MetaPortal Final Report: Building Ontological Hypermedia with the OntoPortal Framework

Leslie Carr, Simon Kampa, Timothy Miles-Board

May 18, 2001

Contents

1 Introduction

This report details the OntoPortal project conducted for the Defence Evaluation and Research Agency (DERA), UK. The specification called for a well-interlinked and incrementally updateable web site detailing the latest research in metadata. The type of information to integrate included information on literature, projects, researchers, organisations, software and standards.

Due to the implicit relations evident between these resources (e.g. a research *works* on a project, literature *discusses* a project) an *ontological hypermedia* [10] philosophy was adopted, providing a principled and structured approach to navigation within the research portal.

The generic OntoPortal framework was developed to realise this methodology, and was used to create the MetaPortal web site for this project. The remaining report details both the features and technical details of framework and its MetaPortal application.

2 Principles

We investigated the use of ontologies to improve the linking of research literature together with the Web sites and home pages of related projects, institutions and individual researchers, by providing a principled way of describing both the topic under discussion and the process by which it was produced, in order to allow other researchers to better understand the work.

Ontology is the study of "things that exist", a formal model that allows reasoning over concepts and objects that appear in the real world.

Hypermedia is the study of "what can be said" using computer media, databases and links. Hypermedia provides computer mediated extensions to familiar textual communication. This is important because real-world objects, or the "things that exist", have complex relationships, and so complex structures are required for expressing and exploring these relations when we make hypermedia statements about them.

The design goal of an ontological hypertext system is then to produce a methodology for a building a hypertext system to improve the navigation facilities available to users through links that reflect real-world relationships rather than structural/hierarchical relationships. We wanted to take advantage of the features of both hypermedia and the Semantic Web [9] that are not in widespread evidence on the WWW (such as large-scale associative linking, and the annotation of resources with metadata), and produce a semantic hyper-web of research information that encapsulates the knowledge required to become thoroughly immersed in a research field (in this case, the metadata research field). By modelling the domain of a research field using an ontology, a hypertext system has the power to intelligently communicate, analyse and reason over the knowledge.

Our design further specifies that rather than keeping this underlying ontological data model hidden from the user, we promote it to the forefront of the interface for exploring the research field that it encapsulates. Figure 1 illustrates how the user is always aware of their location within the ontology and how the currently viewed concept is related to the rest of the community. On the left in Figure 1, the user is currently viewing literature (indicated by the reverse-video icon labelled Lit). Its relation to the rest of the community is indicated by highlighting the related nodes. As the user moves through the ontology, the diagram changes to reflect the new context (Figure 1, right).

Using the ontology in the interface layer is possible as we have ensured that it is relatively uncomplicated, focused and intuitive. Using an ontology as a navigation tool has many advantages, as both typed links and a fixed linking structure are enforced. Named relationships between ontological concepts

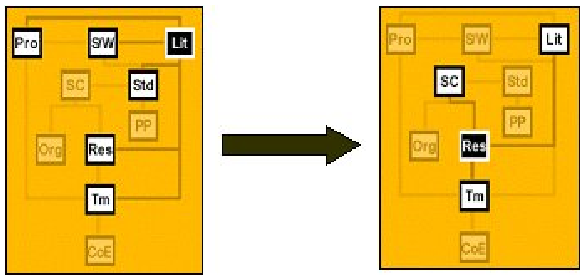


Figure 1: Using the MetaPortal ontology as a navigational cue

naturally provide a link taxonomy from which the interface can derive presentation techniques for displaying different link types. These relationships also restrict the range of permitted links between different concepts. Using the underlying ontological model as a navigational metaphor also enforces bi-directional linking between related concepts, and n-ary links in the case that the ontological model contains a one-to-many relationship.

Ontological navigation also allows users to effectively answer queries using a "query-by-linking" approach (rather than the more traditional "query-by-searching" approach), using facts that they are able to assert in order to discover new facts through exploration. For example, a user having just read a research paper describing a particular standard wishes to find out whether any other papers describe the standard, perhaps with contrasting viewpoints. Using WWW and Digital Library technology as it stands, resolving this type of query usually involves resorting to searching for similar papers using (possibly) several search engines ("query by searching"). However, using ontological navigation, the researcher can realize this query quickly and effectively by exploring the *now* explicit link between all research papers that describe the standard ("query by linking").

3 OntoPortal version 1

The objective of the first OntoPortal system was to realise the principles of ontological hypertext and demonstrate their benefit in a real-world scenario. We derived an ontology to describe the domain of metadata research that was both comprehensive enough to capture the field, yet simple enough to be quickly grasped by users and employed as an interface tool.

The ontology was constructed through discussions with various researchers and from experience in work within the IAM Research Group. The focal elements of interest to researchers, as well as the relationships between them, were identified and formally visualised (Figure 2).

Each node in Figure 2 represents a *class*, a tangible "real-world" concept with which researchers are familiar. Multiple relations exist between the classes.

Knowledge about the metadata research domain is captured using a simple HTML forms interface. Each form provides fields for authors to enter the properties of an ontological instance (e.g. title, description). Authors can describe relations between class instances by indicating which other class instances are associated. OntoPortal uses the data entered to create suitable XML fragments and stores each instance in a unique file.

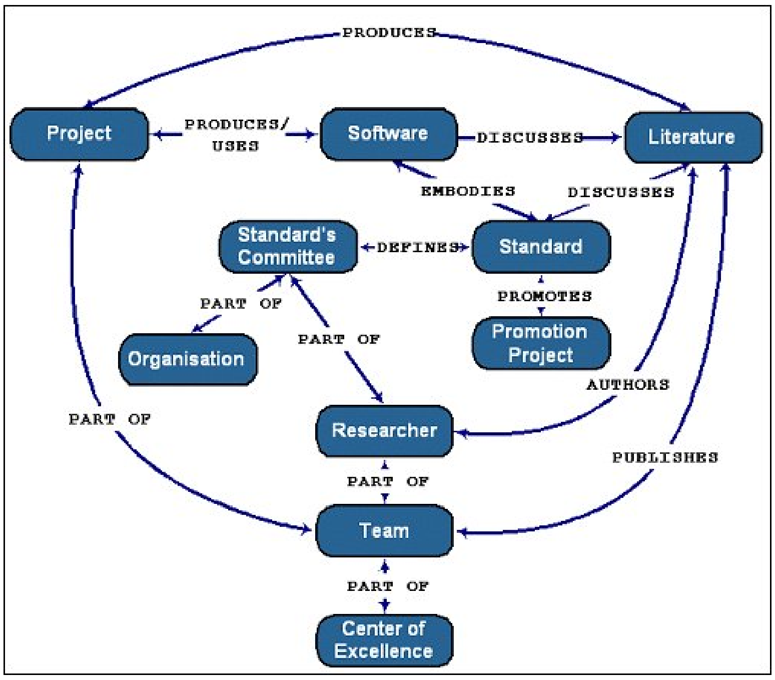


Figure 2: The MetaPortal Ontology

Although XML is by no means entirely suitable for describing ontological instances (RDF Schema and OIL being better candidates), it is expressive enough to describe our simple ontology. Furthermore, the abundance of publicly available XML tools, and the possibility of using XSLT to convert the XML records into a suitable presentation form (e.g. HTML), convinced us to adopt XML.

A research area can be comprehensive and difficult to dissect if presented in one collection. Therefore, we decided to allow the creation of subject areas (or themes) within any topic (e.g. Introduction to Metadata or Semantic Web and Metadata). It is within each theme that ontological hypertext principles are applied to its constituent instances.

To capture the experience of our theme populators, simple peer level commentary facilities were added, enabling authors to add an opinion (their personal thoughts on that class) and an analysis (an examination of that instance).

Screen shots of this version of OntoPortal are displayed in appendix A.

OntoPortal version 1.0 was relatively simple and was produced more to investigate the methodology than to populate a substantial database of knowledge about metadata research. Consequently there were several inherent problems. As the file system was being used to store the classes, file system limitations caused scalability problems. Provisions for handling duplicate classes in different themes also proved difficult. Forcing the client side to deal with the XSLT processing not only slowed down the display of pages, but also alienated users of non-XSLT enabled Web browsers.

The next section describes how OntoPortal version 2 builds on the theoretical foundations layed out in OntoPortal version 1 and overcomes the inherent problems.

4 OntoPortal version 2

After the implementation of OntoPortal version 1, it was clear that the next major implementation stage, OntoPortal version 2 (hereon refered to as "OntoPortal2") needed to be able to address the problems of version 1: namely scalability, data integrity and overall performance, as well as providing support for evolving ontological needs (this requirement eventually led to the implementation of OntoPortal2 as a generic portal framework on which portal applications could be built).

4.1 Overview of features

This section briefly discusses the major features of OntoPortal2, and provides pointers to implementation details in Section 5.

The migration of the data gathered using the OntoPortal version 1 system to a scalable database-based architecture (see Section 5.1) solves many of the problems inherent in the version 1 data storage model (see Section 5.2). Perhaps most importantly, resources are able to be represented in different contexts across many themes without duplicating the underlying data describing the resource. When entering information about resources, theme populators are able to "import" a resource description from another theme, and describe its relationship and commentary relevant to the theme's context.

An important feature of OntoPortal2 is it that it is implemented as a generic framework, onto which an application can be built according to the ontological needs of the domain to be modelled (see Section 5.3). MetaPortal is just such an application - it uses the OntoPortal2 framework to model the domain of metadata research by providing an ontological model of the concepts and relationships in the domain, and presentation rules for the display of the capture information about the domain.

Future applications of the OntoPortal2 framework include the ontological modelling and knowledge capture of the resources surrounding the members of the IAM Research Group at the University of Southampton, UK.

As well as knowledge capturing facilities, the OntoPortal2 framework also provides discussion (Section 5.4) and search (Section 5.5) facilities.

The dynamic data retrieval and presentation architecture of OntoPortal2 applications makes it difficult to distribute the captured knowledge to offline users. The OntoPortal framework therefore also includes a facility for generating a static HTML "snapshot" of the domain being modelled (see Section 5.6).

5 Implementing OntoPortal2 - technical tour

5.1 Architectural overview

This section examines the architecture of the OntoPortal2 system, describing the features illustrated in the architectural overview diagram (Figure 3).

The user interacts with the OntoPortal2 system through a WWW browser, placing no constraints on the browser other than it must support JavaScript in order to take advantage of the ontological navigation tool. The user is able to interact with the system through four discrete "interfaces" (CGI scripts implemented in the perl scripting language) - *Explore*, *Update*, *Search*, and *Discuss*.

The *Explore* interface allows the user to browse the ontologically-linked resources in the OntoPortal2 database. The user can also supply username/password credentials through the Explore interface, which results in the resources being browsed being decorated with additional links into the *Update* interface. Following these links allows the currently displayed resource to be edited, or an entirely new resource to be created. Appropriate username/password credentials (or an authentication cookie from a previous successful login) also result in the Explore interface providing links from each resource into the *Discuss* interface, allowing the user to browse and participate in any threaded discussions surrounding the resource (see Section 5.4 for a more detailed explanation of the Discuss interface).

The Explore interface also contains an entry point to the *Search* interface, allowing the user to query the resources stored in the OntoPortal2 database for specified terms (see Section 21 for a more detailed explanation of the Search interface).

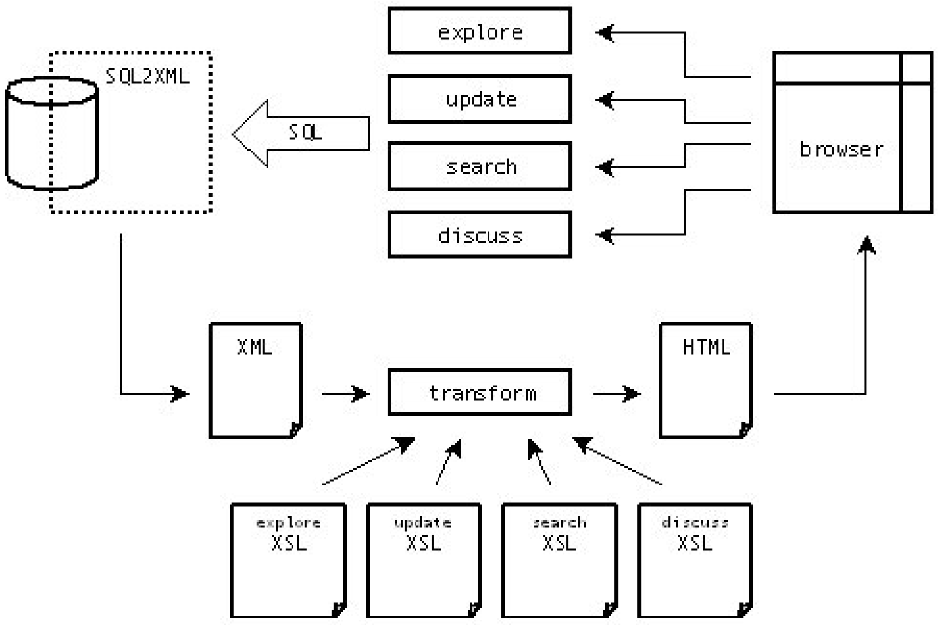


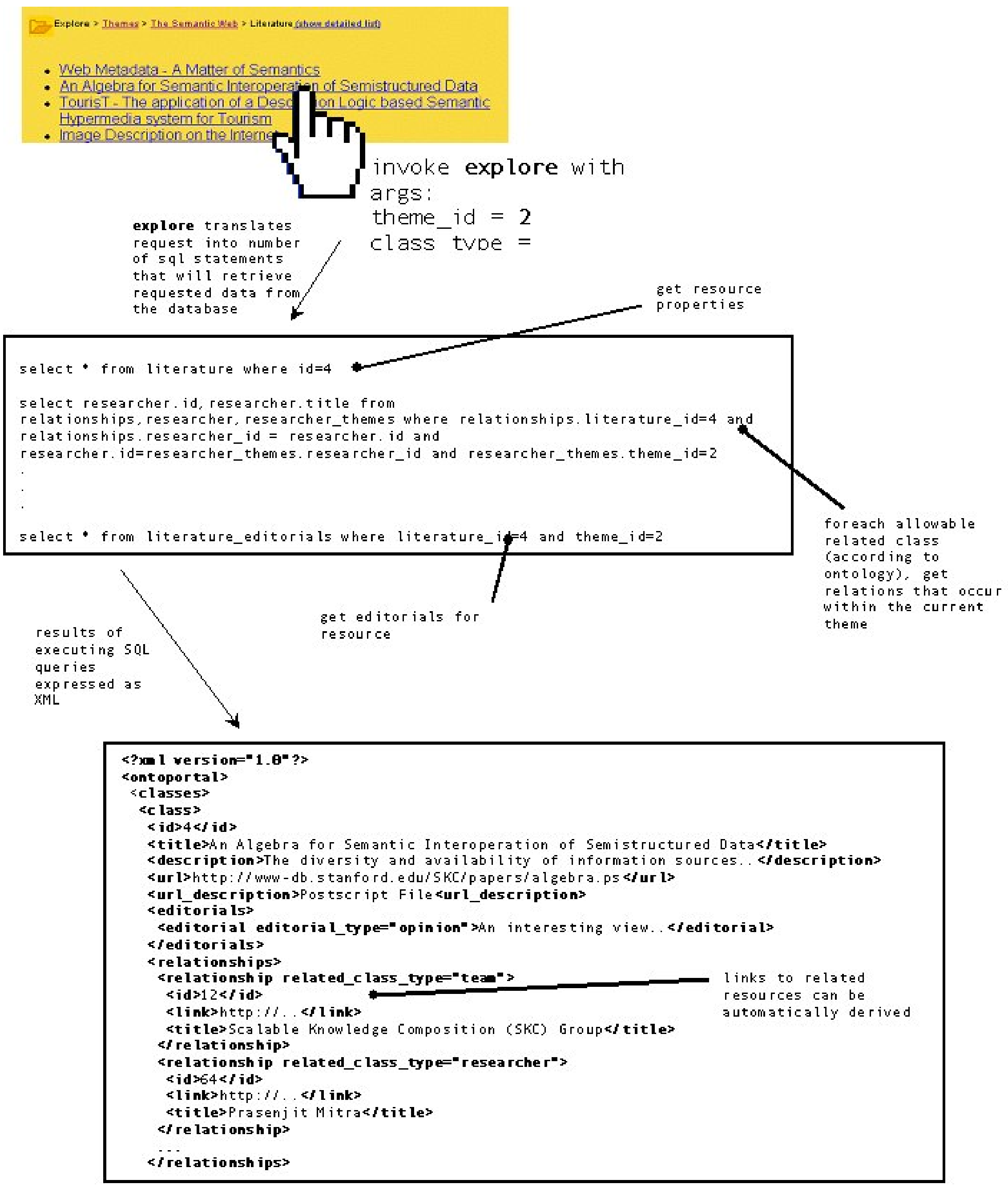
Figure 3: OntoPortal2 architectural overview.

All four interfaces carry out user requests by translating the request parameters into a number of SQL statements, which are then executed across the database using the SQL2XML module. This module was specially developed to wrap the results of SQL queries in an XML format. The specifics of this conversion are controlled by arguments passed to the SQL2XML module. All communication with the database is carried out through the Database Independant (DBI) standard. This means that although the OntoPortal2 system currently uses the MySQL database [4], any database can be substituted with minimal effort.

The resulting XML document is then transformed using an appropriate stylesheet into an HTML document which is sent back to the user's browser. Figures 4 and 5 demonstrate this process. XSLT transformations are carried out using the Microsoft XSL transformer (MSXSL3) [3], chosen for its impressive throughput capabilities. Since this transformer is invoked through the Microsoft Windows dependant COM interface, provisions have been made to allow other XSLT transformers to be "slotted in" in its place (for example, a slot-in replacement has been implemented for the Xalan XSLT transformer [5], to allow transformations to be carried out under the Linux environment).

The OntoPortal2 architecture is advantageous in that it clearly separates the data held in the database from its presentation. Combined with the "generic ontology" feature (see Section 5.3), this results in a generic application framework, which can be tailored to a particular application (for example, MetaPortal) by defining an ontology, and (XSLT) presentation rules specific to that application.

The architecture also allows a user centric, adaptive interface to be constructed. Figure 6 shows how the same XML data is transformed to its presentation format according to different user parameters. Users with no authentication (for example, casual users) are presented with limited interactivity. Registered users are presented with an augmented interface and may take part in threaded discussions. Finally, editors are presented with the highest level of interactivity, and may actually alter the content of the database.


Explore > Themes > The Semantic Web > Literature (show detailed list)
Web Metadata - A Matter of Semantics
An Algebra for Semantic Interoperation of Semistructured Data
TourisT - The application of a Description Logic based Semantic Hypermedia system for Tourism
Image Description on the Internet
invoke **explore** with args: theme_id = 2 class type =
explore translates request into number of sql statements that will retrieve requested data from the database
get resource properties
foreach allowable related class (according to ontology), get relations that occur within the current theme
get editorials for resource
results of executing SQL queries expressed as XML
links to related resources can be automatically derived

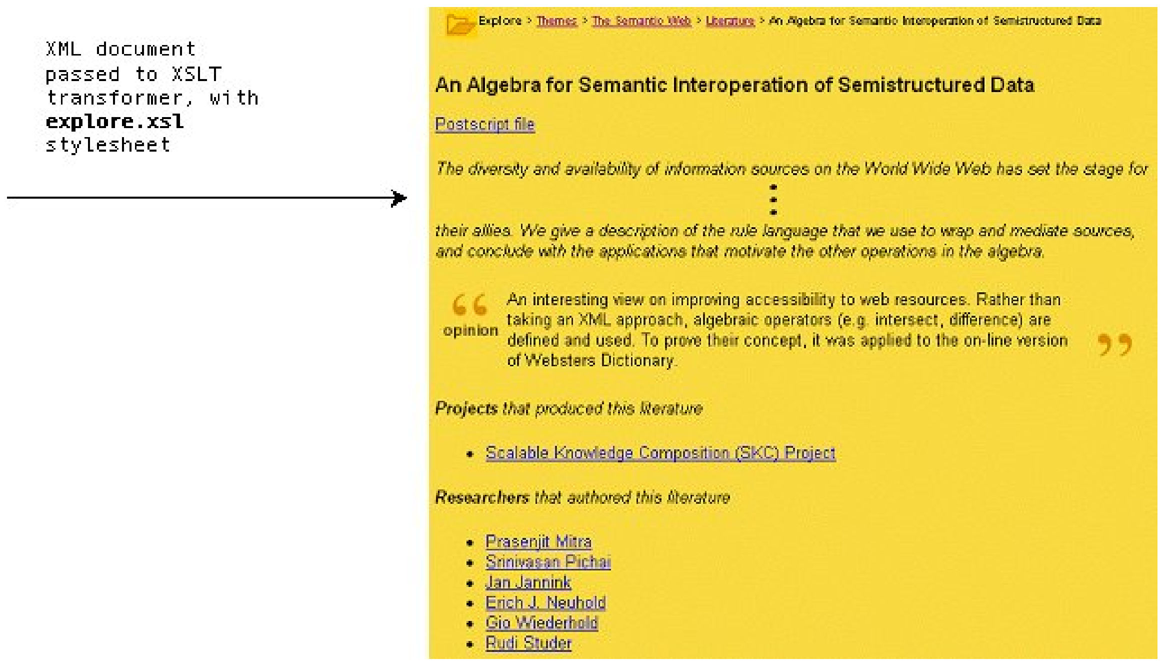
```
select * from literature where id=4

select researcher.id, researcher.title from
relationships, researcher, researcher_themes where relationships.literature_id=4 and
relationships.researcher_id = researcher.id and
researcher.id=researcher_themes.researcher_id and researcher_themes.theme_id=2
.
.
.

select * from literature_editorials where literature_id=4 and theme_id=2
```

```
<?xml version="1.0"?>
<ontoportal>
 <classes>
  <class>
   <id>4</id>
   <title>An Algebra for Semantic Interoperation of Semistructured Data</title>
   <description>The diversity and availability of information sources..</description>
   <url>http://www-db.stanford.edu/SKC/papers/algebra.ps</url>
   <url_description>Postscript File<url_description>
   <editorials>
    <editorial editorial_type="opinion">An interesting view..</editorial>
   </editorials>
   <relationships>
    <relationship related_class_type="team">
     <id>12</id>
     <link>http://..</link>
     <title>Scalable Knowledge Composition (SKC) Group</title>
    </relationship>
    <relationship related_class_type="researcher">
     <id>64</id>
     <link>http://..</link>
     <title>Prasenjit Mitra</title>
    </relationship>
    ...
   </relationships>
```

Figure 4: The user makes a request through the Explore interface, which is translated into a series of SQL queries. These queries are executed across the database, resulting in an XML document describing the requested data. This XML document is then tranformed into a presentation format (Figure 5).



Explore > Themes > The Semantic Web > Literature > An Algebra for Semantic Interoperation of Semistructured Data

An Algebra for Semantic Interoperation of Semistructured Data

Postscript file

The diversity and availability of information sources on the World Wide Web has set the stage for

⋮

their allies. We give a description of the rule language that we use to wrap and mediate sources, and conclude with the applications that motivate the other operations in the algebra.

“ opinion An interesting view on improving accessibility to web resources. Rather than taking an XML approach, algebraic operators (e.g. intersect, difference) are defined and used. To prove their concept, it was applied to the on-line version of Websters Dictionary. ”

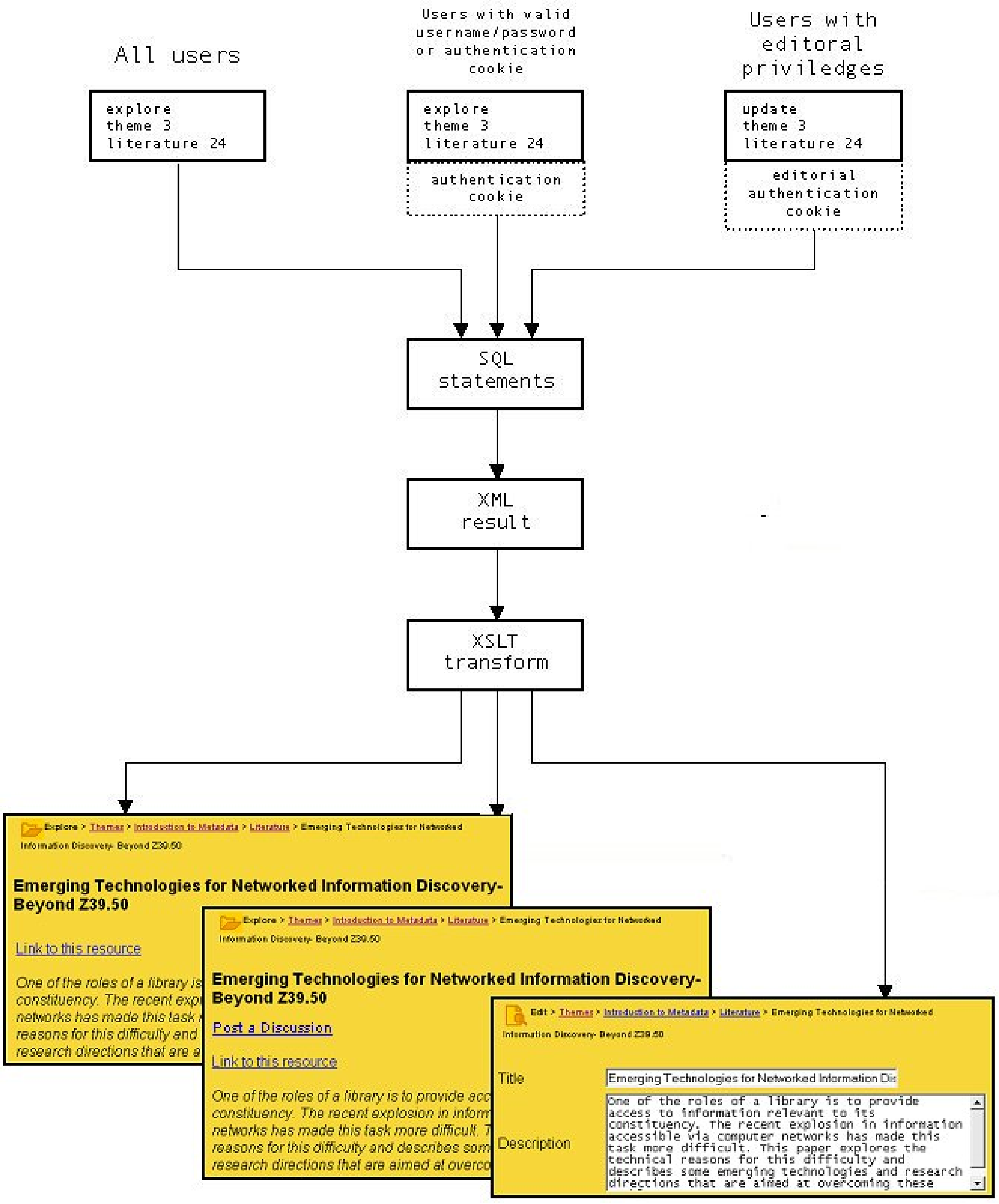
***Projects** that produced this literature*

- Scalable Knowledge Composition (SKC) Project

***Researchers** that authored this literature*

- Prasenjit Mitra
- Srinivasan Pichai
- Jan Jannink
- Erich J. Neuhold
- Gio Wiederhold
- Rudi Studer

Figure 5: The XML document resulting from user interaction with the Explore interface 4 is transformed into a HTML presentation format using an XSLT stylesheet.


