

Design of Simulators for Job Group Resource Allocation Scheduling In Grid and Cloud Computing Environments

Dr.Prasadu Peddi

Assistant Professor, Dept. of CSE, Priyadarshini Institute of Technology and Science, Tenali, A.P, India

ABSTRACT: Grid Computing aims to manage complex applications through a set of computing resources from multiple distributed nodes. It differs from traditional computer systems, such as group computing, because of its poorly coupled, large-scale, highly dynamic and heterogeneous nature. Cloud computing provides hardware and software resources as a service on the Internet. These services are classified as infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS). Given the constraints of implementing a real network or a cloud computing environment in terms of the required heterogeneous equipment and large-scale networks for research purposes, researchers widely apply simulation tools to model and evaluate network / cloud behavior. This paper reviews the most applied cloud and cloud simulation tools, as well as their capabilities in different aspects of applications.

KEYWORDS: Grid Computing, Cloud Computing, Distributed Systems, Simulation Tools

I. INTRODUCTION

1.1 GRID COMPUTING Grid computing provides the large-scale infrastructure used to strengthen education, business, science, research and entertainment [1]. Grid computing is mainly based on sharing resources and solving problems in the virtual organization. The geographical distribution and heterogeneity of resources, large-scale infrastructure, coordination and exchange of resources, multiple management, autonomy, scalability and dynamics are the characteristics of grid computing [2].

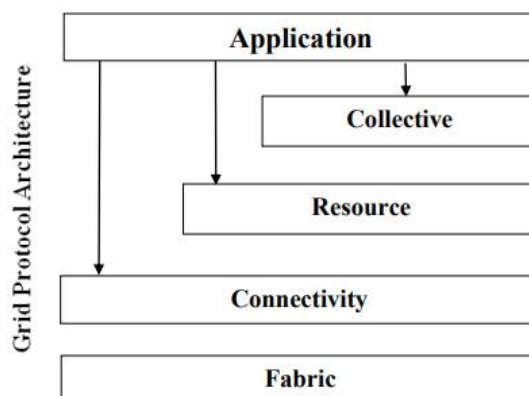


Figure 1.1: Layered Architecture of Grid Computing

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

The architecture of grid computing is based on a layered approach where the top-level layer uses the functions of the lower-level layer. The network architecture is developed by the Global Grid Forum (GGF), which anticipates the advancement of open energy for the evolution and support of new standards for network computing. It also exchanges information and sets standards for IT and network. The architecture of the grid is composed mainly of five layers, [4], as shown in Figure 1.1. The fabric layer in the lower layer and this layer integrates the networked computers, the resource management system, the database system and the storage system. The connectivity layer operates for secure authentication communication and authorized access to network resources. The resource layer defines protocols for resource negotiation, resource discovery, resource monitoring, resource allocation, and resource calculation. The collective layer interacts with and oversees the resources and directory services of virtual organizations. The application layer is a top layer and encompasses the user application and the application program interface. It works with virtual administration. Grid computing provides the benefits of resource sharing, seamless access to heterogeneous resources, fault tolerance, and reduced application processing time on a large scale; reduce the number of servers and compute capabilities.

Types of grid computing the grid computing system is generally classified into the following categories.

Computational Grid: Computing Grid provides superior combined computing capacity for unique applications compared to the capacity developed for these applications in the system. Networks use mutual resources to solve a single problem. In addition, the calculation grid is divided into two categories: the distributed computed grid, the high-speed grid, and the scan grid. Distributed computing network system runs the application in parallel in multiple systems to reduce processing time, weather forecasts and nuclear simulation require intensive computing. The broadband network increases the completion rate of a workflow. Monte-Carlo simulations use high-speed computing. The cleanup grid is used to discover inactive servers and computers in the machine cycle for use in resource-intensive tasks.

Data grid: Data networks provide a large-scale infrastructure to create new data and information from digital libraries and store this data and information in data warehouses.

Service grid: Service grid provides the services necessary for the proper functioning of the grid system without any individual machine. These services may be maintenance services, resource monitoring, work scheduling, security, etc. The service network is divided into three sub-categories: network on demand, collaborative network, and multimedia network. The on-demand network dynamically combines several different resources to deliver the new services. The collaboration grid connects the user and the application collaboratively via the virtual workspace. The multimedia grid provides an infrastructure for multimedia applications in real time.

