# 6

# Expertise-Based Peer Selection

Ronny Siebes[1], Peter Haase[2], Frank van Harmelen[1]

[2]Vrije Universiteit Amsterdam, The Netherlands
{ronny,frankh}@cs.vu.nl
[1]Institute AIFB, University of Karlsruhe, D-76128 Karlsruhe, Germany
haase@aifb.uni-karlsruhe.de

## 6.1 Introduction

Peer-to-Peer systems are distributed systems without centralized control or hierarchical organization, in which each node runs software with equivalent functionality. A review of the features of recent Peer-to-Peer applications yields a long list: redundant storage, permanence, selection of nearby servers, anonymity, search, authentication, and hierarchical naming. Despite this rich set of features, scalability is a significant challenge: Peer-to-Peer networks that broadcast the queries to all peers do not scale - intelligent query routing and network topologies are required to be able to route queries to a relevant subset of peers. In this chapter we give an overview and an evaluation of the model of expertise based peer selection as proposed in [4] and how it is used in the Bibster system 18. In this model, peers use a shared ontology to advertise semantic descriptions of their expertise in the Peer-to-Peer network. The knowledge about the expertise of other peers forms a semantic overlay network, independent of the underlying network topology. If the peer receives a query, it can decide to forward it to peers about which it knows that their expertise is similar to the subject of the query. The advantage of this approach is that queries will not be forwarded to all or a random set of known peers, but only to the ones that have a good chance of answering it.

The organization of the sections in this chapter is as follows: In the next section, we give a small overview of related work in the domain of Semantic Overlay Networks. In section 3 we provide our generic model on expertise-based peer selection. In section 4, we instantiate the generic model with the Bibster case-study. In section 5, we show simulation experiments and their results on the selection method. Section 6 shows the results of an evaluation study on the Bibster application which was installed on different machines of interested people. Section 7 shows a comparison between the simulation results and the results obtained from the field study. Section 8 concludes our work.

## 6.2 Related Work on Semantic Overlay Networks

Peers that keep pointers to other peers which have similar content to themselves form a Semantic Overlay Network (SON). Edutella [6] is a schema based network where peers describe their functionality (i.e. services) and share these descriptions with other peers. In this way, peers know about the capabilities of other peers and only route a query to those peers that are probably able to handle it. Although, Edutella provides complex query facilities, it has still no sophisticated means for semantic clustering of peers, and their broadcasting does not scale well. Gridvine [3] uses the semantic overlay for managing and mapping data and metadata schemas, on top of a physical layer consisting of a structured peer-to-peer overlay network, namely P-Grid, for efficient routing of messages. In essence, the good efficiency of the search algorithm is caused not clustering of semantically related peers based on the semantic overlay, but by efficient term storage and retrieval characteristics of the underlying DHT approach for mapping terms to peers.

Another SON approach is to classify the content of a peer into a shared topic vector where each element in the vector contains the relevance for that given peer for the respective topic. pSearch [8] is such an example where documents in the network are organized around their vector representations (based on modern document ranking algorithms) such that the search space for a given query is organized around related documents, achieving both efficiency and accuracy. In pSearch each peer has the responsibility for a range for each element in the topic vector, e.g. $([0.2 - 0.4], [0.1 - 0.3])$. Now all expertise vectors that fall in that range are routed to that peer, meaning that, following the example vector, the expertise vector $[0.23, 0.19]$ would be routed to this peer and $[0.13, 0.19]$ not because $0.13$ does not fall in between 0.2 and 0.4. Besides the responsibility for a vector range, a peer also knows the list of neighbors which are responsible to vector ranges close to itself. The characteristic of pSearch is that the way that peers know about close neighbors is very efficient. A disadvantage of pSearch is that all documents have to be mapped into the same (low dimensional) semantic search space and that the dimensionality on the overlay is strongly dependent of the dimensionality of the vector, with the result that each peer has to know many neighbors when the vectors have high a dimension.

Another approach is based on random walk clustering [9], where peers with similar content are going to know each-other. The assumption is that queries posted by peers are semantically closely related to the content of the peer itself. This results in a high probability that the neighbors of the peer (the peers in the cluster of that peer) have answers to the query. The problem of this approach in the domain of full-text searches, is what information a peer has to tell to another peer so that they are able to determine if they are related or not. When there is no shared data-structure (like a fixed set of terms) in which they can describe their content, the whole content has to be shared. This results in that much data has to be shared between peers for determining closeness.

