SDDS Based Hierarchical DHT Systems for an Efficient Resource Discovery in Data Grid Systems

Riad Mokadem(✉), Franck Morvan, and Abdelkader Hameurlain

Institut de Recherche en Informatique de Toulouse (IRIT),
Paul Sabatier University, 118 Route de Narbonne, 31062 Toulouse, France
{mokadem, morvan, hameur}@irit.fr

Abstract. Most of the existing hierarchical Distributed Hash Table (DHT) systems, used for a resource discovery, generate considerable maintenance overhead which affects the routing efficiency in large scale systems. In this paper, we propose a Scalable Distributed Data Structures (SDDS) based Hierarchical DHT (SDDS- HDHT) solution for an efficient data source discovery in data Grid systems. Our solution deals with a reduced number of gateway peers running a DHT protocol. Each of them serves also as a proxy for second level peers in a single Virtual Organization (VO), structured as a SDDS. The performance evaluation of the proposed method proved the discovery cost reduction especially for intra-VO resource discovery queries. It also proved significant system maintenance save especially when peers frequently join/ leave the system.

Keywords: Resource discovery · Data grid · Peer to peer systems · Distributed hash table · Scalable distributed data structure

1 Introduction

A resource discovery constitutes an important step on a query evaluation in unstable and large scale environments [28], e.g., data Grid systems. It consists to discover resources (e.g., computers, data) that are needed to perform distributed applications [27]. Throughout this paper, we focus only on data source discovery in data Grid systems.

Classical resource discovery approaches in Grid systems are either centralized or hierarchical and were proved inefficient as the scale of Grid systems rapidly increases. In fact, excessive access to a centralized peer generates a bottleneck and its failure paralyzes the entire system and the using of web services, inspired from hierarchical models, has been explored in several research works as [42]. Although the advantage of being Open Grid Service Architecture (OGSA) [5] compliant, i.e., each resource is represented as a web service, this strategy is not adapted for grid environments since the dynamicity property in such environments [11]. Also, the using of the agents based hierarchical model gives partial results despite its capabilities to address scalability [34]. Several research works have adopted the Peer-to-Peer solutions to deal with resource discovery in Grid systems [24, 36]. P2P routing algorithms have been classified as structured or unstructured [37]. Although the good fault tolerance properties in

E. Simperl et al. (Ed.): ESWC 2012 Satellite Events, LNCS 7540, pp. 327–342, 2015.
DOI: 10.1007/978-3-662-46641-4_25

P2P unstructured systems (e.g., KaZaa [15]), the flooding –used in each search- is not scalable since it generates large volume of unnecessary traffic in the network. Structured Peer-to-Peer systems as DHT are self-organizing distributed systems designed to support efficient and scalable lookups in spite of the dynamic properties in such systems. Classical flat DHT systems organize peers, having the same responsibility, into one overlay network with a lookup performance of O(log(N)), for a system with N peers. However, the using of a flat DHT do not consider neither the autonomy of virtual organizations and their conflicting interests nor the locality principle, a crucial consideration in Grids [12]. Moreover, typical structured P2P systems as Chord [35] and Pastry [32] suffer not only from temporary unavailability of some of its components but also from churn effect. It occurs in the case of the continuous leaving and entering of peers into the system.

Recent research works as [27] proved that hierarchical overlays have the advantages of faster lookup times, less messages exchanged between peers, and scalability. They are valuable for small and medium sized Grids, while the super peer model is more effective in very large Grids [32]. In this context, several research works [6, 7, 14, 22, 23, 25, 33] and [43] proved that hierarchical DHT systems based on the super peer model [41] can be advantageous for complex systems. Hierarchical DHT solutions employ a multi level overlay network where peers are grouped according to a common property such as resource type or locality for a lookup service used in discovery [6]. In this context, a Grid can be viewed as a network composed of several, proprietary Grids, virtual organizations (VO) [23] where every VO is dedicated to an application domain (e.g., biology, pathology). Within a group, one or more peers are selected as super peers to act as gateways to peers in the other groups. Furthermore, most existing hierarchical DHT solutions neglect the churn effect and deal only with the improving performance of the overlay network routing. They mainly generate significant additional overhead to large scale systems [39]. Zöls et al. [43] gives a cost-based analysis of hierarchical DHT design. Performances depend on the ratio between super peers and the total number of peers. In fact, super peers are put more under stress especially if the leaf peers number increase. Several solutions have been proposed to reduce maintenance costs [6, 10, 16, 17, 21, 31] and [44]. Despite a good strategy to manage a churn in [34] through a lazy update of the network access points, inter-organizations lookups were expensive because of the complex addressing system. The SG-1 algorithm, proposed in [21], is based on the information exchange between super peers through a gossip protocol [1]. It aims to find the optimal number of super peers in order to reduce maintenance costs. However, most of these solutions add significant load at some peers which generates an additional overhead to large scale systems.

This paper deals with both improving lookup costs and managing churn while minimizing the overhead added to the system. We propose a Scalable Distributed Data Structure (SDDS) based Hierarchical DHT (SDDS- HDHT) solution for an efficient resource discovery in data Grids. It combines SDDS [20] and DHT routing schemes. Our solution employs a two level overlay network which deals with two classes of peers: super peers called also gateway peers (GP) and second level peers (SLP). Gateway peers establish a structured DHT based overlay. Only one peer, per VO, is considered as a gateway. Then, each of them serves as a proxy for second level peers in a single VO, structured as a SDDS. SDDS were among the first research works dealing

