University of Southampton

Faculty of Engineering, Science, and Mathematics
School of Electronics and Computer Science

Integrating institutional repositories into the Semantic Web

Harry Jon Mason

Thesis submitted for the degree of

Doctor of Philosophy

June 2008

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE, AND MATHEMATICS

SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

INTEGRATING INSTITUTIONAL REPOSITORIES INTO THE SEMANTIC WEB

by Harry Jon Mason

The Web has changed the face of scientific communication; and the Semantic Web promises new ways of adding value to research material by making it more accessible to automatic discovery, linking, and analysis. Institutional repositories contain a wealth of information which could benefit from the application of this technology.

In this thesis I describe the problems inherent in the informality of traditional repository metadata, and propose a data model based on the Semantic Web which will support more efficient use of this data, with the aim of streamlining scientific communication and promoting efficient use of institutional research output.

Contents

Declaration

I, Harry Jon Mason

declare that the thesis entitled

Integrating institutional repositories into the Semantic Web

and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- none of this work has been published before submission.

Signed:

Date:

Acknowledgements

Firstly I would like to thank Leslie Carr for his supervision, guidance, feedback, and moral support. I am also grateful to Hugh Glaser and Nick Gibbins for their constructive criticism, and Chris Gutteridge and Tim Brody for technical help.

Thanks to my family and friends for their encouragement, and Emma for both motivation and distraction.

This work was funded by an EPSRC grant.

1 Introduction

The starting point of this research is As We May Think[1]: the vision held by Vannevar Bush of the future of science. As the Director of the Office of Scientific Research and Development he coordinated scientific research in the United States. Writing in 1945 as the Second World War ended, he considers that the efforts of scientists should be a force for the benefit of society having been freed from their wartime obligations.

Bush observed the difficulties facing scientists relating to creating and referring to research material. He noted that it was likely that important contributions risked going unnoticed by those who could benefit, simply because they were unable to manage the large and increasing volume of published material even in highly specialized fields.

He proposed that contemporary and anticipated future technology could allow scientists to work more flexibly. His predictions include a desk sized complete library stored in microfilm, text and speech recognition apparatus, and calculating machines capable of higher mathematics such as calculus and logic. These are based on the idea that creative thought is concerned with identifying appropriate action, and it is therefore desirable to automate the uncreative details of performing the action. Bush also considered the problem of storage and retrieval: how a potentially vast repository of material might be indexed, arranged in a hierarchy, or queried. As predicted, the modern digital computer, database, and software tools provide many of these features and are invaluable for scientists today.

Perhaps the outstanding idea in Bush's paper is the Memex, which is a device for both creating and referring to resources in a microfilm library. It allows the user to browse through reference material using an index, add to it using a built in camera, and link resources by creating trails. Trails are an associative structure representing the branching chain of connections between different resources; they are intended to model the thought process which identified the connection and allow it to be retraced, modified, or shared with another person. Significantly, this aspect of the Memex aims to directly support the intellectual element of the user's work rather than simply automating routine tasks.

The associative trails Bush describes were the inspiration of an entire research field, of crafting expressive structures to represent and communicate linked information. Building on this idea, Ted Nelson coined the term "hypertext" and

further explored how links in documents can help capture the thought patterns of their author. The field of hypertext is the topic of the next chapter.

Although Bush's article is widely acknowledged as the inspiration of the field of hypertext, many similar ideas were described years earlier in the work of Paul Otlet, a pioneer of information science[2]. Otlet's work was largely lost as a consequence of the Second World War[3] and difficulties in funding. In 1934 in *Traité de Documentation* he hypothesised a mechanical information retrieval system, consisting of a universal repository of all knowledge, including text, images, sound, and video, remotely accessible and indexed in an all-encompassing classification scheme. This was the conclusion of forty years of research into bibliography and the organization of knowledge, combined with an active interest in cooperation between nations. His Mundaneum was intended to be an international library at the centre of a utopian world city, part of Belgium's bid to host the League of Nations; it was classified with index cards according to his extension of the Dewey Decimal System. The project was initially successful but was abandoned by the Belgian government and all but destroyed in the German invasion.

Like Bush, Otlet was concerned about the unmanageable proliferation of knowledge[4], and he sought technical, but also social methods of controlling this. A key difference between their points of view however is that Otlet considered knowledge as an objective ideal, separable from its medium of expression and stored in a single structured compendium. Rayward[2] compares this reductionist view to the work of Bush and later authors who instead placed the user at the centre of the system. Bush's description of the Memex emphasises the creative role of the scholar in managing their own information, making connections, and producing new ideas.

However foresighted his vision, Bush postulated only quantitative improvements in the technology of his time, and did not anticipate the possibilities now available with digital systems, computing, and the Internet. Though his microfilm and trails could be mechanically copied and exchanged in person, they could not be made immediately available to all other Memex users across the world. Digital storage means that perfect copies can be made and shared for negligible cost. The power of modern computers allows advanced graphics, multimedia, and interactivity, which can be the best ways of communicating ideas.

It is notable that Bush focused on the use of the Memex to organize the user's own thoughts, integrated with external sources of information, but made only passing reference to the dissemination of new material once created. The po-

tential for real time, effortless collaboration between scientists made possible by the Internet is beyond what he anticipated.

Though modern technology provides solutions to many of the problems Bush saw facing scientists, the area of academic communication still contains much scope for improvement. Communication among scientists is of course a key aspect of their work. Discoveries are often the result of direct collaboration, whether face to face or spread across the world. Scientists read and cite others' work to build upon, and publish or present their own work to influence other scientists and have an effect on the world. Therefore improvements in the tools and techniques of scientific communication, for example relating to authorship, publication, reading, or analysis, support the progress of science as a whole; and like the Memex would relate to the intellectual, creative aspects of science rather than the routine.

After the Second World War Bush hoped that scientists could devote themselves to improving people's lives. Supporting scholarly communication works to achieve this goal, and improved technology provides the mechanism.

1.1 Research contribution

The objective of this research is to identify technical means to promote scholarly communication on the Web. Work has centred on the data submitted into institutional repositories, and how the Semantic Web can improve access to that data. Though Bush focused specifically on scientific research, the benefits of improved communication are also not limited to science; this thesis is concerned with all scholarly discourse, and indeed communication in general.

The contributions made are:

- Evaluating institutional repositories in the context of the Semantic Web and identifying ways in which more useful data could be gathered.
- Proposing a new data model for institutional repositories which includes identifiers for the different entities which exist in the metadata.
- Developing and evaluating user interfaces which demonstrate the benefits of improved metadata.

1.2 Outline

Sections 2, 3, 4, and 5 provide background. Sections 6 (RAEPrints and WWW-Conf), 7, and 8 present original work.

Section 2 introduces hypertext and the World Wide Web as technologies for scientific communication and the expression of complex structure.

Section 3 introduces the next stage in the evolution of the Web: the Semantic Web. A set of protocols, formats, and standards, it is a way of making the meaning of Web data accessible to computer interpretation. This section describes the technology of the Semantic Web and several systems which are relevant to scientific communication and are based on this technology.

Section 4 describes institutional repositories and their place in scientific communication, including repository software, metadata, and interoperability standards.

Section 5 describes the issues in referring to resources using an identifier, or formal name. It discusses the characteristics of identifiers in the Semantic Web and how they relate to the data connected with a repository.

Section 6 analyzes three repositories, considering the quality of the data and how high quality metadata can add value to the collected data. It discusses how the data structure and user interface can contribute to collecting better data.

Section 7 describes two prototype tools which show a practical application for integrating Semantic Web identifiers with repository data. Analysis of these tools demonstrates how repositories could directly incorporate support for identifiers.

Section 8 presents a proposal for modifications to institutional repositories, which will improve the quality and utility of the data they collect.

Section 9 concludes the thesis and suggests possibilities for future research.

1.3 Glossary of terms

Anchor The point in a document at which a link starts or ends. In HTML, source anchors are clickable and usually highlighted and underlined.

Coreference When two different identifiers refer to the same entity.

Eprint An electronic document, often an academic publication. In this work, an eprint usually corresponds to a record stored in an institutional repository.

Formality In the context of this thesis, formal data has a defined format which can be interpreted automatically. Informal data might be understandable by a human reader, but cannot be reliably parsed automatically.

Generic link A link where the anchors are not explicit regions of a document but are automatically selected, for instance wherever a particular keyword occurs.

HTML Hypertext Markup Language. The standard document language used on the Web. HTML documents include structure, style, and hypertext features.

Hypermedia Hypertext extended to include multimedia elements.

Hypertext Text with a structure too complex to be represented on paper. Hypertext systems are generally composed of nodes with links between them. The Web is a hypertext system, though other examples have more powerful features.

Identifier A formal name assigned to an entity. In the Semantic Web identifiers are URIs, which are an extension of Web addresses allowing them to refer to physical or abstract entities as well as digital data.

Link In a hypertext, an object connecting a source to a destination. Links can be followed manually by user action or automatically behind the scenes. More complex links may contain multiple source or destination anchors.

Linkbase A database of links, which can be queried to obtain links to be applied to a document.

Metadata Data about data. For example, the metadata of a file includes its name, creator, and modification date. Metadata in the Semantic Web can facilitate automatic processing of the data by describing its meaning.

OAI-PMH The Open Archives Initiative Protocol for Metadata Harvesting. A protocol supported by repositories for efficiently exchanging interoperable metadata.

Ontology A formal specification of a domain of knowledge. An ontology defines the concepts in the domain and their relationships.

Open Access A movement in academic publishing, based on the idea that publications should be accessible to all potential readers without charge or restriction. Open Access is dependent on the ubiquity of the Web and the development of institutional repositories.

Open hypermedia A hypermedia architecture in which the hypermedia services are decoupled from the document, allowing a range of clients to use the services. This means links are standalone entities which are stored and accessible separately from the documents they refer to.

RDF Resource Description Framework. A standard language for encoding data and metadata in the Semantic Web, structured as a graph of triples.

Repository A digital collection of scholarly materials, combined with a commitment to preservation and access. In the context of this work, repositories are generally provided by an institution to collect and disseminate its research output. A repository is based around a piece of software, such as EPrints, but also consists of the policy and administrative frameworks surrounding it.

Resource In the context of the Semantic Web, a resource is an entity which can be assigned a URI, and thus be identified and have properties. Literals, such as numbers and strings, can be the value of a property but cannot have their own properties, and are therefore not resources.

Semantic Web An evolution of the Web where data is made accessible to automatic interpretation, through the use of self-describing formal data and metadata.

Transclusion In hypermedia, including part of a document within another by reference, without copying. In the Xanadu system transclusion was intended to support tracing quoted text back to its source, providing attribution and copyright control.

Triple In RDF, the basic data structure. A triple consists of a subject, predicate, and object; the subject and predicate are URIs, and the object may be a URI or literal value. Multiple triples form a directed graph with typed links.

Triplestore A database for storing and retrieving triples. Triplestores generally provide low level graph manipulation and query facilities, and focus on efficiency as data expressed as triples can grow very large.

Typed link A hypertext link with an associated meaning, allowing different types of links to be interpreted in different ways.

URI Uniform Resource Identifier. A URI has the same syntax as a URL, but is not necessarily resolvable to the resource it identifies.

URL Uniform Resource Locator, the addressing scheme of the Web. Defines the protocol, server and port, and path to a piece of data.

XML Extensible Markup Language. A standard metalanguage for encoding structured data on the Web, XML can be used to construct arbitrary markup languages which can be read by a common parser, or combined within the same document through namespaces. HTML can be written in an XML compatible style.

2 Hypertext

This section introduces the concept of hypertext as a mechanism for organising documents and information. It provides background information on early hypertext systems and their significant features, and discusses the structures used in hypertext with the insights they give to the general problem of managing structured data.

This section also introduces the World Wide Web, the most successful global hypertext system, which is a key element of the later parts of this research.

Hypertext is important to my research as it is the basis of the Web, now an essential feature of scholarly communication; but also because of its influence on the Semantic Web. The principles of managing structured information in hypertext relate strongly to the Semantic Web, and the issues of naming and referring to resources are closely related between the fields.

2.1 What is hypertext?

The term "hypertext" was coined by Ted Nelson as part of his design of a non-sequential writing system[5]. Like the structures of Bush's Memex, the concept was based on an idea about the nature of thought processes: observing his own thought patterns, he noted that the ideas which make up a written document are formed independently of the linear structure of the resulting text. The text takes shape iteratively, by writing, revising, and combining fragments as the writer thinks about the overall shape of the document and how to best express their ideas. This means the eventual structure of a document may not be evident at the outset.

Nelson's PRIDE system was designed to provide an authoring environment which intrinsically supported this process. It handled fragments of information separately, then combined them to form a document. The system was to support the ongoing development of the work by keeping track of versions and managing the overall structural elements, chosen by the author, for example an outline, index, and references. The flexibility in managing these elements is a characteristic of his novel cross-linked list based data structure, "zippered lists". The author could for instance refer to the index while writing to assist in collecting their thoughts, or maintain alternatively arranged drafts of a section. As well as managing writing, Nelson imagined the same system storing reference mate-

rial, genealogical data, or even program code—anything which is structured or interconnected.

This system gave rise to the idea of a hypertext, defined by Nelson[5] as a piece of text with a structure too complex to be conveniently represented on paper. As the text is not created in a linear fashion, there is no reason why it must make up a linear document constrained to mimic a printed page. Nelson argued that hypertexts could be particularly suitable for education, by using an adaptive structure allowing each reader to follow a different path according to their choices and needs. Hypermedia need not simply be text; he also imagined pictures and films with complex underlying structure.

Nelson's most well known hypermedia system, Xanadu[6], suffered technical and financial setbacks and did not become the universal system he had hoped. However, his ideas about structured data systems have been a strong influence on the field.

He is a vocal critic of the World Wide Web[7] and argues[8] that the Web is inadequate as a hypertext system for serious scholarship and discourse. In its basic form the Web lacks several features of Nelson's ideas—robust links external to the text, versioning, multidirectional links, and transclusion: a quoting mechanism which is particularly significant to the topic of scholarly discourse.

Transclusion is

> a system of visible, principled re-use, showing the origins and context of quotations, excerpts and anthologized materials, and content transiting between versions.[8]

In a hypertext which supported transclusion, the quote given here would not be copied, but included from its source by reference. A reader wishing to obtain the context of the quote could follow it back to the source. The advantage of this in a scholarly context is clear, as it facilitates review and analysis; it also permits parallel visualization of related documents. It is also the basis of Nelson's proposed system of copyright protection and micropayment, which would allow the originating server to control delivery of the quoted content.

2.2 The importance of hypertext

Though Nelson conceived of hypertext primarily as a way of writing literature, the same principles can be applied for the more general purpose of organising

information. The field is closely integrated with developments in the Semantic Web, often sharing common structure and ideas because of the similarity between linked documents and linked data. Because of this, issues in managing hypertext information may be just as relevant to managing general data—the Memex, which shares the information management goal of an eprint repository, is a hypermedia system based on observations of data management in business.

At its heart, hypertext deals with reference—the connections between related things, and how to model and make use of those connections. This is also the central concept of the Web and the Semantic Web, and as Bush argues, human thought and scientific enquiry in general[1]. The insights gained from studying hypertext are therefore relevant in the remaining sections of this work.

2.3 Early hypertext

While Bush famously described a visionary hypermedia system, and Nelson defined the term, the first digital implementation was the NLS (On-Line System)[9], designed by Douglas Engelbart. This system shared the goal of assisting scientists' communication with awareness of the patterns of human thought processes, and was distributed over the early ARPANET. After this there were many notable hypertext systems, demonstrating variations on the concept in different usage scenarios.

Many early hypertext systems used the *frame* as the basic indivisible unit of information. A frame is a self contained fragment of information, designed to be displayed completely on the screen, with links made between one frame and another. KMS (Knowledge Management System)[10] was an early example, which featured two types of hyperstructure: a hierarchical arrangement of frames in a tree, and associative links between arbitrary frames. HyperCard[11], popular due to its distribution with Apple computers, organized information as a stack of cards; although not inherently a hypertext system, its scripting functionality could be used by developers to produce hyperstructure.

The constraints imposed by these designs contrast with the fluidity of structure imagined by Nelson. Later systems removed some of the boundaries and began to more closely resemble his hypothetical system. It is however notable that none of the systems discussed here provide all of the features he described.

Notecards[12] used the structure of frames and links to provide a visual overview of the hypertext. It also featured *typed links*, where links had different interpre-

tations and were displayed with different patterns in the visualization. Typed links were later a key part of Textnet[13], where a taxonomy of types related to scientific discourse are used. The concept of typing is significant in the Semantic Web, discussed later.

A notable usability improvement was pioneered in HyperTIES[14], a system designed to represent linked encyclopedic content which was similar in features to the Web. In this system links were first represented as highlighted *anchors* within the flow of text itself rather than separate icons, menus, commands, or codes.

The Sun Link Service[15] and later Intermedia[16] stored links separately from the information they relate to. This concept is the foundation of *open* hypermedia, where interoperability between systems is key. Earlier examples were self-contained applications; these systems provide hypermedia services to multiple applications and allow links to bridge the gap between them. Links were stored in *linkbases* and could be managed independently of documents, for instance by providing appropriate links for different readers. In Microcosm[17], declarative *generic links* could automatically be applied to relevant documents, linking from wherever a particular phrase is encountered in a compatible application.

A further move towards interoperability, HyTime[18] was a standard for hypermedia constructs which could be applied to any SGML based markup, providing compatibility between the hypermedia aspects of different systems. Features included temporal and structural anchors, where a link could refer to part of an audio or video sequence or a specific element in the SGML structure of a document.

As hypertext grew as a field of research there were also theoretical models developed which aim to abstract all the features of hypermedia systems. The Dexter Model[19] was intended to be a base for comparing different systems and developing interoperability standards. It separated the structure, storage, presentation, and interaction layers to provide terminology for future research. Thompson[20] had a similar aim, presenting an idealized architecture using which hypermedia systems could be developed, analyzed, or standardized further. However his modular architecture aimed to encourage independent progress within each module, for instance versioning, persistence, or linking, to build consensus within the research field and avoid wasted effort. Open Hypermedia Protocol[21] provided a standard interface to decouple link services and applications. Each client could then support a single protocol with a shim connecting it to any open hypermedia service, avoiding compatibility problems with propri-

etary services. The Fundamental Open Hypermedia Model[22] generalizes this abstraction across three related domains: navigational, spatial, and taxonomic hypermedia, modelling links as generic objects with directionality, context, and behaviour.

There is a trend towards interoperability through the history of hypertext, mirroring the trend of software in general. The earliest systems were standalone, closed applications, meaning that links could only be made between objects contained within the application. This is the simplest way of storing a link as it can be modelled as a pointer to a database record. If any data is changed the link can be updated automatically. However, it restricts the use of the hypertext to a limited environment. External links, to other servers on a network, or between different applications allow the hypertext to achieve a global scale. The problem with connecting to external systems is that care must be taken to ensure reliability and robustness against changes in remote data. Without guaranteed data integrity, linking to remote resources must depend on naming or describing the resource. The issues raised by naming resources will be discussed later.

The most popular hypertext system, the World Wide Web, arguably owes its success to its pragmatic acceptance that reliability cannot be guaranteed[23].

2.4 The World Wide Web

The Web's origins lie in scientific research and collaboration. Specifically it was designed to manage and share the large volume of information produced by the thousands of scientists at the CERN particle physics laboratory. Tim Berners-Lee observed[24] that newly arriving scientists would take time to find out about the relevant facilities, resources, and people in their field of work causing an initial period of low productivity. At the same time, departing scientists would take away with them their knowledge and experience of the CERN environment. Continual turnover of scientists therefore caused a gradual loss of information; for example, that an experiment about to be run had already been done and where the results could be found.

To combat this loss, Tim Berners-Lee proposed "Mesh", a linked information system based on ideas from his earlier Enquire system[24] (which was similar to HyperCard, but multiuser). It could support a wide variety of information, including some automatically obtained from existing data sources and linked in, represented as nodes containing "hot spots" linking to other nodes. CERN's diverse, distributed environment gave rise to its distributed client-server archi-

tecture and support for clients on heterogeneous computer systems. Interoperability was achieved by the use of standard protocols, markup, and naming: HTTP, HTML, and the URL syntax. The first clients were a graphical browser and editor for NeXT systems, and a simple portable text browser.

Though it was initially intended for internal use, the system was soon named the World Wide Web[25] and released freely to the world at large. Its strengths as an information system for CERN proved equally valuable on a global scale. As it was distributed and decentralized new sites could immediately begin to use it; this was aided by the simplicity of its protocol, which encouraged the development of graphical and text browsers for a variety of platforms. Accessibility was assured as no assumptions were made about the types of documents which could exist, and documents could be written easily by non-technical users or even built in a WYSIWYG editor. Though Web browsers could use other protocols (such as Gopher) which could be expressed in the URL syntax, it was also possible to translate legacy data to the Web dynamically.

The World Wide Web is of course a valuable tool for scientists in its present form. It is notable however that as the Web's rapid adoption worldwide was partly due to its simplicity[23], the same simplicity means that many key features of Bush and Nelson's designs are missing. In fact the Web has a relatively low level of functionality compared to both earlier and later hypertext systems[26, 27]:

- Links are one way, and are not robust. The target of a link may be removed or modified, making the link useless or misleading. Link markup must be embedded in the source document, and to link to part of a document also requires an anchor at the destination, making it dependent on maintenance by the document authors.

- It lacks expressive hypermedia data structures which could assist authors and readers: transclusion, linkbases, generic links, or annotations.

- In hypermedia systems there is often the possibility of being disorientated. Hyper-G[28] and Intermedia, for example, maintain an overview of the location in the system; but without this global organization the user can become "lost in hyperspace". History and bookmark features in browsers assist to some extent, however, and Bernstein argues[29] that the problem is not significant in a well designed text.

- Information discovery is not part of the Web's architecture, but is dependent on search engines, which work by harvesting data from all indexed

pages. The protocol does not provide any standardised way of searching, or identifying what data is available from a server other than traversing links.

- There is no explicit semantic information on nodes or links, which could be used to inform users about their relevance.

Since the Web's creation it has of course evolved dramatically, both in terms of technical improvements and the scope of its content. The protocol supports metadata transmission, allowing any file type and improving scalability by supporting caching. HTML is dramatically more flexible, supporting advanced styling, structure, and active scripting. Browsers are ubiquitous and user friendly, encouraging personal and commercial as well as academic use. CGI and server side programming languages provide dynamic content. These developments mean that the browser has become the universal front end for other applications. The extensibility of the protocols, combined with client scripting and intelligent processing server side, also make the Web a natural base on which to build more capable hypermedia systems:

- Hyper-G[28, 30] provides a richer data model than the Web with an overall organized structure and open hypermedia features. While interoperating with the Web and other contemporary information systems such as Gopher and WAIS, it requires a custom proprietary client to support its full feature set.

- The Distributed Link Service[31] provided open hypermedia capabilities to the Web. Link services, implemented as standard Web server CGI scripts, provide links when requested by a client side browser integrated tool. Anchor points in documents to be augmented with links can be specified exactly or as keyword matched generic links.

- COHSE[32] extended the generic link functionality of the DLS to produce a conceptually driven hypertext. Documents annotated with metadata about their meaning could be targeted more accurately with appropriate generic links.

- Devise Hypermedia[33] adds complex open links and collaborative editing to the Web, using embedded scripts inside standard Web browsers.

- XLink[34] is a language for open hypermedia links intended as a new Web standard for the next generation of browsers. It is built on a layered set

of standards for naming and structuring documents to maximize interoperability. As an XML application, other languages can make use of XLink to provide hypertext features; SVG[35] is an example. This is equivalent to the service HyTime provided for SGML.

- Wikis, such as the WikiWikiWeb[36] and Wikipedia[37], are Web-based hypertext systems with collaborative editing features, including versioning and edit conflict resolution.

- The Semantic Web is a linked, distributed data graph built on the Web. It is capable of providing the necessary support for advanced hypermedia features such as typed links, links inferred from a document's meaning, logical reasoning over hyperstructure, and nodes and relationships with complex structure. The topic is discussed in full in the next chapter.

2.5 Specifying links

Though the Web is a recent development relative to the history of hypertext, as the ubiquitous hypertext system it is a useful starting point when comparing hypertext systems. It is familiar and well understood, has flexible naming and retrieval features, and uses a simple markup language (HTML) but has enough expressivity to be a base for more complex structures.

Clickable links in HTML are the kind of hypertext link familiar to most users, but they are only one example of how to represent complex structure in a hypertext. HTML includes other features which embody hyperstructure[38], and other hypermedia systems have a wide range of different link types.

The HTML `<a>` (anchor) tag creates a simple one way link between an area of the source document and a point in the same or another document. The source of the link is specified by enclosing the desired text inside the tag, and the target is specified by its URL, optionally including a fragment to link to a specific point in the document. Therefore links can only be added to a document by its author, as they are embedded in the markup, and can only point to a position within a document if an appropriate target anchor exists. The URL naming syntax allows links to other formats or using other protocols.

The second kind of link does not involve an anchor, but describes relationships between whole documents. Unlike `<a>`, the `<link>` tag has explicit semantics: though commonly used to relate a document to an external stylesheet, it can also provide navigation within a tree structure, or refer to alternative expressions of

the same content. In the modern Web the navigation features are supported by some browsers but rarely used, though increasingly an alternative version in RSS format is provided for aggregation. The link provided can even define the semantics of the inverse relationship, from target to source. These capabilities could express hyperstructures like Nelson's multidimensional lists, though no definitive interpretation is specified.

The other kind of link available is automatically followed by Web browsers, as it represents the inclusion of another resource into the document. Various tags are used to describe this: the generic `<object>` tag from HTML 4, and the older `<img>` for images, `<iframe>` for HTML, and `<embed>` for multimedia. At first glance these links merely support the inclusion of different media into the document, and do not provide any structure more complex than paper. However, the link target is a URL rather than data bundled with the parent resource, thus supporting inclusion of data stored on a different server or sharing of common information between documents.

Even though it is designed purely as a markup language, HTML's different link types, which provide association, hierarchy, and composition, demonstrate the synergy between hypertext structures and general data structures.

2.6 Extending the concept of the link

With HTML as a base, considering how links and anchors are handled in other systems will lead into issues affecting linked data management in general. These issues will arise again in the Semantic Web.

2.6.1 Displaying links

Following an HTML link by clicking on it is a definite action directly associated with a region of the document. Other types of link might use other ways of presenting a link to the user. The earliest Web browsers, like the Enquire system which was its precursor, displayed the links in a list at the bottom of the document, and the document-wide navigational links provided in HTML can still be rendered this way. Xanadu's links were designed to be displayed as lines connecting two documents displayed side by side. Here the link is followed implicitly, and all available links are active at once. Links in an adaptive hypermedia system might be intended to compose fragments of text into a single

document: the user might not see the links at all, but they would shape the structure of the document behind the scenes.

