

# Open Hypermedia as a Navigational Interface to Ontological Information Spaces

Mark J. Weal, Gareth V. Hughes, David E. Millard, Luc Moreau  
IAM Group  
Department of Electronics & Computer Science  
University of Southampton  
SO17 1BJ, UK  
phone +44 023 8059 4059  
{mjw, gvh, dem, l.moreau}@ecs.soton.ac.uk

## ABSTRACT

Ontologies provide a powerful tool for distributed agent-based information systems. However, in their raw form they can be difficult for users to interact with directly. Different query architectures have used structured query languages as an interface but these still require the users to have an expert understanding of the underlying ontologies.

By using an Open Hypermedia model as an interface to an ontological information space, users can interact with such a system using familiar browsing and navigation techniques, which are translated into queries over the underlying information. Coupled with dynamic document generation, this allows complicated queries to be made without the user having to interact directly with the ontologies.

Our key contribution is a notion of hypermedia links between concepts and queries within an ontological information space. This approach is demonstrated with a Dynamic CV application built around the SoFAR agent framework and the Fundamental Open Hypermedia Model (FOHM). In addition to abstracting the interface, Open Hypermedia allows alternative linkbases to be used to represent different “query recipes”, providing different views and navigational experiences to the user.

**KEYWORDS:** Ontological Information Spaces, Fundamental Open Hypermedia Model (FOHM), Agent Based Systems.

## INTRODUCTION

Agent-based processing is particularly well suited to the highly dynamic nature of distributed information management [19, 9, 18, 23]. In this context, ontologies are used

to promote inter-operability in that they represent a shared understanding that agents can use to express the information they exchange [15]. They provide a well-grounded, understandable structure for information, giving a reliable and consistent mechanism for communication. Therefore, agent-based information management systems are designed to manipulate information spaces expressed by ontologies, which we call *ontological information spaces*.

While the rigour and unambiguity of ontologies make them suitable for agent-based processing and reasoning, they can be understood only by experts. When the information is presented to a user of the system, the raw ontologies often prove too complex to be readily understandable. An even more difficult task is finding particular information in the information space. Different query architectures [5, 4, 2] have used structured query languages and ontologies in order to find information from multiple sources. However, they require the same level of expertise from the user. Alternatively, keyword searching, although simpler for the user, results in an imprecise search because it ignores the inherent structure of the information space.

Hypermedia provides users with a well-understood methodology for interacting with information via browsing and navigation. Additionally, *Open Hypermedia* [1, 14, 30, 11, 27] is a paradigm that allows information to be customised and adapted to the user. In this paper, we argue that *Open Hypermedia is the right approach to solve the problem of interacting with an ontological information space*.

In order to investigate the potential of the approach, we have overlaid a hypermedia model on top of an existing agent-based ontological information space to provide a more intuitive and natural interface to interacting with the underlying information. We use hypermedia links that resolve to queries over the information space. In this way, querying the information space becomes a process of link following. Documents are constructed on the fly from the knowledge retrieved from the ontological information space. Furthermore, by using Open Hypermedia, the links, i.e. queries, can

be customised and adapted to the user.

Our key contributions are the following:

1. We introduce the new notion of hypermedia links associating ontological concepts with queries over an ontological information space. During navigation, the traversal of a link resolves to queries over the information space.
2. We extend the Fundamental Open Hypermedia Model (FOHM) [22] to support this form of link between concepts and queries.
3. We adopt an Open Hypermedia approach by which different linkbases (collections of links) represent different “query recipes”, providing different views and navigational experiences to the user.
4. We have prototyped a system called the *Dynamic CV* which is overlaid on an existing information space managed by an agent framework. The engineering of this system allows us to report on the modularity, customisability and separation of concerns offered by our approach.

Let us note that there is a range of activities involving hypermedia and ontologies. In our approach, we are concerned with browsing an existing information space managed by distributed agents. We promote the use of hypermedia to drive the querying of information space through navigation. Other authors investigate the application of ontologies to hypermedia, with a view to improve the quality of navigation. [7, 8, 17, 28, 6, 10]. We will contrast the approaches in detail later in the paper.

This paper is organised as follows. First, we introduce the notion of ontological information spaces, focusing on their particular realisation in the SoFAR distributed agent architecture [23]. We then explain the lack of suitable solutions to interacting with such information spaces and identify Open Hypermedia as the paradigm that would provide us with all the desired characteristics. We detail our reasons for adopting the Fundamental Open Hypermedia Model (FOHM) to express the associations between concepts and queries over the information space. We then present the Dynamic CV application, which illustrates our approach. Finally, we analyse the system, before related research is examined and conclusions are drawn.

## ONTOLOGICAL INFORMATION SPACES

In this section we describe a notion of information spaces based on ontologies and manipulated by agents performing distributed information management tasks. We call these *ontological information spaces*.

