A HYBRID RESOURCE DISCOVERY MODEL FOR GRID COMPUTING

Mohammed Bakri Bashir, Muhammad Shafie Bin Abd Latiff, Aboamama Atahar Ahmed, Yahaya Coulibaly , Abdul Hanan Abdullah and Adil Yousif

Faculty of Computer Science and Information System
Universiti Teknologi Malaysia
Skudai, Johor, 81300, Malaysia
`mhmdbakri@gmail.com, shafie@utm.my , aboamama@utm.my ,`
`yaya@agetic.gov.ml,hanan@utm.my, and adiluofk@gmail.com`

ABSTRACT

Grid computing system is a generalized spread network-computing infrastructure with machines distributed across multiple organizations and administrative domains. The aim of a grid is to provide uniform access to heterogeneous resources and services such as computing, network and storage. Among different functions of a grid system, such as resource selection, resource discovery (RD), resource monitoring, and others, RD remains the most important issue in grid computing environment. The success of other functions depends on RD, especially in large grid systems. RD involves finding suitable resources that match user requirements. In this paper, the results of a review of current RD schemes are presented. Additionally, a novel RD technique which is under evaluation is proposed. Thus, the exact contribution of this paper is a review of the state-of-art of resource discovery techniques in grid computing.

KEYWORDS

Grid Computing; Distributed Computing; Resource Discovery; P2P Grid; Hierarchical Grid; Large-scale Grid Systems, Reliability, Dynamicity;

1. INTRODUCTION

The rapid growth of scientific applications has led to the development of new generation of distributed systems such as Grid Computing [1]. Grid systems are distributed over networks whose resources are managed, used, owned by several organizations, and are dynamic in the sense that participants can join or leave the system at any time[2]. These systems facilitate uniform access to a large number of services and heterogeneous resources such as workstations, networks, storages, and computing power that belong to several organizations and administrative domains [3]. Moreover, during the last decade, a new generation of grid systems emerged. This new generation of grid is known as World Wide Grid (WWG) [4]. Similar to the well-known www, wwg aims at establishing a scientific and a computational worldwide grid that anybody, around the globe, can access and use its services depending on his requirements.

The dynamicity, the heterogeneity, and the distribution attributes of grid resources which is considered as advantages of a grid system, has become a challenge for extending grid service to large scale systems due to the complexity of resource discovery[5] on which the grid system relies to find appropriate resources for a given job [6]. Thus, resource discovery plays an

important role in the success of any grid system, as the system needs this process to seek and to locate suitable resources for the execution of a given task in a reasonable time in spite of the heterogeneity and the dynamicity of the grid resources. Resource discovery techniques are classified into the following four main types: centralized techniques, P2P methods, agent-based schemes and hierarchical techniques.

When designing a resource discovery technique, one must consider scalability, reliability and dynamicity. These are essential criteria for designing competitive resource discovery schemes [7].

Scalability: This is related to computational performance of the grid resource discovery technique, because the grid system is a large-scale environment in their nature. The resource discovery system should be able to scale with a growing number of users, events, and resources. The performance of static resource discovery technique will decrease very fast when the size of the grid environment is grown. This problem causes the RD technique to work poorly in this environment.

Reliability: This is a very important criterion, especially in the large-scale grid where failure rate is high. Failure can be due to false-positive errors caused by the usage of Time-To-Live limitation, server (head node, manager node) failure as a result of a high amount of queries.

The third and final requirement for a good resource discovery mechanism is dynamicity. This condition affects the performance of grid system by its influence on the reliability of the system. The nodes in the grid environment might be very dynamic in terms of leaving and joining the grid system. The dynamicity of some nodes like central server affects the domain of the queries, and affects the reliability of the system by turning the central server into a single point of failures. For more details on resource discovery, the reader is advised to refer to[7].

In this paper, the findings of a critical review of current resource discovery techniques are presented, which are believed to be of help for researchers in grid computing, particularly the new comers in the field. Based on these findings, a novel Resource Discovery Technique (RDT) is proposed. The new model is called HPPGrid and it is based on P2P and hierarchical techniques to overcome the issues of reliability and scalability in both P2P and Hierarchical RDT. In this model, we combine the positive points of the individual systems and avoid their shortcomings; by doing so, the model is expected to overcome the drawbacks of the previous works. However, the evaluation of this model is out of the scope of this paper.