with structured P2P systems. [40] noted numerous similarities between Chord and the best known SDDS scheme: LH* (Linear Hashing) [20]. Both implement key search and have no centralized components. Resource discovery queries, in our system, are classified into intra-VO and inter-VO queries. The intra-VO discovery consists to apply the principle of locality by favoring the discovery in a local VO through the efficient LH* routing system. Key based queries in LH*, in its LH*$_{RS}^{P2P}$ versus, need at most two hops to find the target when the key search in a DHT needs $\log_B(N)$ hops [40], when N is the number of peers in the system and B typically equal to 4. In fact, super peers are not concerned by intra-VO queries unlike previous solutions as [43] which put super peers more under stress. Regarding Inter-VO queries, they are first routed to the reduced DHT overlay which permits to locate the gateway peer affected to the VO containing the resource to discover. Then, another LH*$_{RS}^{P2P}$ lookup is done in order to discover metadata describing this resource. The proposed solution takes also into account the continuous leaving and joining of peers into the system (dynamicity properties of Grid environments). Only the arrival of a new VO requires the DHT maintenance. The connection/ disconnection of gateways do not require excessive messages exchanged between peers in order to maintain the system. This is done through a lazy maintenance which avoids high maintenance costs [16]. A simulation analysis evaluates performances of the proposed solution through comparison with previous hierarchical DHT solution performances. It shows the reduction of lookups costs especially for intra-VO queries. It also provides a significantly maintenance costs reduction, especially when peers frequently join/leave the system. The rest of the paper is structured as follows. Section 2 recalls hierarchical DHT and SDDS principles. Section 3 presents our resource discovery solution through the proposed protocol. It also describes the maintenance process. The simulation analysis study section shows the benefit of our proposition. The final section contains concluding remarks and future work.

2 Preliminaries

2.1 LH*$_{RS}^{P2P}$ Scalable Distributed Data Structure

Many variant of SDDS have been proposed [20, 40]. In this paper, we deal with LH*$_{RS}^{P2P}$ scheme which improves later LH* variants (LH*$_{RS}$, LH*$_g$…) and assume that the reader is familiar with a linear hashing algorithm LH* as presented in [19]. Scalable Distributed Data Structures (SDDS), designed for P2P applications, are a class of data structures for distributed systems that allow data access by key in constant time [40]. Each peer in SDDS stores records in a bucket which splits when the file grows. Every LH* peer is both client and, potentially, data or parity server which interacts with application using the key based record search, insert, update or delete query or a scan query performing non key operations. In the resource discovery process, we deal with the key search operations (e.g., a database relation) in order to search metadata describing the data source.

Basic linear hash functions are $H_i(C) = C \bmod 2^i$ with i the file level which determines the linear hash function H_i to be applied. Each record in LH * is identified by its key. The key *C* determines the record location according to the linear hashing

Algorithm [20]. The file starts with one data bucket and one parity bucket. It scales up through data bucket splits, as the data buckets get overloaded. It can be occurred when a peer splits its data bucket. In old SDDS scheme, one peer acted as a coordinator peer. It was viewed as the single peer knowing the correct state of the file or relation. However, [40] ameliorates this scheme. Split coordinator does not constitute a centralized peer for the SDDS scheme. It intervenes only to find a new data server when a split occurs and never in the query evaluation process. Any other peer uses its local view 'image', which may be not adjusted, to find the location of a record given in the key based query. Suppose that it send its key based query by using its image which can lead to an incorrect bucket. An outdated image could result. The peer server applies another algorithm $LH^*_{RS}{}^{P2P}$ [40]. It first verifies whether its own address is the correct one by checking its guessed bucket level in the received query against its actual level (the level of LH function used to split or create the bucket). If needed, the server forwards this query. The query always reaches the correct bucket in this step i.e., if forwarding occurs, the new address has to be the correct one. It sends an Image Adjustment Message (IAM) informing the initial sender that the address was incorrect and the sender adjusts its image reusing the LH* image adjustment algorithm described in [20]. It was not the case with LH* which may need an additional hop. Proof of this property is in [40]. So, the most important property here is that the maximal number of forwarding messages for key-based addressing is one. Moreover, [40] proved the efficient recovery of more than one peer without any duplication. $LH^*_{RS}{}^{P2P}$ scheme allows the correction possibility through parity calculus. It is done using Reed Solomon (RS) codes [20]. Another advantage of using SDDS is the possibility to support range queries very well and the less vulnerability in the presence of high churn [40].

2.2 Hierarchical Distributed Hash Tables

Structured P2P systems (e.g., Chord [35], Pastry [32], CAN [30] and Tapestry [45]) offer deterministic query search results. In this paper, we have focused on a Pastry DHT system [32] which requires Log_B (N) hops, where N is the total number of peers in the system and B typically equal to 4. Pastry system is also adapted for several different applications including a global persistent storage utility as PAST [2] and a scalable publish/subscribe system called SCRIBE [29]. Pastry peer permits also to easily locate both the right ad left neighbors in the DHT. These reasons motivate us to choose the Pastry routing system. But, our method can be applied to other DHT systems. Flat DHT solutions do not take into account the autonomy of organizations. Hierarchical DHT systems partition its peers into a multi level overlay network. Because a peer joins a smaller overlay network than in flat overlay, it maintains and corrects a smaller number of routing states than in flat structure. In such systems, one or more peers are often designated as super peers. They act as gateways to other peers organized in groups in second level overlay networks.

Throughout this section, we interest to two previous hierarchical DHT solutions which we consider comparable to our solution. In Fig. 1-left, super peers establish a structured DHT overlay network when second level peers (called leaf peers) maintain only connection to their super peers. This corresponds to the Super Peer HDHT

(SP-HDHT) solution [43]. However, [22] proved that this strategy can maintain super peers more under stress by maintaining pointers between super peers and their leaf peers. Furthermore, a super peer stores metadata of all leaf peers which it is responsible and acts as a centralized resource for them. Then, performances depend on the ratio between super peer's number and the total number of peers in the system. Multi-Gateway Hierarchical DHT (MG-HDHT) solution [23] is another example of 2-levels hierarchy system having multiple gateways per VO (Fig. 1- right). The system forms a tree of rings (DHTs in this example). Typically, the tree consists of two layers, namely a global ring as the root and organizational rings at the lower level. A group identifier (*gid*) and a unique peer identifier (*pid*) are assigned to each peer. Groups are organized in the top level as DHT overlay network. Within each group, peers are organized as a second level overlay. This solution provides administrative control and autonomy of the participating organizations. Unlike efficient intra-organization lookups, inter-organization lookups are expensive since the high maintenance cost of the several gateway peers. Hence, there is a trade-off between minimizing total network costs and minimizing the added overhead to the system.

