Performance Analysis of Information Services in Computational Grids

Shakti Purohit\textsuperscript{a} and Sanjay Chaudhary\textsuperscript{b}

\textsuperscript{a}Persistent System Pvt. Ltd., Pune - 411016, India, shakti_purohit@persistent.co.in
\textsuperscript{b}Dhirubhai Ambani Institute of Information and Communication Technology Gandhinagar - 382007, Gujarat, India, sanjay_chaudhary@daiict.ac.in

The primary objective of a Grid Information Service is to discover resources out of large number of grid nodes. While deploying a grid, it is a challenging task for the grid administrator to opt for the GIS which discovers resources within acceptable time period. This work aims at analyzing the behavior and performance of two most widely deployed GIS viz. Monitoring and Discovery Services (MDS) and Distributed Hash Tables (DHT). A quantitative comparative analysis of the two schemes has been performed. Experiments have been designed and performed with GridSim \cite{1}, to understand the behavior of the two services in presence of large number of users and resources. Four experiments are conducted, scalability of the two schemes has been studied, and the parameters that form bottlenecks in the behavior of these two schemes, have been identified. Limiting bounds have been established over various parameters identified. These parameters are number of users an index server can support, number of information providers, response time and throughput of the index server. Finally the results obtained establish that DHT index server scales better and provide higher throughput as compared to MDS index server. The results provide an aid in deploying information services in a Grid and help in future development work.

1. INTRODUCTION

With the increased popularity of internet, availability of high performance computers, and high speed networks as low cost commodity, it has become possible to use networks of computers as a single unified computing resource. This seamless integration of geo-graphically distributed resources such as CPU cycles, storage systems, data sources, and devices forms computational grid. This integration of resources helps solving problems, which cannot be solved on a single machine. The wide range of resources available on a grid needs to be discovered in a dynamic grid environment, where attributes of resources change frequently. In such a scenario the scheduler has to map the discovered resources against user requests within defined time constraints. Resource discovery is a service which locates resources as requested by users and grid applications in a dynamic grid environment where the size is expected to scale to millions of resources provided by large number of resource providers.

In this paper we discuss issues of performance and scalability of two information services: Monitoring and Discovery Services (MDS) deployed in the Globus Toolkit \cite{2} and Distributed Hash Table (DHT) \cite{3} deployed in various P2P systems viz. \cite{4}, \cite{5}. We analyzed the two architectures and identified a set of parameters which affect behavior of the two information services. Based on these parameters, we designed four experiments and performed simulations for the two under similar environment. Design of experiments, setup and simulation results are explained in Section 4. Recommendations for a grid administrator based on the results obtained and constructive suggestions for performance improvements are discussed in Section 5. Conclusions and future follows in Section 6.
in case of non cached data increases linearly and stabilizes at 2.7 seconds which is way higher than acceptable limits. The response time in case of cached data is as low as 0.056 seconds. Since the central GIIS has all the local information and real grid request patterns show same requests repeated large number of times. Caching is an effective technique.

5. RECOMMENDATIONS

We propose following recommendations for grid administrator as guidelines to configure and improve performance of grid information services.

1. The results obtained from experiment 5 strongly suggest the deployment of caching at GIIS. For an MDS system to perform in acceptable response time limits caching must be enabled at GIIS. However, the concern about caching is what type of data to be cached and how long it should be kept in the cache. This issue can be resolved depending on the type of application running on grid as different applications have different requirements of retrieving the data.

2. Experiment 1 suggests an upper bound over the number of information providers the MDS systems can sustain without degradation in performance. As the number of information providers exceeds and performance of GRIS starts degrading, it is recommended to introduce a new GRIS server in the hierarchy at the same level. Here, the number of information providers is kept as 150.

3. Experiment 4 establishes an upper bound over the number of users a GRIS server can handle concurrently. It is recommended to direct concurrent user queries to other GRIS server before a GRIS server reaches saturation, i.e., its response time stabilizes. In case of experiments conducted the response time at which GRIS server stabilizes is 0.65 seconds, which is too high. In such cases, the queries from the GRIS server should be directed to other GRIS server before response time becomes 0.65 seconds.

