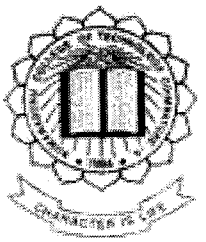


JOB SCHEDULING IN GRID COMPUTING ENVIRONMENT

By

M.Shanmuga Vadivu

Reg. No.: 71205621046

Of

Kumaraguru College of Technology

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

For the award of the degree

Of

MASTER OF COMPUTER APPLICATIONS

ANNA UNIVERSITY
CHENNAI 600 025

July, 2008

BONAFIDE CERTIFICATE

Certified that this project report titled **"Job Scheduling in Grid Computing Environment"** is the bonafide work of **Ms. M.Shanmuga Vadivu** (Registration Number: 71205621046) who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate

SUPERVISOR

HEAD OF THE DEPARTMENT

Submitted to Project and Viva Examination held on 01.07.2008

INTERNAL EXAMINER

EXTERNAL EXAMINER

[redacted]

[redacted]

[redacted]

[redacted]

- The **Project** is satisfactorily [redacted]
- All materials supplied [redacted]
- have been returned [redacted]

Thanks

For Amary Technologies

Lakshmi
Project Manager

ABSTRACT

The project "Job Scheduling in Grid Computing Environment" is to allocate jobs to various clusters connected in the grid environment. The administrator can view the details of jobs in a particular cluster,number of engines available,number jobs feed, and the status of jobs.

Job Feeding and job processing on various cluster on Distributed computing environment in secured way in the primary goal of the project.

Development and enhancement of Application, which leads to complete Grid Conceptualization of Distributed Grid computing process on terms of cluster, engines, job feeders, and web Front end for administration.

These all above modules run of Distributed operating system over the large computers with cluster and engines to analyze the needs of resource, which will be displayed graphically.

Development and enhancement of application on grid computing architecture for feeding jobs parallel execution on distributed computing Environment on various clusters with secured job scheduling over the internet implementation of Risk-Resilient Heuristics and Genetic Algorithms for security assured Grid job scheduling.

We addressing Grid heterogeneity problem, algorithms in our project have rectified security assurance and risk resilient strategies.

Using Heuristic algorithm, for each job, the resource site that gives the earliest expected completion time is determined first. The job that has the minimum earliest expected completion time is determined and then assigned to the corresponding site.

Genetic Algorithms (GAs) are a popular technique used for searching large solution spaces. The operations involved are 'selection', 'crossover', and 'mutation' operations.

- Selection → keep good solutions
- Crossover → global optimization
- Mutation → local jumping

The modules involved in this project are

- Enhanced Cluster.
- Scheduling strategy for Job Feeder.
- Risk Resilient scheduling for Job Engines.
- Secured Computing job allocators.
- Dynamic Allocation through web services.
- Enhanced Web interface

The project is used by the system administrator to allocate the job and monitor the job process and delete the jobs that are completed. The engine are used to monitor the clusters and the engines feed the job to particular cluster and monitors whether the job is executed and checks for arrival of new jobs.

ACKNOWLEDGEMENT

First and foremost I thank God for his good will and blessings showered on me throughout the project. The success of this project needs cooperation and encouragement from different quarters. Words are inadequate to express my profound and deep sense of gratitude to those who helped me in bringing out this project successfully.

I wish to express my deep unfathomable feeling of gratitude and indebtedness to **Dr. Joseph V. Thanikal, Principal, Kumaraguru College of Technology, Coimbatore** for the successful completion of the project work.

I am very gladly taking this opportunity to express a special word of thanks to **Dr. M. Gururajan, Head of the Department, Department of Computer Applications,Kumaraguru College of Technology, Coimbatore** for encouraging me to do this work.

I am very much indebted to **Mr S.Hameed Ibrahim, Senior Lecture, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore** for his assistance, guidance and support given to me throughout my project.

I am very much indebted to **Mr M.Manikantan, Senior Lecture, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore** for his complete assistance, guidance and support given to me throughout my project.

It's always a pleasure and privileges to be associated with a prestigious outstanding esteemed organization **"Xmarv Technologies "**, Coimbatore. My hearty thanks to my Project mentor **Mr.Kumar [Team Leader]** of **Xmarv Technologies**, for their valuable guidance throughout the project. Also, I am grateful to my parents and friends who were the real source of my project.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

CHAPTER 1

INDRODUCTION

1.1 ORGANIZATION PROFILE

"XMARV TECHNOLOGIES" is one among the fasted growing information Technology in the country. We are into service since august 2002, operating our registered office at Coimbatore and branch offices at Chennai and Trichy. Located in the heart of the Coimbatore and is easily accessible from all corners of the city by all modes of transportation.

As an information technology consultant we do software development mainly on utility packages, mini ERPS and technical services and assistances in legacy mainframe technology, data ware housing, WAP technology. Pollux R&D team in software takes crucial leads in forecasting technologies that are emerging in the IT field.

The objective of the company is to develop right product driven by new technology and should be shipped to the client at the right time. As a management provide service in human resource management, business performance improvement, strategic planning, corporate restructuring and much more related services. Strong professional team of technical and management personals carry these two broad categories of our services.

Once in a year a program is organized to enable the project trainees to acquire prevailing state of art knowledge and skill, in basic disciplines and functional areas of business computing and to develop analytical and innovate attitudes and skills so as to facilitate changes and increase the effectiveness and efficiency of the organizational systems.

1.2 ABSTRACT

The main aim of the system is to allocate the job to cluster in grid environment. The engine is used to monitor the jobs, allocates the job to available cluster,checks the status of the jobs. The administrator can delete the job using web front end and can view the status of all jobs in a particular cluster. Development and enhancement of Application, which leads to complete Grid Conceptualization of Distributed Grid computing process on terms of cluster, engines, job feeders, and web Front end for administration.

This project works on grid computing architecture for feeding jobs parallel execution on distributed computing Environment on various clusters with secured job scheduling over the internet implementation of Risk-Resilient Heuristics and Genetic Algorithms for security assured Grid job scheduling.

1.3 PROBLEM DEFINITION

In existing scenario the job feeding often fails and the process has to be rescheduled. The clusters available are less and allocating job to these clusters is time consuming. The process speed of each cluster is slow and jobs get crashed and jobs get deleted. The security is low, so those unauthorized jobs enter into the environment.

In this project job feeding is done in a secured way using the algorithms Heuristics and Genetic Algorithms. A connection is established to the cluster first then a particular job is feed to that cluster based on it processor availability. The engine at a periodic time interval checks for whether there is any new jobs arrived. If new jobs has arrived then it feeds those jobs to the available cluster. The web front end form is used to view the status of the job of a particular cluster. The user can delete a job from the web front end form. This is mainly designed for web usage. The details of clusters,jobs,processor and engines are all monitored by the administrator. Grids distributed resource clusters work at different autonomous domain

SYSTEM ANALYSIS

CHAPTER 2

SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

- Unexpected behavior of cluster, engine is probable.
- Insecurity might occur on situations like distributed job allocation on engines through feeds.
- Grid Job scheduler is less security driven and resilent in response to all risky situation
- Most of conditions jobs refuse to get allocated by scheduler and expected to die.
- Inconsistency is major problem in Computational environment.

2.2 PROPOSED SYSTEM

- Our new proposed system is vulnerable to security threats and network congestion.
- Large scale computational of Grid is possible and large number of user jobs can be allocated to the dispersed sites or other large scale distributed system.
- Grids distributed resource clusters work at different autonomous domain.
- Grid job scheduler is highly security driven and resilent in all risky conditions.
- Three-risk resilient strategies have been effectively implemented namely preemptive, replication based and delay tolerant.
- To make the system more reliable and consistent we are implementing scheduling strategy, heuristic algorithms and genetic algorithms.
- Grid server is highly persistent as it will implement using XML.
- Web services have been integrated with our system for wide area distributed computational grid system.

2.3 USER INTERFACE REQUIREMENTS

User Interface is required for the following modules.

- **Enhanced Cluster:** As cluster form main source of secured job execution through job feeders, job processors and Job engines. We are implementing highly sensitive and less delay tolerant cluster design, which in turn communicate with job processors and job feeders, which are distributed in multiple Grid sites.

- **Scheduling strategy for Job Feeder:** Replicated mode, risky mode and delay tolerant mode has been effectively implemented on **genetic and heuristic algorithms** to form newer strategies for job feeding on globally distributed clusters.

- **Secured Computing job allocators:** Secured job scheduling through job allocator is accomplished by SOAP protocol, which adds entire security to job feeders, clusters and job engines or processors.

- **Risk Resilient scheduling for Job Engines:** Engines in other way have been designed to acquire jobs from clusters in variable of their location. Here engines are very sensitive to grid server to acquire job from clusters to perform online job scheduling model against job failure and delay tolerant model.