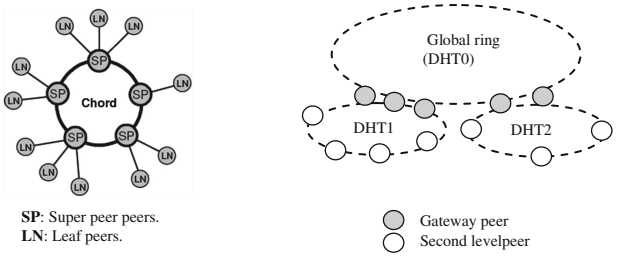

Fig. 1. SP-HDHT (left) and MG-HDHT (right) Solutions.

3 Resource Discovery Through SDDS Based Hierachical DHT Systems

A data source discovery is an important step in the query evaluation on large scale environments. It consists in search metadata describing data sources (e.g., the profile of a relation *Doctor* which is associated to a domain concept). In large scale environments, this task is complex since data sources are highly heterogeneous and constantly evolving due to data source autonomy. Furthermore, the dynamicity of peers is a major problem since the continuous joining/ leaving of peers generates prohibitive maintenance costs. In this section, we propose a SDDS based hierarchical DHT (SDDS-HDHT) solution for a resource discovery in data Grids. It aims to reduce both lookup and maintenance costs while minimizing overhead added to the system.

Resource Discovery through our solution deals with two different classes of peers: gateways (GP) and second-level peers (SLP). A Grid can be viewed as a network composed of several, proprietary Grids, virtual organizations (VO) [13] as shown in Fig. 2. Every VO is dedicated to an application domain (e.g., biology, pathology) [16]. It permits to take into account the locality principle of each VO [12]. Within a VO, one

peer is selected as a gateway peer (GP). It acts as a super peer or a proxy for other peers, called second level peers (SLPs). GPs communicate with each other through a DHT overlay network. Each of them knows, through the $LH^*_{RS}{}^{P2P}$ routing system, how to interact with all SLPs belonging to the same VO. In this context, [6] proved that a DHT lookup algorithm required only minor adaptations to deal with groups instead of individual peers. In order to make a resource in VO_i visible through the DHT, hash join *H* is applied to this resource, when it joins the system, to generate a group identifier *gid*. Then, another hash function *h* is applied to this resource in order to generate a peer identifier *pid*. This permits to associate each resource to its VO [22]. We may assume that GPs are relatively more stable than SLPs. In contrast, GPs establish a structured DHT based overlay when each VO -regrouping SLPs- is structured as a SDDS. We consider here the peers as homogenous. Recall also that we have not interesting on the assignment of a joining SLP to an appropriate GP, i.e., loads balancing. We defer these issues to future work.

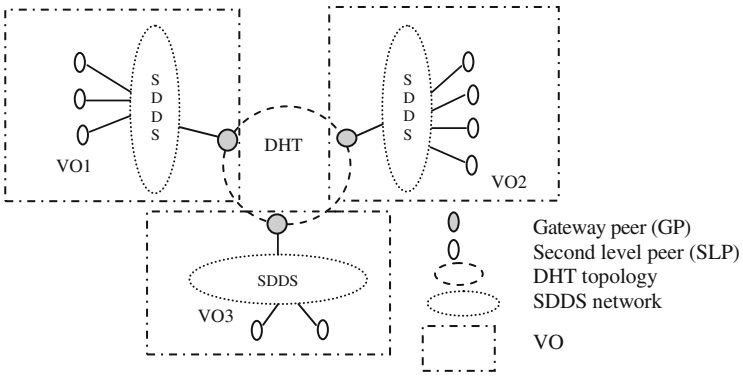

Fig. 2. SDDS based Hierarchical DHT Architecture.

3.1 Resource Discovery Protocol

After describing how VOs are connected, we present the resource discovery protocol used in the proposed SDDS-HDHT solution. Suppose that a SLP $p_i \in VO_i$ wants to discover a resource *Res* through a resource discovery query *Q*. Let the peer p_J be the peer responsible for *Res*. Let Gp_i the GP responsible for VO_i, Gp_i_list the list of its neighbors in the top level DHT (e.g., the left and the right neighbors) and *Response* the metadata describing the data source *Res*. Thus, a lookup request for *Res* implies locating the peer responsible for *Res*. In our system, data source discovery queries can be classified into two types:

(i) Data source discovery within a single VO, i.e., peers p_i and p_j belong to the same VO. Then, the query Q corresponds to an intra-VO resource discovery query.
(ii) Data source discovery between VOs, i.e., peers p_i and p_j are in different VOs. Then, the query Q corresponds to an inter-VO resource discovery query.

Intra-VO resource discovery queries are evaluated through a classical $LH^*_{RS}{}^{P2P}$ routing system which is completely transparent to the top level DHT. Generally, users

often access data in their application domain, i.e., in their VO. In consequence, it is important to search metadata source first in the local VO_i before searching in other VOs. This solution favors principle of locality [12]. Recall that finding a peer responsible of metadata of the searched resource requires only two messages. Finally, the peer p_J sends metadata describing Res (if founded) to p_i, the peer initiator of Q.

When the researched resource *Res* is not available in the local VO_i, resource discovery is required in other VOs. This corresponds to an inter-VO resource discovery process. Before introducing the resource discovery process, described by the algorithm shown in Fig. 3, let us recall that we have defined an interval of time noted RTT (Round- Trip Time) as in [27]. Hence, if a peer does not respond after one RTT, it is considered to be disconnected. The manner in which the RTT values are chosen during lookups can greatly affects performances under churn. [31] has demonstrated that a RTT is a significant component of lookup latency under churn. In fact, requests in peer to peer systems under a churn are frequently sent to a peer that has left the system. In our solution, a RTT is mainly useful to maximize time to discover resources when a failure occurred in a GP. In this case, p_i do not expect indefinitely. When RTT is exceeded, it considers that Gp_i is failed and consults the gateway neighbours list Gp_i_list received in the connection step. Then, p_i sends its query to one of the peers founded in Gp_i_list. Let us recall that in the connection step of any gateway peer Gp_i, this latter send its list neighbors Gp_i_list to p_0 in its VO. Then, p_0 forwards Gp_i_list to all other SLPs. It is done just on the connection step.

```
  Discover (p_i, Res, VO_i)
{ if (LH*_RS^P2P_Lookup(Res,VO_i)==1)then Send Response to p_i;
                                    Else forward Q to Gp_i
  if Gp_i do not respond      then{ consult Gp_i_list;
                             Forward Q to neighbours of Gp_i;}
  if  DHT_route(Res)==Gp_J    then
       if (LH*_RS^P2P_Lookup(Res, VO_J)==1)
                  then send Response to p_i.
                  else Response = {}is sent to  p_i.
    else Response = {}is sent to p_i.
  }
```

Fig. 3. Data Source Discovery Algorithm.