The rest of this paper is organized as follows: Section 2 discusses and analyzes the state-of-art of resource discovery techniques; Detailed descriptions of the proposed resource discovery model are found in Section 3; we conclude this paper and highlight future works in Section 4.

2. RELATED WORK

For its importance in grid systems, a number of techniques have been proposed in the literature to solve the problems related to resource discovery in grid computing. These techniques can be classified into different classes. For instance, on one hand, the work done by Lei and Jiuyang in [8] categorises resource discovery schemes into two types: Centralized and Distributed schemes. On the other hand, the researchers in [5], have classified resource discovery techniques into 3 classes: 1. Web services; 2. Agent-based methods and 3. Peer-to-Peer based schemes. However, resource discovery techniques can be further classified to take into account some other features of a grid system, such as network structure, resource types and others. Therefore, in this paper, RDTs are classified into the following four categories and reviewed based on those classifications. Figure 1 is a graphical representation of the four classes.

I.   Centralized Schemes.

II. Hierarchical Methods;

III. Peer-To-Peer Approaches;
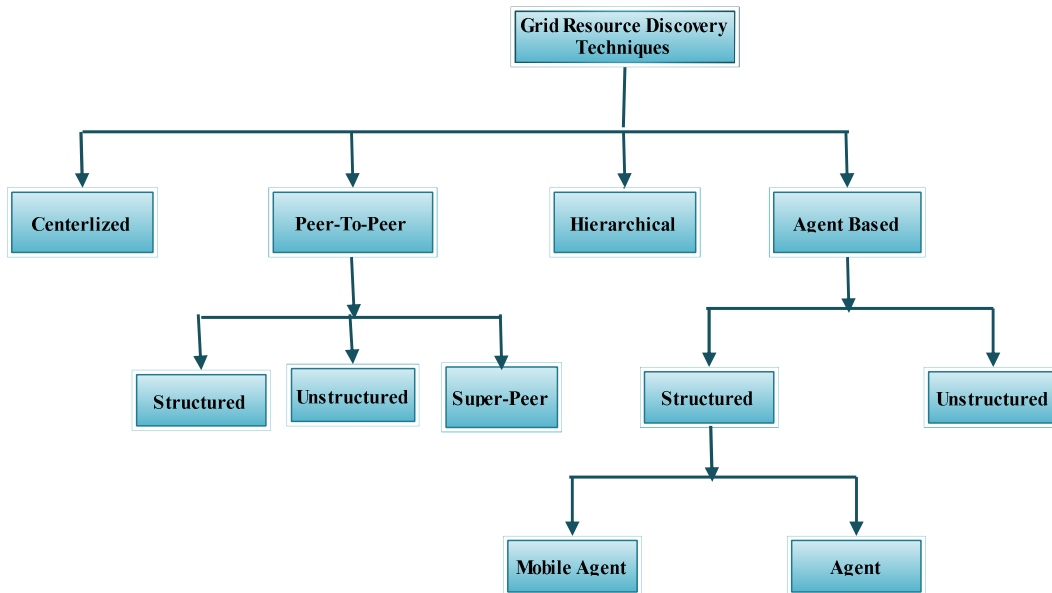
IV. Agent-based Methods.

Figure 1. A general RDT classification

2.1. Centralized Techniques

In centralized resource schemes, a centralized database is used to save the information of resources. Grid Systems such as UNICORE[9], and MDS2[10-12], and [13] are examples of centralized RDT. In these systems, the information of grid resources is stored in a central database. Although this is simple to implement, apparently, and cost effective, such systems are susceptible to single point failure, unless back-up servers are deployed which is cost constrained. Moreover, centralized systems create bottleneck issue in case of a large number of user queries to discover resources and frequent updates of the resource status. These issues have discouraged researchers to adopt centralized resource mechanisms as they remarkably decrease the performance of the grid systems, which are naturally large by definition.

