CHAPTER 5

GMIS Processes and Interface Modeling with Experimentation Results

This chapter discusses about the GMIS framework processes and interface modeling with the help of process flow charts. It presents the typical usage of Inter Process Communication (IPC) library for GMIS components communication. It presents the sample screen shots of GMIS viewer. It discusses about the testing environment and GMIS simulator called GMIS-SIM to test the GMIS services. Finally it presents some experiments and results.

5.1 Framework Components Processes

A set of processes are developed to achieve GMIS functionality. The following table 5.1 shows mapping between GMIS components and processes:

<table>
<thead>
<tr>
<th>GMIS Component</th>
<th>Process name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Server</td>
<td>Resource Discovery Manager (RDM)</td>
</tr>
<tr>
<td>Data Collection Manager</td>
<td>Data Collection Manager (DCM)</td>
</tr>
<tr>
<td>Database</td>
<td>Data Handler (DH)</td>
</tr>
<tr>
<td>Topology Manager</td>
<td>Topology Information Process (TIP)</td>
</tr>
<tr>
<td>Monitor Service</td>
<td>Information Provider Service (IPS)</td>
</tr>
<tr>
<td>Resource Selector</td>
<td>Resource Broker Process (RBP)</td>
</tr>
<tr>
<td>Communicator Server</td>
<td>Communicator Server Process (CSP)</td>
</tr>
<tr>
<td>Agent Software</td>
<td>Communicator Agent (CA), Resource Discovery Agent (RDA), Critical Information Agent (CIA)</td>
</tr>
<tr>
<td>Viewer</td>
<td>Web Client</td>
</tr>
<tr>
<td>GMIS-GMIS Communication</td>
<td>Peer 2 Peer Communicator Service (P2PCS)</td>
</tr>
</tbody>
</table>

Table 5.1: Mapping of GMIS framework Components to GMIS Processes
To have control over these processes a process called “gmis_controller” process has developed. This process functionality is to start/stop GMIS processes and to monitor the GMIS processes. If any GMIS process is down then gmis_controller will restart that particular process. This process runs along with GMIS processes. Each GMIS process has a common functionality to handle “gmis_controller” messages. The following subsections show the development details of GMIS processes with process flow charts.

5.1.1 Resource Discovery Manager (RDM)

Resource Discovery Manager (RDM) process is developed for achieving Discover Server functionality of GMIS. RDM process receives resource discovery requests from the viewer and resource information from the communicator server. The following figure 5.1 shows the RDM process flow chart.

![Figure 5.1: Resource Discovery Manager Process flow chart](image)

5.1.2 Data Collection Manager (DCM)

Data Collection Manager process polls the resources and parses the responses. It also handles the events generated by the resources. This process flow chart is shown in the following figure 5.2.
5.1.3 Data Handler (DH)

Data Handler process is responsible to insert, update, delete and retrieve resource information from the GMIS database. The following figure 5.3 shows the data handler process flow chart.

Figure 5.2: Data Collection Manager Process flow chart

Figure 5.3: Data Handler Process flow chart
5.1.4 Topology Information Process (TIP)

Topology Information Process (TIP) creates, maintains resource object hierarchy. It creates MIB tree in a shared memory to be utilized by other GMIS processes like communicator server. It sends request to data handler to update resource information in database. TIP process flow chart is shown in the following figure 5.4.

![Figure 5.4: Topology Information Process flow chart](image)

5.1.5 Information Provider Service (IPS)

Information Provider Service (IPS) is responsible for providing resource information. It gets the request from viewer and retrieves resource/job information from the database using data handler. IPS sends retrieved information to the viewer. The following figure 5.5 shows the IPS process flow diagram.
5.1.6 Resource Broker Process (RBP)

Resource Broker Process (RBP) is responsible for resource selection, generation of activation graphs and execution of jobs. The following figure 5.6 shows the RBP process flow chart.

