Pankti Dharwa/ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 4, July-August 2012, pp.972-976

Light-Weight Communication System in Grid

Pankti Dharwa
(Department of Computer Science,Sobhasaria Engg. College, RTU)

ABSTRACT

The advantage of high-performance networks in conjunction with low-cost, powerful computational machines have made possible the development of a new set of technologies termed "computational grid". These technologies are making possible the creation of very large-scale distributed computing systems by interconnecting geographically distributed computational resources via very high-performance networks. One particularly difficult issue is that of utilizing fully available bandwidth while being in some sense fair to competing traffic flows. It has been widely demonstrated that TCP, the communication protocol of choice for most distributed application, often performs quite poorly in the emerging high-bandwidth high-delay network environments. This has led to significant research on the development to user-level applications that can circumvent some of the performance problems inherent in TCP. The goal of this paper is to develop Light-Weight Communication System in the Grid. The first initiative is to apply multi-casting features to this protocol & the algorithm for the same is developed. Then apply the algorithm on the Grid by using MPI (Message Passing Interface). Reduce the System call, which is not necessary by the Windows at all the time and which consumes more time in background processes.

Keywords – Alchemi Framework, Grid computing, Light Weight Communication, Message Passing Interface.

1. INTRODUCTION

The main objective of this paper " Light-weight communication system in Grid" for building an efficient, flexible, and scalable inter-grid communication. To make a Communication Light-Weight in Grid, because of the limited network bandwidth, high throughput, more reliability, by reducing the unnecessary System call and applying the Multi-Cast Protocol with MPI for the communication. Challenges in designing such infrastructure arise due to the distributed nature of the resources to be used, the distributed communities, the size of the data to be shared and the limited network bandwidth.

Grid computing is an interesting research area that integrates geographically-distributed computing resources into a single powerful system. Many applications can benefit from such integration. Examples are collaborative applications, remote visualization and the remote use of scientific instruments. Grid software supports such applications by addressing issues like resource allocation, fault tolerance, security, and heterogeneity. Parallel computing on geographically distributed resources, often called distributed super-computing, is one important class of grid computing applications.

2. RELATED WORK

Based on the existing grid security technologies, an authentication and access control framework at Virtual Organization (VO) level for group communication in grid environment. The implementation of prototype based on GridShib. GridShib, developed to solve cross-domain authentication and distributed authorization [1]. The system manages each part as a separate process (the single program, multipledata model) that communicates via messages [5].
The Multi-Link Translator and Display System provides three basic capabilities: to receive and display tactical data link information, to translate between tactical data link message sets, and to route the data link information between various physical interfaces [6]. User management model developed for the lightweight multilevel grid system GrOSD (Grid-aware Open Service Directory). Main purpose in making the design was the simplicity and scalability [8].

• Role-Based Access Control (RBAC), in group communication systems, there exist many kinds of roles, such as group creator, group controller, many kinds of application managers and application members. In which they used attribute-based approach [1].

• Dynamic scheduling system communication, monitoring module monitor the running of sub-tasks in the simulation process of the operation, it can get the CPU utilization rate, delay time and port traffic, etc, also can display in real-time. Dynamic scheduling module receives the monitoring results information,
and then under the dynamic algorithm migrate simulation task to achieve load balancing [3].

3. EXISTING METHODOLOGIES

A grid application is defined simply as an application that is to be executed on agrid and that consists of a number of grid threads. Grid applications and grid threads are exposed to the grid application developer via the object-oriented Alchemi .NET API[4].