- **Dynamic Allocation through web services:** Dynamic allocation of jobs from job feeders over internet and intranet is implemented through web services. Connection across the network and internet between various grid components is established by web services for dynamic security and SOAP for security.

- **Enhanced Web interface:** Advanced web interface will be useful to monitor the status of clusters, number of engines , pending jobs, memory and processor usage on every grid component . Web interface will communicate with grid clusters to retrieve the information on each grid component status at regular interval of time.

DEVELOPMENT ENVIRONMENT

CHAPTER 3

DEVELOPMENT ENVIRONMENT

3.1 HARDWARE ENVIRONMENT

Main Processor	– INTEL PENTIUM IV
RAM	– 512 MB
Hard Disk Capacity	– 40 GB
Monitor	– HP17"
Keyboard	– HP 106 keys
Mouse	– HP Optical Scroll Mouse

3.2 SOFTWARE ENVIRONMENT

Operating System : Windows 2000

Environment : Visual Studio .Net 2005.

Programming Language: C#, VB.Net and ASP.NET.

3.3 PROGRAMMING ENVIRONMENT

3.3.1 C#.Net

Visual C# is a simple, type-safe, object-oriented, general-purpose programming language. It provides code-focused developers with powerful tools and language support to build rich, connected Web and client applications on the .NET Framework.

3.3.1.1 C#.Net Overview

- Visual C# 2005 is the modern, innovative programming language and tool for building .NET-connected software for Microsoft Windows, the Web, and a wide range of devices.
- With syntax that resembles C++, a flexible integrated development environment (IDE), and the capability to build solutions across a variety of platforms and devices, Visual C# 2005 significantly eases the development of .NET-connected software.

3.3.1.2 Features of C#.net

- Immediately familiar to C++ and Java developers, C# is a modern and intuitive object-oriented programming language that offers significant improvements, including a unified type system, "unsafe" code for maximum developer control, and powerful new language constructs easily understood by most developers.
- An advanced inheritance model enables developers to reuse their code from within any programming language that supports .NET.
- .NET Framework class library to gain powerful built-in functionality, including a rich set of collection classes, networking support, multithreading support, string and regular expression classes, and broad support for XML, XML schemas, XML namespaces, XSLT, XPath, and SOAP.

- Using Visual C# 2005, developers can construct powerful Web services that encapsulate business processes and make them available to applications running on any platform.
- Visual C# 2005 also enables developers to build the next generation of Windows-based applications. With visual inheritance, developers can greatly simplify the creation of Windows-based applications by centralizing in parent forms the common logic and user interface for their entire solution.
- With native support for the .NET Compact Framework, mobile Web devices, and embedded applications available as part of Visual Studio 2005 Professional Edition, C# developers can now target a wide variety of mobile devices, including Pocket PCs, mobile phones, and devices powered by the Windows CE operating system. Programmers can become immediately productive by using the same programming model and tools for creating powerful device-based software as they use for building robust Windows- and Web-based solutions.

3.3.2. VB.NET

Visual Basic.NET is a complete re-engineering of Visual Basic for the Miscrosoft .NET framework. Visual Basic.NET Windows applications are event-driven. Visual Basic.NET can be complied using Visual Studio.NET or vbc.exe, a command-line complier supplied with the .NET Framework.

3.3.2.1 VB.NET Overview

- With Visual Basic.NET you are able to quickly build Windows-based application, web-based application and eventually, software for other devices, such as palm computers.
- Visual Basic.NET uses Windows 2000,Windows XP or Windows NT.

- Visual Basic.NET is governed by an event processor. Once an event is detected, corresponding event procedure is located and the instructions provided by that procedure are executed.

3.3.2.2 Features of VB.NET

- All new,easy-to-use,powerful Integrated Development Environment(IDE)
- Full set of controls – you 'draw' the application
- Response to mouse and keyboard actions
- Clipboard and printer access
- Full array of mathematical, string handling, and graphics functions
- Can easily work with arrays of variables and objects
- Sequential file support
- Useful debugger and structured error-handling facilities
- Easy-to-use graphic tools
- Powerful database access tools
- Ability to develop both Windows and internet applications using similar techniques
- New common language runtime module makes distribution of application a simple task.
- Improved support to incorporating help systems in applications

SYSTEM DESIGN

CHAPTER 4

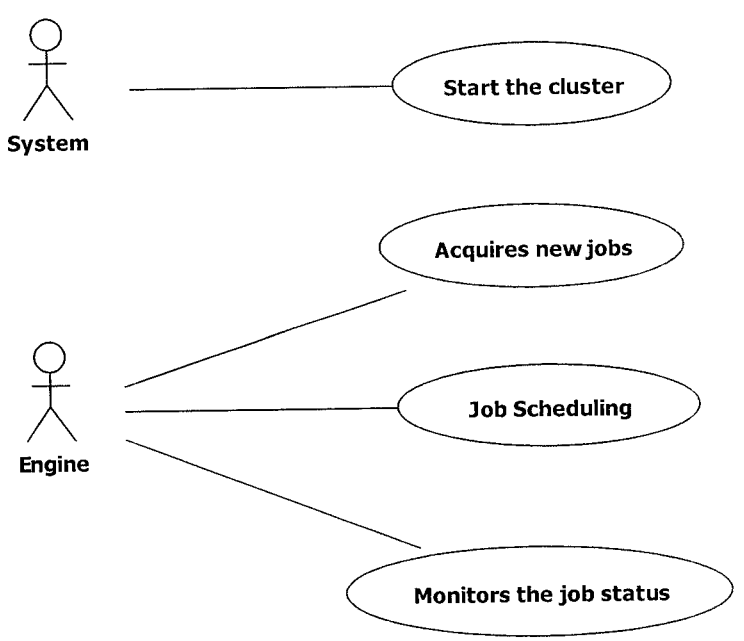
SYSTEM DESIGN

4.1 PROCESS MODEL

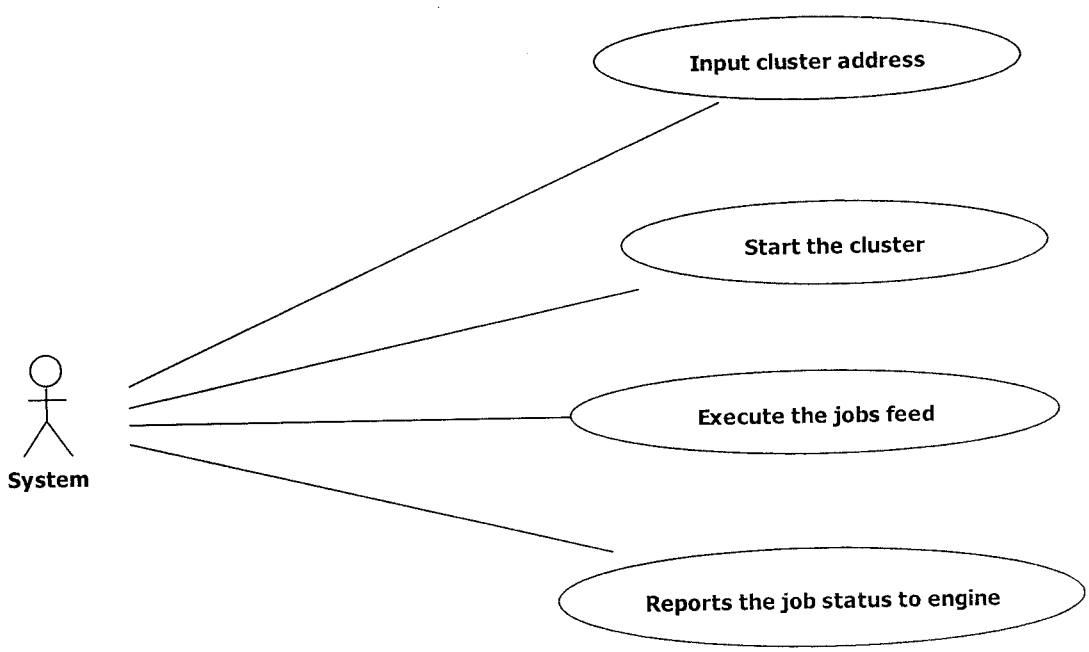
4.1.1 Usecase Diagram

A **usecase diagram** is used to present a graphical overview of the functionality provided by a system in terms of actors, their goals, represented as use cases and any dependencies between those use cases.

