Efficient data store and discovery in a scientific P2P network

Alexander Löser
Computation and Information Structures (CIS)
Technische Universität Berlin
10587 Berlin, Germany
aloeser@cs.tu-berlin.de

Martin Wolpers
Wolf Siberski
Wolfgang Nejdl
Learning Lab Lower Saxony
30167 Hannover, Germany
wolpers, siberski,nejdl@learninglab.de

Abstract

In this paper we propose to use schema-based peer-to-peer systems for the exchange of scienitific documents. Compared with recent file sharing networks highly domain-oriented scientific documents are shared in relatively small research communities with tight internal and only a few external connections. The documents are annotated with small a but well-defined sets of metadata using standard taxonomies and are stored in many distributed, autonomous, heterogenous data repositories. As a recent development super-peer networks emerge improving the network performance by clustering peers at super-peers. By using *Semantic Overlay Clusters (SOC)* for clustering the super-peer networks we enable the creation of context-specific, logical views over the physical P2P network topology according to the research communities demands. SOCs define peer clusters according to the meta-data description of peers and their contents. The respective clustering policy expresses the demand on the peers for a particular research context. In this paper we show how SOCs can be defined based on policies. We detail the definition of conditions that we use for our ECA-rule like policiy definition approach.

Semantic Overlay Clusters in schema-based P2P networks

Using peer-to-peer systems for the exchange of files, especially of music files, is a quite common application. Recently peer-to-peer networks have also been used successfully to interconnect between distributed heterogenous scientific data stores enabling the exchange of scientific documents and the search in complex heterogenous meta data structures. Examples for this new class of peer-to-peer networks, so called schema based peer-to-peer networks, are (Aberer, Cudré-Mauroux, & Hauswirth 2003; Halevy *et al.* 2003; Bernstein *et al.* 2002; Nejdl *et al.* 2003; Löser *et al.* 2003b). Such networks combine approaches from peer-to-peer research as well as from the database and semantic web research areas. These networks allow the aggregation and integration of data from autonomous, distributed data sources. They build upon peers that use explicit schemas to describe their content. Naturally such meta data is pretty heterogenous as documents stem from a wide variety of domains and communities. However, current schema-based P2P networks still have some shortcomings, e.g. broadcasting all queries to all data store and are not scalable. Therefore, intelligent routing- and network organization strategies are essential in such networks enabling queries to be routed to a *semantically chosen subset of peers* able to answer parts or whole queries. First approaches to enhance routing efficiency in a clustered network have already been proposed by (Ng, Sia, & King 2003) and (Semantic Overlay Networks 2002).

Current schema based Peer-to-Peer Systems therefore distinguish between a technical network layer and a semantic clustering layer. While the technical layer provides efficient algorithms for maintaining an real existing network topology, a *Semantic Overlay Clusters (SOC)* (Semantic Overlay Networks 2002) layer provides a virtual context-specific view on selected peers. The SOC Layer therefore abstracts from the underlaying technical infrastructure and topology and enables the use of efficient integration technologies and an context specific query routing in schema based Peer-to-Peer networks.

We expect that for scientific publication researchers gather in rather small communities with some connections among each other. In the context of P2P systems we can therefore assume that most document searches (queries) must not be broadcasted to all peers. Instead, queries should be send to only those peers that are able to answer them. By employing SOCs within the P2P network we advance the state-of-the-art for restricting complex query broadcast to only those peers capable of meaningful query-answering and of integrating small groups of schemas for a particular context (clustering) (Löser *et al.* 2003a). SOCs are designed for large and highly distributed networks improving search and semantic interoperability enabling either a search-driven or integration-driven clustering of the network in logically portions. As there are a lot of scientific communities with a large amount of scientific documents a P2P network employing SOCs is just perfect, also in regard to the dynamic behavior of communities. Similar to the creation of views in database systems SOCs are defined by human exports. They act as a virtual, abstract, independent views of selected peers in a schema-based P2P system.

