Chapter 30
EcoGrid:
A Toolkit for Modelling and Simulation of Grid Computing Environment for Evaluation of Resource Management Algorithms

Hemant Kumar Mehta
Devi Ahilya University, India

ABSTRACT

This chapter presents a toolkit for evaluation of resource management algorithms developed for Grid computing. This simulator named as EcoGrid and it is devised to support large number of resource or computing nodes and processes. Generally, grid simulators represent each resource using a thread that occupies large amount of space on the thread stack in main memory. However, EcoGrid models each node by an object instead of a thread. Memory space used by an object is much smaller than a thread, thus EcoGrid is highly scalable as compared to state-of-the-art simulators. EcoGrid is dynamically configurable and works with real as well as synthetic workloads. The simulator is bundled with a synthetic load generator that generates the workload using appropriate statistical distributions.

INTRODUCTION

Recent development in hardware and internet technology has changed the way of development of commercial and scientific applications. Almost every application is either migrated to web or a candidate to be migrated on web. These applications can be implemented either as a web application or as a web service. Most of these applications process huge amount of data for example gaming, particle physics or bio-informatics applications. These applications are called as e-Science applications and demand computing environment that can process data of petabytes, exabytes or even higher scale. Till date these computing environment are capable of computing up to few petaflops. This computing power can only be achieved by integrating computing powers of a large number of geographically distributed resources

DOI: 10.4018/978-1-5225-0196-1.ch030

such as Folding@home has integrated computing power of 6.0 petaFLOPS (Beberg, 2009). *Grid* computing integrates processing power of several organizations to form a single Virtual Organization (VO). Management of these independent organizations is much easier and cost effective. This evolution in distributed computing, results in development of a large number of complex and new applications. Processing needs of these applications are increasing exponentially hence management of such environment is still remains as challenging task. Performance enhancement of resource management algorithms devised for such environment is focus of many researchers.

To enable evaluation of these resource management algorithms a highly scalable simulation environment is required. Large applications require simulation of *Grid* having lakhs of resources (Anderson, 2002). The scheduling algorithm used for such applications must be evaluated for large number of resources and processes. Grid computing was evolved with a vision to work for the applications that can process the data of petabytes scale such as Large Hadron Collider (LHC) at CERN Geneva. However, in last three years this demand of few petabytes in an year is exponentially increased and now applications are demanding few petabytes of processing in a day such as the gaming applications developed by Zynga.

Performance indices of traditional scheduling algorithms are throughput, resource utilization turnaround time etc. However, in most of the current distributed computing environments decision to accept a new process is based on cost of execution. Therefore, these performance indices must also include economy of process execution. To get proper Return On Investment (ROI) on technology within an enterprise, its resources must not only be utilized effectively but their profitability should also be considered. Economy based scheduling algorithms consider this economy and profitability. Hence, several researchers are working in the area of economy-based grid scheduling. These researchers require a suitable simulation environment for testing the performance of their algorithms. A simulation environment that can evaluate economy based scheduling algorithm on large number of resources is the demand of time.

A number of grid simulators were developed in last decade like GridSim, SimGrid, OptorSim, Brick and an emulator MicroGrid etc. MicroGrid is a grid emulator that provides virtual environment for execution of grid applications. However, we limit the discussion to GridSim and SimGrid, since the other simulators are not under active development.

The "*SimGrid*" is a widely used grid simulation tool presently available to the researcher (Casanova, 2001; Legrand, 2003; Bobelin, 2012). *SimGrid* is discrete event simulator developed in C language. *SimGrid* provides framework to build simulator for the custom application. It is having configurable network topology and users can choose the topology as per their need. It also supports distributed scheduling agent. The resources can be modeled as the time-shared resources however, it doesn't support space-shared resources. *SimGrid* is a multithreaded simulator and doesn't support dynamic configuration of the simulation environment.

The GridSim is another popular discrete event simulator developed on top of the "SimJava" (Howell, 1998; Buyya, 2002). *GridSim* mainly developed to support the economy based scheduling algorithms. It supports simulation of computational as well as the data Grid (Buyya, 2002; Sulistio, 2008). Since its inception GridSim is under continuous development and widely used. The extensions to GridSim include concept of advance reservation, fault detection, incorporation in network topology etc. (Caminero, 2007; Sulistio 2004; Sulistio 2007). *GridSim* can be configured for the custom simulations using the API provided in the toolkit by writing a Java program. However, there must be an option for dynamic configuration of various parameters from an interface. *GridSim* models each node or resource as a separate thread of execution causing run time overheads and adversely affects its scalability.

Related Content