

# Semantic Resource Discovery in Large Scale Environments\*

Imen Ketata, Riad Mokadem, Franck Morvan  
Institut de Recherche en Informatique de Toulouse IRIT  
Paul Sabatier University  
118 Route de Narbonne F-31062  
Toulouse Cedex 9  
France  
{ketata, mokadem, morvan}@irit.fr



**ABSTRACT:** *In biosciences, data mining is concerned with processing large amount of data which is characterized by heterogeneity, ever changing and spread in different complex environments. Resource discovery from massive data poses a formidable task for many newer as well as routine applications. The issues addressed in the massive data environments so far are the heterogeneity issues and the semantic focus is less. In the current work, we deal with the resource discovery in large-scale environments (as data grid systems) considering data semantic heterogeneity of biomedical sources. There are many benefits such as-(i) allowing a permanent access, through an addressing system, from any domain ontology  $DO_i$  to another  $DO_j$  (inter-domain discovery) despite peers' dynamicity, (ii) reducing the maintenance cost and (iii) taking into account the semantic heterogeneity.*

## Categories and Subject Descriptors

**E.1 [Data Structures];** Distributed data structures: **H.3.3 [Information Search and Retrieval]** J.3 [Life and Medical Sciences]; Biology and genetics

**General Terms:** Semantic data grid, Biomedical resources discovery, Distributed data

**Keywords:** Semantic Grid, Biomedical Data Sources, Resource Discovery

**Received:** 21 February 2011, Revised 30 March 2011, Accepted 5 April 2011

## 1. Introduction

Computer applications in the biomedical domain accumulate a large number of data sources. These sources are voluminous, daily published (dynamicity of the environment) and with high heterogeneity. Thanks to its large storage capacity and scalability, the data grid is the most appropriate infrastructure adopted in this domain. Resource discovery presents an important step in the SQL-query processing in such environment. Several resource discovery solutions have been proposed. However, the structural heterogeneity problems [7] have been more widely studied than semantic ones. In this paper, we deal with the resource discovery in a large-scale environment as data grid systems, with considering semantic heterogeneity (e.g., polysemy, synonymy, abbreviation, etc).

There are first the traditional approaches based on centralized [27], [16] and [18] and hierarchical systems [24] and [8]. Whereas,

they do not achieve scalability in such environments [13] when the peer-to-peer systems have proved their efficiency with respect to this scalability (e.g., the biomedical application: "Drug Target Validation [7]). However, most of the resource discovery works, dealing with peer-to-peer techniques do not effectively respond to the semantic heterogeneity problems because of the use of keyword method. For example, matching relations are established between keywords in PeerDB [21]. Whereas, this approach may return wrong results because of the strong semantic heterogeneity in biomedical domain, e.g., Doctor in biomedical domain corresponds to a Practitioner and Doctor in a biological domain corresponds to a Biologist having a Ph.D. degree in biology. Piazza [12] also establishes correspondence relations (*mapping*) between peers (in pairs or by groups) generating a large number of mappings. Keeping these mappings up to date becomes a complicated task. To discover resources more semantically, Hyperion [4] uses expressions and tables of mapping [17] to illustrate these correspondence relations. This technique helps to describe any semantic relationship between existing peer's schemas, whereas, such mapping is manual. Hence, having a completed and updated resource discovery result with this kind of mapping is not always obvious and easy. Otherwise, besides keyword approach, there is also global ontology solution [11] and [7]. However, using such ontology shared by all peers in the network is still a very complex task regarding the large number of domains in biomedical environment. The emergence of domain ontology solution, i.e., an ontology is associated to each domain [20], deals more efficiently with problems of semantic heterogeneity. For example, [3] applies a peer-to-peer technique as Chord [26] for an efficient resource discovery by handling the problem of semantic heterogeneity using domain ontologies. Hence, a domain ontology is duplicated in each peer. Nevertheless, semantic mappings between domain ontologies, specifying how those latter are linked together have not been processed.

