An Improved Resource Discovery Approach Using P2P Model for Condor: A Grid Middleware

Anju Sharma, and Seema Bawa

Abstract—Resource Discovery in Grids is critical for efficient resource allocation and management. Heterogeneous nature and dynamic availability of resources make resource discovery a challenging task. As numbers of nodes are increasing from tens to thousands, scalability is essentially desired. Peer-to-Peer (P2P) techniques, on the other hand, provide effective implementation of scalable services and applications. In this paper we propose a model for resource discovery in Condor Middleware by using the four axis framework defined in P2P approach. The proposed model enhances Condor to incorporate functionality of a P2P system, thus aim to make Condor more scalable, flexible, reliable and robust.

Keywords—Condor Middleware, Grid Computing, P2P, Resource Discovery.

I. INTRODUCTION

COMPUTATIONAL Grids as well as peer-to-peer (P2P) networks are emerging technologies enabling widespread sharing of distributed resources. Researchers are trying to include the concept of P2P into the Grids. The term peer-to-peer refers to the concept that in a network of equals (peers) using appropriate information and communication systems, two or more individuals are able to spontaneously collaborate without necessarily needing central coordination. The P2P models can ensure scalability in Grids: Designers can use P2P philosophy and techniques to implement non-hierarchical decentralized Grid systems. P2P techniques can particularly be useful to manage two key services in Grid information systems namely: membership management (or simply membership) and resource discovery. The resource discovery service is invoked by a node when it needs to discover and use hardware or software resources having specified characteristics.

Resource discovery is the systematic process of determining which grid resource is the best candidate to complete a given job in shortest amount of time, with most efficient use of resources, and at minimum cost. Resource discovery in large heterogeneous Grid environments becomes increasingly difficult when request is very specific [1]. Resource discovery in Grid environments is based mainly on centralized or hierarchical models [2]. Such models are built to address the requirements of organizational Grids; these do not deal with more dynamic, large-scale distributed environments, in which useful information servers are not known a priori. The numbers of queries in such environments quickly make a

Authors are with Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala, India (e-mail: {anjusharma, sbawa}@tiet.ac.in).

client server approach ineffective. To overcome these limitations, future Grid systems should implement a P2P-style decentralized resource discovery approach that can support Grids as open resource communities. To make the concept of the resource discovery process more efficient and managed, Iamnitchi [18] proposed P2P resource discovery approach for Grids.

In this paper, we propose a model by using P2P approach for making Condor to behave as the P2P system. Basically, Condor is a specialized workload management system for compute-intensive jobs that can be used to manage a cluster of dedicated compute nodes. Condor maintains the High Throughput Computing resource and job management software originally designed to harness idle CPU cycles on heterogeneous pool of computers. In essence a workload management system for compute intensive jobs, it provides means for users to submit jobs to a local scheduler and manage the remote execution of these jobs on suitably selected resources in a pool.

The aided aspect of our model is to improve the functionality of Condor. Some key issues behind this model are: interprocess communication, network communication, handles the multiprocess jobs etc.

II. RESOURCE DISCOVERY

The basic service in Grid Computing is Resource Discovery. In large scale Grid environments, resource discovery is made challenging by potentially large number of resources and considerable heterogeneity in resource types and resource requests [3]. Resource discovery can be defined as a directed to the spontaneous network's environment [4].

Many resource discovery mechanisms have been proposed in the literature of Grid environments. Some of them, such as MDS2 *(Metacomputing Directory Service)* [5] and Data Grid Resource Discovery [6], are specific resource discovery services. However, the great majority of them are contained in more general grid proposals such as MDS3 *(Monitoring and Discovery Services)* [7], NWS *(Network Weather Service)* [8], VIRD *(Vega Infrastructure for Resource Discovery)* [9], GLOPERF [10] and NimRod/G [11]

The resource discovery mechanisms proposed for grids are either hierarchical or centralized. Most of them mechanisms retrieve data related to the machines that compose the grid (operating system used, CPU load, and memory occupation, among others). This section discusses the concept of existing resource discovery process in the Condor that is centralized

and then it has been correlated with the resource discovery in P2P systems i.e. hierarchical.

A. Condor Resource Discovery

The Condor Project [12] focuses on distributed high-throughput computing for the past eighteen years. Condor maintains the High Throughput Computing resource and job management originally designed to harness idle CPU cycles on heterogeneous pool of computers [13].

