Knowledge Discovery and Reuse in a Distributed Hypermedia System

S. Wiesener¹, W. Kowarschick²

¹Bayerisches Forschungszentrum für Wissensbasierte Systeme (FORWISS), München
²Fakultät für Informatik, Technische Universität München
email: [wiesener,kowa]@informatik.tu-muenchen.de

Abstract

Conventional distributed hypermedia systems suffer from some well-known problems that make effective, goal-directed document retrieval and maintenance impossible. Relationships between documents are modeled on a very low level of abstraction. That prevents knowledge reuse, is prone to redundancies, and such relationships are very costly to maintain.

Here we present the SemaLink approach of a distributed resource discovery and delivery system that offers access to information through querying and knowledge-based hypermedia browsing. Relationship knowledge is modeled on a high level of abstraction apart from documents. Navigation paths are computed by the system, combining semantic networks with information retrieval methods.

1 Introduction

Actually more than 40 million participants are estimated to have access to the Internet worldwide, many of them being not only passive consumers but also qualified information providers. Electronic publishing in the Internet became an integral part of scientific information exchange, though there is no homogeneous content-based access structure to information resources. Since the bandwidth and global development of wide area networks is increasing continuously the publishing process is becoming more and more decentralized, often down to the level of single authors. As a direct consequence, the information resource discovery problem has reached a new dimension.

As the process of human information search usually consists of both querying and browsing steps, two different sorts of meta-knowledge are necessary to support the user in finding information [8]. Firstly, to access documents by querying documents have to be described by a set of attributes. Typical attributes suitable for document retrieval are Title, Author, Keywords, Classification, Mediatype etc. Even full-text retrieval can be seen as an attribute search where all occurring words are stored in one attribute. Classical digital library and catalog systems [1], [2], [9] offer attribute retrieval only.

Secondly, relationships between documents have to be modeled in the system to enable hypermedia browsing. A relationship consists at least of a source description and a destination description of related documents. In addition, it may contain further attributes like link type or weight [6]. In most hypermedia systems (e.g. Hyper-G [7] or World-Wide Web [4]) hyperlinks are modeled on a very low level of abstraction. They explicitly point from one chunk of information to another. This aggravates the well-known lost-in-hyperspace syndrome in large-scale systems. The intentions, background-knowledge, and context the authors had at link creation time usually do not match with those that individual users have during retrieval. Users are therefore condemned to follow a lot of links pointing to non-relevant information. Even if a link is appropriate to a user’s context, it does not guarantee to deliver all related documents that might be relevant in that context. So hypermedia retrieval can degenerate into a mixture of depth-first and breadth-first docuverse exploration soon overchar ging the user’s cognitive abilities.
Besides, the consistent administration of global hypermedia webs is costly, if not completely impossible.

To overcome those problems and to achieve a system-supported goal-directed navigation through the document space two fundamental changes are necessary. At first, knowledge about relationships between documents has to be modeled on a higher level of abstraction. That means, source and destination descriptors of hyperlinks should, in general, not address information by means of explicit pointers. Rather descriptors should contain attributes that describe related documents in a declarative manner. Secondly, authoring, maintaining, storing, and distribution of relationship knowledge should be done independently from documents.

The main idea of the SemaLink approach is that authors produce both documents and/or relationship knowledge [12]. Relationship knowledge interconnects different but related concepts. It is modeled in many independent, delimited semantic networks [10]. Nodes of these networks represent concepts; typed and weighted edges express relationships between them. The whole distributed hypermedia is no longer a statically connected graph but a set of documents and rather small semantic networks. Documents and semantic networks now form a virtual hypermedia graph structure (see Figure 1), virtual in the sense that relationships between documents are partly computed by the system. They may cross several independent semantic networks. The joining of documents and semantic networks is achieved by means of information retrieval: Similarities amongst attributes of semantic nodes and documents are computed. Furthermore, the whole process can be supported by intelligent deductive retrieval clients that adapt the virtual graph structure according to a user model.

2 Knowledge-Based Hypermedia Document Retrieval

Although SemaLink is not based on a statically connected hypermedia web, the basic components are nodes with attributes as well as typed and weighted links (cp. Dexter Reference Model [6]). We distinguish document nodes from semantic nodes. While document nodes contain mul-

---

**Figure 1: A virtual hyperlink passing two knowledge-bases**

The main idea of the SemaLink approach is that authors produce both documents and/or relationship knowledge [12]. Relationship knowledge interconnects different but related concepts. It is modeled in many independent, delimited semantic networks [10]. Nodes of these networks represent concepts; typed and weighted edges express relationships between them. The whole distributed hypermedia is no longer a statically connected graph but a set of documents and rather small semantic networks. Documents and semantic networks now form a virtual hypermedia graph structure, virtual in the sense that relationships between documents are partly computed by the system. They may cross several independent semantic networks. The joining of documents and semantic networks is achieved by means of information retrieval: Similarities amongst attributes of semantic nodes and documents are computed. Furthermore, the whole process can be supported by intelligent deductive retrieval clients that adapt the virtual graph structure according to a user model.
timedia data, semantic nodes are used to model relationship knowledge. Links are static, inter-
connect semantic nodes and describe relationships between them. Document nodes, on the other
hand, are connected to semantic nodes by means of computed attribute value similarities. Fur-
thermore, independent semantic networks may be connected in the same way. Thus, relation-
ship knowledge between documents is formulated on a high level of abstraction.

