

Conceptual Linking: Ontology-based Open Hypermedia

Leslie Carr, Wendy Hall
Intelligence, Agents, Multimedia
Department of Electronics and Computer Science,
University of Southampton,
Southampton SO17 1BJ, UK
{lac,wh}@ecs.soton.ac.uk

Sean Bechhofer, Carole Goble
Information Management Group
Department of Computer Science
University of Manchester
Oxford Road
Manchester M13 9PL, UK
{seanb, carole}@cs.man.ac.uk

Abstract

This paper describes the attempts of the COHSE project to define and deploy a Conceptual Open Hypermedia Service. Consisting of

- *an ontological reasoning service which is used to represent a sophisticated conceptual model of document terms and their relationships;*
- *a Web-based open hypermedia link service that can offer a range of different link-providing facilities in a scalable and non-intrusive fashion;*

and integrated to form a *conceptual hypermedia* system to enable documents to be linked via metadata describing their contents and hence to improve the consistency and breadth of linking of WWW documents at retrieval time (as readers browse the documents) and authoring time (as authors create the documents).

Keywords

Open hypermedia, link service, ontology, navigation, metadata.

Introduction: concepts and metadata

Metadata is data that describes other data to enhance its usefulness. The library catalogue or database schema are canonical examples. For our purposes, metadata falls into three broad categories:

- *Catalogue information: e.g. the artist or author, the title, a picture's dimensions, a document's revision history;*
- *Structural content: e.g. headings, titles, links; for a picture its shapes, colors and textures;*
- *Semantic content: what the document/picture is about e.g. football, sport, person holding trophy, hope, joy.*

Metadata activities have been a major focus of interest for the WWW community, especially for information providers, publishers and digital libraries. The takeup of the eXtensible Markup Language (XML) has been particularly concerned with its applications for expressing data about documents, and has most recently been used to define the Resource Description Framework (RDF) [26]. The aim of RDF is to provide a standard framework for expressing statements about data objects, especially statements giving information about authors, publishers, version and keyword information (these attributes being standardized as the Dublin Core [31]).

An ontology is a formal model of the kinds of concepts and objects that appear in the real world, together with the relationships between them. Ontologies take a variety of forms, from hierarchical classification schemes such as Yahoo! directories to logic-based models. All these forms include at least a vocabulary of terms and some specification of the meaning of those terms.

Using Metadata for Linking

Providing conceptual content-based information as the attributes of web pages is an important activity, enabling search engines to provide query results that are more pertinent. Currently such concepts are usually simple keywords. Hypermedia systems such as the Distributed Link Service [9, 10] may make use of this information to provide a rudimentary "conceptual hypermedia" by clustering documents with the same tag value keyword for retrieval purposes and linking documents with the same tag value for navigation. Keywords effectively classify documents into clusters that share the same set of keywords, or variations of them if stemming is used.

To achieve the kind of diversity of association required for non-trivial Web applications, documents need to be linked in many dimensions based on their content. Constructing such links manually is inconsistent and error-prone [17]. Furthermore, it obfuscates one of the chief reasons for associating documents; that their contents are similar in some way. Conceptual Hypermedia Systems (CHS) specify the hypertext structure and behaviour in terms of a well-defined conceptual schema [7, 28, 33]. This types documents and links, and includes a conceptual domain model used to describe document content. Consequently, information about the hypertext is represented explicitly as metadata that can be reasoned over, for example, using the domain model as a classification structure to classify the documents; documents that share metadata are deemed to be similar in some way. Authoring links between documents becomes an activity of authoring with concepts; concepts are linked and hence their associated documents are linked.

Open Hypermedia Systems and Link Services

Common usage of the Web involves embedding links within documents in the HTML format; in this sense the Web can be considered a 'closed' hypermedia system. However, there is nothing inherent in the Web infrastructure that prevents hypertext links from being abstracted away from the documents and managed separately, for example, by using XLink's third-party links [14]. In Open Hypermedia Systems (OHS) links are first class objects, stored and managed separately from multimedia data; like documents they can be stored, transported, cached and searched, and their use can be instrumented. OHS have been well researched by the hypermedia community [21, 29] and increasingly Web publishing applications adopt the open hypermedia approach [27, 32].