In essence a workload management system for compute intensive jobs, it provides means for users to submit jobs to a local scheduler and manage the remote execution of these jobs on suitably selected resources in a pool. Condor does not require an account (login) on machines where it runs a job.

For Resource Discovery, Condor uses the matchmaking approach. The basic idea of matchmaking is simple: Entities which provide or require a service, advertise their characteristics and requirements in classified advertisements (classads). A designated matchmaking service (matchmaker) matches classads in a manner that satisfies the constraints specified in the respective advertisements and informs the relevant entities of the match. The responsibility of the matchmaker then ceases with respect to the match. The matched entities establish contact; possibly negotiate further terms; and then cooperate to perform the desired service [14]. Condor's key activities - job-resource allocation, job startup and execution, metadata collection and display – are kept separate, allowing compartmentalization of Condor into clearly defined components, distributed amongst submission site, central manager and execution site, as illustrated in Fig.1:
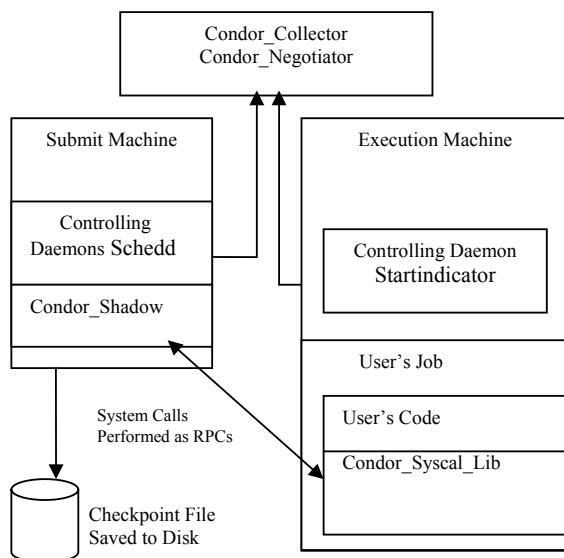

Fig. 1 Condor Architecture Overview

B. Resource Discovery in Peer - to - Peer based Approach

P2P paradigm is based on the principle that every component of the system has the same responsibilities acting simultaneously as a client and as a server, as opposed to the traditional client-server model. Resource Discovery service acts as an intermediary between users and resource providers. Resource discovery in grid environments shares many elements in common with resource discovery in P2P networks. P2P systems are divided into two main categories based on the connection protocol they employ and the way that the peers are organized, namely centralized and distributed.

Centralized resource discovery schemes such as Napster [15] and Globus (original MDS implementation) [16] are based on a cluster of central servers hosting a directory of resources. These introduce issues such as scalability, since the entire index must be stored on a single cluster. Centralized schemes can be vulnerable to attack, such as denial-of-service (DoS) attacks, because there is a single point of failure. Distributed solutions such as Gnutella [17] discover resources by broadcasting queries to all connected nodes. When a node receives a query, its list of local resources is checked for matches, and any results are back propagated to the requester. Regardless of whether a match has been found, a time-to-live (TTL) counter is decremented and, if greater than zero, the query is forwarded to neighboring nodes.

Centralized architectures arguably provide the best performance in terms of number of messages required to give an answer to a query, but suffer weaknesses such as of less scalability and single point-of failure.

Today, the Grid community agrees that in large scale Grids it is not feasible to adopt a centralized or hierarchical approach for providing a scalable resource discovery service. To overcome this Iamnitchi, proposed peer-to-peer resource discovery approach for a large collection of resources [18]. Different resource discovery problems are there in large distributed resource-sharing environment especially in grid environment [19]. In this paper, four different architectural components and four environment parameter factors are identified, which dominate the performance and design strategies for a resource discovery solution. Using four axes framework, it is possible to design any resource discovery architecture in a grid [20].

III. PROPOSED HYBRID MODEL

A hybrid model can be defined as a mixture of two performance based models i.e. impression based (CPM) and performance based (CPC or CPA) [21]. It is seen as a way to further split the risk between the consumer and provider or publisher and advertiser. For the good understanding of it, we propose a hybrid model by using the four axis framework proposed by the Iamnitchi [22] in his P2P approach. This model aims to improve the performance of Condor. The following sections discuss the four axis framework i.e. Membership Protocol, Overlay Construction, Preprocessing and Request Processing. After it we define them in terms of Condor that how we use it?