3.1 Basic Architecture of Communication System

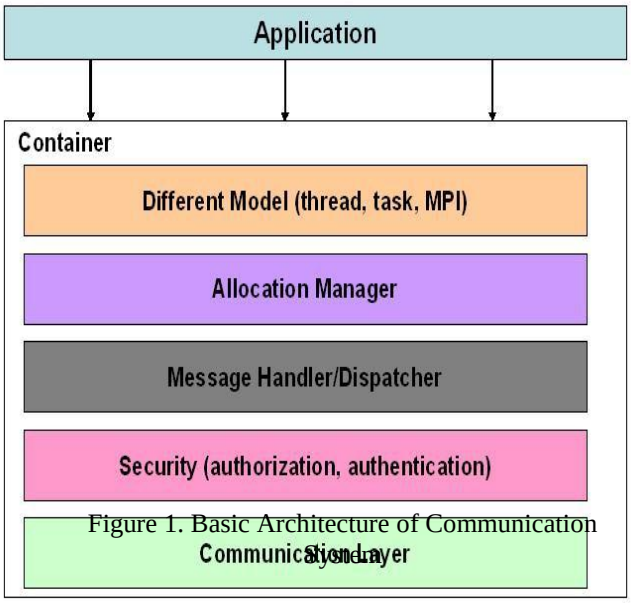

Figure 1. Basic Architecture of Communication System

The basic architecture of the communication system is as described below. The user application is send to the Grid head node(Manager), which has container in
which all layer for the communication system is described as the Application mode llike ,thread, task, etc. Then the Allocation Manager check the available resources and
then the Message Handler/or Dispatcher, send or receive the message for the communication. There is security mechanism like, Authentication,Authorization is used for the communication according to the gird protocol. And finally the Communication layer can communicate to the other Grid Execution node(Executor) [9].

3.2 Working Flow of Grids

The normal flow of the Alchemi grid is as described below. There is/are the user application or application client sends the application thread or the query message to the grid head node(Manager), which contains the services like message handler/dispatcher services, directory services, etc. Grid head node see the available resources, and send the query message or the application thread to other grid node. The grid executor or the execution node compute the application thread or the query message and give the output back to the grid head node(Manager), and that node collect the output of from all other grid execution node and combine it to gather as one and give it back to the application client or the user.

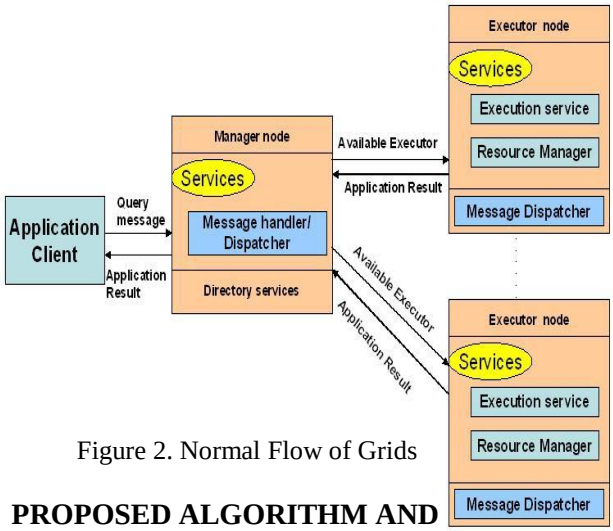

Figure 2. Normal Flow of Grids

4. PROPOSED ALGORITHM AND METHODS

4.1 Optimization of Light-Weight Communication

Optimize communication services is essentially any type of semantics associated with communication beyond the simple, reliable unicast, transfer of data from point A to point B, or even the multicast of data from one to many [7].

There are several ways to Optimize the Light-Weight Communication.some ways are,

- Using multiple networks in parallel
- Simplifying LAN-wide host naming
- Simplifying communication protocol
- Avoiding temporary buffering of messages
- Pipelined communication path
- Avoid system calls for communication

- Lightweight system calls for communication
- Providing very low-level mechanisms

