ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

Reservation of Resources in Advance for Job Execution in Heterogeneous Grid Environment

Lata Gadhavi
Nirma University, Ahmedabad, Gujarat, India

Abstract
A Grid computing has emerged as the next-generation parallel and distributed computing that aggregates dispersed heterogeneous resources for solving a range of large-scale parallel applications in science, engineering and commerce. In most Grid scheduling systems, submitted jobs are initially placed into a queue if there are no available resources. Therefore, there is no guarantee as to when these jobs will be executed. This causes problems in parallel applications, where most of them have dependencies among each other. Advance Reservation (AR) is a process of requesting resources for use at a speci c time in the future. Common resources whose usage can be reserved or requested are CPUs, memory, disk space and network bandwidth. AR for a grid resource solves the above problem by allowing users to gain concurrent access to adequate resources for applications to be executed. AR also guarantees the availability of resources to users and applications at the required times.

Keywords
Job Execution, Advance Reservation of Resources, Advance Scheduling

I. Introduction

Grid Computing is an active research area which promises to provide a exible infrastructure for complex, dynamic and distributed resource sharing system. Globus a middleware which is a de facto standard for grid computing. This scenario contains setup for a grid environment, in which 50s or 100s of execution nodes out of which one has been choosen as a submission node, and the other two are container nodes. These container nodes are actually having the schedulers of open PBS, SGE and also condor which creates heterogeneous environment for clusters. When the submission node submits jobs to the containers, then these jobs are scheduled to run on other nodes attached to the OpenPBS and SGE clusters in parallel. Advance Reservation (AR) is a process of requesting resources for use at a specific time in the future. Common resources whose usage can be reserved or requested are CPUs, memory, disk space and network bandwidth. AR for a grid resource solves the above problem by allowing users to gain concurrent access to adequate resources for applications to be executed [1]. AR also guarantees the availability of resources to users and applications at the required times. There are some systems that support AR capability, such as Globus Architecture for Reservation and Allocation (GARA) and Maui Scheduler [2].

Some tools are available for application scheduling simulation in the Grid computing environment, such as Bricks , SimGrid and OptorSim. However, none of them have the capability of simulating reservation-based systems. To address this weakness, we extend GridSim to support AR mechanisms.

II. GridSim

GridSim is a software platform that enables users to model and simulate the characteristics of Grid resources and networks with different configurations. Study Grids, or test new algorithms and strategies in a controlled environment. By using GridSim, they are able to perform repeatable experiments and studies that are not possible in real dynamic Grid environment. GridSim is Java-based grid simulation package that provides features for application composition, information services for resource discovery, and interfaces for assigning applications. GridSim also has the ability to model heterogeneous computational resources of variable performance.

In this work, GridSim has been extended with the ability to handle:

1. Creation or request of a new reservation for one or more CPUs;
2. Commitment of a newly created reservation;
3. Activation of a reservation once the current simulation time is the start time;
4. Modification of an existing reservation; and
5. Cancelation and query of an existing reservation.

A. Salient Features of GridSim

1. Applications with different parallel application models can be simulated.
2. Application tasks can be heterogeneous and they can be CPU or I/O intensive.
3. No limit on the number of application jobs that can be submitted to a resource.
4. Multiple user entities can submit tasks for execu-tion simultaneously in the same resource, which may be time-shared or space-shared.
5. Network speed between resources can be specified.
6. It supports simulation of both static and dynamic schedulers.
7. Statistics of all or selected operations can be recorded and they can be analyzed using GridSim statistics analysis methods.