In contrast to the previous approach, the last SON approach that we discuss here lets peers describe their content in a shared set of terms. Mostly these terms are organized in a topic network or hierarchy making it able to determine the semantic similarity between terms. Each peer is characterized by a set of topics that describe its expertise. A peer knows about the expertise topics from other peers by analyzing advertisement messages [4] or answers (chapter 5). In this way peers form clusters of semantically related expertise descriptions. Given a query, a shared distance metric allows to forward queries (described by a shared set of terms) to neighbors of which their expertise description is semantically closely related to the query. The advantages of this approach are threefold:

- *Peer autonomy* Each peer can, in principle, have its own distance measure, peer selection mechanism and clustering strategy. This allows peers, for example to keep their neighbor list or similarity metric secret. Also peers can decide at any time to change their visibility on the network by sending advertisement messages.
- *Automatic load balancing* When some content is provided by many peers also the semantic cluster on that content will contain many peers. In this way, load balancing is an emergent property of this approach.
- *Robustness/fault tolerance* When peers leave the network or do not respond to a query, the only consequence is that they probably will not be asked a next time until they send new advertisement messages or are recommended by other peers. In contrast, most DHT approaches have to move routing tables to other peers in order to restore the overlay.

However there is also a disadvantage: terms that are not shared can not be found. For example, imagine that a peer has some documents containing the word ’abstract’, but the shared data-structure only contains the term ’summary’, then two things can be done (1) extend the shared data-structure with the word ’abstract’ so that peers are able to query and describe their expertise with that term or (2) the functions that extracts the expertise description and abstract the queries should be intelligent enough to see that ’summary’ is a good replacement for ’abstract’. Note that in this case the original query still contains ’summary’, but the routing mechanism uses the shared term ’abstract’ to route it to the peer that registered itself on that term. Both solutions have their own problems, the first one will lead eventually to very large data-structures, the second one depends very heavily on the quality of the extraction and abstraction algorithms.

## 6.3 A Model for Expertise Based Peer Selection

In the model that we propose, peers advertise their expertise in the network. The peer selection is based on matching the subject of a query and the expertise according to their semantic similarity. Figure 6.1 below shows the idea of the model in one picture.

In this section we first introduce a model to semantically describe the expertise of peers and how peers promote their expertise as advertisement messages in the network. Second, we describe how the received advertisements allow a peer to select

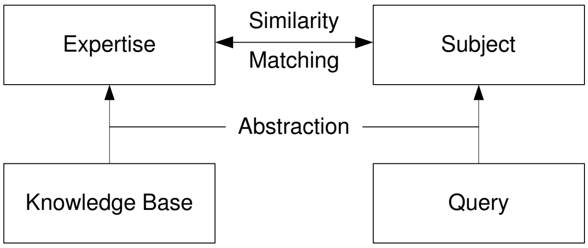


**Fig. 6.1.** Expertise Based Matching

other remote peers for a given query based on a semantic matching of query subjects against expertise descriptions. The third part describes how a *semantic overlay network* can be formed by advertising expertise.

### 6.3.1 Semantic Description of Expertise

***Peers***

The Peer-to-Peer network consists of a set of peers $P$. Every peer $p \in P$ has a knowledge base that contains the knowledge that it wants to share.

***Shared Ontology***

The peers share an ontology $O$, which provides a shared conceptualization of their domain. The ontology is used for describing the expertise of peers and the subject of queries.

***Expertise***

An expertise description $e \in E$ is a abstract, semantic description of the knowledge base of a peer based on the shared ontology $O$. This expertise can either be extracted from the knowledge base automatically or specified in some other manner.

***Advertisements***

Advertisements $A \subseteq P \times E$ are used to promote descriptions of the expertise of peers in the network. An advertisement $a \in A$ associates a peer $p$ with a an expertise $e$. Peers decide autonomously, without central control, whom to promote advertisements to and which advertisements to accept. This decision can be based on the semantic similarity between expertise descriptions.

### 6.3.2 Matching and Peer Selection

***Queries***

Queries $q \in Q$ are posed by a user and are evaluated against the knowledge bases of the peers. First a peer evaluates the query against its local knowledge base and then decides which peers the query should be forwarded to. Query results are returned to the peer that originally initiated the query.

***Subjects***

A subject $s \in S$ is an abstraction of a given query $q$ expressed in terms of the shared ontology. The subject can be seen a complement to an expertise description, as it specifies the required expertise to answer the query. We do not make any assumptions about the abstraction process, which preferably is done automatically. For example, a string matching approach could determine which parts of the ontology match with strings in the query.