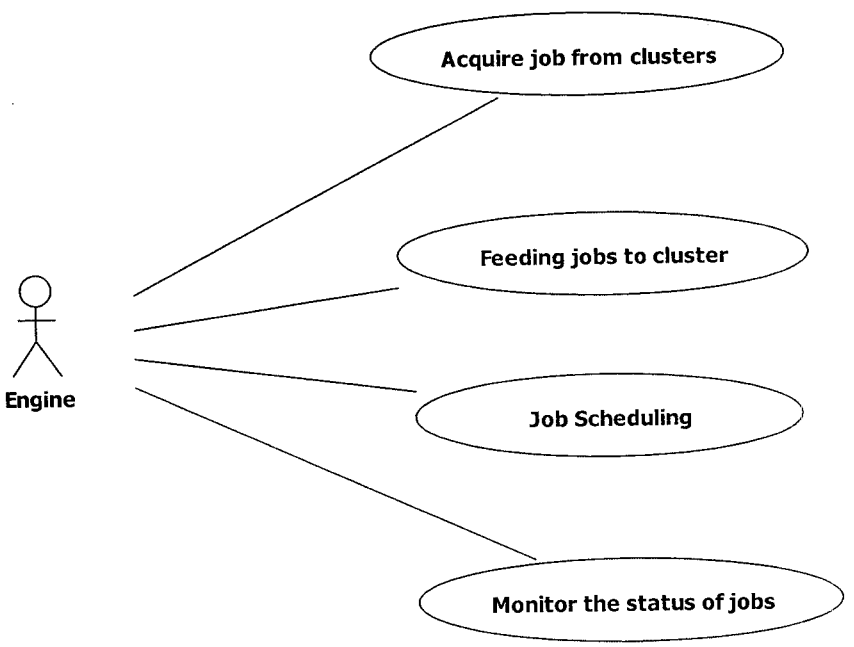
4.1.1.1 Main Use Case Diagram



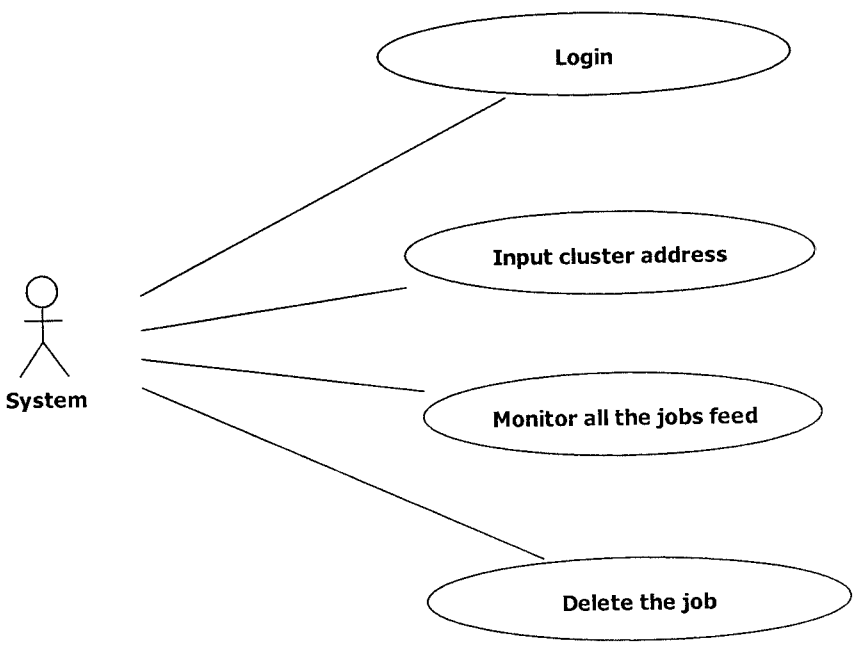
4.1.1.2 Enhanced Cluster Use Case Diagram



4.1.1.3 Job Scheduling Use Case Diagram



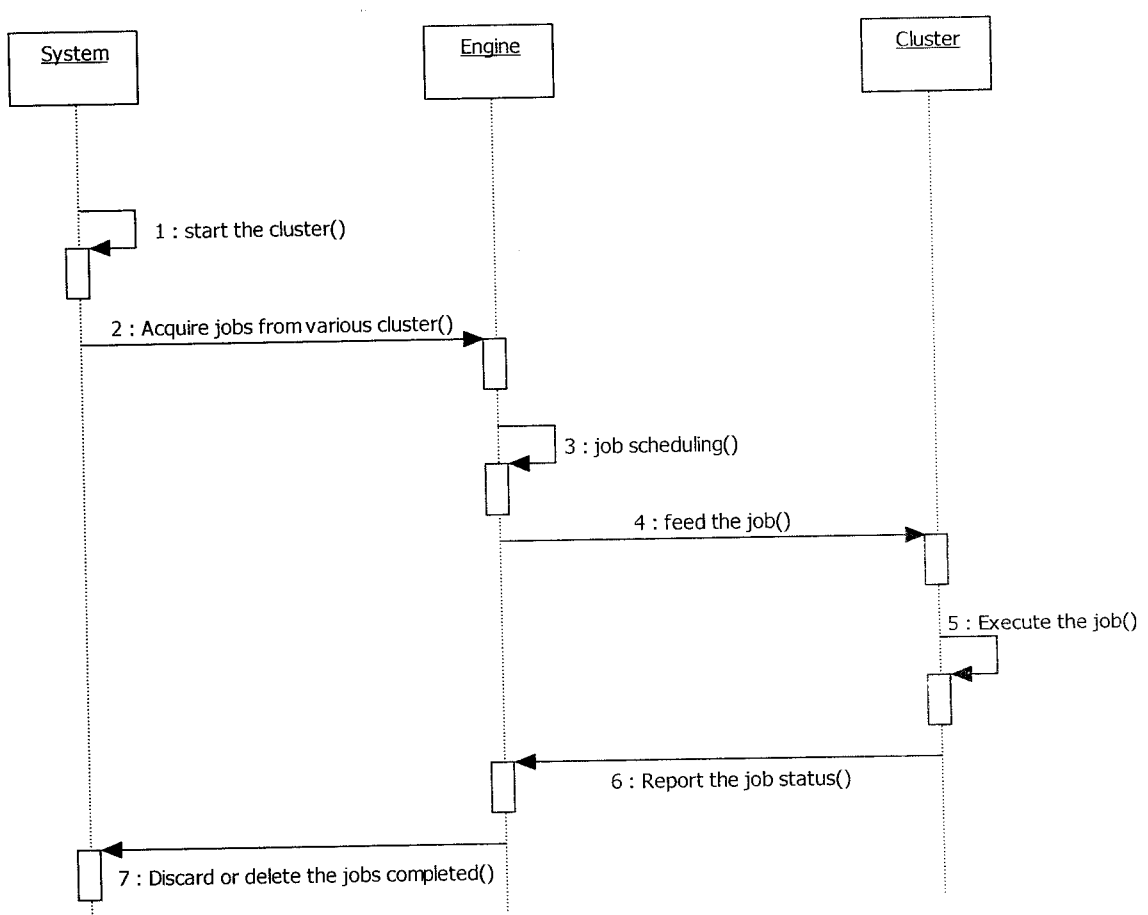
4.1.1.4 Web Interface Use Case Diagram



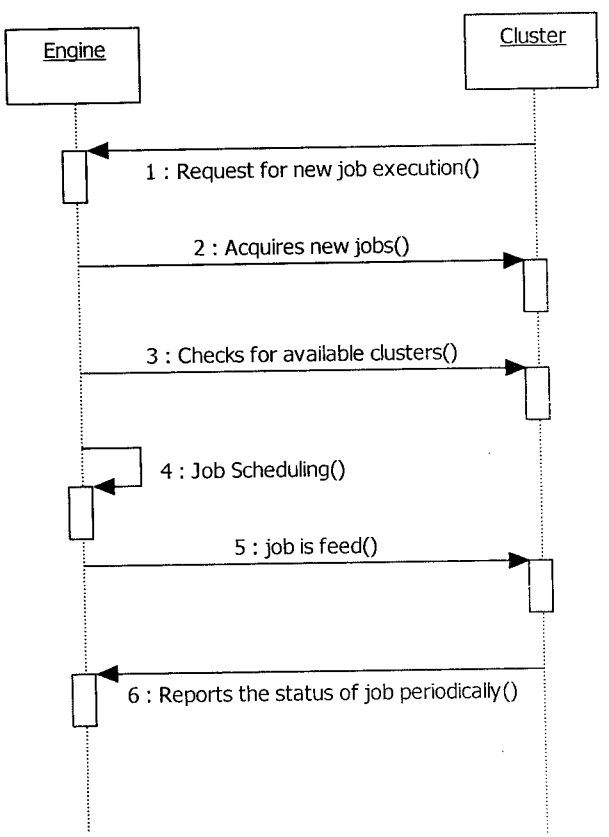
4.1.2 Sequence Diagram

A sequence diagram is a form of interaction diagram which shows objects as lifelines running down the page, with their interactions over time represented as messages drawn as arrows from the source lifeline to the target lifeline. Sequence diagrams are good at showing which objects communicate with which other objects; and what messages trigger those communications.