In this paper, we introduce a method for data resource discovery in a biomedical data grid environment considering data semantic heterogeneity. The resource discovery concerns metadata describing the data sources. For this goal, peers are grouped by domains and an ontology is associated to each domain in which all peers use the same vocabulary. Thus, a community called Virtual Organization (VO) is associated to each domain. For efficiency purposes, we are based on structured peer-to-peer technique, e.g., a Distributed Hash Table (DHT) [26]. Hence, the broadcast of resource discovery queries in one VO is realised by a classic DHT lookup. Otherwise, when a resource discovery query is transmitted from one domain ontology to another, this

\*This work was supported in part by the French National Research Agency ANR, PAIRSE Project, Grant number -09-SEGI-008.

technique is no applied. In this case, the search is mainly based on mapping's relationships taking into account the dynamic nature of peers. Our method allows permanent access from any domain ontology  $DO_i$  to another one  $DO_j$  ( $i \neq j$ ) despite peers' dynamicity. Thus, we have developed an addressing system allowing a low cost maintenance system. The main advantages of our method are: (i) applying the principle of data locality during the metadata discovery phase, (ii) allowing a robust inter-domain discovery with respect to the connection / disconnection of a large node number requiring low maintenance cost and (iii) taking into account the semantic heterogeneity.

The rest of this paper is organized as follows. In the second section, we propose a resource discovery method dealing with semantic heterogeneity. We illustrate our method by resource discovery process examples. We also discuss the maintenance process. We present some related work in the third section. Finally, we conclude our work by considering some prospects.

## 2. Resource Discovery taking into account Semantic Heterogeneity

The resource discovery process is an important step in the query processing in a large-scale environment. In such environment, every node manages several data sources. Moreover, the instability of such environment presents a major problem. Thus, we can not have a centralized and global schema to locate all resources. The duplication of this type of schema will also cause update problems. Consequently, the most appropriate solution is the distribution of such schema [19]. In this section, we focus on the distributed resource discovery on large-scale environments taking into account the semantic aspect. Specifically, we are interested in the metadata which describes data sources in a biomedical database context with minimizing maintenance costs. We cite the example of metadata discovery of a relation  $R$  which we associate to a concept in an ontology. These metadata contain: (i) attributes of  $R$ , (ii) placement of  $R$  (IP address of each fragment, fragmentation, information duplication and construction rules of  $R$ ) and (iii) the different statistics of  $R$ , e.g., the size.

### 2.1 Architecture

Developing a global ontology is a difficult task to achieve, given the diversity of domains in a biomedical environment. Indeed, each biomedical domain accumulates large masses of heterogeneous data sources. One solution is to decompose the environment into domains as shown in Fig. 1. Hence, each domain can be handled independently. Thus, a community called Virtual Organization (VO) is associated to each domain of biomedicine (see Fig. 1). By this way we allow to take into account the locality and autonomy principles of each VO. On the other hand, solving problems such as synonymy and polysemy in resource discovery process is a difficult task to accomplish using only the schema principle. Then, a domain ontology [20] is associated to every VO (biomedical domain) as shown in Fig. 1. Hence, all peers in one VO use the same language (vocabulary).

For resource discovery efficiency reasons, a structured peer-to-peer system is associated with each VO. Thus, a DHT (Distributed Hash Table) [26] is associated to peers belonging to the same domain ontology. The resource discovery within a single domain ontology is then a classic discovery by a DHT. Now, the way in which domain ontologies communicate between each other should be defined. Indeed, ontologies must be connected. Hence, we use relations (rules) of mapping between peers belonging to different domain ontologies. After that, we define a protocol to link VOs between each other.

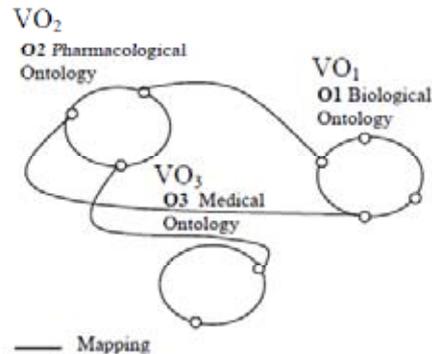


Figure 1. Example of Virtual Organisation Distribution

Let  $N(VO_i)$  be the set of virtual organizations connected to  $VO_i$  through mapping relationships ( $VO_s$  neighbors of  $VO_i$ ).  $VO_s$  as well as mapping relationships form a graph noted:  $G(N, E)$ , with  $N$  the set of vertices presenting the set of VOs and  $E$  the edges presenting mappings. We note that the edge  $e_{ij} \in E$  only if there exists a mapping between  $VO_i$  and  $VO_j$ . Let  $|N(VO_i)|$  be the number of VOs neighboring  $VO_i$ .

