RESEARCH PAPER | Engineering | Volume - 5 | Issue - 1 | Jan Special Issue - 2015 | ISSN - 2249-555X

Selection of Best Resources and Advance Resource Reservation for Complex Jobs in Grid Computing Environment

KEYWORDS: Job submission, monitoring, advance reservation,

Lata Gadhavi	Viren Patel
Asssistant Professor, Department of Computer science and Engineering, S.P.B. Patel Engineering College, Mehsana, Gujarat	Asssistant Professor, Department of Computer science and Engineering, S.P.B. Patel Engineering College, Mehsana, Gujarat

ABSTRACT Computational grids are a new trend in distributed computing systems. In the distributed environment every node has its own operating system, own resources, own processing speed, so the responsibility of communication between different platforms have been done by middleware. For this purpose, middleware takes the jobs from users and according to the job specification resource discovery has been done by middleware as a first step then assign the job to the particular resource, which has been discovered. So the middleware is responsible for the resource discovery, resource allocation and job will be executed successively in the grid environment. To reduce the load from the middleware & provide better facility to grid for job submission, selection of best resource from all the available resources & advance reservation of resources are to be handled. In addition to processor utilization, it is important to consider the waiting time, throughput, and response times of jobs in evaluating the performance of grid scheduling strategies. To handle above mentioned issues, best resource selection & advance reservation of resources system model is designed and deployed in this paper.

I. Introduction

In today's complex world of high speed computing, computers have become extremely powerful, even home-based desktops are powerful enough to run complex applications. But still we have numerous complex scientific experiments, advanced modeling scenarios, genome matching, and astronomical research, a wide variety of simulations, complex scientific & business modeling scenarios and real-time personal portfolio management, with the need of pool of resources. To satisfy some of these aforementioned requirements, Grid Computing [1] is born. Grid Computing can be consideration of dispersed and significant cluster/grid computing and as a structure of distributed parallel processing [2]. Grid Resources [3] fall into the categories of computation (i.e. a machine sharing its CPU), storage (i.e. a machine sharing its RAM or disk space), communication (i.e. sharing of bandwidth or a communication path), software and licenses and special equipment (i.e. sharing of devices). The architecture of the Grid is often described in provisos of different levels, which providing a specific function. In general, the higher layers are focused on the user, whereas the lower layers are more focused on computers and networks [5]. Grids have to be able to recognize messages that are relevant to them, and filter out the rest. This is done with communication protocols, which let the resources speak to each other, enabling exchange of data, and authentication protocols, which provide secure mechanisms to verify the identity of both the users and the resources involved [4].

The services include:

- Keeping directories of available resources updated at all times
- Brokering resources
- Monitoring and diagnosing problems on the Grid
- Replicating key data so that multiple copies are available at different locations

II. Related Work

This section covers the review of different papers on grid computing. As discussed in paper [6] Grid computing paradigm unites geographically distributed and heterogeneous computing storage, and network resources and provide united, secure, and pervasive access to the combined capabilities.

The increasing demand for high performance computing resources [7] has led to new forms of collaboration of distributed systems, such as grid computing systems. Moreover, the need for inter operability among different grid systems through the use of common protocols and standards has become increased in the last few years. In this paper authors described and evaluated scheduling techniques for multiple grid scenarios. In particular, they consist of the proposed "best Broker Rank" broker selection policy and two different variants. Oliveira, J. F. A & Lechuga, T. A. [8] has defined the pricing of resources on computational grids where In order to determine a price for the resources usage, first it is necessary to identify the target of consumption, i.e., what is being ordered. Following the logic that an outstanding characteristic of the grids is heterogeneity, it is known that a certain job may have different execution times depending on the peer to which it is submitted.

The task of a Grid resource broker and scheduler is to dynamically identify and characterize in paper [9] the available resources, and to select and allocate the most appropriate resources for a given job. The resources are typically heterogeneous, locally administered, and accessible under different local policies.