1.2 CLOUD COMPUTING Cloud computing is the latest emerging focus on utilities. It is based on the concept of dynamic provisioning and virtualized infrastructure used, on-demand hardware and software services, IT outsourcing and service level agreements. Cloud computing helps companies, governments, public and private institutions and research organizations to define a more real and demand-driven IT system [5]. Virtualization, on-demand service, elastic computing, resource sharing, high computing power, reliability, security and ease are the main features of cloud computing. The cloud computing architecture is also based on a layer approach and consists of four layers, as shown in Figure 1.2.

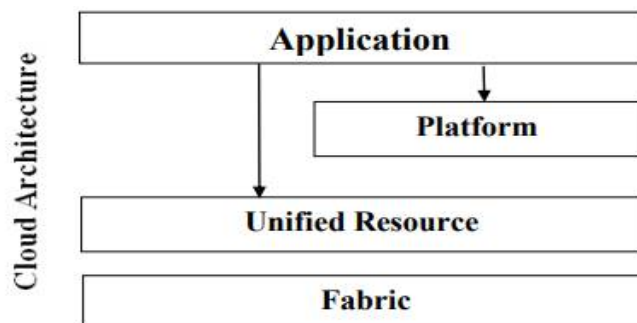


Figure 1.2: Layered Architecture of Cloud Computing

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

The lowest layer is the framework layer that includes network resources, calculates resources and storage resources. The unified resource layer contains resources that can be compacted through virtualization, and these compact resources can be visible to the upper layer and the end users as combined resources. A logical file system and a database system is an example of a unified resource layer. The platform layer improves with middleware and lower layer services, that is, the unified resource layer, to provide an established and organized platform. The application layer includes a series of cloud applications that run in a cloud environment. Amazon Web Services, Salesforce.com, Google App Engine and Azure Services Platform are examples of cloud service providers. Cloud computing has three service models [3]: SaaS, PaaS, IaaS.

Software as a service (SaaS) is a service model that attempts to make the software available in the cloud without having to install this software on your local computer. Google Docs and Customer Relationship Management software are examples of SaaS models. The PaaS model (platform as a service) provides a platform (middleware, database and development tools) for software development and the extension of existing applications. Microsoft Azure and Google App Engine are examples of PaaS. The Infrastructure as a Service (IaaS) model provides users with an infrastructure to obtain and use computing resources (memory, processing capacity and storage) to organize their applications. S3 and Amazon Web Services EC2 are examples of IaaS. Scalability, flexibility, economy, reliability, ease, greater security and lower cost are the benefits provided by cloud computing.

II. LITERATURE REVIEW

A network computing system is a virtual machine that consists of a set of heterogeneous networked machines that are willing to share their local resources with each other. Grid and Cloud is a large-scale network computing system that adapts to Internet-sized environments with machines distributed across multiple organizations and administrative domains. The resource management system is the central component of grid computing and the cloud computing system. Network and cloud resources are distributed, heterogeneous, autonomous and unpredictable. A resource management system allocates requests to resources, combined resources program, and executes requests using scheduled resources. The planning in the environment of the network and the cloud depends on the characteristics of the tasks, the machines and the connectivity of the network.

2.1 RESOURCE MANAGEMENT IN GRID COMPUTING Grid Computing is a coordinated solution for shared resources and problem solving in dynamic and inter-institutional virtual organizations [6]. Grid Computing is a flexible, secure and coordinated exchange of resources between dynamic collections of people, institutions and resources. Resource management is fundamental to the functioning of a network. The basic function of resource management is to accept requests for resources from the machine on the network and to assign specific machine resources to a request from the general resource group of the network for which the user has access. a right of access. A resource management system maps requests to resources, schedules matching resources, and executes requests using scheduled resources. Grid resources are entities such as a processor, disk space, memory, network bandwidth, and so on. managed by the resource management system. The resource scheduling process of the grid can be defined as the process of mapping a resource request, described in terms of required characteristics, to a set of resources meeting the expressed requirements. Task scheduling consists of assigning tasks to specific physical resources, taking into account the user-specified performance optimization function.

2.1.1. Types of Grid Resource Management

SLA-supported Resource Management: This architecture reflects the operational needs of the service provider and points to a more automatic and autonomous resource configuration through the introduction of a conversion factory [7]. Service level agreements that use formalized business objectives (business level objectives), complexity analysis and knowledge of previous configurations.