### The Notion of Ontology

An ontology can be defined simply as a conceptualisation of a domain into a human understandable and machine readable

format, characterised by the entities, attributes, relationships and axioms of the domain.

In philosophy, the word “ontology” denotes the science or study of being. Closer to our interest, the notion of “ontology” has received a lot of attention in the knowledge engineering community. One of the best known definitions is Tom Gruber’s [15]: *An ontology is an explicit specification of a conceptualisation.*

Pragmatically, an ontology is constituted of a vocabulary describing a certain domain and a set of explicit assumptions regarding the intended meaning of the vocabulary words. We shall see some specific realisations of ontologies later in the paper.

Ontologies are *designed artifacts*, and it is an engineering discipline to craft such ontologies [15]. Guarino and Giaretta [16] even see ontologies as knowledge bases of a special kind which can be read, sold or physically shared. An essential property of ontologies is that they are a *shared understanding* of some domain that can be communicated across people and computers.

From the knowledge engineering point of view, ontologies can be used to *enhance communications* and to *promote inter-operability*.

*Communication Enhancement:* In a specific application domain, users insist on using a normative model composed of a set of terms for communications. Such a specific model expresses a network of relationships which is regarded by the designers and the community as a good communication device.

Additionally, well-defined ontologies are consistent and unambiguous. Natural language is full of ambiguity: for instance, the word “orange” can be an adjective or a noun. A properly designed ontology will distinguish “orange” as a colour from “orange” as a fruit. Furthermore, ontologies are the result of a design involving several people, and therefore integrate several perspectives of a same domain.

*Inter-Operability:* Ontologies are good for promoting inter-operability by enabling reuse of knowledge. Applications can share information expressed in a common ontology and they can understand and reason about it.

## Agent-Based Distributed Information Management

Distributed Information Management (DIM) is the term used to describe the set of activities that allow users to manage the entire life-cycle of information in a distributed environment. The activities, also referred to as *DIM tasks*, involve, amongst others, document creation and publication, information discovery and integrity maintenance.

The large volume of highly dynamic information involved in DIM tasks is an ideal subject for agent-based processing. This has been exemplified in several research projects, such

as Pattie Maes' agents that reduce users' overload [19] or the numerous agents applied to the Internet or the WWW [9, 18]. Our vision is that distributed information management may be regarded as the result of coordinating a multitude of simple DIM tasks performed by agents [23].

In the context of agent-based DIM, ontologies play an essential role:

1. They are the lingua-franca of agents, used to express the contents of messages.
2. They are a common medium, which various information formats can be converted to and from, typically through the use of wrapper agents.
3. They represent an unambiguous shared understanding, which participating agents have all signed up to.
4. They constitute a knowledge that is application independent and can be used for multiple different purposes.

Therefore, agent-based information management systems are designed to manipulate information spaces expressed via ontologies, which we call *ontological information spaces*.

As an illustration of these ideas, we have designed an ontology to characterise the set of activities taking place in our research laboratory. A sample excerpt of this ontology appears in Figure 1, where broad categories are events, roles, publications and projects.

### SoFAR ontologies

SoFAR (Southampton Framework for Agent Research) is an agent framework designed to address the problem of distributed information management. In this section, we explain how ontologies are defined in the framework; the reader interested in other technical details about the framework is referred to [23].

In SoFAR, ontologies are organised along a hierarchy based on single inheritance. Terms of ontologies are defined by the unique parent they extend and a (possibly empty) set of typed fields they contain. A term's definition is expressed in an XML syntax. For instance, a `Person` can be defined as an entity composed of three fields.

```
<term name="Person" extends="Entity">
  <field type="String" name="title"/>
  <field type="String" name="personal"/>
  <field type="String" name="family"/>
</term>
```

It is sometimes convenient to define a term as abstract; such a declaration only defines a type, for which there cannot be any instance. An example of an abstract term is `Entity`, which is extended by two terms `Person` and `Group`.

```
<term name="Entity" extends="Predicate"
  abstract="yes">
</term>
```

```
<term name="Group" extends="Entity">
  <field type="String" name="name"/>
</term>
```

The root of the hierarchy is the type `Term`, which is also abstract. Any concept or relation in a SoFAR ontology is an extension of `Term`. Additionally, we introduce a notion of `Predicate` that is a kind of `Term` we can query about.

```
<term name="Predicate" extends="Term"
  abstract="yes" >
</term>
```

A definition of a term contains a set of fields and their types. Types may be any term declared in an ontology or any of the following primitive types: Integer, Long, Object, String, URL, Byte, Double, Character, Float, Short, Time.

An example of a relation between newly defined types is illustrated below, where a `Person` and a `Group` are associated by an `InGroup` relationship.

```
<term name="InGroup" extends="Predicate">
  <field type="Person" name="person"/>
  <field type="Group" name="group"/>
</term>
```

The ontological definitions allow us to define typed data structures, but also, they provide the foundation of a query language over sets of such data structures. The benefit of this approach is a uniform handling of ontological terms and queries over them. Our query language is based on pattern-matching and requires adding *variables* and *constraints* to ontological definitions.