How connections are displayed to the user is also an issue in the linked data world of the Semantic Web. There are systems which browse pure RDF data by directly following links on the graph, such as Berners-Lee's RDF Tabulator[39]; faceted browsers, which allow exploring via an object's metadata, such as Longwell[40] and mSpace[41, 42]; domain specific tools to present certain types effectively, such as the SIMILE timeline component[40]; and metalanguages to describe how to display other data, such as Fresnel[43]. This subject will be revisited later in the context of displaying repository data.

2.6.2 Open hypermedia

One direction to extend the concept of the link is by removing it from the document, storing it in a linkbase as a first class object of equal importance to the document text. Systems of this type are known as open hypermedia systems, and are in some ways more complex but benefit from increased flexibility and expressive power.

The key benefit of open hypermedia is that the links relating to a document can be maintained separately from the document itself, or even by a third party. Links can therefore be organized independently from documents, for example by grouping them according to topic, providing an institution-specific set of links to internal resources, or providing targeted links for different groups of readers. The added complexity to support this includes identifying the appropriate set of links, which might involve searching a number of linkbases based on the document's metadata; finding appropriate anchor points for the links, in documents which may change independently; and applying them to the document in an appropriate style to provide clear navigation.

The ability to make statements about third party resources is central to the Semantic Web, as it encourages the reuse of data in new contexts. Therefore the structures and protocols used in open hypermedia will see comparable roles in handling linked data. The problem of managing the provenance of third party data, how to obtain and certify trustworthy information is also especially pertinent.

The related topic of external anchors—link endpoints—is discussed below.

2.6.3 Complex links

As a first class object with its own attributes, a link is free to have a complex structure. Another way of extending the link is therefore to remove the constraint on the number of endpoints. If two alternative targets for a link are equally valid, such as alternative versions of the same document, this could be expressed as two links from the same point to different documents or as a single link to two places. Similarly, a single link could have two source anchors pointing to the same document, or in fact any number of sources and targets.

Complex links potentially provide benefits in maintainability, interaction, and analysis. A change made in a multi-anchor link affects all of the documents pointed to by the link; if it were expressed as many simple links, a change would have to be duplicated in each link making maintenance more complex and error prone. A simple example is providing a link from all pages about a specific topic to a summary of that topic hosted on another site; if the summary page moves, the address only needs to be changed in one place.

Following or interpreting such a link is not as simple as redirecting the viewer to the target document, but complex links can also lead to richer possibilities in interaction. The user could just be presented with a list of alternatives, or one might be chosen depending on preferences, context, and the semantics of the document or link. For example, two alternatives might have a different style for different age groups, with metadata identifying the target group and a browser preference specifying the reader's age. By displaying this as a single link rather than two, it is clear to the reader that the two targets are alternatives and they are unlikely to need to read both documents. This could be extended to build an adaptive hypertext document, where links define the structure and are followed automatically to compose a customised document.

Complex links are intrinsic to linked data structures, because a property linking two objects essentially forms a link between them. In the case of a relationship common to many objects, the process of analysing or displaying this connection is a similar process to handling a complex hypertext link. Conversely, the existence of a hypertext link between two entities implies a connection between what those entities represent, and a multiway link implies this for all its members. Hypertext systems are potentially rich resources for data mining, and complex links therefore enhance this further. Even if the meaning of the connection is not made explicit, analysing the link may allow conclusions to be drawn about the relationship between the documents. With complex links it may be possible

to draw stronger conclusions about these relationships if the link implies the same kind of connection between all its members.

2.6.4 Semantic links

All hyperstructure has intrinsic meaning. A link exists for a reason, and it is followed on the premise that appropriate data can be found at the other end. In a hypertext document, the text around or inside the anchor might provide information which helps the user decide whether to follow it:

> The World Wide Web is a Hypertext system invented by Tim Berners-Lee at CERN.

A hypertext data system might have links displayed in groups of related resources:

- Back
- Forward
- Index

Browse by

- Year
- Author
- Subject

However, these clues only have meaning to an intelligent human reader.

Document-wide links in HTML however, specified by the `<link>` element, can contain an attribute encoding the meaning in a machine readable way. Several standard values exist, such as[38]:

Alternate A substitute version for a different media type, or a translation.

Stylesheet A reference to an external style sheet to be applied automatically to the current document.

Next/Prev Adjacent documents in an ordered series.

Bookmark A link to a labelled entry point in a long document.

Compared to an anchor element, which only specifies the target, these semantic links can be handled in the most appropriate way for the context in which they are found. For example, navigation provided by regular links is suitable for a human with a standard Web browser, but the meaning inherent in document scoped links could build a table of contents for a printed copy, organize browser bookmarks, or provide efficient shortcuts on a handheld device.

Notecards[12], for instance, used the knowledge of link types to provide a useful graphical overview for the user, with different line styles for each type of link. Here the type information is used to help the user maintain their orientation in the hypertext. Other systems which use data types are MacWeb[44] (to allow authors to construct adaptive hypertexts which adjust to different users) and Textnet[13] (which used link typing to assist scientists in expressing their arguments).

If a hypertext system can interpret the meaning of data, it can therefore enhance its value. This is a general principle which will be revisited, particularly as a key aspect of the Semantic Web, which deals with a linked database rather than linked documents.

2.7 Anchors

Links are related to another fundamental structure: anchors. While a link defines the connection between entities, an anchor defines those entities. Depending on the hypertext system, the boundary between links and anchors may be more or less clear. In HTML, source and target anchors are created by markup in each document, but in more advanced hypermedia systems anchors can be defined in complex ways which are potentially more expressive.

The simplest anchor possible is the document or node. In systems with this model, nodes have no internal structure but are simply a fragment of text (or other media). Links connect whole nodes together, and might be displayed in a list below the content of the node. Therefore the anchor at each end of the link need only be a pointer (in local-only closed systems) or node address. KMS[10] and Gopher menus[45] demonstrate this anchor type. Xanadu[6] can be considered a system of this type, though nodes might be composited at the user interface level to give the appearance of internal structure.

If nodes have structure or markup, anchors can be specified more precisely. HTML clickable links are unidirectional, therefore the anchors at either end use

different markup. The source is specified as a range of characters or elements within a document. The target is often a whole document but may also be a range, but practically this specifies only a point part way through a document. Both source and target require markup embedded in the document, therefore the valid points for link targets must be included by the author, though linking to the whole document is always possible. Other systems with similar anchors include HyperCard[11], Texinfo[46], and wikis[36]. HyTime[18] specifies "clinks", which are embedded in the document like HTML, as well as external links.

As with links, the concept of an anchor can be extended to gain a greater expressive power.

2.7.1 Open anchors

Just as with links, anchors could be specified externally from the document. It is impossible for a URL to link to any arbitrary point in an HTML document, unless the document contains an anchor embedded by its author. Also, browsers interpret the target anchor by scrolling to the anchor point, which effectively makes the target a point rather than a region; the only way to clearly refer to only part of another document is to quote it, without server or client code to work around the problem in a non-standard way.

To effectively allow open hypermedia links it must also be possible to specify anchors externally; ideally, by describing meaningfully the area of the document which is relevant, an external anchor can remain accurate when the document is modified. Microcosm[17] and the Distributed Link Service[31] use a combined character offset and keyword search. HyTime[18] links describe a region of an SGML document by reference to its structure, which is precise while also permitting minor changes to the document; the emerging W3C standards XPath[47] and XPointer[48] apply this to the modern Web and XML documents.

The disadvantages of externally stored links are that changes to the documents may make the link invalid or misleading; documents and links must both be updated if the document is revised; and extra complexity is required to render a complete document, including locating and incorporating appropriate linkbases. The benefits are that as objects with their own identity, links can be produced managed separately and can have their own metadata.

2.7.2 Generic anchors

An HTML link connects two precisely specified points specified uniquely by URL. Generic anchors are described by their characteristics, for example by keyword, thus creating a link from each instance of that keyword. They can be applied to an HTML document by client-side scripting or browser integration, server preprocessing, or by a proxy server.

Generic links are another way of facilitating third party links which are more resistant to change and require less maintenance than using specified anchor points. Minor changes in the structure or text of a document do not affect the link's validity provided the keyword is still present. Links can even be added to a wide range of unrelated documents without precise knowledge of the document's content; this allows a wide variety of generic links to be in effect at one time while still only displaying links which are appropriate to the current content, for instance a dictionary or glossary which automatically defined uncommon words.

It may in some cases be difficult to create appropriate generic links. Obviously if a keyword has more than one meaning simply matching that word would sometimes be incorrect. The desired link may not be a particular word or phrase, but could be described by a wide variety of word combinations or common words. They also share the same disadvantage as explicit external links, of discovering and applying an appropriate linkbase, and are only useful in situations where matching a keyword will produce the desired type of links, such as glossaries or technical documentation.

Various techniques can be applied to make generic links less indiscriminate. Users could be allowed to limit the available links by dividing them into small linkbases with narrow scope. Microcosm allows document local generic links, where the document is specified explicitly but the anchor by keyword[17]. COHSE uses annotations describing document semantics to refine the keyword matching process, selecting links from an appropriate contextual linkbase[32]. Open Journal used a simple keyword database, but avoided overwhelming users by maintaining central editorial control; limiting the keywords to the ones which provided the most valuable links within the domain[49]. In each of these examples the accuracy is improved by more accurately describing the intended anchor points.

2.7.3 Abstract anchors and identifiers

Generic links as described above point to regions of a document which are described, rather than specified explicitly. The idea of an anchor being a description of a resource can be taken further, to make links between abstract digital or even physical end points.

A citation in an academic work is a link. Though before digital publishing it would have existed only on paper, it creates a unique connection between two resources with exactly the same meaning as a digital link would. The source anchor is embedded text—a number, or name and date, but the target is a good example of how objects can be referred to by their attributes rather than a unique name. The components of a journal citation, for example, allow the reader to find a physical copy of the target in a printed copy, or to search the journal online by title, or to find an alternative copy self-archived in an institutional repository by searching for the authors' names.

In pervasive computing, physical objects can be considered anchors in a hyperstructure which is navigated by the user's movement in the space[50]. The Mack Room and Ambient Wood are examples of pervasive computing projects which involve hypertext and a close relation between digital and physical resources. In the Mack Room this is a gallery of exhibits with associated digital metadata[51]; in Ambient Wood, an area of woodland providing virtual measurements and interaction[52]. The underlying hypertext systems in both projects used the transition between sensor boundaries to trigger navigation in the hypertext. The sensors can therefore be considered physical anchors, with a corresponding digital representation in the model and a mapping between them.

For closed special purpose systems such as these the mapping may be simple and defined by application code. However, to extend a similar concept to a Web scale system with distributed control and authority over the data would require consideration of how the physical and digital components of the anchors are identified. The problem of how to model links gives rise to a more fundamental question: how can a physical object, or even an abstract concept, be given a digital identity? This question is central to the Semantic Web.

The problem of assigning identifiers to people demonstrates the problem, and is considered in detail later. Other examples are geographical regions, which would require a boundary to be described; or periods of time, perhaps to refer to historical facts, where the exact bounds of time may be unknown.

A more abstract example is an institution: a pervasive hypermedia system could conceivably use "the University of Southampton" as the start point of a link, to be displayed when physically present on the campus, or when communicating with a staff member, or reading a news article or publication about the university. In the same way a reference could exist in the Semantic Web connecting the same abstract identifier to the area of land, the staff member, and the article.

To integrate a hypermedia system into the physical world requires devising identifiers for physical objects and making links between them. The Semantic Web also uses such identifiers to describe objects with metadata. This is an area of close convergence between the two fields. In Semantic Web systems, the presence of a link between resources need not imply that it could be "followed", the meaning of which may be unclear from a hypertext perspective, but that a relationship between them exists and can be analyzed.

2.8 Summary

Hypertext is a mechanism for managing structured information. This could take the form of a document: a piece of literature, but with a complex non-linear structure; or multiple interconnected documents, like an encyclopedia; or a linked data bank, to record and browse structured data.

The success of hypertext in the form of the World Wide Web demonstrates the value of a linked, browseable, global information system. However, many valuable concepts from hypertext research are missing from the Web, notably open hypermedia, explicit semantics of data, and overall organization.

More recent hypertext systems have shown a trend towards openness and therefore interoperability between different systems and across network boundaries. This gives rise to the issue of identifying related resources through naming: in the World Wide Web the URI is a general purpose solution.

Understanding hypertext is valuable when considering the Semantic Web, the topic of the next section. The Semantic Web removes the document and text elements, leaving only a linked data structure; but as demonstrated above this has many properties in common with a hypertext. Semantic Web technology can be applied to add value to data on the Web by providing better organisation, improved browsing and searching capabilities, and the ability to automatically manipulate data.

3 Semantic Web

Bush argued that science is a creative process, and that the laborious calculation and processing required could be performed automatically by machine[1]. He first described automation of the simplest routine operations, but predicted that improved technology would allow mechanization of higher level processes, such as symbolic manipulation and logic.

These low level operations in the context of the Web are the obtaining and rendering of Web pages to the reader. The computer has an 'understanding' (metaphorically) of the syntactic aspect of the data: the structure and formatting of pages. The semantic aspect is accessible only to a human reader, and represents the intellectual content of the page: the knowledge that this Web page is someone's personal home page, their name is in the page title, the links point to their publications, and so on. Similarly a computer can find all pages containing a particular set of words, and use heuristics to judge which pages are most likely to be relevant, but a human must examine each to find the answer to their question.

The Semantic Web is a technology which provides a standard framework for the association of meaning with data. This facilitates scientific communication by allowing more advanced processing to take place to assist the reader at a higher level, and by making it easier for different tools and datasets to work together.

This section will describe the background of the Semantic Web, the technology and standards which have developed as a result, examples of systems which build on it, and the implications for scientific communication.

The Semantic Web is not a single application or system, but an abstract idea about how to make the best possible use of material on the Web: formally defined metadata to support automatic processing. From this idea a set of interconnecting standards have developed for encoding, structuring, manipulating, and distributing this data. Its core technology is the RDF graph specification syntax, which is intended to form a worldwide linked data system alongside and integrated with the Web's existing linked document system.

3.1 Why a Semantic Web?

From its earliest incarnation the Web was intended to be processed by automatic systems as well as human readers. Tim Berners-Lee at wrote about his first proposal at CERN:

> An intriguing possibility, given a large hypertext database with typed links, is that it allows some degree of automatic analysis. It is possible to search, for example, for anomalies such as undocumented software or divisions which contain no people[...]
>
> It is also possible to look at the topology of an organisation or a project, and draw conclusions about how it should be managed, and how it could evolve[...]
>
> Perhaps a linked information system will allow us to see the real structure of the organisation in which we work.[24]

Citation linking is another example which demonstrates the potential of machine interpretation. Citations essentially link publications into a hypertext which might be analyzed for authoritative references and communities of practice: taking the implicit knowledge from the academic community of what is significant, and making it explicit and easily communicated to new members of the community. Brody et al.[53] use this technique to evaluate the impact of research. PageRank, the algorithm used by Google to select better quality search results, is another example of automatic analysis of a hypertext; it works by analyzing links between documents, judging documents linked from many important places as the most important[54].

Citation linking contains good examples of the difficulties in automatic processing of Web data. If the text of a publication is simply made available on a Web page, the citations would be in plain text and not linked. Links would have to be added, either manually or by processing the text and looking for strings which may be citations. The manual method may be time consuming, which may be insurmountable for a large corpus, while the automatic method is vulnerable to errors in parsing. Though standard citation formats exist, it is a challenge to accurately pick out citations, and an even greater one to break down the format and identify each component; this would need to be done before the link could point to the correct target. Even then, the Web address of the cited document needs to be resolved. In each stage of automatic processing there is possible ambiguity and therefore the chance of incorrect linking.

Despite the challenges it is possible to parse citations reasonably accurately, especially if the format is known (such as when it conforms to a publisher's house style), and so value added services can indeed be built. Simply linking a citation to its target, as done by the ACM Digital Library, supports the user while browsing. At the other end of the scale, given reliable citation information attempts can be made to analyse the citation graph, for instance to produce metrics for a publication's impact[53], or identify possibly significant publications in a field of research to aid new students or provide feedback for funders[55]. This example demonstrates that if the problem of extracting the meaning implicit in the text of a citation is solved, it becomes possible to make greater use of the document.

Parsing and extracting the relevant information from the hypertext of linked citations produces a data structure: a graph of linked nodes. However, the methods for extracting and analyzing citation data are specific to that task because the source data is designed for human interpretation. To combine it with another source would require a similar—but incompatible and equally error prone—information extraction process.

The Semantic Web is a mechanism by which these kinds of services could be provided more reliably for data on the Web. It is built on earlier efforts to annotate Web pages and documents with metadata, but extended to encompass any type of structured, linked data[56].

Over a decade after his original proposal, Berners-Lee advocated an evolution of the Web with the goal of supporting automatic tools, aiming to provide a better experience for Web users. His vision, called the Semantic Web[57], has software agents performing advanced tasks for users involving querying, analysis, and aggregation of data:

> At the doctor's office, Lucy instructed her Semantic Web agent through her handheld Web browser. The agent promptly retrieved information about Mom's *prescribed treatment* from the doctor's agent, looked up several lists of *providers*, and checked for the ones *in-plan* for Mom's insurance within a *20-mile radius* of her *home* and with a *rating* of *excellent* or *very good* on trusted rating services. It then began trying to find a match between available *appointment times* (supplied by the agents of individual providers through their Web sites) and Pete's and Lucy's busy schedules. (The emphasized keywords indicate terms whose semantics, or meaning, were defined for the agent through the Semantic Web.)

This is made possible not with advanced artificial intelligence interpreting the meaning of text, but with formally specified explicit metadata augmenting the existing human readable content, allowing the tools to reason and make logical connections. Like the early Web it focuses on standardizing data exchange formats and protocols, and is an addition to existing Web resources designed to be incrementally upgraded as the technology develops.

For the Web as a whole the vision of the Semantic Web promises to provide a better user experience, by making the right information easier to find, supporting automated tasks, and simplifying the sharing and reuse of data. These benefits are also specifically applicable to the task of finding scholarly material for researchers. Because the value of research can be measured by its impact, both on the academic community and the world, it is an area where encouraging use and distribution of information are particular goals.

3.2 Semantic Web technologies

The Semantic Web has a layered architecture as a consequence of its step by step design and implementation, its use of open standards, and also because research in the upper layers is ongoing. At the bottom are syntactic elements and low level protocols which define standard text, data, and address encodings, some of which are based on existing Web standards. On top of these are the components which form logical reasoning: first knowledge representation, then descriptions of that knowledge, then argument and verification (Figure 1). Most of these standards are defined by formal specification to ensure interoperability, though some have emerged as de facto standards. The application of these layers leads to universal compatibility first at the level of simple data exchange, then structured data, then knowledge interpretation, and Berners-Lee's idea of a global database used by software agents.

3.2.1 HTTP and HTML

The existing content on the Web, and the protocol used to transfer it, remain an integral part of the Semantic Web. As an incremental step in making the human-readable Web accessible to machines, common scenarios will be making judgements about the properties of a document, for instance to determine its value or relevance to the user, or interpreting data within the document for

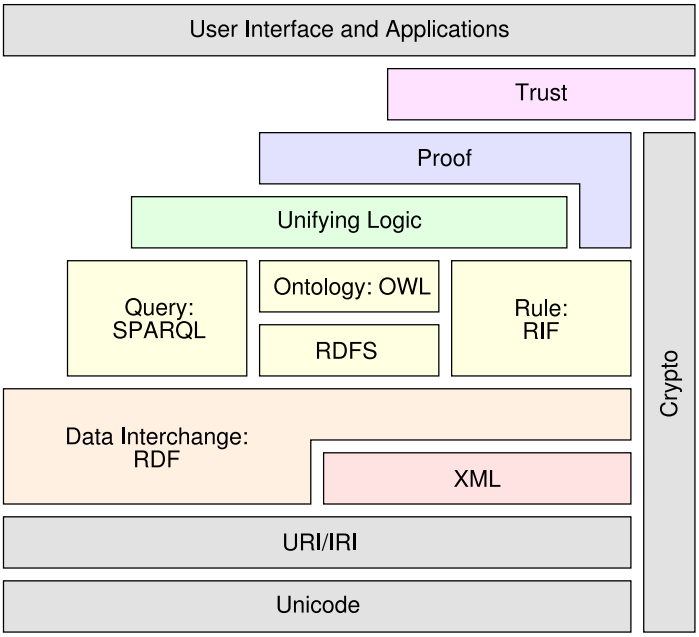

Figure 1: The Semantic Web 'layer cake'. From an example in [58]

it to be processed automatically, as in Berners-Lee's scenario of booking an appointment.

HTML[38] is the standard document format on the Web, so it is important that HTML documents can encapsulate or be annotated with Semantic Web metadata. There are various methods used; for example the Tabulator[39] interprets the `<link rel="meta">` tag. HTML has been annotated with knowledge before the existence of the Semantic Web, in SHOE[59], and with metadata, in PICS[60, 61]; but the Semantic Web is more interoperable, providing equivalent and extensible functionality for all document types as well as abstract data.

HTTP[62] is the transfer protocol for Semantic Web data. Its content negotiation, redirection, and metadata headers are used to integrate the Semantic Web with the existing Web architecture.

3.2.2 Unicode

One of the most fundamental standards necessary to exchange data is a universal text encoding. There are many character encodings in use on the Web, which may be incompatible and not accurately distinguishable; for example, the many

8-bit character sets referred to as "extended ASCII" have the characters from ASCII in common but use the remaining space for language specific special or accented characters. As the Semantic Web is intended to be global it is important that its standards take into account internationalization issues.

Recent Web standards have taken into account the existence of multiple encodings by allowing the encoding to be specified in the header. However, the Unicode standard[63] is a superset of all these 8-bit encodings and aims to incorporate all known character sets, so it is therefore possible to write text in multiple languages using a single Unicode compatible encoding throughout. UTF-8 is an ASCII-compatible encoding which has become the de facto default encoding of the Semantic Web; also the DNS (through Punycode[64]) and URIs (through IRIs[65]) are starting to support Unicode extensions.

3.2.3 URLs and URIs

The Uniform Resource Locator has been an essential feature of the Web since its origin[66, 67]. The concept of a stateless protocol (HTTP) and universal naming syntax for retrieval of resources set the Web apart from other hypertext systems, and contributed to its success by encouraging rapid growth despite introducing breakable links. To ensure links never break, a hypertext system would have to track linked resources through pointers and keep tight control of data integrity; this would require either central control or expensive distributed mechanisms, increasing complexity. In encouraging adoption and keeping the protocol simple, this design paved the way for the Semantic Web.

Links based on naming made the Web extensible. The prefix component allows URLs to refer to any resource including those accessible through other protocols, for example `mailto:`, `gopher:`, or `callto:`, potentially naming all retrievable resources. However, this is inadequate for the Semantic Web. A superset of the URL syntax, a Uniform Resource Identifier[66] refers to a resource which is not necessarily retrievable. A URI need not be the address of an accessible piece of data; it is meaningful to assign a URI to a physical object or even an abstract concept, and allows statements to be structured within the standard Web hierarchical namespace where appropriate. Semantic Web systems can therefore use this mechanism to make statements connecting physical and digital resources in a generic way[68].

The question remains of how to assign identifiers to resources, both physical and digital, in an appropriate way. It is of particular significance to the topic of Semantic Web features for repositories, so is discussed in detail later.

3.2.4 XML

The Extensible Markup Language[69] is a commonly used format to encode structured data on the Web. It is based on SGML[70] but is a restricted subset designed to be simpler to parse and validate. XML is expressed in plain text in any chosen encoding, and can include characters outside its encoding through the use of Unicode character entities. The meaning of data in an XML document is not specified by the XML standard[69]; the syntax is fixed, and a mechanism is provided to specify the semantics in an extensible way. Therefore XML is not itself a language but a standard way of defining languages; SVG[35] and XHTML[71] are examples of such languages. Much of the syntax of XML is compatible with HTML, so HTML documents can be easily converted to XHTML but still be parsed by existing parsers.

XML is suited for encoding Web data for many reasons. It is human-readable and also parsed easily. It can express arbitrarily complex data structures, though particularly suited to documents, lists, and hierarchies. Many parsers have been written for a great variety of platforms and programming languages, including free software implementations; combined with XML's inherent multilingual support this means it is accessible to as many potential users as possible. Documents can be built which contain data in more than one XML based language ("XML application") due to the use of namespaces: URIs which qualify the name of an element, allowing languages which use the same short name to be mixed in the same document. DTDs[69], adapted from the SGML standard, define the structure of an XML document; XML Schema[72] and RELAX NG[73] are more recent standards which define XML within XML and support more complex validation rules.

3.2.5 RDF

If XML is suited to encoding hierarchical, well defined structured data, RDF (Resource Description Framework) though apparently more constrained is suited to more flexible, graph structured, loosely defined data. It is used as a generic way of representing arbitrary information about Web resources[68]. The RDF data model is even simpler than XML, and can only represent one data type,

the triple: a statement with a subject, predicate, and object, with each part a URI (or literal string). However this simple structure is expressive enough to make arbitrarily complex assertions, including statements about other RDF resources.

Like XML, RDF is a metalanguage, which defines semantics suitable for creating domain specific languages. The meaning of an RDF statement is abstract and depends on the interpretation of the URIs which compose it, which is domain specific. However, a key use case for RDF is expressing metadata, so the specification does define core features expected to be used in this situation. A core concept is the "type" of a resource, which asserts that the entity referred to by the resource URI has a certain abstract data type. The standard also includes features for standardizing common complex data structures, including sets and lists. Beyond this it is possible to specify data using an ontology: a formal description of data types, properties, and the relationships between classes.

An RDF graph can be serialized for storage or transmission into several standard representations. A common, though verbose, mechanism is RDF/XML[74]: a constrained XML application which adapts the namespacing and hierarchy of XML to RDF. Because of the different syntactic rules of XML and RDF, it is not possible to use XML validation or transformation tools with RDF/XML, use namespaces, nor define a unique mapping from each RDF graph to XML. A more recent alternative is Turtle[75] which has a concise syntax designed specifically for RDF.

3.2.6 Ontologies

RDF has a schema language which plays the same role that a DTD or schema plays in XML. The difference is that while XML is a document, RDF is loosely structured data and may represent entities and their properties. RDF Schema[76] introduces an object orientated architecture, specifying a class hierarchy, compound data types, and restricting the domain and range of properties.