Agent-based Grid Resource Management: An agent-based approach to resource management [8] is flexible, robust and scalable. Resource Management Agents combine the requests with the available resources and arbitrate between the requests according to the values of the company. Each agent implements one or more assignment semantics and announces its capabilities to the name / discovery service. Their behavior depends on the assignment semantics they

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

implement and extends the basic behavior of matching a requested resource profile with a resource available in its set of main resources.

An Enterprise-Based Grid Resource Management System: An enterprise-based resource management system is a highly scalable resource management system created with Commercial Off-The-Shelf (COTS) components used in Web server infrastructures. Web services and business components were the core of the system, providing fault tolerance, performance, and scalability.

Agreement-Based Resource Management: Resource management based on agreements is based on next-generation standardization activities and resource management services. The agreement is a local administration policy that represents policies of fundamental resources and provides sources to create alternative strategies to manage the resources of the network. An agreement provides a powerful mechanism for the virtualization or abstraction of a resource. A resource may have a complex internal policy to decide authorizations, relative priorities and other planning and management procedures.

2.1.2 Resource Management Models: The resource management model provides information about the relationship between users and the resource provider, how resources are chosen for users, and how resources are allocated. There are several different models for the network resource management system and these different models can be used in the same system.

Economic Model for GRM It offers a decentralized resource management capability and is adaptable to changing environments and the needs of users. It is a scalable, controllable, measurable and easily understandable resource management policy. There are several business models based on the market, such as the market for basic products, tenders and auctions, as well as architecture and algorithms for their implementation in grid computing systems.

Architectural Models for GRM For the network resource management architecture, three different models are: hierarchical model, abstract owner model, and market model. The hierarchical model shows the approach followed in many contemporary network systems. The abstract owner model follows a request and delivery approach in the presentation of works and the collection of results. The market / economic model follows the economic model in discovering and programming resources that can coexist or work with contemporary systems and captures the essence of hierarchical and abstract proprietary models.

Pricing Model for GRM This pricing model for network resources is based on the theory of the financial option. The calculation of the value of the option is used to use the resources of the network and to select the best point of the exercise of the option to use any of the resources of the network. This helps the user, as well as the provider of network resources, to optimize resources to maximize profitability. A variable price function that controls the price of resources and ensures that network users obtain the best value at the best price and the resource provider to generate a reasonable income with the current base price. A price variant function in the pricing model is to adjust the charges for resources at different times to keep the network busy. This feature helps the resource provider to occupy the network and recover the investment in the infrastructure within a predefined time.

III. PROBLEM DESCRIPTION

A simulation study can be divided into the following steps for the problem solving approach [9]:

- The first step is the formulation of the problem, which can be formulated by defining a clear and unambiguous description of the problem, defining the objectives of the study and specifying the methodology used to evaluate the problem.
- The second step is the construction of the model. There are no standard rules for creating a model suitable for all kinds of situations. You can follow some specific guidelines for building the model. Model construction involves key features of the system and selects and modifies the basic assumption that characterizes the system.
- The third step is data collection. The type of data to be collected will depend on the purpose of the study. Data is not only required at the input of the model, but some data is also used to validate the simulation model.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

- The fourth step is programming the model. Numerous general and special simulation languages are available for writing simulation programs, and simulation software has also been developed specifically for specific simulation modeling problems.
- The fifth step is validation, which involves both validation of logic and precision of programming. Validation is an iterative process of comparing the model to the actual behavior of the system, identifying gaps, applying corrections, and comparing performance again.
- The sixth step is experimentation and analysis. For the results to be meaningful, the simulation experiments must be designed so that the results obtained are within the specified tolerance limits. The results obtained are analyzed to estimate the performance of the system.
- The last step is documentation and implementation. The documentation of a simulation program is necessary because the program can be used by the same analyst or by another analyst in the future. Implementation of the simulation program after full understanding of the model and its input-output requirements.

IV. EXPERIMENTAL RESULTS

Total completion time of all tasks will be sum of completion time of all tasks groups (CTG),

$$CT = CTG_1 + CTG_2 + CTG_3 + \dots + CTG_n$$

and the execution of each group is independent and leaves the average time to execute a group of tasks β , then an average number of task groups executed per time unit equal to $\mu = 1 / \beta$. [10] Therefore, the total completion time of the task group follows a negative exponential distribution.