The DLS provides a powerful framework to aid navigation and authoring and addresses some of the issues of distributed information management [16]. Using an intermediary model [2], the DLS adds links and annotations into documents as they are delivered through a proxy from the original WWW server to the ultimate client browser. It uses a number of software modules to recognize different opportunities for adding various kinds of links to the documents, creating a user-specific navigational overlay that can be used to superimpose a coherent interface to sets of unlinked or insular resources (such as the Eprint archives addressed by the Open Citation project [24]).

The DLS treats link creation and resolution as a service that may be provided by a number of link resolution engines. For example, it uses resolvers which recognize keywords, names of people and bibliographic citations as potential link anchors according to different heuristics and knowledge bases. These link resolvers are hardwired into the monolithic system or chained sequentially [15] so that each one sees the document with links added by the previous resolver. This inherently synchronous arrangement means that any delay is a delay in the critical path of document delivery, hence all processing must be relatively light-weight and tightly coupled.

By contrast, a COHSE needs a Distributed Link Resolution Service (DLRS) to allow link resolvers to be distributed across multiple servers and decoupled from the delivery of the document. The aim is to allow complex computation, such as involved with implementing conceptual inferencing logic to provide added value for document authoring and browsing without impeding the delivery of the core document itself.

COHSE

A terminology-based query system can be added to the portfolio of link resolvers to provide consistent navigation links based on the concepts contained in the contents or meta-data of the multimedia pages that are being browsed.

For the purposes of this demonstration, the link service is integrated into the browser client, and consists of a Java applet that monitors the user's interaction with the browser together with a set of JavaScript functions that manipulate the HTML DOM. The components of the link service (Figure 1) are brought to bear on the web page as soon as it has been received by the browser, and so no longer form an obstacle to the delivery of the document. Once the set of links has been chosen, the page is refreshed and redisplayed.

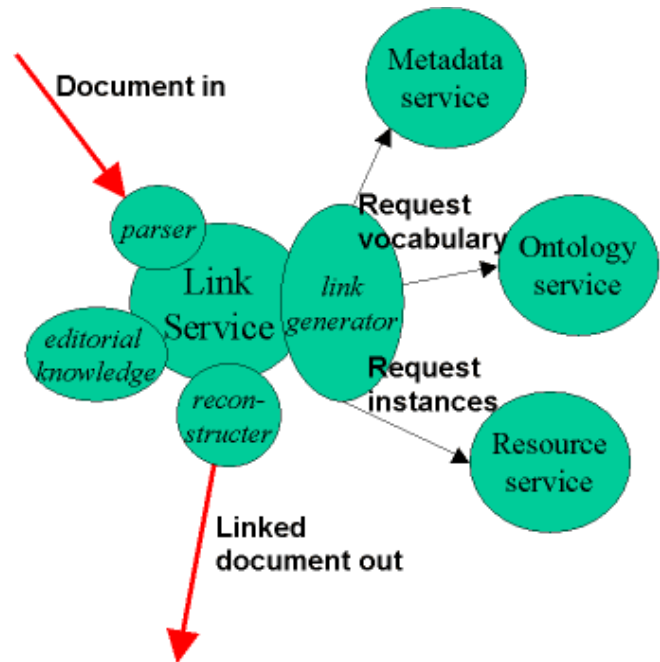


Figure 1: The Ontology Service maps between natural language terms and a concept graph. The Resource Service obtains Web pages representing the concepts. The Link Generator uses the ontology terms to make links. Editorial knowledge is used to prune or expand the links using ontology semantics.