Let us consider an instance of the `Person` term defined above.

```
Person("Dr", "Luc", "Moreau")
```

If we regard this term as a query to an agent *A*, it has the following meaning: is a `Person` with title `Dr`, first name `Luc` and family name `Moreau` known to the agent *A*? Any of the fields can be replaced by a typed variable. For instance, the following term denotes all the persons with a first name `Luc`.

```
Person(?String, "Luc", ?String)
```

Variables are not restricted to primitive types, but can be used to denote any terms of an ontology. The following query is expected to return all the persons with a first name `Luc` and their associated group.

```
InGroup(Person(?String, "Luc", ?String), ?Group)
```



space, the reader may wonder why such ontologies could not provide the type of hypermedia activity we just described. The reason is as follows. Generally, ontologies are designed to be task independent, so that they can be used in a wide range of applications. The requirements for browsing an information space are by far better fulfilled by hypermedia techniques, which rely on their own domain conceptualisation (based on links, anchors, directionality, etc.). We feel it is essential to use the right tool for every task, and our information space browsing is better handled by hypermedia.

### Summary

Ontologies are good for agents and experts but they are not suitable for end-users. The paradigm of hypermedia navigation is also appropriate for navigating an information space of ontological terms. With this vision, link traversing initiates queries of the ontological space, whose results are formatted as documents presented to the user.

In the next section, we introduce the linking model we have adopted and in a subsequent section show how it can be used to make hypermedia associations between concepts and queries in an ontological information space.

### FOHM

The **Fundamental Open Hypertext Model (FOHM)** [22] grew out of the Open Hypermedia Protocol (OHP) developed by the Open Hypermedia Systems Working Group (OHSWG) [12] but it expands the OHP data model to describe a broader set of hypermedia “domains”. FOHM also makes no assumptions about the protocol it is running over or the systems that are using it. It is a model for describing hypertext structures that requires binding to a syntax before it can be used.

### An Overview of Hypertext Domains

In its work on interoperability, the OHSWG considered the requirements of several domains of hypertext. The three most frequently mentioned were Navigational, Spatial and Taxonomic Hypertext. The OHP protocol was always more concerned with Navigational Hypertext, however FOHM is capable of expressing all three domains. Before we can examine FOHM it is necessary to define these domains.

Navigational Hypertext is the most traditional domain of hypertext, exemplified in Open Hypermedia Systems such as Chimera [1], DHM [14], HyperForm [30], Microcosm [11] and the HB/SP series [27]. Authors create *Links* between parts of documents that are related. Users can then click on those links to move between documents. Although Navigational Hypertext systems can be quite sophisticated, by far the most popular system, the World Wide Web, is also one of the simplest.

Spatial Hypertext systems allow users to organise their information visually in a process known as “Information Triage” [21]. Relationships between nodes are expressed by their visual characteristics such as proximity, colour or

shape. This results in visual collections, or *Spaces*, such as lists and sets. Spatial hypertext systems are therefore ideal for an evolving organisation of data. Examples of such systems include VIKI [20] and CAOS [26].

Taxonomic Hypertext is the organisation of information artifacts into *Categories* [24]. Where authors disagree about the categorisation, the Taxonomy can branch into different *Perspectives* [29]. Applications can allow users to navigate the information space by moving between overlapping Categories and can also reason about the relationships that artifacts have with one another.

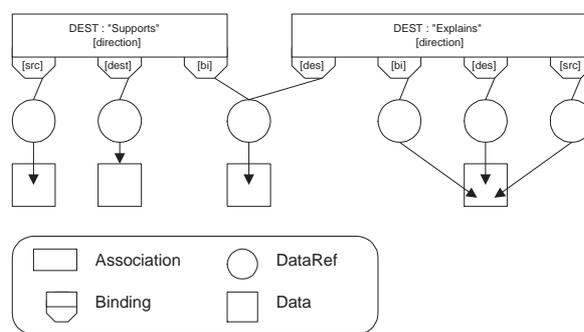
### A Description of FOHM

In FOHM we describe four first-class objects that are analogous to the objects in the OHP data model. *Associations* are structures that represent relationships between *Data* objects. These Data objects are wrappers for any piece of data that lies outside of the scope of the model. They normally represent a document although one could represent any file, stream or other item. It is exactly this feature that we shall use to make links between concepts and queries.

Data objects are not directly placed in the Associations. Instead *Reference* objects are used, these either point at Data objects in their entirety or at parts of those Data objects, for example the second paragraph of a text document, or the second scene of a film. They are attached to the Association object via *Bindings*. Each Association also has a structure type and a feature space; each Binding must state its position in that feature space, effectively stating how it is bound to the Association’s structure.