All users
explore theme 3 literature 24
Users with valid username/password or authentication cookie
explore theme 3 literature 24
authentication cookie
Users with editoral priviledges
update theme 3 literature 24
editorial authentication cookie
SQL statements
XML result
XSLT transform

Explore > Themes > Introduction to Metadata > Literature > Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Link to this resource

One of the roles of a library is constituency. The recent exp networks has made this task reasons for this difficulty and research directions that are a

Explore > Themes > Introduction to Metadata > Literature > Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Post a Discussion

Link to this resource

One of the roles of a library is to provide acc constituency. The recent explosion in inform networks has made this task more difficult. T reasons for this difficulty and describes som research directions that are aimed at overco

Edit > Themes > Introduction to Metadata > Literature > Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Title: Emerging Technologies for Networked Information Dis

Description: One of the roles of a library is to provide access to information relevant to its constituency. The recent explosion in information accessible via computer networks has made this task more difficult. This paper explores the technical reasons for this difficulty and describes some emerging technologies and research directions that are aimed at overcoming these

Figure 6: The same XML data is transformed in a user-centric fashion according to the user profile.

5.2 Migrating from OntoPortal version 1 to version 2

We have already seen that the data storage mechanisms used in OntoPortal versions 1 and 2 are significantly different. This section briefly outlines the migration process which was designed and carried out in order to populate the version 2 database with the data already harvested and stored in the version 1 data file hierarchy.

The data file hierarchy of OntoPortal version 1 (Figure 7), consists of a directory for each theme (derived directly from the name of the theme), and contains subdirectories for each resource type defined in the ontology. Within each of these resource subdirectories, a data file is stored for each resource record, with the name of the resource used as the filename of the data file (The name of the resource also serves as the unique identifier for that resource). The data files describing the themes themselves are stored in the top level directory of the hierarchy, again with the name of the theme dictating the filename under which the theme data files are stored.

The use of resource names as unique identifiers is demonstrated in Figure 8, where relationships between the resource titled "Review of Metadata Formats" are identified by recording the names of the related resources. Following this storage model, each of the data files describing the resources "Dublin

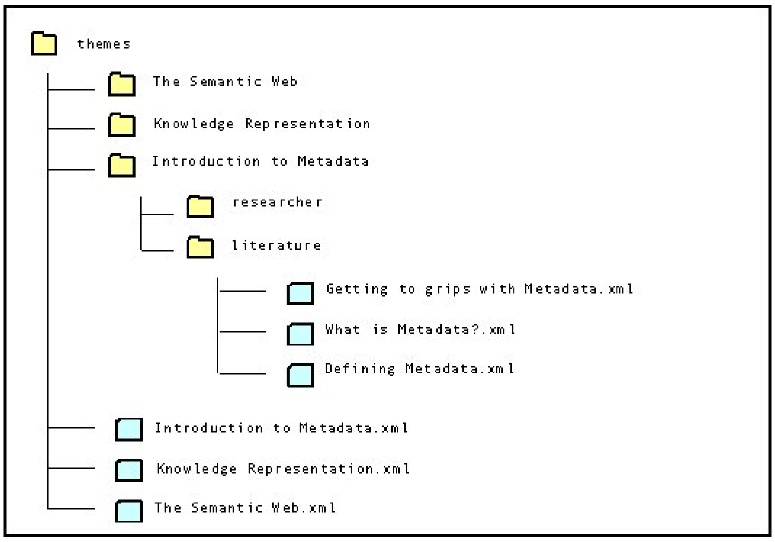


Figure 7: The data file hierarchy of OntoPortal2 version 1. In this example, the system stores information about resources classified into three themes, *The Semantic Web*, *Knowledge Representation*, and *Introduction to Metadata*. Information describing each of these themes is stored under the appropriately named data file at the top level of the hierarchy (in a well-defined XML format). A directory is also created for each theme, which contains a subdirectory for each resource type (*Introduction to Metadata* theme directory shown expanded); in this case the defined resource types are *researcher* and *literature*. Records of resources corresponding to these resource types are stored in well-defined XML data files within these resource subdirectories (*literature* shown expanded); in this case three (literature) resources relevant to this theme have been described - *Getting to grips with Metadata*, *What is Metadata?*, and *Defining Metadata*.

Core", "Rachael Heery", and "ROADS team" will contain a reference to the "Review of Metadata Formats" resource in their `relations` construct.

```
<literature>
    <title>Review of Metadata Formats</title>
    <abstract>Increasing use of the Internet...</abstract>
    <url>http://...</url>
    <url_desc>PDF version</url_desc>
    <literature_relations>
        <standard>
            <title>Dublin Core</title>
        </standard>
        <researcher>
            <title>Rachel Heery</title>
        </researcher>
        <team>
            <title>ROADS team</title>
        </team>
    </literature_relations>
</literature>
```

Figure 8: Example of XML data file format used to describe themes and resources.

5.2.1 Problems with the version 1 data storage approach

A number of problems arising from the nature of the data file hierarchy have become apparent during the development of OntoPortal version 1. The use of filenames as unique identifiers for resources was found to be particularly problematic - since every operating system prohibits certain characters appearing in filenames, a platform-dependant encoding/decoding process had to be applied to resolve the filename of the file containing the data from the unique identifier for a particular resource and vice versa. Furthermore, the restriction of filename length in many operating systems also had to be taken into consideration, meaning that filename truncation and reconstruction also had to be dealt with by the encoding/decoding process in the case that the name of a resource (and thus its unique identifier) exceeded the character limit. The truncation of filenames may also potentially lead to clashes, where more than one “unique” identifier is resolved to the same physical data file.

Furthermore, the need to sequentially access and process the XML data files in the hierarchy limited the scalability of the system - a distinct degradation in access speed was noticeable as the number of resources stored in the system increased.

The strict subdivision of resources into directories corresponding to themes also proved unwieldy, particularly if a resource was appropriate to more than one theme. In this instance, a duplicate record had to be made in each relevant theme subtree. Since the system provided no warning if an author added a resource that had already been described as part of another theme, duplicate resources were often found to contain different data compared to descriptions of the same resource in other themes.

Figure 8 has highlighted the method used to describe relationships between the resources, where a record of the existence of the relationship is stored by each resource participating in the relationship. We found that this approach led to higher maintenance, particularly in the avoidance of “dangling relationships” when relationships or resources are deleted.

5.2.2 Applying a database storage model to version 2

In the context of the data file hierarchy of version 1, we reasoned that problems with using resource names as unique identifiers could be resolved by assigning unique id values to resources and using these id values as filenames. This would also solve the problem of long resource names having to be truncated in order to derive a filename for the corresponding data file.

Rather than being held by each participating resource, relationships between resources could be managed separately to resource data. For example, a single file that contains all resource relationship information could be used. However, accessing this potentially enormous file each time resource relationships needed to be determined would result in a performance bottleneck.

Finally, duplicate resource information could be avoided by using symbolic links files in other theme subtrees. However, not all environments (e.g. Microsoft Windows) support this approach.

It was observed that the overhead required to manage resources in this way could be reduced considerably by using a relational database system. The assignment of unique ids to records is managed automatically by the Database Management System (DBMS). SQL queries are used to quickly and efficiently retrieve information about resources. Information about relationships between resources is stored separately from the resource data (i.e. in a separate table), and accessed efficiently using SQL queries. The duplication of information is also explicitly avoided through the data normalisation process that is an important stage of database design.

5.2.3 Data migration strategy

The migration of data from the data file hierarchy storage model of version 1 to the relational database storage model of version 2 was carried out as follows:

1. Construct database according to existing ontology used to define resource types and properties in version 1 (see Section 5.3 for further details).

2. Transform all XML resource data files into single CSV (comma separated value) files, using XSLT stylesheet. A single CSV file should be created for each type of resource.

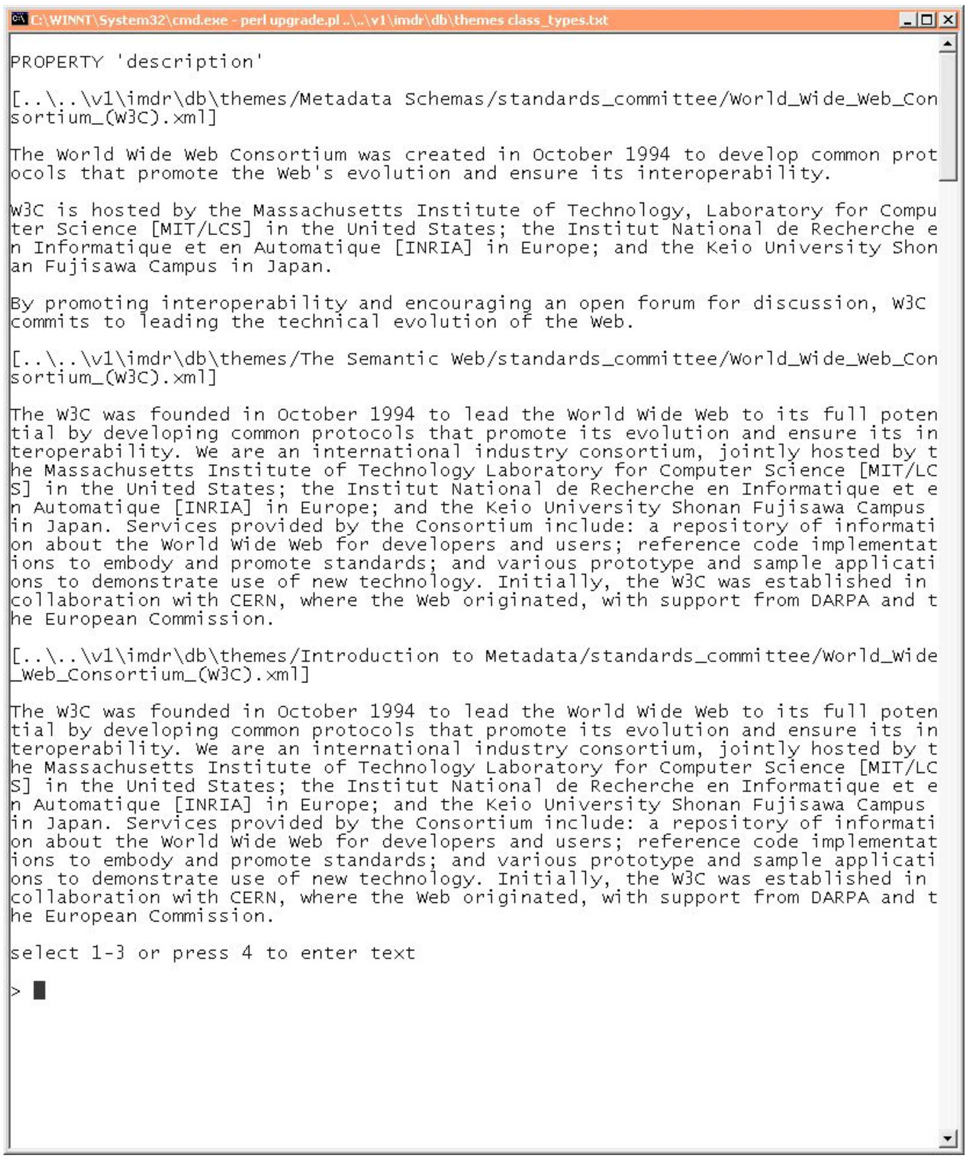
3. Identify, and manually resolve duplicates during the transformation process (see Figure 9).

4. Assign integer id values to all resources and themes, such that a (theme_id,class_type,class_id

key uniquely identifies a resource (i.e. the id value is unique within a particular resource type).

5. Import CSV files into database.

6. Data now available to scripts which access database.


C:\WINNT\System32\cmd.exe - perl upgrade.pl ..\..\v1\imdr\db\themes class_types.txt

```
PROPERTY 'description'

[..\..\v1\imdr\db\themes/Metadata Schemas/standards_committee/World_Wide_Web_Con
sortium_(W3C).xml]

The World Wide Web Consortium was created in October 1994 to develop common prot
ocols that promote the Web's evolution and ensure its interoperability.

W3C is hosted by the Massachusetts Institute of Technology, Laboratory for Compu
ter Science [MIT/LCS] in the United States; the Institut National de Recherche e
n Informatique et en Automatique [INRIA] in Europe; and the Keio University Shon
an Fujisawa Campus in Japan.

By promoting interoperability and encouraging an open forum for discussion, W3C
commits to leading the technical evolution of the Web.

[..\..\v1\imdr\db\themes/The Semantic Web/standards_committee/World_Wide_Web_Con
sortium_(W3C).xml]

The W3C was founded in October 1994 to lead the World Wide Web to its full poten
tial by developing common protocols that promote its evolution and ensure its in
teroperability. We are an international industry consortium, jointly hosted by t
he Massachusetts Institute of Technology Laboratory for Computer Science [MIT/LC
S] in the United States; the Institut National de Recherche en Informatique et e
n Automatique [INRIA] in Europe; and the Keio University Shonan Fujisawa Campus
in Japan. Services provided by the Consortium include: a repository of informati
on about the World Wide Web for developers and users; reference code implementat
ions to embody and promote standards; and various prototype and sample applicati
ons to demonstrate use of new technology. Initially, the W3C was established in
collaboration with CERN, where the Web originated, with support from DARPA and t
he European Commission.

[..\..\v1\imdr\db\themes/Introduction to Metadata/standards_committee/World_Wide
_Web_Consortium_(W3C).xml]

The W3C was founded in October 1994 to lead the World Wide Web to its full poten
tial by developing common protocols that promote its evolution and ensure its in
teroperability. We are an international industry consortium, jointly hosted by t
he Massachusetts Institute of Technology Laboratory for Computer Science [MIT/LC
S] in the United States; the Institut National de Recherche en Informatique et e
n Automatique [INRIA] in Europe; and the Keio University Shonan Fujisawa Campus
in Japan. Services provided by the Consortium include: a repository of informati
on about the World Wide Web for developers and users; reference code implementat
ions to embody and promote standards; and various prototype and sample applicati
ons to demonstrate use of new technology. Initially, the W3C was established in
collaboration with CERN, where the Web originated, with support from DARPA and t
he European Commission.

select 1-3 or press 4 to enter text

> █
```

Figure 9: Transforming individual resource data files into single CSV files for importing into the database. When a duplicate resource is identified, the interactive transformation process displays the conflicting properties (in this case, three resources entitled "World Wide Web Consortium (W3C)" have been identified in three different themes, and the administrator is prompted to interactively merge the data into a single record).

5.3 OntoPortal2 as a generic application framework

OntoPortal2 provides a generic application framework for modelling information domains. Section 5.1 demonstrated how the nature of the OntoPortal2 architecture provides a generic application framework that can be tailored to a particular application by defining an ontology which models the domain to be captured, and defining application-specific user interface presentation rules.

5.3.1 Defining a new ontology

OntoPortal2 currently supports only the simplest ontological modelling concepts. Modelling techniques such as subsumption (i.e. inheritance), facets, cardinality, and axioms are not supported. A domain must therefore be described in terms of the basic, non-hierarchical, ontological model shown in Figure 10. The target domain is divided into a number of *themes*, which represent distinct research areas. Each theme contains a number of resources or *classes*, classified by type, equating to real-world resources.

To describe the purpose and content of a particular theme (OntoPortal2 presents this information as introductory material to the resources contained within), the properties common to all themes can be defined in the ontology.

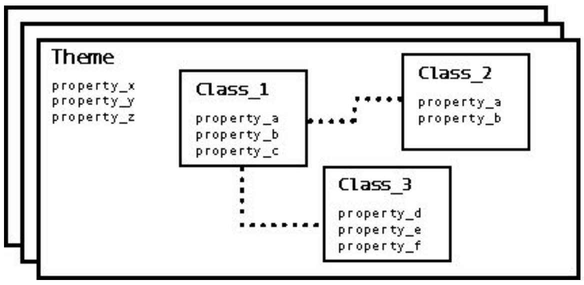


Figure 10: Basic structural model for knowledge capture using OntoPortal2

These properties may include single and multi-value properties, and may be assigned user-defined types. Similarly, the properties common to each type of resource can be defined in the ontology. The ontology also makes explicit the permitted relationships between the different resource types.

Figure 11 illustrates a simple ontology which might be used to model a research domain. Themes contain the properties *title* and *description*, and may have many related *links*. Each theme may contain a number of resources, classified as either *Researchers* or *Literature*. We wish to capture the *name*, *description*, and *email* of influential researchers in the research domain, and the *title*, *abstract*, and *url* of any relevant online literature. We also want to capture the relationships between *Researchers* and *Literature* (in this case, the simple relationship between a paper and its authors), and any *editorial* comments added by data harvesters as these literature resources are identified.

The steps required to build a domain-modelling application from the On-

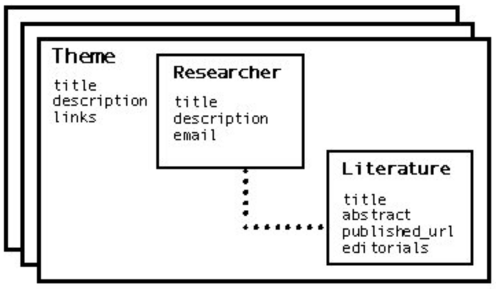


Figure 11: Example Ontology definition

toPortal2 framework using this example ontology are described below, and illustrated in Figure 12:

1. Describe ontology using schema-like XML notation (see Figure 13).

2. Apply XSLT stylesheet to ontology definition to transform definition into an XML 'database description'. This 'database description' describes the tables making up the relationship database that needs to be built in order to capture the knowledge in the domain modelled by the ontology (see Figure 14). Note that each table in the database description automatically receives an `id` field - these store the unique identifier assigned to each theme, resource, relationship, and user-defined type.

3. Translate 'database description' into series of SQL statements which actually construct the tables in the database.

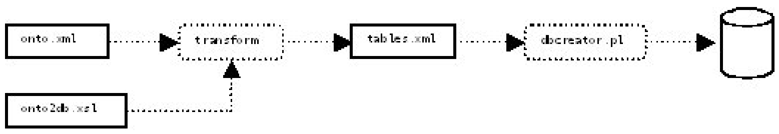


Figure 12: Database construction

```
<ontology>
    <themes>
        <properties>
            <property name="title" type="TEXT"/>
            <property name="description" type="TEXT"/>
            <property multivalued="yes" name="resources" type="resource"/>
        </properties>
    </themes>
    <classes>
        <class type="researcher">
            <properties>
                <property name="title" type="TEXT"/>
                <property name="description" type="TEXT"/>
                <property name="email" type="TEXT"/>
            </properties>
            <relations>
                <related_class type="literature"/>
            </relations>
        </class>
        <class type="literature">
            <properties>
                <property name="title" type="TEXT"/>
                <property name="abstract" type="TEXT"/>
                <property name="published_url" type="TEXT"/>
                <property  multivalued="yes" name="editorials" type="editorial"/>
            </properties>
            <relations>
                <related_class type="researcher"/>
            </relations>
        </class>
    </classes>
    <user_defined_properties>
        <user_defined_property name="editorial">
            <property name="editorial" type="TEXT"/>
            <property name="editorial_type" type="VARCHAR(8)"/>
        </user_defined_property>
        <user_defined_property name="resource">
            <property name="title" type="TEXT"/>
            <property name="url" type="TEXT"/>
            <property name="url_description" type="TEXT"/>
        </user_defined_property>
    </multi_value_properties>
</ontology>
```

Figure 13: XML description of ontology described in Figure 11

```
<db name="imdp">
    <table name="themes">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes"/>
        <field name="title" type="TINYTEXT" />
        <field name="description" type="TEXT" />
    </table>
    <table name="theme_resources">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="theme_id" type="INTEGER" />
        <field name="title" type="TINYTEXT" />
        <field name="url" type="TEXT" />
    </table>
    <table name="literature">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="title" type="TINYTEXT" />
        <field name="abstract" type="TEXT" />
        <field name="url_published" type="TEXT" />
    </table>
    <table name="literature_themes">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="theme_id" type="INTEGER" />
        <field name="literature_id" type="INTEGER" />
    </table>
    <table name="researcher">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="title" type="TINYTEXT" />
        <field name="description" type="TEXT" />
        <field name="email" type="TEXT" />
    </table>
    <table name="researcher_themes">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="theme_id" type="INTEGER" />
        <field name="researcher_id" type="INTEGER" />
    </table>
    <table name="researcher_editorials">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="theme_id" type="INTEGER" />
        <field name="researcher_id" type="INTEGER" />
        <field name="editorial" type="TEXT" />
        <field name="editorial_type" type="VARCHAR(8)" />
    </table>
    <table name="relationships">
        <field name="id" type="INTEGER" primary_key="yes" auto_increment="yes" />
        <field name="literature_id" type="INTEGER" />
        <field name="researcher_id" type="INTEGER" />
    </table>
</db>
```

Figure 14: XML 'Database description'; derived directly from ontology definition in Figure 13

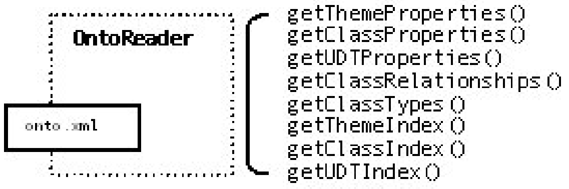
5.3.2 Enforcing ontology structure onto data stored in database

The ontology from which the database is derived explicitly enforces single- and multi-valued property names and types of both theme and class properties - the fact that the ontology description is used to generate the database ensures that all properties and types are reproduced in the database. In the same way, the ontology explicitly enforces the permitted types of resource in the database.

5.3.3 Enforcing ontological relationships between resources

The ontology enforces relationships between resources, by acting as a mediator when new resource descriptions are added to the database, ensuring that only permitted relationships between the new resource and resources already in the database can be asserted.

We implemented a module which provides information about the ontological structure of the domain being modelled by consulting the ontology description directly. This module, OntoReader, is described in more detail in Figure 15. The `getClassRelationships()` method can therefore be used to determine the permitted ontological relationships between newly added resources and existing resources in the database.



getThemeProperties (opt) returns theme properties (opt specifies all properties, single-value properties only, or multi-value properties only)

getClassProperties (class_type, opt) returns class properties for class_type (opt specifies all properties, single-value properties only, or multi-value properties only)

getUDTProperties (udt_type) returns properties of user defined type (UDT) udt_type

getClassRelationships (class_type) returns allowable ontological relationships between class_type and other classes

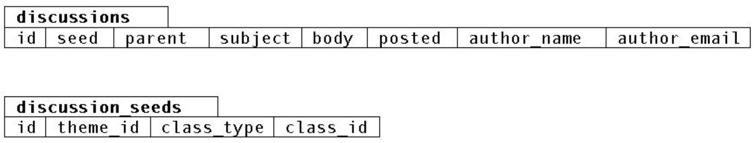
getClassTypes () returns all class types in ontology

getThemeIndex () returns database index for full-text searching of themes

getClassIndex (class_type) returns database index for full-text searching of class_type

getUDTIndex (udt_type) returns database index for full-text searching of user defined type (UDT) udt_type

Figure 15: The OntoReader module provides a means of consulting the ontological structure of the domain being modelled, in order to ensure that this structure is enforced in the database.



discussions							
id	seed	parent	subject	body	posted	author_name	author_email

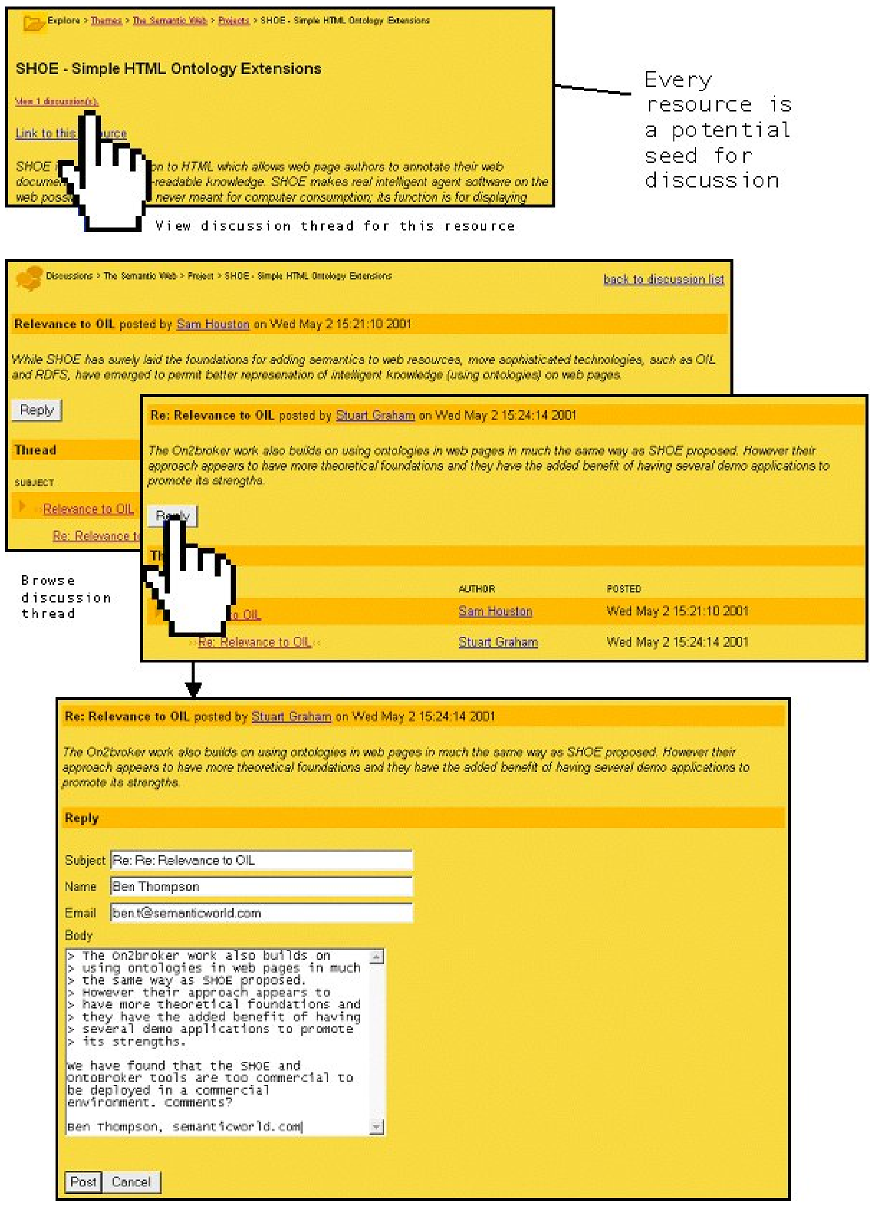
discussion_seeds			
id	theme_id	class_type	class_id

Figure 16: Tables used to store information about discussion threads in the database. Details of the messages posted to the discussion threads are stored in the `discussions` table. The `discussion_seeds` table provides the context for the discussion, by recording the theme id, class type, and class id of the resource each message discusses.

