Intelligent Agent Based Resource Sharing in Grid Computing

V.V. Srinivas[1] and V.V. Varadhan[2]

[1] Department of Computer Science and Engineering,
National Institute of Technology – Tiruchirappalli
srinivas15j1988@gmail.com
[2] Department of Information Technology,
Madras Institte of Technology - Chennai
varadhan3n90@gmail.com

Abstract. Most of the resource present in grid are underutilized these days. Therefore one of the most important issue is the best utilization of grid resource based on users request. The architecture of intelligent agent proposed to handle this issue consists of four main parts. We discuss the need and functionality of such an agent and propose a solution for resource sharing which satisfies problems faced by today's grid. A J2EE based solution is developed as a proof of concept for the proposed technique. This paper addresses issues such as resource discovery, performance, security and decentralized resource sharing which are of concern in current grid environment.

Keywords: grid, resource sharing, intelligent agent, decentralization.

1 Introduction

Grid computing is distributed, large-scale cluster computing, as well as a form of network-distributed parallel processing. Each computer present in grid has computational power and resources such as memory, printer etc., which are underutilized. In order to utilize resources and provide service to customers resource sharing was introduced. Resource sharing [1] provides access to a particular resource on a computer to be accessed by clients on grid. The need for resource sharing arises in case of complex mathematical modeling and simulations like the network simulation or simulation of automatic test pattern algorithms, virtual supercomputing or DNA mapping.

Resource sharing involves three main process namely: resource discovery, resource management and resource allocation. Resource discovery is finding resources available in grid. This problem is solved in tools such as Globus and Condor Matchmaker [2]. Resource management involves collecting resource. The challenge involved is finding the right quantity of resource [3].

1.1 Contribution

There are a number of ongoing research in the field of grid computing most of them trying to address some of the challenges faced in grid environment. This paper

V.V. Das, G. Thomas, and F. Lumban Gaol (Eds.): AIM 2011, CCIS 147, pp. 106–110, 2011.

addresses aspects like: resource discovery in grid, security, decentralized resource sharing, scheduling of resources and threshold based resource allocation.

The rest of this paper is organized as follows: Section 2 discusses ongoing research followed by section 3 discussing proposed architecture. Section 4 describes experimental results. In section 5, the paper deals with application and advantages followed by future research and conclusion.

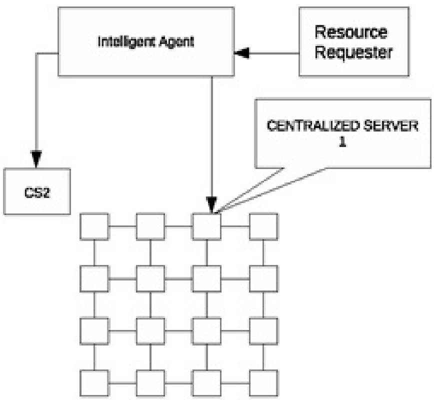

Fig. 1. Resource discovery

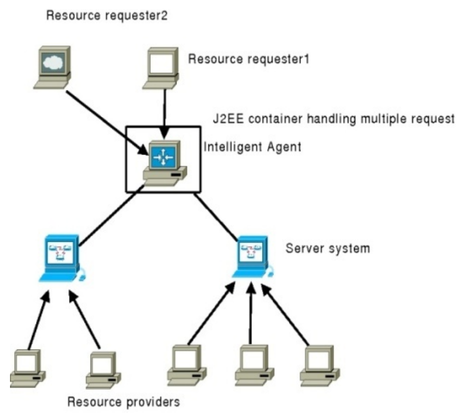

Fig. 2. Grid model of entire system

2 Ongoing Research

2.1 Globus Toolkit

Globus is an open source toolkit that is used for construction of grids. It provides access to resource present within the grid located in any geographical area [5]. Globus uses GSI (Grid Security Infrastructure), GRAM (Grid Resource Allocation and Management) and MDS (Monitoring and Discovery Services) for resource management and discovery. Globus provides a remote front end to multiple batch systems. Our paper tries to preserve all the advantages of Globus and eliminates all complex configuration and installation of number of tools.

2.2 Meta-broker Architecture

The meta-broker architecture focuses on how to allocate a particular resource present in some other network or grid to a user requesting for resource [6]. Previous works deals with MESS (Multi Engine Search Services) and ISS (Internet Search Service) based on CORBA [7] and meta-broker architecture for management of grid resources [8].

2.3 Negotiation Algorithms

This work mainly deals with negotiation protocols between the client and provider. The key focus is on contracts [9]. One important algorithm used is G-Negotiation algorithm. Our paper discusses a simple mechanism for secure communication.

