The Availability of Workloads for Grid Computing Environments

A. Neela Madheswari
Associate Professor, CSE Department
Mahendra Engineering College
Namakkal, India

R. S. D. Wahida Banu
Principal
Government College of Engineering
Salem, India

***Abstract*—Grid Technology is a growing information technology field where the main purpose of grid is to build a kind of dynamic, distriuted and heterogeneous computing environment and realize collaborative resource sharing and problem solving in dynamic and multiple virtual organizations. It enables sharing, selection and aggregation of suitable computational and data resources for solving large-scale data intensive problems in science, engineering and commerce. To perform a study on grid scheduling or any other important concepts, acquiring a real grid environment is costly. To avoid that, we can use simulation tools. This paper provides the list of various simulation tools for grid computing together with the data sets collections. To perform a research based on grid computing environments, these tools and data sets explained in this paper are useful.**

Keywords*—Grid computing, workload, simulation, Bricks, SimGrid, Monarc, GridNet, OptorSim, EcoGrid, GangSim, SimJava

I. INTRODUCTION

Grid computing is the collection of computer resources from multiple locations to reach a common goal. The grid is a distributed system with non-interactive workloads that involve a large number of files. Grid computing tries to bring under one definitional umbrella all the work being done in the high performance, cluster, peer-to-peer and Internet computing arenas. Grid computing enables virtual organizations to share geographically distributed resources as they pursue common goals, assuming the absence of central location, central control, omniscience and an existing trust relationship.

An efficient functioning of a complicated and dynamic grid environment requires a resource manager to monitor and identify the idling resources and to schedule users submitted jobs accordingly. A common problem arising in grid computing is to select the most efficient resource to run a particular program [8].

The basic goal of a grid computing environment is to allow users to access computational resources by just 'plugging in' to the grid, similar to the way electrical energy is supplied when one plugs into the electrical power grid. Grid services are treated like a utility such as electricity, where once the user is connected to the grid it appears as essentially one large computer system [5].

There are a lot number of literature support for grid computing tasks and simulation tools. For example, [11], [7] explains the grid scheduling policies. Some of the applications f grids are given in [1], [4]. Only a few literatures gives the introduction for grid simulation tools [2].

This paper deals with the needs for simulation in grid environments with various workloads available for testing grid environments which is an essential study to perform before doing any research based on grid computing.

II. NEEDS FOR SIMULATION IN GRIDS

For implementing in the real grid environment, we have to know the various existing architectures, their features and their support for job processing. For the simulation, the user must know how to use the tool to obtain the result or analyze the given workload.

Simulation provides the powerful way to measure performance before the system under study has not actually been implemented. Such simulation can capture the dynamic interaction between applications and parallel architectures. Also it offers flexibility as one can take modifications to the simulation model and check their effect easily. Modeling and simulation has emerged as an important discipline and many standard and application-specific tools and technologies have been built.

III. SIMULATION TOOLS FOR GRIDS

There are many number of simulation tools available for grids [12]. Some of them are explained as given below.

A. SimJava

SimJava is a Java-based toolkit designed to simulate complex event-based systems. SimJava is designed to simulate static networks with active entities that communicate with each other through sending/receiving passive event objects. SimJava is able to provide efficient lightweight packages to simulate and model hardware and distributed software systems, including communication protocols, parallel software modeling and computer architectures.

B. Bricks

Bricks is a performance evaluation system developed in Java to analyze and compare the performance of various scheduling schemes in high-performance global computing environments. Bricks provides 1) simulation of various behaviors of resource scheduling algorithms, 2) programming modules for scheduling, 3) network topology of clients and servers in global computing systems and 4) processing schemes for networks and servers. Bricks also gather information on

global computing resources to analyze resource scheduling algorithms.

C. MicroGrid

MicroGrid is developed to provide platforms for developing or implementing virtual grid infrastructures. MicroGrid platforms can be used to analyze grid resource management issues of Globus applications. Virtual grid infrastructures allow analysis of dynamic resource management techniques with a minimum amount of effort to increase transparency of many repeatable or controllable experiments.

D. SimGrid

SimGrid is used to evaluate scheduling algorithms for distributed applications in heterogeneous computational grids. SimGrid is used to 1) provide the right model and level of abstraction for its intended purposes, 2) rapidly prototype and evaluate scheduling algorithms, 3) enable more realistic simulations and 4) generate more accurate simulation results.