To ensure the completeness of resource discovery results, the graph must be connected. Thus, we assume in the rest of this paper, that the graph is connected. So, for two vertices  $VO_i \in N$  and  $VO_j \in N$ , there is a path  $P_{ij} \in E$  from  $VO_i$  to  $VO_j$ . Also, since grid is an unstable environment, a peer  $P_k$  belonging to a  $VO_i$  should be able to initiate, at any time, a discovery process on a peer  $P_k$ , belonging to  $VO_j / VO_j \in N(VO_i)$  and  $i \neq j$ . Then, for each node in a  $VO_i$  we associate  $|N(VO_i)|$  access points ( $AP_k$ ) in the DHT. Access points are peers connected to different nodes of VOs (biomedical domains) available via the network. We note that two nodes belonging to a first  $VO_i$  do not necessarily have the same access point to a second  $VO_j$ . By this way we can avoid having a central access point to one VO (bottleneck). This can also help to avoid the single point of failure in case of disconnection of the access point, especially in an unstable environment as grid. After describing how VOs are connected, we will focus on the resource discovery process.

### 2.2 Resource Discovery

In our system, we classify resource discovery queries into two types: query within a single domain ontology (intra-domain ontology) and queries between domain ontologies (inter-domain ontologies). For inter-domain ontology query (between domains), we define a protocol to connect the VOs between each other.

#### 2.2.1 Intra-Domain Ontology Resource Discovery

As noted earlier, the intra-domain ontology queries are evaluated according to the routing system of a classic Distributed Hash Table DHT [26]. Indeed, structured peer-to-peer systems have proved their efficiency with respect to the scalability and research process. Thus, a hash function is applied whose key is the name of the relation (the concept to discover). If the node, which is defined as responsible of the resource to be discovered, belongs to the same ontology in which the query was occurred, the query is an intra-domain ontology query (Fig. 2). In this case, the complexity to find this node is  $O(\log(N))$  where  $N$  presents the number of nodes [26]. The result of this discovery is the metadata describing the relation (concept).

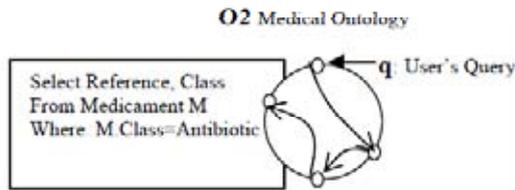


Figure 2. Example of Intra-Domain Ontology Resource Discovery Query

### 2.2.2 Inter-Domain Ontology Resource Discovery

To have a final resource discovery result with the most complete responses, the research should also be made in other peers from other biomedical domains (VOs).