We propose to use a super-peer/peer approach as described in (Yang & Garcia-Molina March 2003) and more specifically for the schema-based Edutella P2P network in (Nejdl *et al.* 2003) for enabling SOCs. The basic idea here is that the super-peer establishes and maintains a specific SOC. See section for a more detailed explanation on super-peers. In order to let super-peers create and maintain SOCs we have to tackle a wide variety of challenges. Among them are the definition of dimensions used for the identification of suitable peers for a SOC. In comparison to views in a centralized DBMS or data ware house, where concrete tables are used to define a view, the definition of SOCs in a schema-based P2P system requires more abstract concepts for their definition. These dimensions are used for identifying and clustering peers and should include semantic contexts and query schemas of the SOC. The sum of all definitions regarding one SOC we call the SOC policy (or policy for short). The policy states the conditions on which a peer is able to join a SOC. It is necessary to establish the respective matching operators. By relying on an already established logical language, like Datalog, the P2P network supports the automated identification of suitable peers for a SOCs within a given search space of dimensions. Furthermore a P2P network in general and schema-based P2P network in particular are fast changing networks with unpredictable behavior. The network layer evolves permanently, e.g. by peers connecting and disconnecting autonomously, so that SOCs have to handle their policy-based clustering algorithm automatically, using event based notification for detecting changes in the network structure.

In this paper we present our first work to establish a schema-based P2P system for the exchange of scientific documents based on SOCs. In section we describe how we define SOCs as abstract views over an evolving P2P network topology thus defining the dimensions to be used in SOC policies. Based on these definitions we investigate possible operators for establishing conditions for mapping/matching the peers descriptions to the policies in section . Section describes how we use such policies in the Edutella P2P network to improve complex query routing.

Clustering Peers- Dimensions and Policies

Usually, peers that act as information provider establish the basic schema-based P2P network. These peers are wrappers to particular information sources, such as a DBMS, a Web-Service or an RDF Store. Each peer is capable of describing its relevant features, such as query capabilities, export scheme(s), classification(s), Peer-ID, Quality of Services, etc. using a metadata based model. The attributes are the possbile dimensions that describe a peer. Deriving the dimensions and subsequently the metadata model that describes a peer is usually done semiautomatically, e.g. (Naumann 2002) presents approaches for obtaining information about peer quality automatically.

We aim at using SOCs for clustering community relevant documents. The documents are usually classified based on taxonomies that the community aggred upon. Therefore, we rely on the *classification* of the peers within domain specific taxonomies where the classification is based on theclassification of the documents stored at the peer. The taxonomies used for classification are well established and agreed upon by the respective community. Examples of such taxonomies are the ACM Computer Science Classification (ACM_CCS, (Association for Computing Machinery 1998) and the Software Engineering Book of Knowledge (SWEBOK, (IEEE Software Engineering Coordinating Committee (SWECC) 2001)) classification.

In order to capture these classifications we define two attributes, **classifiedBy** and **taxonPath**. The attribute *classifiedBy* contains the URL of any recognized taxonomy or any user-defined taxonomy. The attribute *taxonPath* represents an entry in a classification as a path from a more general to more specific entry in a classification.

Both attributes are used to determine if peers have the right content for a particular query. To determine if the peers have the right interface to answer to a specific query we introduce as a second dimension the *query capabilities* of a peer, i.e. the set of queries supported by the peer (Vassalos & Papakonstantinou 1997). Elements used in a query are matched against the schema information of these attributes for a particular information provider peer in order to determine if the respective peer is able to answer the query. A positive match in this context states that a peer understands and is able to answer a specific query, but does not guarantee a non-empty answer set. Note, that generally in schema-based P2P networks query capabilities are described by schemas, e.g. by an RDF schema (Nejdl *et al.* 2003) or XML schema (Galanis *et al.* 2003) at different granularities. For example, a simple peer providing scientific papers of a work group may provide a schema just by using the Dublin Core Schema elements *dc:title* and *dc:subject*.

We define the two attributes **usesExportSchema** and **usesProperty** for describing the schema(s) that the peer uses to describe its content (more attributes are established in (Nejdl *et al.* 2003)). The attribute *usesExportSchema* holds the URI that describes the content of the peer. We thus assume that different peers support different schemas and that these schemas can be uniquely identified by an URI, e.g. the ACM CSS classification. The attribute *usesProperty* enables peers to not rely on a complete schema for their content description but to use parts of such, i.e. only certain properties as in the example above. While this is unusual in conventional database systems, it is more often used for data stores using semi-structured data, and very common for RDF-based systems.

All used properties of a schema can be represented as an tree based graph. In the following RDF-based example[1] we show such a tree based graph model for providing materials for UML education annotated by using selected elements of the ACM classification on Computer Science (ACM CSS) standard. We assume that the classifications are already specified in RDF.