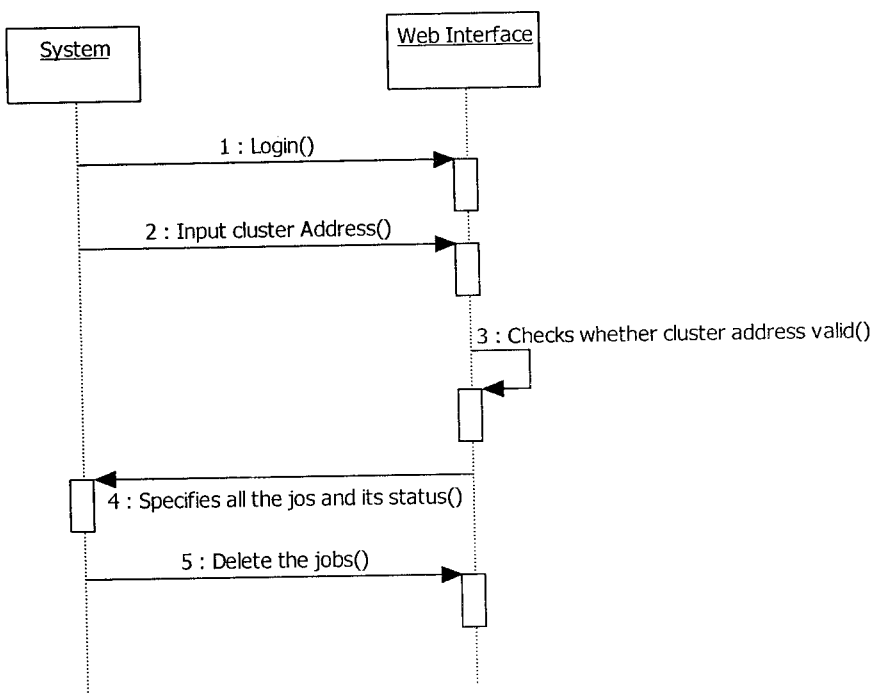
4.1.2.1 Main Sequence Diagram



4.1.2.2 Job Scheduling Sequence Diagram



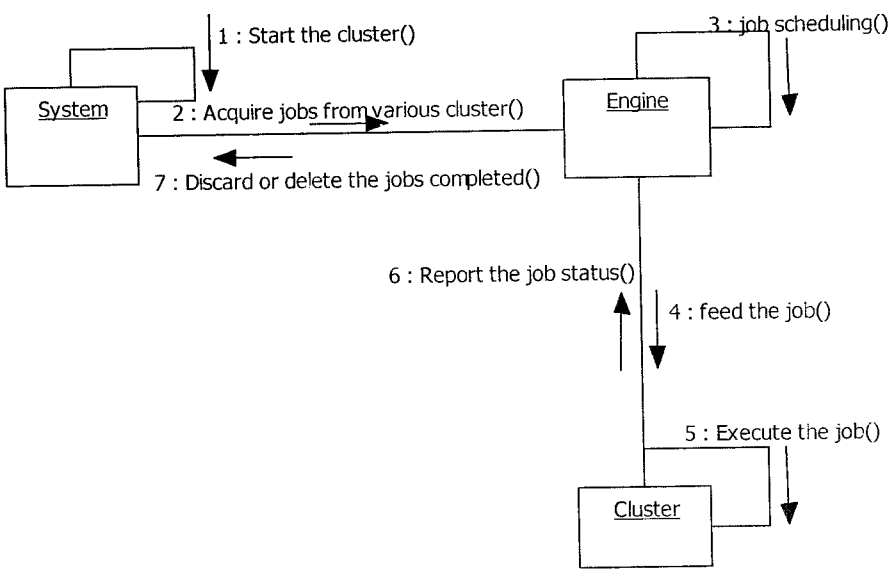
4.1.2.3 Web Interface Sequence Diagram



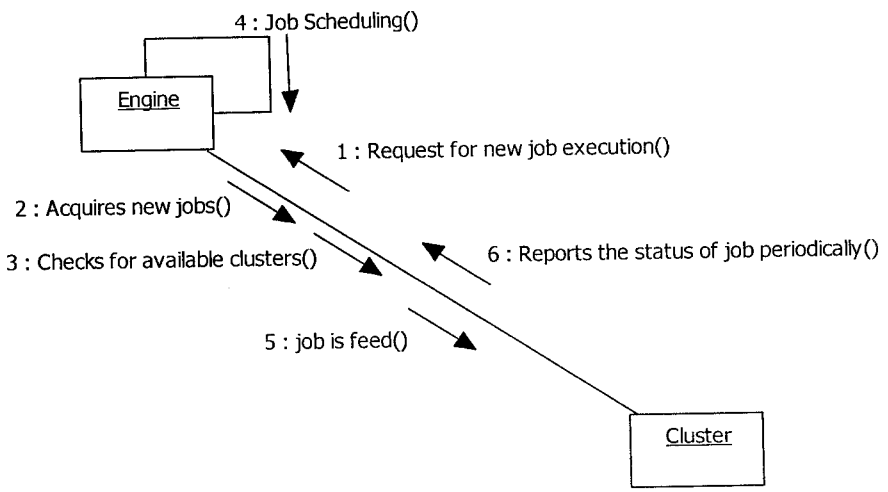
4.1.3 Collaboration Diagram

Collaboration diagrams, like Sequence Diagrams, show how objects interact over the course of time. However, instead of showing the sequence of events by the layout on the diagram, collaboration diagrams show the sequence by numbering the messages on the diagram. This makes it easier to show how the objects are linked together, but harder to see the sequence at a glance.

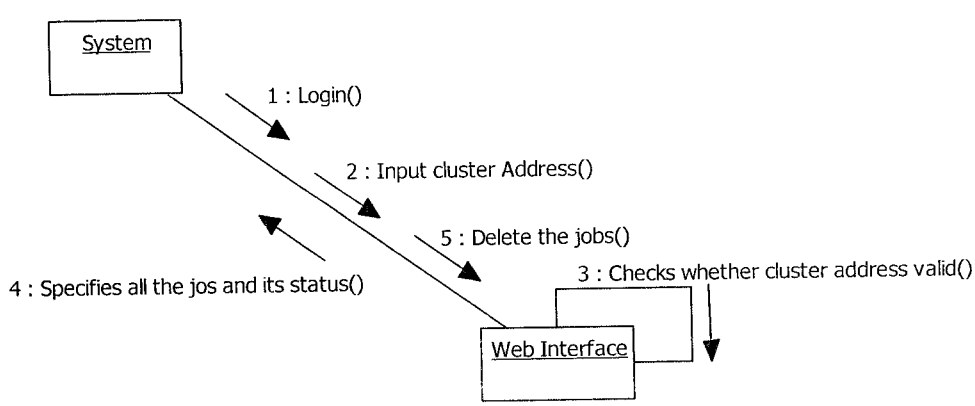
4.1.3.1 Main Collaboration Diagram



4.1.3.2 Job Scheduling Collaboration Diagram

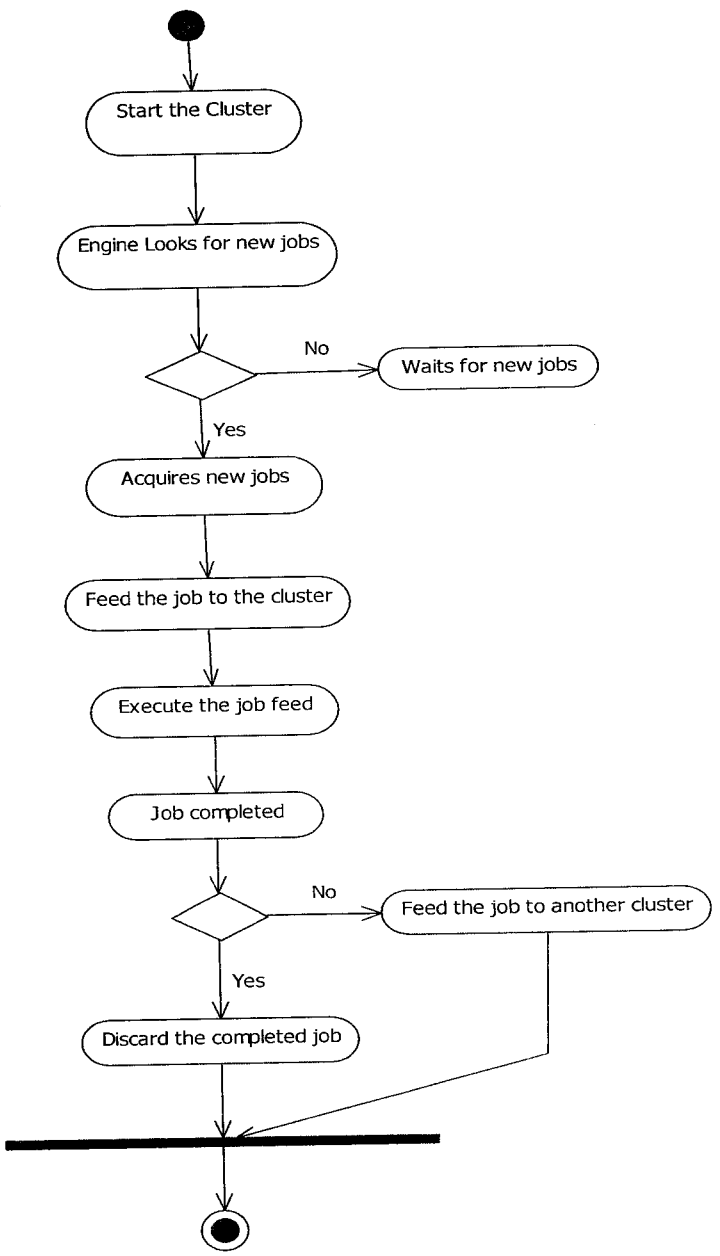


4.1.3.3 Web Interface Collaboration Diagram



4.1.4 ACITIVITY DIAGRAM

Activity diagrams describe the workflow behavior of a system. Activity diagrams are similar to state diagrams because activities are the state of doing something. The diagrams describe the state of activities by showing the sequence of activities performed. Activity diagrams can show activities that are conditional or parallel.