More expressive data description is provided by ontologies, which are a key part of Semantic Web interoperability. The word "ontology" comes from philosophy, referring to the study of the nature of existence; however in this context it refers to a formal representation of knowledge about a particular domain. Gruber offers a widely-used definition:

> A body of formally represented knowledge is based on a *conceptualization*: the objects, concepts, and other entities that are presumed to exist in some area of interest and the relationships that hold them. A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose. Every knowledge base, knowledge-based system, or knowledge-level agent is committed to some conceptualization, explicitly or implicitly.
>
> An *ontology* is an explicit specification of a conceptualization.[77]

An ontology goes beyond specifying the structure of data; it allows reasoning about that data to take place. For example, an ontology in the genealogy domain could specify that the "parent" property is inverse to the "child" property, meaning that if A is B's parent, B is A's child. This fact could be used to infer statements which were not specified explicitly in the data. The ontology could also be used to check the validity of data, such as making sure that a person has no more than one mother and one father, if this constraint were specified. These definitions and constraints are useful for software agents, which in Berners-Lee's vision crawl the Semantic Web for information to help perform tasks, as they can interpret data, deduce what is relevant from what is available and correlate different sources.

OWL (Web Ontology Language)[78] is a more expressive metalanguage built on RDF Schema, allowing more control over constraints. A strength of these RDF metalanguages is that they are expressed in the same syntax and thus integrated much more closely with the data they describe. (This feature is useful in the common case where a schema is used to annotate formal data with human readable descriptions.) OWL builds on earlier languages such as DAML and OIL; as RDFS and OWL have become the recommended Semantic Web standards, a study of the history and details of ontology languages in general is beyond the scope of this work.

Ontologies are vital in developing the vision of a universal web of data. For general purpose tools to work with it, data in common formats should be expressed in a common ontology. As the Semantic Web develops ontologies are being standardized for frequently used concepts, such as FOAF[79] for personal data and Dublin Core[80] for bibliographic data. Using a preexisting ontology for the standard parts simplifies the difficult task of ontology design as well as, by being interoperable, extending the domain in which new data is usable. In the event of conflicting ontologies being used to model the same data, the

conversion process may run the risk of losing precision or detailed semantics. Automatic or assisted mapping between incompatible ontologies is an active research area but will not be discussed here.

It is important to note that an ontology can define the relationships between concepts but cannot formally give an explicit meaning, as the meaning of data is intrinsic to the way that data is used.[1] Simply including an ontology therefore cannot guarantee that data is interpreted or used correctly, but it does provide a mechanism for validation, ensuring data satisfies the constraints of the ontology and leads to correct deductions. It is also possible to validate ontologies themselves, to ensure they are not contradictory.

3.3 Triplestores

Serialized formats such as RDF/XML are suitable for data exchange but are of course not intended to be used as an internal format for live manipulation. A triplestore is like a relational database for RDF: a repository for efficient storage and retrieval of RDF data. They are frequently backed by a relational database for persistent storage, and provide an API, query interface, or reasoning tools.

The proliferation of alternative software implementations of Semantic Web tools is a beneficial consequence of its standards and data exchange driven design. Applications are therefore free to choose software appropriate for their technical requirements, or develop their own versions as necessary. This software ecosystem is likely to contribute to the growth of the Semantic Web. In the case of triplestores there are notable differences in functionality between implementations rather than just compatibility. The following examples highlight the differences in features between some alternative triplestore implementations.

- Redland is a simple triplestore and associated tools, comprising RDF parsing and serializing, storage in a variety of backends, and query execution.[82] As an early implementation which predated many current standards, it is simple and portable but is designed to access triples individually, precluding many optimizations which can be found in other systems.

[1] This is the "symbol grounding problem"[81]. It is impossible to give an absolute definition of the meaning of abstract data, in the same way as it is impossible for a dictionary to define a word simply in terms of other words; definitions must rely on knowledge or experiences external to the dictionary.

Redland triples can optionally be annotated with a "context" URI: this is outside the visible data model but is useful for triplestore maintenance when merging data from many sources.

- 3Store is the University of Southampton's triplestore developed for the AKT project, which required storage and querying over a large body of diverse metadata. It contains RDQL and SPARQL query interfaces with some support for ontology based inference. Queries are executed by translating them into SQL statements to be interpreted by MySQL; thus they can benefit from its internal optimizations such as index analysis[83].

- Jena is a Semantic Web framework for Java, including a triplestore implementation, model API, and query interpreter. Its initial goal was to provide an expressive RDF API which would be easy to use for Java programmers.[84] Since then it has added support for new standards with a focus on optimization.[85]

  Jena's database structure is optimized to retrieve statements efficiently by reducing joins, and improve the performance of applications using reified RDF statements. Its frontend includes RDFS and partial OWL inference with an extensible reasoning engine.

- Sesame is another popular Java framework. Its focus is on integration with RDF Schema at the API and storage layers.[86] Sesame's database structure is influenced by the RDF schema applicable to the data, which optimizes queries involving inference but is dependent on the data model. Querying support has concentrated on SeRQL, a custom language, but SPARQL is now also included.

Different tasks will require different triplestore features. Run-time inference can be avoided if the dataset does not change, by computing the inferred triples and asserting them ahead of time. This may be significant where performance is a factor. As the de facto standard RDF query language, SPARQL support is desirable if the store is to be made queryable through the Web. Some applications may make extensive use of reification to express statements about triples.

Later in this work Redland is used for experimentation, because of its context support which simplified manipulating RDF data in bulk, combined with ease of setup and software prototyping due to its well designed API and tools.

3.3.1 Querying RDF

Query languages are an important supporting technology for RDF, which provide an abstract way for applications to access triplestores, above the level of direct manipulation of statements through an API. They have a number of benefits particularly relevant to the diverse reuse of distributed data which is the core of the Semantic Web user experience.

- Query languages, when combined with a common protocol such as HTTP, provide a standard way to access distributed RDF data. This supports the goal of integrating data spread across the Web and simplifies access to the data by users and developers.[87]
- A triplestore may be large or diverse, containing information which is nothing to do with the requirements of the particular query. Transferring a large amount of irrelevant or redundant RDF data could waste network bandwidth, and also requires processing power on the client to parse and query the data. Executing queries on a server with efficient access to the store optimizes this process.
- A query language provides an abstraction between the storage and application layers. It is simpler to write and modify a query, than to write a program to find out the information by examining the triplestore directly. This also permits the underlying triplestore implementation to change without affecting tools which communicate with it. Another benefit is that optimizations to the query engine can be aware of the triplestore's internal structure, and affect all the tools using it.

Many query languages exist, and newer languages are often designed to fix deficiencies identified by users of older ones. A detailed comparison of languages is not included here; a summary can be found in [88]. However, in response to the proliferation of incompatible query languages, the W3C's RDF Data Access Working Group[89] analyzed the existing query languages, produced a list of requirements, and developed the SPARQL standard to satisfy them.

3.3.2 SPARQL

SPARQL[87] is a W3C Recommendation and has become the predominant query language for the Semantic Web. Its syntax is plain text (some other languages use RDF or XML) and resembles SQL, the standard relational database query