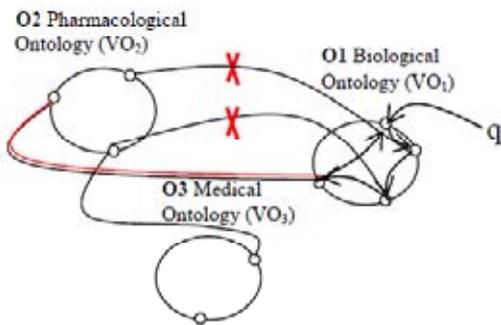


Figure 3. Example of Inter-Domain Ontology Resource Discovery Query Process

Let  $P_i$  be the node receiving the user query. For an inter-domain ontology resource discovery,  $P_i$  contacts all its access points. The discovery process is realized by translating the original concept of the initial  $VO_i$  (where the user's query has been occurred) to a  $VO_k$  through a mapping. To avoid the endless waiting of an access point response, we define an interval of time, e.g., a RTT (Round-Trip-Time). Thus, if an access point does not respond after one RTT, it is considered to be disconnected. In this case, the peer  $P_i$  looks for another access point by contacting the nearest neighbor in the DHT. This research is done on the  $P_i$ 's neighbor and then its neighbor and so on, until retrieving a peer presenting an access point to the researched distant  $VO_k$ . If all neighbors have been contacted without any response, the concerned  $VO_j$  is considered to be disconnected and only the administrator can reconnect it. We discuss this case in the further disconnection section. We demand to this access point one of its neighbors. This is done to avoid that all nodes of the graph converge towards a single access point. The response of resource discovery found is sent to the first query sender node  $P_i$  and noted  $R_{Sender}$  in the Inter-Domain Ontology Resource Discovery Algorithm (Fig. 4). At this moment, we take the opportunity to update  $VO_j$ 's access points. We keep, of course, the path that the query followed along the discovery process for its translation from one VO to another. Indeed, this path will be used to translate the user's query (concepts) when it is sent from one domain ontology to another. Fig. 4 shows the algorithm of inter-domain ontology discovery. To simplify this algorithm we have not described the update of the access points in case of disconnection.

```

//R: Relation to be discovered (presenting
researched //concept name).
//R_Sender: Initiator resource discovery node
(which //have received the user query).
//VO: Current virtual organisation
//APk: Access point from one VO towards
another.
//APS: Set of access points.
//Path: Resource discovery process path.
//Lookup(R, VOAPk, R_Sender, Path): Discover the
relation //R in the VO via the APk node.
//TTL: Time-To-Live (limit of the propagation
range of //a message).

Metadata ← Lookup(R, VOAPk, R_Sender, Path);
//Intra-domain-ontology research TTL ←
TTL - 1;

If(TTL != 0) then
For each APk ∈ APS
Metadata ← Metadata U Lookup(Translate(R,
VO, VOAPk),
VOAPk, R_Sender, Path U VO); //Inter-domain-
ontology
//research
If (not Empty(Metadata)) then
Return(Metadata, R_Sender, Path U VO);

```

Figure 4. Inter-Domain Ontology Resource Discovery Algorithm

#### Example 1

Let's have two virtual organisations:

VO<sub>1</sub>: Biological domain and

VO<sub>2</sub>: Pharmacological domain.

Suppose that a node  $N_{11}$  in a VO<sub>1</sub> (biological VO), receives a user's query. In addition to the intra-domain ontology query, this node have to connect to another VO<sub>2</sub> (pharmacological VO) in order to have a more complete response for a relation (concept) in the user's query.

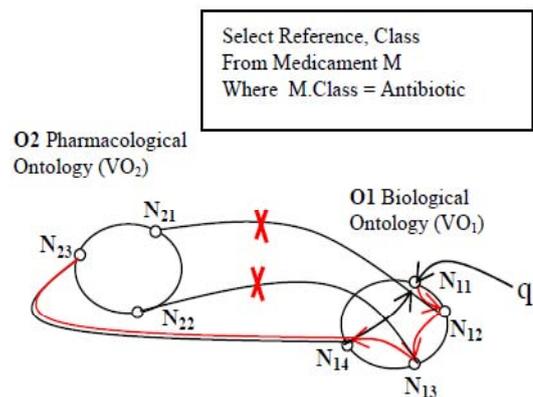


Figure 5. Example of Inter-Domain Ontology Resource Discovery Process

First,  $N_{11}$  looks for, in its neighbors, the nearest access point towards VO<sub>2</sub> which is  $N_{12}$  (connected to  $N_{21}$  in VO<sub>2</sub>). Supposed that  $N_{21}$  is disconnected. Then,  $N_{11}$  will look for the nearest neighbor of  $N_{12}$  with an access point to VO<sub>2</sub> and then the neighbor of this neighbor and so on (recursively) until finding an access point

connected to  $VO_2$ . It corresponds to  $N_{14}$  in our example. Once this  $VO_2$  is reached, the result (the metadata of the researched resource) is sent to the node initiating this discovery process (RSender) which is  $N_{11}$  in our example. Then, this node update its new access point ( $N_{14}$ ) towards  $VO_2$  (Fig. 5).