A. Membership Protocol

The membership management protocol is responsible for incorporating new members or nodes into the system and reallocates members from group that has too few members.

The membership protocol [2] exploits the characteristics of contact nodes. A contact node is a Grid node that plays the role of an intermediary node during the building process of the

Grid network. One or more contact nodes are made available by each Virtual Organization. Whenever a Grid node wants to connect to the network, it contacts a subset of contact nodes and registers at those nodes. In turn the selected contact nodes randomly choose a number of previously registered Grid nodes and communicate their addresses to the requesting node: these nodes will constitute the neighbour set of the new Grid node.

In Condor there is a central manager that maintains all the information about the nodes, but in proposed model maintenance is done by using membership protocols. Here we define different groups and each group has its own leader. In each group, all the nodes are equally treated i.e. there is no specific head node. But in case of inter communication between the different groups, the head node is consider. By implementing these protocols network communication is easy as condor has not much network communication.
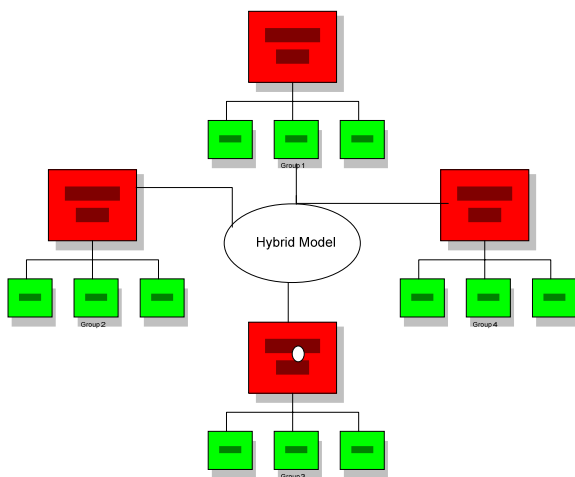

Fig. 2 Proposed Hybrid Model

B. Overlay Construction

The overlay construction functions select the set of active collaborators from the local membership list. In practice, this set may be limited by such factors as available bandwidth, message-processing load, security or administrative policies, and topology specifications [23]. This membership list is maintained by each local node. Also there is one supernode that maintains the information about all the nodes in the group.
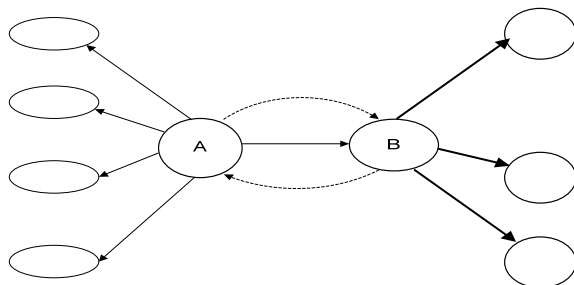

Fig. 3 Random Pointer and Jump

Communication between the different group leaders i.e. the supernodes is done on the basis of the random pointer jump algorithm [24]. This algorithm states that in each round, each node contacts with random neighbor and then random neighbor sends all of its neighbors to the sender node. For Example, in the above figure A and B are two nodes or can be two supernodes. For making the communication between these two nodes, i.e. node A sends the message to the node B. Then node B introduces its neighbours to the node A. In this way the nodes within the group or also between different groups interact. To overcome the limitation of condor is that is interprocess communication is not allowed but, by using this algorithm this is possible.

C. Preprocessing

Preprocessing refers to the off-line preparations for better search performance, independent of request. A very well known example of preprocessing technique is dissemination of resource descriptions that is, advertising descriptions of the local resources to other areas of the network for better search performance and reliability. For this the techniques like Indexing or Dynamic Hash Table (DHTs) [25] is used. In DHTs each node in the network hosts part of the index, and queries are hashed to create a key that is mapped to the node with the matching identifier. There are different DHT algorithms used to handle the request for discovering the resources. These algorithms allow locating a peer in a P2P network which holds some information for a specific key. They provide a single method: (lookup key), which returns the node responsible for the given key. By using indexing in condor middleware we can specify the priority of the jobs like which job is come first and then second and so on.