In addition, the average completion time of each of these n groups is the same, so the sum of the completion time of these groups is also called the Erlang- n distribution. Therefore, the distributed random variable Erlang- n is the sum of n independent and identically distributed exponential distributions. Its density function can be represented by [11]:

$$h_n(t) = \frac{1}{(n-1)!} \cdot \left(\frac{1}{\beta}\right)^n \cdot t^{n-1} \cdot e^{-\frac{t}{\beta}}$$

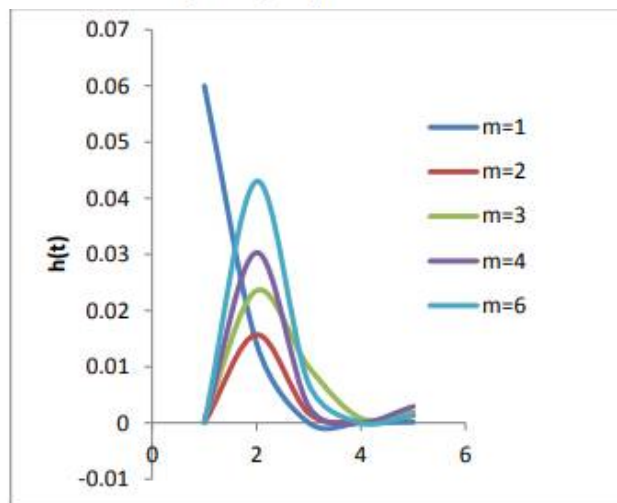


Figure 3.1: Completion Time follows the Erlang distribution

In this chapter, the workgroup resource allocation planner simulator based on a hierarchical structure was examined, so its performance must be evaluated. The simulator is implemented using MATLAB. A comparison is made between two algorithms based on a hierarchical structure, one based on the grouping of tasks and the other on tasks that are not grouped or that indicate the individual execution of tasks in resources according to different performance and time parameters planned execution.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

First, the simulation is done for the end time. Here, you enter a number of tasks and resources and record the time to complete the tasks. Figure 3.2 shows that if the tasks are assigned in groups, the completion time are less than the time assigned to the individual tasks.

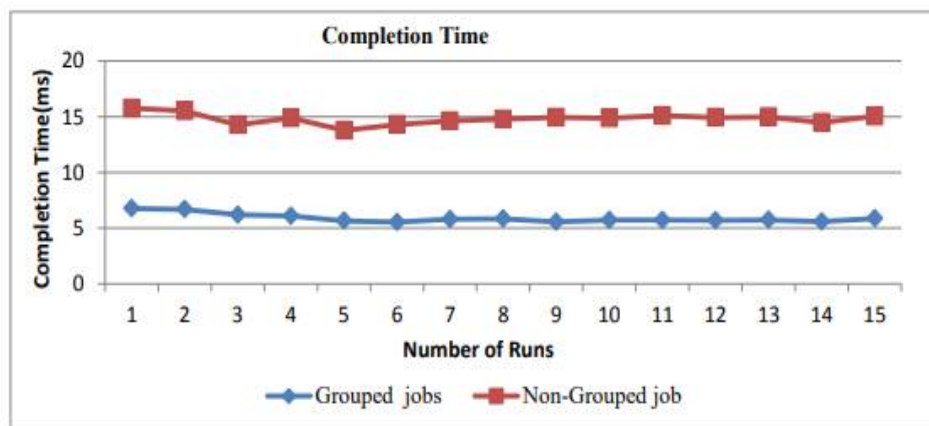


Figure 3.2: Completion Time of Grouped and Non-Grouped Jobs

Second, the simulation is performed for the expected performance. Here, you enter a number of tasks and resources and record the expected performance of all tasks. Figure 3.3 shows that if tasks are assigned to groups, the expected performance is high compared to an individual assignment of tasks to resources.