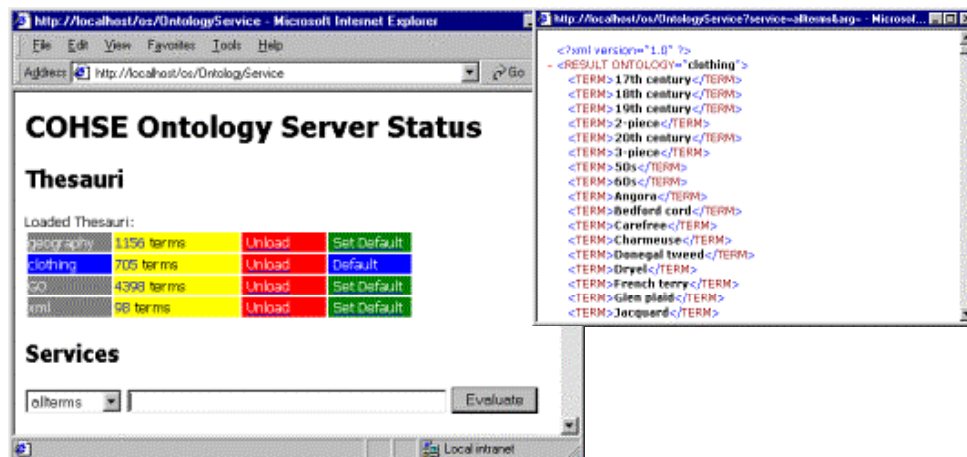


Figure 2: The *Ontology Service* can be queried for terms and concept relationships.

The ontology service (a Java servlet, shown operating in figure 2) manages ontologies, that is to say, sets of concepts that are related together according to some schema. The ontologies currently used take the form of a thesaurus, *i.e.* concepts related by *broader-term*, *narrower-term* and *related-term* relations, and are stored as XML data, with all queries and results being mediated through a simple XML document type.

The link generator module of the link service contacts the ontology service to obtain a complete listing of all the language terms that are used to represent the concepts in the ontology. For each of those terms that are recognised as occurring in the document, the generator first asks the ontology service for a preferred term, and then asks for the preferred term to be mapped onto a concept. Having identified a concept from the strings in the document, the link generator contacts the resource service to obtain a list of documents that contain instances of this concept. At this point, a number of destinations have been identified for a particular link anchor and the editorial module evaluates the number and quality of potential links obtained

from the generator. If the number of links is not consistent with the formation of a *well-linked document*, it will choose to request broader or narrower terms from the ontology service in order to expand or cull the set of anchor destinations. When all the terms in the whole document have been processed, the constructor can add hypertext links with particular presentation styles and behaviours. Figure 3 shows how links are added to an example document (the COHSE control panel appears in figure 3b and the link behaviour in 3c and 3d is shown in debugging mode, so that the link expands within the document text).

The metadata service is another independent servlet which allows documents to be decorated with metadata: language terms from a specific ontology. The service can either harvest specific tags from the documents themselves or apply external 'metadata links' to a read-only document from an independent linkbase. The effect is to declare that a whole document, or any range within it, should be processed with a specific ontology, or that a particular region in the content corresponds to a particular term in the ontology.

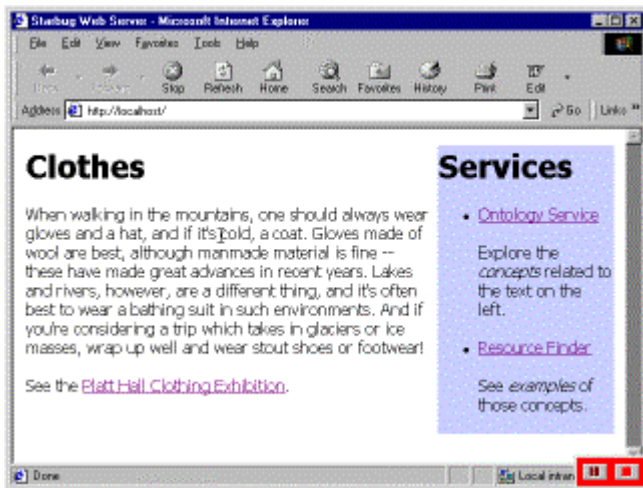


Figure 3a: A page about clothes...

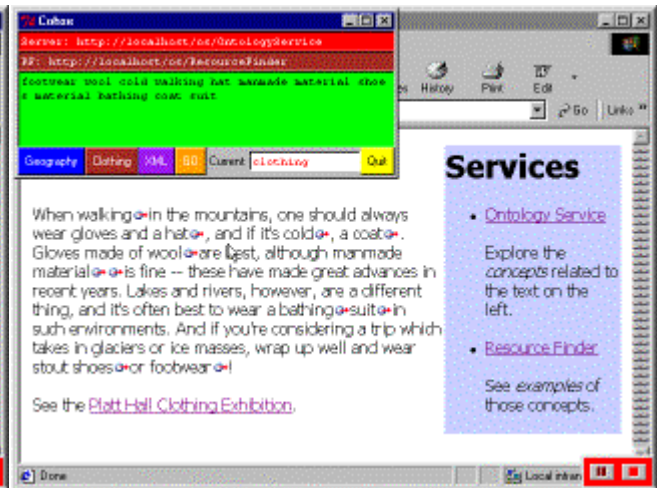


Figure 3b: ... is linked against the clothing ontology...