***Similarity Function***

The similarity function $SF : S \times E \mapsto [0, 1]$ yields the semantic similarity between a subject $s \in S$ and an expertise description $e \in E$. An high value indicates high similarity. If the value is 0, $s$ and $e$ are not similar at all, if the value is 1, they match exactly. $SF$ is used for determining to which peers a query should be forwarded. Analogously, a same kind of similarity function $E \times E \mapsto [0, 1]$ can be defined to determine the similarity between the expertise of two peers.

***Peer Selection Algorithm***

The peer selection algorithm (c.f. Algorithm 1) returns a ranked set of peers. The rank value is equal to the similarity value provided by the similarity function.

**Algorithm 1** Peer Selection

```
let $A$ be the advertisements that are available on the peer
let $\gamma$ be a system parameter that indicates the minimal required similarity between the expertise of a peer and the topics of the query.
$subject := ExtractSubject(query)$
$rankedPeers := \emptyset$
for all $ad \in A$ do
    $peer := Peer(ad)$
    $rank := SF(Expertise(ad), subject)$
    if $rank > \gamma$ then
        $rankedPeers := (peer, rank) \cup rankedPeers$
    end if
end for
return $rankedPeers$
```

From this set of ranked peers one can, for example, select the best $n$ peers.

### 6.3.3 Semantic Overlay

The knowledge of the peers about the expertise of other remote peers is the basis for the Semantic Overlay Network. Here it is important to state that this SON is independent of the underlying network topology. At this point, we do not make any assumptions about the properties of the topology on the network layer.
The SON can be described by the following relation:

$Knows \subseteq P \times P$, where $Knows(p_1, p_2)$ means that $p_1$ knows about the expertise of $p_2$.

The relation $Knows$ is established by the selection of which peers a peer sends its advertisements to. Furthermore, peers can decide to accept an advertisement, e.g. to include it in their registries, or to discard the advertisement. The SON in combination with the expertise based peer selection is the basis for intelligent query routing.

## 6.4 Expertise Based Peer Selection in Bibster

We now describe the bibliographic scenario using the general model presented in the previous section. This scenario is identical to Bibster which is described in Chapter 18.

***Peers***

A researcher is represented by a peer $p \in P$. Each peer has an RDF knowledge base, which consists of a set of bibliographic metadata items that are classified according to the ACM topic hierarchy [1] The following example shows a fragment of a sample bibliographic item based on the Semantic Web Research Community Ontology (SWRC)[2]:

```
<rdf:RDF xmlns=
 "http://www.semanticweb.org/ontologies/swrc-onto.daml#"
  xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:acm ="http://daml.umbc.edu/ontologies/topic-ont#">
<Publication rdf:about="dblp:persons/Codd81">
 <title>The Capabilities of
        Relational Database Management Systems.</title>
 <acm:topic rdf:resource=
   "http://daml.umbc.edu/ontologies/classification#
    ACMTopic/Information_Systems/Database_Management"/>
 <!-- ... -->
</Publication> </rdf:RDF>
```

***Shared Ontology***

The ontology $O$ that is shared by all the peers is the ACM topic hierarchy. The topic hierarchy contains a set, $T$, of 1287 topics in the computer science domain and relations $(T \times T)$ between them: *SubTopic* and *seeAlso*.

### *Expertise*

The ACM topic hierarchy is the basis for our expertise model. Expertise $E$ is defined as $E \subseteq 2^T$, where each $e \in E$ denotes a set of ACM topics, for which a peer provides classified instances.

### *Advertisements*

Advertisements associate peers with their expertise: $A \subseteq P \times E$. A single advertisement therefore consists of a set of ACM topics [1] for which the peer is an expert on.

### *Queries*

We use the RDF query language SeRQL (chapter 1) to express queries against the RDF knowledge base of a peer. The following sample query asks for publications with their title about the ACM topic *Information Systems / Database Management*:

```
CONSTRUCT {pub} <swrc:title> {title} FROM {Subject} <rdf:type>
{<swrc:Publication>};
  <swrc:title> {title};
  <acm:topic>
  {<topic:ACMTopic/Information_Systems/Database_Management>}
USING NAMESPACE
swrc=<!http://www.semanticweb.org/ontologies/swrc-onto.daml#>, rdf
=<!http://www.w3.org/1999/02/22-rdf-syntax-ns#>, acm
=<!http://daml.umbc.edu/ontologies/topic-ont#>,
topic=<!http://daml.umbc.edu/ontologies/classification#>
```

### *Subjects*

Analogously to the expertise, a subject $s \in S$ is an abstraction of a query $q$. In our scenario, each $s$ is a set of ACM topics, thus $s \subseteq T$. For example, the extracted subject of the query above would be *Information Systems/Database Management*.