### Navigational Hypertext in FOHM

Navigational Hypertext is the most common of the domains that can be represented in FOHM. Its notion of directed links can be modeled easily by a single Association feature “direction” to which Data objects are bound with either a “source”, “destination” or “bi-directional” value.



**Figure 2: FOHM Structures: Two Navigational Links**

Figure 2 shows a Navigational structure described within the FOHM model, in this case two links. The first is a link across three different data objects (one of which is referenced in its entirety), the second is a link across one area of one doc-

ument and three different areas within a second document. Notice that Associations can share References and that References can share Data objects.

### Other Domains in FOHM

As each Association object in FOHM has a structure type and feature space, it is possible to create many different kinds of Associations. Just as the Association maps onto Links, it can also represent a Space, such as a List, Map or Matrix. References would then be bound to such a structure using appropriate features such as structural position, colour and even shape.

The Association object can also represent the collections within a Taxonomic hypertext. In this case the structure is always a Set and the relationship type of the Association determines whether it is a Category or Perspective.

### APPLYING FOHM TO ONTOLOGICAL INFORMATION SPACES

We currently have an implementation of FOHM running across the SoFAR agent framework. In the framework, FOHM is itself represented as an ontology which is communicated via the agent infrastructure. The ontology contains a definition for each first-class FOHM object.

#### The FOHM Server

The FOHM Server is a utility agent developed in the SoFAR system. It acts as a persistent storage device, storing FOHM structures created by other agents and serving them up again on demand. To make storage explicit, and to separate it from normal discourse, we defined several storage actions in addition to the basic FOHM definitions.

The FOHM Server uses these principles to store information in memory, with a simple persistence mechanism based on Java serialisation that preserves the data between different executions of the Server.

#### FOHM Links Between Terms and Queries

We showed above how navigational links can be represented in FOHM. As well as applying navigational links to the traditional document to document scenarios, we can also apply these links to the ontological terms themselves, linking from a term that might be obtained from the agent cloud to a query that would result in more information.

Figure 3 shows a link between the 'Person' term and two queries, 'HasPhone' and 'HasEmail'. This link informs the querying application that when it obtains a Person term it may receive extra, relevant information by performing these two queries. FOHM Associations also contain a description of the relationship. Thus the link in Figure 3 could have the description 'Contact details', while another structure that links 'Person' with a query on 'HasPublished' might have the description 'Papers and other publications'.

By changing the links that are served by the FOHM Server,

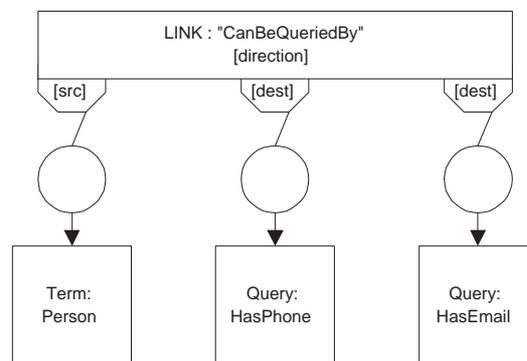


Figure 3: A Link from Term to Queries

effectively changing the linkbase, the calling application can serve up different sets of information and alternative navigable paths through that information.

### THE DYNAMIC CV APPLICATION

We have constructed a prototype application in order to test our ideas of using hypermedia as a navigation technique over ontologies. Over the last two years, we have developed a set of agents that know about our research group's administrative facts, personnel and range of activities. As we populate our system with more and more agents, we are building a complete agent-based distributed information management system reflecting our academic environment.

We conceived the idea of a Dynamic CV, allowing a user to obtain an individual's dynamically constructed Curriculum Vitae (hence the name Dynamic CV). From this, other related information can be accessed, such as publications, research topics, positioning in the research spectrum of the group etc. A typical navigation trail might start from an individual's CV page. Link following can take them to a list of seminars they have given, from which a video recording can be viewed, which in turn may have been annotated with links to relevant papers or other cross references [13].

#### The Information Space

Agents exist within the SoFAR framework which provide access to different types of information about our research group and the people within it. We describe two of the agents used in our applications. One agent supports a suite of ontologies which covers information from our departmental database (ecsinfo), which lists phone numbers, email address etc. for people within the department. Another agent supports an ontology about publications and uses the departmental publications database as its knowledge repository.

All these agents share common ontology fragments. For example, the Person term can be used to build a query to find information about a person's email address or telephone number from the ecsinfo agent. Alternatively, it could be used to establish a person's publications from the publications agent.

## The architecture

Figure 4 illustrates the architecture of the Dynamic CV application. The starting point of browsing is an initial query made by the user from a WWW browser, for example, a request for the CV of an individual. The URL query is translated by a servlet attached to the Web server into a Java Remote Method Invocation (RMI) request to the Dynamic CV Agent.