2.2. Hierarchical Techniques

The second category of resource discovery techniques is based on hierarchical structure. In these RDTs, instead of using a central database, Ganglia and MDS-3 proposed in [14] and [11], [15] and similar works in [16] and [17] organize the information services in a hierarchal structure. Although this structure is more scalable than centralized systems, it still suffers from single point failure mentioned earlier. This is because at each level there is a central data base server responsible for resource update requests. However, the bottleneck problem has been reduced.
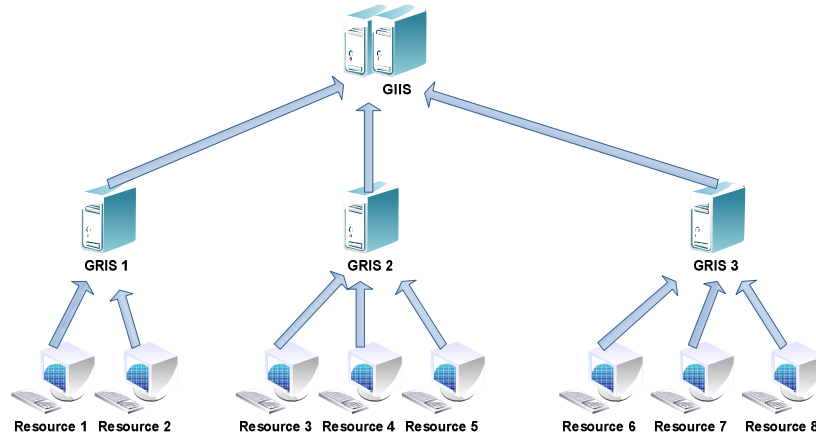
Figure 2 . The architecture of Globus MDS-2 [14].

2.3. P2P Techniques

Decentralization of Grid computing systems based on P2P network approach may overcome the existing limitations of hierarchical and centralized approaches. However, complete decentralized nature of the system raises other serious issues in several domains of grid systems such as resource allocation, resource discovery, and security. P2P initiates a model of self-organization and decentralization of highly independent peers. This new model allows the system to size to a very large number of participating nodes.

The P2P model is different from the traditional client-server model, because every component in P2P systems operates as a server and as a client at the same time. P2P systems can be categorised, depending on the organisation of the peers and on the connection protocol, into three types: (i) Unstructured P2P systems; (ii) Super-peer systems and (iii) Structured P2P systems.

Unstructured P2P resource discovery approaches , such as the works in [18] and [19] handle the dynamicity of resources . The common routing mechanisms used in these approaches prevent the grid to scale; nevertheless, the usage of TTL causes false-positive errors in most of unstructured systems even if the searched resources exist and are accessible on the grid. On other hand, increasing TTL will increase network traffic and negatively will affect the runtime of the algorithms and inefficient use of the bandwidth.

On the other hand, super-peer based methods such as [20],[21], and[22],proposed a RD mechanism in which some selected super-peers nodes operate as directory services. The flooding mechanism for communication used by most super-peers enhances the scalability of the grid systems. Therefore, the reliability of super-peer is affected by (i) the bottlenecks problem when the number of request for the super-peer is very large,(ii) and the losing the resources under control by a super-peer as a result of super-peer failure (single point of failures).

The most structured P2P methods such as in [23] ,[24] ,[25] ,and[26] , reduce the bottleneck problem and ensure the scalability of the system by involved all the resource nodes in the query processing, which ensure that all nodes in the grid will have the equal load. However, most methods distribute the queries in the network by following a defined path. The failure of the node that forward the queries, bring the single point of failure (reliability) problem to the grid system.

2.4. Agent Based Techniques

According to Foster and *et al* [27], an agent is defined as an embedded computer system that is situated in some environment, capable of flexible, autonomous action in that

environment in order to meet its design objectives. Among the common features of an agents are: (1) Specific problem solving entities with clear frontiers; (2) Embedded in a particular environment and act on that environment to produce some desired results; (3) Autonomous in the sense that they have control on both internal state and their own behaviour.

Agent-based techniques are being broadly proposed as a method of solving resource discovery problem in grid computing, mainly because of their autonomy property where agents use their migration policies to determine a new migration sites.

The agent-based grid RD approaches is classified depending on the network topology to structured and unstructured approach.