ARCHITECTURAL DETAILS

CHAPTER 5

5.1. General Architecture of the Grid Computing System:

General architecture of the grid computing system is shown in figure 5.1. is logically divided into three layers – the application layer, the grid resource management layer, and the grid resource layer.

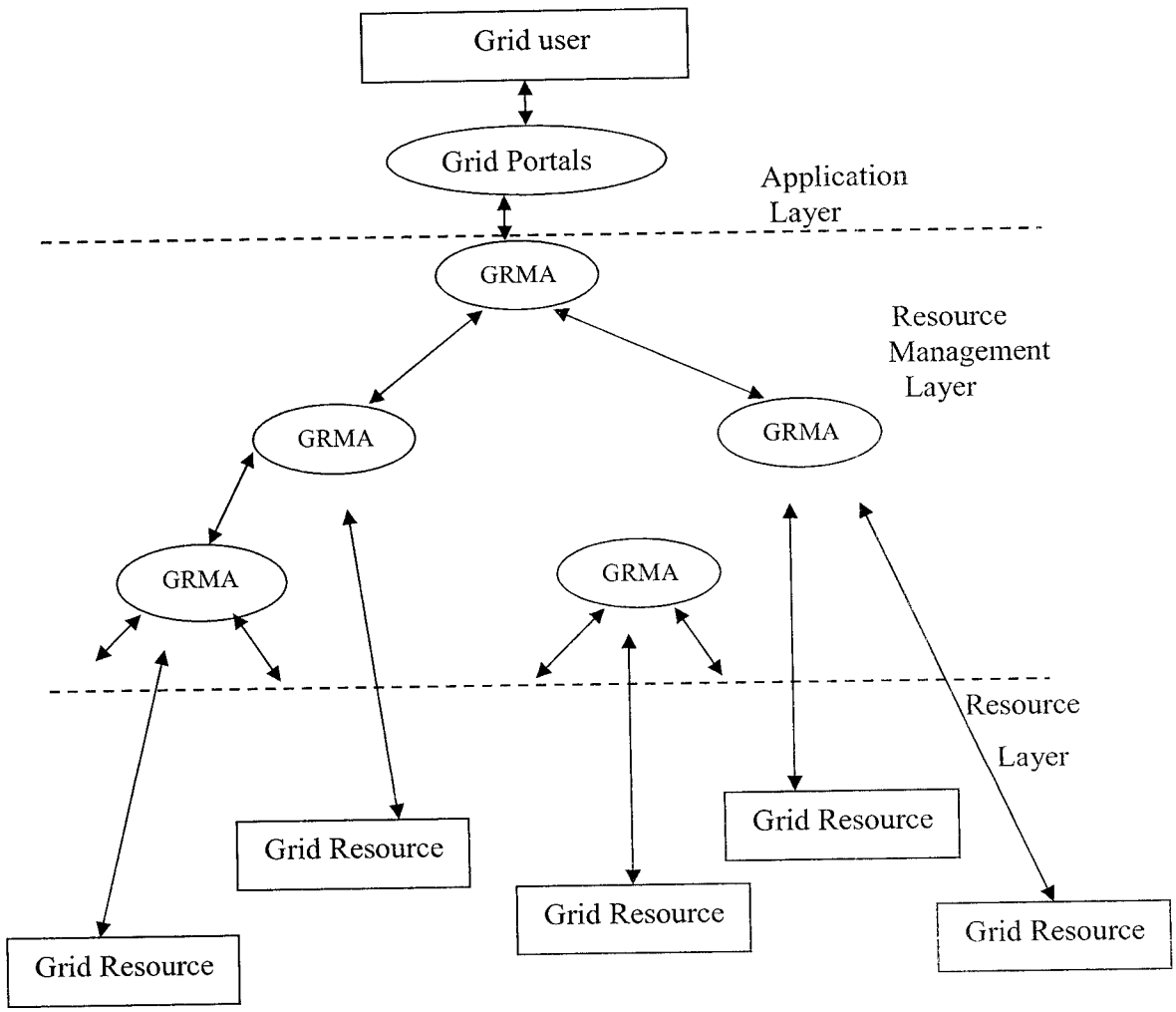


Figure 5.1: General Architecture of the Grid Computing System

In the application layer, grid users send their jobs to the system through the portal of the grid system. The portal then sends the job to the resource management system. In succession, the grid resource management system selects the proper computational resources from the grids and submits the job to it to be fulfilled. Finally, the selected grid resources carry out the scheduled job and return the result to the requestor. The resource management layer is the crucial part of the whole grid computing system, and it is logically composed of *Grid Resource Management Agents* (GRMA).

TESTING

CHAPTER 6

TESTING

Software testing is the process used to assess the quality of computer software. It is an empirical technical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate. This includes, but is not limited to, the process of executing a program or application with the intent of finding software bugs. Software testing methods are traditionally divided into **black box testing and white box testing.**

Black box testing treats the software as a black-box without any understanding of internal behavior. It aims to test the functionality according to the requirements. Thus, the tester inputs data and only sees the output from the test object. **White box testing**, however, is when the tester has access to the internal data structures, code, and algorithms. White box testing methods include creating tests to satisfy some code coverage criteria.

Testing can be done on the following levels:

- Unit Testing
- Integration Testing
- System Testing

Unit Testing

Unit testing tests the minimal software component, or module. Each unit (basic component) of the software is tested to verify that the detailed design for the unit has been correctly implemented.

Enhanced Cluster:

Sl No	Test Case	Test Procedure	Pre-condition	Expected result	Status
1	Cluster_ address_in valid_input	Give some Invalid integer input as '$^a','as3','@#1 2','-345'	None	It should give customized error message such as "You have entered some wrong address "	Passed
2	Cluster_ address_Va lid_Input	Give Some Valid Input as 'http://localhost:8 097/mygird'	None	Should accept the given input	Passed
3	Cluster_ address _Invalid Input	Give some Invalid Character as 'aq1','as@','@% ' and null values.	None	It should give customized error message such as "You have entered some wrong address "	Passed
4	Cluster_ address _NULL_In put	Give null values as input	None	It should give customized error message such as "enter the cluster address"	Passed
5	Webservice _address_I nvalid_Inp ut	Give some Invalid address as 'aq1','as@','@% ' and null values.	None	It should give customized error message such as " You have entered some wrong data "	Passed
6	Webservice _address _Valid_Inp ut	Give Some Valid Input as 'http://localhost7/ mygirdweb/mygr id.asmx'	None	Should accept the given input	Passed
7	Connect_to _Grid_inva lid_input	Click the button connect to grid when there is invalid cluster or webservice address	None	It should give a error message such as "connection is not established between the cluster and the web service"	Passed
8	Connect_to _Grid_inva lid_input	Click the button connect to grid when connection	None	It should give a warring message such as	Passed

		is already established cluster or web service address		"connection already established between the cluster and the web service"	
9	Connect_to_Grid_valid_input	Click the button connect to grid when there is valid cluster or web service address	None	It should give a customized message such as "connection is established between the cluster and the web service"	Passed

Scheduling strategy for Job Feeder:

Sl No	Test Case	Test Procedure	Pre-condition	Expected result	Status
1	Job_feed_Invalid_Input	Feeding the same job to same cluster more than once	None	It should give customized error message such as " the job is already feed "	Passed
2	Job_feed_invalid_Input	Feeding the unauthorized job to a cluster.	None	It should give customized error message such as " the job is unauthorized"	Passed
3	Job_feed_invalid_Input	Feeding the job to a cluster but it takes more time than expected time.	None	Job should be allocated to some other cluster	Passed
4	Job_feed_valid_input	Feeding the job to a particular cluster	None	Job is allocated to the cluster	Passed