The hypertext interface metaphor can now be applied to navigate the distributed set of node and
link objects. Initially users locate start nodes by means of declarative queries. The system
returns nodes satisfying those queries. These nodes may be documents with multimedia con-
tents as well as semantic nodes. Of course, it increases retrieval quality if query terms are
weighted and the matching of query and node attribute terms is fuzzy. Result nodes are pre-
sented to the user in a list ranked according to their degree of similarity to the query. The user
now decides which nodes are to be presented.

An actually presented node is the initial point for a knowledge-based browsing through the dis-
tributed information universe. This action is a sequence composed of the following three steps:

1. The system tries to find semantic nodes which are similar to the actual document node.
   This is achieved by comparing the attributes of the nodes.

2. The system now follows static links starting from those semantic nodes. This traversal
   process may be either controlled manually (i.e., determined by users in single steps) or
   automatically (e.g., determined by rules). However, eventually a set of semantic
   destination nodes is reached.

3. To locate related documents, similarities between the semantic destination nodes and
document nodes are computed. Resulting documents are presented to the user with an
appropriate link type and relevance ranking. Besides, similarities to nodes of other
semantic networks can be computed to integrate further semantic knowledge by step 2.

If path traversal shall be supported by a rule-based system, relevance has to be defined to restrict
the navigation space. To this end paths have to be rated and ranked. Only deduced paths with a
certain relevance are considered for further navigation. The relevance of a path depends on the
following features:

- **Weight of static links between semantic nodes**: Authors can express the strength of a
  single relationship in an attribute *Weight*.

- **Similarity value of virtual edges**: Paths containing virtual edges with a low similarity
  value lose relevance because coherence and context are no more guaranteed.

- **Path length**: With increasing path length coherence decreases even if all edges are
  heavily weighted. For example, if a user is looking at a document that deals with cars
  in a very general way, a system-generated link should not directly point to the technical
description of a sparking plug, even if there is a strong transitive *is-part-of* relation-
ship.

Note that in the SemaLink approach it is not necessary to explicitly link any new document to
all related ones that are already part of the document space. Rather, existing knowledge is used
to compute appropriate links dynamically.

### 3 A Distributed Agent/Broker Architecture

In contrast to conventional distributed hypermedia systems the SemaLink approach is not only
based on resource locators (e.g., URLs of the WWW [3]). This raises a resource discovery prob-
Documents and relationship knowledge have to be located in a large unstructured information space before being combined in the above described manner.

Figure 2 depicts an agent/broker architecture that solves the resource discovery problem. To guarantee a certain degree of scalability, document contents (i.e., memory consuming multimedia data) are separated from document attributes and stored independently. In SemaLink only node attributes and links are used for the whole document retrieval process.

**Document servers** are located at the authors’ hosts. Their main task is to deliver node contents corresponding to a set of node attribute value pairs. Document servers are only accessed when a user decides to have a document presented. Each node provides an attribute pointing statically onto its contents.

**Brokers** manage node attributes and links for a certain number of authors. They serve as well-known collection points from which agents obtain their data. The main purpose of brokers is to avoid the necessity of agents to access many single hosts in order to obtain data, because this would result in extremely high networking costs.

In contrast to brokers **agents** are active units. They connect to brokers and retrieve node attributes and links, i.e., they search for specific descriptions of new documents and delimited semantic networks. Each agent maintains a list of brokers that are to be visited periodically. By executing a *retrieve* operation offered by brokers agents generally use the select predicate to retrieve only those node attributes and links that match their interests. Every agent has a specific profile that is also provided to retrieval clients to facilitate choosing appropriate agents. Besides node attributes and links agents also compute and manage virtual edges, i.e., they finally put up the virtual hypermedia. Though similarities could be, at least in principle, computed at navigation time, it makes great sense to compute them only once and materialize them as virtual edge objects at the agents.
Retrieval clients are user front-ends of the SemaLink approach. They are responsible for user-interaction and manage the communication with agents and document servers. Retrieval clients obtain meta-data (i.e., node attributes, virtual edges, and links) from an agent while they access node contents directly from document servers.

4 Summary and Outlook

In this paper the SemaLink approach for semantic navigation through large distributed document spaces has been proposed. We discussed the combination of semantic relationship knowledge with multimedia documents that results in a coherent, knowledge-based hypermedia. Virtual edges glue together independent but related fragments of knowledge and documents. The so defined hypermedia is navigated by using computed relationships that also can take into consideration user-specific rules and preferences. A broker/agent architecture solves the resource discovery problem arising with the introduction of virtual edges.

5 References