E. GridSim

GridSim is used to simulate application schedulers for distributed computing systems such as clusters and grids. GridSim is Java based and allows simulation of different classes of heterogeneous resources, users, applications, resource brokers and schedulers in a distributed computing environment.

F. GanSim

GangSim is used to support studies of scheduling strategies in grid environments with a particular focus on investigating interactions among local and community resource allocation policies. GangSim models comprise the following real grid elements: a job submission infrastructure, a monitoring infrastructure and a usage policy infrastructure.

G. Monarc

Monarc is developed in Java and is a multithreaded process oriented simulation framework to model large-scale distributed systems. It is designed to provide realistic simulation of a wide-range of distributed system technologies with respect to their specific components and characteristics. It aims to 1) extend and optimize grid modules to provide better simulation of processing nodes, 2) design and run simulation experiments for data processing activities, job scheduling, and minimum spanning tree computation in overlay networks, and 3) make multithreading performance tests on multiprocessor platforms.

H. OptorSim

OptorSim is a time-based simulation package written in Java to investigate the performance of different job scheduling and data replication schemes. OptorSim is composed of computing elements, storage elements, an RB and a replica manager.

I. EcoGrid

EcoGrid is a Java based simulator to evaluate the performance of scheduling algorithms in grids. It is dynamically configurable and supports resource modeling, advance reservation of resources, and integration of new scheduling policies. EcoGrid uses the following components to model grid environment: configuration manager, random number generator, load generator, resource calendar, computer node, computer cluster, media directory, grid process, grid, grid scheduler, statistical analyzer and grid data provider.

J. GridNet

GridNet is a modular ns-based simulator written in C++ to model different data grid configurations and resource specifications. GridNet modules are composed of objects that are mapped into the ns's application level object classes. Different network configurations, different types of nodes, different node resources, replication strategies and cost functions can be built using these ns-based objects.

K. Opportunistic Grid Simulation Tool

It is developed in Java as an extension to the GridSim toolkit. Its main objectives are: 1) to assist developers of opportunistic grid middlewares on validating their new concepts and implementations under different execution conditions and scenarios, 2) to simulate large-scale application and resource scenarios involving several users in a repetitive and controlled way.

IV. WORKLOADS FOR GRIDS

Whenever we are going to perform a simulation for grid computing environment, we have to test the simulated system with any of the real scientific workloads. It is mandatory for any research work. There are various workloads available for grid computing environments. Those are discussed in this section.

There is an important workload archive found for grid computing environment given in [13]. Currently there are up to ten traces available from the grid workload archives. They are given as follows.

A. DAS-2

DAS-2 is the second generation web-based data delivery, visualization, and analysis system built and used by the radio and plasma wave group at the University of lowa. Data are transmitted to clients along with software to manipulate and display the data [14]. The number of jobs observed in the trace is greater than 1 Million and the number of CPUs used is 400 with 500 users.

B. Grid'5000

Grid'5000 is a large-scale and versatile testbed for experiment-driven research in all areas of computer science, with a focus on parallel and distributed computing including Cloud, HPC and BigData. The number of jobs available from the trace is greater than 1 Million. The number of CPUs used is 2500 with 1000 users [15].

C. NorduGrid

NorduGrid takes part and strives to support various projects that help development and proliferation of Grid Middleware in general and ARC (Advance Resource Connector) products in particular. The number of jobs available is 1 Million, the number of CPUs used is 5000 with 500 users [16].

D. AuverGrid

AuverGrid is a production grid platform consisting of 5 clusters located geographically in the Auvergne region, France. AuverGrid project is a regional grid part of the EGEE (Enabling Grids for E-science in Europe) project. This grid employes the LCG (Large Hadron Collider Computing Grid project) middleware as the grid's infrastructure. It is used

mostly for biomedical and high-energy physics research. The number of jobs observed in the trace are 5,00,000 and the number of CPUs used is 1000 with 500 users [17].

E. NGS

NGS gives the trace analysis report for the NGS system. The number of jobs observed in the trace is 1 Million and the number of CPUs used is 1000 with 500 users [18].

F. LCG

LCG log contains 11 days of activity from multiple nodes that make up the LCG. Users submit serial or parallel jobs to resource brokers. The resource brokers find suitable resources for carrying out the computation and send processes for execution on the different systems. The log is at the level of individual processes, and does not contain data about which processes may be part of the same parallel job. The number of jobs available in the trace is 1 Million and the number of CPUs used is 1000 with 500 users [19].

G. GLOW

GLOW gives the details of batch jobs and the average batch size is 15 to 30. There are 50,000 jobs and the number of CPUs used is 5000 with 50 users.