III. System Model

Handling complex use scenario are ordered by increasing complexity, where complexity is loosely defined as the number of decisions that must be made by the grid software, or the scope of those decisions. Steps identified by the handling complex use scenario are,

- **Immediate job execution**

➢ Run a job on a specified grid computer:

Local I/O & non-local file I/O required in a basic approach to remote job submission, the user specifies the execution host to be used and submits a job for which either the code already exists on the target machine or else is uploaded as part of the request.

➢ Run a job on Best grid computer:

Instead of mandating the specific grid computer in the request, a user may run a job on the "best" computer defined to the host that is based on the comparison of

Volume - 5 | Issue - 1 | Jan Special Issue - 2015 | ISSN - 2249-555X

available and required resources.

- Job execution that requires advance scheduling(Advance Reservation)
- Make a reservation for the simultaneous resources.
- Job execution that requires advance scheduling (Advance Reservation)
- Make a reservation for the simultaneous resources.

IV. Methodology

a. Overview:-

Run a job on a specified grid computer:- In a basic approach to remote job submission, the user specifies the execution host to be used and submits a job for which either the code already exists on the target machine or else is uploaded as part of the request.

- Run a job on Best grid computer:- Instead of mandating the specific grid computer in the request, a user may run a job on the "best" computer defined to the host that is based on the comparison of available and required resources. To select the Best Resource from all available resources here some implementation steps are done.
- Criteria for the Best Resource are applied that is Quickest & Cheapest resource, based on the previous history of the resource's performance, then with the comparison of job's parameter requirement for resource and available resources' parameter.

b. Process of Job Execution

Computational grid which explores idle processing power is formed in the lab. This heterogeneous grid environment is configured by the interconnecting 25 nodes with GT 4.2.1 & condor 7.2.1. Head node is loaded with globus & condor, where jobs are submitted by the user. Other nodes are configured with PBS & SGE. Job will be submit on head node and then it will run on the grid node which is selected for to run & it will show the result of the job execution. Flow of an algorithm as shown in figure 1, it elaborates the process of job execution.

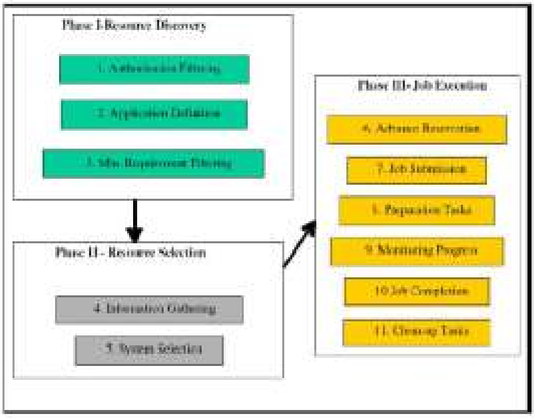


Figure 1: System Model

C. Algorithm for Best Resource Selection

Proposed brokering algorithm will perform a series of tasks, e.g., it processes the xRSL specifications in the job requests, discovers and characterizes the resources available, estimates the TTD for each resource of interest, makes advance reservation of resources, and performs the actual job submission. This algorithm presents a high-level outline of the tasks performed. The input xRSL specification(s) contain one or more job requests including information about the application to run (e.g., executable, arguments, input/output files), actual job requirements (e.g., amount of memory needed, architecture requirements, execution time required, million instructions per second(MIPS),Gflops), and optionally, job characteristics that can be used to improve the resource selection (e.g., listing of benchmarks with performance characteristics relevant for the application).

Pseudo code of the algorithm

```
Brhpsjui n :- Job submission and selection of resource

Jbqvu:-Job(Ji),i=1,2,3...n.(Different Jobs)
        Pi,i=1,2,3...n(Number of processor)
        S=S1 ; V4; 'S3 ; 'Sn (Different size of job)

Action: Job identifier(s) for the submitted job(s) & searching the best resources from the available resources.

Output: Run job on the best resource from all available resources based on the performance.

Step 1: for(every Ji in J) Do

Step 2: if(Rq(Ji) >P) then
            Select Best Resource
            Run Job and Execute
        else if (Rq(Ji) > P) then
            Run through Best fit algorithm
            execute
        else
            Run on all ordinary resources
            (Pji,Si),(Pji,Si),(Pji,Sn)
        end if
Step 2: end for
```

Process of best resource selection through algorithm is shown in the figure 2.