Regarding the inter-VO lookup process in the SDDS-HDHT solution, it requires four steps. In the first step, a peer p_i routed the query to the gateway Gp_i. If a Gp_i failure is detected (RTT is elapsed), it requests one neighbor of Gp_i, already received. Once the query reaches a gateway peer Gp_i, a hash function H is applied to *Res* in order to discover the GP responsible for the VO that containing *Res*. The query arrives at some Gp_J. This is valid whenever a resource, matching the criteria specified in the query, is found in some VO_J. the third step consists to apply the $LH^*_{RS}{}^{P2P}$ routing system in the found VO_J. Gp_J routes the query to the peer $p_J \in VO_J$ that is responsible for Res. Finally, metadata of *Res* are sent to Gp_j which forward it to p_i via the reversing path.

3.2 System Maintenance

The connection/ disconnection of a peer requires the update of the system. Furthermore, the continuous leaving and entering of peers into the system is very common in Grid systems (dynamicity properties). Peer departures can be divided into friendly leaves and peer failures. Friendly leaves enable a peer to notify its overlay neighbors to restructure the topology accordingly. Peer failures possibility seriously damages the structure of the overlay with data loss consequences. Remedying this failure generates additional maintenance cost. In structured peer-to-peer systems, such as Pastry [32] used in our system, the connection/ disconnection of one peer generates $2B*Log_B(N_T)$ messages [32]. Furthermore, the maintenance can concern the connection/ disconnection of one or more peers. Throughout this section, we explore the different factors that affect the behavior of hierarchical DHT under churn (super peer failure addressing, timeouts during lookups and proximity neighbor selection) [31]. Then, we discuss the connection/ disconnection of both GPs and SLPs.

Second Level Peer (SLPs) Connection/ Disconnection. The connection/ disconnection of a SLP p_i do not affect lookups in other peers except the possible split of a bucket if this latter gets overloaded. Let us discuss the only one required maintenance. When p_i joins some VO_i, it asks its neighbor about Gp_i and the Gp_i_list. In consequence, only two messages are required. This process avoid that several new arrival peers asked simultaneously the same GP which can constitute a bottleneck as in SP-HDHT solution. In other terms, when a new SLP arrives, it searches its GP (only one) and neighbors of this one. This process permits also to reduce messages comparing to the complex process in the MG- HDHT solution in which the new SLP should retrieve all GPs.

Gateway Peer (GP) Connection/ Disconnection. For this aim, we propose a protocol in order to reduce the overhead added to the system. When a GP connection/ disconnection occur, we distinguish two types of maintenance: (i) maintenance of the DHT and (ii) maintenance of the neighbour's lists. We will not discuss the first maintenance since it corresponds to a classical DHT maintenance [35]. In the other hand, without any maintenance protocol, a disconnection or a failure of a GP paralyzes access to all SLPs for which the Gp_i is responsible. Addressing this failure generates additional maintenance cost. Before describing the maintenance process, let us analyze the connection steps of Gp_i to VO_i. The gateway peer Gp_i send its list neighbors Gp_i*_list* (the left and right neighbor) to the nearest *SLP* p_0 in VO_i. Then, p_0 contacts peers in Gp_i*_list* to inform them about its existence (in order to have an entry to VO_i in the case of Gp_i failure). After that, p_0 send this list to all SLPs in VO_i via a multicast message. Other SLPs do not report their existence to neighbors of Gp_i.

Recall also that this process is done just once at the initial connection of Gp_i and only p_0 periodically executes a *Ping/Pong* algorithm with Gp_i. It sends a *Ping* message to Gp_i and this one answers with a *Pong* message in order to detect any failure in Gp_i. Let us discuss the case of a GP failure/ update. When a peer Gp_i is replaced by another, the process of maintenance (after the DHT maintenance) is:

(i) The new gateway Gp_{New} contacts the nearest (only one) SLP p_0 and gives him its neighbor's list Gp_{New}*_list*.

(ii) Peer p_0 inform peers in Gp_{New}_list about its existence. But, it does not inform other SLPs about Gp_{New}_list.

Remark that the peer p_0 do not send description of the new gateway peer Gp_{New} and its updated Gp_{New}_list to other SLPs at this moment. A lazy update is adopted. When Gp_i does not respond after a RTT period, a SLP consults its old Gp_i_list to reach other VOs. Thus, it rejoins the overlay network in spite of a GP_i failure. The update of this list is done during the reception of the resource discovery result as in [16]. Also, a failure of p_0 does not paralyze the system since the new GP always contacts its nearest SLP. The entry to the VO can also be done through peer p_0 since this one reported its existence in the connection step. This process allow a robust resource discovery process although the presence of dynamicity of peers. This is not the case in MG-HDHT solution when failures of all GPs in some VO paralyze the input/ output to/ from this VO. Recall also that one of the limitations that our solution suffers from: the failure of both GP_i and its neighbors in Gp_i_list. A solution consists in enriching the neighbors list of gateway peers with neighbors of neighbors.

4 Performance Analysis

In order to validate the proposed SDDS-HDHT solution, we evaluate its performances and compare them to those of three other resource discovery methods: (i) a flat DHT solution in order to measure the benefits of hierarchical system, (ii) the SP-HDHT solution [43] in which GPs establish a DHT overlay network when each leaf peers maintains a connection to its GP and (iii) the MG-HDHT solution in which several gateways are maintained between hierarchical levels [23].