H. TeraGrid

In this trace, there are 1 Million jobs and the number of CPUs used is 100 with 200 users.

I. SHARCNet

SHARCNet is structured as a ‘cluster of clusters’ across south western, central and northern Ontario, designed to meet the computational needs of researchers in adverse number of research areas and to facilitate the development of leading-edge tools for high performance computing. This trace contains upto a year’s worth of accounting records from the SHARCNet clusters installed at seeral academic institutions in Ontario, Canada. There are 1 Million jobs and the number of CPUs used is 10,000 with 500 users.

V. CONCLUSION

Grid computing is one of the main computing technologies in today’s Internet world. To know about the grid computing and its evolution is an essential task. This paper focuses mainly for giving a brief introductory part of various grid simulation tools as well as workloads that are available from real grid scenarios.

REFERENCES

[1] T.Kielmann, H.E.Bal, J.Maassen, R.V.Nieuwpoort, R.Veldema, R.Hofman, C.Jacobs and K.Verstoep, “The Albatross Project: Parallel Application Support for Computational Grids”, Proceedings of the First European GRID Forum Workshop, pp.1-8, 2000.

[2] R.Buyya and M.Murshed, “GridSim: A toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing”, 2002.

[3] I.Foster and C.Kesselman, “The Grid2: Blueprint for a New Computing Infrastructure”, Morgan Kaufmann, USA, 2003.

[4] O.Smirnova, P.Eerola, T.Ekelof, M.Ellert, J.R.Hansen, A.Konstantinov, B.Konya, J.L.Nielson, F.Ould-Saada and A.Waananen, “The NorduGrid Architecture and Middleware for Scientific Applications”, Lecture Notes in Computer Science, pp. 1-10, 2003.

[5] M. Irving, G. Taylor and P.Hobson, “Plug in to Grid Computing”, IEEE Power and Energy Magazine, 2, pp. 40-44, 2004.

[6] A.Iosup, M.Jan, O.Sommez and D.H.J.Epema, “The Characteristics and Performance of Groups of Jobs in Grids”, Euro-Par, LNCS, vol.4641, pp.383-393, 2007.

[7] B.Tang, Z.Zhou, Q.Liu and F.Li, “Market-Driven based Resource Scheduling Algorithm in Computational Grid”, International Conference on Computer Science and Software Engineering, 2008.

[8] M. Kiran, A.Hassan, A.Hashim, L.M.Kuan and Y.Y. Jiun, “Execution Time Prediction of Imperative Paradigm Tasks for Grid Scheduling Optimization”, IJCSNS, Vol.9, No.2, pp.155-163, Feb 2009.

[9] A. Chhabra, G. Singh and G. Kumar, “Simulated Performance Analysis of multiprocessor dynamic space-sharing scheduling policy”, IJCSNS, Vol.9 No.2, Feb 2009.

[10] R.Renuga and Sudha Sadasivam, “Data Discovery in Grid using Content based Searching Technique”, Information Technology Journal, Vol 8, No.1, pp.71-76, 2009.

[11] M. Kiran, A. Hassan A.Hashim, L.M. Kuan and Y.Y. Jiun, “Execution Time Prediction of Imperative Paradigm Tasks for Grid Scheduling Optimizations”, International Journal of Computer Science and Network Security, Vol.9, No.2, pp.155-163, Feb 2009.

[12] J. Taheri, A.Y.Zomaya and S.U.Khan, “Grid Simulation Tools for Job Scheduling and Data File Replication”, 2012.

[13] Details of Grid Workload Archive, www.st.ewi.tudelft.nl/~iosup/project_grid_gwa.html, 12 Feb 2015.

[14] Details of DAS-2 workload, http://www-pw.physics.uiowa.edy/das2/, 12 Feb 2015.

[15] Details of Grid’5000 workload, https://www.grid5000.fr/mediawiki/index.php/Grid5000:Home, 12 Feb 2015.

[16] Details of NorduGrid workload, http://www.nordugrid.org/, 12 Feb 2015.

[17] Details of AuverGrid workload, gwa.ewi.tudelft.nl/datasets/gwa-t-4-auvergrid, 12 Feb 2015.

[18] Details of NGS workloads, http://gwa.ewi.tudelft.nl/datasets/gwa-t-5-ngs, 12 Feb 2015.

[19] Details of LCG workloads, http://gwa.ewi.tudelft.nl/datasets/gwa-t-11-lcg, 21 Feb 2015.