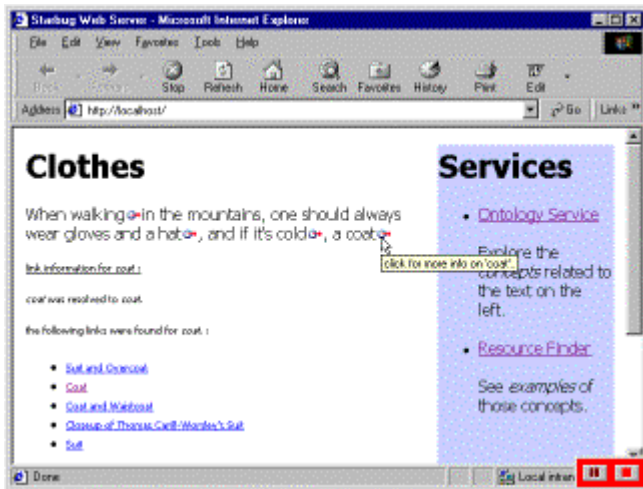


Figure 3c: ... some terms link well with the correct number of destination links...

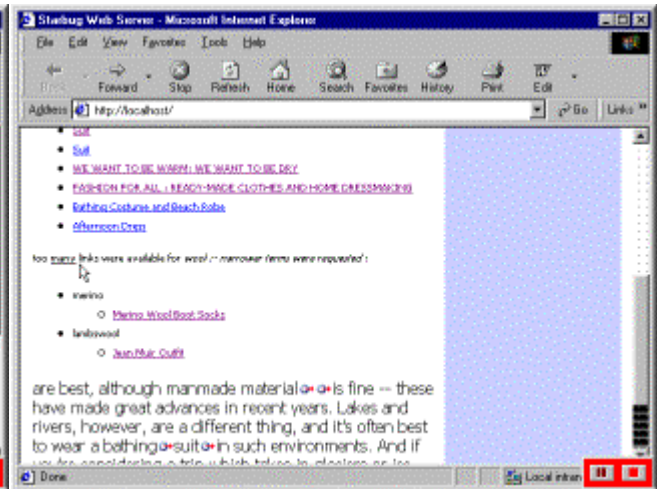


Figure 3d: ... while some are pruned.

The novel part of the link resolution process is the use of the editorial knowledge component to take advantage of the implicit structure of the ontology to make informed decisions about the kind of links to choose. By making a selection from a set of 'narrower terms' the list of links can be usefully reduced whilst broadening the recognised concept can be used as a strategy to increase the number of links.

Alternative Approaches to Concepts

The COHSE uses a predefined ontology to choose candidate anchors for creating links. This section lists some of the other systems that concern themselves with manipulating concepts, the different ways of representing them and the various modes for deploying them.

Meta Tags

HTML's <META> tags allow authors to specify information about web resources. This is highly uncontrolled though, as the tags contain unconstrained terms and are used for a variety of purposes (indicating author, how the page was generated, content, special things for particular applications and so on). In this case, the metadata is tightly bound to the documents. In order to discover the metadata we must examine the document itself. There is no central metadata repository, but process such as web robots can be sent off to harvest and cache the metadata.

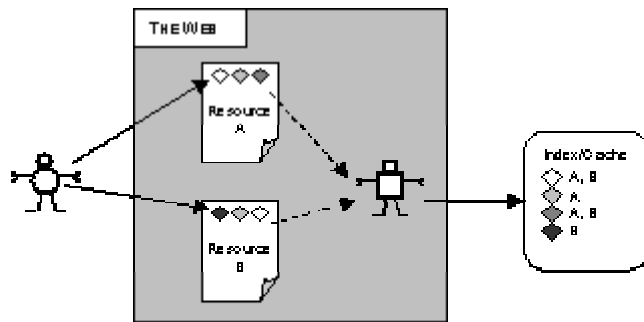


Figure 4: Simple Meta Tags

The index duplicates the metadata from the resources, but provides easy access without having to go out to the Web. Of course with such an approach, the issue of maintenance is crucial. It is difficult to tell whether the index is up to date. The repository or index is also very simple — in general there's no ontological structure, just a list of arbitrary keywords.