### *Similarity Function*

In this scenario, the similarity function $SF$ is based on the idea that topics which are close according to their positions in the topic hierarchy are more similar than topics that have a larger distance. For example, an expert on ACM topic *Information Systems/Information Storage and Retrieval* has a higher chance of giving a correct answer on a query about *Information Systems/Database Management* than an expert on a less similar topic like *Hardware/Memory Structures*.

To be able to define the similarity of a peer's expertise and a query subject, which are both represented as a set of topics, we first define the similarity for individual topics. [5] have compared different similarity measures and have shown that for measuring the similarity between concepts in a hierarchically structured semantic network, like the ACM topic hierarchy, the following similarity measure yields the best results:

$$S(t_1, t_2) = \begin{cases} e^{-\alpha l} \cdot \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} & \text{if } t_1 \neq t_2, \\ 1 & \text{otherwise} \end{cases} \tag{6.1}$$

Here $l$ is the length of the shortest path between topic $t_1$ and $t_2$ in the graph spanned by the *SubTopic* relation. $h$ is the level in the tree of the direct common subsumer from $t_1$ and $t_2$.

$\alpha \geq 0$ and $\beta \geq 0$ are parameters scaling the contribution of shortest path length $l$ and depth $h$, respectively. Based on their benchmark data set, the optimal values are: $\alpha = 0.2$, $\beta = 0.6$. Using the shortest path between two topics is a measure for similarity because Rada et al [7] have proven that the minimum number of edges separating topics $t_1$ and $t_2$ is a metric for measuring the conceptual distance of $t_1$ and $t_2$. The intuition behind using the depth of the direct common subsumer in the calculation is that topics at upper layers of hierarchical semantic nets are more general and are semantically less similar than topics at lower levels.

Now that we have a function for calculating the similarity between two individual topics, we define $SF$ as:

$$SF(s,e) = \frac{1}{|s|} \sum_{t_i \in s} \max_{t_j \in e} S(t_i, t_j) \tag{6.2}$$

With this function we iterate over all topics of the subject and average their similarities with the most similar topic of the expertise.

### *Peer Selection Algorithm*

The peer selection algorithm ranks the known peers according to the similarity function described above. Therefore, peers that have an expertise more similar to that of the subject of the query will have a higher rank. From the set of ranked peers, we now only consider a selection algorithm that selects the best $n$ peers.

## 6.5 Results of Simulation Experiments

In this section we describe the simulation of the scenario presented in section 6.4. With the experiments we try to validate the following hypotheses:

- **H1 - Expertise based selection:** The proposed approach of expertise based peer selection yields better results than a naive approach based on random selection. The higher precision of the expertise based selection results in a higher recall of peers and documents, while reducing the number of messages per query.
- **H2 - Ontology based matching:** Using a shared ontology with a metric for semantic similarity improves the recall rate of the system compared with an approach that relies on exact matches, such as a simple keyword based approach.
- **H3 - Semantic Overlay:** The performance of the system can be improved further, if the SON is built according to the semantic similarity of the expertise descriptions of the peers. This can be realized, for example, by accepting advertisements that are semantically similar to the own expertise.
- **H4 - The “Perfect” SON:** Perfect results in terms of precision and recall can be achieved, if the SON coincides with a distribution of the documents according to the expertise model.

### *Data Set*

To obtain a critical mass of bibliographic data, we used the DBLP data set, which consists of metadata for 380440 publications in the computer science domain.

We have classified the publications of the DBLP data set according to the ACM topic hierarchy using a simple classification scheme based on lexical analysis: A publication is said to be about a topic, if the label of the topic occurs in the title of the publication. For example, a publication with the title "The Capabilities of Relational Database Management Systems." is classified into the topic *Database Management*. Topics with labels that are not unique (e.g. *General* is a subtopic of both *General Literature* and *Hardware*) have been excluded from the classification, because typically these labels are too general and would result in publications classified into multiple, distant topics in the hierarchy. Obviously, this method of classification is not as precise as a sophisticated or manual classification. However, a high precision of the classification is not required for the purpose of our simulations. As a result of the classification, about one third of the DBLP publications (126247 out of 380440) have been classified, where 553 out of the 1287 ACM topics actually have classified publications. The classified DBLP subset has been used for our simulations.