4.2 Multi-Cast Protocol With MPI

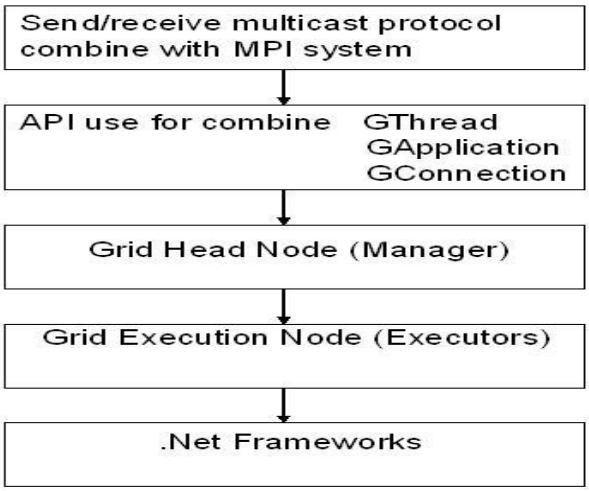

Figure 3. MPI using Multi-cast Protocol

The Multi-cast protocol for send/receive which combine with the MPI(Message Passing Interface)algorithm, it combine with the windows gird by using API- GThread, GApplication, GConnection to the Grid head node. Grid head have storageor the schedular which schedule the application.and give it to the grid other node to execute and take the output or the values back from it. This all process is done on the windows .Net framework.

4.3 Message Passing Implementation

Message Passing is a communication paradigm to develop parallel and distributed applications that require inter-process communication. The Message Passing Interface (MPI)computing models are specifications for parallel that have numerous different implementations [17]. Below are the main functions i will implement for MPI.

1. MPI Init: message passing initialization method.

2. MPI Finalize: message passing finalization method.

3. MPI Comm size: returns the total number of processes.

4. MPI Comm rank: returns the process identifier.

5. MPI Send: sends data to a remote node.

6. MPI Recv: reads the local receiving buffer.

7. MPI Bcast/Mcast: sends data to multiple processes.

4.3.1 Basic Concepts of Data Transfer

- All MPI communication is based on communicator which contains a context and group.
- Contexts can be defined as safe communication space for message passing and also it allows the different libraries to co-exist.
- Group is set of processes.
- Processes can be identify by unique rank in group.

4.3.2 Message passing is required for:

- Data transfer plus synchronization
- Requires cooperation of sender and receiver
- Cooperation not always apparent in code

5. IMPLEMENTATION

Alchemi is a .NET based grid computing framework that provides runtime machinery and programming environment required to construct the desktop grid and developed the grid application. It allows us flexible application composition by supporting the object-oriented grid programming model. Cross platform support is provided by Web-services and flexible execution model supports dedicated execution by grid nodes [16].

5.1 Alchemi Framework

The aim of Alchemi grid computing framework is not to develop the grid software as easy as possible but flexible, scalable, reliable, and extensible. The key features of the Alchemi are,

- Internet based cluster computing for desktop computer without a shared file system.
- Dedicated execution by cluster and individual nodes.
- Object-oriented grid thread programming model.

5.2 Grid Thread Programming

The two central classes in alchemi grid API are GThread and GApplication that represent the grid thread and grid application respectively. GThread is used for the code to be executed remotely and GApplication is for locally executed code [16].

5.3 About the Bandwidth Monitor

Bandwidth Monitor tracks traffic of all network connections and displays real-time download and upload speeds in graphical and numerical forms (refer to screen shot below). The software logs traffic of all network connections and provides daily, weekly and monthly traffic reports. Bandwidth Monitor can also display download and up-load speeds of multiple network connections on a computer at a time. For example, you can monitor bandwidth of multiple network cards on your computer at the same time. Bandwidth Monitor also offers useful built-in utilities: speeds stopwatch, transfer rates recorder, and bandwidth usage notification.

6. EXPERIMENTALRESULTS AND ANALYSIS

The result, have been obtain using the "PERFWIZ" simulator, and "Bandwidth Monitor" are shown in the tables.

Table 1: Various Performance Parameters

Executor	*Total execution Time(sec)*	*Avg Bytes sent(kbps)*	*Avg Bytes receive(kbps)*
1	43.76	1.05	0.32
2	22.23	1.60	0.41
3	15.02	1.27	0.44
4	15.01	1.72	0.54