Figure 5.5: Information Provider Service Process flow chart

Figure 5.6: Resource Broker Process flow chart
5.1.7 Communicator Server Process (CSP)

Communicator Server Process (CSP) establishes communication between the GMIS components and resources. The following figure 5.7 shows the Communicator Server Process flow chart.

![Diagram of Communicator Server Process flow chart]

5.1.8 Communicator Agent (CA)

Communicator Agent (CA) on behalf of resource sends updates to GMIS and responds to GMIS Communicator Server requests. The following figure 5.8 shows the communicator agent process flow chart.

![Diagram of Communicator Agent Process flow chart]
5.1.9 Resource Discovery Agent (RDA)

Resource Discovery Agent process responds to the requests generated by the communicator agent. This process collects resource information and sends to the communicator agent. Resource discovery agent process flow chart is shown in the following figure 5.9.

![Figure 5.9: Resource Discovery Agent Process flow chart](image)

5.1.10 Critical Information Agent (CIA)

Critical Information Agent (CIA) process periodically checks for the resource information. Whenever there is a critical change in the resource, then an event is generated and communicated to Communicator agent. Communicator agent sends this event to GMIS communicator server. The process flow chart of CIA process is shown in the following figure 5.10.

![Figure 5.10: Critical Information Agent Process flow chart](image)
5.1.11 Web Client

GMIS viewer is a web application which is implemented using Flex and Servlets. In web application, browser acts as a web client which displays the user interface and the core process logic is executed on the web server. The communication between the client and the server is over Hyper Text Transfer Protocol (HTTP).

To enable the invocation and functioning of the GMIS viewer application, the GMIS application directory on the web server will contain all the Servlets. Flex SWF and html files to support this integrated application. Servlets are java programs which run on the Tomcat web server receives an input request object and responds with a response object. Servlets are responsible for running the logic on the server. SWF and HTML files are required to support GMIS viewer functionality. This SWF file can be played by the Adobe Flash Player. When a grid user invokes GMIS viewer application, main GMIS SWF file is invoked. On the target GMIS which hosts the Tomcat Apache web server, there is an application directory for each web application. The directory structure of Tomcat web server is as follows:

1. /usr/apache-tomcat-5.5.20/webapps/ - This is a parent directory of Apache web server which has subdirectories for each web application.
2. /usr/apache-tomcat-5.5.20/webapps/gmis – This is the application directory specific to the GMIS application. This contains all the executables, configuration files and directory structure to support the GMIS application.
3. /usr/apache-tomcat-5.5.20/webapps/WEB-INF – This directory is a repository of servlets files like SWF files and the library files.
4. /usr/apache-tomcat-5.5.20/webapps/WEB-INF/classes – This directory contains the list of class files which are known as servlets. All GMIS servlets class files are in this directory.
5. /usr/apache-tomcat-5.5.20/webapps/WEB-INF/lib – This lib directory contains the jar files which are referred by the servlets classes.
6. /usr/apache-tomcat-5.5.20/webapps/gmis/css – This directory contains file gmis_style.css which contains the cascaded style sheet configuration settings.
5.1.12 Peer 2 Peer Communicator Service (P2PCS)

Peer 2 Peer Communicator Service (P2PCS) process maintains the connection and communication among multiple GMISs. It requests other GMIS for a service and processes other GMIS requests. The process flow chart of P2PCS is shown in the following figure 5.11.

![Figure 5.11: Peer 2 Peer Communicator Service Process flow chart](image)

5.2 Communication among GMIS Components

To support communication among GMIS components a set of common Inter Process Communication library functions are developed. The same library is used for communication among the resource agents i.e., communicator agent, resource discovery agent and critical information agent.

The following figure 5.12 shows the typical order in which the IPC services are called. The time at which the un-connected functions are called is irrelevant, as longs as they are called in the right order.
5.3 Viewer

Viewer provides a single system image view for all GMIS users. It is a web based application and is mainly responsible for capturing the inputs from the users to discover resources, add resources and to execute jobs. GMIS users can view the status of nodes.
or status of jobs submitted and current jobs being executed in the entire GMIS managed nodes. The important user interfaces are discussed in this section.