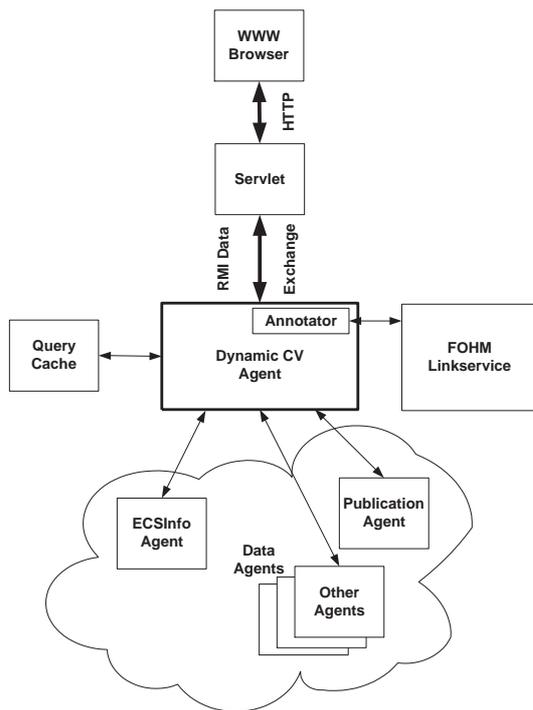


Figure 4: The Dynamic CV architecture

The Dynamic CV agent carries out queries to other agents in the framework to obtain information which it can use to construct a document on the fly. At this stage all the information passed around between agents is expressed in SoFAR ontologies.

Once the Dynamic CV agent has received the information, it passes it to a separate process called the Annotator. This converts the ontological information to an XML representation and uses the FOHM link server to search for links on terms in the information. Retrieved links are used to construct queries which are each given a unique id. These queries are inserted into the XML representation as an annotation that refers to the query id and a description of its purpose. The queries are cached for later use and the annotated document passed back to the Dynamic CV agent.

The annotating process complete, the generated XML is passed back to the servlet via RMI. The servlet transforms the XML into an HTML document using an XSL style sheet. The document is displayed for the user in a standard Web browser.

The conversion to HTML has transformed an annotation for a query in an embedded link containing an encoding of the query id into a URL. When the user follows a link in the document, the servlet extracts the query id, which is passed back to the Dynamic CV agent and the process starts again.

## Linkbases

A number of linkbases were created in the SoFAR FOHM link server. The FOHM associations linked terms to potential queries over the ontologies in the system.

For example, a link was created on the term *Person* to a query concerning the contact information about a person. The two parts of the ontology used in the query are the *HasEmail* and the *HasPhone* predicates.

```
Src    Person(?String,?String,?String)
Dest1  HasEmail(?Person,?Email)
Dest2  HasPhone(?Person,?Phone)
```

In the link stored in the linkbase, the *Person* fields are variables, so the system will match the link to any *Person* term it comes across. Conceptually, each term in the returned information is searched for as a source anchor in the linkbase. Any associations that exist on that term are returned and the destinations used to build further queries.

Because the associations in the linkbase are generic (i.e. on the term rather than an instance of the term) they have to be instantiated before they can be used as queries. This involves taking the destination of the link, which is a query, and filling in the appropriate term with the instance of the term in the generated document.

Using the link described above, the Annotator, on finding a *Person* term, would ask for the links on that term from the linkbase. The link returned to the Annotator would contain the *HasEmail* and *HasPhone* predicates in it. The Annotator would then instantiate the predicates using the value of the *Person* term it was annotating. For example, the term

```
Src'   Person("Dr", "Luc", "Moreau")
```

would be used to instantiate the queries as:

```
Dest1'  HasEmail(Person("Dr", "Luc", "Moreau"),
               ?Email)
Dest2'  HasPhone(Person("Dr", "Luc", "Moreau"),
                ?Phone)
```

The query above would be stored in the cache and a reference to the query would be placed in the generated XML. The caching stage serves a number of purposes, including:

- Minimising the amount of data which needs to be passed to the browser as only the query reference needs to be passed as opposed to the entire query.
- Ensuring that when a link occurs in a number of places on the generated document, the associated query is only stored in full in one place.

- Providing a mechanism for caching the results of queries to speed up the link following process, i.e. if a link is followed on a query that has already been evaluated the cache is able to return the results removing the need to query other agents in the system.