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:psd="http://edutella.jxta.org/psd#">

  <rdf:Description rdf:about="urn:jxta:uuid-123456">
    <psd:classifiedBy>
      <psd:Classification>
        <psd:taxonomy rdf:resource="http://acm.org/CCS"/>
```

[1] Currently we evaluate the use of existing standards, such as DAML-S, for representing the syntax of such a model.

```
        <psd:taxonPath rdf:resource="http://acm.org/
                                       CCS#D.2.1.1"/>
      </psd:Classification>
    </psd:classifiedBy>
    <psd:classifiedBy>
       <psd:Classification>
         <psd:taxonomy rdf:resource="http://swebok.org/
                                            classification"/>
         <psd:taxonPath rdf:resource="http://swebok.org/
                              classification/SoftwareDesign/
                                   SoftwareDesignNotations"/>
       </psd:Classification>
    </psd:classifiedBy>
    <psd:usesProperty rdf:resource="http://purl.org/dc/
                                        elements/1.1/title"/>
    <psd:usesProperty rdf:resource="http://purl.org/dc/
                                      elements/1.1/subject"/>
  </rdf:Description>
</rdf:RDF>
```

For simplifications in the following sections we will only rely on the above defined dimensions. Identifying peers with further dimensions, such as quality aspects, response time, geographical range, are possible as well, yet out of scope of this paper.

Policy-based matching

For establishing a context specific SOC we need to determine if a peer matches a SOC policy. Furthermore we need to detect and react upon changes in the underlying network structure so that such changes are reflected in the SOC. A SOC policy states conditions which must be true for a peer in order to join the SOC. Based on these conditions and in order to react upon network structure changes the policy must also define some events that trigger a certain super-peer behaviour. To state the policy we rely on a notation inspired by *Event-Condition-Action (ECA)* rules in active databases which is enhanced with logical operators: ON event IF condition DO action. Each SOC policy consists of *rules* that define the conditions that must hold for an information provider peer joining the semantic overlay cluster. In table 1 an example is given with five rules that define actions between a semantic overlay cluster c and its information provider peers p.

The rule 1.1 in table 1 states that if a peer p triggers a peer-entering event at a super-peer c the respective condition (several policy constraints are true) must be true in order to let peer p access the SOC established at super-peer c. The following example shows a complete clustering policy expressing the demand on peers containing e-learning materials for software design and using the Dublin Core Standard as query schema:

```
ON Enter (Peer p, Cluster c)
IF (
    (usesSchema="http://purl.org/dc/elements/1.1/")
    AND (classifiedBy="http://swebok.org")
    AND (taxonPath >= "http://swebok.org/SoftwareDesign")
) DO Approve(Peer p, Cluster c)
```

Policy-based clusters for Edutella

Within the Edutella project (Nejdl *et al.* 2002a) we develop a schema-based P2P network by the same name. This network uses an architecture of a centralized topology embedded in decentralized systems thus forming a super-peer network. super-peers introduce hierarchy into the network in the form of super-peer nodes – peers which have extra capabilities and duties in the network. A super-peer acts as a centralized server to a subset of clients, e.g. information provider and information consumer peers. Clients submit queries to their super-peer node and receive results from it, as in a hybrid system. super-peers are also connected to each like peers in pure P2P systems, routing messages over this overlay network, and submitting and answering queries on behalf of their clients and themselves. Examples of super-peer networks are JXTA, Edutella or Morpheus. Because a super-peer network combines elements of both pure and hybrid systems, it has the potential to combine the efficiency of a centralized search with the autonomy, load balancing, robustness to attacks and at least semantic interoperability provided by distributed search.

Each of the super-peers will function as a SOC to a suitable subset of peers. Based on the SOC policy mechanism described above the super-peer can accept or reject a peer to its cluster. Matching the SOC policy conditions to the peer descriptions is based on the query language QEL (Nejdl *et al.* 2002b; Nilsson & Siberski 2003) which is used throughout the whole P2P network. QEL is based on the logical data description language datalog which is well defined – we can use any datalog capable engine for the matching process that evaluations the SOC policy rules. The example below shows the condition from the above example formulated in QEL:

```
# Namespace-Declaration
  @prefix qel: <http://www.edutella.org/qel#>.
  @prefix psd: <http://www.edutella.org/psd#>.
  @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.