Risk Resilient scheduling for Job Engines:

Sl No	Test Case	Test Procedure	Pre-condition	Expected result	Status
1	Engine_ address_in valid_input	Give some Invalid input as '$^a','9as','@#12','-345'	None	It should give customized error message such as "You have entered some wrong address "	Passed
2	Engine_ address_Valid_Input	Give Some Valid Input as 'engine11'	None	Should accept the given input	Passed
3	Engine_Job _acquire_invalid_Input	Engine acquires unauthorised job	None	It should give customized error message such as " the job is unauthorized"	Passed
4	Engine_Job _acquire_invalid_Input	Engine checks whether job feed takes more time than expected time.	None	Job should be allocated to some other cluster	Passed
5	Engine_Job _acquire_valid_input	Engine feeds the acquired job to the available cluster	None	Job is allocated to the cluster	Passed

Enhanced Web interface:

Sl No	Test Case	Test Procedure	Pre-condition	Expected result	Status
1	Username_ invalid_input	Give some Invalid input as '$^a','9as','@#1'	None	It should give customized error message such as "You have entered some wrong address "	Passed
2	Username_ valid_input	Give some valid input as 'mygrid'	None	The input should be accepted	Passed
3	Password_i nvaild_input	Give some Invalid input as '$^a','9as','@#1'	None	It should give customized error message such as "You have entered some wrong address "	Passed
3	Password_ vaild_input	Give some Invalid input as'Mypassword'	None	The input should be accepted	Passed
4	Engine_ address_in valid_input	Give some Invalid input as '$^a','9as','@#1 2','-345'	None	It should give customized error message such as "You have entered some wrong address "	Passed
5	Engine_ address_Va lid_Input	Give Some Valid Input as 'engine23'	None	Should accept the given input	Passed
6	Cluster_ address_in valid_input	Give some Invalid integer input as '$^a','as3','@#1 2','-345'	None	It should give customized error message such as "You have entered some wrong address "	Passed
7	Cluster_ address _Invalid Input	Give some Invalid Character as 'aq1','as@','@% ' and null values.	None	It should give customized error message such as "You have entered some wrong address "	Passed
8	Cluster_ address_Va lid_Input	Give Some Valid Input as 'http://localhost:8 097/mygird'	None	Should accept the given input	Passed

INTEGRATION TESTING

Integration testing is the phase of software testing in which individual software modules are combined and tested as a group. It follows unit testing and precedes system testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

S.No	Test Case	Input Data	Expected Result	Actual Result	Pass/Fail
1	Allocating a new job to a cluster	Job id, job processing speed and memory usage	Completion of job within expected time	The job status is periodically updated and completed	**Pass**

PERFORMANCE AND LIMITATIONS

CHAPTER 7

PERFORMANCE AND LIMITATIONS

7.1 MERITS OF THE SYSTEM

- The 'Job Scheduling in Grid Computing Environment' helps us to allocate the jobs to various cluster and jobs are completed quickly. It increases the efficiency and response time, thus making every customer happy.
- Faster acquiring of jobs by the engine
- Shorter processing times
- Increased accuracy and higher quality of service
- Requires less space to store

7.2 LIMITATION OF THE SYSTEM

Not Applicable

7.3 FUTURE ENHANCEMENT

Though the system has been developed to the complete satisfaction of the user, enhancements are always possible. The system is designed in such a way that new features can be added without much difficulty.

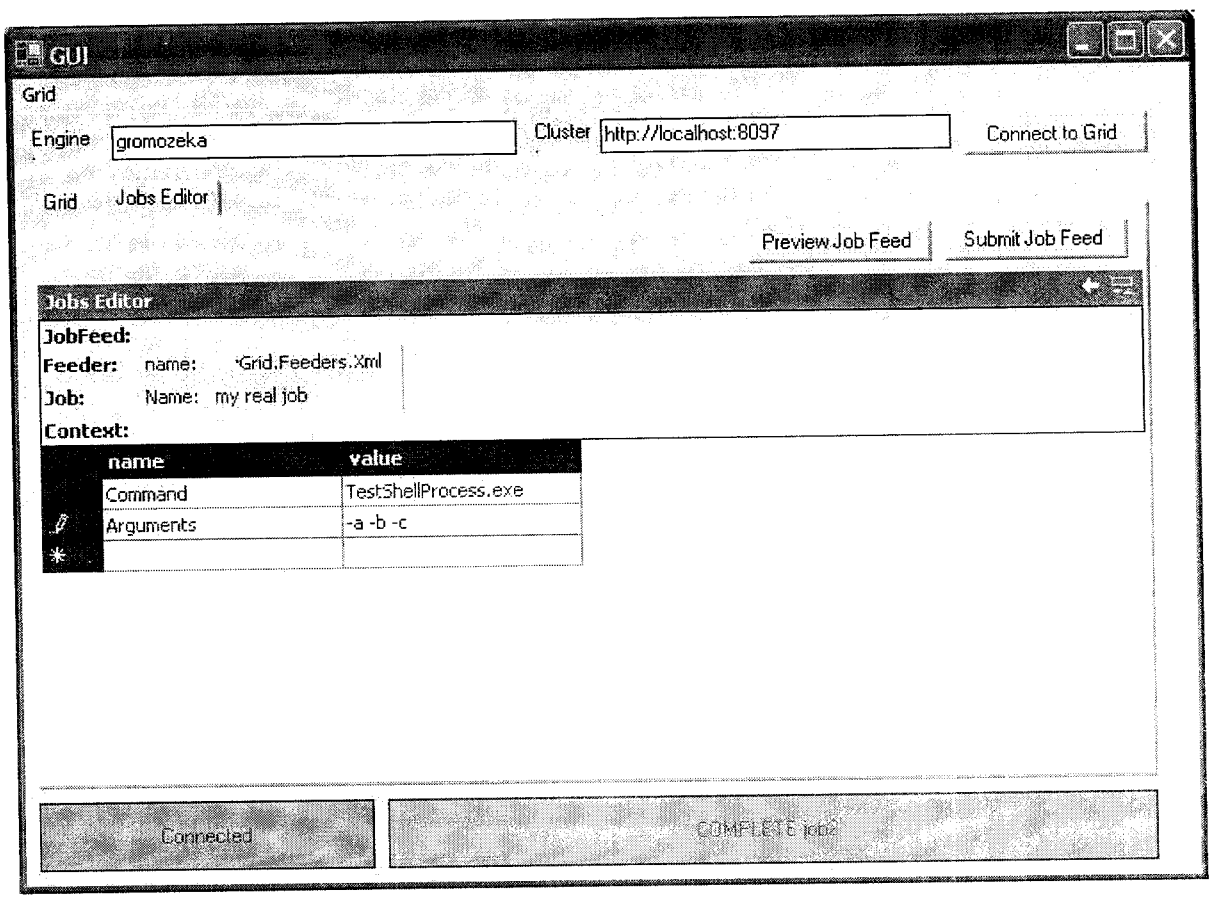
The system is developed in such away that if any modifications and enhancements are needed in the future, it can be done at ease, without disturbing the proper working of the system as we are following the MVC architecture.

The system can be changed easily depending on organizational policy constraints. The reconstruction of the system will increase the system flexibility. We have not provided means for the customer to interact with the system.