language, in common with earlier examples such as RDQL and RQL. A SPARQL query is structured as a 'query by example', consisting of an outline which matches a graph pattern, with placeholders for the requested elements. Data can be returned as columns or as a constructed RDF graph. This example is taken from [87]:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?mbox
WHERE
  { ?x foaf:name ?name .
    ?x foaf:mbox ?mbox }
```

name	mbox
Johnny Lee Outlaw	<mailto:jlow@example.com>
Peter Goodguy	<mailto:peter@example.org>

The features of SPARQL include: numeric and string expressions, including regular expressions; support for language and datatype specifications on literals; named graphs, applying a query to a defined subset of a triplestore; and a concise syntax for triples with common components. It also includes a standard HTTP-based method for performing queries on remote datasets[90].

3.3.3 Trust

Despite the formalism and structure of data on the Semantic Web, it is still an incremental progression from the Web as it exists today. It is built on the same decentralized, uncontrolled, unreliable infrastructure leading to the same flaws—and benefits. The barrier for participation on the Semantic Web is very low, especially for existing Web based resources, and the freedom from control allows user groups to define and evolve their own standards, for example domain specific ontologies. However, it must be considered a hostile environment from the point of view of obtaining trustworthy and reliable information. As with the Web, resources can change or vanish without warning; but worse, there is scope for automatic processing of third party assertions which could lead to accidental or deliberate misinterpretation.

Just as a site can publish false accusations about a person, use incorrect keywords to gain a higher search ranking, or mimic another site to trick users into revealing personal data, it can be dangerous to make deductions based on untrusted statements of RDF. For example, a naive authentication system might

be fooled by the assertion: `<spectre:blofeld> <owl:sameAs> <mi5:bond>`. Metadata can defame or delude just as any other data can.

A significant aspect of this problem is that to a human, the data can seem legitimate; the machine has been fooled by metadata which might not even be displayed. With the potential power granted to Semantic Web agents to make decisions on people's behalf, the risks are possibly even greater than with human-readable data. What if, in Berners-Lee's familiar example[57], the agent had booked and paid for a consultation with a fictitious medical care provider, based on false metadata?

The ongoing research into trust on the Semantic Web is beyond the scope of this thesis. However, it will become more important as the use of Semantic Web technologies grows and diversifies that the issue is taken seriously.

3.4 Semantic Web systems for scientific communication

This section presents several systems which are relevant to the topic of scientific communication and are built on Semantic Web technologies. The examples in this section are intended to make a case for collaborative progress through data sharing, standardization, and organizational support leading to effective participation in the Semantic Web.

3.4.1 Annotea

Annotea[91] is a project to build an infrastructure to handle associations between Web content and metadata. Annotations were chosen as a simple subset of metadata which is easily defined and implemented.

Annotea aims to be a general purpose annotation infrastructure including client and server implementations. The focus is on designing an extensible, adaptable framework based on open standards. The core of Annotea specifies the data structure, format, and transfer protocol but leaving front- and backend detail to implementors. This is an important design decision because it allows future innovation in the way annotations are used, giving them the potential of any other type of metadata. Annotea structures are expressed in RDF, and are based on XPointer and XLink, which means it is capable of being combined with other metadata directly and can be manipulated with general purpose XML tools. The extensibility of RDF also allows new types of annotation to be easily defined through inheritance.

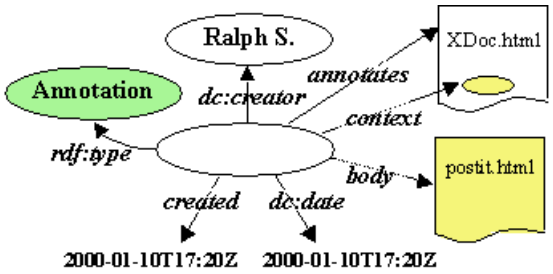

Figure 2: The RDF structure of an Annotea annotation.[91]

One of the major design considerations of Annotea is the distributed nature of its annotations. Similarly to open hypermedia, annotations are stored apart from the document they describe—this is made simple due to the power of RDF as a base format. This means a user can store personal annotations locally, and query author-provided or third party databases for more. Users read and modify annotations on remote servers using standard HTTP requests, so access control is handled using standard HTTP authentication methods.

A demonstration frontend was developed alongside the infrastructure, which is part of the Amaya research browser released by the World Wide Web Consortium[92]. This implementation communicates with an example server developed using Apache and MySQL. More recent efforts have added support for the Mozilla browser and extended the annotation concept to collaborative bookmarking and topic hierarchy generation[93].

Annotea represents an early adoption of the ideas and technologies which make up the Semantic Web. As such its relevance is primarily as an example of an integrated Semantic Web system; its design is simple but would be unlikely to handle a large scale deployment due to issues of scalability, provenance, and expressiveness. First of all, as in open hypermedia the database of annotations is distributed; while this has the advantage of supporting third party annotations, it means that there needs to be a mechanism for locating annotations on the current document. Their implementation submits each URL to every potential annotation server, via a simple HTTP request or RDF query language gateway, requiring a large number of queries as the scope of each server is unlimited, and also exposing the user to privacy issues. Their ontology defines the relationships between content and annotations, and can be extended to supporter a wider set of metadata, however by transferring them using a custom encapsulation in the Annotea protocol it does not completely treat an annotation as a first

class resource. The system also does not consider the impact of malicious use; there is no way to authenticate annotation users or prevent vandalism. The simplicity of its design means that it is a valuable demonstration of its concept, but unsuitable for real deployment.

The issues inherent in the distributed nature of Annotea suggest that the most useful future Semantic Web systems will have a cooperative element to their design and implementation—in the sense that users providing Web content can intentionally encourage third party integration. This will be particularly evident in the repository field as data quality and dissemination are particularly important.

Researchers could use annotations to comment on each other's work, with the thread of discourse modelled through the structure of the underlying RDF data. This is an example of an improved method of scientific communication which is dependent on the online medium and the distributed nature of the Semantic Web.

3.4.2 ScholOnto

An important part of the research process is evaluating new information in relation to existing knowledge. This includes identifying the new ideas put forward in a paper, characterising how these relate to existing ideas, and whether they represent a significant contribution to the field. This process of evaluation will give rise to discussion among researchers, which could be anything from personal annotation, informal discussion groups or formal peer review, and is facilitated in part by initiatives for online archiving.

Relationships between ideas effectively form hyperlinks between papers. These semantic links are at a higher level than simple citation links, in that they also have associated information about why the link exists. The ScholOnto system makes these links explicit, with metadata associated with each paper describing the ideas and their relationships. This is achieved using an ontology representing scholarly contributions and discourse, and a system to allow annotation of research material[94].

The ScholOnto ontology aims to represent claims about a document's accuracy, relationships, and significance. Unlike many ontologies the focus is on managing metaknowledge, not knowledge itself. All fields of research will by definition be dynamic, requiring updates to an ontology which attempts to describe them;

however, the process of research and discourse remains largely consistent. The ontology is not intended as a replacement for the text it describes, but as a summary which can be interpreted as an aid to the researcher.

The approach taken by ScholOnto superficially resembles Textnet[13], which was a hypermedia system with a similar taxonomy of link types. The differences between these systems are in the granularity of the data structure and the way a user is expected to interact with it. Textnet provided complex hyperstructure and was intended to be a new mode of authorship; the details of an argument were expressed in the hyperstructure as well as the text itself. ScholOnto is less ambitious in the change in working practice it expects from authors, observing that the complexity of a detailed link taxonomy can overwhelm users; as such the formal metadata only provides a summary of the concepts in the paper to assist in discovery and is not intended to be a substitute for the text. This requires less commitment from the author and can be more simply applied to existing works; it also benefits from compatibility with Semantic Web tools.

Once described in this form these connections can be analysed, which allows highly expressive queries to be made easily. Some examples are "find papers which extend this idea", "find examples of this method in another domain", or "find criticisms of the foundation of this technique". Previously this kind of searching would require great expertise and would be impossible to perform automatically. This also facilitates automatic reasoning about connections by software agents.

Linking concepts in this way forms a natural layer above traditional hypertext linking, where documents are replaced with ideas or concepts. Many of the same analysis techniques can be extended to this space: instead of authority documents, there are widely accepted theories; instead of communities of practice, there are "perspectives"—schools of thought. Advances in the field of standard hypertext analysis would benefit this type of linking, and vice versa; these semantic links could be used to augment or provide a base for other link discovery systems.

The stumbling block in this effort to formalize the substance of academic discourse is that users are required to organize their thoughts and ideas into an artificial structure. The reader must build statements rigidly defining their view of the concepts in the document and how they relate to each other and related work, based on their indistinct mental model. Since developing the ontology work in this project has moved towards knowledge extraction techniques such as natural language processing, and designing user interfaces to assist in collect-

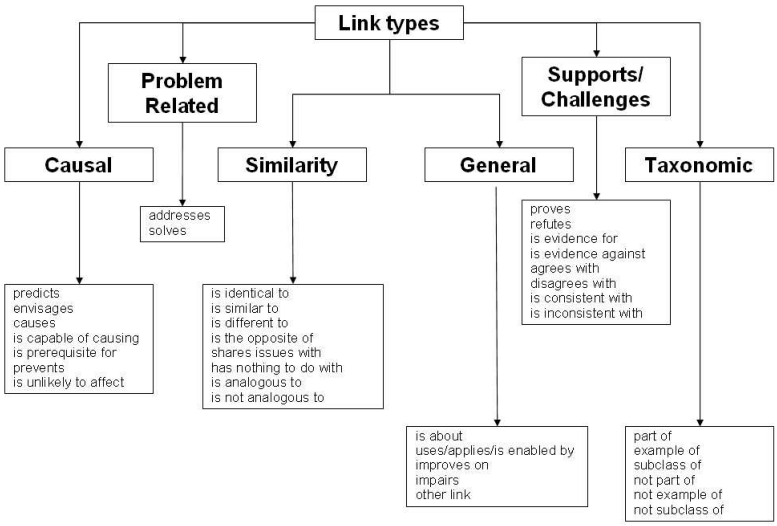

Figure 3: The relationships modelled in ScholOnto between concepts of academic discourse.[95]

ing this type of data.[96, 95] User evaluation revealed that dealing with formal representations of concepts and relations was problematic; more recent user interface experiments have attempted to abstract this behind a tagging system or provide a visual representation.[97]

Compared to Annotea, ScholOnto operates at a higher conceptual level. While Annotea's ontology describes the relationships between annotations, it stops short of trying to describe the meaning of the content; the body of an annotation is generally human-readable formatted text. Annotea can therefore hide the details of its internal structure behind a simple user interface to create the different types of annotation, whereas a ScholOnto user must be aware of the structure of their model as they are directly constructing it. The lesson here is that users will engage more readily with a system which hides the complexities of the underlying data, and presents them with a free text or multiple choice interface. To enhance repositories with Semantic Web technologies it is therefore desirable to keep the formal definitions in the domain of expert knowledge engineers, and permit the user to interact at a more natural level.

3.4.3 Magpie

Magpie[98] is a system which aims to overlay knowledge from the Semantic Web onto a user's Web browsing, to provide them with contextual information and supporting services with the minimum of effort for the user. It resembles a citation linking system except that it includes links between arbitrary concepts, such as people and research projects, extracted from ontologies selected by the user. As a semantic link and data browsing engine Magpie can be compared to KIM[99], which focuses on natural language extraction techniques with a similar user interface, and WiCK[100] which implements similar data awareness features in the office/document space.

The user interface uses browser integration to mark keywords detected in a Web page which correspond to concepts in the ontology. This process is similar to the open hypermedia links provided by the DLS[31] and COHSE[32] but is rather intended to provide supporting knowledge awareness rather than hypertext linking, and is thus more related to the Semantic Web than to hypermedia; Magpie focuses on data rather than documents. As well as linking connected concepts Magpie informs users when another user is viewing related data, and faciltiates communication between them. Like other systems discussed here, Magpie aims to hide the details of the Semantic Web data model and support natural interaction. A notable interface is the trigger mechanism, which monitors the entities associated with the user's browsing activity and proactively identifies related resources.

While Magpie is a general purpose system for working with semantic data, it relies on text extraction techniques which are highly specific to the domain of the ontology chosen. It is therefore arguable that its key benefits could be implemented in a more cooperative way, requiring integration with the appropriate sites but thereby making the information more valuable in general. For example, the problems of trust and privacy exist here as in other systems, which would be exacerbated by the push model of notifications outside a controlled environment. This mode also depends on two way communication between the browser and dispatcher components. A cooperative approach could use page annotation with appropriate ontologies, with a separate notification framework based on RSS, increasing bandwidth and processing requirements but giving the client control over information flow.

Magpie has a flexible architecture for associating information with search and collaboration services, so a site driven approach with a pull model could be

added. A repository could be an ideal centre around which to build a community of collaboration, but a structure centralized around the repository would be more appropriate as it would have access to the internal data structures, and could therefore more accurately and logically annotate each page with concepts. Given the effort already invested in creating and maintaining a repository, making use of the formally structured metadata is likely to yield more useful results than a generalized keyword matching engine. Narrowing down the scope of keyword matching is the method used by COHSE to produce better targetted links, but this aspect of keyword matching is a manual process driven by the user in Magpie. The authors argue in [101] that semantic links between keywords and concepts could make an area of knowledge more accessible to newcomers. However in academic publishing this goal may be better served by more refined relationships inherent in the citation and community of practice structures. An example of this is WiCK's awareness of document context (the domain is filling in specific structured forms for research funding) where acting on a link from a particular section of a document to an entity pulls in data appropriate to that section.

3.4.4 CS AKTive Space

CS AKTive Space is a Semantic Web application which allows users to explore a database about Computer Science research in the UK[42]. The system combines data storage and querying, knowledge acquisition, ontology development, and interaction. It is part of the AKT project at Southampton, which aims to apply knowledge management techniques to make best use of data on the Web.

CS AKTive Space consists of a triplestore, 3store, which provides storage and querying in RDQL. Alongside are tools to maintain and manipulate the triplestore: these include Armadillo[102], a search service which uses existing knowledge as an aid to searching the Web for more information, which is then added to the triplestore; Ontocopi[103], a Community of Practice analysis tool to allow users to search for people in the same research community; and harvesting tools which keep the repository up to date based on Web sources. On top of everything is a user interface layer which allows exploration of the data guided by its semantics; including a node centric browser and mSpace[41], a column based flexible searching/browsing tool.

The data is formally described by the AKT Reference Ontology[104], which includes both the central concepts (people, publications, photos, institutions,

and so on) and a full formal description of their properties in OWL for verification and extension. The effort involved in producing and applying this ontology makes the stored data more suitable for third party reuse, including the possibility of fully automatic processing via ontology mapping. Most of the data in CS AKTive Space was imported from many heterogeneous sources and translated to the AKT ontology using dedicated tools, though the more desirable model would be to have updates pushed from the source in a standard form. As part of the import process both manual and automatic coreference handling was required.

The user interface for CS AKTive Space is based on direct exploration of the data. While the details of RDF syntax is hidden, and identifiers are replaced by labels where they are available, navigation follows the structure of the RDF graph from a node to its related resources. The mSpace interface provides an interactive exploration interface, but it is also closely tied to the properties directly associated with the target resource. Therefore, the ontology must explicitly model the relationships likely to be of interest to users. This is generally true in CS AKTive Space but may not be for all Semantic Web systems; an example from CS AKTive Space is the Ontocopi Community of Practice analysis tool, which queries the RDF to discover relationships which are not asserted explicitly. The only other user-accessible interface which can search for indirect relationships is the RDQL engine.

CS AKTive Space demonstrates the value of a rich collection of Semantic Web data, and highlights the importance of a good user interface and knowledge management techniques. The effort required to produce such a collection was substantial, partly because most of the primary data sources were in custom formats and needed to be processed and integrated with manual intervention. Were this data, already collected by various authorities for internal use, published in a standard machine-readable format as a matter of course it would be simple to build aggregation services which add value to data in this way. Open Access and the OAI (discussed later) support this idea for publications, but the user interaction in CS AKTive Space shows that easier access to other related data would further improve this by supporting browsing and searching. An effort to publish such data by publishers, conference organizers, institutions, and other related organizations could benefit all parties; an example is the publication in RDF of metadata from the International Semantic Web Conference[105], and the tools which were developed to interact with it such as Flink[106].

3.4.5 SIMILE

SIMILE is an MIT project aiming to apply the Semantic Web to improve interoperability between collections of digital data. Specifically the project focuses on the DSpace repository software[107] (discussed later) and a set of RDF exploration and visualization tools. Several notable tools have emerged from MIT's work in this area[40]:

- Fresnel[43], an ontology for displaying RDF. The ontology defines "lenses", which for each data type describe the style features and associated properties which should be displayed. Fresnel is intended to be used by RDF visualizers to provide a consistent rendering of data in different systems.
- Welkin[108], a graph visualizer for model builders and analysts to examine the structure and characteristics of the dataset. It provides an overview of the ontologies and predicates commonly used, the degree of connectivity, and the density of the graph.
- Longwell[109], a generic faceted browser for end-user exploration of RDF data. Faceted navigation is the process of narrowing down a search by specifying the values of several separate properties; as opposed to hierarchical browsing, the properties are independent. Longwell resembles mSpace superficially but has a different interaction model; mSpace focuses on the flexibility of user rearranging columns, multiple selections, and highlighting for interactive exploration; Longwell uses predefined facets and concentrates on efficiency and interoperability.
- Piggy Bank[110], a user interface for aggregating and browsing data gathered from the Web. Using screen scraping templates it can generate RDF from structured HTML, which users can combine to make "mashups" which repurpose the data. Therefore the data can be explored outside the domains specified by the website which provided it, and through different visualization methods such as maps.
- Haystack[111], a related project for managing personal information with an RDF backend, which can be considered a Semantic Web browser. Haystack exploits the universal data model provided by RDF to build a single information client which can integrate all the different pieces of data normally managed by separate tools: email, calendar, photo album, and publications are all examples.

Overall SIMILE focuses on RDF as a universal data type. The tools developed manipulate RDF directly, using it as the internal storage format and providing display, processing, and query features. This approach is clearly appropriate for the consumer of heterogeneous Semantic Web data, as in the use case for Piggy Bank; but when applied to producers and archivers of predictably structured data where efficiency may be a greater concern, RDF may only be suitable as a data interchange format with traditional relational databases used internally. The Sesame triplestore bridges this gap by using an RDF schema to define the structure of a relational database, but this is achieved by sacrificing the ability to store arbitrary RDF data. Note that DSpace uses an optimized structure internally despite being a focus of the SIMILE effort (as does EPrints); also the mSpace project had to move from RDF-based storage to a traditional indexed RDBMS to provide acceptable interactive performance with a large data set[112].

3.4.6 Tabulator

The Tabulator[39] is a generic browser for the Semantic Web, which integrated with a Web browser allows distributed exploration of the global RDF graph in a tree or a variety of domain independent visualizations. Compared to other RDF browsers it aims to provide a rich user interface but avoid being tied to a closed data set and domain, and so provide a resource-centric view of the linked data of the Semantic Web. It specifically concentrates pragmatically on the distributed nature of the data, attempting to resolve any URIs requested by the user or which might contain an ontology or further information, and also parsing the headers of HTML documents discovered for links to alternative RDF versions.

Intended as an open ended experiment, the Tabulator provides a useful service for a technical audience of Semantic Web developers and computer experts. It usefully investigates the problem of browsing a potentially unbounded graph and demonstrates the benefits of RDF as a universal data format. In particular it highlights the difficulty in discovering relevant RDF data; the mechanisms used by the Tabulator work well but rely on following explicit links in the source data, and as such are largely in the control of the original source. As a general, universal RDF browser the Tabulator user interface is admittedly inferior to domain specific browsing systems where they are appropriate; suggested solutions are integration between domain and generic systems and user interface ontologies such as Fresnel.

This is another example of how cooperation is an essential feature of a successful large scale Semantic Web. Standards and policies which assist in locating and describing related resources can support both generic and domain specific browsing tools. For example a future Tabulator could take advantage of the W3C's standard remote SPARQL query protocol[90] if it were adopted by a critical mass of data providers, allowing a larger or more relevant set of related data to be obtained.

3.5 Searching the Semantic Web

How will the Semantic Web be searched? This is really two related questions, as Semantic Web data can have a dual purpose. It can be metadata describing a resource which the user wishes to find, in which case a search engine could conceivably use this data to refine its search algorithm by resolving ambiguity or as an alternative to text-based keyword extraction. Data can also be itself the target of the search, as the machine accessible nature of the Web develops. This is in line with Berners-Lee's idea of an agent system but could also be accomplished with a traditional search engine.

People discover material on the Web through three mechanisms: direct navigation to a page by entering its URL, browsing by following hyperlinks from other known resources, and searching an index using a search engine. In the first case the URL might be seen in advertising or recommended by a friend, and in the second by reading a hyperstructured document or catalogue, or looking up a citation. Both of these techniques require bootstrapping; to find a new resource a related one must already be known. Searching is different, as it only requires knowledge of the characteristics of the desired resource, usually keywords.

Web search engines use a variety of mechanisms to find new pages and include them in search results. Pages can be submitted manually or discovered by following a hyperlink from an existing page. Keywords to describe each page may be taken from the text of the document, incoming and outgoing links, or metadata in headers. The quality of each page is determined with an algorithm based on these characteristics and ranked to present to the user.

In restricted domains smaller scale search tools can be used which can be more expressive, such as a library catalogue, e-commerce site, or corporate database. Knowledge of the domain can allow the user to refine their search by specifying metadata fields more precisely or appropriately limiting the range of the data set. Examples include searching a library for books about Charles Darwin, by

filling in the title field, rather than books *by* Charles Darwin which would also appear in a general Web search; limiting a personnel database search to only include full-time staff, where the underlying database holds this information; or looking for the novel rather than the film of "Sense and Sensibility" on an online shop. Even choosing to use a site's search rather than a general Web search is a useful restriction; a result guarantees that the item is sold by that retailer, or available in that library. The key aspect of these searches is that they operate based on knowledge of the internal structure of the data, and so are by definition more expressive than a pure keyword search.

Of course the data returned by these restricted searches, if public, could be found by carefully choosing keywords in a general search; but this approach has drawbacks. The right keywords must appear in the target page, so vocabulary and language differences might prevent the page being found. On the other hand, there may be no appropriate keyword which limits the result to what the user wants, as the words might be commonly found on many unrelated pages. Also the lack of formality requires a human with experience in using search engines to select keywords to come up with results quickly and reliably.

The Semantic Web promises to extend the power of these specific searches to the whole Web. Whether searching the Semantic Web is based on active agents or passive spidering, the key is to publish and standardize the data. For actively interrogating a database, SPARQL or another language can be accessed through an HTTP interface, and raw RDF can be integrated with the human-readable content or provided for bulk indexing. Both approaches can be supported, and as the data appears the tools will arise to search it, as occurred with the Open Archives Initiative protocol.

3.6 Summary

The Semantic Web is a primarily machine-readable layer of linked data which integrates with the current Web. Its aim is to encourage automatic processing of Web data for the benefit of users.

Like the current Web, the Semantic Web is likely to be an anarchic, unreliable, hostile environment for services attempting to gather useful data. However, the success of search engines demonstrates that is possible to analyze such data and its connections and making a judgement about its value. By openly publishing data, defining its structure with an ontology, using standard definitions where

appropriate, and cooperating with others, services can encourage this process and provide a more accessible and higher quality information resource.

Semantic Web technologies are suitable for modelling and interacting with all types of data, but technical, management, and usability issues may mean a specified data model may be more appropriate for clearly definable systems. Despite this, the Semantic Web can provide interoperability and extensibility for such systems, particularly to encourage data reuse.

The examples in this chapter suggest that complex user interfaces are only appropriate for technical users. In non-technical settings RDF data should be used to enhance the user experience behind the scenes, but not be viewed directly.

Where it is desirable for information to be shared, the Semantic Web provides an efficient mechanism. Repositories of academic material are an example of a situation where this is appropriate, as they store linked, structured data which is disseminated as a matter of principle, according to the idea that efficient dissemination of research leads to greater value. Therefore the technologies of the Semantic Web should be adopted within repositories, with explicit awareness of standards and the needs of the wider community, to facilitate better scientific communication.

4 Repositories

Repositories form the central point in this work, around which the technology of the Semantic Web is applied for the benefit of scientific communication. This section will describe the role of repositories in academic dissemination, repository server software (focusing on EPrints, the server developed at the University of Southampton), and the technical and organizational issues which are significant.

4.1 What are repositories?

The term "repository" does not simply mean one of the software tools discussed later. A repository is also a social construct: a commitment on the part of an organization to collect, manage, and disseminate digital materials. It is also intimately connected with the community of users which surround it.

Crow presents a brief definition, that institutional repositories are:

> [...] digital collections capturing and preserving the intellectual output of a single or multi-university community.[113]

Crow's position paper advocates repositories as a new mechanism for scholarly communication which is more accessible and provides economic benefits to institutions, escaping the domination of the commercial interests of journal publishers. In this scheme repositories are the primary method of dissemination in the first instance, with review and publicity the responsibility of third party services and the institution. Hence Crow's definition also highlights interoperability and the publicity benefits for the institution:

> Stated broadly, a digital institutional repository could be any collection of digital material hosted, owned or controlled, or disseminated by a college or university, irrespective of purpose or provenance. Here, however, we will narrow our definition to focus on a particular type of institutional repository—one capable of supporting two complementary purposes: as a component in a restructured scholarly publishing model, and as a tangible embodiment of institutional quality.
>
> Defined for our purposes then, an institutional repository is a digital archive of the intellectual product created by the faculty, research

> staff, and students of an institution and accessible to end users both within and outside of the institution, with few if any barriers to access. In other words, the content of an institutional repository is:
>
> - Institutionally defined;
> - Scholarly;
> - Cumulative and perpetual; and
> - Open and interoperable.[113]

Lynch's interpretation has repositories in a supplementary role to existing communication methods, with a greater focus on long term preservation and the broader organizational commitment required to maintain a repository. He defines a university institutional repository as:

> [...]a set of services that a university offers to the members of its community for the management and dissemination of digital materials created by the institution and its community members. It is most essentially an organizational commitment to the stewardship of these digital materials, including long-term preservation where appropriate, as well as organization and access or distribution. [...] At any given point in time, an institutional repository will be supported by a set of information technologies, but a key part of the services that comprise an institutional repository is the management of technological changes, and the migration of digital content from one set of technologies to the next as part of the organizational commitment to providing repository services. An institutional repository is not simply a fixed set of software and hardware.[114]

He argues that "scholarship has become data intensive"; and as the institution is in the position to be most sensitive to the needs of its faculty, institutional repositories should preserve whatever supplementary data and tools are produced alongside the published material. This is required due to the importance of long term preservation to scholarship in general:

> Preservability is an essential prerequisite to any claims to scholarly legitimacy for authoring in the new medium; without being able to claim such works are a permanent part of the scholarly record, it's very hard to argue that they not only deserve but demand full consideration as contributions to scholarship.[114]

Lynch also emphasises the distinction between scholarly publishing and scholarly dissemination. In his view a repository is not intended to replace traditional publication methods in the short term, but to open new avenues of scholarly communication alongside. For example:

> Institutional repositories also have roles beyond disseminating and managing the works of individual scholars that are part of the dialog of scholarly communications[...] to digitally capture and preserve the many of the events of campus life–symposia, performances, lectures. Institutional repositories offer a framework for organized stewardship and accessibility of these materials.[114]

Lynch and Crow both concentrate on university-managed repositories, but others are also of interest. Lynch sees disciplinary repositories as building upon existing institutional infrastructure using harvesting protocols, to simplify the process for submitters. Crow points out that discipline specific repositories do not benefit from the stability inherent in an established institution, and have been most successful in fields where there is existing practice of preprint communication, such as cognitive science[115], economics[116], and physics[117].

To summarize, for an institution a repository is a way to take responsibility for its own research output. It ensures that work is always accessible and never lost, and that the potential impact of the research is maximized by ensuring its accessibility.

The Open Archival Information System (OAIS) reference model[118] is a formal description of the process of long-term preservation of material in an archive, and is applicable to institutional repositories. It is an abstract model and set of terminology[119] which can be used to describe the components of a repository and ensure it satisfies institutional needs for ingest, preservation, and dissemination[120].

4.2 Academic publishing

How does submitting work to a repository relate to the traditional mechanism of formal scientific communication: publishing in a journal?

A journal provides several services, including dissemination, peer review, and content control. Physical dissemination has been made less significant, or even redundant, by the ubiquity of online distribution; however aspects of this service

are still relevant. Peer review aims to ensure that the article is scientifically sound and of high quality by subjecting the work to the scrutiny of experts in the field for anonymous criticism. Editorial oversight ensures that articles have a consistent style and contain clear language and graphics. Journals also provide other types of quality control, for instance by accepting only articles in a defined field so researchers need only read the journals which are appropriate to them.

Since the World Wide Web journals have moved online, but in general retain the structure and format of the paper version. Though the expense of printing and distributing the paper copy has been removed, institutions must generally still pay a subscription to be granted online access to the journal's content; as this is how the journal's services, including peer review, are funded. This is paradoxical as authors' aim is for their research to have the greatest impact, but restricting access only to subscribers will permit only subscribers to be influenced by the article. Harnad humorously explores the paradox in this situation in [121]; apparently to protect an author's interests, universal access to their work is prohibited by the journal who holds the copyright, leading to limited impact and reduced success as a researcher.

Harnad considers[122] peer review to be the critical component of academic communication, serving to highlight the best research by ensuring its accuracy and selecting it for the most prestigious journals; he makes the distinction between the service of peer review and the additional products offered by journals, which are optional extras from the point of view of authors and readers.

It is worth noting however that the peer review process is not a magic bullet. It can suffer from bias and conflicts of interest, and is ultimately vulnerable to human error. A reviewed paper could be discredited by future research, or contain subtle mistakes or deliberate fraud. Despite this, it is still considered a vital part of scientific discourse[122, 123].

In an alternative funding scenario, the cost of organizing peer review could be met by the author's institution, paid for out of the savings in journal subscriptions[121], with publishers scaled down to become primarily organizers of peer review and selectors of content. On the other hand, Morris argues that the journal submission process maintains quality standards, and the "branding" of a journal publicises and places research in context, benefitting authors and readers[124]. Her position is that in the event of a large scale shift towards self-archiving in author behaviour publishers may need to adapt to survive, but that the services they provide besides peer review are of value to the academic community and should be preserved[125].

4.2.1 Open Access

Given that the marginal cost of online dissemination of research is minimal, the high charges and restrictions that journals impose for access have become harder to justify. The Open Access movement is an effort to remove these constraints, allowing access to scholarly works to be available to all potential readers. A formal statement of this aim arose in the Budapest Open Access Initiative:

> An old tradition and a new technology have converged to make possible an unprecedented public good. The old tradition is the willingness of scientists and scholars to publish the fruits of their research in scholarly journals without payment, for the sake of inquiry and knowledge. The new technology is the internet. The public good they make possible is the world-wide electronic distribution of the peer-reviewed journal literature and completely free and unrestricted access to it by all scientists, scholars, teachers, students, and other curious minds. Removing access barriers to this literature will accelerate research, enrich education, share the learning of the rich with the poor and the poor with the rich, make this literature as useful as it can be, and lay the foundation for uniting humanity in a common intellectual conversation and quest for knowledge.[126]

As journals frequently require copyright to be assigned to them by the author as part of the publication agreement, initially it would seem that they are in a position to prevent authors archiving their own works. However, Harnad identifies two mechanisms which provide free Open Access to every reader: the "gold road" and "green road". In the former case, the journal grants free access to its material and adopts an alternative funding model; in the latter, authors archive the preprint, over which they retain copyright, and subsequently publish corrections after the work has been peer reviewed, self-archiving in an institutional repository. The BOAI advocates both self-archiving and publishing in open access journals as complementary routes to the goal of Open Access.

Harnad's vision of Open Access to research material through self-archiving was a driving force behind the development of institutional repository software, and the Open Archives Initiative promoting interoperability and standards. As such this work focuses on Open Access research material. However, technology which improves access to such material can also be of benefit to restricted subscription-only material, provided that the appropriate metadata is available.

4.3 OAI

The Open Archives Initiative is an effort to promote cooperation among digital archives of scholarly works. This aim is achieved through defining standard specifications for data exchange, supporting interoperability and the development of added value third party services.

The OAI emerged from an exploratory meeting in Santa Fe[127], sparked by the growth in eprint archives and the implications this had on the process of scholarly communication. At the time the earliest examples of successful archives, such as arXiv and CogPrints, were demonstrating that an alternative model to formally published journals could be viable and take advantage of the Web, grant freedom to authors, and solve the shortcomings of the peer review process.

The key technical development after the Santa Fe meeting was the Protocol for Metadata Harvesting (OAI-PMH[128]), which is an important interoperability mechanism for several of the tools which work with repositories. The use of a common protocol simplifies the construction of services based on data from multiple archives, by allowing the metadata to be harvested and processed centrally[129]. For example, OAIster[130], CiteSeer[131], and Google Scholar[132] (until recently) use OAI-PMH to provide metadata search services in this way, and CiteBase[133] harvests the full text and parses citations for linking and analysis.

4.3.1 OAI-PMH

Designed to be generic, simple, and easy to implement, OAI-PMH provides a standard way to harvest metadata records from a repository, as well as obtain basic information and supported metadata types[134]. The protocol is HTTP-based, using a common URL with a parameter to select the type of request. Notable features are: extensible metadata schemas, but with mandatory Dublin Core for base level interoperability; timestamps on metadata updates to support incremental harvesting only of changed content; hierarchical sets to permit harvesting of defined subsets; and a worldwide identifier scheme in the `oai:` URI namespace, to support a central resolution service.

An OAI record is an XML document with a standard header, metadata payload, and metadata about that metadata, such as rights information. The protocol defines the following “verbs”:

Identify Obtains basic information about the repository and its level of support for OAI-PMH.

ListMetadataFormats Lists the metadata schemas which are supported by the repository, or which are available for a particular item. Each schema has a namespace URI and a prefix, which is used to select the desired type when requesting metadata.

ListSets Lists the named groups of records available for selective harvesting. These are defined by the repository to represent meaningful subsets, such as collections or records of a particular type. Each set has a Dublin Core record describing its meaning.

GetRecord Obtains an individual record in a particular metadata schema.

ListRecords Obtains a group of records by named set or date range, and metadata type.

ListIdentifiers Obtains a list of record headers without metadata.

To facilitate harvesting large datasets over HTTP, which is stateless, the protocol provides resumption tokens as temporary identifiers for an ongoing session. Each individual HTTP response returns a limited batch of records; a request with a resumption token will return the next batch.

4.3.2 OAI-ORE