5.4 Providing a threaded discussion facility

We have already seen that the Explore interface augments resources with links to discussion threads when registered users interact with the system (Section 3). This section describes the implementation of the threaded discussion facility in more detail.

Any resource stored in the system can potentially act as impetus for debate (Figures 17 and 18). Information about messages posted by registered users to the discussion threads are stored in two tables in the database - these tables are constructed automatically when the database is built from the ontology description file (Section 5.3). The messages themselves are stored in one table, and the other table is used to join each message to the resource which it discusses (Figure 16).


Every resource is a potential seed for discussion
View discussion thread for this resource
Browse discussion thread

Explore > Themes > The Semantic Web > Projects > SHOE - Simple HTML Ontology Extensions

SHOE - Simple HTML Ontology Extensions

View 1 discussion(s).

Link to this urce

SHOE on to HTML which allows web page authors to annotate their web docume -readable knowledge. SHOE makes real intelligent agent software on the web poss never meant for computer consumption; its function is for displaying

Discussions > The Semantic Web > Project > SHOE - Simple HTML Ontology Extensions

back to discussion list

Relevance to OIL posted by Sam Houston on Wed May 2 15:21:10 2001

While SHOE has surely laid the foundations for adding semantics to web resources, more sophisticated technologies, such as OIL and RDFS, have emerged to permit better represenation of intelligent knowledge (using ontologies) on web pages.

Reply

Thread

SUBJECT

Relevance to OIL

Re: Relevance t

Re: Relevance to OIL posted by Stuart Graham on Wed May 2 15:24:14 2001

The On2broker work also builds on using ontologies in web pages in much the same way as SHOE proposed. However their approach appears to have more theoretical foundations and they have the added benefit of having several demo applications to promote its strengths.

Reply

Th

	AUTHOR	POSTED
o OIL	Sam Houston	Wed May 2 15:21:10 2001
Re: Relevance to OIL	Stuart Graham	Wed May 2 15:24:14 2001

Re: Relevance to OIL posted by Stuart Graham on Wed May 2 15:24:14 2001

The On2broker work also builds on using ontologies in web pages in much the same way as SHOE proposed. However their approach appears to have more theoretical foundations and they have the added benefit of having several demo applications to promote its strengths.

Reply

Subject	Re: Re: Relevance to OIL
Name	Ben Thompson
Email	ben.t@semanticworld.com

Body

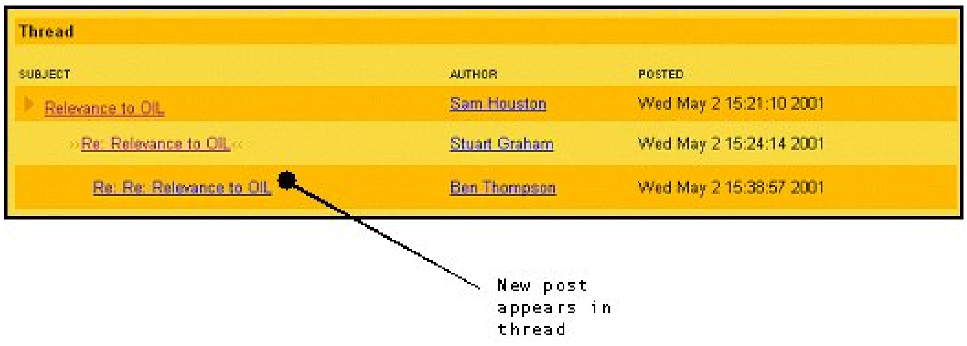
```
> The On2broker work also builds on
> using ontologies in web pages in much
> the same way as SHOE proposed.
> However their approach appears to
> have more theoretical foundations and
> they have the added benefit of having
> several demo applications to promote
> its strengths.

We have found that the SHOE and
OntoBroker tools are too commercial to
be deployed in a commercial
environment. Comments?

Ben Thompson, semanticworld.com
```

Post Cancel

Figure 17: Any resource in the system can potentially act as an impetus for discussion. The Explore interface provides entry points into the discussion threads for each resource. The discussion threads can be browsed, and messages can be posted in response to previous postings or posted as a new thread to initiate a new direction of discussion.



Thread

SUBJECT	AUTHOR	POSTED
Relevance to OIL	Sam Houston	Wed May 2 15:21:10 2001
››Re: Relevance to OIL‹‹	Stuart Graham	Wed May 2 15:24:14 2001
Re: Re: Relevance to OIL	Ben Thompson	Wed May 2 15:38:57 2001

Figure 18: Result of posting a message to the discussion thread in Figure 17

5.4.1 Indenting discussion threads

In the presentation of discussion threads, we wanted to achieve the “indented list” style used by many online discussion forums, where a message posted in reply to a previous message is indented further to the right than its predecessor. The solution came when we considered the XML representation of a discussion thread (Figure 19).

```
<discussions>
  <message id="5">
    <subject>Relevance to OIL</subject>
    <author email="sh@kfg9.net" name="Sam Houston" />
    <posted>Wed May 2 15:21:10 2001</posted>
    <replies>
      <message id="6">
        <subject>Re: Relevance to OIL</subject>
        <author email="sg@msn.com" name="Stuart Graham" />
        <posted>Wed May 2 15:24:14 2001</posted>
        <replies>
          <message id="7">
            <subject>Re: Re: Relevance to OIL</subject>
            <author email="ben.t@sw.com" name="Ben Thompson" />
            <posted>Wed May 2 15:38:57 2001</posted>
            <replies />
          </message>
        </replies>
      </message>
    </replies>
  </message>
</discussions>
```

Figure 19: XML description of a discussion thread, generated in response to a request to view the discussion threads surrounding a particular resource in the database.

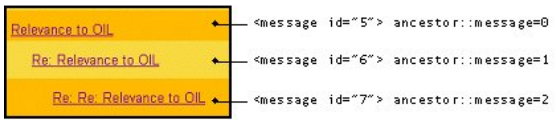

Relevance to OIL
Re: Relevance to OIL
Re: Re: Relevance to OIL
<message id="5"> ancestor::message=0
<message id="6"> ancestor::message=1
<message id="7"> ancestor::message=2

Figure 20: Discussion thread presentation, with message subjects appropriately indented according to their position in the discussion.

The logical hierarchical structure of the discussion thread suggested that we could use the axis `ancestor` to collate the ancestor message nodes of any particular message node in the structure. For example, if we are at node `<message id="7">`, `ancestor::message` would return a node set containing the nodes `<message id="6">` and `<message id="5">`. The XSLT creates the appropriate indentation for each message in the thread as follows, by inserting a number of small "spacer" images before each message header:

```
<xsl:for-each select="ancestor::message">
    <IMG height="20" width="20" src="b.gif" align="center"/>
</xsl:for-each> <xsl:value-of select="subject"/>
```

Figure 20 shows the resulting presentation of the discussion thread.

5.5 Providing a search facility

We have already seen that the Explore interface provides an entry point to the Search interface, allowing the user to query the resources stored in the OntoPortal2 database for specified terms (Section 3) - Figure 21 illustrates this entry point, and resulting interaction with the Search interface. This section describes the implementation of the search facility in more detail.

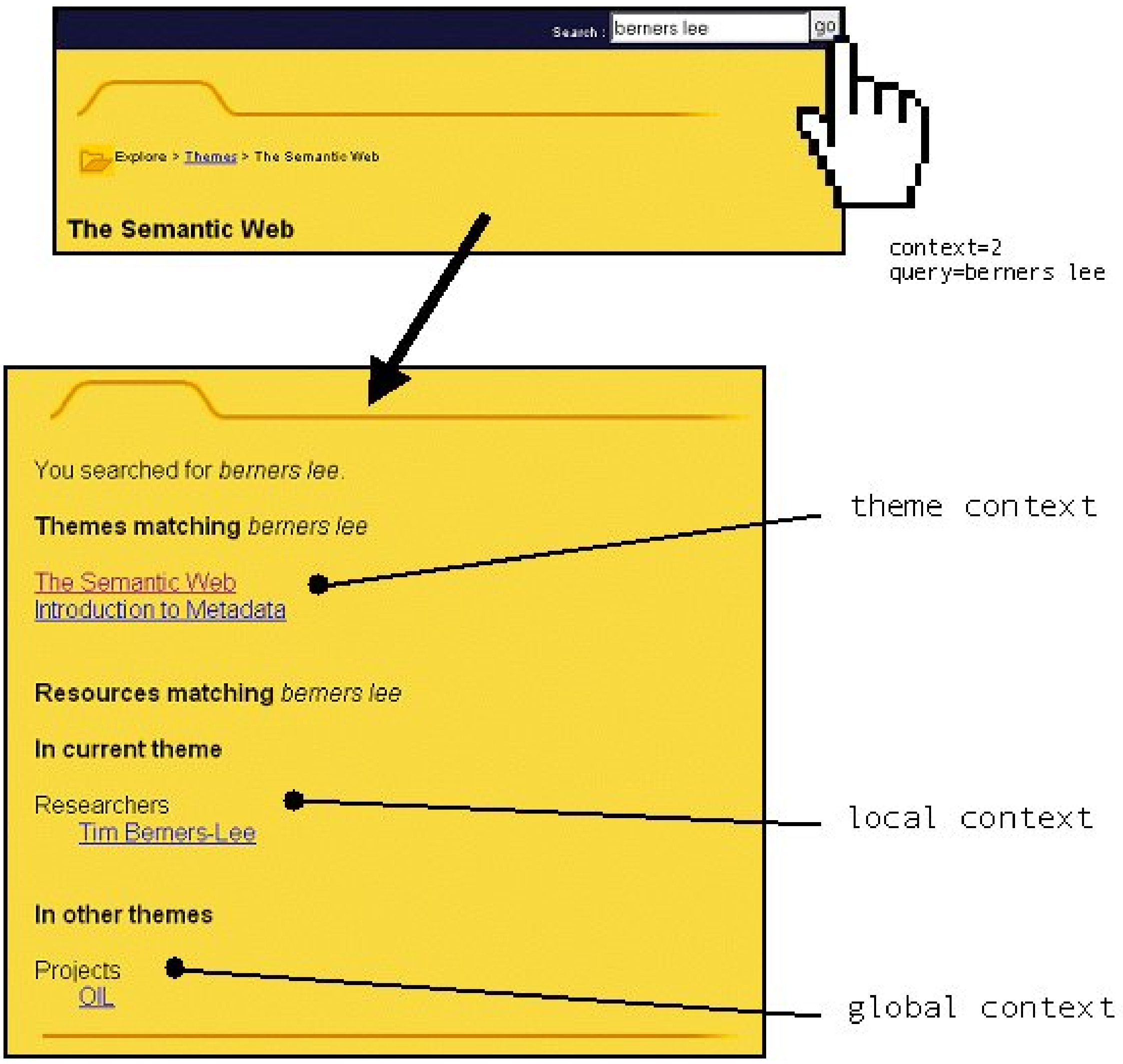
In order to allow full text searching across the resources in the database, we needed to index the tables in the database, according to which particular resource properties we wanted users to be able to search. We decided to allow the ontology definition to define which properties should be used to construct the full text search index for each resource (Figure 22).

5.5.1 Generic searching

In order to perform a full-text search across a database table, the following SQL syntax is applied:

```
SELECT fields FROM table WHERE MATCH (index) AGAINST (terms)
```

(where `index` is a comma-separated list of the fields in the making up the index). Example SQL queries in the context of a database constructed from



You searched for *berners lee*.

Themes matching *berners lee*

The Semantic Web
Introduction to Metadata

Resources matching *berners lee*

In current theme

Researchers
Tim Berners-Lee

In other themes

Projects
OIL

Figure 21: The entry point to the Search interface appears in all interactions with the Explore interface. By entering search terms, the user can quickly find relevant themes, and local (within the current theme) and global (across the whole database) resources.

```
<ontology>
    <themes>
        <properties>
            <property name="title" type="TEXT" index="yes"/>
            <property name="description" type="TEXT" index="yes"/>
            <property multi_valued="yes" name="resources" type="resource"/>
        </properties>
    </themes>
    <classes>
        <class type="researcher">
            <properties>
                <property name="title" type="TEXT" index="yes"/>
                <property name="description" type="TEXT" index="yes"/>
                <property name="email" type="TEXT"/>
            </properties>
            <relations>
                <related_class type="literature"/>
            </relations>
        </class>
    </classes>
    <user_defined_properties>
        <user_defined_property name="resource">
            <property name="title" type="TEXT" index="yes"/>
            <property name="url" type="TEXT"/>
            <property name="url_description" type="TEXT"/>
        </user_defined_property>
    </user_defined_properties>
</ontology>
```

Figure 22: Example ontology definition augmented with `index` attributes where a property is to be included as part of the full text search index for that resource. In this example, themes will be indexed using the `title` and `description` properties, `researcher` resources will be indexed using the `title` and `description` properties, and the user-defined type `resource` will be indexed using the `title` property.

the ontology definition in Figure 22 are:

```
SELECT id FROM themes WHERE MATCH (title,description) AGAINST

('metadata')
```

```
SELECT id FROM researcher WHERE MATCH (title,description) AGAINST

('metadata')
```

```
SELECT id FROM theme_editorials WHERE MATCH (editorial) AGAINST

('metadata')
```

```
SELECT id FROM theme_resources WHERE MATCH (title) AGAINST ('metadata')
```

In order to construct these queries *generically*, we need to be able to derive the appropriate `index` at run-time. We do this by using the `OntoReader` module (see Section refgeneric-db), which provides the methods `getThemeIndex()`, `getClassIndex()` and `getUDTIndex()`. For example, the strategy for performing a full-text search on themes becomes:

1. Get index for theme - `getThemeIndex()`
2. Construct SQL statement for extracting the id of all themes which match the query
3. Execute SQL statement, storing id of all themes that match the query
4. Foreach UDT (user defined type), get the index for the UDT (`getUDTIndex`), construct SQL statement for extracting the id of all themes for which the matching UDT data is a property, and store returned theme id values
5. Foreach (unique) theme id stored, extract the the title of the theme and return this as a positive match result.
6. Apply `search.xsl` stylesheet to present results to user.

5.6 Creating an offline distribution

This section describes the issues involved in creating a “distribution” version of domains modelled by OntoPortal2. A distribution version would allow the data to be browsed (for example, from a CD) without having to be online or having to configure the OntoPortal2 framework (i.e. the database/CGI backend) to power exploration of the domain. Rather than have each data view dynamically generated according to user interactions, a distribution version consists of interlinked static HTML pages that “snapshot” the state of the dynamic pages at the time of creation.

We wanted to use an existing tool to create a static version - Gnu `wget` [2]. From `wget` documentation:

> Wget is a network utility to retrieve files from the Web using http and ftp, the two most widely used Internet protocols. It works non-interactively, so it will work in the background, after having logged off. The program supports recursive retrieval of web-authoring pages as well as ftp sites– you can use wget to make mirrors of archives and home pages or to travel the Web like a WWW robot.

We found that when wget makes a request to the Explore interface (we are currently only concerned with capturing the Explore interface for distribution), the filename of the saved HTML file is created as follows:

```
REQUEST:
http://www.ontoportal.org.uk/explore.cgi?theme=1&class_type=literature&class=24
FILENAME:
www.ontoportal.org.uk/explore.cgi@theme=1@class_type=literature@class=24
```

These mirrored HTML files cannot then be browsed easily by a HTML browser (since there is no `.html` extension), and the links between HTML pages do function. The hierarchical feel of the Explore interface is lost, since files are stored in flat directory structure.

5.6.1 An offline solution

We redesigned the method by which the Explore interface receives parameters from user interactions, which gives the appearance of browsing a hierarchical directory of HTML files, rather than passing arguments to a CGI script:

```
EXPLICIT PARAMETERS:
http://www.ontoportal.org.uk/explore.cgi?theme=1&class_type=literature&class=24
IMPLICIT PARAMETERS:
http://www.ontoportal.org.uk/explore/themes/1/literature/24.html
```

The Explore interface, invoked through the `explore.cgi` script (renamed to `explore` to continue the illusion of browsing a directory hierarchy) is still being invoked, but the arguments are passed to it differently. The first invokation explicitly passes the named arguments to the script in the conventional manner. In the second invokation, the parameters are derived implicitly from the additional information after the script name (/themes/1/literature/24.html). However this invokation is treated as a hierarchical directory structure by `wget`, which proceeds to mirror the structure (Figure 23).

The `wget` tool can then be used to create a static snapshot of the dynamic Explore interface, producing a hierarchical directory of inter-linked HTML files which can be distributed on CD. To remove references to the dynamic Search, Update, and Discuss interfaces, an alternative "static" XLST stylesheet was derived from existing stylesheets, with all entry points to interactive content removed.

```
OntoPortal2.org.uk/
    themes/
        1/
            literature/
                1.html
                2.html
                index.html
            software/
            researcher/
            project/
        2/
        3/
        index.html
        1.html
        2.html
        3.html
```

Figure 23: Directory structure automatically created by `wget` in mirroring the dynamic Explore interface to a distributable static, inter-linked HTML version.

6 MetaPortal: An application of OntoPortal

To create the MetaPortal web site, the OntoPortal system was used. The ontology was defined, as discussed above, and selected members of the IAM Research with the correct level of experience, were used to provide the content for MetaPortal. Each author was in charge of one or two themes. The following themes were selected for MetaPortal.

- Bibliographic Metadata for Digital Libraries
- Inferences and Metadata
- Introduction to Metadata
- Metadata Acquisition
- Metadata for Expertise Models
- Metadata Schemas
- Middleware and Metadata
- Ontologies and Metadata
- Reasoning and Knowledge
- The Semantic Web

The total number of concepts added is detailed below.

Centers of Excellence	4
Literature	157
Organisations	9
Projects	46
Promotion Projects	2
Researchers	129
Software	33
Standards	25
Standards Committees	12
Teams	44

641 relationships exist between all the concepts.

7 Related Work

The practice of adding semantics to web resources (using standards such as XML [18], RDF [17], Dublin Core [1], OML [15], and OIL [8]) in order to build a knowledge base has been the focus of previous research.

SHOE [12] allows researchers to annotate their WWW resources with meta-data, in order to build a distributed knowledge base. Ontologies are used to declare the desired characteristics and relationships of a Web resource and SHOE-specific markup is used to annotate the resource and make its properties explicit. The real potential behind SHOE is the ability to draw on the ontology to infer supplementary knowledge not directly stated within the facts describing the web resource. (KA)2 [6] applies a similar approach although places a greater emphasis on the ontological engineering process as well as supporting inference. [16] describes how the (KA)2 initiative has now evolved to provide a coherent set of tools with which to design community Web portals. Finally, topic maps [13] are used to classify concepts (topics) and their relationships in a web site. This semantic layer is then used as a concept browser where users are provided with consistent and accurate access to the information.

COHSE [7] is an open hypermedia system whose links are defined by an independent Ontology Service. ScholOnto [14] uses an ontology of argumentation to model relationships in the literature, focusing on the claims scholars make in their articles. The ontology supports patterns of argumentation between literature, such as refutation, support, extension or modification. These relationships are recorded in a central knowledge base, from which higher-level relationships between the literatures can be inferred, such as *Has anyone built on the ideas in this paper, and in what way?*.

Zope [11] is an open source web application server that enables collaborate content creation and management, much like OntoPortal, but also provides more advanced features like membership, personalization and e-commerce. Where they differ however, is in the representation of the knowledge. OntoPortal uses an ontology to structure the knowledge within a domain and provide the navigation paradigm, while Zope enforces an object-oriented structure over familiar web concepts such as folders, documents and images.

8 Further Work

While a huge improvement over OntoPortal1, OntoPortal2 has room for further work.

Personalisation It would be interesting to take *more* advantage of the extensibility of XML and XSLT to produce a personalised view of the data on a per user basis. At the moment the personalisation simply applies to the authentication status of the user (i.e. not logged in, logged in to participate in discussions, logged in to author).

Better ontological support Currently the range of ontological structures supported is severely limited. At the very least, support for subsumption needs to be added which would allow OntoPortal to support most ontological structures, albeit perhaps in a limited fashion. Other ontological methods such as facets, cardinality, and axioms are also desirable although not as vital.

Generic stylsheet construction Although the back-end processes of OntoPortal are generic in the sense that any ontology (conforming to OntoPortal2) can be plugged in to create a new application, the construction of the accompanying stylesheets is a manual process requiring a significant time overhead. Naturally, this does have the advantage that the user can fully customise the look and feel of her OntoPortal application, and this ability must not be removed. However, it would be useful to have a set of tools to either create a set of *standard* stylesheets using the defined ontology, or at least create skeleton stylesheets for the administrator to complete.

Discussion moderation In OntoPortal2 only authorised users are permitted to post a discussion about a resource within the application. This process is unmoderated, relying on the authorised users to conduct an appropriate discussion. This approach has two disadvantages. Firstly, unregistered users are unable to participate. Secondly, the possibility of inappropriate discussions evolving exists. A moderated system on the other hand, will solve both problems. Any user is allowed to post discussions and the possibility of inappropriate discussions appearing is eliminated. The downside is the time lag between a user posting a message and it appearing on the site.

The OntoPortal framework makes it easy to create knowledge portals such as MetaPortal. The framework was recently used to create an educational site for lecturers called TPortal. TPortal allows lecturers to gain access to course related material to enable them to better understand *all* available resources and their relationships. A simple ontology was designed (Figure 24) and

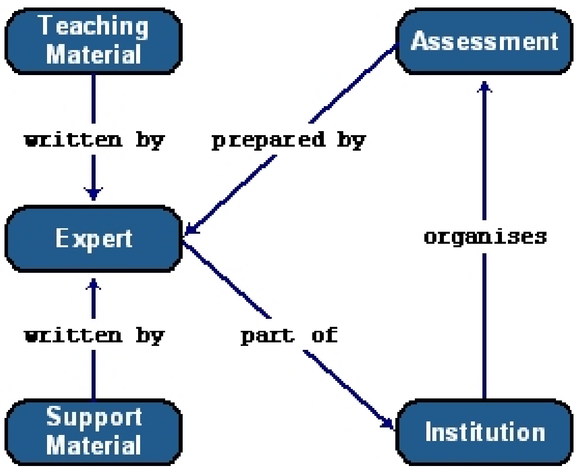
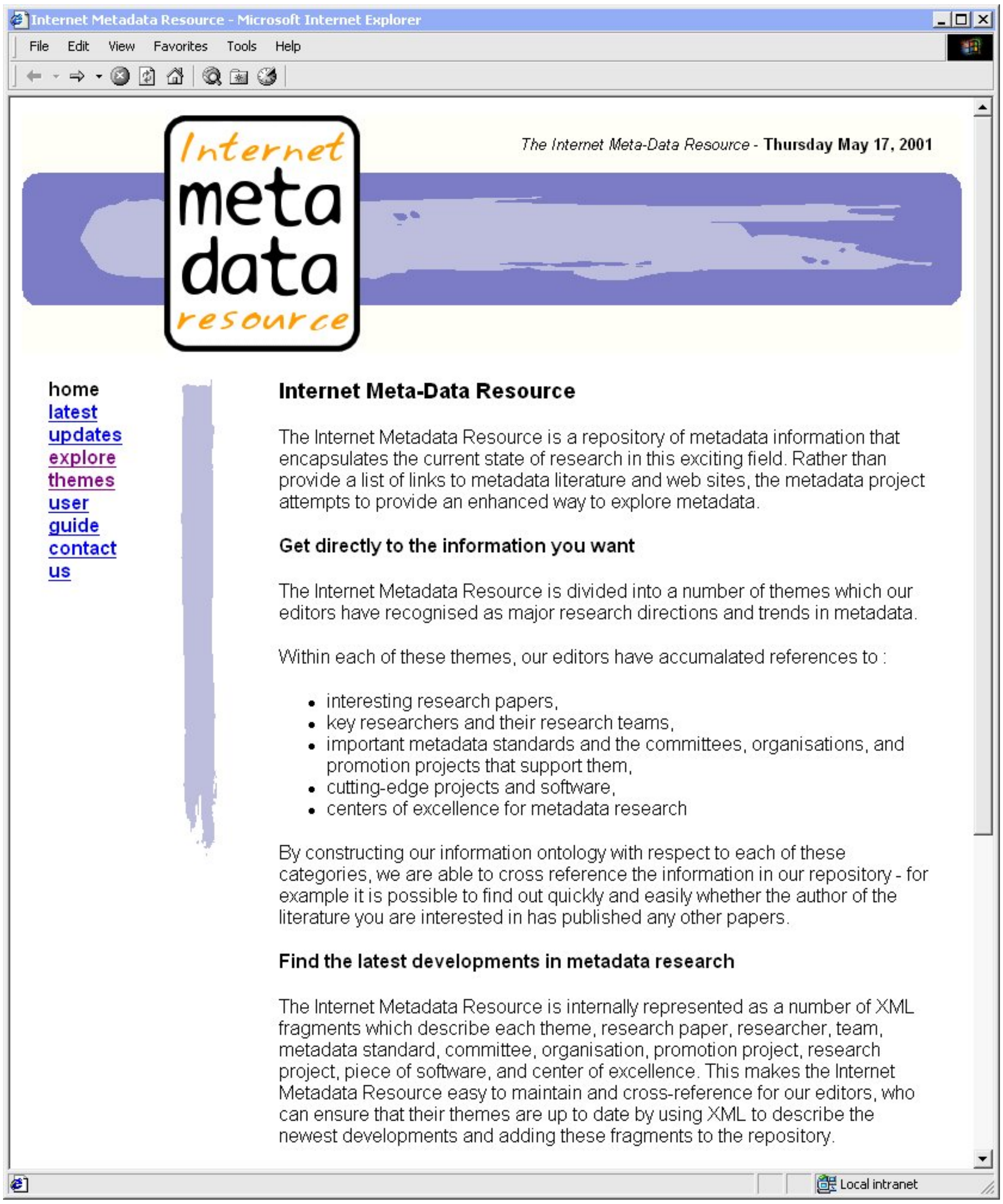


Figure 24: The TPortal ontology

initially a single theme, entitled *Java*, was populated. It is envisaged that TPortal will be expanded to cover more themes and then be used by the computer science department at the University of Southampton.

