Conclusions and Future Enhancements

7.1 Summary

In Grid environment, the resources are heterogeneous and geographically distributed with varying capacities, usage and constraints. The management of resources and job scheduling in such large and distributed environment is a complex task. We have developed an architectural resource management framework called Grid Management Information Server (GMIS) to address this complex task. To support the thesis, that an efficient grid resource management system can deliver a significant value to users as compared to traditional approaches, we have:

- Identified key requirements that an efficient resource management system that needs to support.
- Identified important design considerations for efficient resource management system.
- Compared with the traditional resource management frameworks.
- Identified and included several features not available in the other popular resource management frameworks.
- Developed a distributed resource management framework called Grid Management Information Server (GMIS).
- Designed resource discovery algorithms with three different strategies i.e., dynamic, event based and user based discovery.
- Developed monitoring and data collection manager services to provide users with wide range of options to select resources based on their capabilities and historical performance.
- Developed push and pull based algorithms for gathering resource information, to ensure that the grid users will have latest status about the resources as well as submitted jobs.
- GMIS provides support for activation graph creation based job submission and execution i.e., submitted jobs can be divided into multiple tasks and results can be analyzed at each task level.
• Shown how GMIS can be used as distributed computing environment over a WAN for UH Multidisciplinary Design Optimization framework.
• Demonstrated how a virtual cluster can be formed using GMIS.

7.2 Conclusions

Grids have evolved rapidly and are becoming a good alternative for the otherwise costly supercomputers. The tools and number of applications being built around these technologies are also increasing day by day. The resource management in grid environment is a complex task, because resources are distributed over wide area network and dynamic nature of resource availability.

The resource management systems used in grid environments need to be adaptive so that they can handle dynamic changes in availability of resources and user requirements. At the same time, they need to provide scalable, controllable, measurable, and easily enforceable policies for management of the resources. To address these requirements, a resource management framework called Grid Management Information Server (GMIS) has been developed. The GMIS provides dynamic discovery of resources, monitoring of resources and brokerage service. Since the GMIS framework is a component based architecture implementation can be easily done for different middleware technologies.

The information services used in grid environments need to be highly flexible in nature to handle the dynamic behavior of the grid network resources. Additionally, they need to be scalable and easily implement able in grid based environments. The GMIS framework addresses these requirements. Several features and design considerations in GMIS framework are not available in the other popular frameworks.

The GMIS framework describes the resource management for grid environment that supports resource discovery, monitoring and brokerage with integrated topology information. This frame work is an open and extensible resource monitoring system, based on the Client - Server Architecture. The GMIS prototype is developed and efforts are currently underway to develop complete services.
7.3 Future Enhancements

This thesis formulated a comprehensive distributed resource management architectural framework. Resource management algorithms have been developed to meet grid users’ requirements. This work is laid as a foundation for the efficient resource management and it opens up several avenues for future work in efficient resource management in grid environment.

As the complete implementation requires more effort, only a prototype has been developed. Complete implementation will be done in future as part of UH Grid development projects. As part of this, there is a need to focus on synchronization issues, since this framework depends on message communications among the GMIS components and as well as with resources agents communications. The scope of future enhancements includes, a feature displaying the current status of job and its details, as currently it shows only a set of states like job execution started, completed or failed. Some of the important enhancements are listed below:

- **Authentication and Authorization:** Currently the authentication and authorization of GMIS users is manual process. User requests for GMIS services access permissions and GMIS administrator manually verifies the user request. This process can be automated similar to generation of digital certificates of Globus.

- **Middleware support:** GMIS framework needs to be integrated with other frameworks like Globus, GridBus for which a set of APIs development is required. Current developed APIs is useful for scheduler level frameworks.

- **Enhance GMIS-SIM:** Current GMIS-SIM is a simple process which simulates resources from response configuration files. This needs to be enhanced to simulate the network model which supports various types of networks with different static and dynamic configurations. A lot of real time examples should be tested on the framework, so that the shortcomings of the framework could be seen. For testing real time examples the GMIS-SIM need to be enhanced.
• **Enhancements to Brokerage Service:** Currently GMIS brokerage service supports only dynamic resource allocation. There is no provision of reserving resources in advance. This need to be incorporated in GMIS brokerage service. Economic models also need to be adapted to the brokerage service.

• **Crash Recovery:** Crash recovery is an important area, which needs to be addressed in GMIS framework. If the node executing a task which is responsible for supplying input data to the other remote node crashes, then the job or activation graph execution could fail. Hence a mechanism for crash recovery also needs to be evolved in GMIS framework, so that even if the node fails in supplying input data, then the job execution should restore to a stable state. This can be achieved by having check points in job execution process.

• **Poll Package Optimization:** While packing Object Identifiers (OIDs) for polling a set of managed resource objects, the number of OIDs that need to be polled in a single poll request packet can be optimized based on the OID lengths and message buffer limits. Packing related object OIDs into smaller number of poll packets reduces the traffic between the GMIS communicator server and the resources.