### *Document Distribution*

We have simulated and evaluated the scenario with two different distributions, which we describe in the following. Note that for the simulation of the scenario we disregard the actual documents and only distribute the bibliographic metadata of the publications.

**Topic Distribution:** In the first distribution, the bibliographic metadata are distributed according to their topic classification. There is one dedicated peer for each of the 1287 ACM topics. The distribution is directly correlated with the expertise model, each peer is an expert on exactly one ACM topic and contains all the corresponding publications. This also implies that there are peers that do not contain publications, because not all topics have classified instances.

**Proceedings Distribution:** In the second distribution, the bibliographic metadata are distributed according to conference proceedings and journals in which the according publications were published. For each of the conference proceedings and journals covered in DBLP there is a dedicated peer that contains all the associated publication descriptions (in the case of the 328 journals) or inproceedings (in the case of the 2006 conference proceedings). Publications that are published neither in a journal nor in conference proceedings are contained by one separate peer. The total number of peers therefore is 2335 (=328+2006+1). With this distribution one peer can be an expert on multiple topics, as a journal or conference typically covers mutliple ACM topics. Note that there is still a correlation between the distribution and the expertise, as a conference or journal typically covers a coherent set of topics.

*Simulation Environment*

To simulate the scenario we have developed and used a controlled, configurable Peer-to-Peer simulation environment. A single simulation experiment consists of the following sequence of operations:

1. *Setup network topology:* In the first step we create the peers with their knowledge bases according to the document distribution and arrange them in a random network topology, where every peer knows 10 random peers. We do not make any further assumptions about the network topology.
2. *Advertising Knowledge:* In the second step, the SON is created. Every peer sends an advertisement of its expertise to all other remote peers it knows based on the network topology. When a peer receives an advertisement, it may decide to store all or selected advertisements, e.g. if the advertised expertise is semantically similar to its own expertise. After this step the SON is static and will not change anymore.
3. *Query Processing:* The peers randomly initiate queries from a set of randomly created 12870 queries, 10 for each of the 1287 ACM topic. The peers first evaluate the queries against their local knowledge base and then propagate the query according to their peer selection algorithms described below.

*Experimental Settings*

In our experiments we have systematically simulated various settings with different values of input variables. In the following we will describe an interesting selected subset of the settings to prove the validity of our hypotheses.

***Setting 1***

In the first setting we use a naive peer selection algorithm, which selects n *random* peers from the set of peers that are known from advertisements received, but disregarding the content of the advertisement. In the experiments, we have used n=2 in every setting, as a rather arbitrary choice.

***Setting 2***

In the second setting we apply the expertise based selection algorithm. The *best* n (n=2) peers are selected for query forwarding. Here the peer selection algorithm only considers *exact* matches of topics.

***Setting 3***

In the third setting we modify the peer selection algorithm to use the ontology based similarity measure, instead of only exact matches. The peer selection only selects peers whose expertise is equally or more similar to the subject of the query than the expertise of the forwarding peer.

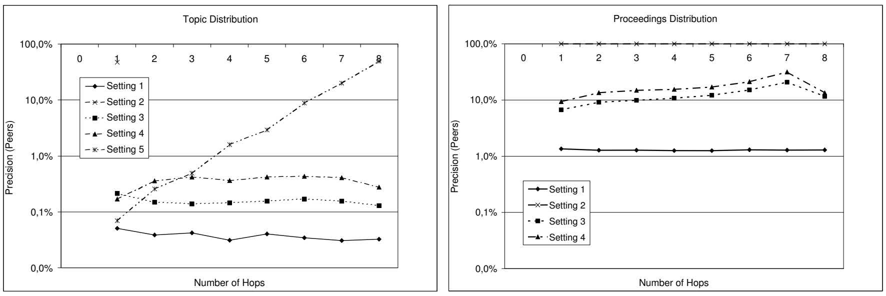


**Fig. 6.2.** $Precision_{Peers}$

### *Setting 4*

In the fourth setting we modify the peer to only accept advertisements that are semantically similar to its own expertise. The threshold for accepting advertisements was set to accept on average half of the incoming advertisements.