A OntoPortal Version 1 Screenshots


Internet Metadata Resource - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Internet meta data resource
home
latest
updates
explore
themes
user
guide
contact
us
Local intranet

The Internet Meta-Data Resource - **Thursday May 17, 2001**

Internet Meta-Data Resource

The Internet Metadata Resource is a repository of metadata information that encapsulates the current state of research in this exciting field. Rather than provide a list of links to metadata literature and web sites, the metadata project attempts to provide an enhanced way to explore metadata.

Get directly to the information you want

The Internet Metadata Resource is divided into a number of themes which our editors have recognised as major research directions and trends in metadata.

Within each of these themes, our editors have accumalated references to :

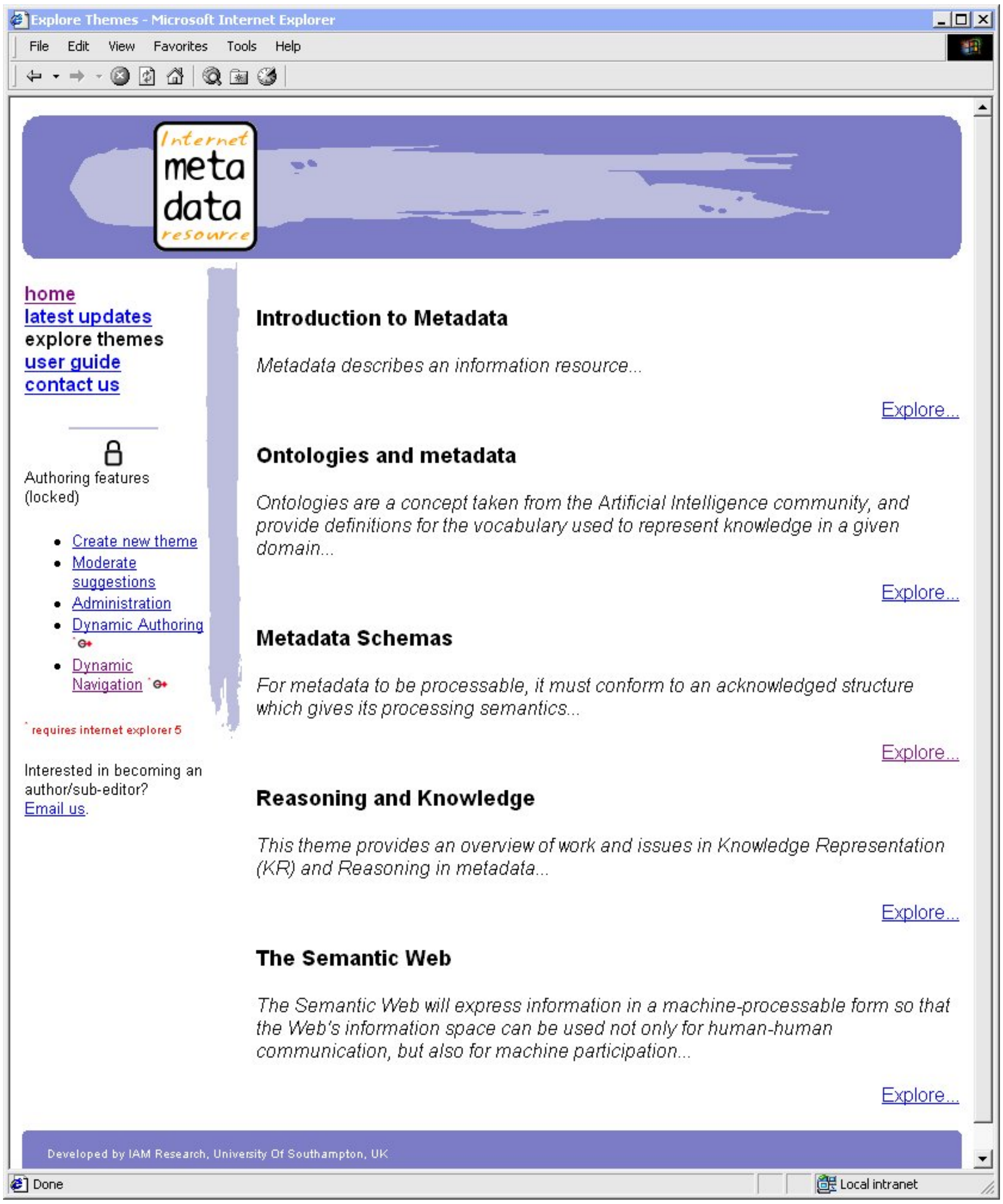
- interesting research papers,
- key researchers and their research teams,
- important metadata standards and the committees, organisations, and promotion projects that support them,
- cutting-edge projects and software,
- centers of excellence for metadata research

By constructing our information ontology with respect to each of these categories, we are able to cross reference the information in our repository - for example it is possible to find out quickly and easily whether the author of the literature you are interested in has published any other papers.

Find the latest developments in metadata research

The Internet Metadata Resource is internally represented as a number of XML fragments which describe each theme, research paper, researcher, team, metadata standard, committee, organisation, promotion project, research project, piece of software, and center of excellence. This makes the Internet Metadata Resource easy to maintain and cross-reference for our editors, who can ensure that their themes are up to date by using XML to describe the newest developments and adding these fragments to the repository.

Figure 25: The OntoPortal1 home page (formerly known as the Internet Meta-Data Resource)


Explore Themes - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Internet metadata resource

home
latest updates
explore themes
user guide
contact us

Authoring features (locked)

- Create new theme
- Moderate suggestions
- Administration
- Dynamic Authoring *
- Dynamic Navigation *

* requires internet explorer 5

Interested in becoming an author/sub-editor? Email us.

Introduction to Metadata

Metadata describes an information resource...

Explore...

Ontologies and metadata

Ontologies are a concept taken from the Artificial Intelligence community, and provide definitions for the vocabulary used to represent knowledge in a given domain...

Explore...

Metadata Schemas

For metadata to be processable, it must conform to an acknowledged structure which gives its processing semantics...

Explore...

Reasoning and Knowledge

This theme provides an overview of work and issues in Knowledge Representation (KR) and Reasoning in metadata...

Explore...

The Semantic Web

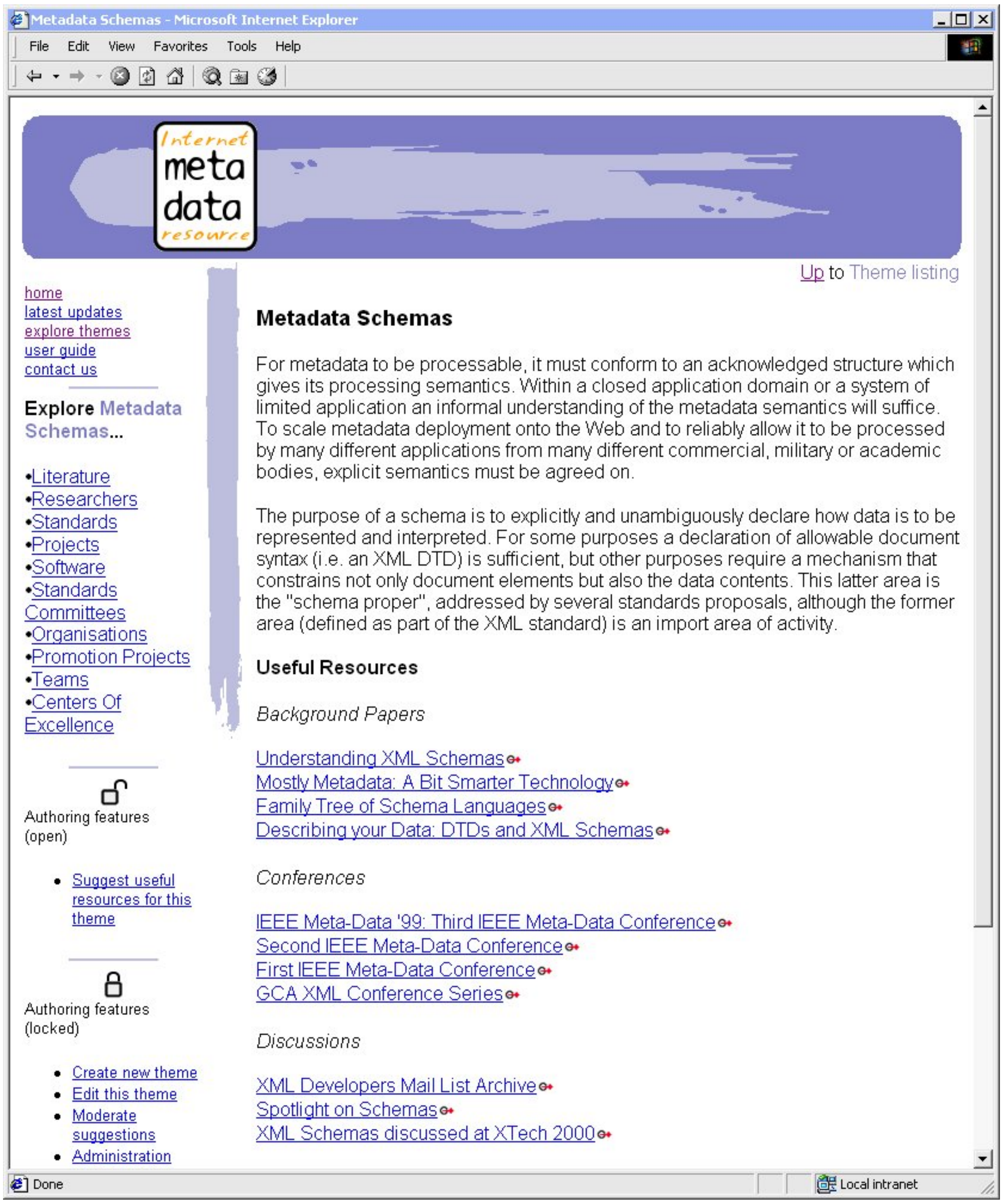
The Semantic Web will express information in a machine-processable form so that the Web's information space can be used not only for human-human communication, but also for machine participation...

Explore...

Developed by IAM Research, University Of Southampton, UK

Done Local intranet

Figure 26: Themes available for exploration in OntoPortal1


Metadata Schemas - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Internet meta data resource
Done
Local intranet

Up to Theme listing

home
latest updates
explore themes
user guide
contact us

Explore Metadata Schemas...

- Literature
- Researchers
- Standards
- Projects
- Software
- Standards Committees
- Organisations
- Promotion Projects
- Teams
- Centers Of Excellence

Authoring features (open)

- Suggest useful resources for this theme

Authoring features (locked)

- Create new theme
- Edit this theme
- Moderate suggestions
- Administration

Metadata Schemas

For metadata to be processable, it must conform to an acknowledged structure which gives its processing semantics. Within a closed application domain or a system of limited application an informal understanding of the metadata semantics will suffice. To scale metadata deployment onto the Web and to reliably allow it to be processed by many different applications from many different commercial, military or academic bodies, explicit semantics must be agreed on.

The purpose of a schema is to explicitly and unambiguously declare how data is to be represented and interpreted. For some purposes a declaration of allowable document syntax (i.e. an XML DTD) is sufficient, but other purposes require a mechanism that constrains not only document elements but also the data contents. This latter area is the "schema proper", addressed by several standards proposals, although the former area (defined as part of the XML standard) is an import area of activity.

Useful Resources

Background Papers

Understanding XML Schemas
Mostly Metadata: A Bit Smarter Technology
Family Tree of Schema Languages
Describing your Data: DTDs and XML Schemas

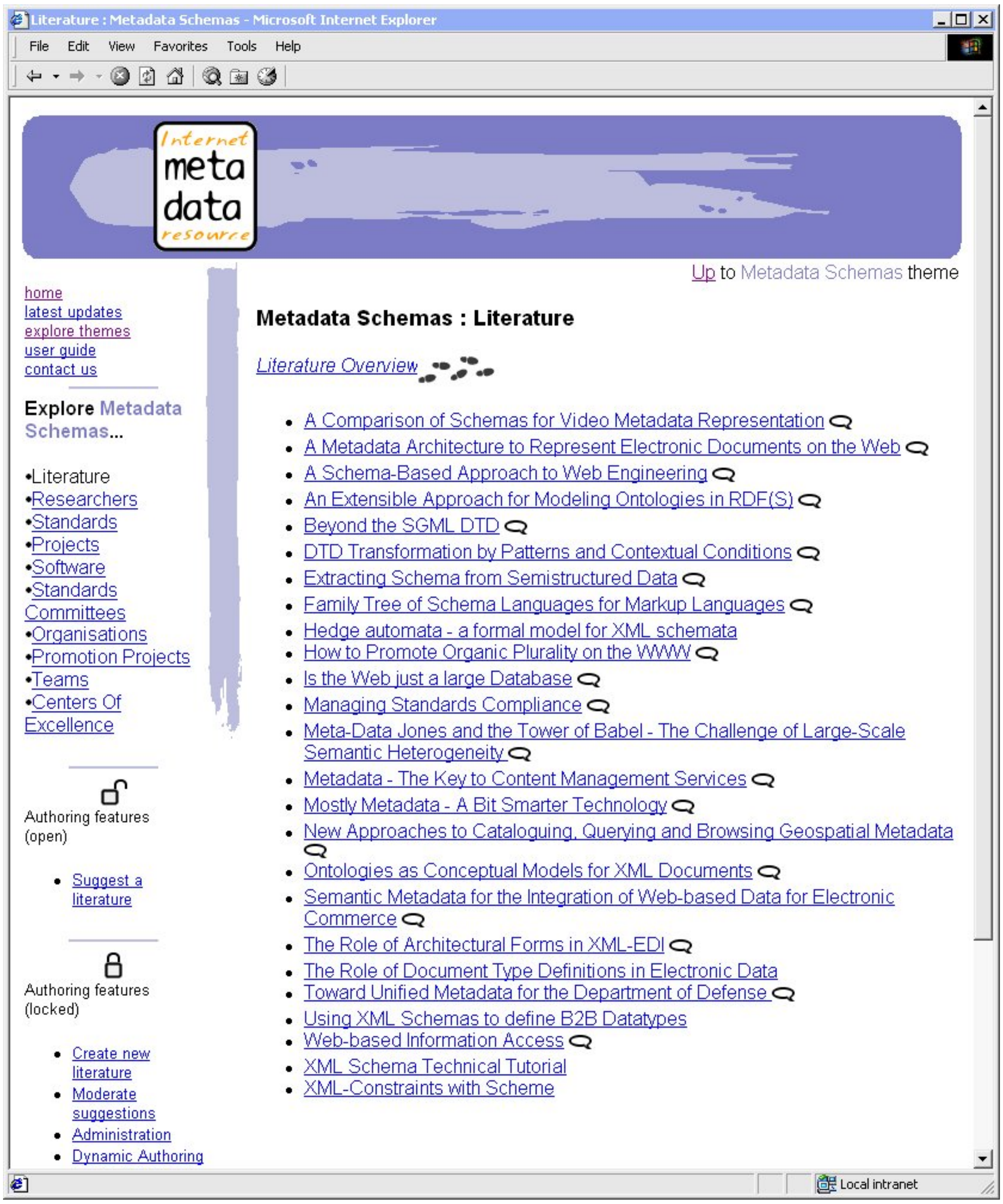
Conferences

IEEE Meta-Data '99: Third IEEE Meta-Data Conference
Second IEEE Meta-Data Conference
First IEEE Meta-Data Conference
GCA XML Conference Series

Discussions

XML Developers Mail List Archive
Spotlight on Schemas
XML Schemas discussed at XTech 2000

Figure 27: Browsing the "Metadata Schemas" theme


Literature : Metadata Schemas - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Up to Metadata Schemas theme

home
latest updates
explore themes
user guide
contact us

Explore Metadata Schemas...

•Literature
•Researchers
•Standards
•Projects
•Software
•Standards Committees
•Organisations
•Promotion Projects
•Teams
•Centers Of Excellence

Authoring features (open)

- Suggest a literature

Authoring features (locked)

- Create new literature
- Moderate suggestions
- Administration
- Dynamic Authoring

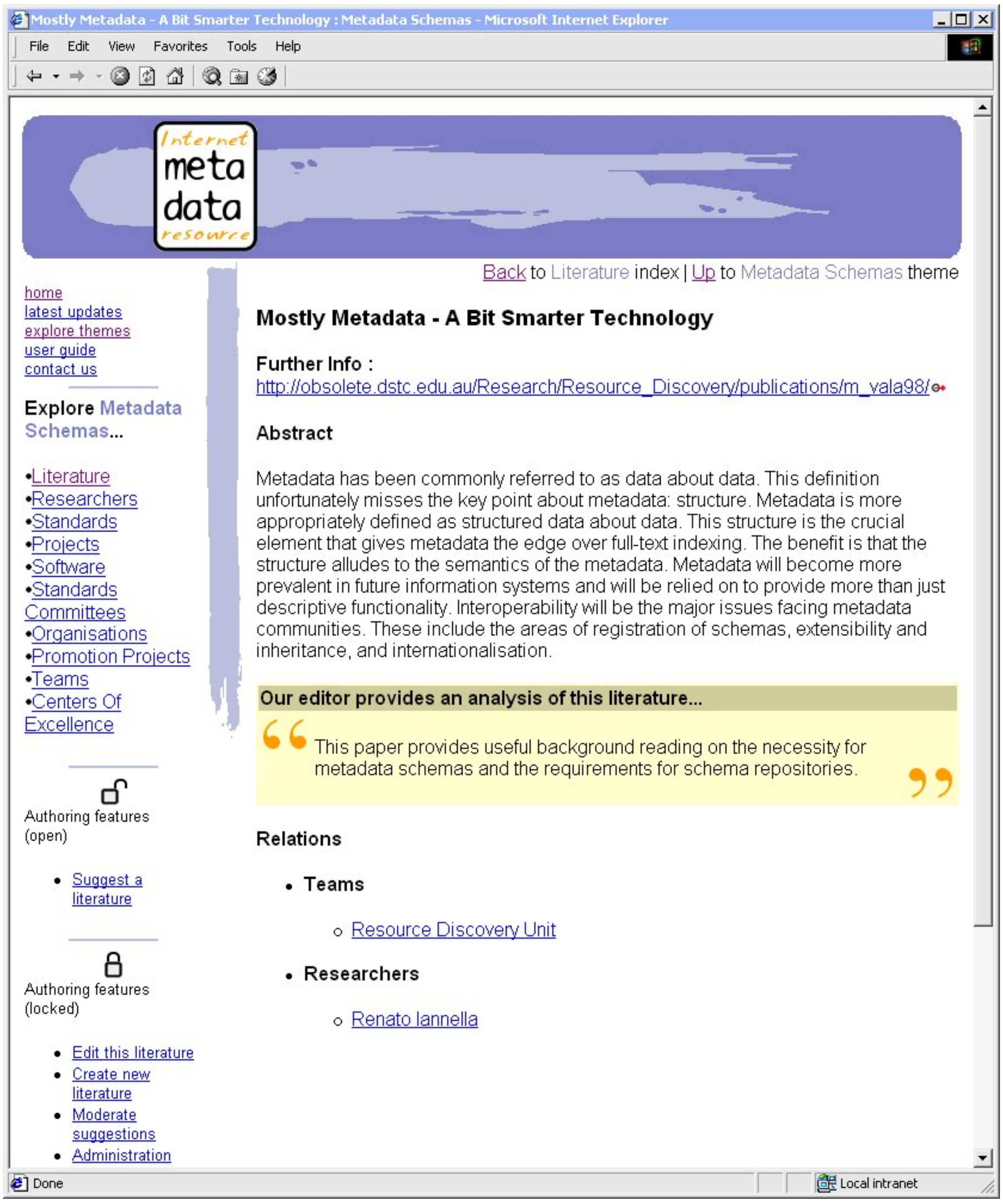
Metadata Schemas : Literature

Literature Overview

- A Comparison of Schemas for Video Metadata Representation
- A Metadata Architecture to Represent Electronic Documents on the Web
- A Schema-Based Approach to Web Engineering
- An Extensible Approach for Modeling Ontologies in RDF(S)
- Beyond the SGML DTD
- DTD Transformation by Patterns and Contextual Conditions
- Extracting Schema from Semistructured Data
- Family Tree of Schema Languages for Markup Languages
- Hedge automata - a formal model for XML schemata
- How to Promote Organic Plurality on the WWW
- Is the Web just a large Database
- Managing Standards Compliance
- Meta-Data Jones and the Tower of Babel - The Challenge of Large-Scale Semantic Heterogeneity
- Metadata - The Key to Content Management Services
- Mostly Metadata - A Bit Smarter Technology
- New Approaches to Cataloguing, Querying and Browsing Geospatial Metadata
- Ontologies as Conceptual Models for XML Documents
- Semantic Metadata for the Integration of Web-based Data for Electronic Commerce
- The Role of Architectural Forms in XML-EDI
- The Role of Document Type Definitions in Electronic Data
- Toward Unified Metadata for the Department of Defense
- Using XML Schemas to define B2B Datatypes
- Web-based Information Access
- XML Schema Technical Tutorial
- XML-Constraints with Scheme

Local intranet

Figure 28: Browsing the literature resources relevant to the “Metadata Schemas” theme



Mostly Metadata - A Bit Smarter Technology : Metadata Schemas - Microsoft Internet Explorer

File Edit View Favorites Tools Help
Internet meta data resource

Back to Literature index | Up to Metadata Schemas theme

home
latest updates
explore themes
user guide
contact us

Explore Metadata Schemas...

- Literature
- Researchers
- Standards
- Projects
- Software
- Standards Committees
- Organisations
- Promotion Projects
- Teams
- Centers Of Excellence

Authoring features (open)

- Suggest a literature

Authoring features (locked)

- Edit this literature
- Create new literature
- Moderate suggestions
- Administration

Mostly Metadata - A Bit Smarter Technology

Further Info :
http://obsolete.dstc.edu.au/Research/Resource_Discovery/publications/m_vala98/

Abstract

Metadata has been commonly referred to as data about data. This definition unfortunately misses the key point about metadata: structure. Metadata is more appropriately defined as structured data about data. This structure is the crucial element that gives metadata the edge over full-text indexing. The benefit is that the structure alludes to the semantics of the metadata. Metadata will become more prevalent in future information systems and will be relied on to provide more than just descriptive functionality. Interoperability will be the major issues facing metadata communities. These include the areas of registration of schemas, extensibility and inheritance, and internationalisation.

Our editor provides an analysis of this literature...