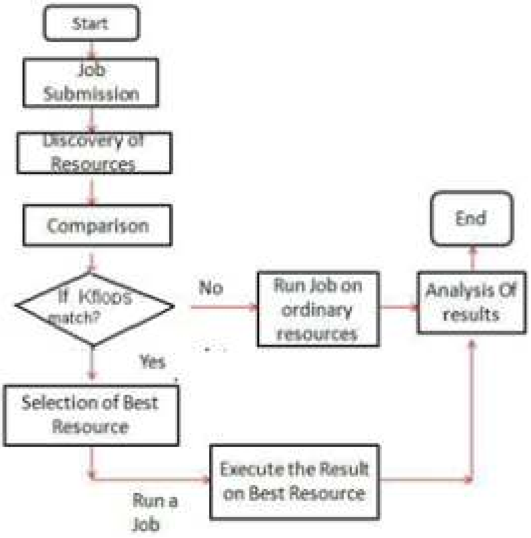


Figure 2: Flow Graph for Best Resource Selection

V. Experimental Environment

This section shows the steps for the algorithm configuration with the Meta computing Directory Services & other functionality. Three main nodes of grid are

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

- guser03-nodeB.grid.nirma.com
  (Ip-10.1.3.19)(PBS node)

First user will submit job on main node. In the user interface, user has to enter the job id & job location for the selected job, then user will submit a job on the main node of the grid, job will submitted successfully then that job will compare it's kflops with the available resources which are configured with the grid, by the best fit algorithm it will match best suitable resource to run job on that resource. So selection of resource is implemented in this phase. After the selection of the resource, current job will run on that resource, which calls best resource. It shows the parameters related job which covers job location, Job type, MIPS, kflops(floating point instructions per second),time taken for execution ,etc.

Steps for job requirements are,
1. Availability of resources is checked against the required resources.
2. Then algorithm selects the best resource to satisfy the need of job.
Job properties and specifications are shown in the following table.

No	Job Location	Job Type	Kflops	Time(sec)
1	D:/bc/bin	tsp.c	140	0.01
2	D:/bc/bin	mm1.c	60	0.03
3	D:/bc/bin	perf.c	100	0.02

Table I: Job Specification

From the WebMds getting the list of all available resources then it will give all information of the all available resources. These available resources will give the information like machine name, Machine IP, Total condor Load Average, Total Load Average, MIPS of Machine, Kflops of Machine & operating System. Based on the available resources and requirement of resources for job there is best resource will found. When webmds run on the local host it shows the all information of the connected grid resources like address of the resource, GRAM version,host name,total CPUs,etc.

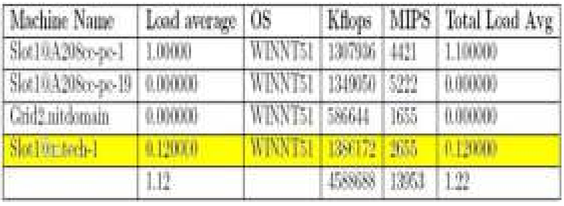
Machine Name	Load avg	Kflops	MIPS	Total Load Avg
Slot1@A208cc-pc-1	1.000	1307936	4421	1.1000
Slot1@A208cc-pc-19	0.000	1349050	5222	0.0000
Grid2.nitdomain	0.000	586644	1655	0.0000

Table II: Available Resources

- Comparison between job & available resources' parameters

Here comparison is held based on the kflops criteria. Programs kflops & Available resources kops are compared. By comparing the kflops got the results for Best resource, which resources kops is best suitable with the demanding kops so by using the programming on the grid it will select the Best resource from the all available resources.