We based on a virtual network as 10000 peers to prove the efficiency of our solution in large scale networks. We deal with a simulated environment since it is difficult to experiment thousands of peers organized as virtual organization in a real existing platform as Grid'5000 [9]. We based our experiments on a platform having four features: (i) emulation of peers, (ii) simulation of homogenous bandwidth networks and local network 100 Mb/s, (iii) using of FreePastry [4], an implementation of the Pastry DHT protocol and (iv) ${LH^*}_{RS}^{P2P}$ SDDS prototype implemented by Litwin's team in Dauphine University [18]. Variables used bellow are defined as follows: N_T is the number of peers in the system, N_G the number of GPs, N_{SL} the number of SLPs and α the gateway ratio. It is the ratio between GPs and the total number of peers ($N_G = \alpha . N_T$). Key of the searched data source corresponds to a database relation name in our experiments. For the detection of failed peers, we set a TTL to 1 s. Throughout this section, we deal with four classes of experiments: (i) Lookup performances experiments in which we are interested in elapsed times which includes the query processing and communication costs, (ii) maintenance overhead experiments in which we simulate a join/leave peers scenario and interest to the required update messages, (iii) experiments to find the optimal ratio between GPs and SLPs in order to evaluate the impact of the gateway ratio on performances (we have varied N_G but the total number of peers always stay constant) and (iv) experiments to measure the impact of the gateway session's length on the system maintenance [3].

4.1 Lookup Performances Analysis

The first experiment simulates a flat DHT solution. This equates to a configuration with $N_T/N_G = 1$ in the SDDS-HDHT solution. When we analyze the hops number required to discover one resource in both solutions, our results are always better when it concerns an intra-VO resource discovery query. In fact, ${LH^*}_{RS}^{P2P}$ lookup requires a maximum of two (2) messages when this number is always $log_B(N_T)$ in flat DHT solutions. For inter-Vo queries, we have showed in [26] that the theoretically worse case corresponds to $O(log_B(N_G)) + 4$ hops with SDDS-HDHT scheme. By a simple calculation, we deduce that flat DHT performances are better when our DHT overlay is composed by more than 1000 GPs. In other terms, from 10 SLPs per VO (i.e., $\alpha < 1$ %), our results are better. This is due to the fact that adding new SLPs do not influences ${LH^*}_{RS}^{P2P}$ lookup performances. However, these results correspond to theoretical numbers of hops for only one resource discovery query. In the case of simultaneous resource discovery messages, the results should take into account that all messages are forward to the same GP (in one VO). This generates some congestion in this peer. To confirm this, we have experimented systems with (i) 2000 gateways (5 SLPs/ VO, $\alpha = 20$ %) and (ii) 500 gateways (20 SLPs/ VO, $\alpha = -5$ %). We also interest to the number of simultaneous resource discovery queries. It is useful since it shows if the SDDS-HDHT solution is also scalable in the presence of high number of messages.

Figure 4 shows elapsed response times for resource discovery queries (intra and inter-VO queries). It confirms that SDDS-HDHT performances are always better when queries constitute intra-VO resource discovery queries. Elapsed response times are 50 % better than flat DHT solution. This is due to the reason mentioned above. Let analyze performances of inter-VO queries. When we experiment with $\alpha = 20$ %, performances are almost similar for a reduced simultaneous discovery queries. But, elapsed responses time increase from 20 queries/s. It is due to the fact that all queries transit by the same GP in each VO. However, a great leaf peers number ($\alpha = 5$ %) improves significantly our performances which are better. The save is close 10 % compared to the flat DHT solution in spite of the simultaneous messages. It provides from the gain in the DHT lookup. In fact, the probability to find the searched resource in a local VO is greater.

We have also compared SDDS-HDHT performances to both SP-HDHT and MG-HDHT performances. [43] proved that best performances are obtained with small number of gateways. We simulate a network with 100 VOs (with 100 level peers/ VO). Figure 4 shows that the SP-HDHT solution is slightly better for intra-VO queries when less simultaneous messages are used. From 70 messages/s, our solution is 10 % better than SP-HDHT solution. We explain this by the fact that intra-VO lookups are done without any GP intervention when a bottleneck is generated in each GP in the SP-HDHT solution. This is the reasons why the simultaneous messages influenced significantly the SP-HDHT results. We remark that the average response time is almost constant when we have several simultaneous messages in both SDDS-HDHT and MG-HDHT solution. We conclude that the save can be better if we experiment with great number of simultaneous discovery queries. Note that these experiments do not include the more costly connection step.

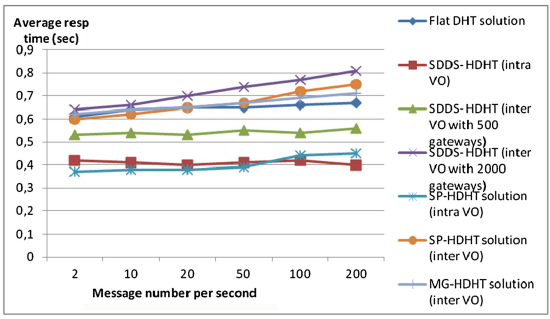

Fig. 4. SDDS-HDHT Performances vs. Flat DHT, SP-HDHT, MG-HDHT Performances.

For inter-V0 queries, simultaneous resource discovery queries influences performances of both SP-HDHT and SDDS-HDHT solutions. Bottleneck is generated since all queries transit by the same GP which increases response times. SP-HDHT response times are slightly better when we have less than 70 messages per second. From this value, results are almost close for the two solutions with slight advantage to SDDS-HDHT solution since intra-VO queries always precede inter-VO queries. We conclude that in inter-VO queries, we have dependence between performances and simultaneous queries for these two solutions. The same impact is observed with a reduced α. Regarding the MG-HDHT performances, they are better (rate of 5 %) especially for high simultaneous messages seen as inter-VO queries are propagated through several GPs.