D. Request Processing

The request processing has the two components:

Local processing: Looking up the requested resource in the local information. (e.g., a request for A and B could be broken into two distinct requests to be treated separately), and/or applies local policies, such as dropping requests unacceptable for the local administration. The local information is maintained by the every node using the above mentioned preprocessing techniques.

Remote processing: The remote component implements the request propagation rule: sending requests to other peers through various mechanisms (flooding, forwarding, and epidemic communication). This is current area of research in P2P approach. In hybrid model we are using the random pointer and jump algorithm to remotely interact with the system. By designing these algorithms in the hybrid model multiprocess jobs are run carefully.

IV. ADVANTAGES OF PROPOSED MODEL

The various advantages our system will provide are:

- **Independence:** from central global control: Our model will step towards the pure decentralized system for Condor using the membership protocol.
- **Fast discovery of resources:** using DHTs and indexing concept the resources can be fastly discovered.
- **Scalability:** As the control is not decentralized, the scalability of the system is increased. Also with the use of grouping concept, we can extend the Condor as the global grid.
- **Support for:** intermittent resource participation.

The Table I summarize the basic differences between the original Condor and for hybrid proposed model.

TABLE I
COMPARISON OF CONDOR WITH P2P

Feature	Condor	Proposed Hybrid Model
Centralized vs Distributed	Managed by central manager	Independence from central manager global control
Discovery Mechanism	Discovery is done using Matchmaking	DHTs based technique is implemented for making the discovery fast
Scalability	Scalability is limited	Scalability is Increased.
Environment	Powerful, good connection	Highly Complex

V. CONCLUSION

The P2P model is emerging as a new distributed computing paradigm because of its immense potential to harness the computing, storage, and communication power of hosts in the network to make their underutilized resources available to others. P2P shares this goal with the Grid, which was designed to provide access to remote computing resources for high-performance and data-intensive applications.

We have proposed in this paper, a hybrid model for the Condor to make it a Peer-to-Peer system. P2P approach has been used for resource discovery to make the Condor middleware more scalable. In our hybrid model the four axis framework is used.

Firstly, membership protocol defines the multi grouping approaches for the management of different groups.

Secondly, in the overlay construction we have used the Random Pointer and Jump algorithm to make the communication between the various groups possible.

Thirdly, for making the resource discovery more efficient and fast, we have preprocessed the resources using the indexing and the DHT techniques.

At last, the request for the resources is done in two ways, the first is the Local Processing and the second one is the Remote Processing. Local Processing lets the members to have the information about the members in the groups and remote processing is used for making the communication between different groups.

A basic framework has been developed which can further be enhanced, customized and implemented. This model aims to improve most of the centralized systems by letting them behave as the P2P system.

REFERENCES

[1] Yili Gong, Wei Li, Yuzhong Sun, Zhiwei Xu, "A C/S and P2P Hybrid Resource Discovery Framework in Grid Enviornments", ICPP 2005, pp. 261-268.
[2] Carlo Mastroianni, Domenico Talia and Oreste Verta, "A P2P Approach for Membership Management and Resource Discovery in Grids", ITCC 2005, Volume II, pp. 168-174.
[3] Sanya Tangpongprasit, Takahiro Katagiri, Kenji Kise, Hiroki Honda, Toshitsugu Yuba, "A time-to-live based reservation algorithm on fully decenteralized resource discovery in Grid Computing" Elsevier Parallel Computing 31, 2005, pp 529-543.
[4] G. Coulouris, J. Dollimore, T. Kindberg, "Distributed Systems: Concepts and Design", Addison-Wesley, 2001, 772p.
[5] S. Fitzgerald, I. Foster, C. Kesselman, G. Von Laszewski, W.Smith, S. Tuecke. A directory Service for Configuring High Performance Distributed Computations. Proc. 6th IEEE Symposium on High Performance Distributed Computing, pp. 365-375, 1997.
[6] S. jayasena, C. Yee, J. Song, A. Stoelwinder, C. W. See and W. Wong, " Data Resource Discovery in a Computational Grid", Proceeding of the Grid and Cooperative Computing International Workshops, 2004, pp. 303-310.
[7] T. Sandholm, J. Gawor, " lobusToolkit3 Core – "A Grid Service Container Framework", Technical Report, available at www-unix.globus.org/toolkit/3.0/ogsa/docs/gt3_core.pdf,pp. 22.
[8] R. Wolski, N. Spring and J. Hayes, "The Network Weather Service: A Distributed Resource Performance Forcasting Service for Metacomputing", Future Generation Computer Systems, vol. 15, no. 5-6, pp. 746-768, October 1999.
[9] Y. Gong, F. Dong, W. Li and Z.Xu, Vega Infrastructure for Resource Discovery in Grids. Journal of Computer Science Technology, 18(4): 413-422, July 2003.
[10] C. A. Lee, J. Stepanek, R. Wolski, C. Kelleseman and I. Foster. " A Network Performance Tool for Grid Environments", In Proceeding of 7th IEEE Internatinal Symposium on High Performance Distributed Computung, pp 260-267, 1998.
[11] R. Buyya, D. Abramson and J. Giddy, Nimrod/G: "An Architecture for a Resource Management and Sheduling Syatem in a Global Computational Grid", Proceding of the High Performance Computing in the Asia-Pacific Region, China, pp 283-289, 2000.