The unstructured approaches such as [28],[29],[30],and [31] do not suffer from the single point of failures problem and bottleneck problem . The flooding approach used by the agents for requesting a resource affects the scalability of the Grid. When we use the mobile agent more smart routing methods used to increase the scalability of the system. The queries using routing methods follow a single path, but the queries may be lost in the network in which mobile agents are used due to the failure on any node on the path of query routing. While the mechanism does not use central mangers, the mechanism will be free from single point of failure problems. Furthermore, the false positive error eliminated by remove the TTL value from the queries.

Structured networks [32],[33],[34],and[35] ,eliminate the problem of bottleneck while the nodes have the same load. Moreover, the structured nature of the grid causes the system to use efficient routings algorithms for queries, to enhance the scalability of the grid system. However, the crash of a node in grid will cause loss of queries in the network. Moreover, the crash of the central node that manages set of nodes will prevent the queries from reaching these nodes even if they exist.

2.5. Comparison of Different Techniques

The previous sections discussed some current studies related to diverse methodologies used in resource discovery. Those studies are classified into four types. For each type of approach, the main approach described and discussed synthetically with detailed analysis. With respect introduced criteria (scalability, reliability, and dynamicity). These provide an evaluation of methods behaviour in high dynamic large-scale environment. A comparative study done between all resource discovery approaches (Peer-to-Peer techniques, hierarchical, centralized, and agent systems) ,which permitted to mention the advantages and drawback of different approaches.

Firstly, the methods that depend on centralized/ hierarchical topology have some advantages of being Open Grid Service Architecture OGSA compliant. Nevertheless, the grid systems characteristics, large scale and dynamicity of nodes, prevent the proposed methods to work in such environments. The resource discovery methods must be scalable and reliable. The convergence between P2P, agent , and grid systems have encouraged several researchers to develop diverse types of resource discovery approaches, which can improve the reliability and scaling of grid systems.

The agent-based techniques being broadly proposed in the resource discovery systems, because of their autonomy property. The agents use their migration policies to determine new migration sites. The grid systems can benefit from the agent property to accomplish efficient rout selection for queries. On the other hand, many proposed methods suffer from false-positive problems brought by non-deterministic nature of approach that uses inefficient flooding techniques. During the queries go across the network the resource discovery methods process the queries inside the resource nodes, which provide up to date resource information.

On the other hand, during the last year many researchers adopted P2P technology in grid systems to enhance the reliability and scalability of the grid systems. Most current P2P approaches

increase queries performance by using structured Distributed Hash Table (DHT) systems[36]. The usage of DHT enables the P2P systems to be scalable and reliable because the resource discovery process involves all nodes in the systems.

Table 1: Results of Different Grid Resource Discovery Techniques

Resource Discovery Schemes	Resource Discovery Criteria				
	Scalability	Reliability		Dynamicity	
		Avoidance of False Positive Errors	Avoidance of Single Point of Failure	Support of Dynamic Resource Node	Support of Dynamic Indexing Node
Centralized	No	Yes	No	Yes	No
Hierarchical	Yes	Yes	Limited	Yes	No
Unstructured P2P	No	No	Yes	Yes	Yes
Structured P2P	Yes	Yes	No	Yes	Yes
Super-Peer	Limited	No	No	Yes	No
Unstructured Agent	No	Yes	Yes	Yes	No
Structured Agent	Yes	No	No	Yes	No

3. THE PROPOSED *HPPGRID* MODEL

3.1. Introduction

From section 2, it is clear that resource discovery remains one of the main issues of grid systems, particularly large-scale ones. Thus, it is essential to develop new schemes that effectively reduce resource discovery time taking into consideration user requirements. Reliability, scalability and dynamicity are decisive criteria for resource discovery schemes. Most of the existing RDTs do not satisfy these three criteria as pointed in Section 2. Therefore, in this paper, an ongoing effort is being made to design and implement a novel RDT that takes into account these design principles. The proposed RDT, HPPGrid, is a hybridization of the Hierarchical and P2P resource discovery schemes.