3 Proposed Architecture

The entire work is split up into several modules which are discussed in detail. First is the ***server module*** in which client registers to provide resource. The client is provided with an address. The details registered include resource type, amount of resource and time duration when the resource would be available. ***Resource discovery*** [4] module is used to keep check on parameters such as processor time, print queue, system threads, disk queue length and cpu usage. The values are obtained from performance logs and alerts. The retrieved values are stored in .csv format. These obtained values

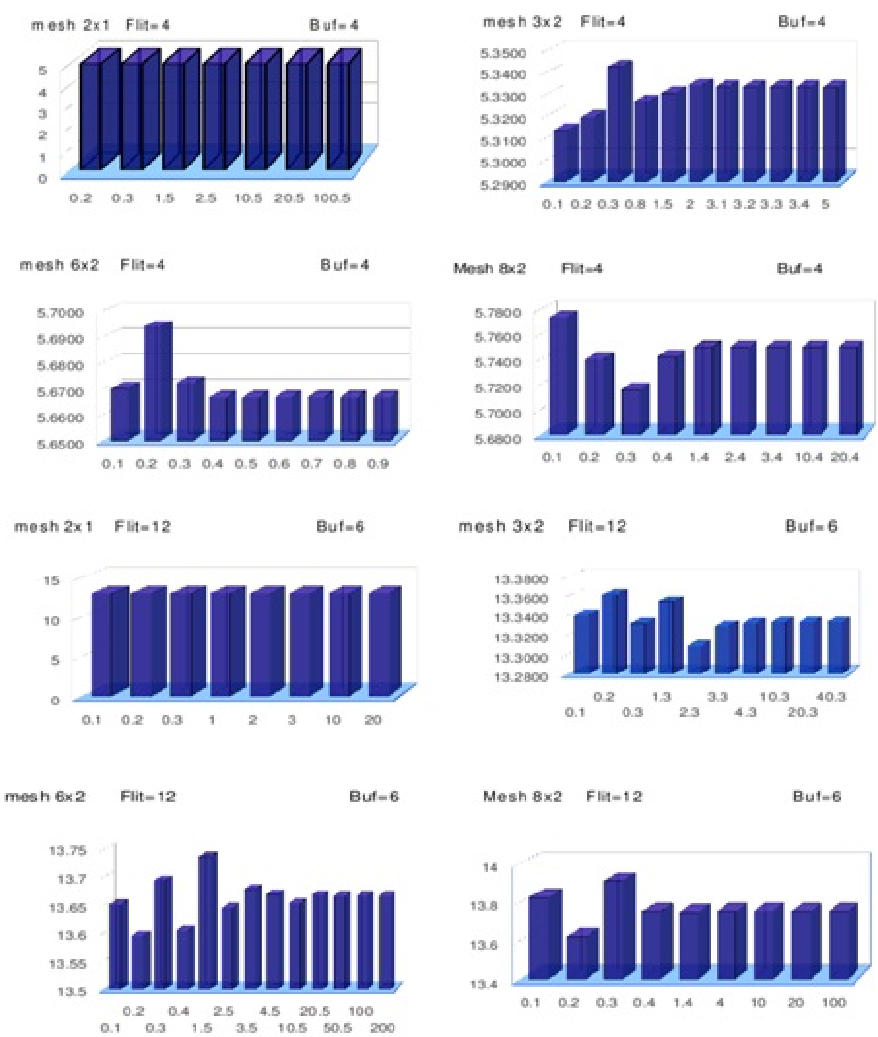

Fig. 3. Four program instances run on 2, 3, 6 and 8 nodes. X axis represents injection rate and Y axis represents latency. Buffer size are taken as 4 and 6.