### 2.3 Maintenance

Since the dynamicity properties of grid environment, each node can join or leave the system at any time. In consequence, the system update will be required. In structured peer-to-peer systems such as Chord [26], the update of the DHT generates  $\text{Log}^2(N)$  messages for each connection / disconnection of a node. In this section, we show that our method can significantly reduce the cost of the system maintenance. In this context, we discuss two types of maintenance: (a) maintenance of the DHT and (ii) maintenance of the nodes with access points to other  $VO_j$  belonging to  $N(VO_i)$ .

We will not deal with the first case because maintaining the system is done by a classic maintenance of a DHT [26]. This occurs at the connection / disconnection of a node. Otherwise, the connection / disconnection of nodes presenting access points raises more complicated update problems. Maintaining such system is updating the access point nodes in their connection (disconnection). In what follows, we will consider two cases: (i) the connection of a node and (ii) the disconnection of a node.

#### 2.3.1 Node Connection

Supposed that a new node, called NewN, is connected to a  $VO_i$ . It looks for the access points to all other  $VO_s$ . For this reason, it contacts the nearest neighbor in  $VO_i$  to get its access points to these  $VO_s$ . If an access point towards a  $VO_k$  ( $k \neq i$ ) does not respond during a certain period of time (RTT), it is considered to be disconnected. Then, NewN contacts the neighbor of its neighbor. This corresponds to (Neighbor (Following)) in the maintenance algorithm shown in Fig. 6. It repeats this recursively until getting connected access point to the appropriate  $VO_k$ . The test is done via the `check()` function. If all the neighbors are contacted without response (a failure case), the appropriate  $VO_k$  is considered to be disconnected. If the access point is found, NewN sends it a message to get

```

//AP: Set of access points.
//Check(k): Check the connexion of a node
Pi with
//respect to a VOk.
//P(k): Access point retrieved in VOk after
a check
//test.

For k ∈ N(VO)
{
Found ← False;
Following = This.Neighbor;
While(Following != This) and (not Found)
{
Found ← Check(APk → Following());
If (Found) then
APk → Following();
Else Following = Neighbor(Following);
}
}

```

Figure 6. Maintenance Algorithm of Routing System (Node Connection Case)

its neighbor in the distant appropriate  $VO_k$ . This explain the instruction ( $AP_k \rightarrow \text{Following}()$ ) in Fig. 6. Then, NewN has a new access point to this  $VO_k$ .

#### Example 2

Let's have the same two VO's as example 1:

$VO_1$ : Biological domain and

$VO_2$ : Pharmacological domain.

And let's take also a third  $VO_3$ : Medical domain.

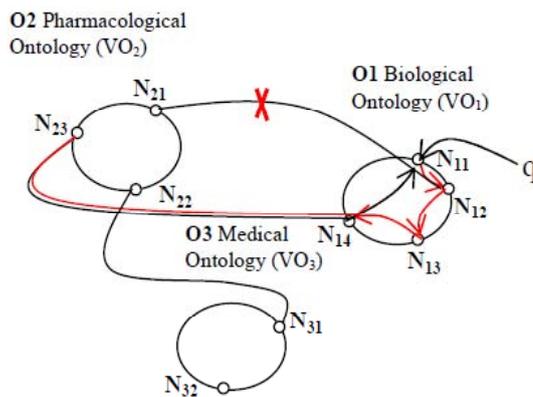


Figure 7. Example of a System Maintenance after a Node Connection

Suppose that a new node  $N_{11}$  is connected to a  $VO_1$ . This node wants to get access point to the other  $VO_2$  and  $VO_3$ . For this, it contacts the nearest neighbor  $N_{12}$  to get its access points to these two  $VO_s$ . In fact,  $N_{11}$  contacts  $N_{21}$  (the access point from  $VO_1$  to  $VO_2$ ).  $N_{21}$  does not respond during the RTT, it is considered to be disconnected and  $N_{11}$  contacts the neighbors of  $N_{12}$  recursively, to establish other access point to the appropriate  $VO_2$ . Therefore,  $N_{11}$  contacts the next neighbor  $N_{14}$ , an access point to  $VO_2$ . The corresponding node in  $VO_2$  is  $N_{23}$ . Thus,  $N_{11}$  gets  $N_{23}$  as access point towards  $VO_2$  and thereafter sends it a message to get its neighbors in  $VO_2$ .  $N_{11}$  process by the same way to get an access point to  $VO_3$ . But in our example there is no access point from  $VO_1$  to  $VO_3$ . That is way  $VO_3$  is considered to be disconnected.