The motivation for hybridizing hierarchical and P2P is that each of these systems satisfies at least one of the design criteria of RDT. For instance, on one hand, P2P resource discovery schemes are scalable and do not suffer from the Single Point Of Failure (SPOF) problem; on the other hand, hierarchical resource discovery techniques fulfil both dynamicity and reliability design criteria. However, false positive error remains a drawback in hierarchical systems and P2P; dynamicity of the nodes is the major disadvantage of P2P systems. HPPGrid is designed to combine the positive points of hierarchical and P2P and avoid their negative points.

3.2. Architecture Layers of *HPPGrid*

HPPGrid is a decentralized resource discovery model, which is described as *G (C, N, R, J, U).* In this Model,

➢ C represents the total number of clusters in a grid system, G ,and C= *{$c_1, c_2, ..., c_{|C|}$};*

6

➢ N is the total number of nodes in a cluster, c_i, and N= $\{n_1, n_2, \dots, n_{|N|}\}$];

➢   R is used to represent resources available in a grid node N and R= $\{r_1, r_2, \dots, r_{|R|}\}$;

➢ J represents jobs to be processed by the grid and J= $\{j_1, j_2, \dots, j_{|J|}\}$;

➢   *the* number of users requesting service from the grid system is represented by U= $\{u_1, u_2, \dots, u_{|U|}\}$.

The model is divided into three distinct layers: Resources Layer (RL), Cluster Manager Layer (CLM) and Super Manager Layer (SML) as depicted in Figure 2. In this architecture, the functions of CLM and SML are implemented using the P2P communication model to exchange the information between the Cluster Managers (CMs) and also to exchange the information between the Super Managers (SMs) in the same layer. Grid resource information of a node in a given cluster is gathered using Simple Network Management Protocol (SNMP)[37-38]. This function is implemented in the Cluster layer (CL). Thus, the model relies on SNMP, a cross-platform network management protocol, to discover set of resources within a cluster and uses point-to-point connections among different clusters managers. Within each cluster, the model performs, via SNMP, the following tasks:

(i)   Check the status of a grid node by periodically sending heartbeat messages;

(ii)   Collect information about cluster resources.

The overall model architecture layers and their functions are described as follows:

a. Resource Layer:

This is the grid infrastructure layer, and it contains computational resources (CPUs), storage resources (RAM, Hard disks, etc…), services resources (Applications), and network resources (Link capacity and status, etc…) to support the grid systems activities. The resources in this layer are preconfigured in a grid middleware such as *glite, Globus* toolkits. Moreover, SNMP must be configured to discover grid resources in this layer.

b. Cluster Manager Layer:

The nodes in this layer are peer-to-peer nodes, and each node is responsible of the following:

• Discover the resources of all nodes in a cluster by using *SNMP get() message*;
• Categorize and save the discovered information in the directory service of the cluster;
• Exchange the information of this cluster with other peer clusters that belong to the same Super Manger;
• Register the new resource in the cluster directory service;
• Send the updated information of the resources of a the cluster to its Super Manager;
• Send periodically heartbeat to check the live status of the resources.

C. Super Manager Layer (SML):

The nodes in this layer are also peer-to-peer nodes and communicate with each other over a point-to-point connection. The functions of the Super manager are:

• Listen to user requests;
• Analyze and compare the user requirements with the information stored in the Super Manger's Directory Service to determine which clusters have the required resources;

- Send request to clusters to discover the status of their resources;
- Get the updated information about the resources from the clusters;
- Exchange the information with other peer Super Manager;
- Appoint a new Cluster Manager in the event of the failure of the exiting one.