In the following table it shows the all resources which are available and kflops of the resources & selected resource for job submission. It will send the which resource's kops match with the job's requirements, which calls kflops of job and other functionality parameters related with the job, which is submitted on the main node.



Machine Name	Load average	OS	Kflops	MIPS	Total Load Avg
Slot1@A208cc-pc-1	1.00000	WINNT51	1307936	4421	1.10000
Slot1@A208cc-pc-19	0.00000	WINNT51	1349050	5222	0.00000
Grid2.nitdomain	0.00000	WINNT51	586644	1655	0.00000
Slot1@mtech-1	0.12000	WINNT51	1386172	2655	0.12000
	1.12		4588688	13953	1.22

Table III: Comparison of available and required resources

Load Monitoring:-
Ganglia is a scalable distributed monitoring system for high performance computing systems such as clusters and Grids[8].Figure 3: Load Monitoring

- **Advance Reservation:-**

Make a reservation for simultaneous resources
-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.
-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.

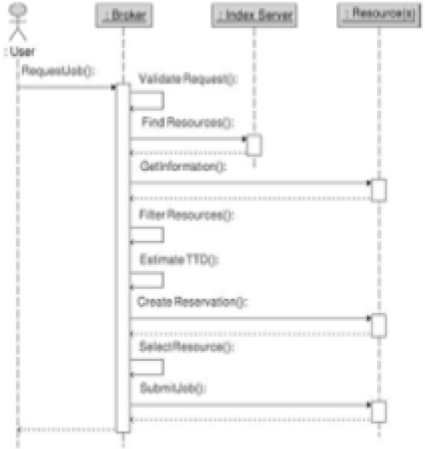


Figure 4 : Sequence Diagram for Job Submission

From this configuration steps we can conclude that which node are reserved in advance for which job, which is shown in the below table. Here it shows that job1 from Mtech group has reserved the node 1 for execute that job on this node. When once a node is reserved for job execution then it comes 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.

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

Table IV : Node reservation status

VI. Conclusion and Future Work

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. Out of the identified phases,1st phase which is "selection of best resource" is implemented. Resource brokering algorithm finds a suitable & best match in the existing grid environment. In the conclusion, it is observed that best grid nodes improve the performances of the job. 2nd phase of the scenario "Advance Reservation" is implemented. 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.

In future, the improvements can be done by also taking into account the dynamic behavior of the grid resources. Multiple users can submit job from different middleware. At present this work for local grid but next this can developed in VO(virtual Organization).

REFERENCE

1. Kleinrock, L. MEMO on Grid Computing, University of California, Los Angeles,1969. | 2. Foster, I., Kesselman, C. and Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing applications, Vol. 15, No. 3, pp. 200-222, 2001. | 3. Jarek Nabrzyski, Jennifer M. Schopf & Jan Weglarz, Grid Resource Management State of the art and Future trends, Kluwer Academic Publisher. | 4. Yanmin ZHU, A Survey of Grid Scheduling, Department of Computer Science Hong Kong University of Science and Technology | 5. Ferreira, L., Bieberstein, N., Berstis, V., Armstrong, J., Introduction to Grid Computing with Globus, Redbook IBM Corp. | 6. Chunlin Li,Layuan Li."Three-layer control policy for grid resource management", Department of Computer Science,Wuhan University of Technology, Wuhan 430063,PR China, Dec,2008. | 7. Ivan Roderoa,Francesc Guimb,Julita Corbalan,"Grid broker selection strategies using aggregated resource information",Computer Architecture Department, Technical University of Catalonia (UPC),Spain,2009. | 8. Oliveira, J. F. A. & Lechuga, T. A."Pricing of resources on computational grids using Globus " Faculdade Ruy Barbosa, Salvador, Bahia, Brazil. February,2007 | 9. Erik Elmroth , Johan Tordsson, Department of Computing Science,"Resource selection based on performance predictions" Umea University,Sweden ,June 2007.