More recently the OAI has developed a protocol to exchange complex digital objects. OAI-ORE (Object Reuse and Exchange)[135] provides a similar interoperability mechanism for objects as OAI-PMH provides for their metadata.

ORE models aggregations, where an object is made up of component parts, each of which has a URI. The aggregation object defines the relationship between each of the components. An ORE Resource Map contains an aggregation object and its metadata, represented in RDF/XML or the Atom syndication format. A typical usage of ORE would be to represent a compound object, such as an eprint made up of multiple documents, for data exchange with another repository. The Resource Map can differentiate between an eprint with multiple renderings of the same content, and one with multiple related materials combined.

ORE is not a replacement for PMH. Resource Maps can be transferred over simple HTTP requests or a batch transfer protocol such as PMH.

4.4 Repository software

The earliest Open Access repository, arXiv[117], and other early archives used custom software. However, it was concluded at the second OAI meeting that a generic repository solution was necessary to promote the creation of such archives[136]. The first version of EPrints was produced to satisfy this need. Since then EPrints has been enhanced and other repository systems have been developed. They will be briefly examined here; a more thorough technical evaluation is available in [137].

4.4.1 EPrints

EPrints[138] is the institutional repository software developed at the University of Southampton to promote the OAI protocol. It provides a database of submitted documents with descriptive metadata and full text, accessible through a Web interface for browsing and searching, submitting new material, and administration. Available for Unix and Windows platforms, it is free software written in Perl and driven by Apache and MySQL.

The software was created to fill the need for a simple way to set up an OAI-compliant repository, based on the success of CogPrints[115], a repository of self-archived papers modelled on arXiv[139] which demonstrates the potential of Open Access. It was extended to to support customization for the dissemination of research in general and the use of OAI-PMH for interoperability.

4.4.2 DSpace

DSpace[107] is a digital library server created by a joint MIT and HP project. Though generally similar to EPrints it has significant technical and administrative differences. It is presented as an organization-wide information management solution with a broader scope to EPrints, requiring more significant planning and initial investment but focused on organized collections and preservation rather than immediate access and rapid deployment. It is written in Java and based on Apache Tomcat and JSP, with PostgreSQL or Oracle backends.

Notable features of DSpace include organization of objects into collections with different submission workflows, an integrated database but separate interfaces for different organization units, OAI-PMH support and syndication with RSS,

and integration with the Handle System[140] to assign a permanent identifier to each object and collection.

4.4.3 Fedora

Fedora[141, 142] is a storage and retrieval system based on Web Services technology which can be used as the backend for a repository. It models data and metadata streams with associated behaviour objects but relies on external modules to provide a user interface. The core component provides an object model, relationships between objects and data streams, and preservation.

Fedora is notable for its scalability and modular architecture. It is designed to have broad applications in general archiving of digital objects and connect to diverse external systems through Web Services, potentially providing a common information layer for a heterogeneous environment.

4.4.4 Comparison

From the point of view of hosting an institutional repository, these systems all have a broadly similar purpose. Their main differences are technical, such as the choice of programming language, and the assumptions made about an archive's typical configuration which stem from their origins. EPrints was written to kick-start the use of OAI-PMH, and was initially intended to be run by individual departments or research groups similarly to CogPrints. DSpace had a broader goal of managing an entire organization's output, which was typically the domain of libraries; it therefore has structure and metadata taken from a typical library scenario. Fedora is even more abstract and was designed as a generic information retrieval system, focusing on scalability but requiring front end integration by the user. Despite these differences, any of them would have been suitable as a platform for Semantic Web development as the code for each is available and could be extended.

Both Fedora and DSpace default to Dublin Core basic metadata. Dublin Core was created to be a ubiquitous metadata scheme on the Web which would be simple and applicable to describing any type of data[143]. However, it suffers from broad semantics which can mean the meaning of a particular Dublin Core property is variable and dependent on context. EPrints, by pushing for Open Access, is designed to capture whatever metadata is appropriate to make it a suitable repository for any type of content by default. Based on the idea that the

user's needs cannot be reliably anticipated, it includes a broad default scheme with many non-Dublin Core optional fields, while also making it easy to add new fields. A drawback of this approach for interoperability is that the OAI-PMH interface must be updated to map internal fields to Dublin Core, or else use a non-standard schema or omit those fields.

OAI-PMH provides an interoperability infrastructure which is common to each of these repositories. Thus to a certain extent they are interchangeable from the point of view of an end user or an aggregation service, as each can store the basic set of Dublin Core metadata, and this is the foundation for services such as OAIster[130] and Citebase[133].

My work with repositories focuses on EPrints primarily because it is developed at Southampton, which meant technical assistance was readily available when necessary. It is also the software used to host archives for the university and the School of Electronics and Computer Science, which provided a source of data for experimentation. However, the fact that EPrints is particularly designed to support Open Access[136] also makes it suitable for experimenting with the technology of the Semantic Web:

- A guiding principle of the software design is that it is impossible to anticipate the needs of all users. Therefore the Perl source code is written so it can be easily modified to support new features. This is particularly true of EPrints 3, which has a plugin framework for certain types of features including strong support for converting between input and output formats.

- EPrints is designed to have customizable metadata. While a default metadata scheme exists, it is not restricted to a standardized set such as Dublin Core or MARC, and new fields can be added easily by the administrator. This allows the metadata to be restructured to support identifiers; this is discussed in the next chapter.

- EPrints separates the concepts of a record and the documents it contains. A record is an abstract work, such as an article, and its metadata contains fields such as the author and publisher; documents are files or groups of files which are expressions of that work, with metadata about the file format and its relationship to the overall record. This feature can be used to store multiple formats of the same content, or separate but related content such as the data supporting a scientific paper. Each of these structures has

its own identity and can be given different metadata, either inside EPrints or in the Semantic Web.

- Metadata fields can be structured, with separate subcomponents. This feature can be used to associate a machine-readable identifier with a value. In EPrints 2, personal name fields as a special case were associated with an ID field, which could be used to unambiguously relate the name to an external source such as a staff database. EPrints 3 extends this concept to any field by supporting arbitrarily structured fields.

In the OAIS model[118], the EPrints software is a component in the Ingest, Archival Storage, Data Management, and Access functions of an archive. Its Web front end interactively accepts and validates a Submission Information Package; data can also be imported programmatically. The Archival Information Package produced is stored on disk, with the associated Descriptive Information in a MySQL database. Archival Storage, reliable preservation of stored data, is provided through the operating system, backup procedures, and institutional policy. Data Management is the storage and handling of the Descriptive Information (metadata) and consists of the EPrints back end API and scripts. Access to the stored data as a Dissemination Information Package is through the Web interface for end user browsing, and the API and OAI interfaces for data exchange and bulk processing.

OAIS is a model of the whole archive ecosystem, not just the software, so each of these functions involves human intervention, such as making policy, maintenance, and ensuring that the archive meets the needs of the Designated Community—the target audience of the stored data. With Open Access, the diversity of this audience is an important factor in the software design.

4.5 Repositories for general data

Institutional repositories are the focus of the remainder of this work. However, repository software provides a flexible platform for storage of data in general. The same storage, management, and retrieval mechanisms which apply to academic papers also apply to museum collections or image galleries; they store digital resources with metadata and provide browsing and searching facilities. Particularly of interest are archives which store scientific data rather than publications, for instance the Crystal Reports and Protein Data Bank archives[144, 145], as these might be usefully linked with an institutional repos-

itory. The data in archives such as these is rich and potentially forms interesting connections between other resources, for example by finding all the papers which discuss a particular protein; or crystals likely to be related due to their occurrence together in multiple publications.

4.6 Metadata

Though a repository might store publications, or tables of data, or multimedia as its core data type, repository software also deals with the management of the associated metadata. The mere storage and retrieval of documents would be served by a simple static Web page; therefore the specific purpose of collecting rich machine-readable metadata should be made explicit. Why is metadata important?

A useful description is given in [143]:

> Cultural heritage and information professionals such as museum registrars, library catalogers, and archival processors are increasingly applying the term *metadata* to the value-added information that they create to arrange, describe, track and otherwise enhance access to information objects.
>
> [...]
>
> In short, in an environment where a user can gain unmediated access to information objects over a network, metadata:
>
> - certifies the authenticity and degree of completeness of the content;
> - establishes and documents the context of the content;
> - identifies and exploits the structural relationships that exist between and within information objects;
> - provides a range of intellectual access points for an increasingly diverse range of users; and
> - provides some of the information an information professional might have provided in a physical reference or research setting.

MARC[146] is a family of standards defining the standard metadata schema used in libraries, including classification, physical, and authority records. Here standardization benefits users because bibliographic resources can be prepared

centrally and will be compatible with all library systems. The schema is detailed and includes provenance, physical description such as binding and dimensions, and multimedia information.

In a physical library such schemes are necessary to store works in an arbitrarily imposed logical arrangement: by author for fiction, by subject for non-fiction, and by title and chronologically for periodicals; as while a work can be classified in multiple places, it can only be physically stored in one. A logical ordering is essential when each work must be retrieved from a shelf, but it also serves to group potentially related information together, which is still important in a digital collection. To be able to browse a digital resource in the same fashion as a physical one, the same basic metadata is required. The summary page listing each work's metadata is the equivalent of the spine of a book; having found a document, it enables the reader to decide quickly whether it should be opened and read.

When replacing or supplementing a physical library, It is apparent that a digital library generally requires the same core metadata (aside from the purely physical aspects, which do not apply to digital resources). However, removing the physical constraints means that more diverse data can be stored, and organized arbitrarily; with computer-mediated access to the material, there is also the possibility of new kinds of organization, such as rating a work's popularity, or analyzing a reader's personal preferences or browsing history. Making metadata machine readable paves the way for these types of advanced processing and interaction.

In general metadata exists to improve access to the data it describes. It is therefore particularly important for academic publishing where the impact of research on others is the primary goal. Metadata can make information easier to find, evaluate, interpret, or discuss. Here are some examples:

- Metadata can facilitate better searching and browsing.

  In the paper medium publications are organized into a standard structure: journal, issue, article. For a researcher to discover an article of interest that is not cited or linked from an already known resource, they must find a journal covering an appropriate topic, choose an issue by date or title, and read the contents. It is difficult to follow any other structure, such as browsing by date before topic, as each journal would need to be searched separately.

Online, search engines are likely to be the primary mechanism for finding a copy of a particular article[147]. Metadata here works behind the scenes, providing data to the search engine ensuring an article appears in the list and is indexed with relevant keywords. Metadata is also important as a mechanism to find related material, through browsing or searching across particular fields in an alternative to the default hierarchical arrangement. Rich metadata also supports faceted browsing, a flexible interaction mode where the user can restrict their view of a dataset according to the values in metadata fields. This is the interface provided by mSpace[41] and Longwell[40].

- Metadata can support improved linking between resources.

  Citations as part of the document's metadata, rather than just the human-readable text, can be used for automatic citation linking, thus speeding up browsing and allowing the document to be understood more quickly. Web of Science[148] and MEDLINE[149] are abstracting services which provide both citation linking and the ability to search across multiple sources of data; MEDLINE also features a standard vocabulary, MeSH[150], which organizes the whole collection and aims to assist in classifying and exploring resources.

  OpenURLs[151] are a metadata-based linking mechanism. Using an appropriate proxy an OpenURL can be generated from a resource's metadata to link to a physical or digital copy appropriate to the user's context.

  In online social networking, tagging is a method of allowing third parties to assign metadata to a resource. By assigning a short, free text label to a piece of content defining its topic or attributes, it is classified and linked with others with the same assigned tag. Common in blogs and the del.icio.us social bookmarking site, Connotea[152] provides this service for academics by helping readers organize their citations and discover research tagged by others. Unlike typical Semantic Web systems, tags are generally free text with no formal taxonomy, and limited to the site in which they are used; their virtues are their accessibility and ease of use by nonexperts, and the way a community can evolve its own de facto "folksonomy".

- Metadata can provide new mechanisms for analysis.

  Citations and multiple authors create associations between people and papers which can be used to identify communities of practice. Citation tracking can discover the most significant publications and measure the

impact of a piece of research on the academic community. CiteBase[133] is an OAI-PMH driven service which performs citation analysis for repositories; it is hampered by having to parse references from documents as they are not yet commonly stored as separate metadata. The Digitometric Framework[53] builds on this data with advanced visualisation and hypertext features.

On the Web metadata can include tracking the downloads of a paper through Web server logging. This data, though prone to misinterpretation, can be analyzed to determine the popularity of a paper, identify the geographical location of readers (for example to provide translations in the most important languages), or perform bibliometrics. IRStats[153] is a package for EPrints which allows administrators to perform download analysis on their own repository. Brody et al. correlate download tracking alongside citation tracking to provide earlier estimates of impact[154].

4.7 Summary

It is my thesis that repositories are in an ideal position to benefit from the development of the Semantic Web and promote scientific communication. It promises a generic extensible data description mechanism, an identifier scheme, standard processing and reasoning systems, and universal interoperability. Repositories have the potential to be catalysts for the Semantic Web, as they already exist to preserve and disseminate data and contain the structure and formalism required. Incorporating Semantic Web interfaces into EPrints would by being interoperable extend the benefits of OAI-PMH towards the wider Web.

While initially focusing on metadata, Semantic Web integration with the content of academic works is also possible using the same principles. Rich markup languages, such as HTML and the XML based languages Open Document Format and Office Open XML, can have internal semantics which allow data to be extracted, analyzed, and linked by the repository. Standardization in this area could lead to a rich information-aware working environment which would be highly valuable to scientists. However, the remainder of this work will be limited in scope to considering document metadata, as incremental developments in this area can work with existing repository data and would not require authors to first change their working practises.

In the next section, problems which arise from bringing repository data to the Semantic Web will be explored.

5 Identifiers

This section describes the complex issue of using a name, or identifier, to refer to an entity. Already discussed from the point of view of hypertext and the Semantic Web, it will now be explored relative to repositories, identifying issues which arise from the characteristics of the data likely to be stored in repositories and the organizational and technical aspects of scientific publishing.

There are many entities in digital scientific publishing about which statements could be made on the Semantic Web, thus requiring an identifier. A paper, its author, their institution, a publisher, a journal, an issue; these are all named entities with their own metadata. Each subject in a classification hierarchy; these are entities in order to model the hierarchical structure. A particular representation of the paper, in HTML format, is a separate entity from the paper itself. The abstract concept of the HTML format—and the abstract concept of "file format"; the concepts of "paper", "journal", and the relationship "published in". For this information to be accessible to Semantic Web processes, each class and instance has a URI.

5.1 Naming

In hypertext systems links can be separate from the documents they appear in, and can refer to documents on different servers. The endpoints must therefore be referred to by identifiers. The same situation is present on a larger scale in the Semantic Web, where identifiers are required to describe abstract or physical as well as digital entities, and are an integral part of the linked data graph. The problem of naming can be subdivided into several related problems, which depending on the type of entity may be between straightforward and impossible to solve completely:

- Assigning an identifier to a new entity. An identifier is necessary in order to make statements about a resource, and needs to uniquely identify the resource to be able to describe it unambiguously.
- Resolving an identifier. Many entities have a meaningful Web presence, and mapping them to a URL with their content or more metadata would be useful. However, identifiers can also refer to 'non-information resources', abstract entities or things which only exist in the physical world:

> Other things, such as cars and dogs (and, if you've printed this document on physical sheets of paper, the artifact that you are holding in your hand), are resources too.[155]

Resolving an identifier of this type could still return useful data, but it cannot entirely capture the "essential character of the resource".

- Looking up the identifier which already exists for an entity; in other words, finding out what entity you have. This could be a problem of searching the Semantic Web, using a resolution service, or using the same algorithm to regenerate the same identifier independently, depending on context.

In the world of traditional offline publishing the identifier for a publication, which allows it to be looked up, is based on its metadata: the reference information in a bibliography entry. This reference is not opaque; the details allow the reader to find the cited material but also give some context, helping the reader decide whether or not to follow the citation link. The title of course reveals whether it is relevant to the reader, while knowing the authors and publication name provide some degree of provenance. Though each component of a reference may not uniquely identify its referent, as a whole it uniquely identifies a publication successfully and provides useful data to a human reader.

The title and author do not always uniquely identify a publication; for instance where an edited version of the same work appears with errors corrected. The truly unique metadata (excepting misprints or invalid assignments) is a book's ISBN, or the issue and page range assigned implicitly by the journal. Including these attributes in the bibilography entry also let the citation be resolved, by finding a paper copy in a library or now perhaps through the journal website. When making a new link the title page of an article provides disambiguation, by including all the information required to cite it; and for lookup the bibliography provides details of citation links whenever a paper refers to another. Given that the information in citations can be reliably interpreted by the reader, the body of published academic papers form a hypertext or linked data structure which is navigated via the citation links.

The Semantic Web aims to allow computers to interpret data formerly accessible just to human readers. To move therefore from a citation as an identifier, which is resolved manually and physically but encapsulates information about the resource, to an opaque digital identifier which cannot be interpreted, requires explicitly storing the data which was previously implicit. Mapping the string of characters, "Communications of the ACM", to its physical incarnation as a

printed journal is easy for a human, but a computer can only reliably make this leap if a link is made between the string and an abstract object representing the journal. To take the next step and allow the computer to retrieve the cited paper needs further annotation, linking the object to a Web address. In the true vision of the Semantic Web, for an agent to analyse the citation and evaluate whether it might be of interest, based on the topic, reputation of the authors, and recommendations of other researchers, would depend on a huge structure of cross-linked and authenticated data. For this data to be interpreted usefully and reliably, unambiguous identifiers could be required for the paper, its authors, the journal, other papers which are linked by citations—and other concepts which are fuzzy or difficult to define, such as the topics covered and the user's areas of interest.

Although a human may use their judgement and experience to understand each part of the citation, there is potential for ambiguity when interpreted algorithmically. Although citations often use standardised formats, these may differ between journal styles. There is also lack of formality, meaning that even a correctly formatted citation may be misinterpreted. In this example: "Smith, A. B. An Experiment. Journal of Science." a human would understand that the author is 'Smith, A. B.' and the title is 'An Experiment', but it could be 'Smith, A.' and 'B. An Experiment.'—which is judged to be unlikely by a human, but is equally valid for a computer. These are rare examples, but imply that data obtained from automatic citation linking is possibly incomplete, and is more likely to be so in a larger corpus. This shows that formally specified, unambiguous digital identifiers are vital to the Semantic Web, and particularly for scientific communication[156].

5.1.1 Digital bibliographic identifiers

ISBN and ISSN are examples of such unambiguous identifiers. They are ISO standardized systems for assigning unique numbers to manifestations[157] of books and periodicals respectively; they are persistent, and managed by international and national organizations to ensure they are globally unique. They are however designed for physical printed resources; a more flexible system is needed to manage digital identifiers; one commonly used by journals and repositories is the DOI system.

A Digital Object Identifier (DOI)[156] is a unique and permanent identifier for an electronic document. The scheme forms part of the Handle System[140], a

general purpose identification and resolution system for all kinds of digital data. DOIs are short strings with a prefix denoting the assigning organization, such as a publisher, and a suffix in an arbitrary format chosen by the organization, thus: `doi:10.1000/182` refers to the DOI Handbook. The system provides a centralized addressing system which guarantees that an identifier is unique, and a resolution service to obtain metadata or a copy of the document. This level of indirection makes a short string semantically equivalent to a complete citation, with the added benefits of removing potential ambiguity and resolving to a digital resource. Using this flexible format allows DOIs to be issued based on existing naming schemes, for instance ISBN and ISSN codes.

DOIs and the Handle System infrastructure predate the Semantic Web's use of URIs as identifiers, but share common goals. Both URIs and DOIs potentially identify the intellectual content of the resource and not the Web address at which it is located. Treating identifier and target address separately allows identifiers in both systems to resolve to metadata as well as an electronic copy of the resource; meaning more than one copy of the document can be referenced, in different contexts or with alternative expressions of the same content, and means identifiers can be given to non-information resources: abstract objects which cannot be said to have a definitive retrievable digital manifestation. While this reinforces the logical separation of distinct concepts, it is an advantage of the Semantic Web that an identifier may be resolveable directly in circumstances where this is appropriate. While DOIs are intended to be resolveable it relies on client support or rewriting the DOI to use an HTTP proxy, e.g.: `http://dx.doi.org/10.1000/182`; the URI based scheme is intended to take advantage of the preexistence of Web infrastructure. On the other hand, the separation of the DOI infrastructure from URLs is guaranteed to decouple the technical constraints of how resources are stored and addressed from the organization of their identifiers[158].

A key feature of the DOI resolution infrastructure is that it is centralized: new identifiers are assigned and managed by a small set of organizations. For a restricted domain like publishing, a central naming authority is an efficient way to ensure uniqueness of names and ensure they can always be resolved. The infrastructure required to support this means that assigning a DOI carries a charge (resolution is free). This is also true of ISBNs. Being centralized also ensures that DOIs are only assigned to documents, not other data types; handles in general can be assigned to other types, but are still centralized.

OpenURLs are an alternative mechanism to provide local resolution for publications[151, 159]. With an awareness of the user's context an OpenURL resolver can link to subscription or printed resources as well as publicly available material. An OpenURL consists of the URL of a resolver service, followed by parameters describing the metadata of the target. The same parameters can be given to a different resolver to potentially obtain an alternative local copy. Traditional metadata such as title and author can identify a resource, or the DOI system can operate in synergy with OpenURL, with the DOI unambiguously defining the target and the resolver finding a copy in context. The service is designed for the scholarly library scenario where paid subscriptions and access to physical copies restrict the availability of material.

OpenURLs are a complementary technology to persistent identifiers. While an identifier service maintains a link between the identifier string and the abstract work (and also potentially metadata, and a way of obtaining a digital copy), an OpenURL resolver goes from the abstract work to a physical or digital copy in an appropriate way for each individual user. The source of the link is required to use an unspecified mechanism to discover which OpenURL service would be appropriate for that user; for example IP address matching or a saved user preference.

Because an OpenURL may be different for each user, and represents a query rather than a particular resource, it is unsuitable for use as an identifier. The benefits of OpenURL are also less important for Open Access research, where resolving to a single digital resource for all users is acceptable. The relationship between OpenURLs and the Semantic Web in the future may therefore be a purely technical one; a local, contextual retrieval service could be made simpler by using RDF for data exchange, which could reduce the need for integration with the remote site.

5.1.2 The semantics of an identifier

An important characteristic of an identifier, or a component of one, is whether it conveys meaning or simply allows different things to be distinguished. Outside the digital realm identifiers are generally of the meaningful type: people talk about books by their title, not their ISBN, because the embedded information makes it more familiar and memorable and provides useful context. However, this need not hold true for digital identifiers; a computer can distinguish equally well between identifiers in any format, but cannot reliably extract any implicit

meaning. A principle of the Semantic Web is that formal specification of metadata is more reliable than informal methods of information extraction, which introduce the possibility of errors in interpretation. Accordingly any information implied by an identifier should also be asserted explicitly.

This does not preclude URIs having a format which can assist people, or which can be generated in a logical way based on related data; but for the system to make use of this knowledge it must be asserted explicitly as a property. This also introduces the possibility that if the data changes, the explicit properties of the resource may diverge from the implied semantics of the URI and render it misleading. This holds true particularly for the types of objects which might exist in repositories, but is less important for universal concepts which are unlikely to change, or types which are defined solely by their properties. Examples of these are units of measurement, substances, theoretical data structure definitions, or well established standards.

Another consideration is the need for technical concerns to dictate components of an identifier. These concerns are likely to be the reason an identifier is forced to change, and a justification for persistent naming schemes.

Outside the internet domain, telephone numbering in Britain is an example of these characteristics of identifiers. A complete number (eg. 01234 567890) consists of an area code (01234) and local number (567890). The local portion has no intrinsic meaning. The area code corresponds to the geographical location of the customer, which could also be used to route calls within the network. Separating the area code benefits the user as they can dial fewer digits for calls within their area, and may simplify call routing; however, it means a person's phone number may not be permanently fixed.

- Relying on the prefix to route calls to improve efficiency means that a customer cannot move and keep their number. It also means the number can be traced back to its location, which may not always be desirable. To get around this there are special prefixes with no geographical meaning, available at extra cost, which provide redirection. This is analogous to an indirect addressing system which proxies the request to conceal the true URL of a resource.

- Technical issues occasionally require large scale renumbering to take place. This occurs locally when a prefix runs out of available numbers, or nationally in the case of a shortage of prefixes. This corresponds to a Web site reorganizing its structure or changing its host name.

Were telephone numbers simply assigned serially, with no prefix or geographical addressing, there would be no need to change identifiers once they were assigned. The disadvantages are that people can no longer use the meaningful parts of the number as an aid to memory, and that the network must provide a centralized system for looking up an identifier and routing the call.

5.1.3 Naming authorities

ISBNs, DOIs, telephone numbers, and DNS share the common characteristic of a naming authority. Each system guarantees that every resource in its scope can be given a globally unique identifier, by centrally verifying that the identifier is available. A naming authority therefore allows entities without an existing unique key to be assigned one. Publications play the same role for traditional citations; while there could theoretically be two papers with the same title, by identically named authors, the journal name and page reference can be used to distinguish them. Authorities exist for many different data types and are commonly used by libraries; for example the US Library of Congress maintains an authority on preferred names for people, organizations, conferences, and geographical features[160].

Central coordination of a namespace helps ensure that it is organized coherently. Where it is possible to standardize or mandate an authority this can be more efficient for users. Another example of where centralised control is beneficial is defining a subject hierarchy. It is valuable for users to be able to browse by subject to find new material in their area of interest, and in a distributed environment sharing a common set of subjects helps the user find things more quickly as well as making classification of new items easier. Therefore standard subject hierarchies exist in traditional publishing, developed by expert authorities and only updated where necessary; these can be usefully extended to the digital world. The Dewey decimal system[161], the Library of Congress subject hierarchy[162], and the ACM Computing Classification System[163] are examples of such hierarchies.

The problem with naming authorities becomes evident when considering a distributed, decentralized system like the Web, where the process of publishing is simply adding a document to a server. Where an authority is desirable it cannot be mandated, because there is no way to enforce central control over Web content, but must be the result of a standardization process or community accepted decision. As this lack of central control is inherent and fundamental to

the Web, there will also be some cases in which a naming authority could not be established. The scope of the Semantic Web extends to assigning metadata to every imaginable resource, not least every Web-accessible document, without requiring the intervention of the resource's creator.

The Semantic Web uses URIs as a mechanism for distributing authority over identifiers. If a site only assigns identifiers which include its own domain name, it is unlikely to accidentally collide with another site's assignment. This avoids the greater problem of two different resources sharing one identifier, by permitting the lesser problem of one resource with multiple identifiers. It is a lesser problem because additional knowledge can assert that the two identifiers refer to the same resource. Unification of identifiers will be considered later.

The ultimate reason to choose URI-based identifier schemes is a pragmatic one. For the Semantic Web to be accessible to all Web users, the only guaranteed infrastructure it can rely on is that of the Web itself. Alternative identifier schemes can be incorporated into the URI syntax, with an organization providing guarantees that the appropriate functionality will remain, thus supporting the semantics of any other scheme. Sites can therefore maintain persistence of resources they publish, and their identifiers, using whatever scheme is appropriate to them.

Proponents of systems such as DOI/Handle System and ARK[164]—a similar scheme with additional emphasis on explicit preservation commitment—argue that URIs are insufficient to provide long term preservation and reliability of identifiers, because of the coupling between the identifier and the technical means to resolve it[156, 164]. However, it is not evident that Semantic Web technology cannot potentially provide the same guarantees while remaining more flexible. more flexible while potentially providing the same guarantees.

- Handle resolution requires a centralized paid for service. An annual fee is charged to be allocated a Handle System prefix; alternatively DOIs incur a one off charge on creation. While the Web infrastructure of DNS and HTTP is not free, using the same infrastructure for the Semantic Web means there is no extra charge.

  For an academic institution's own research output these charges are easily affordable, but for other scenarios (such as a personal collection of photos, or to catalogue a large existing collection of third party work) they may be a significant portion of the total cost. Therefore as a general mechanism for identifying all types of Semantic Web data, URIs are more appropriate.

- Users concerned with long term preservation, who have the foresight to use an indirect addressing scheme such as the Handle System, could equally have used a logical URI scheme or redirection service. To use EPrints as an example, the familiar URLs assigned to eprints are simply the host name of the archive, plus a serial number. Administrators are also encouraged to use a generic host name (e.g. `eprints.ecs.soton.ac.uk`) rather than the computer's real host name.

  This scheme still means that the URI contains the organization name. This presents two problems: the organization changing name, and the merging of two organizations.

  The Handle System resolves the first problem by using a meaningless, numeric identifier for an organization. This solution is unnecessary for the Semantic Web in general; in most cases identifiers will be hidden from users making their textual representation immaterial, and an organization can choose to use a URI-based redirection service such as `purl.org`, or host their own equivalent service, if required.

  The reorganization of handles has the same difficulties present when URIs, or indeed physical resources, are reorganized. For example, several university departments might set up repositories for their research output, and describe them using URIs and RDF. Each repository is hosted on a server in each department. Subsequently, the university sets up its own repository and wishes to migrate all records to the central location. In EPrints there is no way each record could keep its original serial number in the new archive, as these would conflict; the solution would be to assign all records new numbers, and redirect their URIs from the original ones using standard HTTP redirection. New records would receive new numbers in the centralized scheme. In the Handle System, identifiers are already indirect, so the resolver would be reconfigured to point to the new locations; but the batch remapping of handles is equally complex as the batch remapping of URIs, except that the configuration takes place on the handle server. In DSpace there is a hybrid approach[165]: both the Handle System server and each DSpace installation need to be reconfigured in the event of server consolidation.

  This is therefore not an inherent advantage of handles over URIs, but one of anticipating the need for change and preemptively using indirect references. However, the Semantic Web can also use HTTP redirection and RDF statements to remap identifiers after they are created, which

is essential in the cases where there is no universally accepted central authority to assign an official identifier (personal names are an example). Bearing in mind this point it is valuable to coin URIs which are structured in a way to make remapping easier.

- Handles represent a commitment to preservation, at least in that the identifier will only ever refer to one resource. This is however an organizational practice and cannot be enforced technically. A university could make the same commitment about its URIs; however, it has the additional possibility of explicitly and formally stating its preservation policy in RDF. The institution is free to make the same commitments about its resources as users of ARKs and handles can, or make a contract with a third party to provide that guarantee, as the Handle System does.

- An argument put forward for handles[156] is that in future a handle can resolve to a different kind of resource, while a URI is constrained to the Web. For long term commitment to resolution, the system must consider that in future the Web may be obsolete.

  I argue that URIs will be just as suitable for identifying future non-Web resources as handles are for Web resources. Currently a handle may be resolved through the Web to a URI, but can point to any target. Handles can be encoded into URLs using a proxy. This situation is exactly analogous to a future non-Web system, which could use a proxy to resolve URIs to their replacement, treating them opaquely as handles are treated on the Web.

  This point leads to the conclusion of the argument: if a repository administrator wishes to use the Handle System, they can use their handles as URIs via a proxy or URL scheme and still interoperate seamlessly with the rest of the Semantic Web. Using URIs as identifiers permits this kind of flexibility while lowering the barrier to participation on the Semantic Web in other scenarios.

In conclusion, URIs are adequately flexible to be used as identifiers for concepts in the Semantic Web, including non-information resources. Where features beyond the basic standard are desirable, such as long term preservation commitments and indirect resolution which are advantageous for repositories, these can be supported by standard HTTP features and documented formally in RDF.

5.2 Identifiers in the Semantic Web

Given that URIs are to be used to identify resources, what are their characteristics? How are they assigned, obtained, and resolved?

5.2.1 Assignment and coreference

There are two ways of dealing with ownership of identifiers. Either clearly define an authority to issue identifiers in a particular situation, agreed by all parties; or allow anyone to coin identifiers (within their own namespace), and support unifying them. As some situations will preclude the former, notably identifiers for people, the latter must be supported, and implementations must have a mechanism for handling coreference: when multiple identifiers refer to the same entity and need to be unified.

The Semantic Web as a whole is built on the same protocols and architecture as the current Web, and will therefore inherit its anarchic and decentralized characteristics. It is possible to impose local order, but overall any Semantic Web system must deal with errors, data being moved or deleted, multiple copies of the same resource in different places, and untrustworthy data. Acceptance of this state is, with hindsight, a strength of the Web as it encouraged universal participation and contributed to its success[23]. For an application of Semantic Web technology to be successful it must also tolerate these problems rather than attempting to prevent them.

Any attempt to automatically unify identifiers, besides potentially introducing errors in the data, also precludes the possibility that multiple identifiers are desirable. A person may wish, for example, to keep their personal and professional identities separate, and the decentralized ownership of identifiers supports this. However it is advantageous to avoid identifier proliferation when it is unnecessary. Universally using the same identifier means that when related data sources are combined, connections exist automatically between the data from each. For example, if each journal publisher made metadata available in RDF about the publications they produce, and each author had a reliable unique identifier, a single query on a combined data set could return all of a person's publications or their frequent collaborators independent of where the data was originally obtained.

Within RDF it is possible to unify identifiers in a standard way. The OWL ontology includes the `owl:sameAs` predicate, which indicates that the two URIs