The model in Figure 3 is a hybridized decentralized grid resource discovery that strives to satisfy high performance RDT criteria such as scalability, reliability and dynamicity. The resource discovery method (pseudo code) is described in Figure 4.In the following section, we describe the expected performance of the proposed model.
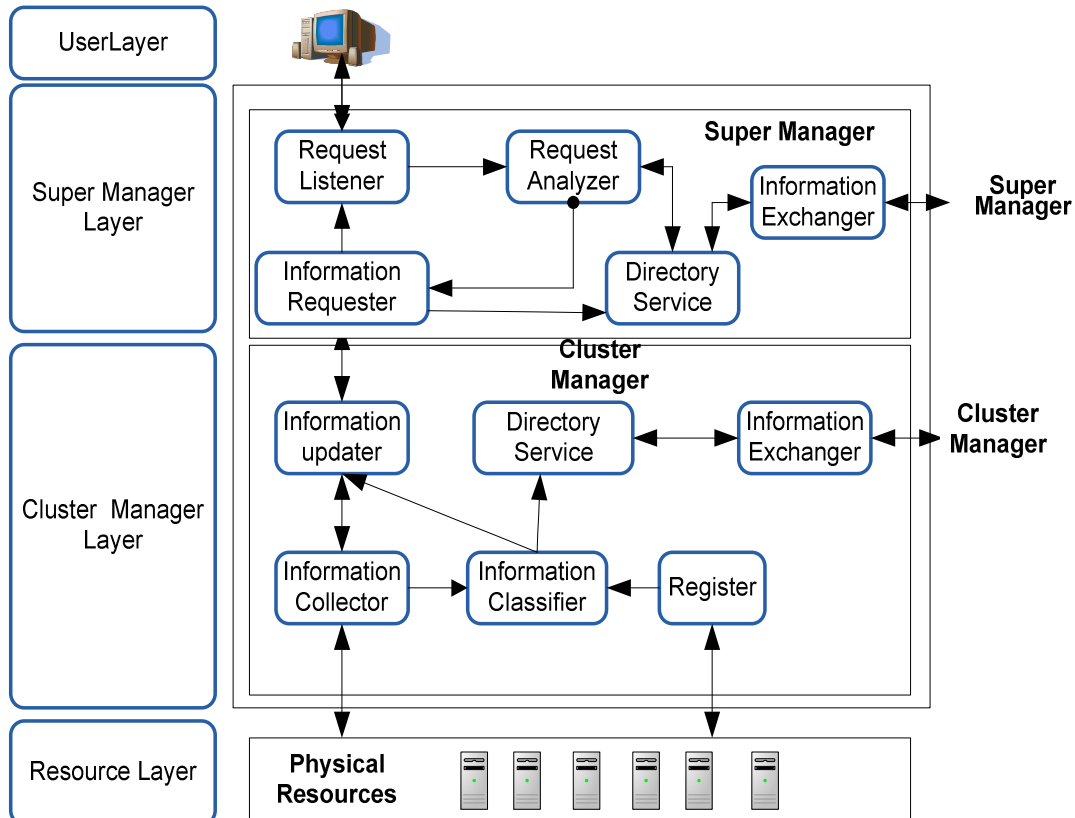

Figure 3. *HPPGrid* Model Architecture

3.3. Features of HPPGrid

The major issues addressed by HPPGrid are scalability, reliability and dynamicity, three main criteria of resource discovery in grid systems. These requirements stem from the distributed and heterogeneity nature of the grid system.

To reduce resource's discovery time and increase their usage efficiently as required by the resource providers, HPPGrid combines Hierarchical and P2P systems. By integrating P2P in the two upper layers of the model, the model aims at achieving scalability and reliability. Platform independence is a unique feature of the model, and it is attributed to the use SNMP to discover grid resources across multiple platforms.

u_i needs to run $j_1 j_2, \dots, j_n$
u_i requests set of resources $r_1, r_2, \dots, r_n$
While (J != 0)
do
SM analyzes the u_i *request*,
Cl-set= is a set of $c_1, c_2, \dots, c_n \in C$ that satisfy u_i *request*
For each $c_i \in Cl\text{-}set$
    SM sends a message to c_i to request $r_1, r_2, \dots, r_n, \in R.$
    CM_i receive *SM* request and broadcast *SNMP get()* message to all $n_1, n_2, \dots, n_n, \in CM_i$
cluster
    Each $n_1, n_2, \dots, n_n, \in CM_i$ *cluster* responses with its resources status.
    CM_i updates resource information in its *directory service*
    CM_i exchanges the information with other *Cluster Mangers* belong to the same *Super*
Manager
End For
SM collects the resource information from all $c_i \in Cl\text{-}set$
SM responses to user with a set of resources that satisfies the user's requirement
SM saves the updated information in its *directory service*
End While

Figure 4. The Pseudo code of HPPGrid Model

SNMP is a network management protocol that supported by most of the current operating systems such as windows, UNIX and its derivatives, and MacOS. Additionally, SNMP is an integral part of most network devices. Thus, using SNMP allows HPPGrid to access and collect resource information from grid nodes regardless of the underlying operating systems, hardware and grid middlewares.