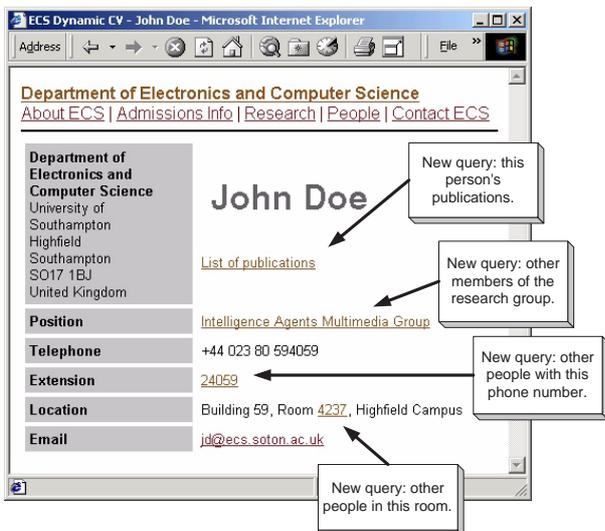
The generic source term of the link may have several occurrences in the query predicates. For example:

`Supervises(?Person, ?Person)`

In this case it is not clear which of the `Person` terms is instantiated by the Annotator. Alternatively, we might like to instantiate some of the fields of the link source, as opposed to the source itself. To overcome these problems, SoFAR ontologies allow us to name variables and perform substitution in a manner very similar to  $\lambda$ -calculus.

### The Interface

The completed XML is sent to a servlet via RMI. The servlet transforms the XML using an XSL stylesheet into an HTML page. This is handled by the Servlet which avoids any need for the browser to support either XSL or XML. The use of XSL stylesheets allows for alternative rendering of the information, providing a further abstraction of the interface.



**Figure 5: Screenshot of a Generated CV**

Figure 5 shows the front end interface to the system with a number of pieces of information on the pages having been augmented with links to queries within the system. The captions are used to highlight the links and indicate the queries to the system that lie behind them.

### Other Hypermedia Views of Ontological Information Spaces

As mentioned previously, FOHM is capable of modelling several different domains of hypermedia. The Dynamic CV application described above uses the Navigational mapping of FOHM to store the queries that should be made when dif-

ferent terms are retrieved. However, Spatial and Taxonomic mappings could also be of use.

Spatial Hypermedia could be used to organise a particular view of an ontological information space visually. Queries and terms could be organised spatially in exactly the same way as documents. This would result in a visual information view that dynamically reflected any changes made to the underlying ontological information space.

Some of the information in the ontological space might be more naturally expressed using set like structures. For example, the people who work on a particular project comprise a subset of the people within a group which in turn form a subset of those people within a department. This hierarchical structure is easily reflected in a hypertext taxonomy, and would also benefit from the alternative perspectives and navigation aids that the Taxonomic domain offers.

### Summary

Because the linkbase model is open, a number of different linkbases can be created and either used in conjunction with each other or separately. In this way, the content of the Dynamic CV can be prescribed by the links stored in the linkbase. For example, a simple linkbase could be used to generate basic contact information or a more complicated linkbase to provide a full academic CV, composed of publications and active projects.

A CV query might require information from a number of different agents in the system. The use of the linkbase model allows complex queries involving a number of Predicates to be stored as a single association, which we call a *query recipe*.

In addition, because links are held separately, processing can be carried out to ensure that links are only provided where the destination query can be resolved by a currently running agent. The separation of links also allows metrics to be run on the linkbase. For example, the linkbase can be analysed to ensure that there are no dead ends or unreachable ontology fragments.

### RELATED WORK

There is an important literature on query architectures, and in particular, on ontology-based architectures able to query multiple information sources. Infosleuth [5] is able to collect information from heterogeneous distributed databases using queries expressed in SQL (possibly generated from a graphical user interface), and then converted into an ontological representation. Tambis [4] has a similar goal but queries are expressed in a description logic, which is also used to express the ontologies of a biomedical application. SIMS [2] uses an alternative description logic language to express queries. While our application shares similar requirements with these systems, we are the only ones to propose Open Hypermedia as the paradigm for browsing information: this allows us to customize and prepare navigation trails for non-expert users.

Others use ontologies to “enrich” the content of documents, by providing information about their content. The benefit of using ontologies for such annotations is the inter-operability they provide (as a shared understanding of a domain) and the possibility of making inferences over them. Such enrichment can subsequently be used to drive the navigation of documents. This idea is in fact very much the notion of “semantic Web” defended by Tim Berners-Lee [8]. For this purpose, a number of annotation languages are being investigated such as RDF [25], OIL [6], DAML [10] and SHOE [28]. Two specific implementations of this approach are COHSE and the ontology portal, which we now describe.

Bechhofer et al’s motto is very similar to ours: “Navigation of links is currently and will remain for humans if not machines, a key mechanism for exploring space” [7]. The key differences between the two approaches are the kind of space and the link generation process. Their purpose is to navigate a document space whereas ours is to browse an ontological information space, produced and managed by DIM agents. Their linking generation process is based on ontologies, whereas ours is manual but aimed at issuing queries over ontologies. In our context, a link base is a carefully crafted set of queries (and associated terms) that present a coherent and customised view of an information space, typically generated by experts for end-users.

Kampa *et al.* [17] describe an ontology portal that helps to navigate literature archives. Ontologies specialised to the domain of scholarship are used to suggest relevant links to the user.