4. Though experiments have not been performed by varying the hierarchy level in MDS, but from Experiment 2, it can be inferred that as the levels in hierarchical structure of MDS increases, response time exceeds acceptable limits. It is suggested not to increase more levels in hierarchy.

4. Though experiments have not been performed by varying the hierarchy level in MDS, but from Experiment 2, it can be inferred that as the levels in hierarchical structure of MDS increases, response time exceeds acceptable limits. It is suggested not to increase more levels in hierarchy.

<table>
<thead>
<tr>
<th>Resource Discovery</th>
<th>MDS</th>
<th>DHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of Servers per VO</td>
<td>Centralized</td>
<td>Decentralized</td>
</tr>
<tr>
<td>Indexing Scheme</td>
<td>Directory Based</td>
<td>Hash Based</td>
</tr>
<tr>
<td>Naming Convention</td>
<td>LDAP Convention</td>
<td>Key Based</td>
</tr>
<tr>
<td>Architecture</td>
<td>Hierarchical</td>
<td>Flat</td>
</tr>
<tr>
<td>Range Queries</td>
<td>Supported</td>
<td>No Support</td>
</tr>
<tr>
<td>Lookup</td>
<td>Acceptable</td>
<td>Efficient</td>
</tr>
<tr>
<td>Resource Updation</td>
<td>Complex</td>
<td>Efficient</td>
</tr>
</tbody>
</table>

6. CONCLUSION

This work provides an understanding of the working and behavior of MDS and DHT. Tables in Figure shows a quantitative comparison between the two schemes based on experiments and study of the architectures of these two schemes. On the basis of results obtained, it can be concluded that DHT scales better, shows acceptable response time, and higher throughput as compared to MDS grid information services. Based on the results and conclusions discussed, a set of recommendations for a grid administrator have been suggested.

7. FUTURE WORK

The future work includes; (i) The experiments have been conducted assuming static nature of
Figure 4. Registration Time Comparison between MDS and DHT

Figure 5. Effect of IP on Response Time of MDS and DHT

Figure 6. Effect of Concurrent Users on Response Time of MDS and DHT
attributes. In real grid environment state and attributes of resources change frequently. Future experiments should be conducted, which involves dynamic updation in the state of resources. (ii) The experiments have not considered range queries. Range queries are the resource requests to discover resources whose attributes lie in a range, e.g., "Computers with CPU speed between 100 MHz and 1000MHz". There must be a provision of range queries in future work to be undertaken. (iii) Future work could extend to implementing and replacing current index services in Globus Toolkit by DHT based index services. Globus toolkit mainly consists of two components: grid Service and Index Service. The index service receives resource request, searches for it and returns back the resource location. As suggested in [12] globus architecture can be modified as shown in Figure 9. A new component called peer service, which processes resource request to forward it to index service. A peer service is also used for establishing peer connections.

REFERENCES

2. Web site: http://www.globus.org/
Figure 8. Proposed Implementation of DHT in Globus Toolkit


**Shakti Purohit** is working as a software engineer with Persistent Systems Pvt. Ltd., India. He has completed M.Tech. (ICT) in the year 2006. His areas of interests are Grid Computing and Computer Networks.

**Dr. Sanjay Chaudhary** is an Associate Professor at Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT), Gandhinagar, India. His research areas are Distributed Computing, Grid Computing, Semantic Web, and ICT Applications in Agriculture. He has authored a number of research papers in international conferences and workshops. He has served on many program committees of International conferences and workshops. He has authored three books and a number of book chapters. He holds a doctorate degree in computer science from Gujarat Vidyapeeth. Prior to joining DA-IICT, he worked with Gujarat University. He has worked on various large-scale software development projects for corporate sector, cooperative sector, and government organizations. He is actively involved in various consultancy and enterprise application development projects.