The login or home page of the GMIS viewer is shown in the following figure 5.13. This home page is accessed through URL as http://<GMIS_IP_ADDRESS>:8080/gmis. For example, http://192.168.100.15:8080/gmis, in this example on 192.168.100.15 machine GMIS processes and Tomcat Server are running. In this page user requires to enter username and password to access GMIS services.

![Viewer Home page of GMIS Services](image)

**Figure 5.13: Viewer Home page of GMIS Services**

If users don’t have user credentials then users can register for the services, by clicking “New User” option on this home page as shown in the following figure 5.14. After providing user information, user verification will be done by the GMIS administrator. A pop up message shows a message saying that “Authentication will be done by GMIS administrator and confirmation email will be sent to your e-mail address”. GMIS administrator gets a notification about the user registration. GMIS administrator validates the user information. If the registration information is valid information then this user details will be updated into the GMIS database and a confirmation email will sent to the user. After receiving confirmation the user can access GMIS services.
After successful login, the GMIS services page will be as shown in the following figure 5.15.
This page layout designed into two frames. The first frame contains the resources tree hierarchy of resources and jobs. Upon selection it displays the status of resources and jobs.

Second frame contains menu options like File, User Management, Resources, Jobs and Help. File menu will have two options, one is logout and second one is exit. Upon selection of logout, the current home page returns to login page of GMIS. Upon selection of exit the viewer will be closed. User Management menu contains options for changing user password and viewer settings. Resources menu contains various options like Add Node, Add Cluster, Add Grid, Add Service, Discover Node, Discovery Cluster, Discovery Grid, Discovery Service and Resource Availability. These options provided to achieve the user based resource discovery and dynamic grid network discovery services. These options are shown in the figure 5.16.

![Figure 5.16: Resources Discovery Services View](image)

Jobs file menu contains the options of activation graph creation, execution, job submission and Auto Job execution options as shown in the figure 5.17. The last menu
option Help contains the help contents of GMIS services and how to use the GMIS viewer.

![Image of GMIS Monitoring View](image)

**Figure 5.17: Job Execution Services View**

### 5.4 Testing Environment

A simulator called GMIS-SIM has developed to test the GMIS framework. It functions as a set of resources that are configured to interface to the GMIS. The GMIS-SIM is developed for the testing of GMIS services for a variety of reasons. Real applications may run for very long time and it would not be feasible to perform statistically significant number of experiments. Using real resources make it difficult to explore wide variety of resource configurations. The dynamic nature of resources makes it difficult to repeat the experiments. Other reasons include difficulty to create all situations, for example, to create or test on a large cluster or grid, say 100 nodes with heterogeneous computing resources, or test multiple jobs arrived at a single instance is very difficult and some times not feasible. A common practice followed by the researchers is to create a simulator, and simulator is used to create any number of nodes/clusters/grids artificially.
GMIS-SIM simulator is a simple application, which creates a homogeneous and heterogeneous nodes, clusters and grids using response configuration files which consist of sample resource information. There are four types of response configuration files.

1. **Resource Response Configuration file:** This contains basic resource information in the following format.

   Resource_type|FQDN|Host_name|IPaddress|Status|Parent_host_name

   Where,
   - **Resource_type** indicates the type of resource. “1” indicates grid, “2” indicates cluster and “3” indicates node.
   - **FQDN** is Fully Qualified Domain Name.
   - **Host_name** is host name of the resource.
   - **IP address** is physical IP address of the resource. For nodes IP address is mandatory. If any node is acting as a head node for a cluster or grid then that IP address is used for cluster or grid. Other wise it will be marked as “0.0.0.0”.
   - **Status** indicates the status of resource. It indicates three values. “1” indicates the resource is available for GMIS users. “2” indicates the resource is busy in performing some job executions. “3” indicates not available to the GMIS users.
   - **Parent_host_name** indicates the parent host of the resource. In case of node, parent host name should be the cluster host name. For cluster parent host name should be a grid host name. For grid it will be the GMIS server host name, which will be mentioned as “0”,

   e.g., for a Grid the entry is as

   1|GRID_1@uohyd.gridproj1.ernet.in|GRID_1|0.0.0.0|1|0

   Cluster entry is as

   2|CLUSTER_1_GRID_1@uohyd.gridproj1.ernet.in|CLUSTER_1_GRID_1|0.0.0.0|1|GRID_1
Node entry is as