6.1.1 Various Grid Nodes and Execution Time

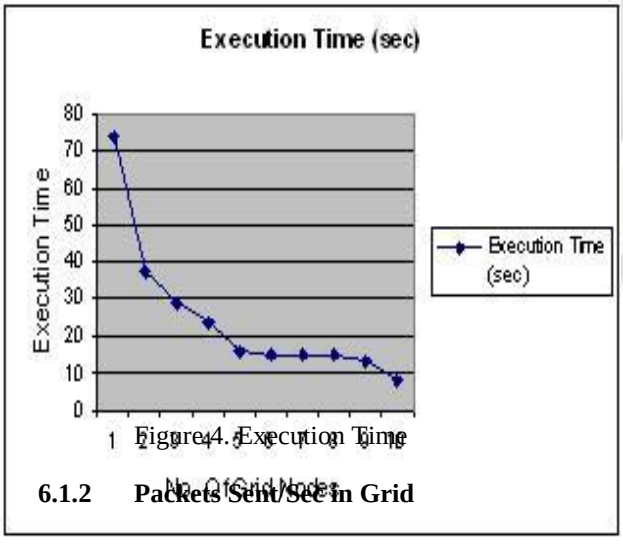

Figure 4. Execution Time

6.1.2 Packets Sent/Sec in Grid

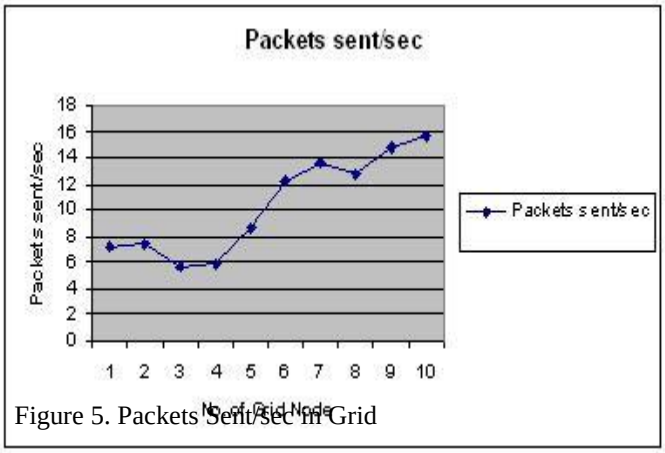

Figure 5. Packets Sent/sec in Grid

6.1.3 Packets Receive/sec in Grid

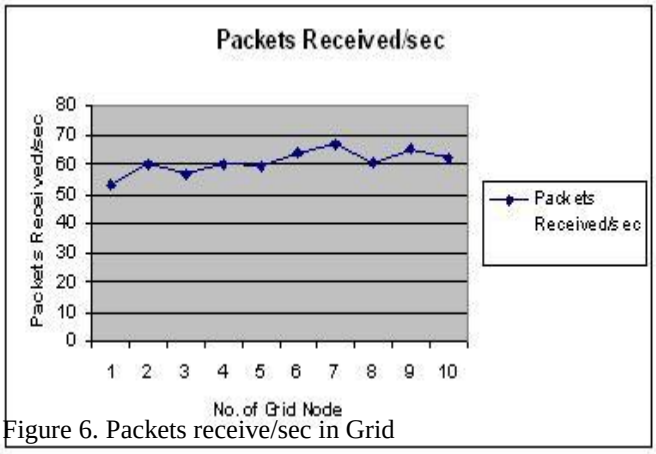

Figure 6. Packets receive/sec in Grid

7. RESULT ANALYSIS

From the various kind of graphs, it shows the different parameter for measuring the performance of the grid. In first Graph, shows that the execution time of the applica-tion or the job is decrease according to the increasing the no of grid execution node. In second and third graph, parameters like Packet sent/receive per second. From that, the difference of the execution time and other parameter increase/decrease accordingly with the no. of grid execution node(s). The Parameter's value increasing with the no of grid nodes are, packe tsent/sec, the no of grid nodes are, application's execution time,

8. CONCLUSION

Light-Weight Communication helps in improving the performance parameters like execution time, network bandwidth, memory usage, data sent/receive, etc. Since existing TCP/IP protocol when used in grid, does not provide, priority based routing, rate and burst control, multi-casting, etc. An attempt is made to overcome this and build the Light-Weight Communication Protocol.