4.2 Maintenance Analysis

We measure the impact of the peers connection/ disconnection on the system. We interest to the total messages number required when a peer joins/leaves the network. We tabulate churn in an event-based simulator which processes transitions in state (*down*, *available*, and *in use*) for each peer as in [8]. We simulate a churn phase in which several peers join and leave the system but the total number of peers N_T stays appreciatively constant. The maintenance costs are measured by the number of messages generated to maintain the system when peers join/leave the system.

Let a system with a peers distribution as $\{N_G$ = 100 and 100 peers/ VO$\}$. This configuration corresponds to average results in inter-VO discovery queries performances. In these experiments, when a number of new connections/ disconnections exceed 20 peers, 10 % of them concern GPs. In this context, Flat DHT solution generates the greater number of messages. When only one SLP arrive into the system, 62 messages are required to update the Flat DHT system while this number is only 57 messages with our solution. The save is more important with the connection of 90 SLPs and 10 GPs. In this case, the messages number ratio is 4.5 (5200 messages for the DHT solution and 1600 messages with the SDDS-HDHT solution).

We compare these results to the SP-HDHT performances. Figure 5- left shows the impact of peers connection/ disconnection on the total messages number required to update the system. The numbers of update messages are closes when we have only SLPs connections/disconnections. It corresponds to the case when less than 10 peers

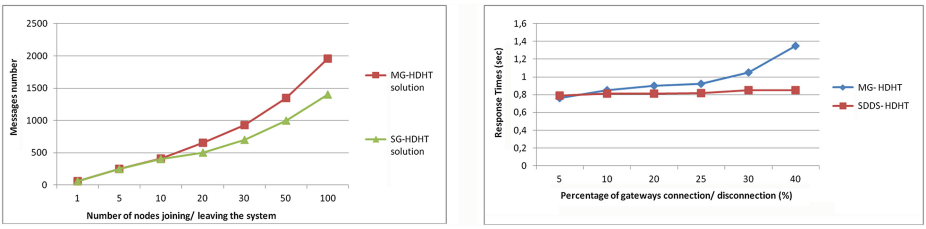

Fig. 5. Impact of: the peer connection/disconnection on the system maintenance (left) and the percentage of the GPs connection/disconnection on the response time (right).

join the system. In fact, all new peers must contact their super peer in SP-HDHT solution. Increasing the number of connection/ disconection of leaf peers can generates a bottleneck in this solution when SDDS-HDHT solution offers a significant maintenance cost gain including when the update occurs in GPs. As the number of GPs connection increase as the gain is important since the required update messages is less with our solution. The save is 59 % for the connection of 90 SLPs and 10 GPs. Certainly, updating the DHT concern both solutions. But, in the SP-HDHT solution experiments the new gateway establishes connections with all its leaf peers. It is also the case in the MG-HDHT solution. The fact that new SLPs in MG-HDHT must contact several gateways generates additional messages. It is not the case in our SDDS-HDHT solution. A new SLP contacts only its neighbour and the connection of a new GP generates only two additional messages with SLPs in the same VO.

We also experiment the impact of the percentage of the GPs arrival/ departure on the total response time as shown in Fig. 5- right. The worst case corresponds to a discovery process under a high churn. When only 5 % of gateways are replaced by other gateways, MG-HDHT solution has slightly better results than SDDS-HDHT performances. However, when this percentage increases, SDDS-HDHT performances remain stable since SLPs used the gateway neighbor's list to reach other GPs in the DHT. In MG-HDHT solution, they used the other not failed gateways in the same VO pending the update of the new GPs. From 25 % of GPs connection/ disconnection in the system, MG-HDHT curve increase significantly. Recall that we have deliberately ensured that not all GPs in the same VO are failed in MG-HDHT solution. Otherwise, a SLP in some VO_i will be not able to contact any gateway of other VO_j ($i \neq j$) until. It is not the case in our solution in which SLPs can use the Gp_i_list. But, recognize that if all peers in the Gp_i_list failed, consequences are also the same as above.

4.3 Impact of the Gateway Ratio on Performances

Through these experiments, our goal is to determine optimal configurations on the three compared solutions. In first experiments, without any peer arrival/departure to the system (Fig. 6- left), a centralized overlay network with only one super peer in SP-HDHT solution generates the lowest traffic costs. The reason is that only lookup and *Ping/ Pong* messages are exchanged between the super peer and its leaf peers.

Same performances are obtained with the configuration (α = 100 %) in the three compared solution since all peers participate in a flat DHT overlay. If the number of GPs increases (N_G > 1), we notice increased lookup costs for the three compared solution as shown in Fig. 6- left. This cost is most important in SDDS-HDHT and SP-HDHT solution, mostly caused by the bottleneck in the only one GP. Indeed, it is due to the fact that all queries transit by the same GP when the several GPs are less in stress on the MG-HDHT solution. This cost decrease from α = 20 % in the SP-HDHT and SDDS-HDHT solutions. It is from α = 10 % in the MG-HDHT solution. We conclude that MG-HDHT solution constitutes the better solution when we have not or very little departures/ arrivals of peers in the system. However, good performances are obtained from α = 10 % with our solution which is close to real grid systems with several VOs.

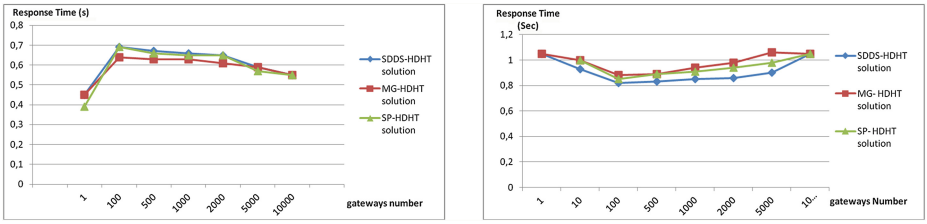

Fig. 6. Impact of the gateway number on performances: without maintenance costs (left) and including maintenance costs (right).