3|NODE_1_CLUSTER_1_GRID_1@uohyd.gridproj1.ernet.in|NODE_1_CLUSTER_1_GRID_1|192.168.100.16|1|CLUSTER_1_GRID_1

2. Node Response Configuration File: This file contains detailed node information in the following format.

   FQDN|IP_Address|Operating_System|Processor_family|Number_of_Processors|MEMORY|SWAP|CPU_usage|Disk_space|Dedicated_flag

   e.g., for a node an entry is as

   NODE_1_CLUSTER_1_GRID_1@uohyd.gridproj1.ernet.in | 192.168.100.16 | Linux8.3|XEON|1|512|1024|28|12000|0

3. Service Response Configuration File: This file contains services available in a node in the following format.

   FQDN|IP_Address|Service_name|Service_location|Input_file_location|Input_file_IP_Address|Output_file_location|Description

   e.g., for a service an entry is as

   NODE_1_CLUSTER_1_GRID_1@uohyd.gridproj1.ernet.in|192.168.100.16|Service_1|/home/applications|/usr/applications/input|192.168.100.16|/usr/applications/output|this is an example service

4. Job Response Configuration File: This file contains a set of jobs in the following format. Status attribute takes various values. “1” indicates new job submitted. “2” indicates job execution started. “3” indicates job execution completed. “4” indicates job execution failed. Priority of Job takes integer values range from 1 to 9. “1” stands for highest priority where as “9” stands for least priority.
<table>
<thead>
<tr>
<th>FQDN</th>
<th>Service_name</th>
<th>Status</th>
<th>Input_FQDN</th>
<th>Input_file_location</th>
<th>Input_file_name</th>
<th>Priority_of_job</th>
<th>Output_FQDN</th>
<th>Output_file_location</th>
<th>Output_file_name</th>
</tr>
</thead>
</table>

e.g., whenever a job is submitted, then this job file gets updated. An entry will look like,

```
NODE_1_CLUSTER_1_GRID_1@uohyd.gridproj1.ernet.in|Service_1|1|NODE_1_CLUSTER_1_GRID_1@uohyd.gridproj1.ernet.in|/home/applications|input|1|
```

These configuration files are manually editable. Whenever a status change is required then the configuration files need to be edited manually. For example, after submitting a job, the status will be “job submitted”, this status can be changed to “job execution started”, “job execution completed” or “job execution failed” manually.

For each GMIS there will be one GMIS-SIM process that runs to simulate a grid environment. An environment variable called “GMIS_SIM_ENABLE” is used to differentiate communication between simulator usage and normal network IP addresses usage. If this environment value is set to “1”, then all requests from GMIS communicator server will reach to the GMIS-SIM process. GMIS-SIM process responds to all the requests that come from the GMIS communicator server. GMIS-SIM reads resource information from the configuration files and sends the resource information to the communicator server. For instance if a request comes from a GMIS communicator server to discover a cluster resources then GMIS-SIM looks into the resource configuration files and if there are any resources available with those IP addresses then those resource information will be sent to the GMIS communicator server. In case of job execution job information will get updated in the job configuration file and by default the status is set to “Execution is in progress”. To change the status of a job or resource, the configuration files need to be edited manually, and an event message need to be sent to GMIS communicator server. Other event messages like addition, modification and deletion of resources also can be sent through the GMIS-SIM. For sending messages, GMIS-SIM sends messages to the GMIS communicator server message queue.
5.5 Some Experiments and Results

5.5.1 Cluster Level Dynamic Network Discovery

To discover resources at cluster level a range of IP addresses need to be given on the viewer. For this select “Resources -> Discovery Cluster” option on menu bar. A viewer appears to enter “Start IP address” and “End IP address”. After entering IP addresses click on “Discover” button. A pop up message comes to the user saying that “Select the resource view to see the discovered resource information”. If no resource information is available with the given IP addresses range, an alert message will display on viewer as shown in the following figure 5.18.