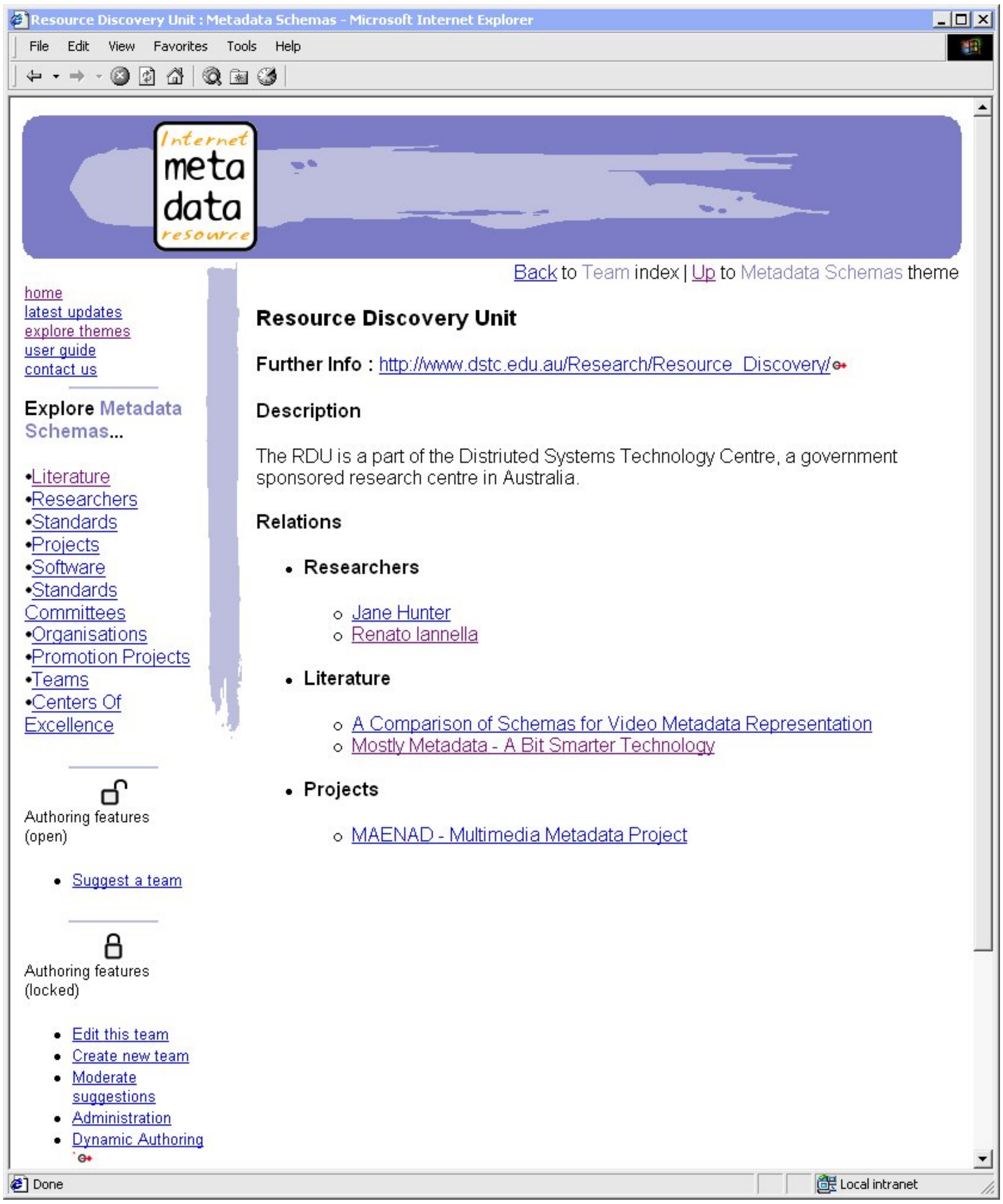
“This paper provides useful background reading on the necessity for metadata schemas and the requirements for schema repositories.”

Relations

- **Teams**
  - Resource Discovery Unit
- **Researchers**
  - Renato Iannella

Done | Local intranet

Figure 29: Exploring a literature resource in more detail (note ontological relationships)


Resource Discovery Unit : Metadata Schemas - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Internet meta data resource

Back to Team index | Up to Metadata Schemas theme

home
latest updates
explore themes
user guide
contact us

Explore Metadata Schemas...

•Literature
•Researchers
•Standards
•Projects
•Software
•Standards Committees
•Organisations
•Promotion Projects
•Teams
•Centers Of Excellence

Authoring features (open)

- Suggest a team

Authoring features (locked)

- Edit this team
- Create new team
- Moderate suggestions
- Administration
- Dynamic Authoring

Resource Discovery Unit

Further Info : http://www.dstc.edu.au/Research/Resource_Discovery/

Description

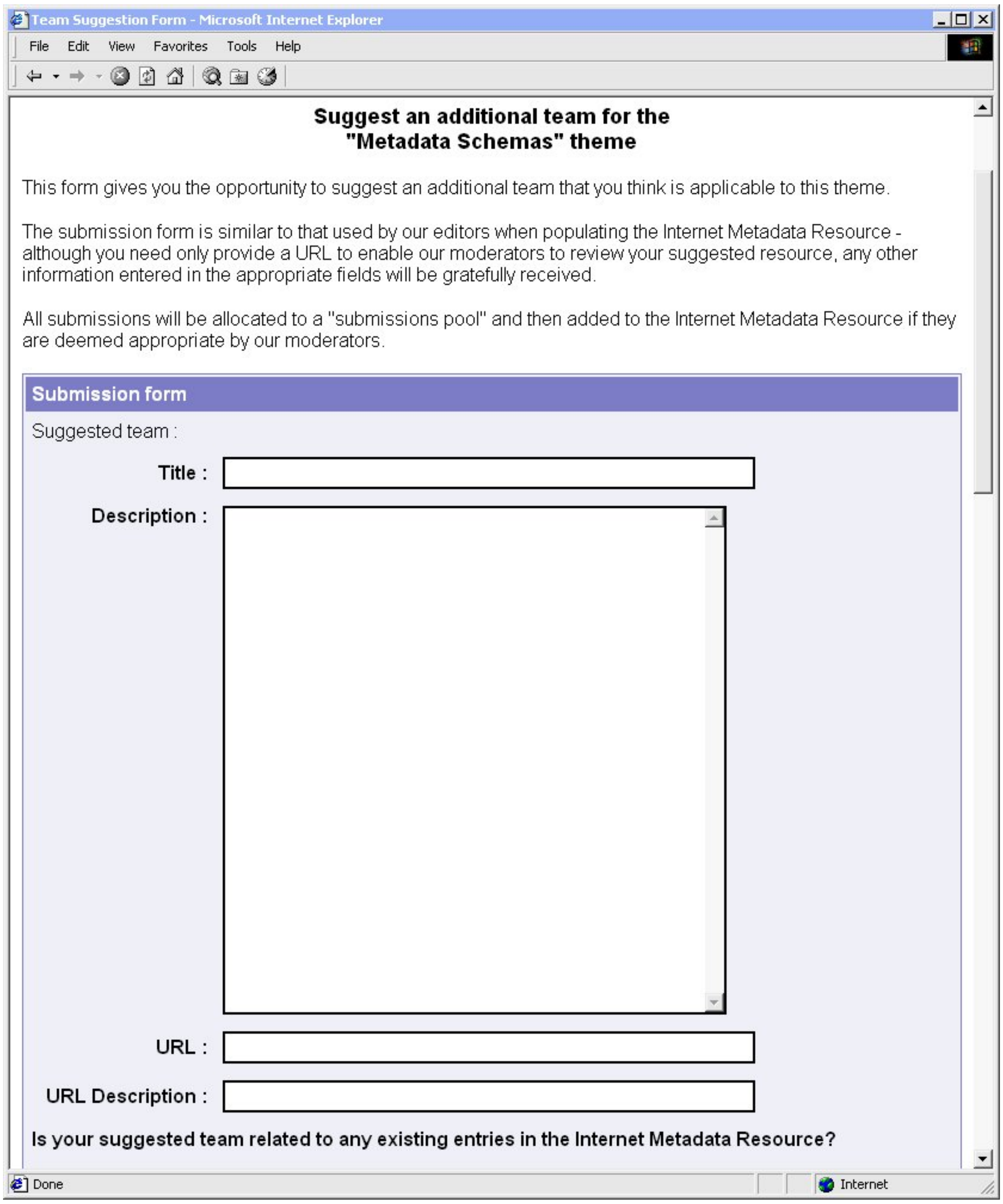
The RDU is a part of the Distriuted Systems Technology Centre, a government sponsored research centre in Australia.

Relations

- **Researchers**
  - Jane Hunter
  - Renato Iannella
- **Literature**
  - A Comparison of Schemas for Video Metadata Representation
  - Mostly Metadata - A Bit Smarter Technology
- **Projects**
  - MAENAD - Multimedia Metadata Project

Done Local intranet

Figure 30: Following an ontological relationship to a team resource


Team Suggestion Form - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Done
Internet

Suggest an additional team for the "Metadata Schemas" theme

This form gives you the opportunity to suggest an additional team that you think is applicable to this theme.

The submission form is similar to that used by our editors when populating the Internet Metadata Resource - although you need only provide a URL to enable our moderators to review your suggested resource, any other information entered in the appropriate fields will be gratefully received.

All submissions will be allocated to a "submissions pool" and then added to the Internet Metadata Resource if they are deemed appropriate by our moderators.

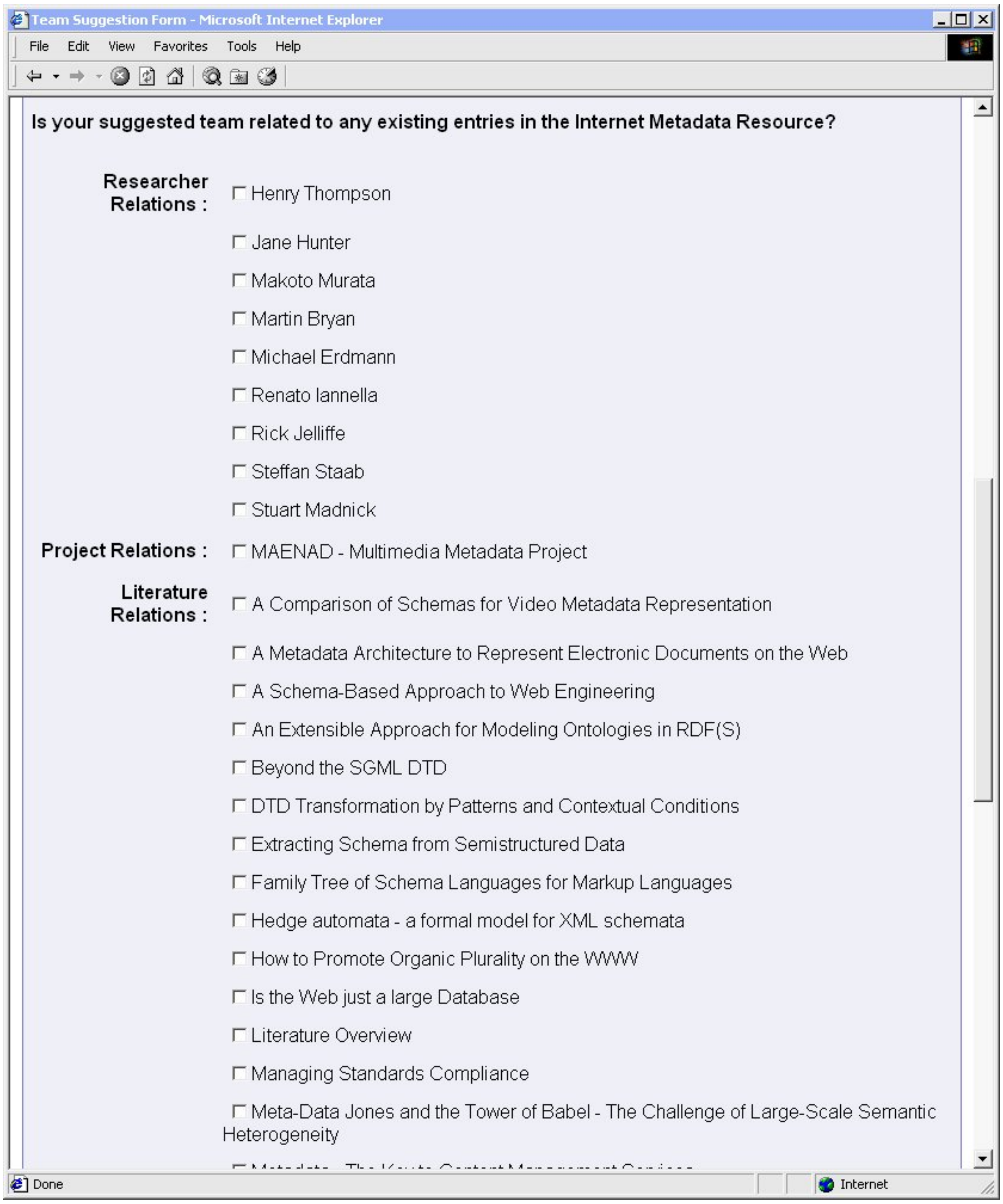
Submission form

Suggested team :

Title :	
Description :	
URL :	
URL Description :	

Is your suggested team related to any existing entries in the Internet Metadata Resource?

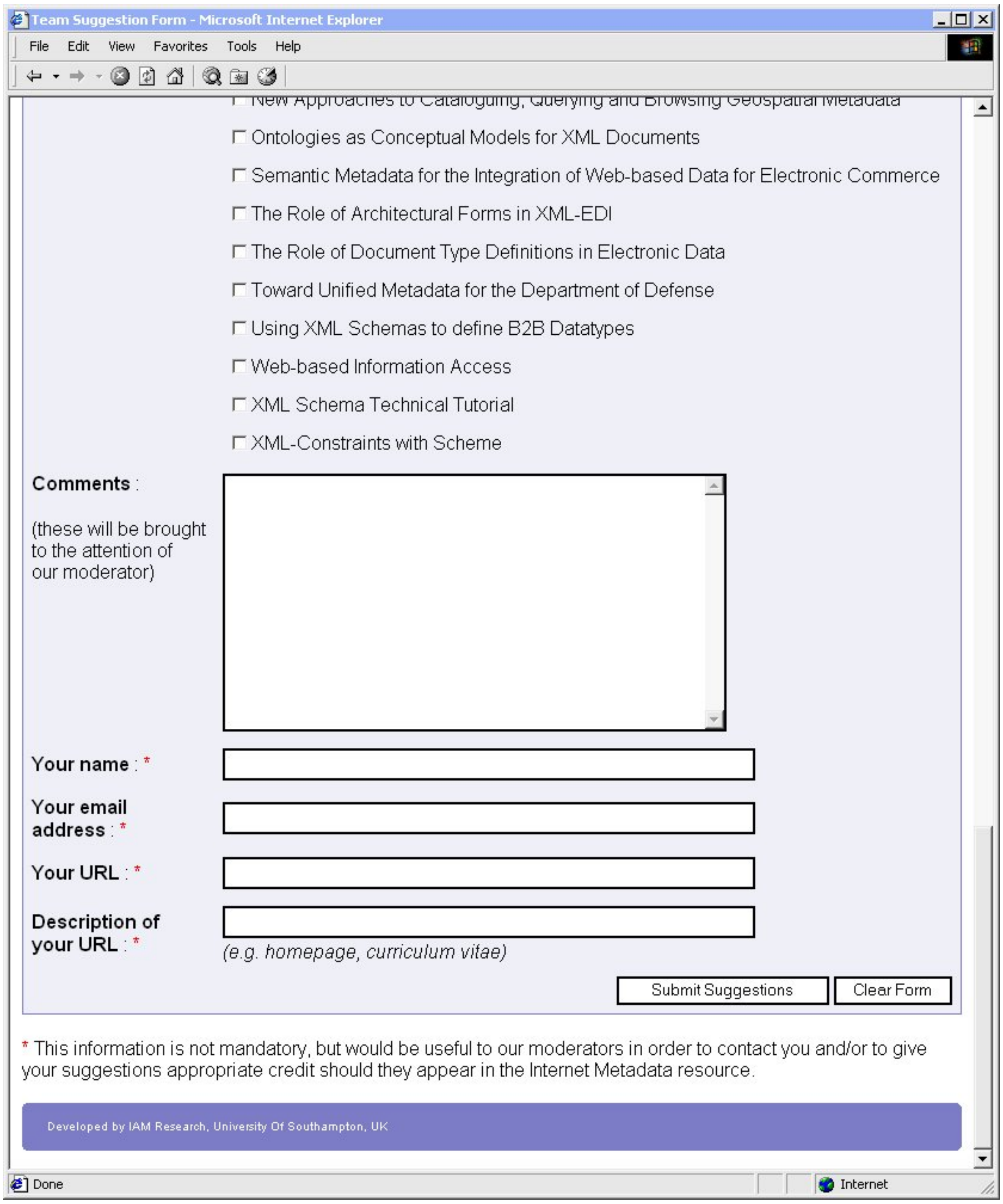
Figure 31: Suggesting a team resource that is appropriate to this theme (facility available to all users)


Team Suggestion Form - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Done
Internet

Is your suggested team related to any existing entries in the Internet Metadata Resource?

Researcher Relations :	☐ Henry Thompson
	☐ Jane Hunter
	☐ Makoto Murata
	☐ Martin Bryan
	☐ Michael Erdmann
	☐ Renato Iannella
	☐ Rick Jelliffe
	☐ Steffan Staab
	☐ Stuart Madnick
Project Relations :	☐ MAENAD - Multimedia Metadata Project
Literature Relations :	☐ A Comparison of Schemas for Video Metadata Representation
	☐ A Metadata Architecture to Represent Electronic Documents on the Web
	☐ A Schema-Based Approach to Web Engineering
	☐ An Extensible Approach for Modeling Ontologies in RDF(S)
	☐ Beyond the SGML DTD
	☐ DTD Transformation by Patterns and Contextual Conditions
	☐ Extracting Schema from Semistructured Data
	☐ Family Tree of Schema Languages for Markup Languages
	☐ Hedge automata - a formal model for XML schemata
	☐ How to Promote Organic Plurality on the WWW
	☐ Is the Web just a large Database
	☐ Literature Overview
	☐ Managing Standards Compliance
	☐ Meta-Data Jones and the Tower of Babel - The Challenge of Large-Scale Semantic Heterogeneity

Figure 32: Associating the suggested team resource with other resources in the theme


Team Suggestion Form - Microsoft Internet Explorer
File Edit View Favorites Tools Help

☐ New Approaches to Cataloguing, Querying and Browsing Geospatial Metadata

☐ Ontologies as Conceptual Models for XML Documents

☐ Semantic Metadata for the Integration of Web-based Data for Electronic Commerce

☐ The Role of Architectural Forms in XML-EDI

☐ The Role of Document Type Definitions in Electronic Data

☐ Toward Unified Metadata for the Department of Defense

☐ Using XML Schemas to define B2B Datatypes

☐ Web-based Information Access

☐ XML Schema Technical Tutorial

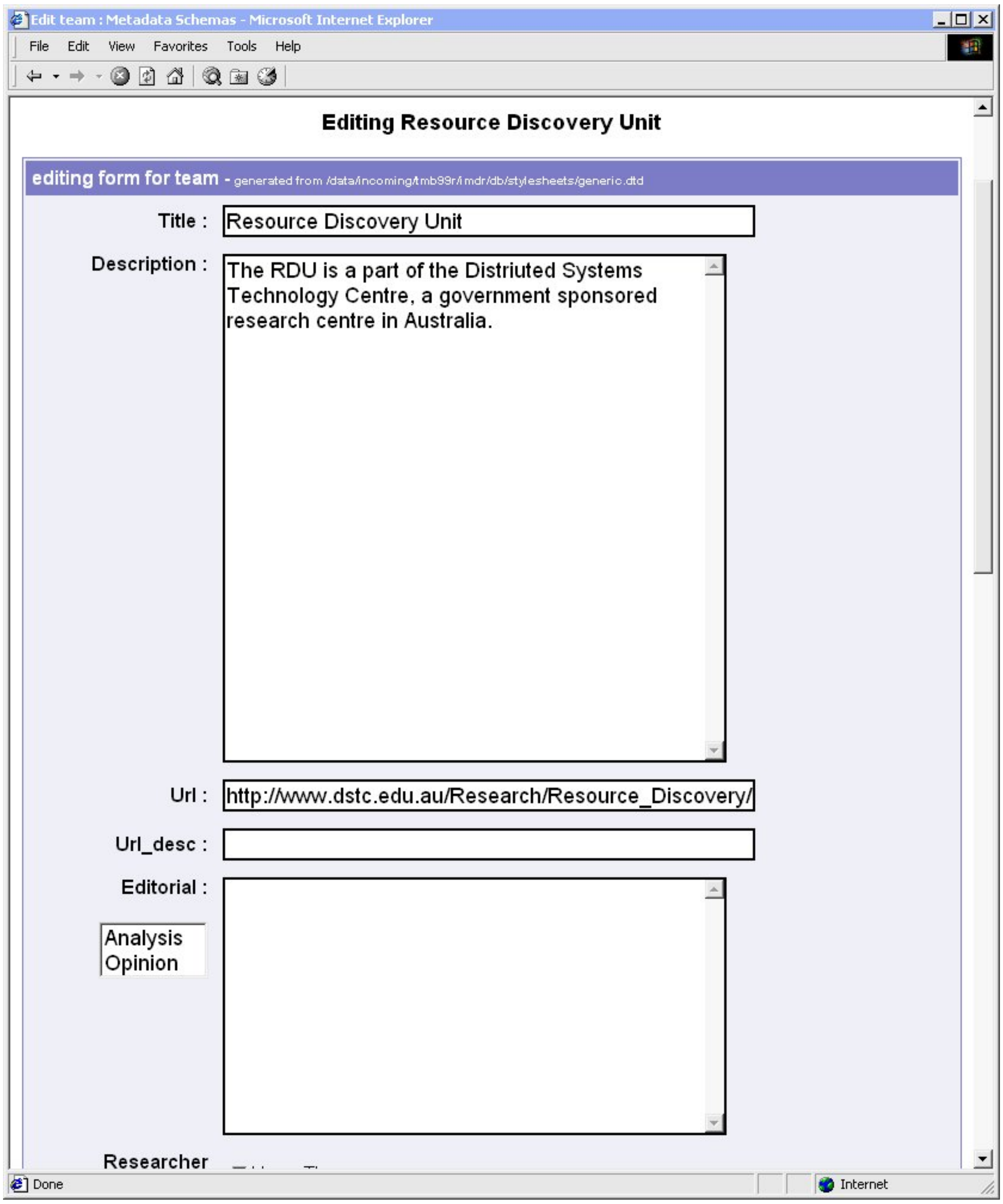
☐ XML-Constraints with Scheme

Comments : (these will be brought to the attention of our moderator)	
Your name : *	
Your email address : *	
Your URL : *	
Description of your URL : *	*(e.g. homepage, curriculum vitae)*

* This information is not mandatory, but would be useful to our moderators in order to contact you and/or to give your suggestions appropriate credit should they appear in the Internet Metadata resource.

Developed by IAM Research, University Of Southampton, UK

Figure 33: Submitting the suggested team resource


Edit team : Metadata Schemas - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Done
Internet

Editing Resource Discovery Unit

editing form for team - generated from /data/incoming/tmb99r/imdr/db/stylesheets/generic.dtd

Title : Resource Discovery Unit

Description : The RDU is a part of the Distriuted Systems Technology Centre, a government sponsored research centre in Australia.