[12] http://www.cs.wisc.edu/condor/

[13] Rajesh Raman, Miron Livny, Marvin Solomon, "Matchmaking: Distributed Resource Management for High Throughput Computing", HPDC 1998.

[14] http://www.cs.wisc.edu/condor/manual/v6.6.10

[15] Saroiu, S., Gummadi, P.K. and Gribble, S.D. (2002), "A measurement study of peer-to-peer file sharing systems", Proceedings of Multimedia Computing and Networking 2002 (MMCN'02), San Jose, CA, 18-25 January.

[16] Fitzgerald, S., Foster, I., Kesselman, C., von Laszewski, G., Smith, W. and Tuecke, S., "A directory service for configuring high-performance distributed computations", Proceedings of the 6th IEEE Symposium on High Performance Distributed Computing, Portland, OR, 1997, pp. 365-76.

[17] Ripeanu, M., Peer-to-Peer Architecture Case Study: Gnutella Network, University of Chicago Technical Report TR-2001-26, University of Chicago, Chicago, IL, 2001.

[18] Iamnitchi and I. Foster, "On Fully Decentralized Resource Discovery in Grid Environments," IEEE International Workshop on Grid Computing, Denver, CO, 2001.

[19] Iamnitchi, I. Foster, Daniel C. Nurmi, "A Peer-to-Peer Approach to Resource Discovery in Grid Environments," Proc. of the 11th Symposium on High Performance Distributed Computing, Edinburgh, UK, 2002.

[20] W. Hoschek, "A Unified Peer-to-Peer Database Framework for Scalable Service and Resource Discovery," Proc. of Third International Workshop on Grid Computing: GRID 2002, Baltimore, MD. , Springer, 2002, pp. 126-144.

[21] http://www.marketingterms.com/dictionary/hybrid_model

[22] A.I. Iamnitchi, "Resource Discovery in Large Resource Sharing Environments", Chicago Illinois, 2003.

[23] M. Harchol-Balter, T. Leighton, D. Lewin, "Resource Discovery in Distributed Networks," 18th ACM Symposium on Principles of Distributed Computing, 1999.

[24] Nick Antonopoulos and James Salter, "Efficient resource discovery in Grids and P2P networks".

[25] Adriana Iamnitchi1 and Ian Foster, "A Peer-To-Peer Approach to Resource Location in Grid Environments".

Anju Sharma, 1981 holds M.Sc. (I.T.) degree form Punjab University Chandigarh. She is currently pursuing Ph.D in the area of Grid Computing from the Thapar institute of Engineering & Technology, Patiala, India**.** Her areas of interests include Parallel and Distributed Computing, Grid Computing, Resource Management, Resource Discovery and Peer to Peer Networks.

Seema Bawa, 1970 holds M.Tech (Computer Science) degree form IIT Khargpur and Ph.D. from Thapar Institute of Engineering & Technology, Patiala. She is currently Professor and Head of Computer Science and Engineering Department. Her areas of interests include Parallel and Distributed Computing, Grid Computing, VLSI Testing and Network Management. Prof. Bawa is member of IEEE, ACM, Computer Society of India, and VLSI Society of India.