Reliability in grid systems is an important criterion to avoid irrecoverable faults and false-positive errors caused by current resources discovery algorithms. HPPGrid overcomes these limitations and assures reliability in two ways. Firstly, resources are organized in a hierarchical structure and secondly CLM and SM, the two layers responsible for indexing, are designed in P2P architecture. Furthermore, the proposed RDT model eliminates false positive error by removing the Time-To-Live (TTL) from the queries. Moreover, each Cluster Manager saves the resource information of its members, and the information of other peers in the same Super Manager. Therefore, the failure of an active Cluster manager does not affect the model, as in such cases, another Cluster Manager will be chosen to replace the failed server. Using peering in the upper layers eliminates the bottleneck problem of current RDTs.

Dynamism is about tracking the status of the resources continuously. This feature is lacking in most of the current RDT. In HPPGrid, this problem is reduced by integrating SNMP in the model where a heartbeat message is sent to every node to check the existence of the nodes. Moreover, after the requested resources has been discovered, the search for available resources does not stop, rather the discovery process continues as long as the grid system is active. By doing so, resource discovery process is accelerated, and thus the response time is cut short for new requesters or the replacement of existing resources.

4. CONCLUSIONS AND FUTURE WORKS

In this paper, different types of grid resource discovery techniques were discussed and analysed. The advantages and the drawback of the discussed RDTs were highlighted. From these results, it is clear that hierarchical and centralized resource discovery schemes are suitable to be used in small size grids, while P2P schemes are suitable for dynamic and large scale grid systems in which reliability of queries is not important. Based on these findings, and due to the fact resource discovery is an important aspect of a grid computing system, a new RD model, HPPGrid is proposed. The proposed model as described in Section 3 is an effort to design a scheme that takes combines the benefits of two well-known resource discovery schemes: hierarchical & P2P, also the solution avoids the limitations of these techniques. The aim of this model is to solve the issues of scalability, reliability, and dynamicity in large-scale grid systems. This goal is achievable by integrating SNMP in a hybridized model. The authors are currently investigating, through real grid environment, the performance of the model and preliminary findings are promising.

ACKNOWLEDGEMENTS

REFERENCES

[1] I. Foster, *et al.*, "Cloud Computing and Grid Computing 360-Degree Compared," *Gce: 2008 Grid Computing Environments Workshop,* pp. 60-69,112, 2008.

[2] R. Ranjan, *et al.*, "Peer-to-peer-based resource discovery in global grids: a tutorial," *Communications Surveys & Tutorials, IEEE,* vol. 10, pp. 6-33, 2008.

[3] I. Foster, *et al.*, "The anatomy of the grid: Enabling scalable virtual organizations," *International Journal of High Performance Computing Applications,* vol. 15, p. 200, 2001.

[4] P. Kacsuk, *et al.*, "Can we connect existing production grids into a world wide grid?," *High Performance Computing for Computational Science-VECPAR 2008,* pp. 109-122, 2008.

[5] A. Hameurlain, *et al.*, "Large scale data management in grid systems: a survey," 2008, pp. 1-6.

[6] A. Sharma and S. Bawa, "Comparative Analysis of Resource Discovery Approaches in Grid Computing," *Journal of Computers,* vol. 3, p. 60, 2008.

[7] A. Hameurlain, *et al.*, "Resource discovery in grid systems: a survey," *International Journal of Metadata, Semantics and Ontologies,* vol. 5, pp. 251-263, 2010.

[8] L. X. X. W. X. Lei and L. J. T. Jiuyang, "Grid Resource Discovery Mechanism Based on Compound Topology," *Computer Engineering and Applications,* vol. 9, 2005.

[9] M. Riedel, *et al.*, "Web services interfaces and open standards integration into the European UNICORE 6 grid middleware," 2008, pp. 57-60.

[10] S. Fitzgerald, *et al.*, "A directory service for configuring high-performance distributed computations," 2002, pp. 365-375.