Url : http://www.dstc.edu.au/Research/Resource_Discovery/

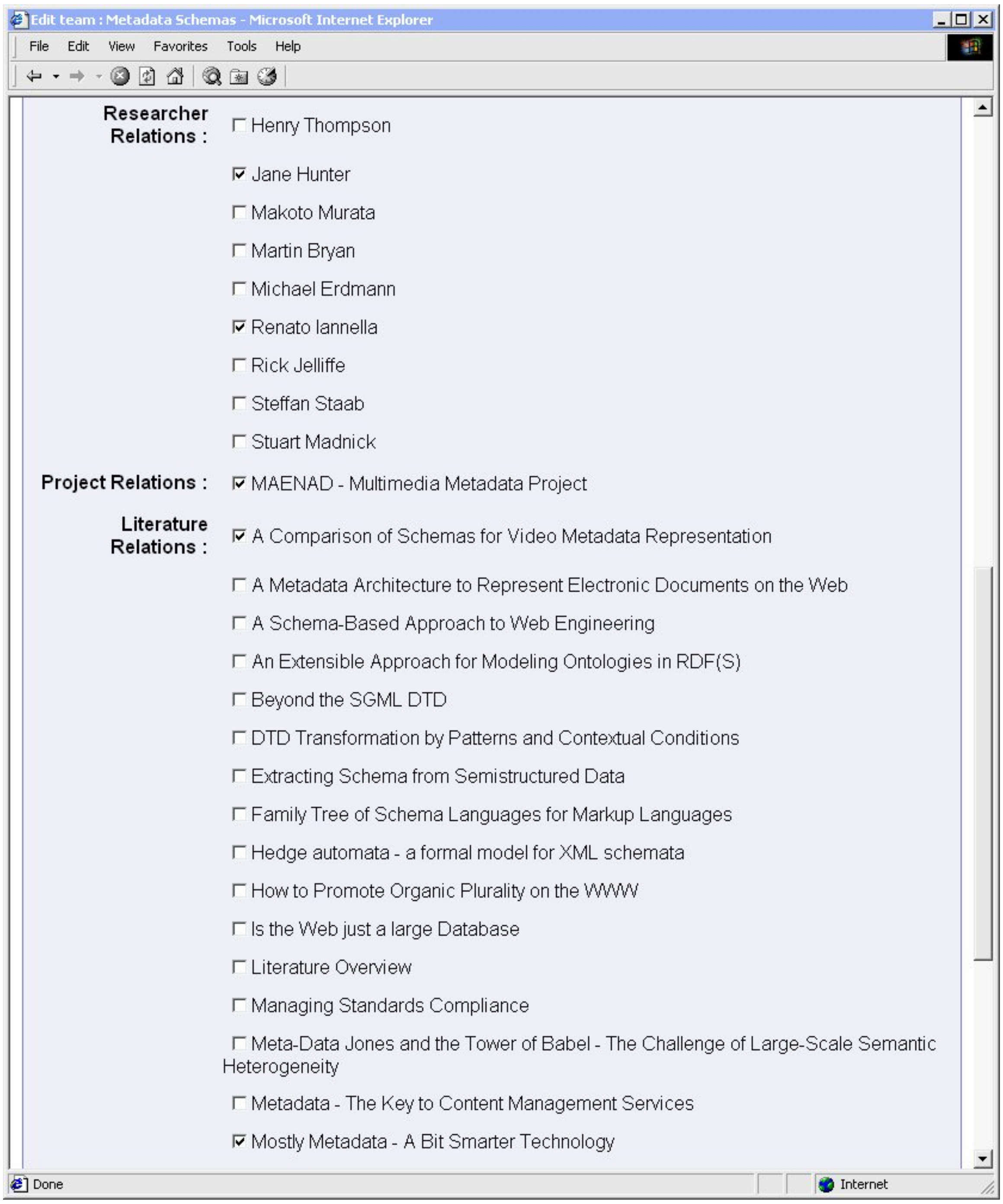
Url_desc :

Editorial :

Analysis
Opinion

Researcher

Figure 34: Editing a team resource (facility available to theme authors only)

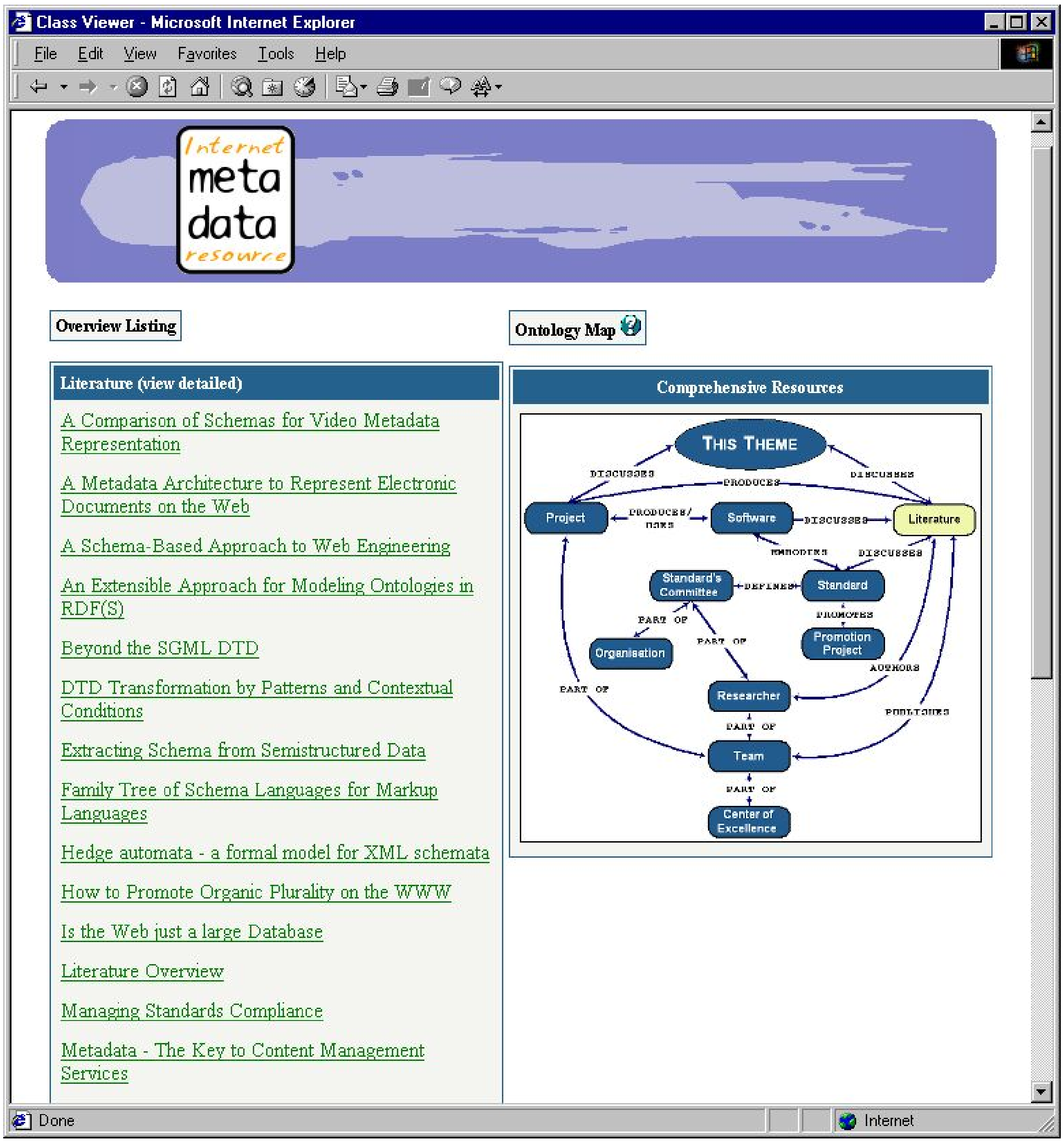

Edit team : Metadata Schemas - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Done
Internet

Researcher Relations :
- ☐ Henry Thompson
- ☑ Jane Hunter
- ☐ Makoto Murata
- ☐ Martin Bryan
- ☐ Michael Erdmann
- ☑ Renato Iannella
- ☐ Rick Jelliffe
- ☐ Steffan Staab
- ☐ Stuart Madnick

Project Relations :
- ☑ MAENAD - Multimedia Metadata Project

Literature Relations :
- ☑ A Comparison of Schemas for Video Metadata Representation
- ☐ A Metadata Architecture to Represent Electronic Documents on the Web
- ☐ A Schema-Based Approach to Web Engineering
- ☐ An Extensible Approach for Modeling Ontologies in RDF(S)
- ☐ Beyond the SGML DTD
- ☐ DTD Transformation by Patterns and Contextual Conditions
- ☐ Extracting Schema from Semistructured Data
- ☐ Family Tree of Schema Languages for Markup Languages
- ☐ Hedge automata - a formal model for XML schemata
- ☐ How to Promote Organic Plurality on the WWW
- ☐ Is the Web just a large Database
- ☐ Literature Overview
- ☐ Managing Standards Compliance
- ☐ Meta-Data Jones and the Tower of Babel - The Challenge of Large-Scale Semantic Heterogeneity
- ☐ Metadata - The Key to Content Management Services
- ☑ Mostly Metadata - A Bit Smarter Technology

Figure 35: Editing the relationships between a team resource and other resources in the theme



Literature (view detailed)

A Comparison of Schemas for Video Metadata Representation

A Metadata Architecture to Represent Electronic Documents on the Web

A Schema-Based Approach to Web Engineering

An Extensible Approach for Modeling Ontologies in RDF(S)

Beyond the SGML DTD

DTD Transformation by Patterns and Contextual Conditions

Extracting Schema from Semistructured Data

Family Tree of Schema Languages for Markup Languages

Hedge automata - a formal model for XML schemata

How to Promote Organic Plurality on the WWW

Is the Web just a large Database

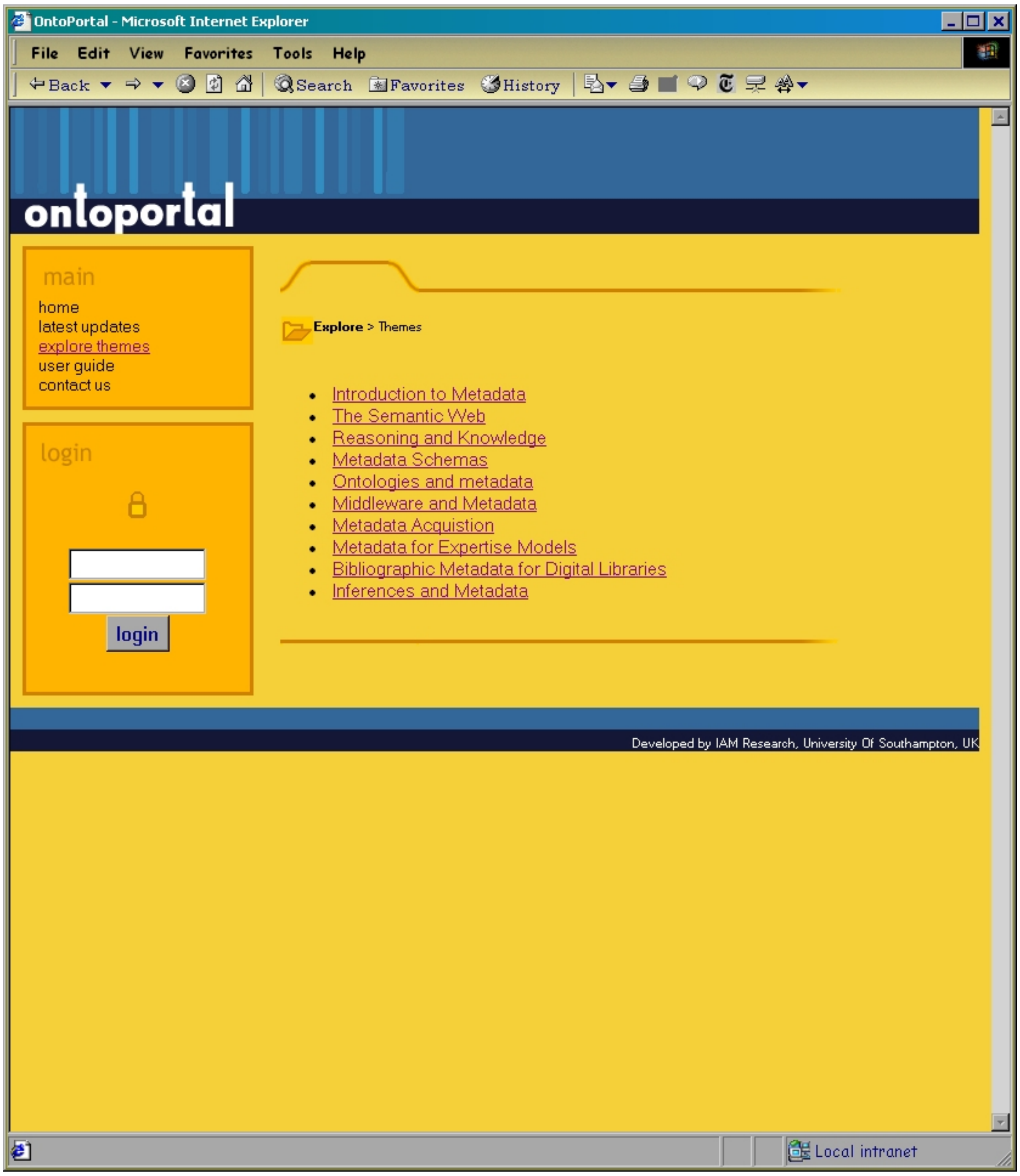
Literature Overview

Managing Standards Compliance

Metadata - The Key to Content Management Services

Figure 36: Using the ontological navigation tool to navigate a theme's concepts

B OntoPortal Version 2 Screenshots

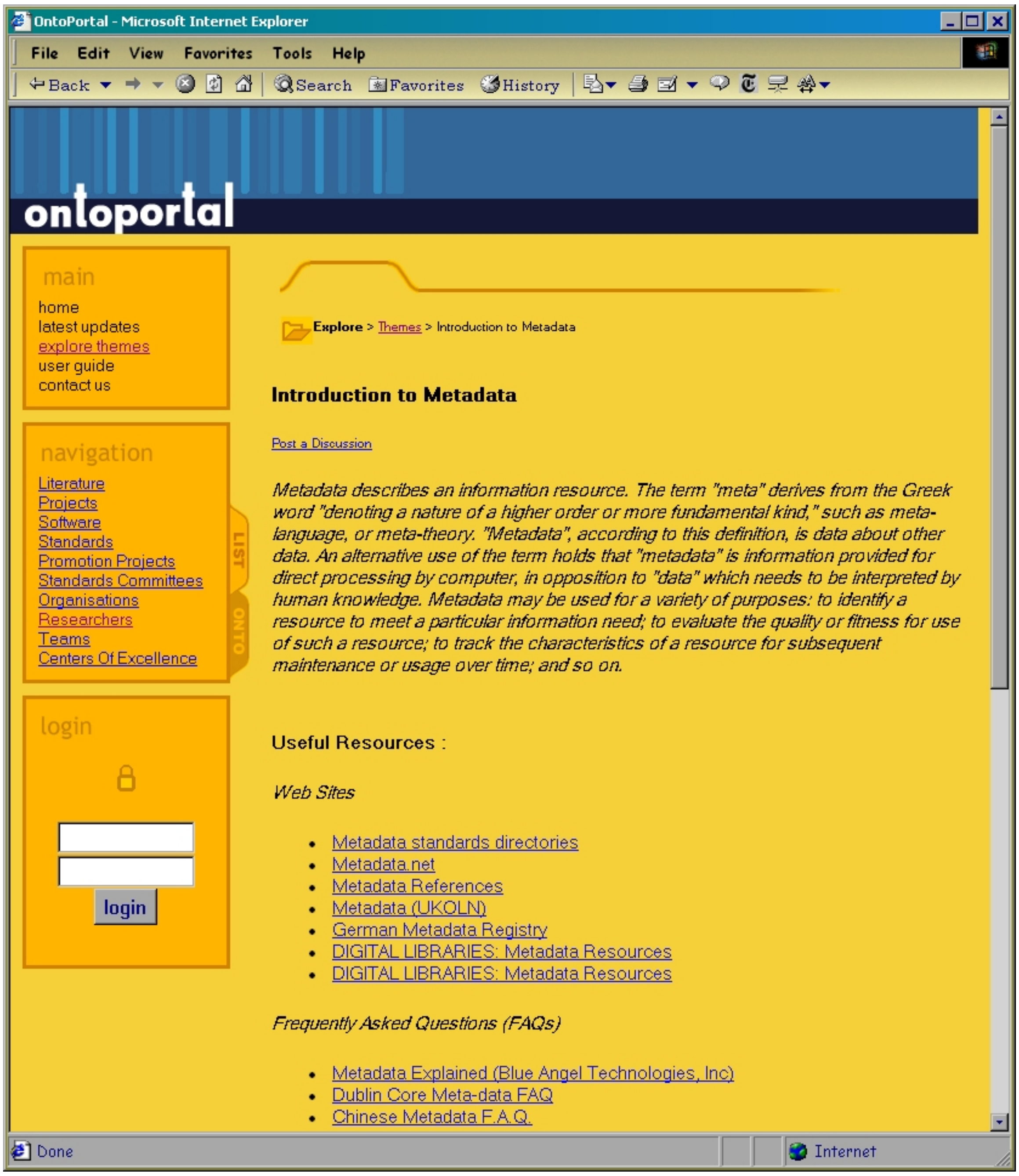

OntoPortal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
ontoportal
main
home
latest updates
explore themes
user guide
contact us
login
login
Local intranet

Explore > Themes

- Introduction to Metadata
- The Semantic Web
- Reasoning and Knowledge
- Metadata Schemas
- Ontologies and metadata
- Middleware and Metadata
- Metadata Acquistion
- Metadata for Expertise Models
- Bibliographic Metadata for Digital Libraries
- Inferences and Metadata

Developed by IAM Research, University Of Southampton, UK

Figure 37: Initially, the user is presented with a list of available themes.



Introduction to Metadata

Post a Discussion

Metadata describes an information resource. The term "meta" derives from the Greek word "denoting a nature of a higher order or more fundamental kind," such as meta-language, or meta-theory. "Metadata", according to this definition, is data about other data. An alternative use of the term holds that "metadata" is information provided for direct processing by computer, in opposition to "data" which needs to be interpreted by human knowledge. Metadata may be used for a variety of purposes: to identify a resource to meet a particular information need; to evaluate the quality or fitness for use of such a resource; to track the characteristics of a resource for subsequent maintenance or usage over time; and so on.

Useful Resources :

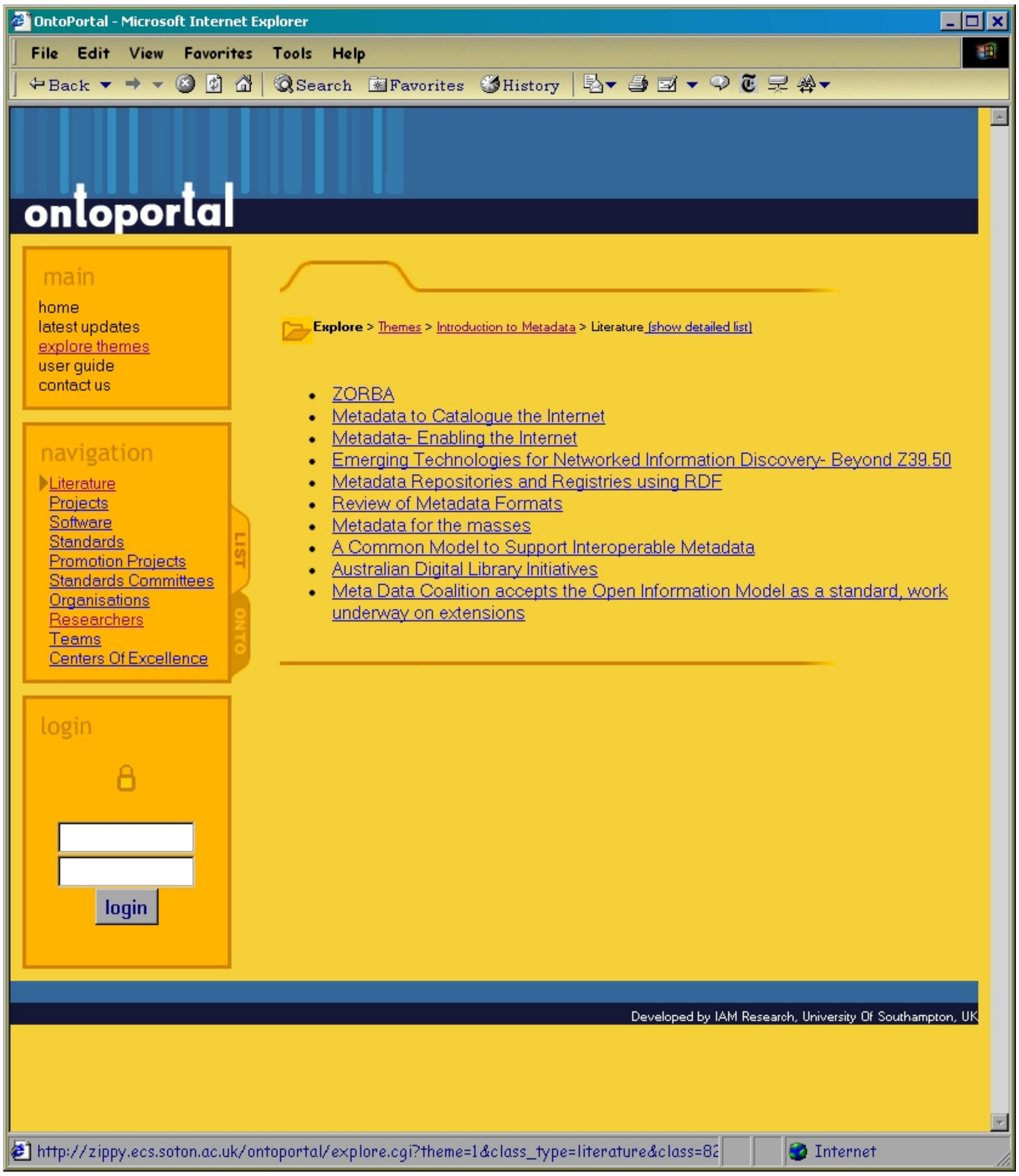
Web Sites

- Metadata standards directories
- Metadata.net
- Metadata References
- Metadata (UKOLN)
- German Metadata Registry
- DIGITAL LIBRARIES: Metadata Resources
- DIGITAL LIBRARIES: Metadata Resources

Frequently Asked Questions (FAQs)

- Metadata Explained (Blue Angel Technologies, Inc)
- Dublin Core Meta-data FAQ
- Chinese Metadata F.A.Q.

Figure 38: Upon selecting a theme, the user is presented with background information to explain the theme.

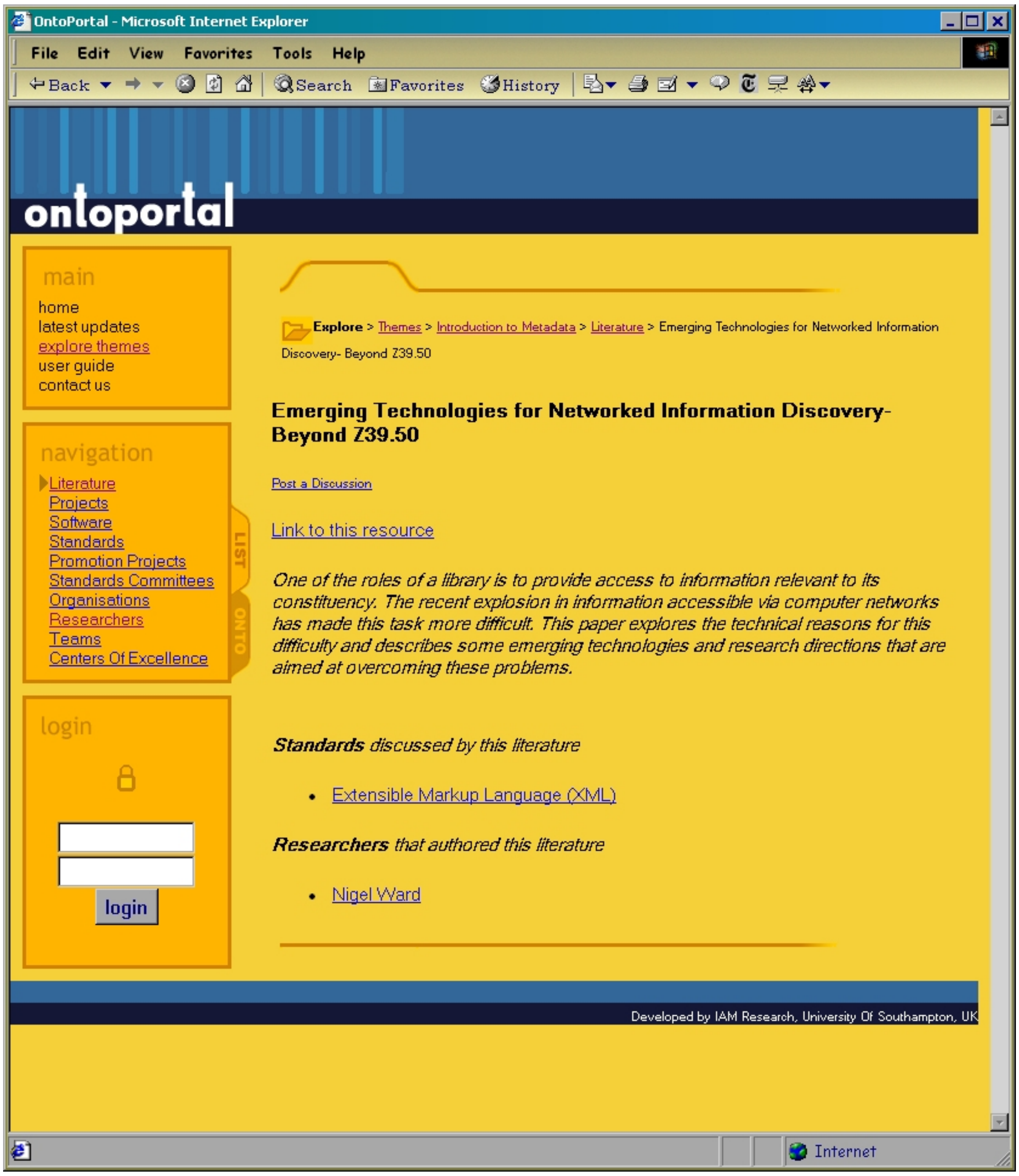


Explore > Themes > Introduction to Metadata > Literature [show detailed list]

- ZORBA
- Metadata to Catalogue the Internet
- Metadata- Enabling the Internet
- Emerging Technologies for Networked Information Discovery- Beyond Z39.50
- Metadata Repositories and Registries using RDF
- Review of Metadata Formats
- Metadata for the masses
- A Common Model to Support Interoperable Metadata
- Australian Digital Library Initiatives
- Meta Data Coalition accepts the Open Information Model as a standard, work underway on extensions

http://zippy.ecs.soton.ac.uk/ontoportal/explore.cgi?theme=1&class_type=literature&class=82

Figure 39: The user selects *Literature* from the navigation menu and is presented with a list of all literature in the current theme.



Explore > Themes > Introduction to Metadata > Literature > Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Emerging Technologies for Networked Information Discovery-Beyond Z39.50

Post a Discussion

Link to this resource

One of the roles of a library is to provide access to information relevant to its constituency. The recent explosion in information accessible via computer networks has made this task more difficult. This paper explores the technical reasons for this difficulty and describes some emerging technologies and research directions that are aimed at overcoming these problems.