### *Setting 5*

In this setting we assume global knowledge to impose a perfect topology on the peer network. In this perfect topology the *knows* relation conincides with the ACM topic hierarchy: Every peer knows exactly those peers that are experts on the neighboring topics of its own expertise. This setting is only applicable for the distribution of the publications according to their topics, as this model assumes exactly one expert per topic.

The following table summarizes the instantiations of the input variables for the described settings:

| Setting # | Peer Selection | Advertisements | Topology |
|---|---|---|---|
| Setting 1 | random | accept all | random |
| Setting 2 | exact match | accept all | random |
| Setting 3 | ontology based match | accept all | random |
| Setting 4 | ontology based match | accept similar | random |
| Setting 5 | ontology based match | accept similar | perfect |

### *Simulation Results*

Figures 6.2 through 6.5 show the results for the different settings and distributions. The simulations have been run with a varying number of allowed hops. In the results we show the performance for a maximum of up to eight hops. Zero hops means that the query is processed locally and not forwarded. Please note that the diagrams for the number of messages per query and recall (i.e. Figures 6.5, 6.3, 6.4) present cumulative values, i.e. they include the sum of the results for *up to* n hops. The

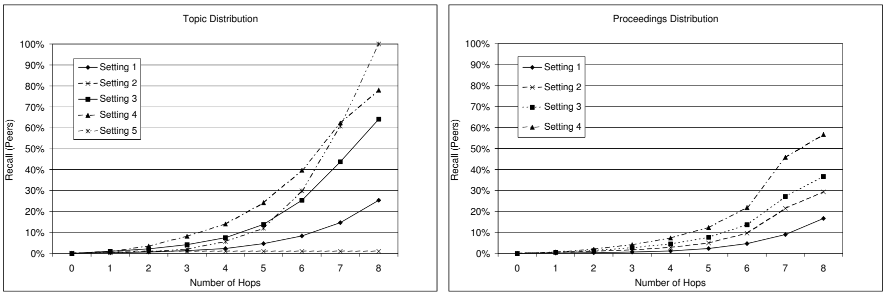
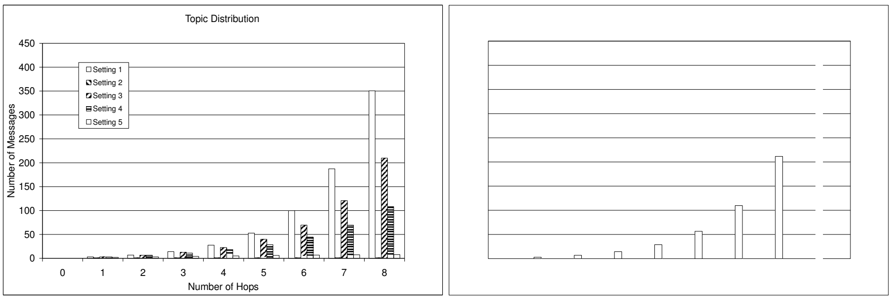


**Fig. 6.3.** $Recall_{Peers}$



for the proceedings distribution. This results in a fairly low recall of peers and documents despite a high number of messages, as shown in Figures 6.3, 6.5, 6.4, respectively. With the expertise based selection, either exact or similarity based matching, the precision can be improved considerably by about one order of magnitude. For example, with the expertise based selection in Setting 3, the precision of the peer selection (Figure 6.2) can be improved from 0.03% to 0.15% for the topic distribution and from 1.3% to 15% for the proceedings distribution. With the precision, also the recall of peers and documents rises (Figures 6.3, 6.5). At the same time, the number of messages per query can be reduced. The number of messages sent is influenced by two effects. The first effect is message redundancy: The more precise the peer selection, the higher is the chance of a peer receiving a query multiple times on different routes. This redundancy is detected by the receiving peer, which will forward the query only once, thus resulting in a decreasing number of queries sent across the network. The other effect is caused by the selectivity of the peer selection: It only forwards the query to peers whose expertise is semantically more or equally similar to the query than that of the own expertise. With an increasing number of hops, as the semantic similarity of the expertise of the peer and the query increases, the chance of knowing a qualifying peer decreases, which results in a decrease of messages.