III. Advance Reservation Design

A. Sequence Diagram for Job Submission

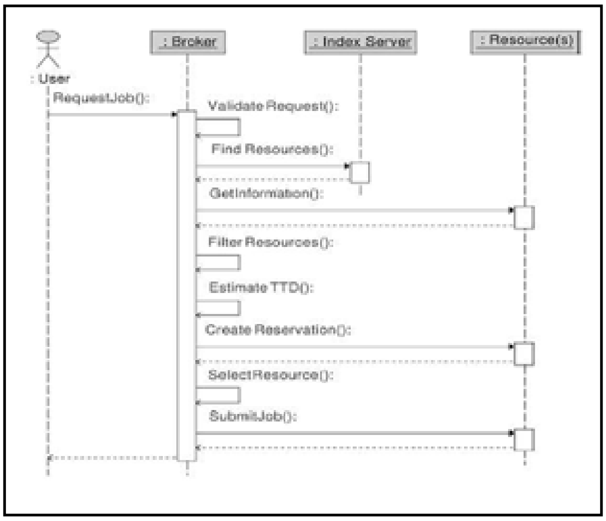

Fig. 1: Sequence Diagram for Job Submission

B. Introduction of the Sequence Diagram

- In step 1, the user's request is processed and split into individual job requests.
- In Step 2, the user discovers what resources are available by contacting one or more index servers like webmds.
- The specific characteristics of the resources found are identi ed in Step 3, by querying each individual resource. Each resource may provide static information about architecture type, memory con guration, CPU clock frequency, operating system, local scheduling system, etc., and dynamic information about current load, batch queue status and various usage policies.
- The actual brokering process is mainly performed in Step 4, which is repeated for each job request.
- In Step 5, resources are evaluated according to the requirements in the job request and only the appropiate resources are kept for further investigation.
- Step 6, predicts the performance of each resource by estimating the TTD, a step that may include the creation of advance reservations.
- Then, the currently considered job is submitted in the loop started at Step 7.
- The loop is repeated until either the job is successfully submitted or all submission attempts fail, the latter causing the job to fail [5].
- In last step, the submission of one job is completed, any nonutilized reservations are released,which shows the advance reservation of the resources.

For the advance reservation there are some requirements, like Rid (node id of the resource) for reserving the particular resource bases on the Rid or performance parameters. Which is shown in the implementation part.

C. States of Advance Reservation

Transitions between the states are de ned by the operations that a user performs on the reservation. These states are defined as follows. A reservation can be one of several states during its lifetime as shown in the fig. 1.

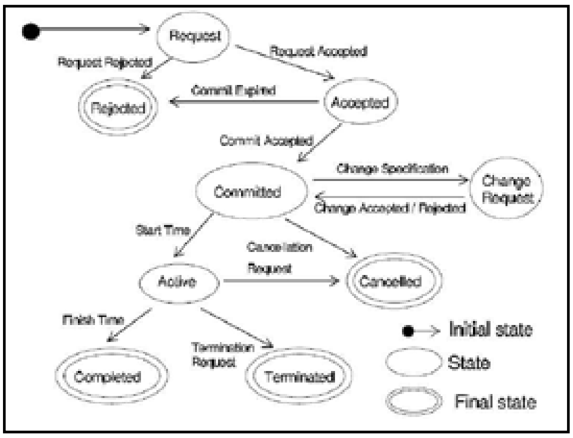

Fig. 2: A State Transition Diagram for Advance Reservation

1. Requested

Initial state of the reservation, when a request for a reservation is rst made.

2. Rejected

The reservation is not successfully allocated due to full slots, or an existing reservation has expired.

3. Accepted

A request for a new reservation has been approved. Committed: A reservation has been confirmed by a user before the expiry time, and will be honored by a resource.

4. Change Requested

A user is trying to alter the requirements for the reservation prior to its starting. If it is successful, then the reservation is committed with the new requirements, otherwise the values remain the same.

5. Active

The reservations start time has been reached. The resource now executes the reservation.

6. Canceled

A user no longer requires a reservation and requests that it is to be canceled.

7. Completed

The reservations end time has been reached.

8. Terminated

A user terminates an active reservation before the end time.

IV. Installation of Pre-requisites and Necessary Components

A. Globus Con Guration

Globus is a grid middleware, considered to be the de facto standard for grid computing. So, for practical experience of the grid computing it is best to start with the installation of Globus [3].
Operating System : Fedora 8.0 Middleware : Globus 4.2.1

1. Nodes for Grid Environment

guser01-nodeB.grid.nirma.com(Ip-10.1.3.13)(server node)
guser02-nodeA.grid.nirma.com(Ip-10.1.3.14)(client node)
guser03-nodeC.grid.nirma.com(Ip-10.1.3.19)(PBS node)

B. Installation of GridSim Toolkit

In the implementation of the Priority Based Queue & Advance Reservation algorithm GridSim Toolkit version 4.0 is used. This open source JAVA based toolkit can be downloaded from the homepage of Grid-Bus Society.
The GridSim [8], toolkit provides a comprehensive facility for simulation of different classes of heterogeneous resources, users, applications, resource brokers, and schedulers. It can be used to simulate application schedulers for single or multiple administrative domains distributed computing systems such as clusters and Grids [7].

V. Implementation

A. Advance Reservation of Resources for Job Execution

From the WebMds getting the list of all available resources then it gives all information of the all available resources. Based on the available resources and requirement of resources there is Best resource is found. Then from multiple jobs to submitting on the middleware here job is selected in queue. Now for reserving the resource in advance to execute job on that resource which was reserved. After the selection of job that job is run on the plus manager tool for reserving the resource [6].

ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

For advance reservation here Plus Manager tool & PBS (TORQUE) is installed on the main node of grid environment. Advance Reservation (AR) is a process of requesting resources for use at a specific time in the future.There are some systems that support AR capability, such as Globus Architecture for Reservation and Allocation (GARA) [4]. Here job1 is taken for submission & it is run on the plus manager tool which is in Grid1 Mtech group.
Configuration Steps for Advance Reservation:-

1. Registering Reservations

-plus reserve -s startTime [-e endTime| - D duration] -n numNodes [-U users] [-mem memSize][-ncpu cpuNum] [-arch arch] [-os os] Rid [-sgeq qNames][jobname] [-x]
As example,
-plus reserve "2011-02-02T12:34:56+09:00 "2011-02-03T02:30:51+09:00 02:32 2 lata 124M 2 x86 WINNT51 pc-1 SGE job1 -X
When using the -x option, the information is displayed in XML format.

2. Canceling Reservations

-plus cancel [-r rsvID] [-o owner]
-plus cancel pc-1 lata reservation is cancelled

3. Changing Reservations

-plus modify -r rsvID [-s startTime] [-e endTime | -D duration] [-n numNodes] [-U users]

4. Discarding Reservations

-plus destroy [-r rsvID] [-o owner]

5. Con rming Reservation Operations

-plus commit -r reservationID

6. Discarding Reservation Operations

-plus abort -r reservation ID

7. Viewing Account of Reservation Usage

-plus account [-s startTime] [-e endTime] [-d days] [-f log le] [-o [owner]] [-n [nodes]] [-l] [-S] "Reservation is created for node pc-1".

VI. Analysis of Results References

Job ID	Group	Status	Reserved Node
Job1	Mtech	Reserved	pc-1
		Unreserved	pc-19,Pc-16,pc-18,pc-14,pc-11,pc-10
Job2	B.Tech	Reserved	Pc-19
		Unreserved	Pc-16,pc-14,pc-18 ,pc-11,pc-10

Fig. 3: Analysis of Results

From this con guration steps we can conclude that which node are reserved in advance for any job,which is shown in the table. Here it shows that job1 from Grid1 Mtech group has reserved the node 1 for exe-cute job on that reserved node. When once a node is reserved for job execution then it come out from all available resources which are not reserved. So new user can not submit any other job on that node at that time. Here for job2 pc-19 is reserved to execute the job on that node.

VII. Conclusion

It is envisaged that the grid infrastructure will be a large-scale distributed system that will provide high-end computational and storage capabilities to differentiated users. Complex scenario appeared while job submission & execution are identified. From the multiple jobs here one job is selected for submission using combination of different algorithms. Then that job is ready to run on that node, where user wants to reserve the node. Selected job is run on reserved node which is reserved in advance. From the results observations are shown that one resource is reserved for particular selected job.

References

[1] Jarek Nabrzyski, Jennifer M. Schopf, Jan Weglarz,"Grid Resource Management State of the art and Future trends", Kluwer Academic Publisher.
[2] Yanmin ZHU,"A Survey of Grid Scheduling", Department of Computer Science Hong Kong University of Science and Technology.
[3] Ferreira, L., Bieberstein, N., Berstis, V., Armstrong, J.,"Introduction to Grid Computing with Globus", Redbook IBM Corp.
[4] Ivan Roderoa, Francesc Guimb, Julita Corbalan,"Grid broker selection strategies using aggregated resource information", Computer Architecture Department, Technical University of Catalonia (UPC), Spain, 2009.
[5] Marek Wieczorek, Mumtaz Siddiqui, Alex Villazon, Radu Prodan, Thomas Fahringer,"Applying Advance Reservation to Increase Predictability of Work ow Execution on the Grid", Institute of Computer Science, University of Innsbruck, May, 2007.
[6] Anthony Sulistio, Uros Cibej, Sushil K. Prasad, Rajkumar Buyya1ym,"GarQ:An E cient Scheduling Data Structure for Advance Reservations of Grid Resources", GRIDS Laboratory, The University of Melbourne, Australia, July, 2008.
[7] Globus Consortium, [Online] Available: http://www.globusconsortium.org
[8] Manzur Murshed Gippsland, Rajkumar Buyya,"Using the GridSim Tool Kit for Enabling Grid Computing Education".
[9] WS-GRAM. [Online] Available: http://www.globus.org/toolkits/docs/4.2.1/execution