From the various result, it can be concluded that using "Light-Weight Communication System in Grid", application's execution time can be reduced up-to 15-25% of the normal execution time. Other identified parameters are volume of data transfer and the network protocol used in Grid, specifically for the multi-casting. The other parameters like network bandwidth,
network topology, the load on the network are very important for a grid to become as "Light-Weight Grids".

9. ACKNOWLEDGEMENTS

I am Pankti Dharwa. I am pursuing MTech (Computer Science) from Sobhasaria Engineering College, Sikar. Rajasthan Technological University. My research area is Grid Computing. I am very thankful to faculty members and to my friends for their support. And at this moment, I would like to express my appreciation to my family members for their unlimited encouragement and support.

REFERENCES

[1] Deqing Zou, Laurence T. Yang, Weizhong Qiang, Xueguang Chen, Zongfen Han, "An Authentication and Access Control Framework for Group Communication Systems in Grid Environment", *21st International Conference on Advanced Networking and Applications(AINA'07) 0-7695-2846-5/07 2007 IEEE.*

[2] I. Foster, C. Kesselman, L. Pearlman, etal.,"The Community Authorization Service: Status and Future", In Proceedings of Computing in High Energy Physics 03 (CHEP '03), 2003.

[3] Fu Yanfang, Liu Bailin,"Research of Communication Technology for Simulation Grid System", *978-1-4244-2800-7/09/ 2009 IEEE.*

[4] R. Buyya, M. Murshed, "GridSim:A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing", *Concurrency and Computation: Practice and Experience, vol.14,pp.1175-1220,November 2002.*

[5] Geoffrey Fox, "MESSAGE PASSING: FROM PARALLEL COMPUTING TO THE GRID" *1521-9615/02/ 2002 IEEE.*

[6] H. T. Ho, The MITRE Corporation, Bedford, MA, "Windows NT Multi-threaded Design in Multi-Link Translator and Display system", *0-7803-5749-3/99/1999 IEEE.*

[7] "High Performance Cluster Computing, Architectures and Systems",Rajkumar Buyya.

[8] Lazar Kirchev, Minko Blyantov, Vasil Georgiev, Kiril Boyanov, Maciej Malawski, Marian Bubak, Stavros Isaiadis, Vladimir Getov, "USER PROFILING FOR LIGHTWEIGHT GRIDS".

[9] George Clapp, Joel W. Gannett, Ronald Skoog,"Requirements and Design of a Dynamic Grid Networking Layer", *0-7803-8430-W04F 2004 IEEE.*

[10] Craig A. Lee, Eric Coe, B. Scott Michel, James Stepanek,Ignacio Solisy, J. Matt Clark, Brooks Davis,"Using Topology-Aware Communication Services in Grid Environments" *Proceedings of the 3rd IEEE/ACM International Symposiumon Cluster Computing and the Grid (CCGRID.03),0-7695-1919-9/03 2003 IEEE.*

[11] Jack Dongarra, Thomas Sterling, Horst Simon, and Erich Strohmaier,"HIGH-PERFORMANCE COMPUTING: CLUSTERS, CONSTELLATIONS, MPPS, AND FUTURE DIRECTIONS", *1521-9615/05/ 2005 IEEE.*

[12] Ryan X. Wu, Andrew A. Chien,Matti A. Hiltunen, Richard D. Schlichting, Subhabrata Sen, "A High Performance Configurable Transport Protocol for Grid Computing", *0-7803-9074-1/05/ 2005 IEEE.*

[13] Marinho P. Barcellos,Maziar Nekovee, Michael Daw,"High-Performance Reliable Multicasting for Grid Applications", *Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing (GRID04)1550-5510/04 IEEE.*

[14] P. G. Viscarola, W. A. Mason,"Windows NT Device Driver Development", *OSR Open Systems Resources, Inc. 1999, p10.*

[15] "A networking approach to grid computing",Willey Edition,2004,Daniel Minoli.

[16] Akshay Luther, Rajkumar Buyya, Rajiv Ranjan, Srikumar Venugopal,"Alchemi: A .NET-based Grid Computing Framework and its Integration into Global Grids".

[17] Carlos Queiroz, Marco A. S. Netto, Rajkumar Buyya,"Message Passing over Windows based Desktop Grids".