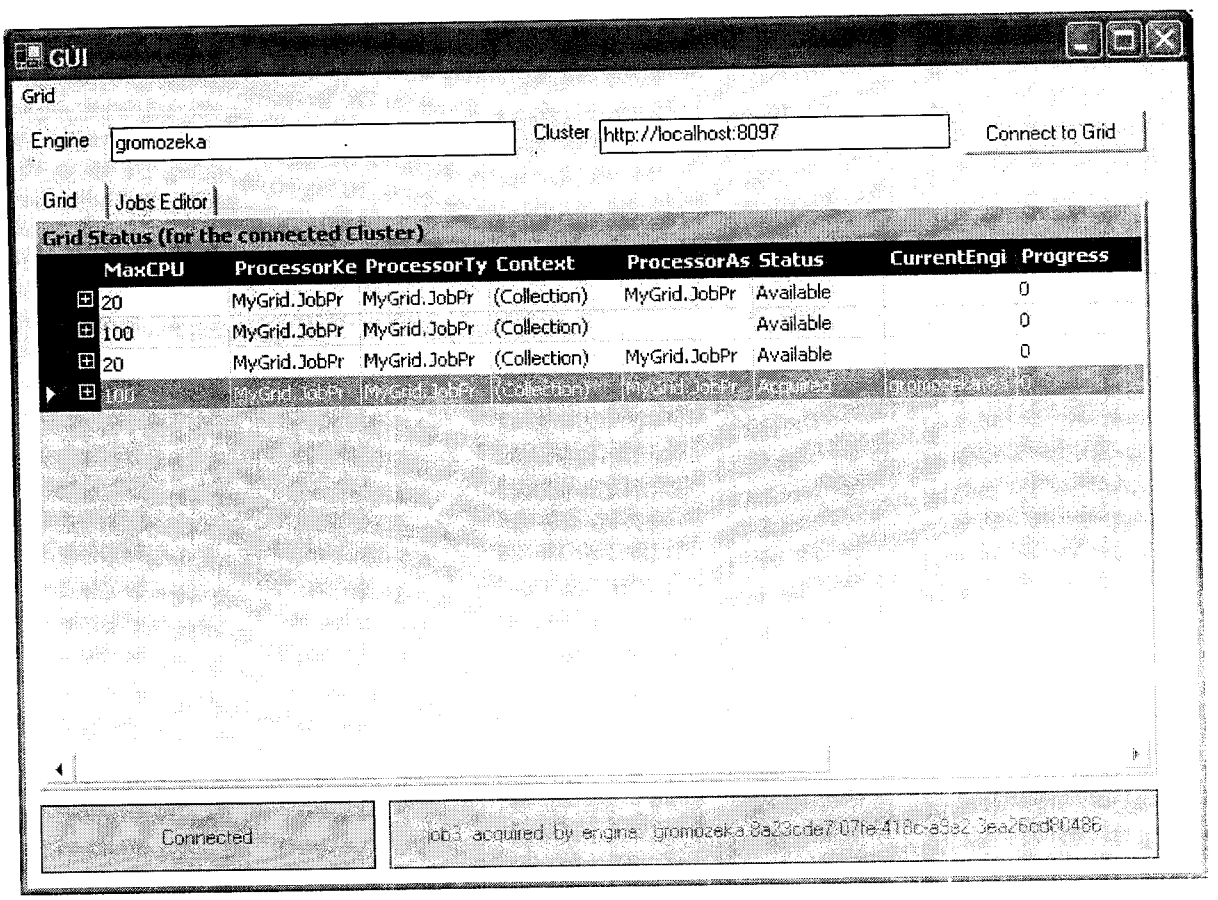
APPENDICES

Start the cluster:

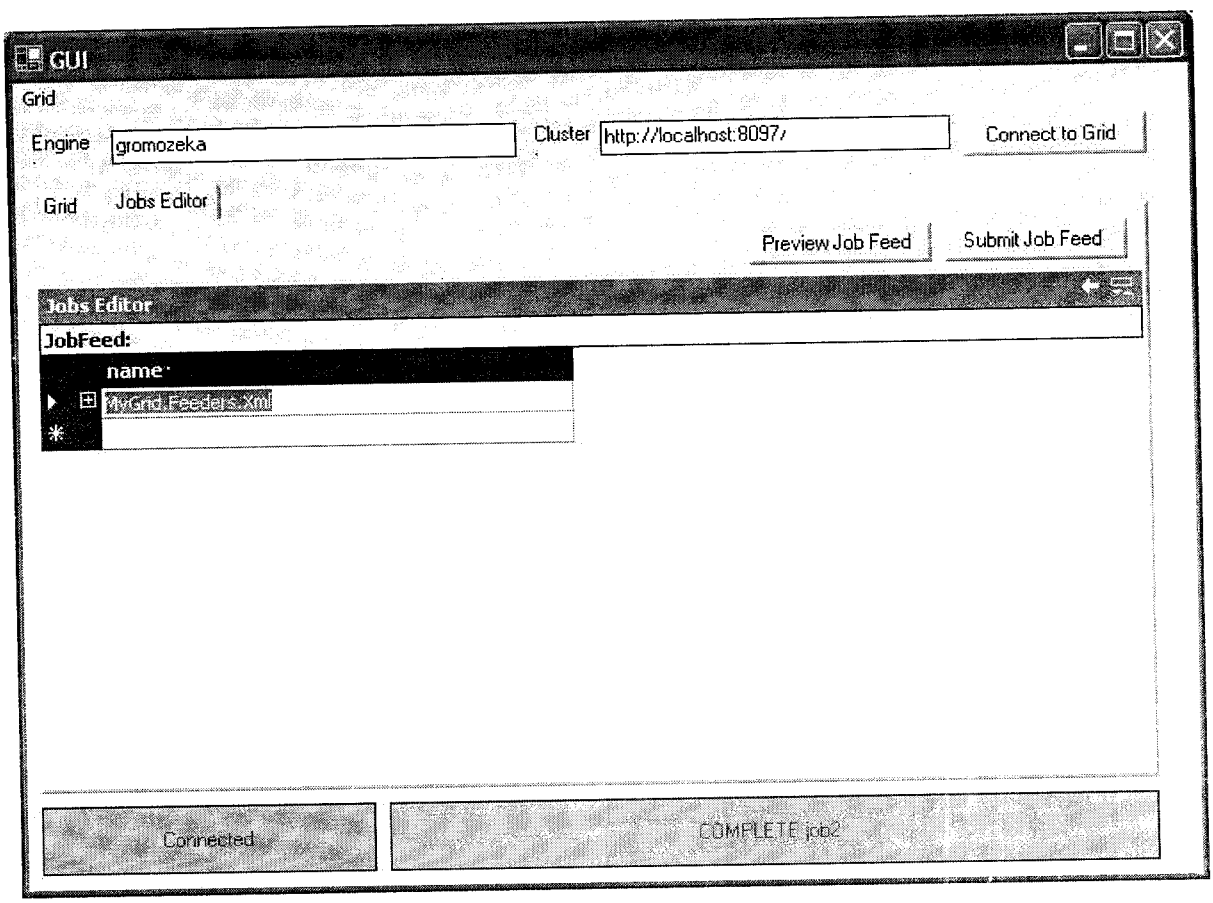
Job Editor:


GUI
Grid
Engine
gromozeka
Cluster
http://localhost:8097
Connect to Grid
Grid
Jobs Editor
Preview Job Feed
Submit Job Feed
Jobs Editor
JobFeed:
Feeder: name: Grid.Feeders.Xml
Job: Name: my real job
Context:
name
value
Command
TestShellProcess.exe
Arguments
-a -b -c
Connected
COMPLETE job2

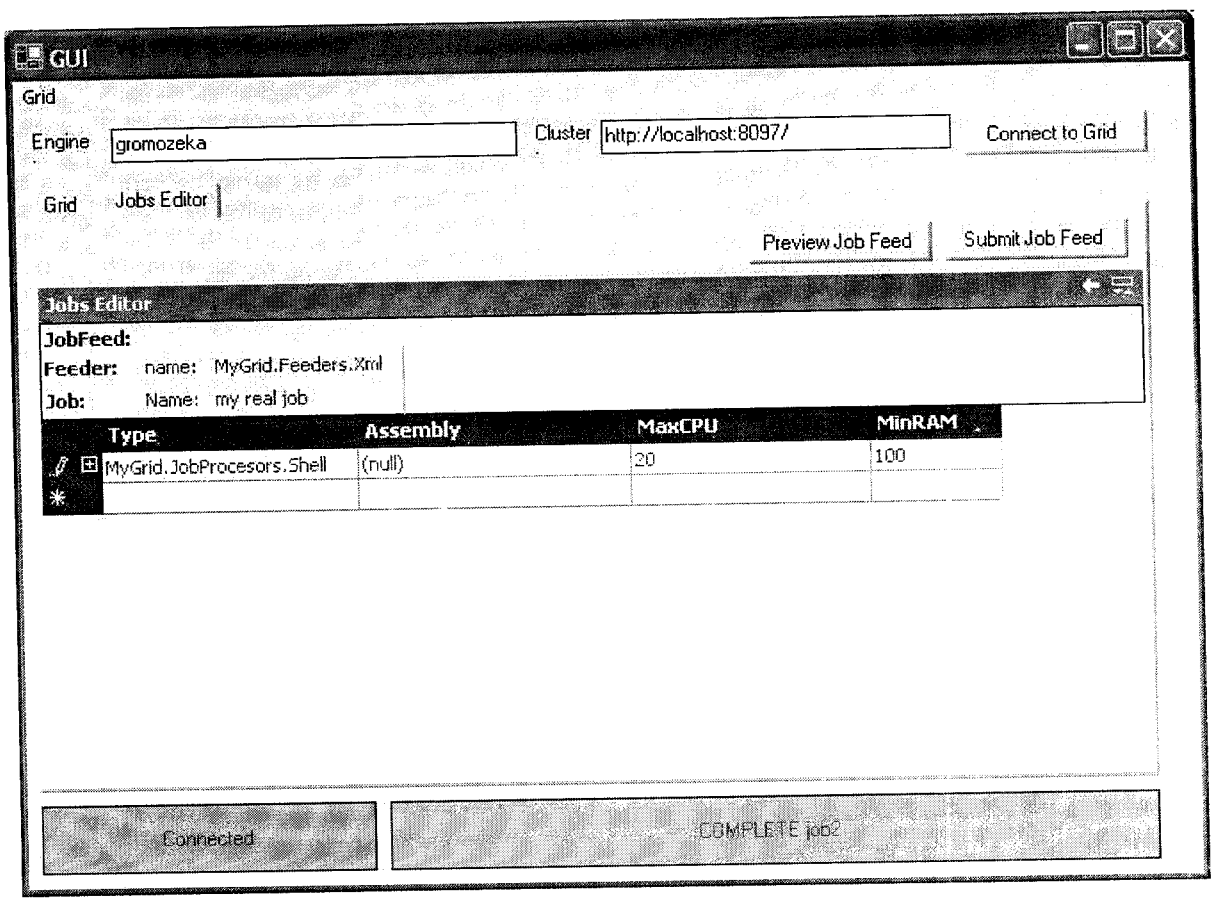
Job Feeder:


GUI
Grid
Engine
gromozeka
Cluster
http://localhost:8097
Connect to Grid
Grid
Jobs Editor
Grid Status (for the connected Cluster)
MaxCPU
ProcessorKe
ProcessorTy
Context
ProcessorAs
Status
CurrentEngi
Progress
20
MyGrid.JobPr
MyGrid.JobPr
(Collection)
MyGrid.JobPr
Available
0
100
MyGrid.JobPr
MyGrid.JobPr
(Collection)
Available
0
20
MyGrid.JobPr
MyGrid.JobPr
(Collection)
MyGrid.JobPr
Available
0
Connected

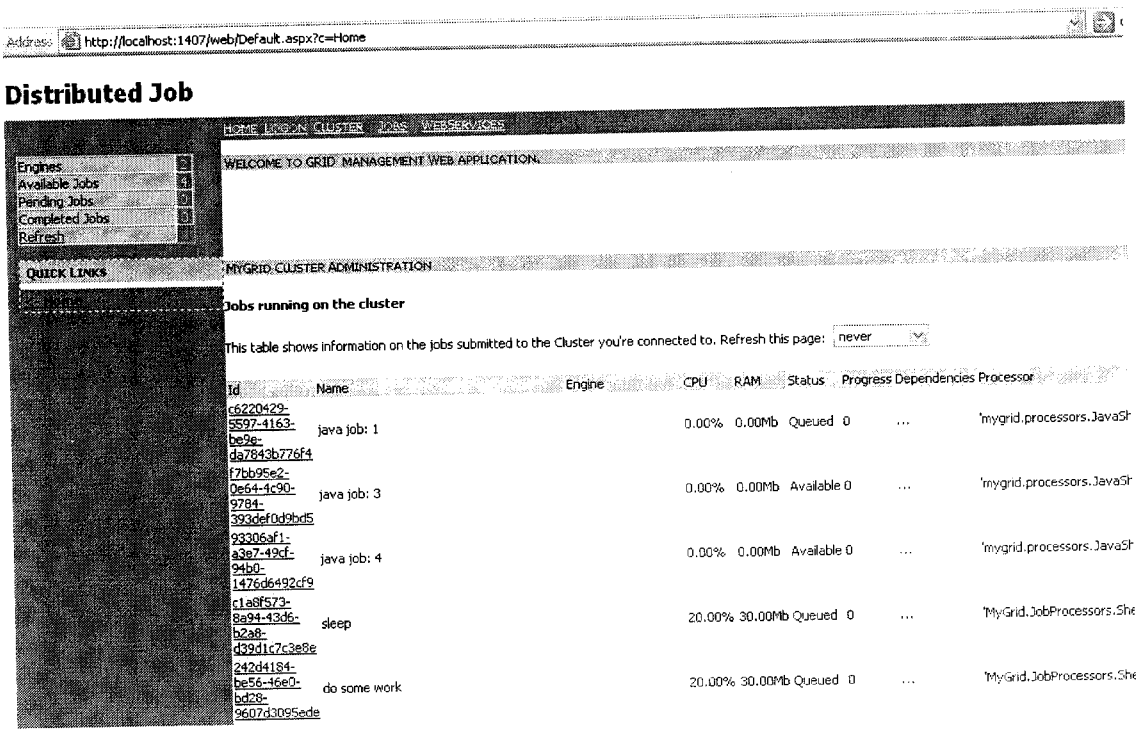
Job Feed by XML:



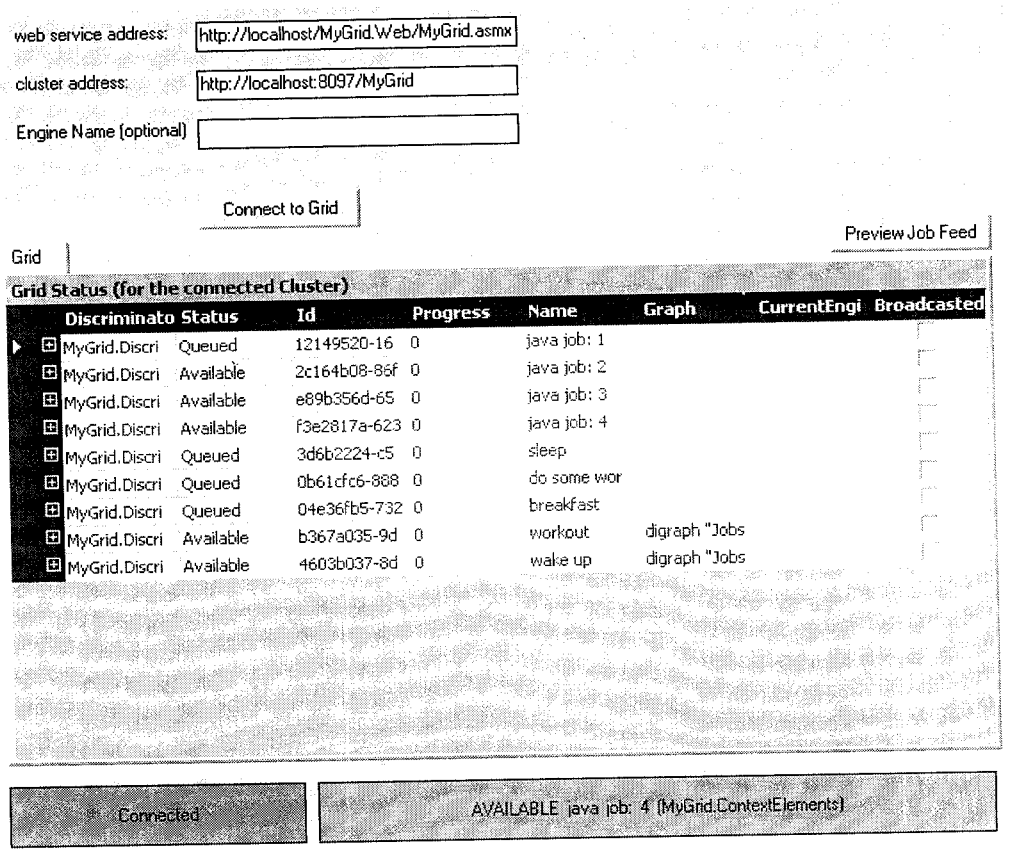
Job Processor :



Web FrontEnd:



GUI Application:



8.2 User Manual

This component 'Job Scheduling in Grid Computing Environment' is mainly used for job scheduling in the grid computing environment. This software runs on .Net framework using VB.NET,C#.NET. The job scheduling is done using Risk-Resilient Heuristics and Genetic Algorithms for security assured Grid job scheduling.

The GUI application helps to establish connection between the cluster and the web service easily and quickly. The cluster machine is executed first and then engine gets started. Once the engine is starts working,it keeps on monitoring for new jobs from various clusters and feeds the new job to the available cluster. The job feeder is the one who feeds job to the cluster,initially some job are feed to cluster by job feeder with the help of XML. When the cluster is executed first these jobs are feed initially.

Advanced web interface will be useful to monitor the status of clusters, number of engines , pending jobs, memory and processor usage on every grid component . Web interface will communicate with grid clusters to retrieve the information on each grid component status at regular interval of time.

REFERENCES

CHAPTER 9

REFERENCES

[1] Available: http://www.aksis.com.tr/d/p8_brochure.pdf, Accessed: 23 April, 2008.

[2] Available: http://www-935.ibm.com/services/us/its/pdf/en0012-281003.pdf, Accessed: 26 April, 2008.

[3] Available: http://www-03.ibm.com/industries/financialservices/doc/content/solution/1330948103.html, Accessed: 15 May, 2008.

[4] Available: http://www- www.C#,net-books.com /industries/financialservices/doc/content/solution/1330948203.html, Accessed: 19 May, 2008.

[5] XML , Available: http://www.simonstl.com/articles/whyxml.htm,Accessed:22 May, 2008

[6] Use case diagrams, Available: http://www.simonstl.com/UML/UMLdiagrams.htm, Accessed:24 May, 2008