid Data Environments Workshop (GGF-10)*, Mar. 2004.
- [9] Mathijs den Burger and Thilo Kielmann, *Collective Receiver-Initiated Multicast for Grid Applications IEEE Computer Society IEEE transactions on parallel and distributed systems*, vol. 22, no. 2, february 2011.