```
<rdf:RDF
    xmlns:coref="http://www.resist.ecs.soton.ac.uk/ontology/coref#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">

    <coref:Bundle rdf:about="http://example.org/#bundle1">
        <coref:hasEquivalentReference rdf:resource="http://example.org/id/1" />
        <coref:hasEquivalentReference rdf:resource="http://example.org/id/2" />
        <coref:hasEquivalentReference rdf:resource="http://example.org/id/3" />

        <coref:hasCanonicalReference rdf:resource="http://example.org/id/1" />
    </coref:Bundle>
</rdf:RDF>
```

Figure 4: A bundle of coreferents expressed using the structure in [168]

refer to the same resource, and are totally equivalent and interchangeable. This assertion can be too strong in cases where meaning can change, and can impose a burden on Semantic Web systems making use of the URIs. Booth[166] and Jaffri et al.[167] argue that a less strict contextual assertion of functional equivalence may be preferred in some situations. Jaffri et al. present a Consistent Reference Service[168] which models equivalence using a bundle, a collection which contains each coreferent as a member and may define a canonical URI (Figure 4). This mechanism allows applications to explicitly resolve coreference when required, and also uses fewer triples at the cost of an extra step in queries.

Semantic Web standards mandate that a URI refers to only one entity[155], but this leaves the problem of more than one URI being assigned to each entity. Less obviously, something which is considered a single entity in some cases may be subdivided in others, for instance a book made up of chapters, meaning that there may be coreference between a single URI and a group of URIs considered together.

Open Access repositories are likely to lead to a proliferation of identifiers for the same work. If each of the authors submit their work to an institutional repository as well as a journal, each record will be independent, with a different identifier and possibly also conflicting metadata and variations in full text. For example, each institution might categorize a paper within a research group, so collaborating authors would put their own group when adding the record.

Identifying coreferent identifiers is an information management problem which has organizational as well as technical aspects. How can the fact that two entities are coreferent be detected, and how can systems make use of this fact to improve the quality of data? What if two identifiers are erroneously or maliciously declared coreferent? The application of Semantic Web technology to digital libraries is a particularly appropriate scenario in which to attempt to resolve these issues:

- Firstly, coreference is likely to occur frequently in repositories. Collaboration between researchers in different institutions is common, thus publications stored in an institutional repository are likely to involve people from other institutions whose metadata may not be available. However, the same paper would be stored in each researcher's local archive, and also potentially a journal or conference repository, each of which may include different metadata or files.
- Handling coreference would provide a valuable service for repository users. A simple example is linking each author to their list of other publications, where managing coreference would allow a more complete list of related publications by merging data from multiple archives.
- The purpose of a digital library is to collect and disseminate data efficiently. It is therefore likely that users and administrators will consider the effort of collecting high quality data and using coreference tools worthwhile as it raises the profile of their research by making it more accessible.
- The scope of potential coreference is limited. Digital libraries are already managed at the institution level and are likely to have an administration infrastructure in place. Institutions can make use of their user account, staff, and organization structure systems to assign URIs which are unique within their namespace.

Resolving coreference might be a purely manual process in simple cases. Fully automatic processing, while desirable, will always run the risk of false deductions causing bad data. Therefore coreference resolution systems which take a heuristic approach are of most interest: where a tool searches for likely matches, but the user is given overall control. More advanced tools may be suitable for archive maintainers, where an an adminstrator can ensure data integrity; ad hoc resolution for end user orientated Semantic Web browsing applications may benefit from a simpler interface with more automatic deductions, potentially at the expense of data integrity.

Alani et al[169] and Lewy et al[170] discuss this topic further, proposing technical methods for resolving the problem of coreference with partially automatic and assistive user interfaces. For instance, they demonstrate a Community of Practice analysis tool, Ontocopi[103], as a heuristic detector of coreferent identifiers[169].

In my work I take a similar user interface centric approach, but attempt to avoid the problem in the first instance by encouraging the user to reuse an existing identifier. In the distributed environment of the Web this will not prevent coreference, but it will make it more manageable by reducing the proliferation of identifiers, reducing the size of the data set over which assisted techniques such as the above can be applied.

5.2.2 Resolving Semantic Web identifiers

A strength of using URIs as identifiers is that where appropriate, they can directly resolve to a document. However, this process is made more complex in the Semantic Web because of the different requirements of human and automatic browsers.

HTTP content negotiation[62] is a mechanism to handle resolving identifiers for humans and Semantic Web agents separately. When a Web user visits a page in their browser, part of the HTTP request made is a list of MIME types the browser prefers. While a typical browser might include `text/html`, `text/plain`, and `image/*` (all images), a Semantic Web application can request an `application/rdf+xml` response and be given linked RDF data instead of HTML. This is the standard application of content negotiation, and is suitable for an information resource which can be accurately represented in either RDF or another format. As these resources have the same URI they cannot have separate metadata, and so must have the same "information content"[171].

Before the Semantic Web, a URL was the identifier for a Web page. Now URIs can refer to abstract concepts which cannot simply be resolved to a digital form. If a person is assigned a URI it refers to them in an abstract way, which allows statements to be made such as "Harry Mason attends the University of Southampton". If this identifier were to resolve to their home page, a possibly useful action for human browsers, it must not thereby imply that the person and their home page are the same entity, which would lead to ambiguous metadata.

To avoid this confusion, it is now recommended[172] to make non-information URIs resolve through HTTP redirection. The ECS URI system demonstrates this technique[173]. Each ECS member has a unique integer ID, which is used to generate a URI. On the website as well as a page showing their metadata (such as publications and contact details) there is a service presenting the same information in RDF. Resolving a person's URI does not directly return either the HTML or RDF version, which are documents with their own URLs; it returns an HTTP 303 redirect code, pointing to whichever version the browser requested through content negotiation. The intended interpretation of this response is that the server cannot return a non-digital resource, but it can return a related resource which describes it. If abstract resources are always resolved indirectly, it disambiguates statements made about them from statements about the document returned.

An unsolved problem with content negotiation is that resources can themselves be RDF, which means there is no way to directly request metadata about the resource as it would have the same type as the resource itself. The problem can be mitigated by linking the RDF to its metadata using `seeAlso`. There is also an issue with returning indirect responses, which according to the HTTP specification should not be cached and might lead to many requests being made for the same content[171].

Features of HTML and RDF can also be applied to obtain the identifiers of related resources. This addresses the problem of distinguishing an abstract resource, such as a paper, from a manifestation of it[157]: a Web page describing the paper, a digital rendering such as a PDF, or a physical copy. In HTML, the `<link>` element specifies a relationship between two documents, and can refer to alternative renderings of the same content, or link to arbitrary related resources. It is also possible to make a compound XML document which contains both XHTML and RDF using namespaces. In RDF the RDFS predicate `seeAlso` can provide the URL of another RDF document with more information. Once RDF is obtained it can describe the original resource more precisely and its relationship with the abstract one.

The Tabulator browser[39] automatically follows these types of links from HTML and RDF, as could software agents, so linking related documents in this way improves user experience as well as resolving ambiguity. It also provides an 'entry point' into the Semantic Web from an ordinary Web page, which removes the need for users to work directly with URIs while still allowing metadata to

be unambiguous. This demonstrates how the existing human-readable Web will integrate with the Semantic Web.

5.3 Identifiers in EPrints

For the objects stored in EPrints to be linked and described with Semantic Web standards, URIs must be assigned to the objects which exist in the archive. Simply assigning an identifier to an eprint is however a more complex task than it seems.

Multiple identifiers, differing in scope, may be usefully assigned to the same object to provide scope to the metadata which uses the identifier. Examples of identifier scopes are:

- An abstract "work"—the intellectual content of a publication. Two records are considered the same here if they would be recognized as having the same content by a person. The same identifier could therefore be applied to:
  - The same content in a different format, for example a conversion of HTML to PDF. This could even include a spoken version in an audio format.
  - A reformatted version, with different style but the same meaning, for instance to conform to publisher guidelines.
  - A translation of the work into another language. The identifier could then resolve differently based on the user's preferred language.
  - The same work in another repository, which might have different metadata. Resolving this identifier appropriately might depend on the relevance or accessibility of each repository to the user.
  - A revised version with corrections. Depending on the nature of the corrections and the user's intentions it may be right or wrong to treat this as the same work.
  - A physical printed copy, if there exists some way to provide it with digital metadata.

  Such a broad scoped identifier could be used in a citation, to resolve to the most suitable expression of the work depending on the reader's circumstances. However, it is clear that to reliably resolve such an identifier

is both a technical and organizational problem. Hypothetically publishers could create a framework for their works, linking alternative versions together with a common identifier and providing a resolution service resembling OpenURL. EPrints allows submitters to link related versions but does not assign a common identifier.

- A particular version of a publication, with guaranteed identical content from the reader's point of view. This scope would be used to annotate or transclude specified regions. It would therefore include alternative file formats or cosmetic formatting, but exclude textual changes.
- A document, which may be a component of an eprint. EPrints allows each record to contain multiple documents, and each document to contain multiple files. This feature is used for providing multiple file formats as alternatives, or supporting data to accompany a paper as an appendix. This identifier might be used to describe the relationship between the publication and its components.
- A specific file, forming all or part of a document. This class of identifier could be based on a hash of the file's contents. Metadata in this scope would describe the file format, for instance the MIME type or software required to view the file.

The FRBR model[157] formalizes this hierarchy, and other types such as events and organizations.

These scopes only consider simple kinds of records—traditional publications. Digital libraries can store data about anything worth archiving, and some special types might need additional classes of identifier. These examples demonstrate quirks which will need to be dealt with to accurately represent knowledge of these types on the Semantic Web:

- Though the idea of hypermedia has existed for some years, publications written as hypertext are unusual. In the academic world this may be due to the traditional model of paper publishing with its familiarity and convenience for authors and readers. Hyperfiction is more common, with "choose your own adventure" stories being a good example. In any case, a digital library including hypertexts may need to support resolving node identifiers, for instance to allow citations to refer to parts of the hypertext.

  Another example is educational material using adaptive hypermedia to provide alternative levels of detail for students with different require-

ments.A paper citing an adaptive hypertext might need to identify the appropriate context, in order to refer to a portion of text visible only at particular level of detail.

An EPrints archive, as a Web based system, would require a hypertext to be represented in a Web accessible form. Given this constraint the features provided by the URL and HTTP standards may be sufficient to identify parts of hypertexts.

- Related to the issue of storing hypermedia is support for assigning metadata to parts of a document. This differs from handling hypermedia in that the metadata might describe any region of the document, and these regions can overlap. EPrints includes, by default, a module which supports transclusions, to support Nelson's ideas of composite documents and copyright control[8]; this allows the retrieval of parts of files and provides an interface to select the desired region. If such a feature were widely used it could become important to associate metadata with a transclusion.

  Transclusions are processed dynamically by EPrints; no state is held on the server and therefore the URL contains all the information required to return the selected data. A consequence of this is that transclusions already have URIs which could be used to assign metadata. However, the metadata relates only to that specific region; in some circumstances a subset of the metadata would ideally inherit the metadata of its parent—if that were clearly defined. Managing this is clearly a complex and error prone operation and will not be considered further.

- Artistic works, such as paintings, crafts, or sculpture, could be digitized and stored in an archive for preservation as well as to permit electronic access. The Kultur project[174] is a recent research project which intends to explore the issues in creating such a multimedia repository. While a paper can be archived in its original form, records describing these works can only include a representation of the true three dimensional form.

  The distinction between a resource and its representation is crucial. If a sculpture is represented by a series of photographs, the "creator" metadata field might name the artist; however, the creator of the photographs, which might be considered works of art in their own right, is the photographer.

  Other art forms have similar problems. Another example is a film of a concert; who should be listed as the "creator"—the maker of the recording, or the performer, or the songwriter? How can metadata about the concert

be distinguished from metadata about the film? The same issue arises with any composite type, such as a book containing articles by different authors.

For example, to accurately hold metadata about music, a solution would be to add a special field linking two types of record—a composition and a performance—and configure the user interface to display this connection. However, this may increase the complexity of data entry and expose the user to the details of the structure; it also introduces issues of shared record ownership which could compromise data integrity.

EPrints can express any of the relationships defined by the Library of Congress in [175]. This allows the expression of multiple different "creator" fields but leaves no way of linking two performances of a composition. This can be considered an instance of metadata which refers to an entity; the topic is discussed below.

Pragmatically, using different identifiers whenever the coreference of two records is in doubt is the best solution. In EPrints, a user wishing to submit an updated version of a record creates a new record based on the existing one. The original record therefore remains unchanged once it is added to the public view of the archive. The new record has a new identifier, but a tree showing the history of the records is shown, and a message is displayed beside any older versions when a new version is submitted. In the Semantic Web, a statement can be made describing the relationship between the resources, so the new one can replace the old in appropriate circumstances. If the same identifier had been used, the resources would be defined as equivalent and indistinguishable in all circumstances.

5.3.1 Entities as metadata

The above list only includes identifiers for the core records in an eprint archive. The metadata describing an eprint often refers to an entity rather than a value: authors, publishers, events, and citations are all examples, each with comparable variations in interpretation. For instance, a person may have more than one role, and their properties may change over time; a conference series is an aggregation of related conferences, workshops, and presentations.

The fields which refer to these external entities, like bibliographic records, still store simple strings. If instead a field contained a URI, unambiguously defining

the entity it refers to, the record becomes part of the global linked graph of the Semantic Web. It is my thesis that capturing identifiers alongside text strings in repositories would improve the quality of captured data and support new possibilities in the user interface.

One possibility for enhancing the interface is that URIs could be used to make links between occurrences of the identifier on different Web sites. This could be done by the repository itself or by external tools. An author could be linked to their home page or a list of all their publications in all Web-accessible archives. A conference could form part of an aggregate conference series, connected and explored as a single entity. Research projects could be integrated across each participating institution's repository to be viewed as a whole.

Data quality is improved by encouraging submission of more accurate, consistent, and complete data. By presenting a submitter with a list of valid choices which match their entry, they can save time by selecting the correct one, thus also avoiding typing errors. This method also allows a common spelling to be used, which is useful for personal names as well as long values such as the names of conferences or journals. It could also fill in related fields, such as event dates and location, making data consistent between records. The benefit of using a URI for the event instead of filling in fields based on existing records in the archive (already a feature of EPrints), is that this data can be preloaded before any records exist; possibly by being obtained automatically from the conference organizer to ensure consistency across all repositories.

When an author submits a record to their institutional repository, they have a mental model of the work they are submitting and its metadata. The other authors, their institution, the publisher, and the subject of the material are entities in the real world about which the author is familiar. In their mind there is no ambiguity between two acquaintances who share the same name. Inputting data into a repository is capturing this knowledge in a form which can be manipulated by the computer; but by collecting references to people by name, the system is in effect discarding the submitter's implicit knowledge which could disambiguate them.

Recording identifiers for the entities in metadata would capture this knowledge as well as providing conveniences in the user interface. In effect, the submitter can say that an author is "this person", not just "a person called X".

5.4 Summary

This chapter has analyzed the use of digital identifiers to refer to the entities which relate to institutional repositories, with the following conclusions:

- An identifier allows unambiguous statements to be made about a resource in the Semantic Web.
- URIs are a suitable identifier scheme for the digital, physical, and abstract resources which exist around a repository. A key benefit is that they can be resolved through standard Web protocols to lead to useful data.
- Organizations who have authority over an entity can produce a canonical identifier, simplifying the process of assigning metadata to that URI.
- The potential for coreference will always exist in a distributed environment. However, there are methods of minimizing it and correcting it after the data is produced.
- Using standard formats and common identifiers makes data interoperable, which promotes added value services. This is particularly significant for repositories where the goal is widespread dissemination of data.

In the next chapter, I will explore the practicalities of applying these conclusions to real repository data.

6 Analysis of repository data

This chapter describes the characterists of data from three repositories: RAEPrints, WWWConf, and ECS EPrints. The first contained records from the RAE and explored issues with large data sets and poor quality metadata. The second aggregated proceedings from the World Wide Web Conference with rich metadata and full text, and demonstrated the improved functionality available with better quality data. The third is the repository actively used by the ECS department of the University of Southampton for its research output. I constructed RAEPrints and WWWConf to perform this analysis.

This investigation demonstrates the problems which can occur in real world repository datasets, and how the mechanisms discussed previously can be applied to resolve them. This leads to conclusions about how the data structure and user interface of a repository could be modified to make better use of the available data, increasing its value as scientific communication.

6.1 RAEPrints

RAEPrints was built from the 2001 UK Research Assessment Exercise data, made available in simple database or CSV formats on the HERO academic portal website[176]. The exercise includes gathering information about up to four publications from each researcher in UK institutions and is used to assess the quality of their research[177]. I used it for an initial investigation into the EPrints software, and it provided an example of the problems which are caused by the lack of formalism and poorly structured metadata.

The RAE data consists of several datasets. The primary set for this archive contains publications for each author, grouped by institution and Unit of Assessment (a broad topic classification). Metadata available is basic, including authors, title, year, publication information, and research group. The formatting of these fields was very variable because of the distributed collection process and informal specification of fields. Units of Assessment, people, research groups, institutions, and additional annotations are included in separate datasets. Some of these were used in the RAEPrints to produce structure and metadata; others were too loosely structured to extract reliable information, or only relevant to the RAE examiners.

This archive held 106,401 records, more than all other repositores registered at eprints.org combined at the time. It was in fact so large that the default

EPrints directory structure had to be modified to avoid hitting a limit on the number of files or directories on the filesystem. No full text was included in the data set; an initial attempt to automatically obtain full text via Web searching was excessively slow, due to the size of the data set, and unreliable, due to the likelihood of multiple documents being returned by the search. This problem is partially solved by systems such as CiteBase[133], which indexes accessible repositories as well as falling back on a general Web search, but the scope of RAEPrints was too large to complete this process in the time available. The archive was built to analyze the metadata and investigate the problems of coreference, data formatting, and aggregation; examining full text was beyond this scope.

Before it could be imported the data first had to be processed, analysed, and converted to the standard XML import format used by EPrints. Several quality issues were identified while importing, relating to data formatting, international characters, and handling subjects.

EPrints requires personal names, such as authors and editors, to be split into their component parts: honourific, "Dr."; given names, "Martin Luther"; family name, "King"; lineage, "Jr.". This is required for proper sorting and for exchanging data with other services. Names in the RAE data were provided as simple strings, unfortunately in a great variety of different formats, even varying at times within the same institution; a significant challenge was reliably identifying these components automatically. Without any unique identifiers for individuals it was naively assumed at this stage that identically formatted names within an institution represented the same person. A surprising 34,220 (32%) of records had no author included.

The simple way author names were processed will inevitably have caused an unknown number of false interpretations. After attempting to eliminate variations in formatting, 76,484 unique author names were listed. Of these 13,074 (17%) occurred in more than one institution; 9,075 in two institutions, 2,268 in three. Five names, "J Wang", "D Smith", "S Jones", "R Smith", and "P Smith", each occurred in more than 20 different institutions, which no doubt means that more than one researcher shares these common names. However, it is impossible to judge how many of these cases are because a researcher has worked with multiple institutions, and how many are two people sharing the same name. Particularly with these popular names, it is also possible that two people at the same institution will share a name, which would be undetected.

On the other hand, one individual's name can have variations in formatting and be detected as two people. These (anonymized) examples of the variations in formatting of author names are all taken from the same institution, and occurred in consecutive sequences of records. A record for "A White" and "B C Black" was followed by "A. White" and "B. Black"; "Green, DE" and "Green, E" occur together; as do "Fred Brown" and "F Brown" (despite the schema specifying initials only). Other institutions varied the capitalization, spacing, and ordering of name components. Often the field was misused, containing inappropriate values such as "The *project* Research Team" (interpreted as "T. P. R. Team"), or metadata such as "A. White (editor)".

177 author names contained international characters and were not represented in a consistent character encoding. There were 2104 records (2.0%) containing a total of 3553 ambiguously encoded characters in any imported field. There is no way to positively identify many character encodings; this is especially true of traditional 8-bit encodings such as the ISO-8859-1[178] standard. The only efficient way of handling this situation was to manually substitute these characters into a standard encoding when they were found, with the aid of several scripts. This involved guessing the encoding of unfamiliar foreign names and searching the Web to see if the resulting name occurred frequently on Web pages. Occasionally it was necessary to search the Web for a copy of the publication for a definitive interpretation, for instance where a name was ambiguous. The clear lesson learned here is that a standard encoding for data entry should be specified when ambiguity may occur; a Unicode based encoding, such as UTF-8, should be used to ensure all characters can be represented.

The subject field in an EPrints record is an important part of the repository's organization, as it groups records together providing a way of discovering related material. With such a large corpus, including all subjects studied at all UK academic institutions, attempting to come up with a good subject hierarchy proved impractical. A rough hierarchy was imposed based on the submissions' Units of Assessment, which are gross categories each covering many fields of research, for example "Physics" or "Computer Science", and subdivided by research group within the institution. As institutions define their research groups freely this produced a poor classification; at Southampton, it is unlikely that every paper produced by the IAM group (Intelligence, Agents, and Multimedia) could be meaningfully considered as being in the same subject except at a very high level. Similarly, groups from two institutions which were in the same subject area would be separated if their names were formatted differently, such

as “AI” and “Artificial Intelligence”. Within the limits of the source data, this at least provided a way of browsing a meaningfully restricted subset of records. Building a high quality subject tree would require domain knowledge, and classifying each submission appropriately would also need time and, given the data available, a significant level of manual intervention.

The lesson learned from building RAEPrints is that information once lost is difficult, if not impossible, to recreate. Once authors are named inconsistently they can only be reconciled manually or by unreliable heuristics. Text strings without knowledge of their character encoding are ambiguous. Appropriate classification is best produced by experts. In the next section RAEPrints is compared to WWWConf, which was based on a richer dataset leading to more reliable and useful metadata.

6.2 WWWConf

The WWWConf archive stored the proceedings of the ACM International World Wide Web Conference. It was built primarily for the 2004 conference to display the most recent submissions, but to also include as much historical data as was available based on an older EPrints archive. Updating this archive also served as another example of the issues in data manipulation and demonstrated the increased utility gained from higher quality metadata, as well as providing me with more experience into the technical details of EPrints.

With this smaller data set it was more reasonable to assume that each author name is unique, therefore it was possible to support browsing by author name. Also the conference tracks could be used as a subject hierarchy, providing navigation by topic. Full text was available, as were structured references which had been manually entered. ParaCite[179] was used to link citations: an experimental tool which searches many Web-based repositories to attempt to provide a link. Conventionally ParaCite takes a citation string and attempts to parse it; as in this case references were stored as XML rather than free text, parsing errors were prevented and ParaCite was used only as an OpenURL resolution service. Internal links (to other World Wide Web Conference papers) were handled because WWWConf was indexed through OAI-PMH by CiteBase, which is queried by ParaCite.

Author and citation linking in WWWConf was text- rather than identifier-based. Nonetheless, because of the limited scope of the repository it allowed these fields to be used for browsing, on the assumption that names were unique

within the archive and references were interpreted correctly. This structure and linking made the data more valuable than the larger, but less complete data set in RAEPrints. Ignoring the fact that full text was not available for the RAE data, each record was a 'dead end' for browsing as it could only be reliably linked to very few other records.

The limited scope also provided a useful subject tree which was guaranteed to cover all papers, and improved the chances of a cited paper existing in the archive. Distributed, Web-scale services would require formal identifiers to achieve this reliability as the potential for ambiguity is increased.