![Figure 5.18: Cluster Discovery View](image)

5.5.2 Monitoring of Resources

To check the status of resources, users have to select the resource view option on left side hierarchical tree. Selection of a cluster shows the nodes and their status on right side frame. Nodes are represented as colored icons. A green icon represents that node is available for providing service. A red icons represents that resource node is busy in performing some job execution. Resource Monitor service decides which node is
available and which is not, based on the node’s CPU and Memory usage. Any of these two values exceeds to a certain threshold values then icon will be represented as a red icon. Otherwise it will be shown as Green color. These threshold values are configurable. Currently, two resource attributes are considered. This can be extended to other parameters like how many jobs it is executing, how much swap is available or instead of values, usage percentages. A blue icon represents that node information is not available due to network problem or some other problem. 48 node cluster status information on viewer is shown in the following figure 5.19.

![Node Status View](image)

Figure 5.19: Node Status View

By clicking on any node, complete node configuration information window will be displayed as shown in the following figure 5.20. This information includes node host name, parent cluster name, parent grid name, FQDN, IP address, operating system version, processor family, number of processors, RAM, SWAP, CPU Idle, Disk space available and whether it is a dedicated node or not.
5.5.3 Job submission

GMIS users can submit the jobs using interface provided on viewer. Select “Jobs-> Job Submission” option to submit the jobs. Enter the Job submission information like FQDN where job need to be executed, Input file location, output file location, priority of job. Upon filling all the information related to job submission, click submit button on the viewer. The information entered by the GMIS user is validated and if the information is valid information, a pop up message shows up saying that “Go to the Jobs view to see the job status” as shown in the following figure 5.21.

There is another option called “Auto Job Execution” provided on the viewer. Here user mentions node requirements, service location, service input file location and output file location desired. GMIS service resource selector selects a best suitable node for the user requirements and submits the job execution on that node using GMIS communicator server. Resource communicator agent gets the service and input files from the user mentioned locations and start execution of the job. After completion of job execution output file sends to the user mentioned location. Other job submission option “Activation Graph” creation and execution is discussed in the next chapter.
Figure 5.21: Job Submission on GMIS Viewer

Select the “Jobs view” and click on the job icon to see the status. The status of job execution will be in the popup message as shown in the following figure 5.22.

Figure 5.22: Job Execution Status
5.5.4 Multiple GMISs' view

Multiple GMISs can be connected for achieving GMIS-GMIS communication. Multiple GMISs resource information will be shown on the left side frame with the under different GMIS names. For instance, if “CS GMIS” is another GMIS which maintains different set of resources, connected to the UH GMIS, then the left side resource monitoring view of UH GMIS shows the CS GMIS information also as shown in the following figure 5.23. At the same time “CS GMIS” monitoring view contains only “CS GMIS” managed resources. User of “UH GMIS” can view the resource status of “CS GMIS” resources and can submit the jobs on “CS GMIS” managed resources.

![Figure 5.23: Multiple GMISs Connected View](image)

5.6 Summary

This chapter discussed about the GMIS processes development and interface modeling using process flow charts. It also discussed about the IPC library services typical usage. It shows the sample screen shots of GMIS viewer like login screen, resources discovery services option, monitoring resources and job submission options. It
discussed about the GMIS-SIM which is a simulator to test GMIS services. Finally it has shown the multiple GMISs view on a single GMIS viewer.

Next chapter will discuss about the Multidisciplinary Design Optimization (MDO) and UH MDO framework which is chosen as a case study for the GMIS. Virtual cluster formation is another application of GMIS is also discussed in the next chapter.