***Standards** discussed by this literature*

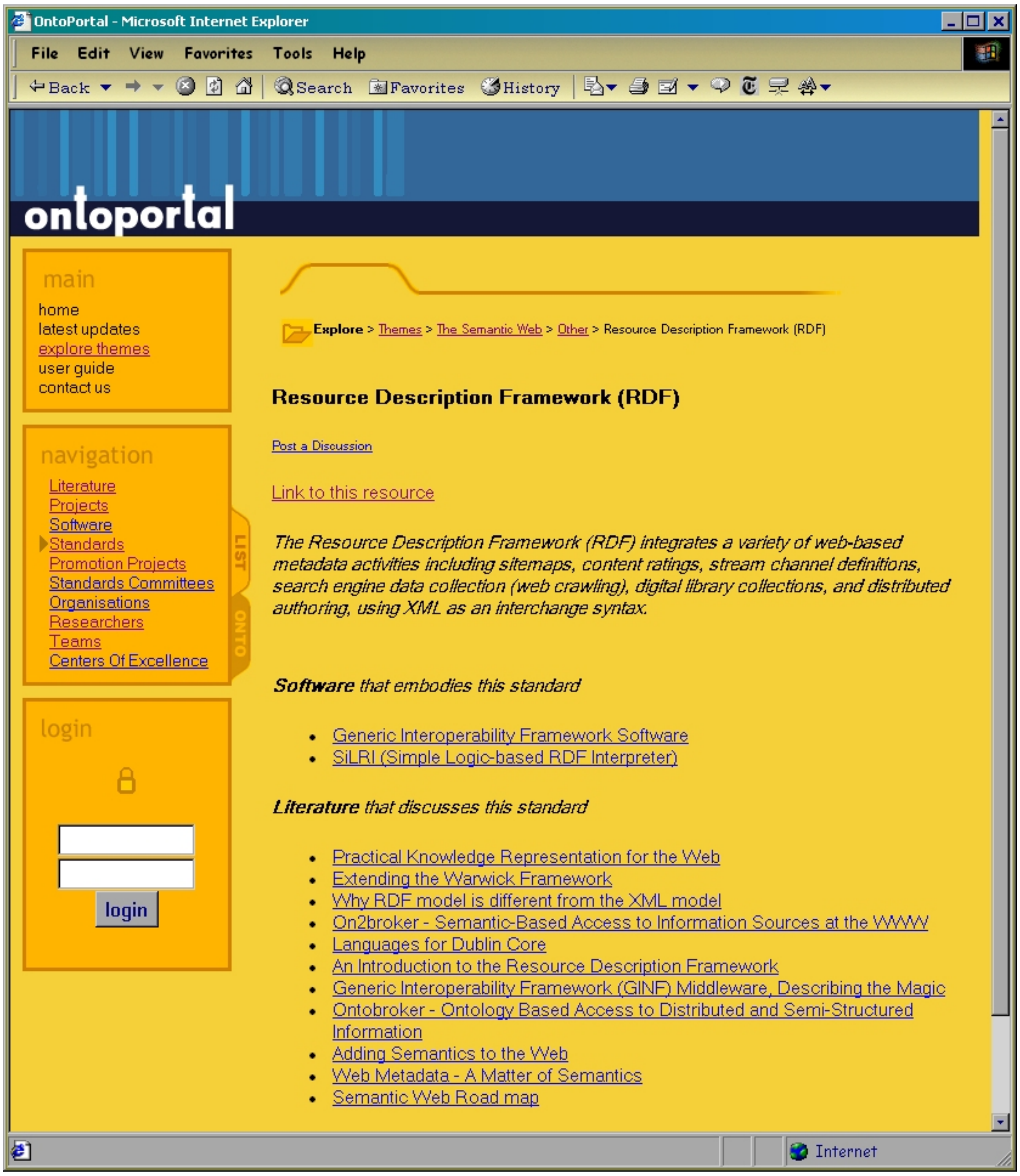
- Extensible Markup Language (XML)

***Researchers** that authored this literature*

- Nigel Ward

Developed by IAM Research, University Of Southampton, UK

Figure 40: Upon selected a particular literature instance, information on that resource as well as its relationships to other resources is displayed.



Explore > Themes > The Semantic Web > Other > Resource Description Framework (RDF)

Resource Description Framework (RDF)

Post a Discussion

Link to this resource

The Resource Description Framework (RDF) integrates a variety of web-based metadata activities including sitemaps, content ratings, stream channel definitions, search engine data collection (web crawling), digital library collections, and distributed authoring, using XML as an interchange syntax.

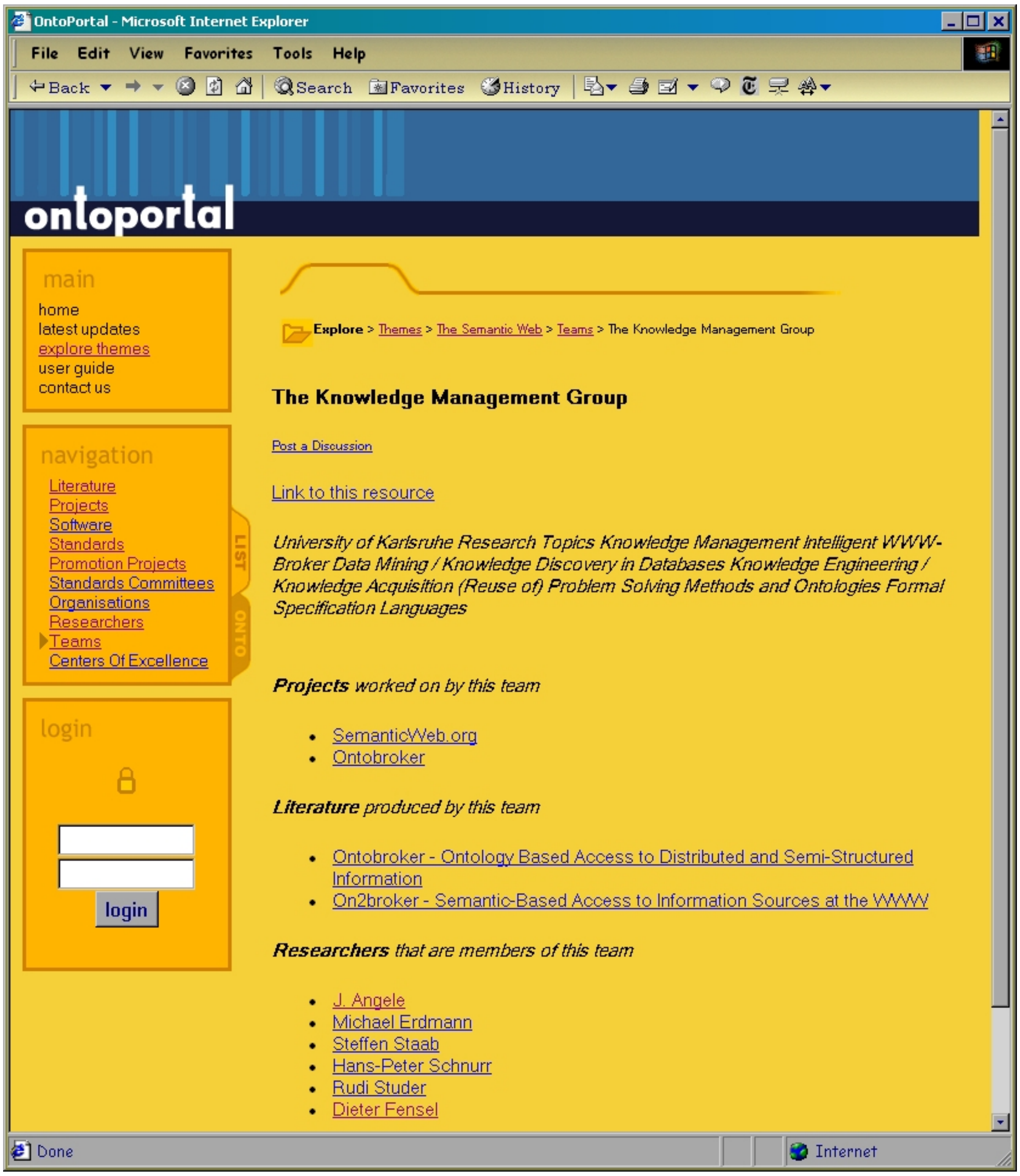
***Software** that embodies this standard*

- Generic Interoperability Framework Software
- SiLRI (Simple Logic-based RDF Interpreter)

***Literature** that discusses this standard*

- Practical Knowledge Representation for the Web
- Extending the Warwick Framework
- Why RDF model is different from the XML model
- On2broker - Semantic-Based Access to Information Sources at the WWW
- Languages for Dublin Core
- An Introduction to the Resource Description Framework
- Generic Interoperability Framework (GINF) Middleware, Describing the Magic
- Ontobroker - Ontology Based Access to Distributed and Semi-Structured Information
- Adding Semantics to the Web
- Web Metadata - A Matter of Semantics
- Semantic Web Road map

Figure 41: The user investigates an instance of a standard.



Explore > Themes > The Semantic Web > Teams > The Knowledge Management Group

The Knowledge Management Group

Post a Discussion

Link to this resource

University of Karlsruhe Research Topics Knowledge Management Intelligent WWW-Broker Data Mining / Knowledge Discovery in Databases Knowledge Engineering / Knowledge Acquisition (Reuse of) Problem Solving Methods and Ontologies Formal Specification Languages

***Projects** worked on by this team*

- SemanticWeb.org
- Ontobroker

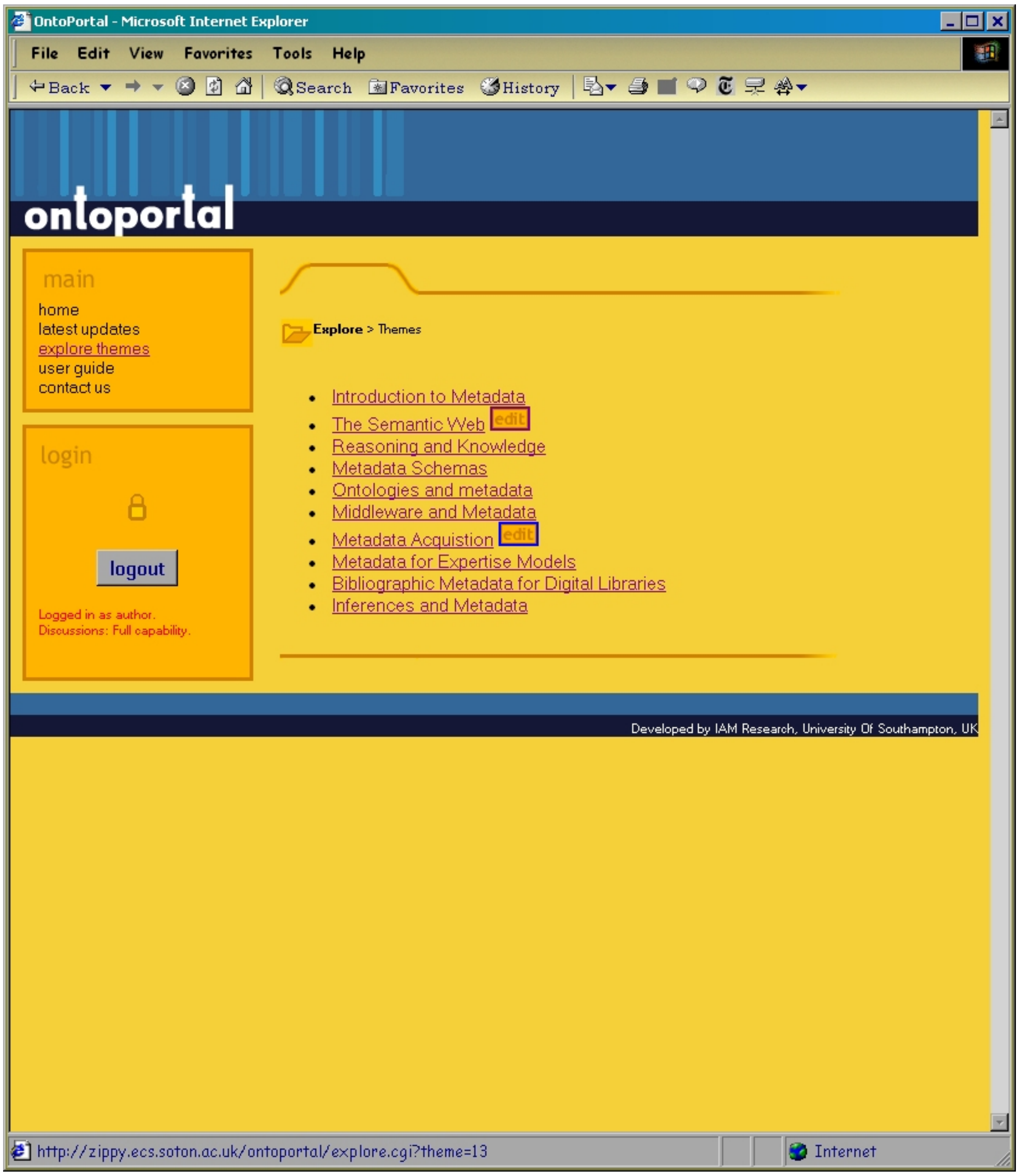
***Literature** produced by this team*

- Ontobroker - Ontology Based Access to Distributed and Semi-Structured Information
- On2broker - Semantic-Based Access to Information Sources at the WWW

***Researchers** that are members of this team*

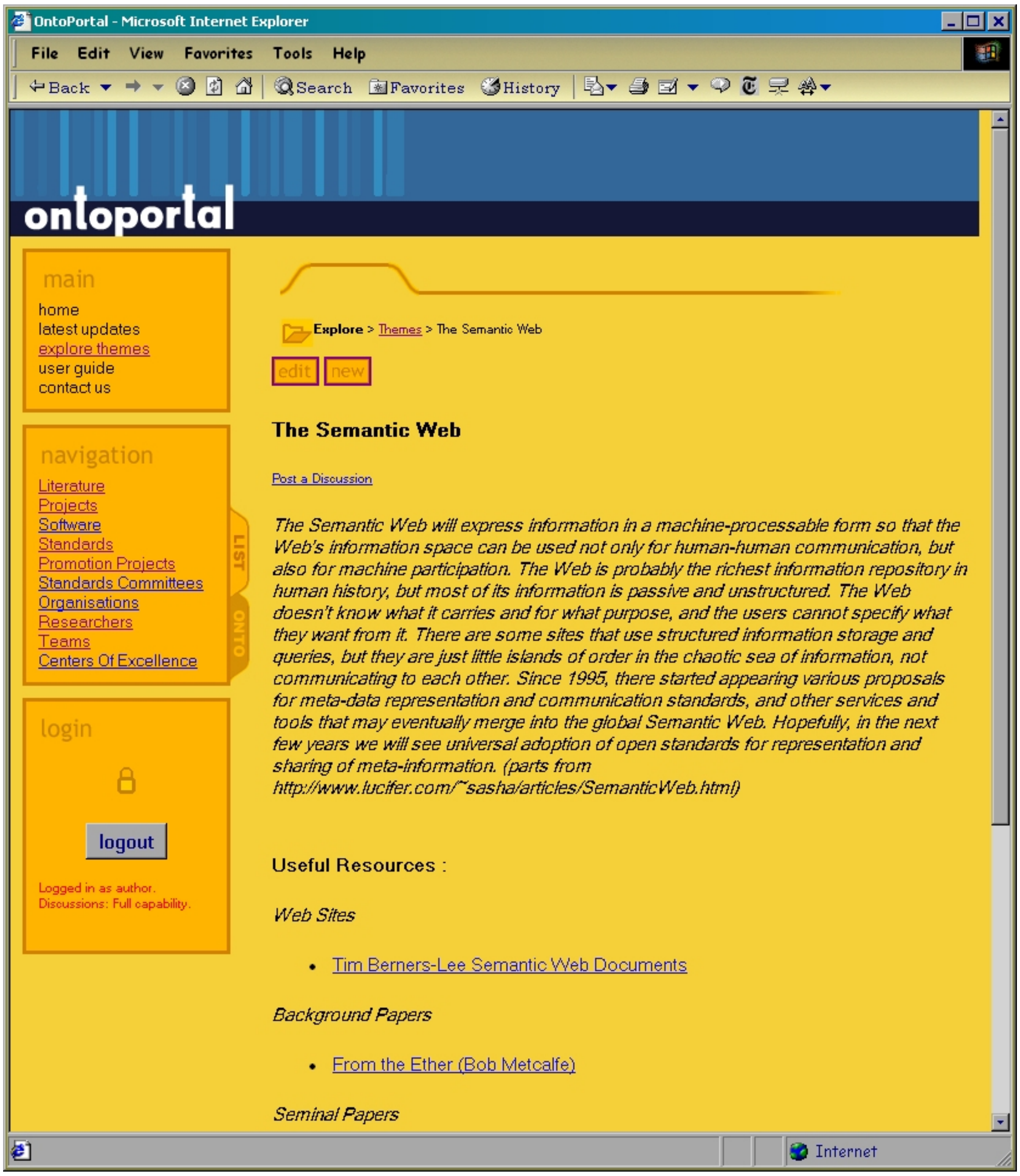
- J. Angele
- Michael Erdmann
- Steffen Staab
- Hans-Peter Schnurr
- Rudi Studer
- Dieter Fensel

Figure 42: The user investigates an instance of a research team.


OntoPortal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
ontoportal
main
home
latest updates
explore themes
user guide
contact us
login
logout
Logged in as author.
Discussions: Full capability.
Explore > Themes
Developed by IAM Research, University Of Southampton, UK
http://zippy.ecs.soton.ac.uk/ontoportal/explore.cgi?theme=13
Internet

- Introduction to Metadata
- The Semantic Web edit
- Reasoning and Knowledge
- Metadata Schemas
- Ontologies and metadata
- Middleware and Metadata
- Metadata Acquistion edit
- Metadata for Expertise Models
- Bibliographic Metadata for Digital Libraries
- Inferences and Metadata

Figure 43: Once the user logs in, a small edit icon is placed next to the themes for which she has authoring privileges.



The Semantic Web

Post a Discussion

The Semantic Web will express information in a machine-processable form so that the Web's information space can be used not only for human-human communication, but also for machine participation. The Web is probably the richest information repository in human history, but most of its information is passive and unstructured. The Web doesn't know what it carries and for what purpose, and the users cannot specify what they want from it. There are some sites that use structured information storage and queries, but they are just little islands of order in the chaotic sea of information, not communicating to each other. Since 1995, there started appearing various proposals for meta-data representation and communication standards, and other services and tools that may eventually merge into the global Semantic Web. Hopefully, in the next few years we will see universal adoption of open standards for representation and sharing of meta-information. (parts from http://www.lucifer.com/~sasha/articles/SemanticWeb.html)

Useful Resources :

Web Sites

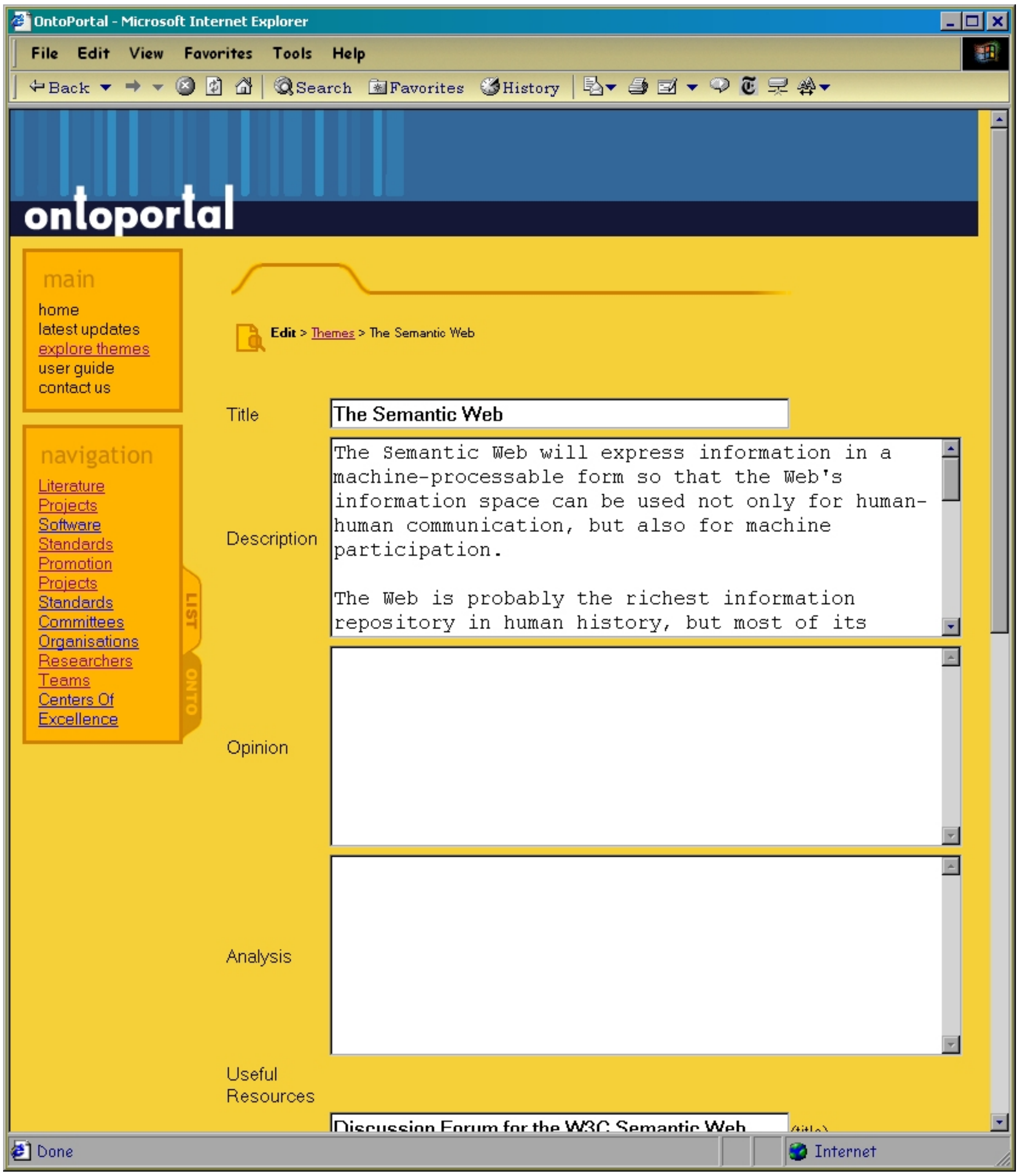
- Tim Berners-Lee Semantic Web Documents

Background Papers

- From the Ether (Bob Metcalfe)

Seminal Papers

Figure 44: The theme instance screen is also marked with authoring icons.



Title: The Semantic Web

Description: The Semantic Web will express information in a machine-processable form so that the Web's information space can be used not only for human-human communication, but also for machine participation.

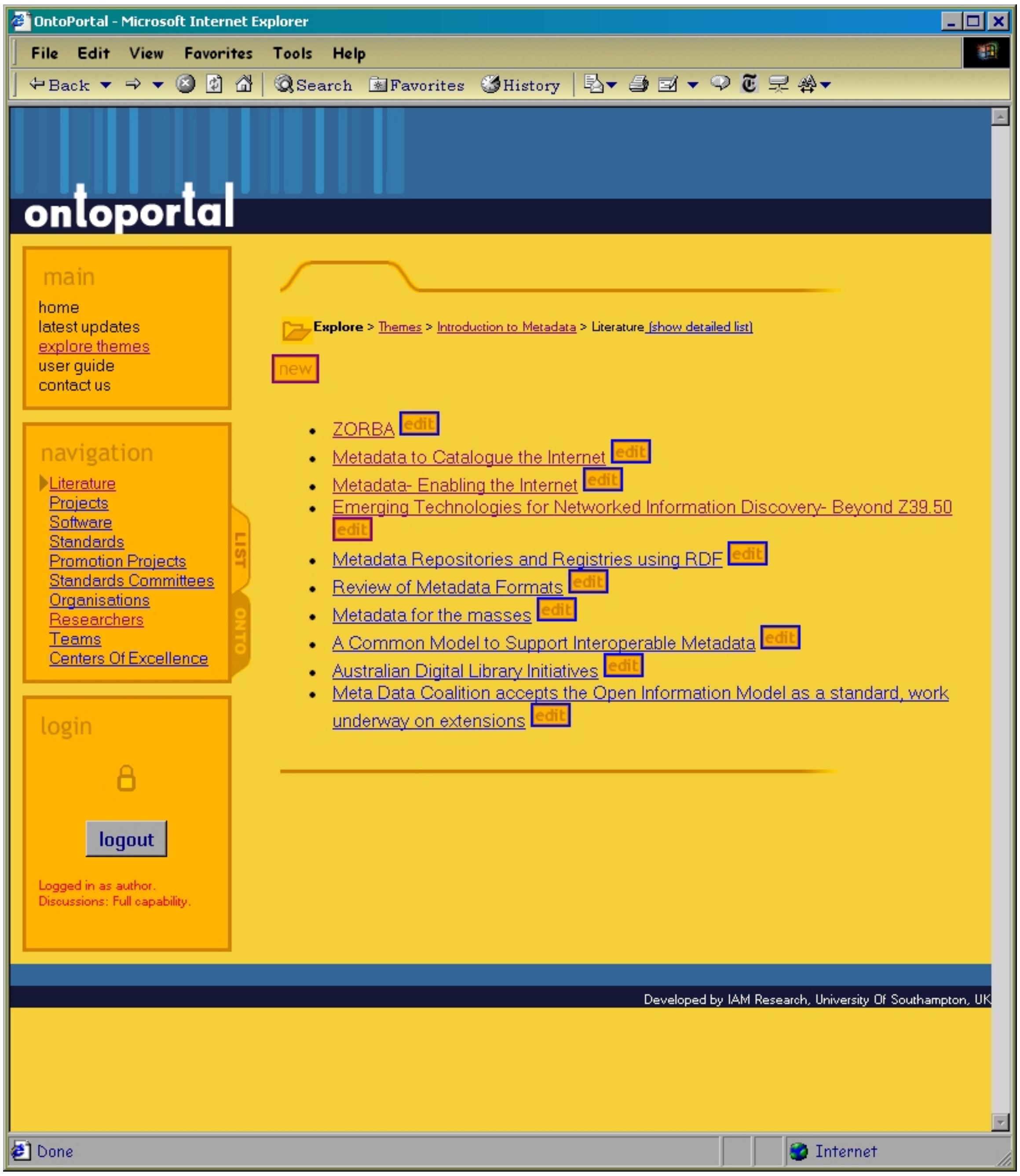
The Web is probably the richest information repository in human history, but most of its

Opinion:

Analysis:

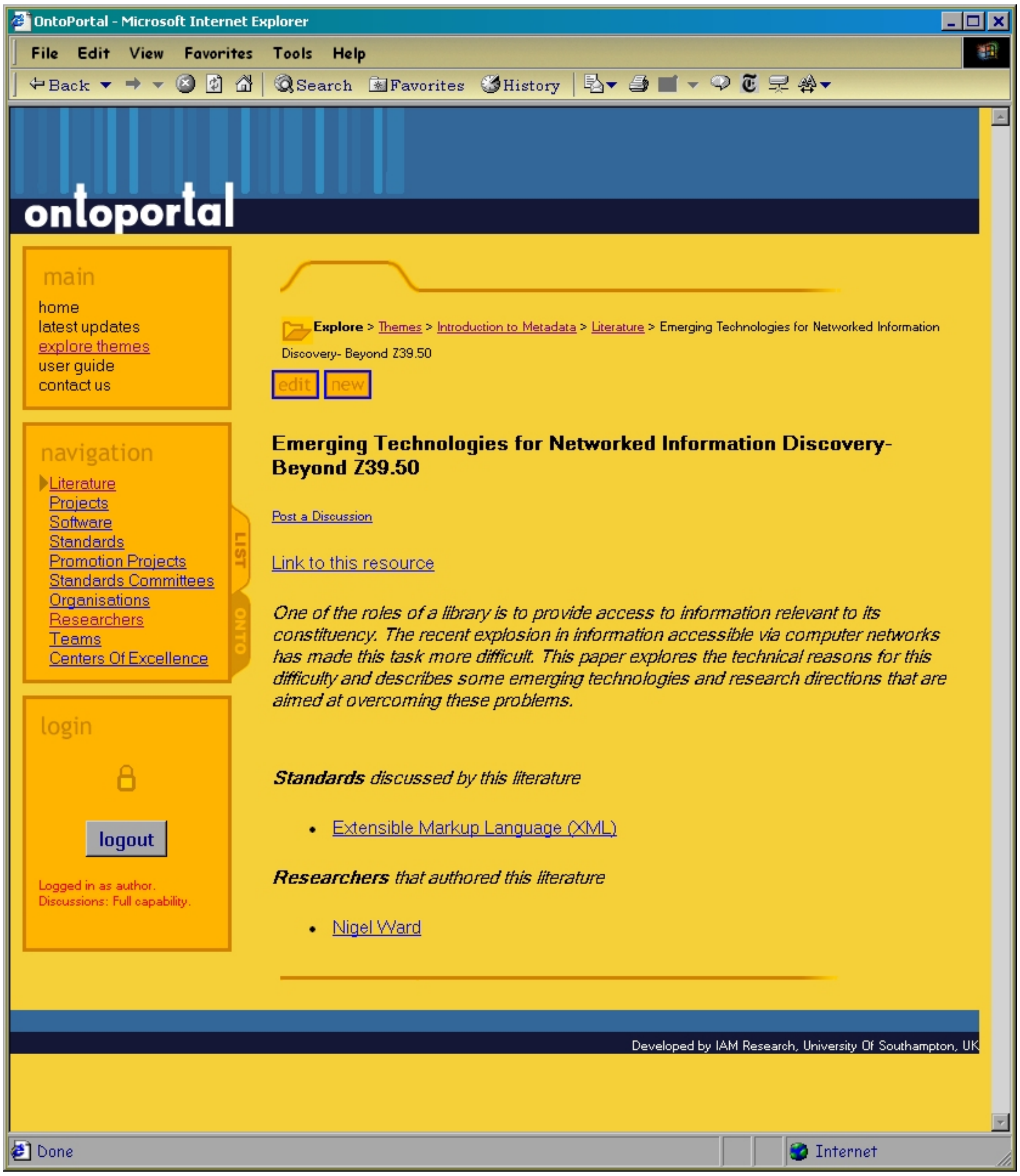
Useful Resources: Discussion Forum for the W3C Semantic Web

Figure 45: By selecting the edit icon in the theme information screen, the user is able to modify the properties of a theme.