## SUMMARY AND CONCLUSIONS

In this paper we have generalised the Open Hypermedia notion of a generic link, by allowing link sources to be terms within an ontological information space. Some computation also takes place to determine the exact link destination, by querying the ontologies using agents in the system. This is not dissimilar to Ashman’s functional link, where both source and destination anchors are computed [3], though all our computations take place at the ontological level, before document generation.

As hypertext concepts are also expressed in the SoFAR ontological framework, we can apply our technique to navigate a space of hypermedia entities. We can regard this as meta-navigation, which could be useful, for instance, to examine the linkbases available to a user.

A number of conclusions can be drawn from this work:

1. By using hypermedia as an interface to an ontological information space, we have achieved a separation of concerns. At the interface level, users work in a navigation and browsing paradigm while the underlying information system trades in ontologies, allowing the right tools to be used for the right job.
2. As all user interactions are made through a hypertext interface, the user can navigate seamlessly from a document space to an ontological one, moving from real documents to dynamically created ones.
3. The interface also shields the user from potentially complex ontological queries to the agent system.
4. Open Hypermedia allows the linkbase presented to the user to change according to preference or context, suggesting different queries according to the users profile.
5. By using the FOHM model it is possible to support different paradigms of interaction by encoding terms and queries in the structures of different hypertext domains.
6. New queries over the information space can be added to the system with no need for recompilation, simply by adding new links to the linkbase.

Although ontologies provide a powerful tool for distributed agent based information systems, in their raw form they can be difficult for users to interact with directly. By using an Open Hypermedia model on top of an ontological space, we have shown that users can interact with such a system using simple browsing and navigation techniques which are translated onto the ontological information space behind the scenes. Coupled with dynamic document generation, this allows complicated queries to be made of agents and the information space with the results being mapped onto a simple hypermedia interface.

## ACKNOWLEDGEMENTS

The starting point for this work was a three day “agent fest” involving the entire IAM group. Although many people contributed to aspects of the SoFAR framework we would particularly like to mention John Revill, Richard Beales, Stephen Harris and Patrick Sinclair.

## REFERENCES

1. ANDERSON, K. M., TAYLOR, R. N., AND WHITEHEAD, E. J. Chimera: Hypertext for heterogeneous software environments. In *ECHT '94. Proceedings of the ACM European conference on Hypermedia technology, Sept. 18-23, 1994, Edinburgh, Scotland, UK* (1994), pp. 94–197.
2. ARENS, Y., AND KNOBLOCK, W. Query reformulation for dynamic information integration. *Information Integration, Journal of Intelligent Information Systems* 6 (1996).
3. ASHMAN, H., AND VERBYLA, J. Dynamic Link Management via the Functional Model of the Link. In *Proceedings of Basque International Workshop on Information Technology* (Biaritz, France, Feb. 1994).
4. BAKER, P. G., BRASS, A., BECHHOFFER, S., GOBLE, C., PATON, N., AND STEVENS, R. TAMBIS: Transparent access to multiple bioinformatics information sources. In *6th Int. Conf. on Intelligent Systems for Molecular Biology* (Montreal, Canada, 1998), J. Glasgow, T. Littlejohn, F. Major, R. Lathrop, D. Sankoff, and C. Sensen, Eds., AAAI Press, Menlo Park, pp. 25–34.