Yellow Pages

With a Yellow Pages service, such as Yahoo!, pages are classified according to their content. A taxonomy or hierarchy is normally used, with subject areas being broken down. This is generally achieved by hand, with both the classification hierarchy and the categorization of the pages done manually. The pages themselves are unaltered, so the situation here is that the metadata is stored externally from the documents and has no real link to the documents, other than through the classification.

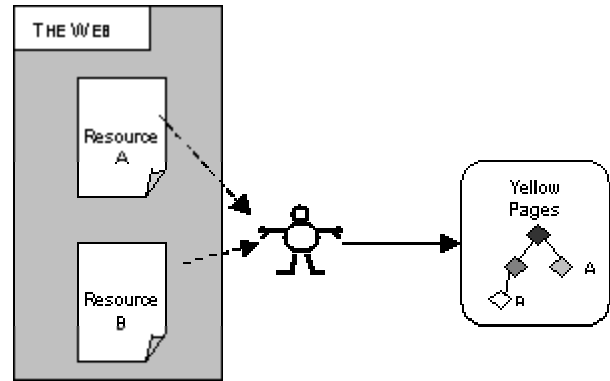


Figure 5: Yellow Pages

Again, this provides a "snapshot" of the situation, so there may be problems that the classification is not up to date. There's also very little automation going on here — this approach may be geared towards supporting humans trying to locate resources rather than providing machine-readable knowledge.

SHOE

The Simple HTML Ontology Extension (SHOE) [22, 23] has been developed by the Parallel Understanding Systems Group in the Department of Computer Science at the University of Maryland. SHOE provides mechanisms that allow the definition of ontologies and the assertion of claims about resources with respect to those ontologies.

Assertions about particular web pages (or resources) are included within pages as mark-up using an HTML based syntax, with a META tag used to inform any agents that the page uses SHOE. The assertions take the form of instance descriptions, asserting membership of classes and relationships between the instances.

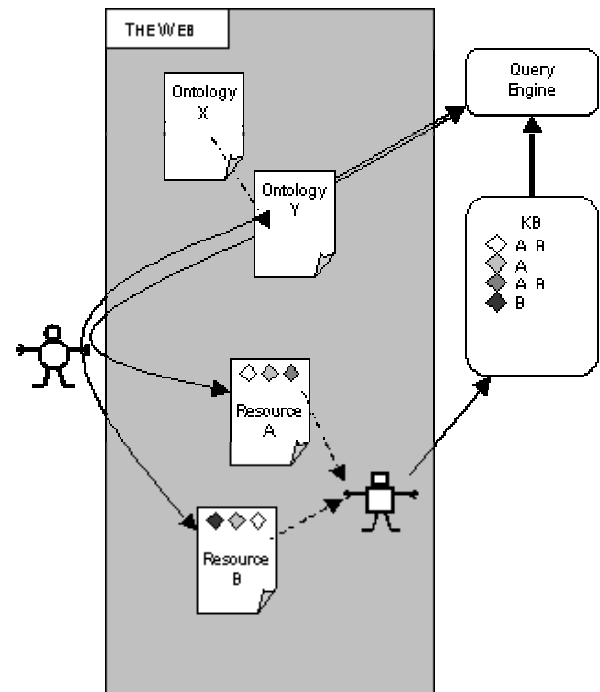


Figure 6: SHOE

In the SHOE model, as shown in figure 6, information is spread around. The metadata is attached explicitly to the documents (through the use of the SHOE HTML extension), but can then be gathered in one place by a robot for later query. Presumably, if one had a SHOE-enabled browser the user could also examine the metadata in situ once they had reached a SHOE annotated page.

Ontobroker

Ontobroker (or On2broker as it is also now known) [13, 19], is a system and architecture from AIFB, Karlsruhe. It is similar in many ways to SHOE and allows the annotation of web pages with ontological metadata. It provides a more expressive framework for the ontologies, using Frame-Logic for the specification of ontologies, annotation data and queries. Ontobroker and SHOE share some characteristics. They both use annotation of the documents themselves, and then rely on web crawlers (crawling through a well-defined docuverse) to harvest the metadata, storing it in a knowledge base. The KB is then queried using the ontology as a schema for query forming. They do differ in a couple of aspects. SHOE provides ontology extension mechanisms and explicitly places the ontologies on the Web. It's less clear how one gains access to the Ontobroker ontologies or how one makes the link between the instance markup and the ontology it applies to.