### R2 - Ontology based matching

The result of Figure 6.2, Setting 2, shows that the exact match approach results in a maximum precision already after one hop, which is obvious because it only selects peers that match exactly with the query's subject. However, Figure 6.3 shows that the recall in this case is very low in the case of the topic distribution. This can be explained as follows: For every query subject, there is only one peer that exactly matches in the entire network. In a sparse topology, the chance of knowing that relevant peer is very low. Thus the query cannot spread effectively across the network, resulting in a document recall of only 1%. In contrary, Setting 3 shows that when semantically similar peers are selected, it is possible to improve the recall of peers and documents, to 62% after eight hops. Also in the case of the proceedings distribution, where multiple exact matches are possible, we see an improvement from 49% in the case of exact matches (Setting 2), to 54% in the case of ontology based matches (Setting 3). Naturally, this approach requires to send more messages per query and also results in a lower precision.

### R3 - Semantic Overlay Network

In Setting 4 the peers only accept semantically similar advertisements. This has proven to be a simple, but effective way for creating the SON that correlates with the expertise of the peers. This allows to forward queries along the gradient of increasing semantic similarity. When we compare this approach with that of Setting 3, the precision of the peer selection can be improved from 0.15% to 0.4% for the topic distribution and from 14% to 20% for the proceedings distribution. The recall of documents can thus be improved from 62% to 83% for the topic distribution and from 54% to 72% for the proceedings distribution.

It is also interesting to note that the precision of the peer selection for the similarity based matching decreases slightly after seven hops (Figure 6.2). The reason is that after seven hops the majority of the relevant peers has already been reached. Thus the chance of finding relevant peers decreases, resulting in a lower precision of the peer selection.

### R4 - The "Perfect" SON

The results for Setting 5 show how one could obtain the maximum recall and precision, if it were possible to impose an ideal SON. All relevant peers and thus all bibliographic descriptions can be found in a deterministic manner, as the query is simply routed along the route which corresponds to the shortest path in the ACM topic hierarchy. At each hop the query is forwarded to exactly one peer until the relevant peer is reached. The number of messages required per query is therefore the length of the shortest path from the topic of expertise of the originating peer to that of the topic of the query subject. The precision of the peer selection increases to the maximum when arriving at the eight hop, which is the maximum possible length of a shortest path in the ACM topic hierarchy. Accordingly, the maximum number of messages (Figure 6.4) required is also eight.

## 6.6 Results of Field Study

In the Bibster system (c.f. Chapter 18) we implemented two different query forwarding strategies that ran at the same time, namely our expertise-based method and a random query forwarding algorithm. In this way we are able to see how our approach performs in real life. The Bibster system was made publicly available and advertised to researchers in the Computer Science domain. The evaluation was based on the analysis of system activity that was automatically logged to log files on the individual Bibster clients. We have analyzed the results for a period of three months (June - August 2004). With respect to query routing and the use of the expertise based peer selection, we were able to reduce the number of query messages by more than 50 percent, while retaining the same recall of documents compared with a naive broadcasting approach. Figure 6.6 shows the precision of the peer selection (the percentage of the reached peers that actually provided answers to a given query): While the expertise based peer selection results in an almost constant high precision of 28%, the naive algorithm results in a lower precision decreasing from 22% after 1 hop to 14% after 4 hops[1].

Figure 6.7 shows the number of forwarded query messages sent per query. It can be seen that with an increasing number of hops, the number of messages sent with the expertise based peer selection is considerably lower than with the naive algorithm. Although we have shown an improvement in the performance, the results also show that with a network of the size as in the field experiment, a naive approach is also

[1] The decrease is due the redundancy of relevant peers found on different message paths: Only distinct relevant peers are considered.

acceptable. On the other hand, with a growing number of peers, query routing and peer selection becomes critical. In the previous discussed simulation experiments, networks with thousands of peers improve in the order of one magnitude in terms of recall of documents and relevant peers.

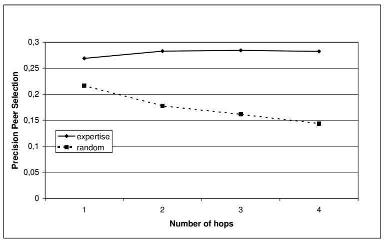


**Fig. 6.6.** $Precision_{Peers}$

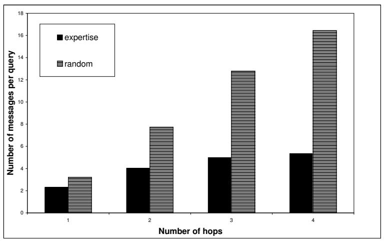


**Fig. 6.7.** $Number_{Messages}$

## 6.7 Comparison with Results from Simulation Experiments

Overall, the results of the simulation experiments have been validated: We were able to improve the precision of the peer selection and thus reduce the number of sent messages. However, the performance gain by using the expertise based peer selection was not as significant as in the simulation experiments[2].