The high quality metadata in WWWConf was dependent on an element of forward planning, both in the user interface and data model. The reason citations could be linked without parsing errors was that they were collected and stored in a structured way, rather than as plain text. First and last names were captured separately and stored in multiple fields, and were recorded in full despite being displayed as initials. The RAE collection process, by simplifying the data model and storing only one field, produced data which was difficult to parse and could not afterwards be used reliably for linking, or even be rendered in a consistent citation style.

Comparing these two archives makes it clear that good quality metadata allows better use to be made of the data it describes, and also that once information is discarded it may not be possible to recreate it. A repository, which exists to disseminate its contents most effectively, should use a user interface and data model which avoid throwing away information in this way.

6.2.1 Topic categorization

It is notable that the hierarchy of topics provided by the conference itself (Figure 5) was a good subject tree—unsurprisingly, as the arrangement had been designed by the conference organizers to group papers on a similar topic for the benefit of delegates. This metadata could be equally valuable in an institutional repository with broader scope, as it would provide a more detailed level of grouping than papers from the same conference or conference series. This is a case where using an identifier to refer unambigously to the topic would be particularly advantageous:

- The titles of conference sessions are likely to be long and open to abbreviation or variations in formatting. Grouping records by looking for matching

- ACM Categories
  - C.2: Computer-Communication Networks
  - D.1: Programming Techniques
  - D.2: Software Engineering
  - D.3: Programming Languages

    [. . .]
- Conference Tracks
  - 2004

      [. . .]
    * B-1: Server Performance and Scalability
    * B-2: Mobility
    * B-3: Web Site Analysis and Customization
    * C-1: Usability and Accessibility

      [. . .]
    * C-2: XML
    * C-3: Semantic Interfaces and OWL Tools
    * C-4: Semantic Web Applications

      [. . .]

Figure 5: A subset of the WWWConf subject hierarchy.

strings will therefore often fail. Selecting a predefined value containing the unabbreviated name makes browsing easier for users unfamiliar with the abbreviations, and is more likely to be valuable for search tools.

- Conferences already have access to the necessary data for internal use. Providing it in a suitable format for exchange with repositories would be simple, especially if a universal format can be agreed by major organizers. These organizations have an interest in promoting their conference, hopefully seeing the value of standardization. WWW2006, where the delegates were likely to be interested in the Semantic Web, produced a comprehensive RDF dataset[180] of the sessions, papers and presentations, speakers, associated events, and timetable, intended for reuse and experimentation. The identifiers in this data could link publications in institutional repositories with the conference itself and make connections between other conference resources, thus raising its profile.

- With an identifier, each topic has its own identity. It can be linked to the conference site or external data for more information, or grouped into a hierarchy to improve browsing. Any time better use is made of the data in an EPrints record, it potentially raises the profile of the research, author, publisher or conference.

Many earlier examples have used author names, so this conclusion helps justify why other entity fields may benefit from the use of identifiers.

6.3 ECS EPrints

In this section my two example repositories are compared to a real world EPrints archive: ECS EPrints[181]. This is a live repository with over 11,000 records, maintained by the School of Electronics and Computer Science and which has evolved over time according to the School's needs.

ECS EPrints demonstrates some of the potential of a repository with Semantic Web features. It combines the ECS RDF service[173] with the identifier support of EPrints 3 to provide reliable linking between publications, their authors, and associated metadata from the repository and elsewhere.

The unique number assigned to every member of ECS is used to disambiguate members of the school in EPrints. Each field which stores a person's name has an associated identifier field, which contains this number if that person is

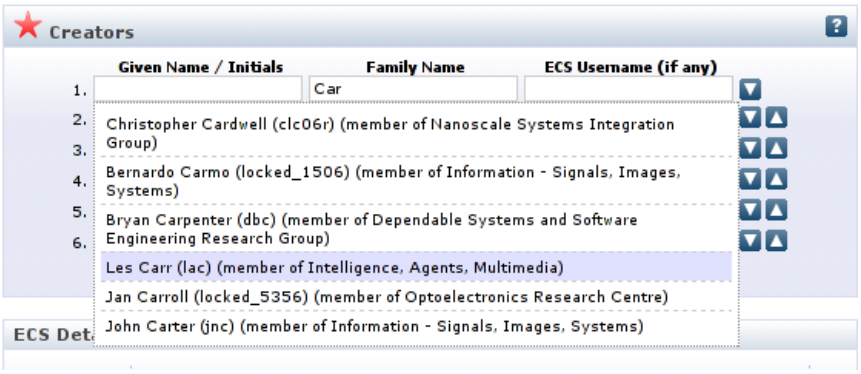

Figure 6: Partially entered text in an EPrints 3 submission. A list of matching values is automatically displayed. These values include both the name and an ECS user ID, a unique identifier for members of ECS.

Figure 7: Selecting an author from the list completes each of the input fields. When the record is submitted, the author name will be unambiguous and linked to a page listing that author's metadata and publications.

Figure 8: Part of the list of authors in MIT DSpace[182]. The same person has three separate entries, formatted differently.

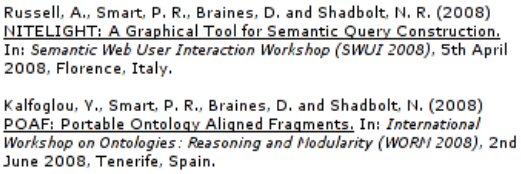

Figure 9: The publications list on Nigel Shadbolt's page, extracted from ECS EPrints[181]. Two formattings of the same name appear together, linked because a common identifier is used internally.

within the school. When submitting a record, typing characters into a name field automatically performs a search on the personnel list, displaying a list of matching names alongside the input field (Figure 6). Selecting a name fills in the name fields but crucially also inputs the identifier, therefore saving time for the user and simultaneously gathering better quality data (Figure 7). This aspect is important as it means entering an identifier is less effort than it would be without this feature.

A demonstration of the value of this identifier can be seen in comparing ECS EPrints with MIT's DSpace[182]. The DSpace author list includes separate entries for "Abelson, H.", "Abelson, Hal", and "Abelson, Harold", as if they were different people; this is because the author names in each submission were formatted differently (Figure 8). In ECS EPrints, the publications list shows some records for "Shadbolt, N." and some for "Shadbolt, N. R."—but visiting Nigel Shadbolt's personal page links to both of these, because the same identifier is common in both cases (Figure 9). Variations in formatting may be arbitrary, but may also be enforced by publishers' varying house styles; this highlights how an identifier can make metadata more useful while still retaining its faithfulness to the official published rendering.

Once a record has been submitted, any names which appear in the metadata are linked to a personal page (outside the repository) which is generated from their metadata. This includes such details as contact information, associated people such as taught students, publications, conferences attended, and tags representing fields of academic or personal interest. The data in this page is obtained from departmental databases, user input, and EPrints itself, and is also available as RDF.

These additional browsing, tagging, and summary features are an example of the possibilities of Semantic Web integration with repositories. The missing links in this scenario are that identifiers are restricted to members of ECS, and that they are only accepted for people.

An advantage of restricting the scope of identifers to within the school is that there is no need for distributed search. As all the linked data associated with the repository is under the control of ECS, it can all be obtained from local databases guaranteeing data availability. Obviously this is not a solution for data which is not under the control of the local institution.

If all of an author's work is submitted to their institution's repository, browsing could in some cases be delegated rather than distributed. The RDF service resolves URIs by redirecting them to a page with that URI's metadata, in either RDF or HTML as appropriate. If this standard were followed by each repository, a user could browse one page at a time, jumping between repositories which each displayed local information but never requiring data to be combined dynamically from different sites. For decentralised data this would only be a solution if every cooperating organization harvested each other's data to produce a combined view. This harvesting could for example be based on OAI-PMH, remote SPARQL interfaces, RSS, or simply periodic manual updates.

This example demonstrates that useful data exists outside the control of the repository, which can be usefully integrated into the repository's interface. In this case research projects, conference attendance, relationships between people, and lists of interests are all available.

However, the mechanism used to link these entities to the repository is limited. Connections between these resources can therefore only exist via people, not other data types; and this is further restricted to members of ECS, due to the identifier scheme. By using URIs as identifiers, given suitable infrastructure the repository could link authors in different institutions and provide similar

features for other data types. This infrastructure would take the place of the departmental databases which provide the backend for the RDF service.

6.4 Conclusions

The examples in this section show that capturing identifiers alongside repository data can add value and improve usability. In the next chapter, I explore the way in which data in this form can be captured and used with two prototype user interfaces.

7 Semantic sidebar

In this chapter I describe the development and evaluation of two prototype user interfaces, based around a server and Web browser plugin, to experiment with methods of interaction between the repository and users. These tools demonstrate the extra possibilities which arise from collecting metadata with associated identifiers.

The first prototype was an RDF browser which allowed direct exploration of linked metadata alongside the standard repository user interface. The goal was to be a demonstration of how Semantic Web technology might work with EPrints and add value to data in an archive.

7.1 Purpose

The publications and authors in ECS EPrints have reliable identifiers, which make it possible to link these objects and their metadata together, producing a linked data graph. The sidebar tool provides a way of exploring this graph through traversal and structured querying, to help a user of the repository discover new material which may be of interest, and thereby highlighting the benefits of providing these identifiers.

Following Semantic Web standards when publishing data leads to the possibility of third party services making use of it. The sidebar demonstrates this by requiring no changes to be made to the repository. The implications of this architecture will be discussed below.

It also shows that linked data removes the constraints of a traditional Web interface, allowing a new interface to be built on top of existing data. The repository and sidebar view are partially integrated; having browsed to a repository item in one view it can be synchronized with the other.

7.2 Architecture of the browsing sidebar

The architecture and relationships between entities are shown in Figure 10.

1. The user's Firefox browser plugin displays a button in their browser toolbar. The button activates the query service based on the URL of the currently viewed page.

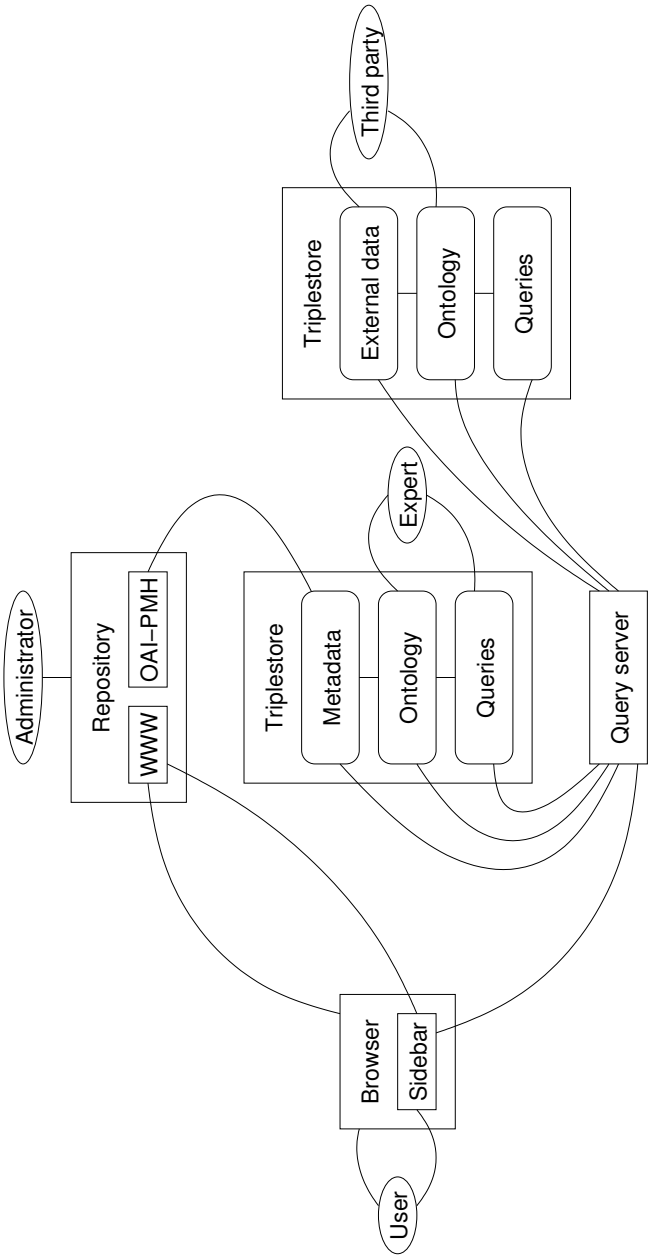

Figure 10: The relationships between entities in the browsing sidebar. The repository is harvested by a triplestore, which is augmented with an ontology and queries based on the repository data. A query server links the repository interface with the triplestore, and possibly external resources, via the user's sidebar, allowing graph traversal and providing recommendations.

Browser integration is required because the security barriers in JavaScript prevent code in a window from accessing information about other windows. The alternative would be adding a link in the HTML returned by the repository, but this method means that the repository need not be modified to provide support.

2. An item of interest in the repository is associated with a URI based on the OAI-PMH identifier, calculated from the URL of the page being viewed. The same identifier is generated from an eprint abstract page and the documents which are part of it.

   Metadata about each item is regularly harvested from the repository using the OAI-PMH protocol. The script performing this harvesting transforms the data into RDF in the Dublin Core ontology, which is also stored in the triplestore.

3. Alongside the metadata and ontology, an expert constructs SPARQL queries associated with each data type (such as eprint or person). These queries are intended to directly link an entity with others which are related, but have a more complex relationship than a single step on the linked data graph. For example, two papers which have a common author are two steps apart, as they have a metadata value in common, but there is no field in a paper's metadata which links directly to another.

   In the prototype only a single query server is supported, and the eprint URI is generated using a simple regular expression from the viewed URL. While a complete implementation could support a configurable set of databases for multiple repositories or query services, using the system would require more effort from the user. Alternatively, a larger scale universal system could be built, but this could introduce issues of data integrity and ownership as well as privacy concerns. Solving this problem is one reason to provide Semantic Web services which are integrated with the repository; this point is discussed later.

   Queries for an eprint could be "all eprints with two or more authors in common with this eprint, ordered by date", or "all authors of papers in the same research project as this eprint, ordered by number of papers". Each query has an identifier and description, and is stored as a string of SPARQL in the triple store.

4. Activating the sidebar causes the triple store to be searched to find the given resource's data type and a list of queries relevant to that type.

(Figure 11) The plugin opens a sidebar (a standard feature of Firefox), and makes an HTTP request to the query server containing the resource identifier.

A list of possible queries is returned to the client software, combined with a list of all triples related to the selected record. This allows the user to browse the metadata directly as well as using the query system. For objects which have a viewable URL as well as an identifier in the triple store (such as eprints in the repository) an additional link is displayed to open the object in the main window, navigating away from the original object.

5. The user chooses a query in the sidebar by following a link, which encodes the query identifier as well as the target resource. The query server obtains the SPARQL for this query, inserting the resource identifier, executes the query, and returns the result. (Figure 12)

   From each result the user can continue to browse metadata, choose queries, or navigate to another record in the repository. Figure 13 shows the view having browsed to another entity.

7.3 Features of the browsing tool

The significant features of this tool are:

- The triplestore is independent of the repository.

  Metadata is extracted from EPrints and stored in a separate triplestore for browsing. This used the OAI-PMH data access API to gather data: a design decision which means it is not necessary to make changes to the repository configuration.

  This separated architecture demonstrates the principle of data exchange in the Semantic Web: how free exchange of data in a standardised way adds value to it. OAI-PMH is an extensible protocol suitable for bibliographic metadata, and is already integrated with EPrints in its default configuration which is why it was chosen here, but it is unsuitable for freely structured data for which RDF was designed. A possibility with the sidebar tool is the ability to incorporate data which may not fit into any predefined structure.

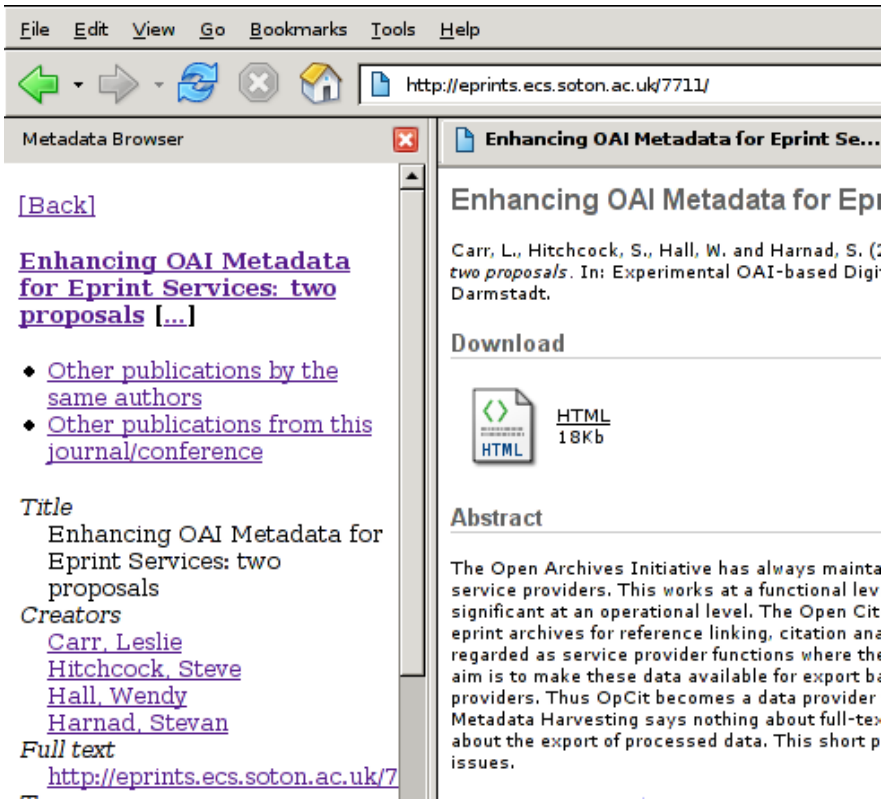

Figure 11: User interface of the first sidebar prototype. When the user activates the sidebar tool when viewing an eprint, its metadata and a list of queries are displayed in the sidebar.

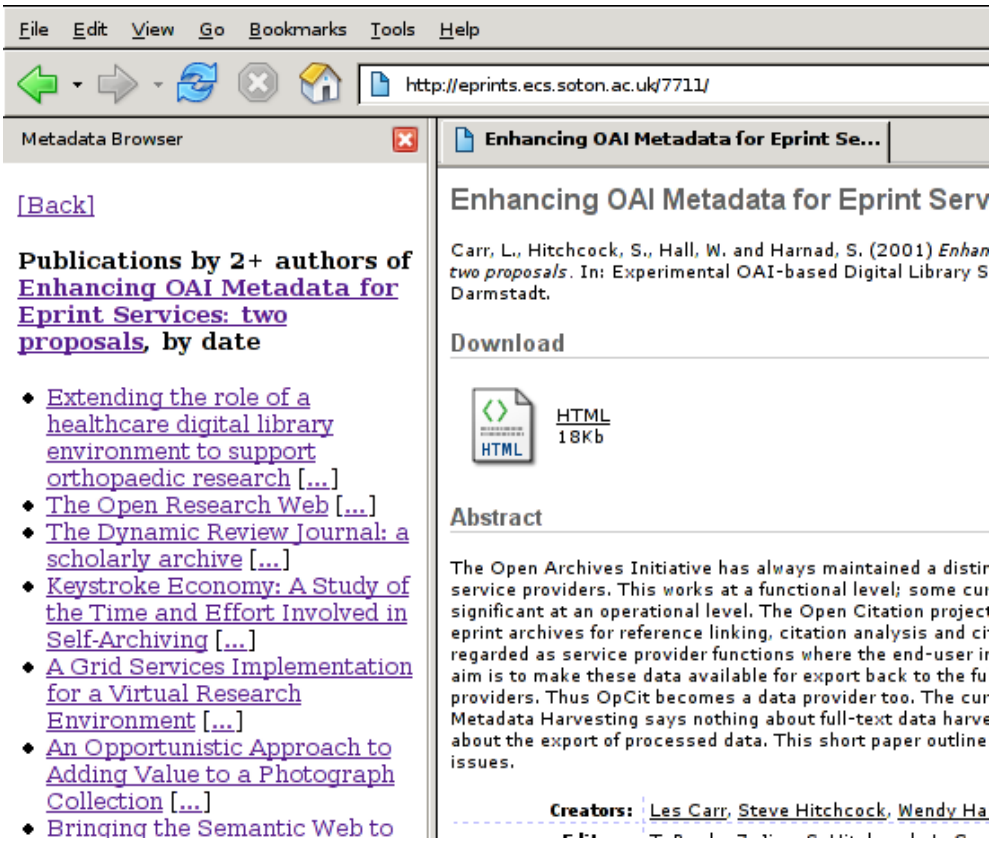

Figure 12: Selecting a query shows a list of results. The main link of each result navigates within the sidebar; the ellipses link to the EPrints record, which updates the main window.

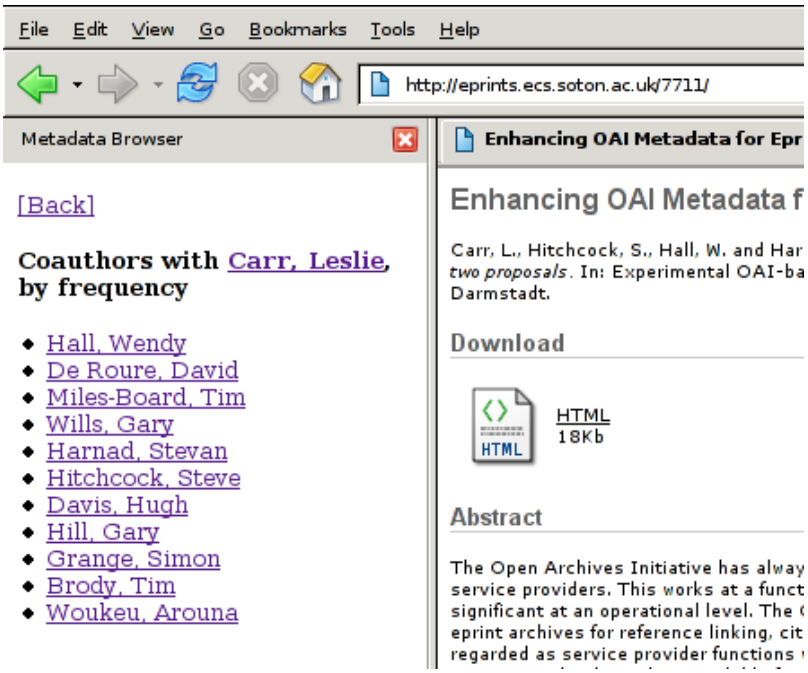

Figure 13: Different data types have a different selection of queries. In this case the results have no corresponding Web page, so no ellipses are displayed.

Though the architecture is an example of a third party Semantic Web service, a greater benefit could arise from integrating the tool into the repository itself. The Semantic Web encourages connections between formerly separate data sources, adding value to both, but this process could improve the quality of the data in the repository as well as disseminating it. For example, the user interface could draw on data in a triplestore to provide suggested input, thus collecting identifiers along with strings in the cases where authors have metadata. The triplestore could also be used to store supplementary data for records in the repository, which could be too loosely structured to store in regular database fields; if a repository stored raw data from experiments, for instance, an ontology describing how to format the data could be stored alongside.

- Metadata browsing is alongside, but integrated with the archive view.

  When the user activates the sidebar tool the archive window is unchanged. However the URL of the currently viewed page, which can be either an eprint overview or document page, is sent to the server side tool and parsed to identify the eprint. The results—metadata and query results—are displayed in an additional panel in the Web browser window.

  The tool has two related purposes: metadata exploration and discovery of associated records. The first allows the user to evaluate the eprint by studying the potentially richer metadata available from the triplestore. The second allows other records in the archive to be found by following the connections implied by the metadata. For the first purpose it is important that the user keeps the context, the currently viewed record, available as it provides core metadata and navigation as well as site structure and branding. For the second, the user wishes to leave the current record and discover others; in this case the context is the sidebar's query results, and the user needs to be able to display the contents in the main browser window.

  Therefore when the user selects an eprint object in the sidebar, an additional icon is displayed which changes the view in the main window. For other data types there is no "contents" web page to display, as the archive stores only publications, so this option is not available; selecting an object displays its properties in the metadata panel.

  The separation of the sidebar and repository content allow it to bridge between different data sources, such as two repositories or a repository and third party service.

- The triple store can contain data from other sources besides the repository.

  EPrints is designed to store a single core data type: the publication. The properties of publications are merely values, and are not considered first class objects with their own metadata. For instance, the bibliographic data on a publication in conference proceedings has separate fields for the name of the conference, when it took place, and its location, even though these are properties of the conference itself and not the paper.

  By associating EPrints with a triplestore, other object types can be described and linked, possibly providing richer information to the user or assisting in information management. Some examples:

  - As a first class object, a conference or journal could be added and given metadata before there are any associated papers. User interface improvements could make use of this data to assist submitters.
  - Providing extra information about other data types becomes possible or simplified, for instance providing authors' contact information. Storing this data within a publication means that updating the information would involve changing the copy in each record, increasing maintenance effort for the administrator or submitter, while also resulting in a new version of the record even though the publication itself was unchanged.
  - Repository data can be linked to external sources, as triplestores can aggregate arbitrarily structured data. A publication could be linked to a video recording of a seminar, which could in turn link to the people who attended, which could list their research interests. Adapting the repository specifically to store each new piece of data would be far more complex than the general solution of integrating a triplestore.

  However, as argued before, linking between different types would require URIs as identifiers for each. EPrints currently does not have an identifier field for any types other than people, so this linking would be unreliable with the data available due to formatting differences.

- Querying support to discover serendipitous information.

  EPrints already includes integrated searching and browsing features, keyed to individual fields: the user can specify values or ranges for a set of fields. This capability is appropriate for finding a known record or searching

with simple criteria, and can be performed efficently by the underlying database.

However, the querying feature of the sidebar tool is intended to provide more complicated searches through a SPARQL query language interface. The aim is to support queries which can reveal related records to the user, which are relevant but associated in a way too complex to be picked out by simple search. The knowledge engineer prepares these queries in advance, and they are presented to the user as suggestions as an alternative to directly browsing the metadata objects.

While viewing a paper, a typical query which could be displayed is: "Find all the papers written in the last year, by any author who has collaborated with at least two of the authors of this paper on more than one occasion." This might help the reader find authors whose work could be of interest, but whose papers might not share any metadata fields in common with the one displayed.

The querying feature is intended as a demonstration of the applicability of Semantic Web tools to a problem. Providing recommendations was chosen as a simple example of a valuable feature which could be built using only standard Semantic Web tools and languages. It is not likely to produce more useful recommendations than a purpose built engine using heuristics and statistical analysis. Recommender systems are an active field of research[183] which use more complex algorithms than my system can express; but Semantic Web standards would still be suitable as a data interchange and description mechanism.

- Coreference is handled transparently for the user. Alternative spellings of names or publications are grouped together, and the associated properties and related items of each are merged.

  Behind the scenes each instance of a string harvested from the archive is converted to an automatically generated identifier: a hash of its string value along with its data type. In the RDF model this allows assertions about this value, as only a URI can be the subject of a triple. The first assertion to be made is the string representation of the URI: the original string. Hence no data is lost, and the record can be displayed the same way, but the new identifier can now be manipulated.

  Indirectly referring to names through an identifier allows the identifier to be unified with others, for instance using the `owl:sameAs` predicate. This is not the same as stating that the strings are the same—of course

they are not—but the entity they refer to may be the same. In another context the same string may not refer to the same entity, such as where two people share a name. With an identifier as the prime data type rather than a string, these cases can be disambiguated explicitly: it is possible to say "*this* person called Harry Mason" instead of "*a* person called Harry Mason" as before.

For simplicity in prototyping, the conversion process makes the assumption that identical strings in the archive correspond to identical entities. This permits "false negative" coreference to be handled by asserting the unity of two identifiers, but not "false positive" coreference, where the same name is assigned to separate objects.

How could both cases have been handled? Identifiers are composed of the data type and value, hashed, but they could be generated or assigned using another method. Manual assignment is the most reliable—there will always be cases where only a human has sufficient capability and knowledge to make a correct decision. The disadvantage is of course that identifier assignment is a specialist task, and all archive submitters cannot be expected to deal with data integrity.

A more pessimistic automatic system could include the record ID as part of the hash. Every occurence of a name would have a different identifier, which alone would be awkward except that coreference statements can be automatically asserted where an identical name already exists. This method permits identical names to be different, as the automatic statements can be removed manually. However it creates a large volume of data which could be awkward to handle without a special tool.

For maximum data reliability the element of manual intervention must remain, for the cases where it is otherwise impossible to produce correct data. For the other cases where automatic assistance is desirable, the system could make an assumption based on a heuristic but give the user the opportunity to correct that assumption. Coreference features built into the repository would support this, as the user interface could present a list of choices when appropriate. Ultimately the submitter may be in the best position to decide whether things are coreferent.

7.4 Critical reflection

This prototype is an example of how repository data might be extended into the Semantic Web. However, it has several shortcomings as a user interface and architecture, and is limited by the data available in the repository.

7.4.1 Browsing interface

The sidebar tool is constraining relative to other RDF browsers. As a user interface it is inflexible and excessively simple, and the same functionality could now be provided simply through generic RDF browser technologies, such as Tabulator[39], IsaViz[184], and Fresnel[43]—or through additions to EPrints itself.

Alongside the standard EPrints interface, which is tailored to the needs of repository users in general and can be configured by each institution, a simple RDF browser is awkward. Unlike similar tools, such as Tabulator, the sidebar does not maintain a visual display of context and the thread of navigation, and the user may quickly become lost browsing a resource unrelated to their initial selection. In this scenario pure exploration of RDF may be inappropriate for users' needs. For the cases where direct graph exploration is useful, data can be obtained from EPrints through OAI-PMH by third party tools and processed offline, and while supporting RDF could streamline this task it would provide no qualitative improvements without first improving the data itself.

7.4.2 Querying

Though the direct graph traversal is not without drawbacks, the query functionality is a new feature which seems that it could be beneficial for users. It is an extension of the exploration possible within EPrints, and a generalization of the concept could link repository data with external resources which cannot be integrated into the EPrints data model.

Choosing appropriate queries for the sidebar requires expertise both in SPARQL and the research domain. While it might be assumed that coauthorship and metadata values in common imply a useful link between papers, there is no guarantee that this is so and a more complex algorithm might be required for useful recommendations. It is also uncertain that users will find browsing via the query mechanism useful, as it makes the details of its selection process explicit

rather than presenting a list of potential results directly, using internal heuristics to evaluate recommendations but hiding the unnecessary details from the user. This would correspond to an exploratory rather than goal-orientated browsing of the data; if the target resource was already known the user could search for it using the existing tools.

Any advanced analysis tool providing linking or recommendations will encounter the problems described above in the analysis of the WWWConf and RAE data. It is not clear that deploying such a service on top of current repositories would provide a real benefit, due to the inherent informality of typically stored bibliographic data. In testing this prototype the only metadata which was reliably useful was the author field, because of the unique identifier optionally stored alongside names of ECS members corresponding to their internal staff identification number. This identifier unified different renderings of the same name, which were common despite being within the same institution. Other entities, such as events and journals, had no identifier and had even greater variation in formatting.

The sidebar prototype makes explicit the mechanisms which might be used to make recommendations using Semantic Web standards. Such recommendations may indeed be useful for repositories, and the Semantic Web could provide a mechanism for them to be distributed and potentially more accurate; however, the approach taken here is too simplistic. Advanced recommender systems could use this type of query behind the scenes, along with statistical and graph analysis techniques to provide more accurate recommendations. This topic is a large research field and will not be explored further here.

7.4.3 Summary

This prototype shows that Semantic Web technology can be applied to add new features for a repository. It is limited by the availability of suitable source data and lack of integration with the repository itself. For the next experiment I focused on input, aiming to build a tool which could mitigate the lack of data suitable for linking.

7.5 Input sidebar

To better demonstrate a clear benefit from integrating Semantic Web technology with a repository, a different direction is more appropriate. To provide quali-

tative improvements for usability (for browsers and submitters) and analysis, and to support systems such as the browsing sidebar described above, Semantic Web ideas can be applied at the user interface to capture better data in the first instance. This prototype was based on the earlier code and used a similar architecture of Firefox plugin and query server backed by a triplestore (Figure 14).

The purpose of this tool is to help repository submitters input identifiers alongside the standard metadata, to improve the value of captured data. It allows the user to select an entity from lists of possible values instead of typing, then automatically fills in the identifier field behind the scenes.

The earlier prototype was modified to detect input form elements on Web pages. Each form field, identified by the URL and element ID, is associated with a data type from the ECS ontology. Selecting such a field, the user activates the browser plugin from a toolbar button; this opens the sidebar and sends a request to the triplestore for entities of that field's type. As the purpose of this prototype tool was to demonstrate the value of identifiers, "person" is the only supported data type, using the unique numbers assigned to ECS members.

Results are returned in a list which can contain richly formatted data. Using data extracted from the ECS website, searching for people could display a picture alongside their name. Selecting an item from the results list copies the values into the selected field on the form; the name into the main field, but also the number into the associated ID field.

Characteristics of this interface are:

- It encourages users to enter the numeric identifier for a person when they submit to a repository, thus collecting better quality data. (Name fields are always presented with an optional identifier field, but the submitter must normally look up the appropriate value manually.)