#### 2.3.2 Node Disconnection

Supposed that one node, called NDisc disconnects from the system. A classic update of a DHT [26] is employed. When NDisc is an access point to a  $VO_i \in N(VO_i)$ , all the associated nodes in  $N(VO_i)$  must be updated. Two solutions emerge. One solution is that NDisc propagates the information to every  $VO_i$  towards which it is connected. We do not adopt this strategy since it proceeds by the flooding. Another solution, adopted in our system, is to apply a lazy maintenance. Each node that has just disconnected does not have to send anything to the appropriate nodes. None of the concerned  $VO_j \in N(VO_i)$  is informed by this disconnection. The access points towards this  $VO_j$  will be updated during the process of inter-domain ontology resource discovery. This reduces the number of node messages. The process of resource discovery is frequently performed so that the system can be updated. Consequently, the maintenance cost is significantly reduced.

#### Example 3

Let's have the same two VO's as examples 1 and 2:

$VO_1$ : Biological domain and

$VO_2$ : Pharmacological domain.

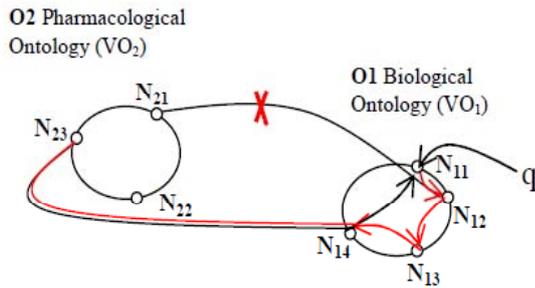


Figure 8. Example of a System Maintenance after a Node Disconnection

The disconnection of a node  $N_{12}$  from the network implies a classic update of a DHT associated to the VO to which it belongs [26]. More, when  $N_{12}$  is an access point to other VOs, all the access points associated with  $N_{12}$  must be updated. As we adopt lazy maintenance in our system, the disconnected node  $N_{12}$ , does not inform any associated nodes ( $N_{21}$  which belongs to  $VO_2$  in our example). Thus,  $VO_2$  is not informed by the disconnection of  $N_{12}$ . The access points towards this VO will be updated during the process of inter-domain ontology resource discovery. Consequently, the maintenance cost is significantly reduced.

### 3. Related Work

Many research works have used the peer-to-peer techniques for resource discovery in large-scale environment [21], [17], [4], [5], [25], [10], [11], [28], [7], [9], [2] and [3]. But, only few studies have focused on the semantic aspect during this discovery [22], [23] and [6]. [5] Introduces the semantic in web service discovery. It uses the unstructured peer-to-peer architecture based on the Gnutella protocol [1] and keywords' mappings. It handles the semantic by using the QDN (Querical Data Network) where the identity of a peer is identified by its data. However, discovery query service employs a broadcast mechanism by flooding, which can saturate the network. Resource discovery in PeerDB [21] and XPeer [25] were also based on keyword search, which is unreliable. Firstly, the connections made between these keywords require user's intervention to select significant query. Secondly, it generates a bottleneck and requires the resolving of fault tolerance problems. In this context, [4] and [10] are based on semantic relationship between elements of peer's schemas (mapping's tables [17]). However, the domain experts' intervention is not always obvious. The resource discovery service in [14] deals with semantics in a grid environment. Each pair has its own ontology presented by a DAG (Directed Acyclic Graph). To discover resources, structured peer-to-peer systems (DHT) are used. They allow the distribution of local graphs (DAGs) through the peer-to-peer network nodes. Thus, a distributed virtual view of all peers' graphs is conceived. [28] Also uses the DHT for the message broadcast between communities. Discovery service is based on the GSBLS method (Location Based Greedy Search). But, the problem of semantic heterogeneity between graphs is also present. SenPeer [9], a peer-to-peer distributed data management system, also uses the principle of keywords but introduces semantics by organizing peers by semantic domains. Each peer can get its data semantically as a graph structured formalism with keywords. Other studies are based on a global ontology or schema shared by all peers [11], [7], [2] and [3]. The Bibster system [11], for example, uses a routing system based on semantic similarity between the discovery query and each pair expertise description in order to select relevant peers. [7] Introduces the semantic mapping describing all semantic relationship between the elements of peer's schemas.