# approve, if Classification and Schema fit
  approve(Peer) :-
    fitsSchema(Peer, <http://purl.org/dc/elements/1.1/>),
             fitsClassification(Peer,<http://swebok.org>,
                     <http://swebok.org/SoftwareDesign>).

# general rules to be stated in every policy:

# a peer declares its usage of a schema directly
# (isClassifiedBy)
  fitsSchema(Peer, Schema) :- qel:s(Peer,psd:usesSchema,
                                                   Schema).

# a peer declares its usage of a property within a
# certain schema
  fitsSchema(Peer, Schema) :- qel:s(Peer, psd:usesProperty,
          Property),s(Property, rdfs:isDefinedBy, Schema).

# the classification matches if its taxonomy and topic
# match
  fitsClassification(Peer, Taxonomy, Topic) :-
    qel:s(Peer,psd:Classification,Classification),
    qel:s(Classification,psd:taxonomy,Taxonomy),
    fitsTopic(Classification,Topic).
```

No	**Event**	**Condition**	**Action**	Explanation
1.1	$P_{Enter}(p, c)$	constraints == True	$Action = Approve(p, c)$	Peer p approved at Cluster c
1.2	$P_{Enter}(p, c)$	constraints == False	$Action = Reject(p, c)$	Peer p rejected from Cluster c
2.1	$P_{Leave}(p, c)$	-	$Action = Delete(p, c)$	Peer p deleted from Cluster c
3.1	$P_{Check}(p, c)$	constraints == True	$Action = Approve(p, c)$	Peer p (re-)approved at Cluster c
3.2	$P_{Check}(p, c)$	constraints == False	$Action = Reject(p, c)$	Peer p rejected from Cluster c

Table 1: Rules within a clustering policy

```
# the classification contains the defined topic (in
# TaxonPath) directly...
  fitsTopic(Classification, Topic) :-
    qel:s(Classification, psd:taxonPath, Topic).