[11] G. Ramos and M. A. de Melo, "An extensible resource discovery mechanism for grid computing environments," 2006, pp. 115-122.

[12] I. Foster and C. Kesselman, "The Grid: blueprint for a future computing infrastructure," ed: Morgan Kaufmann Publishers USA, 1999.

[13] G. Moltó, *et al.*, "A service-oriented WSRF-based architecture for metascheduling on computational Grids," *Future Generation Computer Systems,* vol. 24, pp. 317-328, 2008.

[14] M. L. Massie, *et al.*, "The ganglia distributed monitoring system: design, implementation, and experience," *Parallel Computing,* vol. 30, pp. 817-840, 2004.

[15] Y. Yin, *et al.*, "The grid resource discovery method based on hierarchical model," *Information Technology Journal,* vol. 6, pp. 1090-1094, 2007.

[16] L. Khanli and S. Ebadi, "A New Distributed and Hierarchical Mechanism for Service Discovery in Grid Environment," *Advances in Grid and Pervasive Computing,* pp. 174-182, 2010.

[17] X. H. Song, *et al.*, "The distributed spanning-tree based service discovery in grid environment," 2006, pp. 2379-2386.

[18] A. Iamnitchi, *et al.*, "A peer-to-peer approach to resource location in grid environments," *INTERNATIONAL SERIES IN OPERATIONS RESEARCH AND MANAGEMENT SCIENCE,* pp. 413-430, 2003.

[19] I. Filali, *et al.*, "A simple cache based mechanism for peer to peer resource discovery in grid environments," 2008, pp. 602-608.

[20] C. Mastroianni, *et al.*, "A super-peer model for resource discovery services in large-scale grids," *Future Generation Computer Systems,* vol. 21, pp. 1235-1248, 2005.

[21] D. Puppin, *et al.*, "A grid information service based on peer-to-peer," *Euro-Par 2005 Parallel Processing,* pp. 454-464, 2005.

[22] M. Marzolla, *et al.*, "Resource discovery in a dynamic grid environment," 2005, pp. 356-360.

[23] M. Cai, *et al.*, "Maan: A multi-attribute addressable network for grid information services," *Journal of Grid Computing,* vol. 2, pp. 3-14, 2004.

[24] A. Andrzejak and Z. Xu, "Scalable, efficient range queries for grid information services," 2002.

[25] D. Oppenheimer, *et al.*, *Scalable wide-area resource discovery*: Citeseer, 2004.

[26] D. Talia, *et al.*, "Peer-to-peer models for resource discovery in large-scale grids: a scalable architecture," *High Performance Computing for Computational Science-VECPAR 2006,* pp. 66-78, 2007.

[27] I. Foster, *et al.*, "Brain meets brawn: Why grid and agents need each other," 2004.

[28] S. Ding, *et al.*, "A heuristic algorithm for agent-based grid resource discovery," 2005, pp. 222-225.

[29] X. Tang and L. Huang, "Grid resource management based on mobile agent," 2006, pp. 230-238.

[30] K. Jun, *et al.*, "Agent-based resource discovery," 2000, p. 43.

[31] J. Yu, *et al.*, "Grid resource management based on mobile agent," 2006, p. 255.

[32] M. Yan, *et al.*, "Study of grid resource discovery based on mobile agent," 2007, pp. 570-571.

[33] M. Puh, *et al.*, "Multi-agent system for resource discovery in grid network," 2007, pp. 320-324.

[34] S. Manvi, *et al.*, "An agent-based resource allocation model for computational grids," *Multiagent and Grid Systems,* vol. 1, pp. 17-27, 2005.

[35] G. Kakarontzas and I. K. Savvas, "Agent-based resource discovery and selection for dynamic grids," 2006, pp. 195-200.

[36] H. Huang*, et al.*, "PChord: a distributed hash table for P2P network," *Frontiers of Electrical and Electronic Engineering in China,* vol. 5, pp. 49-58, 2010.

[37] D. Harrington*, et al.*, "An architecture for describing SNMP management frameworks," 1998.

[38] W. Stallings, *SNMP, SNMPv2, SNMPv3, and RMON 1 and 2*: Addison-Wesley, 1999.