We also deal with experiments taking into account the arrival/ departure of peers to the system as shown in Fig. 6- right. We deal with the connection/ disconnection of 10 % of the GPs in the system and 10 % of SLPs in each VO. From α = 1 %, the maintenance cost of the MG- HDHT solution is always the most important since each GP inform all its SLPs in each arrival/ departure. It is also the case with the SP- HDHT solution with better results. This is not the case in the SDDS- HDHT solution which has the best results with α between 1 and 50 %. It is due to the fact that SLPs used a lazy update to update their neighbor's gateway list. For each value of α in this interval, the SDDS-HDHT solution generates the lowest total cost. It is valuable for the case when the major maintenance cost is generated by the departure/ arrival of SLPs but also for the case when the departure/ arrival of GPs constitute the major maintenance cost. We conclude that the best SDDS-HDHT performances are obtained with α ∈ [1 %, 20 %] which is close to real grid systems with several VOs.

5 Conclusion and Future Work

The proposed hierarchical DHT solution combines SDDS, in its $LH^*_{RS}{}^{P2P}$ variant, and DHT routing schemes for a data source discovery in data Grid systems. We aim to reduce lookup costs and manage churn while minimize the additional overhead to the system. We group all peers of a same domain in a same Virtual Organization (VO) in

order to favor the content/path locality principle. Then, only one peer per VO runs a DHT protocol and acts as a gateway peer for second level peers in its VO. Within a VO, structured as a SDDS, the resource discovery process is based on a classical LH* routing system. Hence, the first contribution is the improvement of data source discovery query complexity especially for intra-VO queries since these queries are transparent to the DHT lookup. Regarding inter-VO resource discovery process, we have proposed a new protocol which aims to reduce the exchanged messages between peer. Only the arrival of a new VO requires the DHT maintenance and second level peers update theirs gateway peer neighbours during the resource discovery process. The performance analysis shows the benefit of our proposition through comparisons of our performances to those of previous solutions. It shows the improvement of lookup query performances especially as regards the intra-VO resource discovery queries. This is especially valid when comparing our solution to the flat DHT and SP-HDHT solutions since intra-VO queries do not require the gateway peer intervention. We have also compared our solution with another previous hierarchical DHT solution based on the using of several gateway peers (MG-HDHT solution). While sacrificing response times for simultaneous queries per second, we have important maintenance costs save when several peers join/ leave the system. Hence, it seems more reasonable to have a dynamicity of peers in a large scale environment. This save is mainly due to the lazy maintenance adopted in our solution.

Our solution can be useful in large scale environments since it generates less traffic network. Further work includes more performance studies in more realistic large grid environments with a high number of peers. Also, we would like to study the effects of alternate routing table neighbours as in [38].

References

1. Artigas, Marc S., García, Pedro, Skarmeta, Antonio FGómez: DECA: a hierarchical framework for decentralized aggregation in DHTs. In: State, Radu, van der Meer, Sven, O'Sullivan, Declan, Pfeifer, Tom (eds.) DSOM 2006. LNCS, vol. 4269, pp. 246–257. Springer, Heidelberg (2006)
2. Druschel, P., Rowstron, A.: PAST: a large-scale, persistent peer-to-peer storage utility. In: HotOS VIII, Germany (2001)
3. Fei, T., Tao, S., Gao, L., Guerin, R.: How to select a good alternate path in large peer-to-peer systems? In: Proceedings of the international conference on IEEE INFOCOM (2006)
4. http://Freepastry.org/FreePastry/
5. Foster, I., (ed.), Berry, D., Djaoui, A., Grimshaw, A., Horn, B., Kishimoto, H., (ed.), Maciel, F., Savva, A., Siebenlist, F., Subramania, R., Treadwell, J., Von Reich, J.: The Open Grid Services Architecture, V 1.0. Global Grid Forum (2004)
6. Garcés-Erice, Luis, Biersack, Ernst W., Felber, Pascal, Ross, Keith W., Urvoy-Keller, Guillaume: Hierarchical peer-to-peer systems. In: Kosch, Harald, Böszörményi, László, Hellwagner, Hermann (eds.) Euro-Par 2003. LNCS, vol. 2790, pp. 1230–1239. Springer, Heidelberg (2003)
7. Ganesan, P., Gummadi, K., Garcia-Molina, H.: Canon in g major: designing DHTs with hierarchical structure. In: International Conference on Distributed Computing Systems 2004, pp 263–272 (2004)