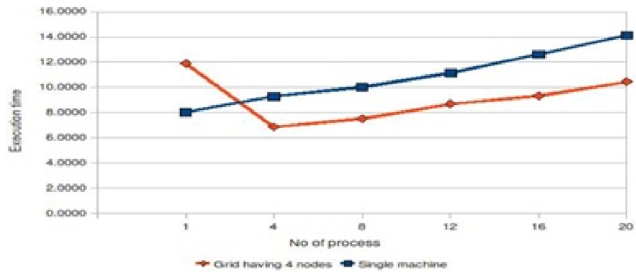

Fig. 4. Performance analysis in terms of execution time

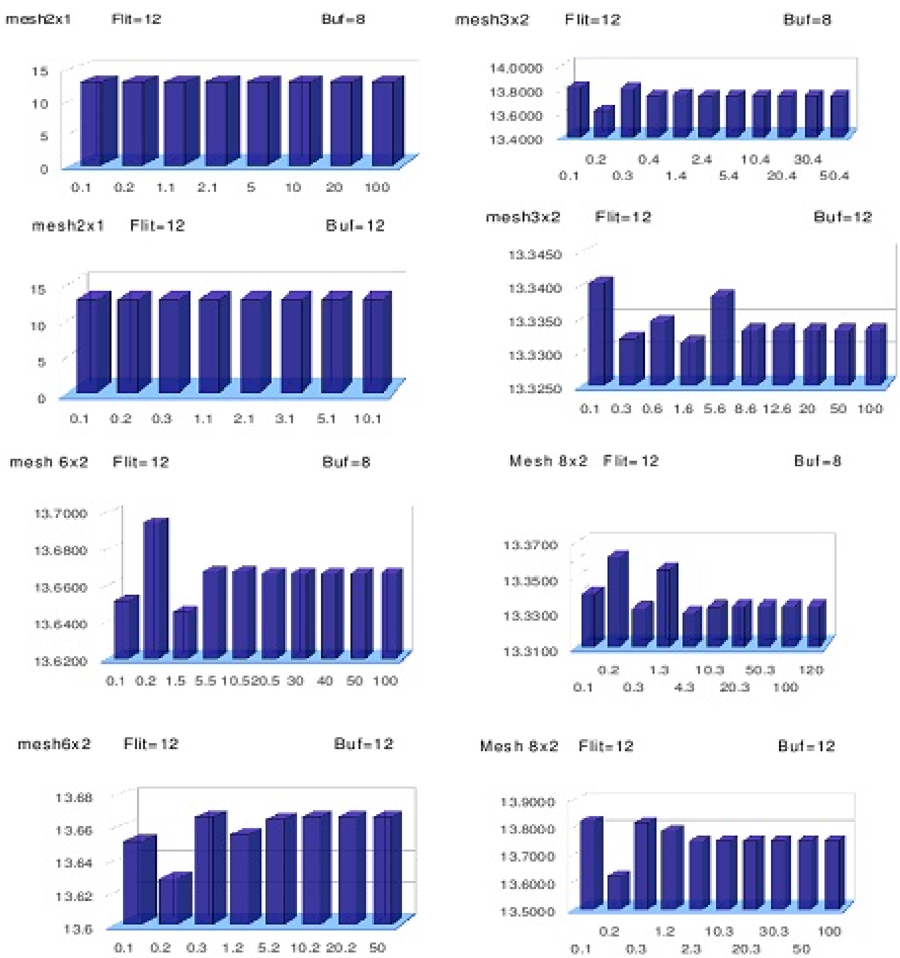

Fig. 5. Four program instances run on 2, 3, 6 and 8 nodes with buffer size are taken as 8 and 12

are compared with threshold values. The third is the ***intelligent agent module*** which reads the files obtained from the resource discovery module running on each machine. The resource files are set to refresh after t_k time. The resource requester requests the agent for a particular resource. The intelligent agent searches and retrieves the various resources it has from various clients. Once the right resource is obtained, a secret key is transmitted to the resource requester and provider. Along with the secret key, the resource requester receives the providers address and vice-versa. This ensures security.

4 Experimental Results

The performance was evaluated between the number of process and execution time. We took the number of resource providers in the grid to be 4 and the number of process was taken as multiples of 4. A graph was plotted for the process running on single system to the process split among the four systems and the resultant graph is shown in Fig. 4.

5 Conclusion

Our paper discusses a solution for resource sharing at the same time preserving features like security, authentication, resource discovery and decentralization. This paper defines a simple solution to implement intelligent agent in grid environment.

References

1. Cruz-Perez, F.A., Ortigoza-Guerrero, L.: Equal resource sharing allocation with QoS differentiation conversational services in wireless communication networks. IEEE Proceedings Communications, 150, 391–398 (2003)
2. De Smet, A.: Computer Science Department. University of Wisconsin Madison, http://www.cs.wisc.edu/condor
3. Li, Y., Wolf, L.: Adaptive Resource Management in Active Nodes. In: 8th IEEE International Symposium on Computer and Communication (2003)
4. Giovanni, A., Massimo, C., Italo, E., Maria, M., Silvia, M.: Resource and Service Discovery in the iGrid Information Service. In: Gervasi, O., Gavrilova, M.L., Kumar, V., Laganá, A., Lee, H.P., Mun, Y., Taniar, D., Tan, C.J.K. (eds.) ICCSA 2005. LNCS, vol. 3482, pp. 1–9. Springer, Heidelberg (2005)
5. Ian, F., Carl, F.: The Globus project: A Status Report. In: Heterogeneous Computing Workshop, pp. 4–18 (1998)
6. Kertesz, A., Kaeushk, P.: Meta-broker for future generation grids: A new approach for high level inter operable resource management. In: CoreGrid workshop (2007)
7. Yue-Shan, C., Hsin-Chun, H., Shyan-Ming, Y., Lo, W.: An agent based search engine based on Internet search service on CORBA. In: Proceedings of International Symposium on Distributed Objects and Applications, pp. 26–33 (1999)
8. Kertesz, A., Kacsuk, P.: Grid Interoperability Solutions in Grid Resource Management. Systems Journal 3, 131–141 (2009)
9. Antoine, P., Phelipp, W., Oliver, W., Wolfgang, Z., Dynamic, S.L.A.: negotiation based on WS agreement. In: Core Grid Technical Report TR-0082 (2007)
10. Karl, W.: The Management of Change among Loosely Coupled Elements. In: Making Sense of the Organization (1982)