[2] In terms of recall, there were no improvements at all, as even the naive algorithm generally was able to reach all relevant peers.

This is mainly due to the following reasons:

- *Size of the network* The size of the network in the field experiment was considerably *smaller* than in the simulation experiments. While the total number of participating peers was already fairly large (398), the number of peers online at a certain point in time was fairly small (order of tens).
- *Network topology* In the field experiment we built the SON on-top of the JXTA network topology. Again, related to the small size of the network, the JXTA topology degenerates to a fully connected graph in most cases. Obviously, for these topologies, a naive algorithm yields acceptable results.
- *Distribution of the content* In the simulation experiments, we distributed the shared content according to certain assumptions (based on topics, conferences, journals). In real world experiments, the distribution is much more heterogeneous, both in terms of the expertise of the peers and the amount of shared content.

## 6.8 Conclusion

In this paper we have presented a model for expertise-based peer selection, in which a SON among the peers is created by advertising the expertise of the peers. We have shown how the model can be applied in a bibliographic scenario. Simulation experiments that we performed with this bibliographic scenario show the following results:

- Using expertise-based peer selection can increase the performance of the peer selection by an order of magnitude (result R1).
- However, if expertise-based peer selection uses simple exact matching, the recall drops to unacceptable levels. It is necessary to use an ontology-based similarity measure as the basis for expertise-based matching (result R2).
- An advertising strategy where peers only accept advertisements that are semantically close to their own profile (i.e. that are in their semantic neighborhood) is a simple and effective way of creating a SON. This semantic topology allows to forward queries along the gradient of increasing semantic similarity (result R3).
- The above results depend on how closely the SON mirrors the structure of the ontology. All relevant performance measure reach their optimal value when the network is organized exactly according to the structure of the ontology (result R4). Although this situation is idealized and in will in practice not be achievable, the experiment serves to confirm our intuitions on this.

Also, the field experiment showed that we were able to improve the precision of the peer selection and thus reduce the number of sent messages. However, the performance gained by using the expertise based peer selection was not as significant as in the simulation experiments. Summarizing, in both the simulation experiments and the field experiments, we have shown that expertise-based peer selection combined with ontology-based matching outperforms both random peer selection and selection based on exact matches, and that this performance increase grows when the SON more closely mirrors the domain ontology.

## References

[1] The ACM Topic Hierarchy.
`http://www.acm.org/class/1998/.`

[2] The Semantic Web Research Community Ontology.
`http://ontobroker.semanticweb.org/ontos/swrc.html.`

[3] K. Aberer, P. Cudré-Mauroux, M. Hauswirth, and T. Van Pelt. Gridvine: Building internet-scale semantic overlay networks. In *3rd International Semantic Web Conference (ISWC2004)*, pages 107–121, Hiroshima, Japan, 7-11 November 2004.

[4] P. Haase, R. Siebes, and F. van Harmelen. Peer selection in peer-to-peer networks with semantic topologies. In Mokrane Bouzeghoub, editor, *Proceedings of the International Conference on Semantics in a Networked World (ICNSW'04)*, volume 3226 of *LNCS*, pages 108–125, Paris, June 2004. Springer Verlag.

[5] Y. Li, Z. A. Bandar, and D. McLean. An approach for measuring semantic similarity between words using multiple information sources. *Transactions on Knowledge and Data Engineering*, 15(4):871–882, July/August 2003.

[6] W. Nejdl, B. Wolf, . Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmer, and T. Risch. Edutella: A p2p networking infrastructure based on rdf. In *Proceedings of the 11th International World Wide Web Conference*, May 2002. schema based searching Presentation: http://www2002.org/presentations/nejdl.pdf.

[7] R. Rada, H. Mili, E. Bicknell, and M. Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30, 1989.

[8] C. Tang, Z. Xu, and S. Dwarkadas. Peer-to-peer information retrieval using self-organizing semantic overlay networks. Technical report, HP Labs, November 2002.

[9] S. Voulgaris, A.-M. Kermarrec, L. Massoulie, and M. van Steen. Exploiting semantic proximity in peer-to-peer content searching. In *10th International Workshop on Future Trends in Distributed Computing Systems (FTDCS)*, Suzhou, China, may 2004.