OntoPortal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
ontoportal
main
home
latest updates
explore themes
user guide
contact us
navigation
Literature
Projects
Software
Standards
Promotion Projects
Standards Committees
Organisations
Researchers
Teams
Centers Of Excellence
LIST
ONTO
login
logout
Logged in as author.
Discussions: Full capability.
Explore > Themes > Introduction to Metadata > Literature [show detailed list]
new
Developed by IAM Research, University Of Southampton, UK
Done
Internet

- ZORBA edit
- Metadata to Catalogue the Internet edit
- Metadata- Enabling the Internet edit
- Emerging Technologies for Networked Information Discovery- Beyond Z39.50 edit
- Metadata Repositories and Registries using RDF edit
- Review of Metadata Formats edit
- Metadata for the masses edit
- A Common Model to Support Interoperable Metadata edit
- Australian Digital Library Initiatives edit
- Meta Data Coalition accepts the Open Information Model as a standard, work underway on extensions edit

Figure 46: The list of literature decorated with the authoring icons.



Logged in as author.
Discussions: Full capability.

Explore > Themes > Introduction to Metadata > Literature > Emerging Technologies for Networked Information Discovery- Beyond Z39.50

Emerging Technologies for Networked Information Discovery-Beyond Z39.50

One of the roles of a library is to provide access to information relevant to its constituency. The recent explosion in information accessible via computer networks has made this task more difficult. This paper explores the technical reasons for this difficulty and describes some emerging technologies and research directions that are aimed at overcoming these problems.

***Standards** discussed by this literature*

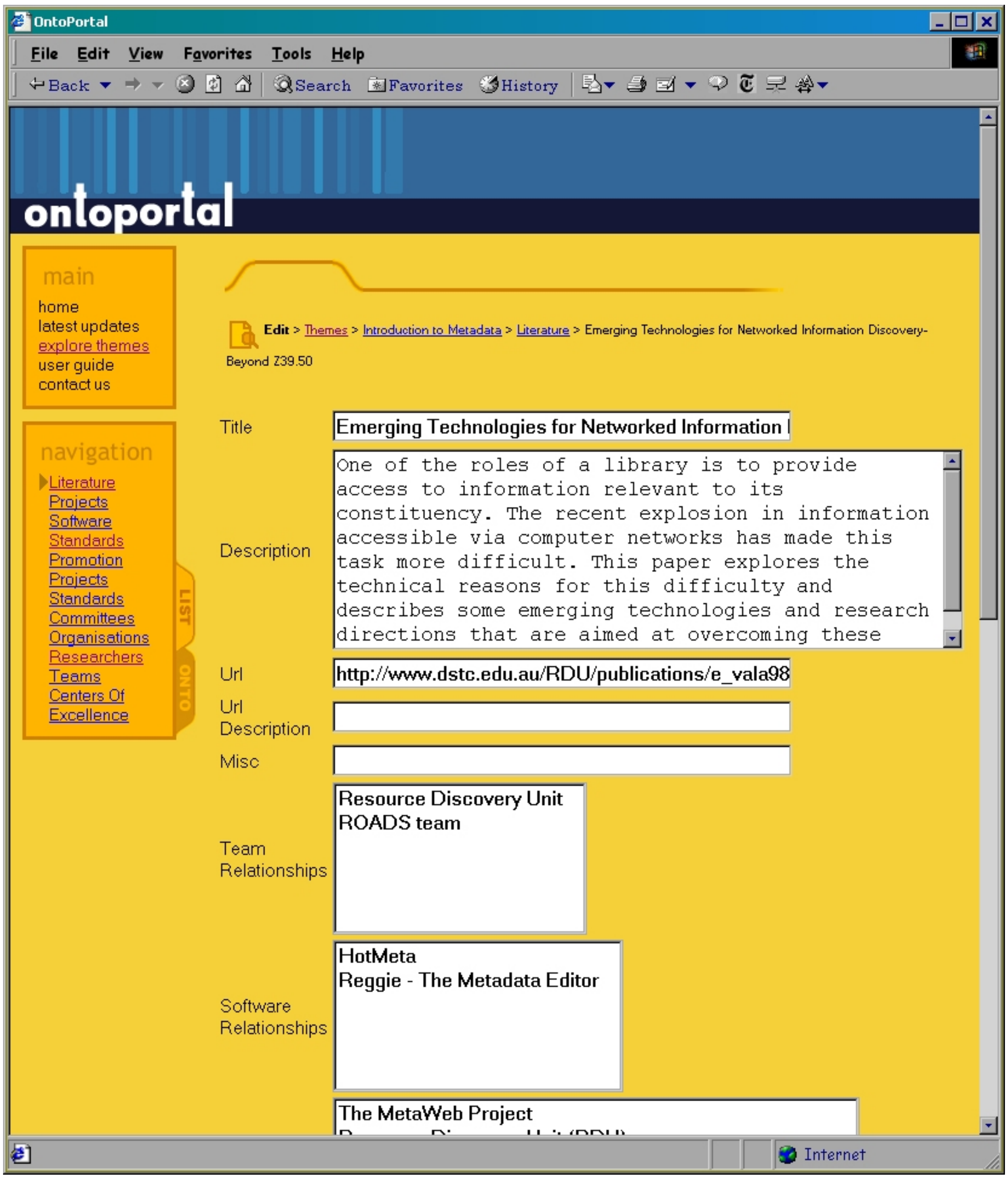
- Extensible Markup Language (XML)

***Researchers** that authored this literature*

- Nigel Ward

Developed by IAM Research, University Of Southampton, UK

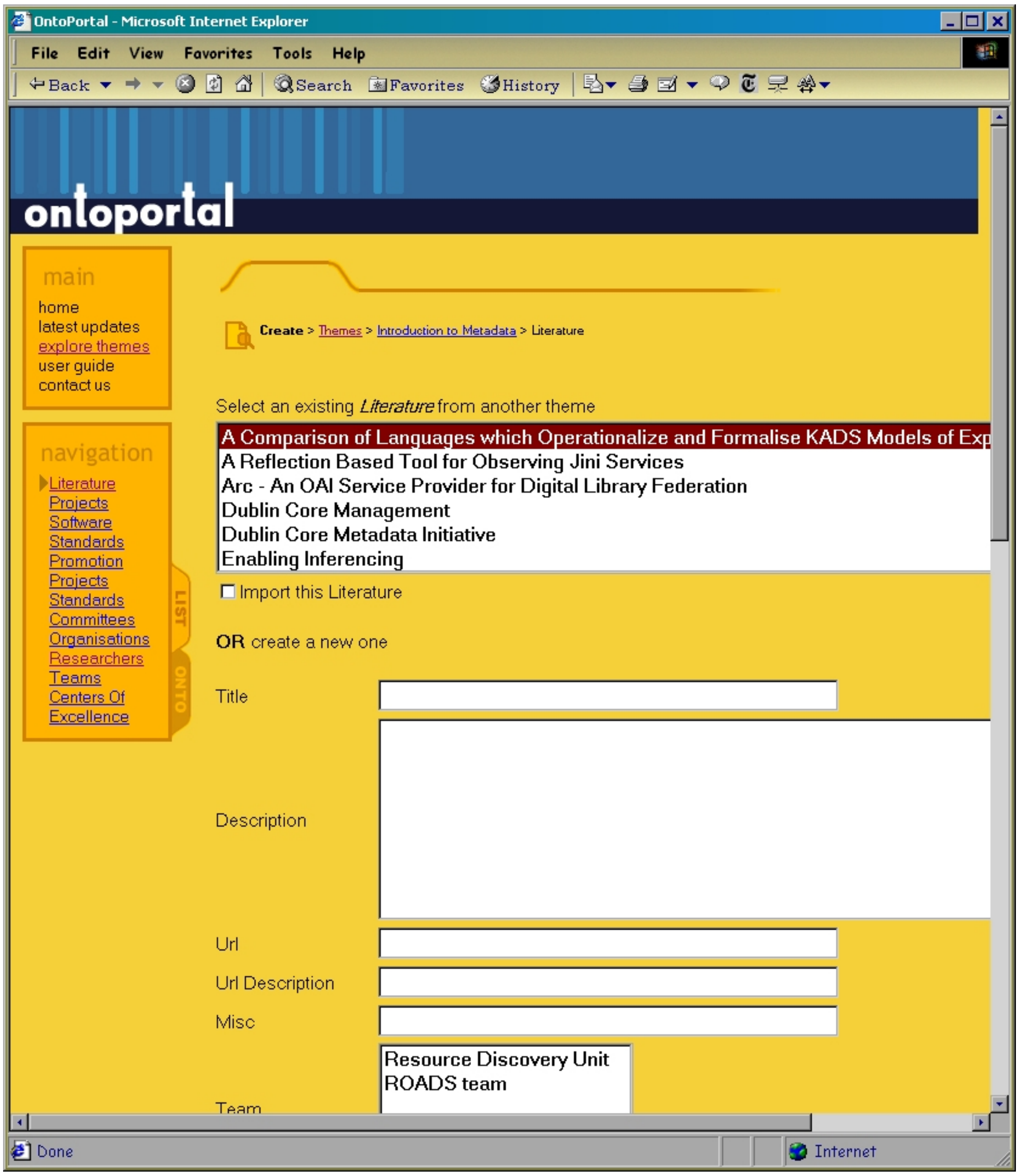
Figure 47: A literature instance decorated with the authoring icons.



Edit > Themes > Introduction to Metadata > Literature > Emerging Technologies for Networked Information Discovery-Beyond Z39.50

Title	Emerging Technologies for Networked Information I
Description	One of the roles of a library is to provide access to information relevant to its constituency. The recent explosion in information accessible via computer networks has made this task more difficult. This paper explores the technical reasons for this difficulty and describes some emerging technologies and research directions that are aimed at overcoming these
Url	http://www.dstc.edu.au/RDU/publications/e_vala98
Url Description	
Misc	
Team Relationships	Resource Discovery Unit ROADS team
Software Relationships	HotMeta Reggie - The Metadata Editor
	The MetaWeb Project

Figure 48: Selecting the edit icon for a literature instance, brings up a form with the property fields completed.



Create > Themes > Introduction to Metadata > Literature

Select an existing *Literature* from another theme

A Comparison of Languages which Operationalize and Formalise KADS Models of Exp
A Reflection Based Tool for Observing Jini Services
Arc - An OAI Service Provider for Digital Library Federation
Dublin Core Management
Dublin Core Metadata Initiative
Enabling Inferencing

☐ Import this Literature

OR create a new one

Title

Description

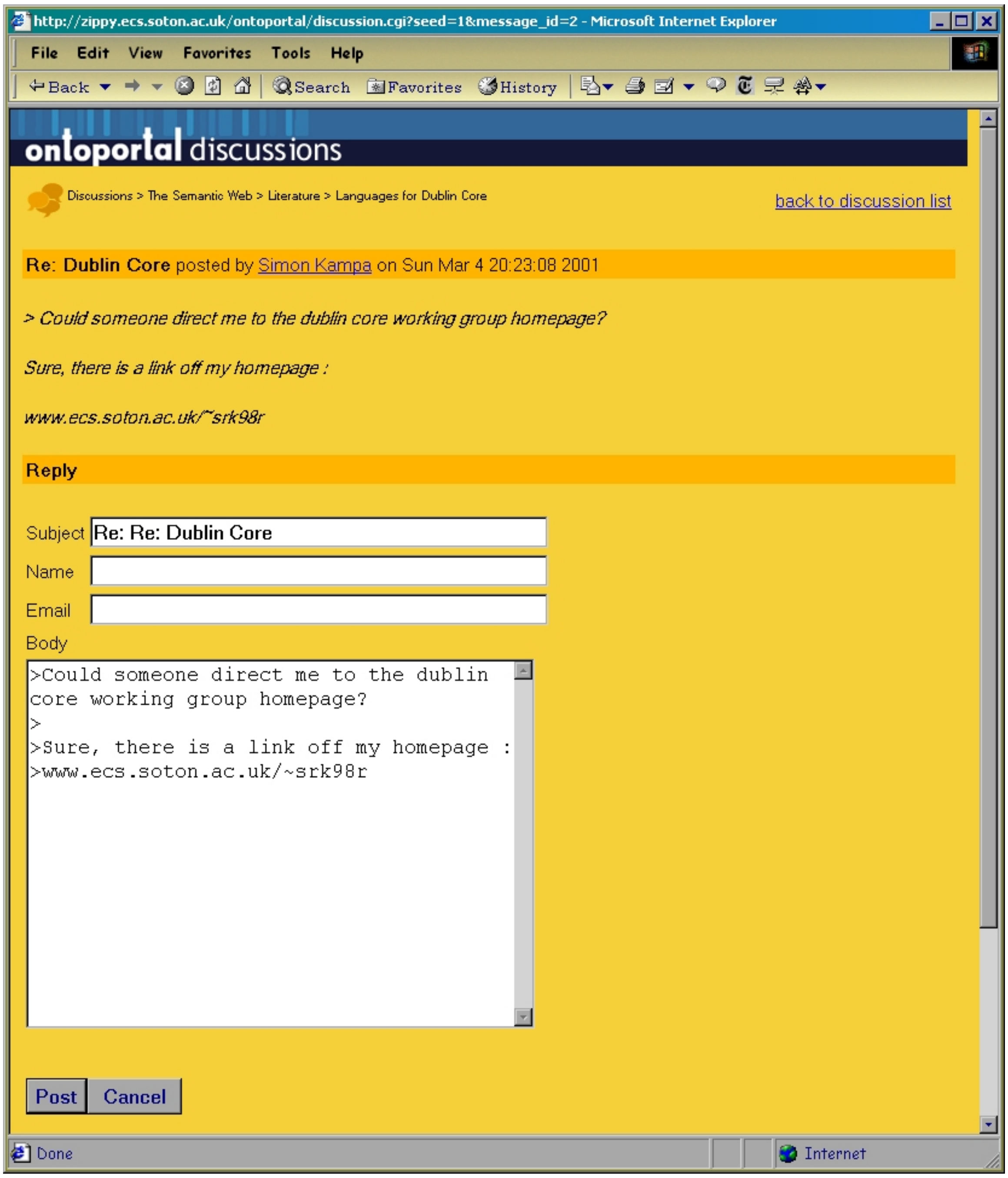
Url

Url Description

Misc

Team: Resource Discovery Unit / ROADS team

Figure 49: Selecting the new icon for a literature instance, brings up a form for the author to complete.


http://zippy.ecs.soton.ac.uk/ontoportal/discussion.cgi?seed=1&message_id=2 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
ontoportal discussions
Discussions > The Semantic Web > Literature > Languages for Dublin Core
back to discussion list
Post
Cancel
Done
Internet

Re: Dublin Core posted by Simon Kampa on Sun Mar 4 20:23:08 2001

> Could someone direct me to the dublin core working group homepage?

Sure, there is a link off my homepage :

www.ecs.soton.ac.uk/~srk98r

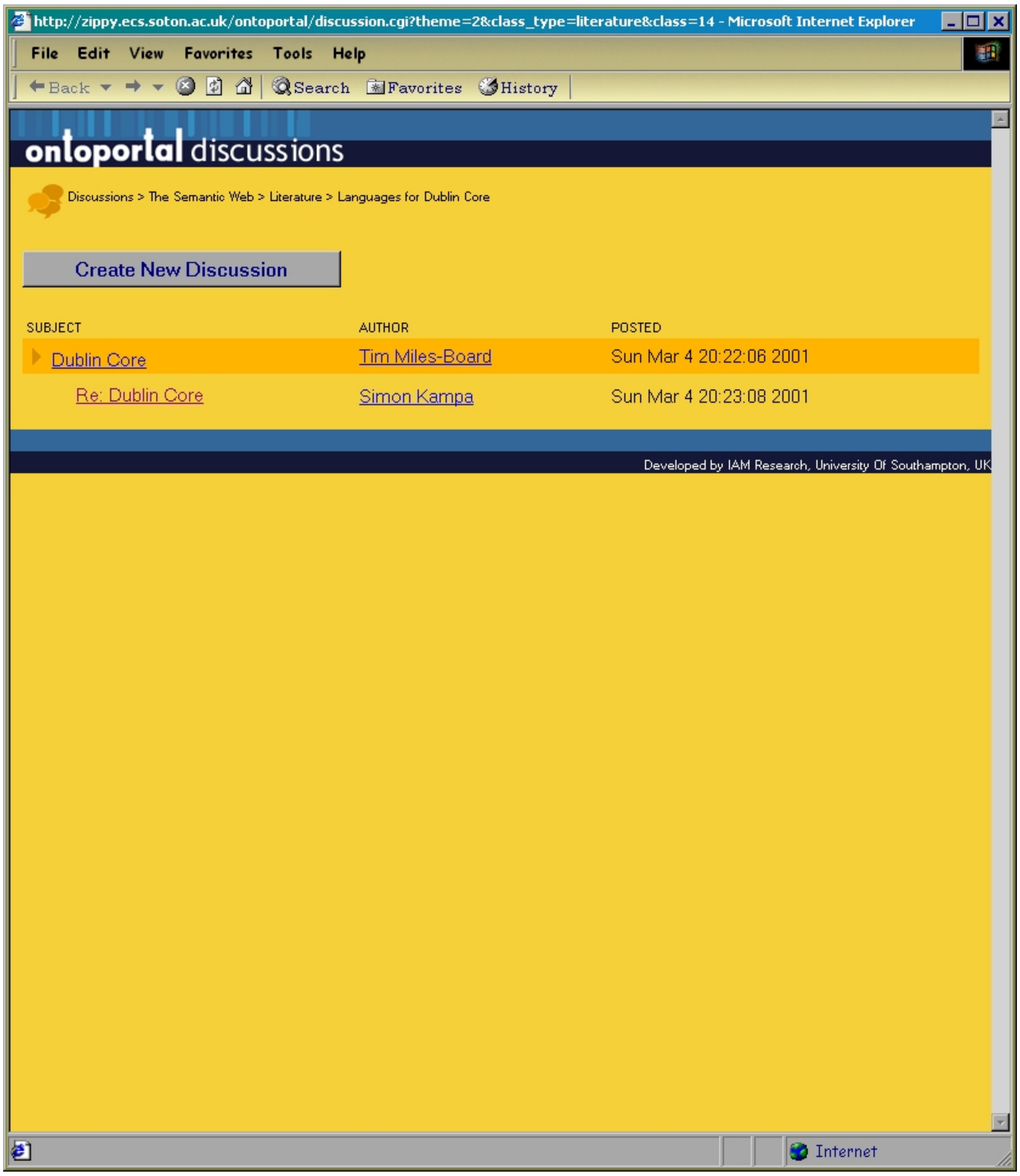
Reply

Subject	Re: Re: Dublin Core
Name	
Email	

Body

```
>Could someone direct me to the dublin
core working group homepage?
>
>Sure, there is a link off my homepage :
>www.ecs.soton.ac.uk/~srk98r
```

Figure 50: Discussion list.


http://zippy.ecs.soton.ac.uk/ontoportal/discussion.cgi?theme=2&class_type=literature&class=14 - Microsoft Internet Explorer

File Edit View Favorites Tools Help
Back Search Favorites History
ontoportal discussions

Discussions > The Semantic Web > Literature > Languages for Dublin Core

SUBJECT	AUTHOR	POSTED
Dublin Core	Tim Miles-Board	Sun Mar 4 20:22:06 2001
Re: Dublin Core	Simon Kampa	Sun Mar 4 20:23:08 2001

Developed by IAM Research, University Of Southampton, UK

Figure 51: Replying to a message.

C Resources

C.1 Contacts

Project Leader

Dr. Leslie Carr
IAM Research Group
Electronics and Computer Science
University of Southampton
Highfield
Southampton
SO17 1BJ
UK

Telephone: (023) 8059 4479
Fax: (023) 8059 2865
e-Mail: lac@ecs.soton.ac.uk

Project Investigator and Implementor

Timothy Miles-Board
IAM Research Group
Electronics and Computer Science
University of Southampton
Highfield, Southampton
SO17 1BJ
UK

Telephone: (023) 80594059
Fax: (023) 8059 2865
e-Mail: tmb99r@ecs.soton.ac.uk

Project Investigator and Implementor

Simon Kampa
IAM Research Group
Electronics and Computer Science
University of Southampton
Highfield, Southampton
SO17 1BJ
UK

Telephone: (023) 80594059
Fax: (023) 8059 2865
e-Mail: srk98r@ecs.soton.ac.uk

C.2 Links

OntoPortal.org

A web site detailing this work can be found at:
`http://www.ontoportal.org.uk`
A static snapshot of MetaPortal is available at this site. A dynamic version will also be added.

References

[1] Dublin core initiative. http://www.purl.org/dc/.

[2] Gnu wget. http://www.gnu.org/directory/wget.html.

[3] Microsoft xml parser sdk. http://msdn.microsoft.com/xml/default.asp.

[4] Mysql. http://www.mysql.com.

[5] Xalan c++. http://xml.apache.org/xalan-c/index.html.

[6] R V Benjamins, D Fensel, and A G Perez. Knowledge management through ontologies. *Proc. PAKM'98 Practical Aspects of Knowledge Management*, October 1988.

[7] L Carr, W Hall, S Bechhofer, and G. Goble. Conceptual linking: Ontology-based open hypermedia. *Proc. 10th International WWW Conference*, pages 334–342, May 2001.

[8] D Fensel, I Horrocks, F Van Harmleden, S Decker, M Erdmann, and M Klein. Oil in a nutshell. http://www.cs.vu.nl/ onto-know/oil/downl/oilnutshell.pdf, October 2000.

[9] D Fensel and M A Musen. The semantic web: A brain for humankind. *IEEE Intelligent Systems*, 16(2):24–25, March/April 2001.

[10] S Kampa, T Miles-Board, L Carr, and W Hall. Linking with meaning: Ontological hypertext for scholars. Technical Report ECSTR-IAM01-005, Electronics and Computer Science Department, University of Southampton, March 2001.

[11] A Latteier. The insider's guide to zope: An open source, object-based web application platform. *WebReview*, March 5 1999.

[12] S Luke, L Spector, and D Rager. Ontology-based knowledge discovery on the world wide web. *Proc. AAAI'96 Workshop on Internet-based Information Systems*, August 1996.

[13] S Pepper. Euler, topic maps, and revolution. *XML Europe*, April 1999.

[14] S Buckingham Shum, E Motta, and J Domingue. Representing scholarly claims in internet digital libraries: A knowledge modeling approach. *Proc. 3rd European Conference on Research and Advanced Technology for Digital Libraries*, September 1999.

[15] J Snyder and U Flemming. The object modeling language (oml) specification. http://seed.edrc.cmu.edu/ACL/ObjModelAPI/objapi.ps.

[16] S Staab, J Angele, S Decker, M Erdmann, A Hotho, A Maedche, H P Schnurr, R Studer, and Y Sure. Semantic community web portals. *Proc. 9th International WWW Conference*, pages 474–491, May 2000.

[17] W3C. Rdf (w3c recommendation). http://www.w3.org/RDF/, February 199.

[18] W3C. Xml 1.0 second edition (w3c recommendation). http://www.w3.org/TR/REC-xml, October 200.