APPA [2] is based on a common schema called CSD (Common Schema Description) to distribute and share data between peers considering semantics. However, the use of a global ontology or schema is still a difficult task to achieve, due to the strong diversity of domains in the biomedical environment.

### 4. Conclusion and Future Works

We have proposed a resource discovery method in a large-scale environment taking into account the semantic aspect. Our solution allows the discovery of metadata describing biomedical data sources despite their semantic heterogeneity.

In the proposed solution, a domain ontology is associated to each biomedical domain whose peers constitute a virtual organization. Resource discovery queries within a domain (intra-domain ontology) follow the routing system of a classic DHT. For inter-domain ontology queries, we rely on the mapping relationships between peers belonging to different domain ontologies. We define a protocol to link virtual organizations between each other. Our method provides a permanent access from any domain ontology to any other one. It reduces resource discovery costs especially for intra-domain ontology query. It also provides low system maintenance costs despite the connection / disconnection of a large number of nodes.

Our method is particularly applicable to data grids which are particularly characterized by the high dynamicity of nodes and heterogeneity of data sources. For our future work, we are planning to carry out a performance evaluation of our method based on real experiences involving several heterogeneous data sources in biomedical environment.

### References

- [1] Gnutella Development Home Page, <http://gnutella.wego.com/>
- [2] Akbarinia, Reza., Martins, Vidal (2007). Data Management in the APPA System. In: Journal of Grid Computing.
- [3] Alking, Raddad Hameurlain, Abdelkader., Morvan, Franck (2008). Ontology-Based Data Source Localization in a Structured Peer-to-Peer Environment. In: IDEAS, Coimbra, Portugal.
- [4] Arenas, Marcelo., Kantere, Vasiliki., Kementsietsidis, Anastasios., Kiringa, Iluju., Miller, Ren\_ee., J Mylopoulos, John (2003). The Hyperion Project: From Data Integration to Data Coordination, In: SIGMOD Record Journal.
- [5] Banaei-Kashani, Farnoush., Chen, Ching-Chien., Shahabi, Cyrus (2004). WSPDS: Web Services Peer-to-peer Discovery Service. In: The International Conference on Internet Computing, Las Vegas, Nevada, USA.
- [6] Comito, Carmela., Mastroianni, Carlo., Talia, Domenico (2009). A Semantic-aware Information System for Multi-Domain Applications over Service Grids. In: IPDPS, Rome, Italy.
- [7] Cruz I. F., Xiao H., Hsu F. (2004). Peer-to-Peer Semantic Integration of XML and RDF Data Sources, In: International Workshop on Agents and Peer-to-Peer Computing AP2PC, New York, NY, USA.
- [8] Elmroth, E., Tordsson., J. (2005). An interoperable, Standards based Grid Resource Broker and Job Submission Service, In: The International Conference on e-Science and Grid Computing, Melbourne, Australia.
- [9] Faye, David., Nachouki, Gilles., Valduriez, Patrick ., (2006). SenPeer : Un système Pair-à-Pair de médiation de données. In: ARIMA Journal .