# ... or contains a superset of the topic
  fitsTopic(Classification, Topic) :-
     fitsTopic(C, Super),
     qel:s(Super, psd:superTopic, Topic).
```

Related work

The idea of placing data nodes together, so queries can be efficiently routed and a semantic integration of the nodes is more automatized, has been discussed in many research projects. In the field of federated databases the tightly coupled mediator-wrapper architecture (Wiederhold 1992) was proposed by Wiederhold, enabling a static integration of domain-specific data stores. Kemper et.al. proposed in (Kemper & Wiesner 2001) the concept of Hyperqueries, a dynamic distributed query processing method on the Internet. Matchmaking Infrastructures, such as InfoSleuth (Kashyap & Sheth 2000) or OBSERVER(Mena *et al.* 1996), match information provider to information consumers in a centralized way using description logics. In the Artificial Intelligence field the conceptual clustering problem has been widely studied in inductive learning systems, such as in COBWEB(Fisher 1987) and LABYRINTH (Thompson & Langley 1991). Other approaches for routing queries directly to existing clusters are proposed by (Ng, Sia, & King 2003). However, most systems assume that documents are part of a controlled collection located at a central database and allow only a centralized matching. Recently semantic overlay networks for peer-to-peer systems (Semantic Overlay Networks 2002) allow overlays for placing data nodes semantically together. However they allow only the use of limited meta data schemes, such as simple filenames, and are designed for pure peer-to-peer networks, without using advantages of super-peer networks.

Conclusion

This paper makes several novel contributions: We introduced the concept of semantic overlay clusters in scientific peer-to-peer networks. SOC's are designed for very large, highly distributed networks improving search and semantic interoperability. We identified two dimensions for clustering the physical network into context specific logical views: content classification and query capabilities. Further on we showed concepts enabling SOCs in a existing peer-to-peer network, allowing a dynamic clustering of data stores: RDF-based models for data stores and clustering policies expressing the demand on data stores based on Datalog semantics. Open work includes studying existing powerful graph based matching techniques between policies and models as well as the investigation of algorithms for a load balanced distribution of peers to clusters.

References

Aberer, K.; Cudré-Mauroux, P.; and Hauswirth, M. 2003. The chatty web: Emergent semantics through gossiping. In *Proceedings of the Twelfth International World Wide Web Conference (WWW2003)*.

Association for Computing Machinery. 1998. ACM Computing Classification System. http://www.acm.org/class/1998/.

Bernstein, P. A.; Giunchiglia, F.; Kementsietsidis, A.; Mylopoulos, J.; Serafini, L.; and Zaihrayeu, I. 2002. Data management for peer-to-peer computing: A vision. In *Proceedings of the Fifth International Workshop on the Web and Databases*.

Fisher, D. 1987. Knowledge acquisition via incremental conceptual clustering. *Machine Learning* 2:139–172.

Galanis, L.; Wang, Y.; Jeffery, S. R.; and DeWitt, D. J. 2003. Locating data sources in large distributed systems. In *Proceedings of the 29 VLDB*.

Halevy, A. Y.; Ives, Z. G.; Mork, P.; and Tatarinov, I. 2003. Piazza: Data management infrastructure for semantic web applications. In *Proceedings of the Twelfth International World Wide Web Conference (WWW2003)*.

IEEE Software Engineering Coordinating Committee (SWECC). 2001. Software Engineering Book of Knowledge. http://www.swebok.org/.

Kashyap, V., and Sheth, A. 2000. *Information Brokering Across Heterogeneous Digital Data A Metadata-based Approach*. Boston/Dordrecht/London: Kluwer Academic Publishers.

Kemper, A., and Wiesner, C. 2001. Hyperqueries: Dynamic distributed query processing on the internet. In *The VLDB Journal*, 551–560.

Löser, A.; Naumann, F.; Siberski, W.; Nejdl, W.; and Thaden, U. 2003a. Semantic overlay clusters within super-peer networks. In *To Appear in:Proceedings of the International Workshop on Databases, Information Systems and Peer-to-Peer Computing in Conjunction with the VLDB 2003*.

Löser, A.; Nejdl, W.; Wolpers, M.; and Siberski, W. 2003b. Information Integration in Schema-Based Peer-To-Peer Networks. In *Proceedings of the 15th International Conference of Advanced Information Systems Engieering (CAiSE 03)*.

Mena, E.; Kashyap, V.; Sheth, A. P.; and Illarramendi, A. 1996. OBSERVER: An approach for query processing in global information systems based on interoperation across pre-existing ontologies. In *Conference on Cooperative Information Systems*, 14–25.

Naumann, F. 2002. *Quality-Driven Query Answering for Integrated Information Systems - doctorial thesis*. Springer Verlag, lecture notes in computer science, 2261 edition.

Nejdl, W.; Wolf, B.; Qu, C.; Decker, S.; Sintek, M.; Naeve, A.; Nilsson, M.; Palmér, M.; and Risch, T. 2002a. EDUTELLA: a P2P Networking Infrastructure based on RDF. In *Proceedings of the Eleventh International World Wide Web Conference (WWW2002)*.

Nejdl, W.; Wolf, B.; Qu, C.; Decker, S.; Sintek, M.; Naeve, A.; Nilsson, M.; and Palm, M. 2002b. Edutella: A p2p networking infrastructure based on rdf. In *11th International World Wide Web Conference (WWW2002)*.

Nejdl, W.; Wolpers, M.; Siberski, W.; Löser, A.; Bruckhorst, I.; Schlosser, M.; and Schmitz, C. 2003. Super-Peer-Based Routing and Clustering Strategies for RDF-Based Peer-To-Peer Networks. In *Proceedings of the Twelfth International World Wide Web Conference (WWW2003)*.

Ng, C.-H.; Sia, K.-C.; and King, I. 2003. Peer clustering and firework query model in the peer-to-peer network. Technical report, Chinese University of Hongkong, Department of Computer Science and Engineering.

Nilsson, M., and Siberski, W. 2003. RDF Query Exchange Language (QEL) Specification. http://edutella.jxta.org/spec/qel.html.

2002. Semantic overlay networks. Submitted for publication.

Thompson, K., and Langley, P. 1991. Concept formation in structured domains. In *D. Fisher, M. Pazzani, and P. Langley, editors, Concept formation: knowledge and experience in unsupervised learning. Morgan Kaufmann.*

Vassalos, V., and Papakonstantinou, Y. 1997. Describing and using query capabilities of heterogeneous sources (extended version). Technical Report Available at http://www-db.stanford.edu/pub/papers/query-cap-ext.ps.

Wiederhold, G. 1992. Mediators in the architecture of future information systems. *IEEE Computer* 25(3):38 – 49.

Yang, B., and Garcia-Molina, H. March 2003. Designing a super-peer network. In *Proccedings of the ICDE*.