Figure 3.3: Expected Performance of Grouped and Non-Grouped Jobs

Third, the simulation is performed for the expected execution time. Here, a number of tasks and resources are entered and the expected execution time of all tasks is noted. Figure 3.4 shows that if the tasks are assigned in groups, the expected execution time is low compared to the tasks assigned individually to the resources.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

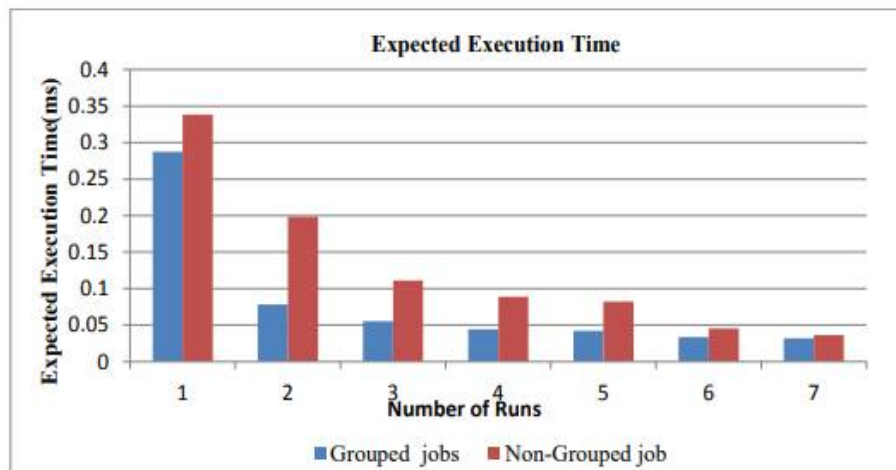


Figure 3.4: Expected Execution Time of Grouped and Non-Grouped Jobs

V. CONCLUSION

Resource allocation is one of the important issues to solve in a grid computing environment. In this chapter, workgroup resource allocation planning simulator based on a hierarchical approach to ungrouped jobs taking into account an objective parameter, namely completion time, expected performance, and time to completion. The results of the simulation show the effectiveness and efficiency of the presented simulator. The presented simulator reduces the execution time, the expected execution time and increases the expected performances. In the future, the workload balance and network delay will be taken into account in order to make a simulator more efficient and effective in group resource allocation work.

REFERENCES

1. A. Blatecky, Grid Computing, EDUCAUSE Center for Applied Research, Research Bulletin, Volume 2002, Issue 17, pp.1-11, September 3, 2002.
2. B. Y. Bichi, T. Ercan, An Efficient Queuing Model for Resource Sharing in Cloud Computing, International Journal of Engineering and Science (IJES), Volume 3, Issue 10, pp. 36-43, 2014.
3. C.H. Papadimitriou, J.N. Tsitsiklis, On stochastic scheduling with in-tree precedence constraints, SIAM Journal of Computing, Volume 16, Number 1, pp. 1-6, February, 1987.
4. C. Kandagatla, Survey and Taxonomy of Grid Resource Management Systems, University of Texas, Austin, pp.395- 406, 2003.
5. D. Minarolli, B. Freisleben, Utility-based Resource Allocation for Virtual Machines in Cloud Computing, IEEE Symposium on Computers and Communications (ISCC), pp. 410-417, 2011.
6. F. Berman, G. C. Fox, A.J. G. Hey, Grid Computing: Making the Global Infrastructure a Reality, Wiley Student Edition, John Wiley & Sons, 2003.
7. Prasadu Peddi (2016), Comparative study on cloud optimized resource and prediction using machine learning algorithm, ISSN: 2455-6300, volume 1, issue 3, pp: 88-94.
8. K. Czajkowski, I. Foster, and C. Kesselman, AgreementBased Resource Management, Proceedings of The IEEE, Volume 93, Number 3, pp. 631-643, March 2005.
9. M. Ma, J. Wu, S. Li, D. Chen, Z. Hu, A Grid-distance Based Scheduling for Grid Resource Management, Proceedings of the Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA'05), pp.1-6, 2005.
10. R. K. Bansal, A. K. Goel, M. K. Sharma, MATLAB and its Applications in Engineering, Pearson publication, 2012.
11. S. Singh, K. Kant, Greedy Grid Scheduling Algorithm in Dynamic Job Submission Environment, Proceedings of ICETECT 2011, IEEE, pp. 933-936, 2011.
12. X. Lu, Z. Gu, A Load-Adaptive Cloud Resource Scheduling Model Based On Ant Colony Algorithm, Proceedings of IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS2011), pp. 296-300, 2011.
13. Prasadu Peddi (2016), Experimental Study on Cloud Resource Prediction and Allocation using Bat algorithm, ISSN: 2455-6300, volume 1, issue 2, pp: 88-94.