- [10] Rodríguez-Gianolli, Patricia., Garzetti, Maddalena., Jiang, Lei., Kementsietsidis, Anastasios., Kiringa, Iluju., Masud, Mehedi., Miller, Renée J., Mylopoulos, John (2005). Data Sharing in the Hyperion Peer Database System. *In: VLDB Conference, Trondheim, Norway.*
- [11] Haase P., Broekstra J., Ehrig M., Menken M., Mika P., Plechawski M., Pyszlak P., Schnizler B., Siebes R., Staab S., Tempich C. (2004). Bibster - A Semantics-Based Bibliographic Peer-to-Peer System". *In: The International Semantic Web Conference (ISWC), Hiroshima, Japan.*
- [12] Alon Y. Halevy, Zachary G. Ives, Peter Mork, Igor Tatarinov, Piazza: Data Management Infrastructure for Semantic Web Applications. *In: WWW, Budapest, Hungary (2003)*
- [13] Deniz Çokuslu, Abdelkader Hameurlain, Kayhan Erciyes, Grid Resource Discovery Based on Web Services. *In: ICITST, London, UK (2009)*
- [14] Heine, Felix., Hovestadt, Matthias., Kao, Odej (2004). Towards Ontology-Driven P2P Grid Resource Discovery. *In: IEEE/ACM International Workshop on Grid Computing (GRID), Pittsburgh, Pennsylvania.*
- [15] Jonquet, Clement., Musen, Mark., A, Shah, Nigam (2008). A System for Ontology-Based Annotation of Biomedical Data. *In: DILS, Evry, France.*
- [16] Kaur, D., Sengupta, J. (2007). Resource discovery in web services based grids. *In: World Academy Of Science, Engineering And Technology.*
- [17] Kementsietsidis, A., Arenas, M., Miller., R.(2003). Managing Data in Peer-to-Peer Systems: Semantics and Algorithmic Issues. *In: SIGMOD, San Diego, California.*
- [18] Moltó, G., Hernández, V., Alonso, J. M.(2008). A service oriented WSRF-based architecture for metascheduling on computational grids, *In: The International Journal of Grid Computing and eScience: Future Generation Computing Systems.*
- [19] Mokadem, Riad., Hameurlain, Abdelkade., Tjoa, A Min(2010). Resource Discovery Service while Minimizing Maintenance Overhead in Hierarchical DHT Systems, *In: International Conference on Information Integration and Web-based Applications & Services (iiWAS), Paris, France.*
- [20] Navas, I., Sanz, I., Aldana, J.F., Berlanga, R (2005). Automatic Generation of Semantic Fields for Resource Discovery in the Semantic Web. *In: DEXA, Bilbao, Spain.*
- [21] Ng, Wee Siong Ooi, Beng Chin Tan, Kian-Lee., Zhou, Aoying (2005). PeerDB: A P2P-based System for Distributed Data Sharing. *In: International Conference of Data Engineering, Bangalore, India.*
- [22] Pirrò, Giuseppe., Ruffolo, Massimo., Talia, Domenico (2008). An Algorithm for Discovering Ontology Mappings in P2P Systems. *In: KES, Zagreb, Croatia.*
- [23] Pirrò, Giuseppe., Talia, Domenico., Trunfio, Paolo. ERGOT: Combining DHTs and SONS for Semantic-Based Service Discovery on the Grid. Technical Report, CoreGRID.
- [24] Ramos, T. G. , Magalhaes, A. C. (2006). An extensible resource discovery mechanism for grid computing environments, *In: CCGRID: IEEE International Symposium on Cluster Computing and the Grid, IEEE Computer Society, Singapore.*
- [25] Sartiani, Carlo., Manghi, Paolo., Ghelli, Conforti., Giorgio Giovanni (2004). XPeer: A Self-organizing XML P2P Database System. *In: EDBT Workshop on P2P and Databases, Heraklion, Crete, Greece.*
- [26] Stoica, Ion., Morris, Robert., Karger, David., Kaashoek, M. Frans., Balakrishnan, Hari (2001). Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications, *In: SIGCOMM, San Diego, California, USA.*
- [27] Yu, J., Venugopal, S., Buyya, R. (2003). Grid market directory: A web services based grid service publication directory. Technical report, Grid Computing and Distributed Systems (GRIDS) (2003)
- [28] Zhu, Cheng., Liu, Zhong., Zhang, Weiming., Xiao, Weidong., Huang, Jincai(2004). An Efficient Decentralized Grid Service Discovery Approach based on Service Ontology. *In: The IEEE/WIC/ACM International Conference on Web Intelligence (WI), Beijing, China.*