Karina

In Karina [12], an ontology is used to describe the content of documents in a multimedia repository. This metadata is then used to construct or author a presentation that fits the needs of a particular user. Karina is using the metadata as an index (resource discovery). In Karina, however, the emphasis is that the ontology will then be used in order to structure the results. Karina is thus closely related to COHSE, in that the ontology is being used to produce a structure.

RDF

RDF (Resource Description Framework) [26] differs from systems like SHOE and Ontobroker as it's a framework rather than a particular implemented system — systems such as Ontobroker may use RDF as a representation format. It's useful to compare it here though. RDF provides a framework which allows us to talk about metadata. The RDF data model is based around ideas of triples, i.e. an object, relationship and value. The intention with RDF is that metadata can be held separately from the documents (using the "about" attribute of RDF), separating

the metadata from the document. With RDF, though, the RDF documents themselves are on the Web, so the repository is accessible.

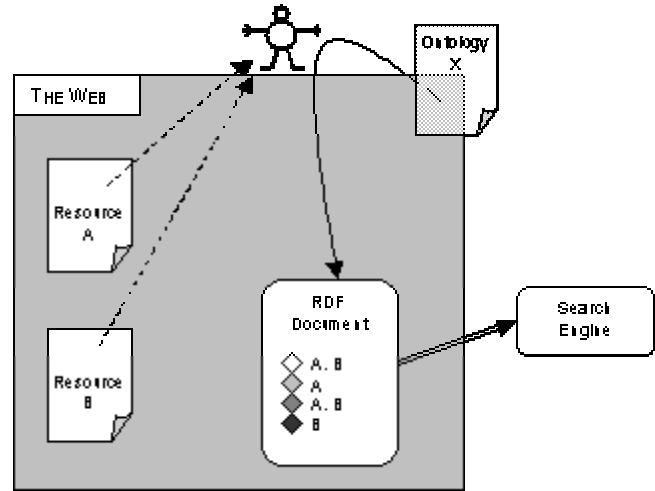


Figure 7: RDF

Figure 7 shows how RDF descriptions might work. Note again that as RDF is simply a framework, this is only one of a number of possible ways that things could be put together. RDF is less prescriptive in its ontology specification — indeed most uses of RDF so far seem to be in order to specify minimum data sets such as the Dublin Core [31], so it is less clear here whether the ontology might sit on the web (as with, say SHOE), or somewhere else. We could consider the RDF document as forming a “knowledge base” or repository with collected metadata about a number of resources.

COHSE

COHSE combines the Distributed Link Service (DLS) [7] architecture with a conceptual model to provide Conceptual Open Hypermedia. By using independent ontology, resource and metadata storage services, concepts referred to in Web resources can be identified and matched against potential ‘link destinations’ for navigational purposes. In the original DLS the ability to map from language terms to hypertext link destinations is governed by the existence of human-authored databases of hypertext links from which to choose. In COHSE, the process is driven instead by the various inter-relationships of concepts in the ontology.

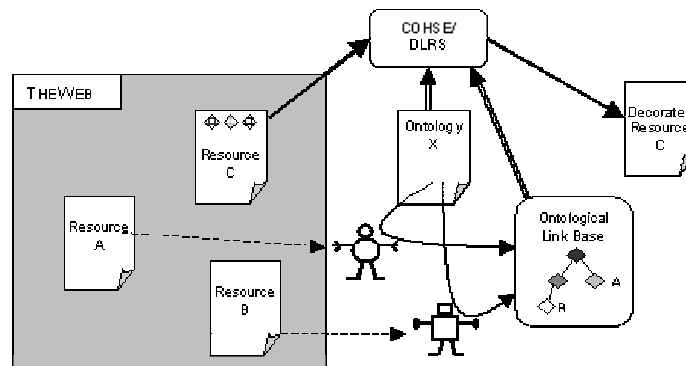


Figure 8: COHSE