8. Godfrey, P.B., Shenker, S., Stoica, I.: Minimizing churn in distributed systems. In: International Conference on SIGCOMM, pp 147–158, Italy (2006)
9. GRID'5000. www.grid5000.org
10. Gupta, I., Birman, K., Linga, P., Demers, A., Renesse, R.V.: Kelips: Building an Efficient and Stable P2P DHT through Increased Memory and Background Overhead. In: Frans Kaashoek, M., Stoica, I. (eds.) IPTPS 2003. LNCS, vol. 2735, pp. 160–169. Springer, Heidelberg (2003)
11. Hameurlain, Abdelkader: Evolution of query optimization methods: from centralized database systems to data grid systems. In: Bhowmick, Sourav S., Küng, Josef, Wagner, Roland (eds.) DEXA 2009. LNCS, vol. 5690, pp. 460–470. Springer, Heidelberg (2009)
12. Harvey, N., Jones, M., Saoiu, S., Theimer, M., Wolman, A.: Skipnet: a scalable overlay network with practical locality properties. In: Proceedings of USITIS, Seattle, USA (2003)
13. Iamnitchi, A., Foster, I.: A peer-to-peer approach to resource location in grid environments. In: Proceedings of HPDC 2002, Edinburgh, UK (2002)
14. Joung, Y., Wang, J.-C.: Chord2: a two-layer chord for reducing maintenance overhead via heterogeneity. Comput. Netw. **51**(3), 712–731 (2007)
15. Kazaa. http://www.kazaa.com/
16. Ketata, Imen, Mokadem, Riad, Morvan, Franck: Resource discovery considering semantic properties in data grid environments. In: Hameurlain, Abdelkader, Tjoa, A.Min (eds.) Globe 2011. LNCS, vol. 6864, pp. 61–72. Springer, Heidelberg (2011)
17. Ketata, I., Mokadem, R., Morvan, Franck: Biomedical resource discovery considering semantic proprieties in data grid environments. In: Joy, M., et al. (eds.) INTECH'11, vol. 165, pp. 12–64. Springer, Heidelberg (2011)
18. http://lamsade.dauphine.fr/∼litwin/default.html
19. Litwin, W.: Linear hashing: a new tool for file and table addressing. In: Stonebreaker, M. (ed.) VLDB 1980, 2nd edn. Morgan Kaufmann, San Fransisco (1995)
20. Litwin, W., Moussa, R., Schwarz, T.: LH*rs a highly available scalable distributed data structure. In: Jin, H., Rana, O.F., Pan, Y., Prasanna, V.K., et al. (eds.) Algorithms and Architectures for Parallel Processing, vol. 4494, pp. 188–197. Springer, Heidelberg (2005)
21. Montresor, A.: A robust protocol for building superpeer overlay topologies. In: IEEE International Conference on Peer-to-Peer Computing (P2P) (2004)
22. Martinez, I., Cuevas, R., Guerrero, C., Mauthe, A.: Routing performance in a hierarchical DHT-based overlay network. In: Euromicro International Conference PDP, pp. 508–515, Toulouse (2008)
23. Mislove, Alan, Druschel, Peter: Providing administrative control and autonomy in structured peer-to-peer overlays. In: Voelker, Geoffrey M., Shenker, Scott (eds.) IPTPS 2004. LNCS, vol. 3279, pp. 162–172. Springer, Heidelberg (2005)
24. Meshkova, E., et al.: A survey on Resource Discovery Mechanisms, Peer to Peer and Service Discovery Frameworks Computer Networks, pp. 2097–2128. Science Direct, Elsevier, New York (2008)
25. Mokadem, R., Hameurlain, A., Min Tjoa, A.: 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 (2010)
26. Mokadem, R., Hameurlain, A.: An efficient resource discovery while minimizing maintenance overhead in SDDS based hierarchical DHT systems. Int. J. Grid Distrib. Comput. (IJGDC) **4**(3), 1–24 (2011)
27. Mastroianni, C., Talia, D., Verta, O.: Evaluating resource discovery protocols for hierarchical and super-peer grid information systems. In: 19th Euromicro International Conference (PDP) (2007)

28. Pacitti, E., Valduriez, P., Mattosso, M.: Grid data management: open problems and news issues. Int. J. Grid Comput. **5**, 273–281 (2007). Springer
29. Rajiv, R., et al.: Peer to peer based resource discovery in global grids: a tutorial. In: IEEE Communication Surveys, vol. 10, No 2, 2 nd Quarter (2008)
30. Ratnasamy, et al.: A scalable content-adressable network. In: Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures and Protocols for Computer Communication, pp. 161–172 (2001)
31. Rhea, S., Geels, D., Roscoe, T., Kubiatowicz, J.: Handling churn in a DHT. In: Proceedings of the General Track: Usenix Annual Technical Conference, Boston, USA (2004)
32. Rowston, A., Druschel, P.: Pastry: scalable distributed object location and routing for large-scale peer-to-peer systems. In: Proceeding of the 18th IFIP/ACM International Conference on Distributed Systems Platforms, vol. 2218, pp. 329–350 (2001)
33. S´anchez-Artigas, M., Garc´ya, P., Pujol, J., Skarmeta, A.G.: Cyclone: a novel design schema for hierarchical DHTs. In: IEEE International Conference on Peer-to-Peer Computing (*P2P)* (2005)
34. Samad, M.E., Morvan, F., Hameurlain, A.: Resource discovery for query processing in data grids. In: Graham, J.H., et al. (eds.) 22nd International Conference on Parallel and Distributed Computing and Communication Systems, PDCCS 2009, 24–26 September 2009, Louisville, Kentucky, USA. ISCA (2009)
35. Stoica, I., Morris, R., Karger, D., Kaashoek, M.F., Balakrishma, H.: CHORD: a scalable Peer to Peer Lookup Service for Internet Application. In: SIGCOMM'O, San Diego, USA (2001)
36. Trunfio, P., Talia, D., Papadakid, H., Fragoupoulou, P., Mordachini, M., Penanen, M., Popov, P., Valssov, V., Haridi, S: Peer-to-peer resource discovery in grids: models and systems. In: Future Generation Computer Systems (2007)
37. Valduriez, Patrick, Pacitti, Esther: Data management in large-scale P2P systems. In: Daydé, Michel, Dongarra, Jack, Hernández, Vicente, Palma, José MLaginha M. (eds.) VECPAR 2004. LNCS, vol. 3402, pp. 104–118. Springer, Heidelberg (2005)
38. Xiang, X., Jin, T.: Efficient secure message routing for structured peer-to-peer systems. In: International Conference on Networks Security, Wireless Communications and Trusted Computing Wuhan, China (2009)
39. Xu, Z., Min, R., Hu, Y.: HIERAS: a DHT based hierarchical P2P routing algorithm. In: Proceedings of International Conference on Parallel Processing (ICPP), pp 187–194 (2003)
40. Yakouben, H., Litwin, W., Schwarz, T.: LH*RSP2P: a scalable distributed data structure for the P2P environment.In: International Conference on New Technologies of Distributed Systems, France (2008)
41. Yang, B., Garcia-Molina, H.: Designing a super-peer network. In: Proceedings of International Conference on Data Engineering ICDE, Bangalore, India (2003)
42. The Web Services Resource Framework. http://www.globus.org/wsrf
43. Zöls, S., Despotovic, Z., Kellerer, W.: Cost-based analysis of hierarchical DHT design. In: International Conference, P2P'06. IEEE Computer Society, Cambridge, pp 233–239 (2006)
44. Zöls, S., Hofstatter, Q., Despotovic, Z., Kellerer, W.: Achieving and maintaining cost-optimal operation of a hierarchical DHT system. In: Proceedings of International Conference. ICC, Germany (2009)
45. Zhao, B.Y., Huang, L., Stribling, J., Rhéa, S.C.: Tapestry: a resilient global scale overlay for service deployment. IEEE Int. J. Sel. Areas Commun. **22**(1) (2004)