5. BAYARDO JR., R. J., BOHRER, B., BRICE, R. S., CICHOCKI, A., FOWLER, J., HELAL, A., KASHYAP, V., KSIEZYK, T., MARTIN, G., NODINE, M., RASHID, M., RUSINKIEWICZ, M., SHEA, R., UNNIKRIISHNAN, C., UNRUH, A., AND WOELK, D. InfoSleuth: Semantic integration of information in open and dynamic environments (experience paper). In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data* (Tucson, Arizona, 13–15 June 1997), J. Peckham, Ed., pp. 195–206.
6. BECHHOFFER, S., BROEKSTRA, J., DECKER, S., ERDMANN, M., FENSEL, D., GOBLE, C., VAN HARMELEN, F., HORROCKS, I., KLEIN, M., MCGUINNESS, D., MOTTA, E., PATEL-SCHNEIDER, P., STAAB, S., AND STUDER, R. An informal description of standard OIL and instance OIL. Tech. rep., University of Manchester, 2000.
7. BECHHOFFER, S., GOBLE, C., CARR, L., DE ROURE, D., AND HALL, W. COHSE: An architecture for the semantic web. Tech. rep., University of Southampton, 2001.
8. BERNERS-LEE, T. *Weaving the Web*. Orion business, 1999.
9. CHEN, L., AND SYCARA, K. WebMate: a Personal Agent for Browsing and Searching. In *Proceedings of the Second International Conference on Autonomous Agents* (1998), pp. 132–139.
10. DAML : The DARPA agent markup language homepage. <http://www.daml.org/>, 2000.
11. DAVIS, H. C., KNIGHT, S., AND HALL, W. Light hypermedia link services: A study of third party application integration. In *ECHE '94. Proceedings of the ACM European conference on Hypermedia technology, Sept. 18-23, 1994, Edinburgh, Scotland, UK* (1994), pp. 41–50.
12. DAVIS, H. C., MILLARD, D. E., REICH, S., BOUVIN, N., GRØNBÆK, K., NÜRNBERG, P. J., SLOTH, L., WIIL, U. K., AND ANDERSON, K. M. Interoperability between hypermedia systems: The standardisation work of the ohswg. In *Hypertext '99, The 10th ACM Conference on Hypertext and Hypermedia, Darmstadt, February 21-25, 1999* (Feb. 1999), ACM, pp. 201–202.
13. DON CRUICKSHANK AND LUC MOREAU AND DAVID DE ROURE. Architectural Design of a Multi-Agent System for Handling Metadata Streams. In *The fifth ACM International Conference on Autonomous Agents* (Montreal, Canada, May 2001).
14. GRØNBÆK, K., AND TRIGG, R. H. Design issues for a dexter-based hypermedia system. *Communications of the ACM* 37, 3 (Feb. 1994), 40–49.
15. GRUBER, T. R. Toward principles for the design of ontologies used for knowledge sharing. Tech. Rep. KSL-93-04, Knowledge Systems Laboratory, Stanford University, Aug. 1993.
16. GUARINO, N., AND GIARETTA, P. Ontologies and knowledge bases: Towards a terminological clarification. In *Towards Very Large Knowledge Bases*, N. Mars, Ed. IOS Press, 1995.
17. KAMPA, S., MILES-BOARD, T., CARR, L., AND HALL, W. Linking with meaning: Ontological hypertext for scholars. Tech. rep., University of Southampton, 2001.
18. LIEBERMAN, H. Letizia: An agent that assists web browsing. In *Proceedings of the International Joint Conference on Artificial Intelligence* (Montreal, Canada, Aug. 1995).
19. MAES, P. Agents that Reduce Work and Information Overload. *Communications of the ACM* 37, 7 (July 1994), 31–40.
20. MARSHALL, C. C., AND SHIPMAN, F. M. Spatial hypertext: Designing for change. *Communications of the ACM* 38 (1995), 88–97.
21. MARSHALL, C. C., AND SHIPMAN, F. M. Spatial hypertext and the practice of information triage. In *Proceedings of the '97 ACM Conference on Hypertext, April 6-11, 1997, Southampton, UK* (1997), pp. 124–133.
22. MILLARD, D. E., MOREAU, L., DAVIS, H. C., AND REICH, S. FOHM: A Fundamental Open Hypertext Model for Investigating Interoperability Between Hypertext Domains. In *Proceedings of the '00 ACM Conference on Hypertext, May 30 - June 3, San Antonio, TX* (2000), pp. 93–102.
23. MOREAU, L., GIBBINS, N., DEROURE, D., EL-BELTAGY, S., HALL, W., HUGHES, G., JOYCE, D., KIM, S., MICHAELIDES, D., MILLARD, D., REICH, S., TANSLEY, R., AND WEAL, M. SoFAR with DIM Agents: An Agent Framework for Distributed Information Management. In *The Fifth International Conference and Exhibition on The Practical Application of Intelligent Agents and Multi-Agents* (Manchester, UK, Apr. 2000), pp. 369–388.
24. NÜRNBERG, P. J., SCHNEIDER, E. R., AND LEGGETT, J. J. Designing digital libraries for the hyper-literate age. *Journal of Universal Computer Scienc* 2, 9 (1996).
25. RDF : Resource description framework. <http://www.w3c.org/rdf/>.
26. REINERT, O., BUCKA-LASSEN, D., PEDERSEN, C. A., AND NÜRNBERG, P. J. CAOS: A collaborative and open spatial structure service component with incremental spatial parsing. In *Proceedings of the '99 ACM Conference on Hypertext, February 21-25, 1999, Darmstadt, Germany* (Feb. 1999), pp. 49–50.
27. SCHNASE, J. L., LEGGETT, J. L., HICKS, D. L., NUERNBERG, P. J., AND SÁNCHEZ, J. A. Design and implementation of the HB1 hyperbase management system. *Electronic Publishing—Origination Dissemination and Design* 6, 1 (June 1993), 35–63.
28. The SHOE project homepage. <http://www.cs.umd.edu/projects/plus/SHOE/>.
29. VAN DYKE PARUNAK, H. Don't link me in: Set-based hypermedia for taxonomic reasoning. In *Proceedings of the '91 ACM Conference on Hypertext, Dec. 15-18, 1991, San Antonio, TX* (1991), pp. 233–242.
30. WIIL, U. K., AND LEGGETT, J. J. HyperForm: using extensibility to develop dynamic, open and distributed hypertext systems. In *ECHE '92. Proceedings of the ACM conference on Hypertext, November 30-December 4, 1992, Milan, Italy* (1992), pp. 251–261.