  This number eliminates the need to parse names and disambiguates in two cases: where alternative renderings of the same person's name exist (though this could be resolved semi-automatically for an existing dataset), and where two people share a name (which after data collection could only be performed by a manual process or statistical methods).

  Once collected, this data can be used as in earlier examples to link or analyze people and their relationships with other data in the repository.

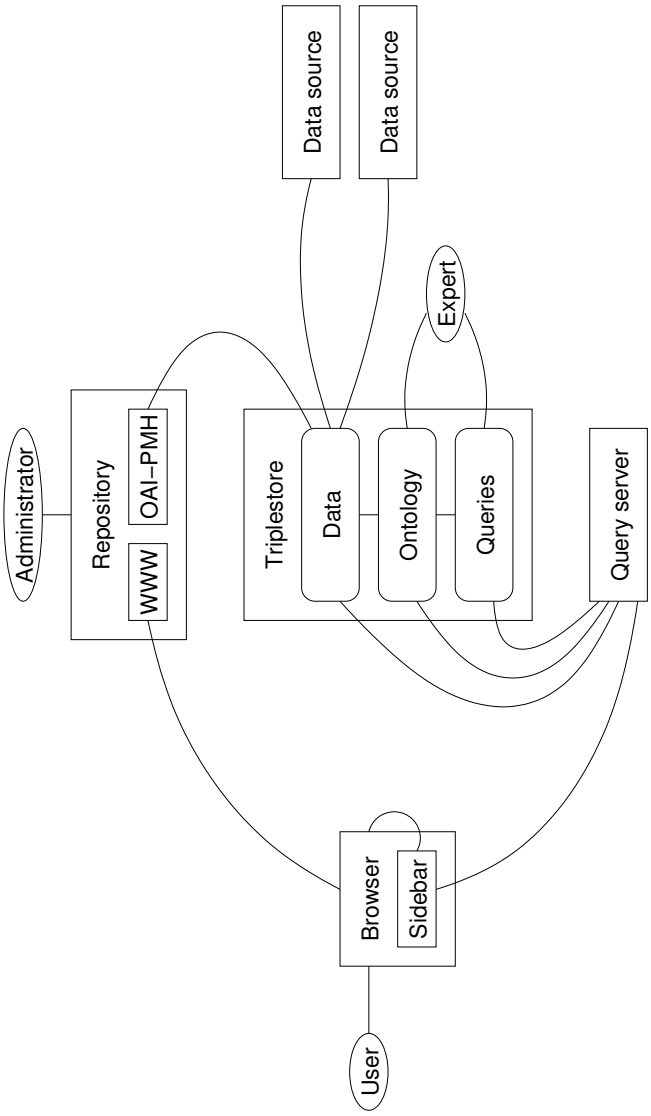

Figure 14: The relationships between entities in the input sidebar. Data harvested from various sources, such as personnel data and the repository itself, is collected in a triplestore. Queries define possible subsets of this data which might help the user find the desired entity. Selecting an entity in the sidebar enters the appropriate values into the repository input fields.

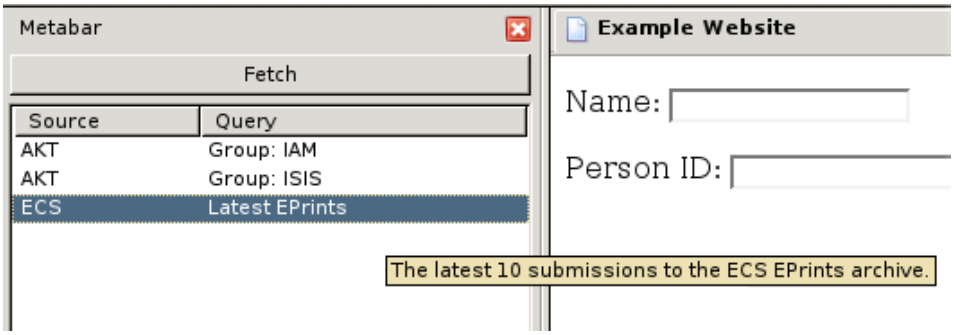

Figure 15: The user interface of the second sidebar prototype. The Fetch button asks the server for a list of data sources which relate to the URL of the page visible in the main window. In the lower sidebar panel a list of entities will be displayed.

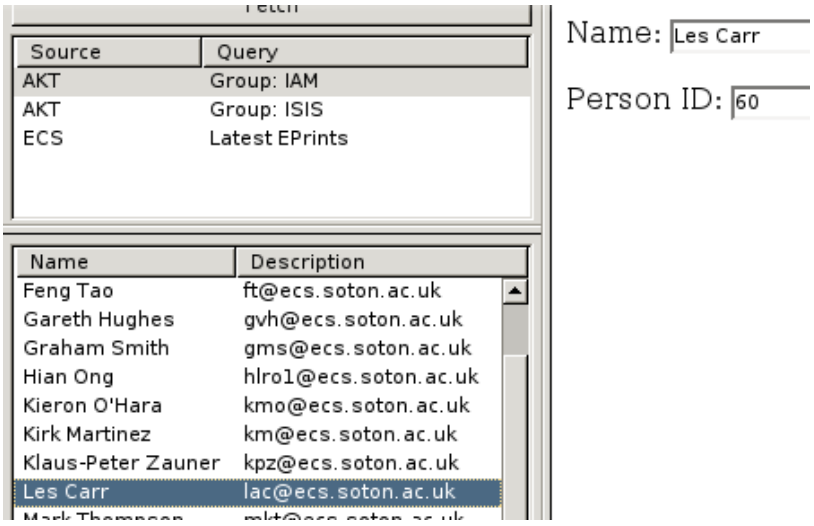

Figure 16: The user has selected a query and result, which automatically copies the appropriate value into the form. Note that more than one field may be filled in, and may include information not displayed in the results list.

- It avoids the user having to type in names, instead being presented with a preferred rendering of that person's name. For other data types, this would be beneficial by making it more likely that a consistent rendering is used throughout the archive. This preferred name can reflect the policies of the archive or be based on authoritative information. In the case of personal names however, it may be more desirable to accurately store the metadata according to the official printed rendering with the numeric identifier unifying any variations.

  Though clicking on a selection saves typing effort for the user, they must instead scroll through a list of possible choices, which is likely to be a far less efficient process than typing the name. Using this prototype also involves a distracting jump between the repository Web page and the browser plugin. A more advanced implementation could integrate input directly into the form field, however, also allowing the user to type a few letters before displaying a list of matches.

  Integrating this service into the repository could potentially enable heuristics to be used which suggest more likely choices first.

- A list containing formatted data can provide additional information to help the user where a name is ambiguous. This is critical in the case where two people have identical names, but might be useful if the submitter only knows the initials and simple text completion would not give enough context. The query functionality also allows the user to pick from a subset of entities, for instance the members of a particular research group, which helps to disambiguate if a large data set is available.

- Separating the repository and sidebar service allows data to be gathered from third party sources. In this example author names were obtained from the triplestore constructed for CS AKTive Space[42]. It means that repository data can be linked inside the triplestore with other sources through common identifiers (where they exist), providing more detail for disambiguation. In this case the email address of each author is displayed.

  However, as before the architecture of a browser plugin separate from the repository has several problems.

  The user is obliged to download and install the plugin before the feature can be used, and if the code is updated they must download an upgrade. It is also tied to the browser they use; in this case only Firefox users could access the service. The server must be kept up to date to be able

to select the correct input fields in each repository it supports, and must periodically harvest data from an appropriate source. For a user to work with two unrelated repositories, they may also need the ability to select between two servers.

The awkwardness of this process demonstrates why the repository should itself contain these features. Once installed, the sidebar makes it easier to collect identifiers alongside data, but the overall process is more difficult for users than typing data in directly. It also requires users to be conscious of the Semantic Web and be proactive in gathering suitable data.

An integrated interface keeps control of the repository's identifiers and Semantic Web features in the hands of the repository administrator. They can harvest data from appropriate sources at their site, structure the repository database to store URIs, and add user interface features to both gather and make use of this linked data without being intrusive for users. By using URIs, these identifiers can be globally recognized even though they are assigned by each repository, and can be unified using the coreference handling techniques discussed earlier.

- The most significant problem, from the point of view of the Semantic Web, is the fact that unique identifiers still only identify people, and (in this scenario) are limited to short integers with only local scope. The architecture of EPrints 2 only provides an identifier field for names, and it has no prescribed formatting. Email addresses and site-specific data types are commonly used, but URIs would both support Semantic Web technologies to link data and allow identifiers to be used for any data type.

  The fact that identifiers have only local meaning in this prototype outweighs any advantage gained by separating the sidebar tool from the repository.

7.5.1 Summary

The input sidebar was developed with the goal of supporting the capture of identifiers in repository data. It achieves this partially but the interface in the prototype is difficult to use and would not scale to a real scenario. This and the previous sidebar prototype together justify the inclusion of identifier handling features directly into the repository, which could provide the benefits of both interfaces at a much lower cost for the user.

Though this prototype could be extended to better integrate with the repository and provide an improved UI and system architecture, it would still require users to be proactive in explicitly supporting the Semantic Web when they input data. On the other hand, providing similar features as part of the repository would require significant code changes and effort for administrators.

7.6 Next steps

Both of the above tools were developed for version 2 of the EPrints software. Since then, EPrints 3 has been released which features a new level of configurability and a streamlined user interface. For instance, autocompletion is a standard feature of all form components and can perform arbitrary searches on the repository or external data. Composite metadata fields exist which can support the ID attribute in EPrints 2 in a generic way for any data type. The plugin based architecture allows administrators to install new functionality easily, which is important if experimental Semantic Web features are to be deployed by archive administrators.

Supporting Semantic Web services is not necessarily considered important by repository administrators. The hypothetical improvements to interoperability and quality of data, which are anticipated by Semantic Web researchers, have a bootstrapping problem; if no other services exist which could be interoperable, there is no benefit. Tim Berners-Lee's visionary scenario[57] is intended to appeal to the "early adopter" community of academics and technologists, whose work will not immediately realize the vision but will progress towards it. EPrints is in a similar position to push development in the repository community, as occurred with OAI-PMH[136], by providing features by default thereby producing a critical mass of supporting sites.

This is why I believe EPrints should integrate identifiers into every appropriate field, using Semantic Web formats and protocols and with user interface support, as a default feature. For archive users it would support disambiguation and simplify data entry, and it would provide the base for linked data services which would enhance usability further and add value to the data in the repository. Therefore by promoting the Semantic Web and bootstrapping tools which would increase the visibility and impact of research, this feature would encourage the progress of science.

8 Recommendations for a Semantic Repository

The conclusion of my exploratory work is that repositories should store URIs which uniquely identifying the entities which exist as metadata in the repository. This will allow repositories to exchange data through the Semantic Web with a variety of other sources, using it to improve the quality and completeness of the data stored in the repository as well as its value to authors, readers, and third parties.

The tools I built demonstrate some of the possibilities. However, EPrints 3 is a more flexible platform with the advantages of being able to support this feature with effective integration into the user interface and submission workflow, and the power to encourage its adoption in the repository community. This section describes a scenario where this feature is added to EPrints and how it interacts with other entities—people and organizations—in the wider research community.

8.1 Proposal

Each metadata field which is resource valued, rather than data valued (e.g. people, journals, publishers, but not titles or dates) should be replaced with a composite field made up of a URI and the original field (Figure 17). In EPrints 3, the software automatically creates the appropriate database structure required, and with the appropriate configuration handles the extra component in the user interface.

The existing autocomplete code should then be modified to search for and store URIs alongside the text entered by the user. Additional validation could optionally be added to avoid data integrity problems, for instance to detect if the URI exists in an institutional database but the text is formatted differently.

In some cases there will be no authoritative database to consult. I propose that the repository nevertheless produces identifiers for submitted data. Where a submission has no URI, one can be assigned based on a hash of the data or a pseudorandom generator. Using standard methods to unify URIs, these can be rationalized later or used as the basis of a gazetteer, feeding back into the user interface. Future submissions can then reuse the assigned identifier when referring to the same entity.

Figure 17: The fields in EPrints 3.1's default metadata which would change, and the proposed new structure.

Existing field	Proposed field
Creators	Creators
· Name	· Name
· ID	· URI
Publication	Publication
	· Name
Publisher	· Publisher
Place of publication	· Place of publication
	· URI
Volume	Volume
Number	Number
	Event
Event title	· Title
Event location	· Location
Event dates	· Dates
Event type	· Type
	· URI
Projects	Projects
	· Name
	· URI

The changes this would require in EPrints would be incremental, based on the configurable database structure, plugin architecture, and JavaScript autocomplete system. Administrators could therefore add this functionality to existing repositories, though I argue that it should become a default feature in future releases of EPrints to achieve a critical mass of support which would encourage the development of related third party tools. Nevertheless, individual adopters would gain disambiguation between entities, the potential for integration with existing databases, and preparation for future expansion.

8.2 Organization

8.2.1 Institutional repository

An institutional repository will be both a producer and consumer of data. A repository makes research material accessible and discoverable, and this will be enhanced by linking with other data sources: other repositories, organizational databases such as the ECS People metadata pages, publishers' sites, and social systems. These URI-based links will be another mechanism by which readers can discover research material and place it into context. Such linked data is the current focus of the Semantic Web[185] and is likely to connect with more diverse services as the Semantic Web matures.

Institutions will use internal and external resources at the user interface level to improve the quality of metadata entered into their repository, thereby encouraging authors to submit their work, streamlining the process and making the repository a more useful resource for the organization to evaluate its research. The UK Research Assessment Exercise is an example of a previously arduous task which would benefit greatly from links between the staff database, repository, analysis results, and the central RAE provided data.

The data stored in repositories will be used by third party analysis systems in the same way as it is now, such as quantifying its impact and identifying significant papers; however, reliable identifiers will simplify the process and remove ambiguities.

Progress in other aspects of the Semantic Web, particularly OAI-ORE and semantic search, will encourage repositories to publish more machine-readable data as it will begin to have a direct effect on research impact and accessibility.

An increased awareness of identifiers could lead to Semantic Web-based citation linking services. Currently URLs and DOIs are useful ways to provide a reliable link as they are easily parseable. If it became common practice to return a response containing (or linking to) RDF metadata, more use could be made of the citation automatically.

8.2.2 Subject repository

Subject repositories, such as arXiv and CogPrints, are in a similar position to those controlled by institutions except that the typical characteristics of their authors, readers, and submissions may be different.

A repository restricted to one field is capable of growing far larger than an IR could (arXiv, for instance, currently hosts over 470,000 records) and is therefore likely to be more useful for analysis. It is also more likely to have internal connections, through citations and collaborations. However, because such an archive would not have institutionally organized supporting databases, providing referential integrity through identifiers would take more organization and effort. User interface improvements to assist this process would still be beneficial but would require more bootstrapping. In this context, collaborative efforts such as the Names project[186] would be of particular value.

8.2.3 Publisher

Publishers facilitate the provision of organization and quality control to research. These roles are equally valid in the subscription-based, restricted access model which is the current status quo, and the fully open access model advocated by Harnad[121]. In either case, it harms a publisher if their articles are difficult to find (even if once discovered, access is restricted), as their commercial success depends on a journal being considered valuable to publish in or subscribe to.

As both online access to published articles and self-archiving become more prevalent, publishers are already adapting their processes[125], for instance by allowing non-subscribers to purchase articles individually. The growing Semantic Web is another way readers might discover or evaluate articles, and it is important that publishers respond to this. It is also inevitable that some self-archiving of subscription restricted content will occur, but journals can still benefit from this through interoperability, allowing repositories to become entry points into the journal via the Semantic Web. Journals could also benefit from adopting

the data structure I describe simply as an aid to usability and data integrity, for instance by supporting faceted browsing.

Though my proposal primarily relates to institutional repositories there would be mutual benefit from the involvement of publishers. For example, in repository metadata journal names are often abbreviated. Publishers could provide this data in full, in a standard form which could be harvested by repositories along with URIs for unambiguous linking. This would provide a robust link between self-archived papers and the journals in which they appear, promoting the journal while simultaneously allowing readers to browse to other related articles. The added value of the journal's editing and quality control services would also become more apparent to readers who found the article in a repository; these are the services publishers will need to promote in the event that the traditional funding model becomes unsustainable[125].

8.2.4 Society

A learned society is concerned with promotion of academic discourse within their discipline, which can include the sharing of research through a journal. Improved access to research serves this goal, but may conflict with the typical funding sources of non-profit societies[187]. As with commercial journals, alternative funding models may ensure their survival by charging authors for Open Access publication[188].

The existence of a rich graph of linked data connecting publications in their field may be a vital resource to societies. A part of the service they provide to members could become organizing and highlighting the significant work, building a community of practice and social network to encourage communication and collaboration.

8.2.5 Library

Libraries are often responsible for institutional repositories, partly because the skills, infrastructure, and communication links with academics already exist[189]. As experts in the management of disparate information, including long term preservation, metadata design, and policy, librarians are likely to appreciate the virtue of more rigorous metadata; they are able to define suitable metadata structures and assist submitters to use it appropriately.

Libraries have access to existing resources which can be applied to improve the user experience. Digital resources under library stewardship can be the source and target of links to the repository based on the improved metadata. Such resources can also be used to feed the user interface, to provide autocompletion lists when entering metadata, or as supplementary information when viewing a record.

8.2.6 Conference

Like the World Wide Web Conference in 2006[180], a conference is a producer of a large amount of metadata. In this scenario authors, delegates, and readers benefit from a rich graph of linked data. The conference produces a description of its sessions for repositories to use for data capture; this data, as RDF, could link the individual sessions to the main conference, providing a subject hierarchy. The same data can be formatted for social networking services and calendar software to assist delegates and build a community.

8.2.7 Virtual Research Community

A VRC is a group of researchers who rely on communications technology to work together despite physically being apart[190]. A Virtual Research Environment is the toolset which facilitates this. The software plays a part throughout the research process, including managing the sources of data, raw experimental results, analysis, collaboration, and publishing.

With digital publishing, experimental data can be published in machine-readable form alongside an article. In the Semantic Web this can include an ontology to define the meaning of the data, and compatibility with other data sources. The metadata describing these resources will also be different, potentially including a range of first class entities and their own links and metadata. Incorporating these features into repositories will enable closer integration with VREs, allowing the data behind a publication to be analyzed by the reader.

8.2.8 Author

The user interface enhancements discussed here will encourage authors to submit their work to their institutional repository and to provide more complete metadata. Besides saving typing and ensuring a consistent formatting, the abil-

ity for the repository to link to external data will demonstrate the benefits for readers.

Authors seeking recognition for their research output gain from improvements in access. Data exchange through Semantic Web protocols will support new methods for discovering relevant work, for instance through cross-repository browsing, distributed search, and recommendation algorithms.

A Semantic Web repository, or the associated resources described in this section, has the potential to be a source of data for knowledge-aware authoring. WiCK[100, 191] is an example; the project has produced tools which combine a standard document editing environment with a knowledge base to assist in preparing structured documents. With identifiers and linked resources obtained from the repository, this process would be more reliable and would provide more data.

A logical extension of this would be formal descriptions of the concepts of scientific discourse—not just the bibliographic metadata but the conclusions and arguments of the work itself. The ScholOnto project also advocates an extension of the repository interface, but for gathering metadata about the claims made in the work[94]. This is a related initiative to the one I advocate, but one which depends far more heavily on author effort to produce high quality data and which also has a high 'critical mass' of use required to become useful. I suggest that ScholOnto would be more likely to succeed on a large scale once authors and repositories are already familiar with the idea that Semantic Web metadata adds value to their work; if could also be successful as a third party service, providing a hypertext style exploration of concepts based on expert analysis of established work for the benefit of students new to the field.

8.2.9 Third party services

Anywhere an entity is linked or referred to, identifiers can both disambiguate and provide metadata through resolution. Several types of third party service could interoperate with a semantic repository:

Aggregation services could reduce the effort of keeping track and harvesting data. For the hypothesised conference metadata, a useful service would be combining all such data from conferences in a subject area, ensuring a consistent format, and republishing for repository use. The Names project[186] aims to provide identifiers for UK researchers; this or another provider could combine

names with any associated metadata available from institutions, linking the person with related resources, and provide this as a service for repositories to further improve their data capture and value.

CiteBase[133] uses repository data to perform searching of Open Access material. Its information is obtained through OAI-PMH and parsed to interpret the citations, which are used to rank search results. A similar service could interact with the interface of a repository to capture and link citations. Harvested papers, with their URIs (CiteBase uses OAI identifiers, which are suitable), could be made available to the repository for autocomplete at the input stage making it easier to enter citation data. Once submitted, the new paper would itself be harvested along with its citations and added to the index.

8.2.10 Reader

Researchers and research students are the ultimate beneficiary of improved repository access and quality. To them, however, these improvements are intended to be behind the scenes. For the convenience of developers the Semantic Web uses human-readable identifiers and protocols, but ordinary Web users are not expected to interact directly with them. Their experience will consist of more and better links, more expressive browsing and searching, and new ways of analyzing the research in their field. This progress will, it is hoped, lead to better use being made of the research being produced by our global society; and hence as Vannevar Bush imagined, an improvement in the progress of science.

9 Conclusions and future work

This thesis began with the observations of Vannevar Bush, that scientists were overwhelmed by the volume of material available to them, but that technology could play a part in effectively making use of it. Though anticipating many key improvements, he did not predict possibly the most significant: global communication over the Internet.

The Web is a piece of technology which has transformed the process of research. Publication on the Web allows scientists to immediately share their work worldwide and read and build upon the work of others. To maximize the impact of a piece of work, it must be accessible to all but it must also be discovered and read. Associations between resources, including hypertext linking, citation relationships, and metadata, are ways by which the body of scientific publications may be explored and analyzed, thus drawing attention to the most valuable work. Automatic tools which navigate the Web are a key part of this process, and the fact that Web content is primarily intended for human readers means that the limits of the Web's potential have not yet been reached.

The Semantic Web is the technology which facilitates machine processing of Web data. It supports automation of the repetitive aspects of scientific research, permitting the researcher to focus on high level abstract and creative work. The essence of the Semantic Web is formal description of structured and linked data which allows it to be reliably interpreted by software; an important aspect is the use of identifiers to unambiguously refer to resources.

Institutional repositories represent a commitment to preserve and distribute scholarly material for the benefit of authors, readers, institutions, and the academic community as a whole. The effort involved in producing such a resource is significant, and must lead to the greatest possible benefit for the participants. It is therefore important that the most useful data is collected, to best increase accessibility and the value of the research. Repositories are an area of synergy between the communication needs of scholars and the technology of the Semantic Web.

I have built and investigated repositories holding publication metadata, and concluded that the process of collection can lead to valuable information being discarded, leaving ambiguous data that is less suitable for automatic processing. Standardized identifiers are a way to resolve ambiguity in the distributed

environment of the Web, allowing systems using Semantic Web technology to add value to data in a repository through building a graph of linked entities.

Using EPrints as a base, I have explored how repository data with identifiers could be collected and used by combining a repository with a triplestore. Data in this form can provide improvements to browsing and discovery of related records, and can be linked with other sources of data for analysis and exploration through interoperable Semantic Web standards.

Finally, I have described a modification of the EPrints data model to natively support URIs for named entities. Were this model adopted by the repository community, it would provide a critical mass of support for third party identifier services related to the entities related to repository records. This would lead to standardized services which would simplify the process of data collection and automatically join new data with the global graph of linked scholarly discourse.

9.1 Research challenges

In the scenario I propose, what research remains to be done? From a technical point of view, compatibility and data exchange with other repository applications is necessary. Fedora's RDF based data model can already store data in this structure[192]. DSpace is currently limited to Dublin Core for the metadata used by its interface[193], but is capable of storing other types and could provide this feature as an extension. OAI-PMH is extensible to other metadata schemas[134] and could support a modification of the `oai_dc` namespace which included identifiers.

The way linked data is presented to users is an ongoing area of research, and how it relates to repository interfaces should be addressed. Standalone interfaces such as Tabulator[39] are generic and allow direct traversal of the data graph. However, as I have demonstrated such an interface is not necessarily the most efficient way to explore the graph of scholarly discourse. The query functionality of my sidebar tool is an example of a specialized interface; with reliable URI-based links between scholarly work more advanced exploratory, visualization, or recommendation systems could be produced. Faceted browsing is a promising example; with standardization a Longwell[109] or mSpace[41] could be applied to the entire corpus. With the availability of advanced interfaces, there is also the question of how they should integrate with the repository. The repository could import and manage external data within its interface, with each repository

providing an appropriate frontend; or the repository could be a source to be harvested and integrated by external services.

Identifiers in earlier examples have been obtained from the ECS URI service[173]. Such services will be important in a larger scale scenario, because the goal is to use a common identifier where possible across the entire community. The Names project[186] is an effort to build an authority on names for UK researchers, and I have envisaged similar databases of journals, publishers, conferences, and research projects. Certain disciplines might particularly benefit from services in their field, for instance a list of substances in the physical sciences.

As many of these services will be similar, a common architecture could be beneficial. The way these services will be used also suggests that a protocol will be required. In EPrints 3, typing a partial name into an input field performs a lookup on the repository and returns a list of matches. Identifier services will need to provide this functionality with low latency, possibly over a distributed dataset, and may need to consider privacy and authentication issues for personal data. Repository administrators will also need a mechanism for discovering which identifier services are useful for their users. Longer term, coreference management may be required to ensure data integrity as the provision of these services changes.

In the long term, the process of authoring may integrate with repositories and knowledge services at a deep level. WiCK[191] is an example of how writing can integrate with a Semantic Web data source, and ScholOnto[94] shows how a document can be augmented by a formal model of the objects and relationships it describes. Modern document formats and OAI-ORE could extend these detailed semantic relationships with the repository. It is possible that a repository could analyze a document for the connections made by smart authoring tools when it was written, and extract them as metadata to link it to related documents in the worldwide linked structure of scientific discourse.

References

[1] V. Bush, "As We May Think," *The Atlantic Monthly*, vol. 176, pp. 101–108, July 1945. Available at http://www.ps.uni-sb.de/~duchier/pub/vbush/vbush-all.shtml.

[2] W. Boyd Rayward, "Visions of Xanadu: Paul Otlet (1868–1944) and Hypertext," *Journal of the American Society for Information Science*, vol. 45, no. 4, pp. 235–250, 1994. Available at http://dx.doi.org/10.1002/(SICI)1097-4571(199405)45:4<235::AID-ASI2>3.0.CO;2-Y.

[3] Alex Wright, "Forgotten Forefather: Paul Otlet," *Boxes and Arrows*, Nov. 2003. Available at http://www.boxesandarrows.com/view/forgotten_forefather_paul_otlet.

[4] P. Otlet and W. B. Rayward, "Introduction," in *International organisation and dissemination of knowledge: selected essays of Paul Otlet*, Elsevier, 1990. Available at https://www.ideals.uiuc.edu/handle/2142/8645.

[5] T. H. Nelson, "Complex information processing: a file structure for the complex, the changing and the indeterminate," in *Proceedings of the 20th ACM national conference*, (Cleveland, Ohio, United States), pp. 84–100, ACM Press, Aug. 1965. Available at http://doi.acm.org/10.1145/800197.806036.

[6] T. H. Nelson, *Literary Machines*. Mindful Press, 1982.

[7] T. H. Nelson and T. Logan, "Visionary lays into the web," *Go Digital, BBC News Online*, 2001. Available at http://news.bbc.co.uk/2/hi/science/nature/1581891.stm.

[8] T. H. Nelson, "Xanalogical Structure, Needed Now More than Ever: Parallel Documents, Deep Links to Content, Deep Versioning and Deep Re-Use." Available at http://www.xanadu.com.au/ted/XUsurvey/xuDation.html.

[9] D. Engelbart, "A conceptual framework for the augmentation of man's intellect," in *Computer-supported cooperative work: a book of readings*, pp. 35–65, San Francisco, CA, United States: Morgan Kaufmann Publishers Inc., 1988.

[10] R. M. Akscyn, D. L. McCracken, and E. A. Yoder, "KMS: a distributed hypermedia system for managing knowledge in organizations," *Commu-*

nications of the ACM, vol. 31, no. 7, pp. 820–835, 1988. Available at http://doi.acm.org/10.1145/48511.48513.

[11] T. Smith and S. Bernhardt, "Expectations and experiences with HyperCard: a pilot study," in *Proceedings of the 6th annual international conference on Systems documentation*, (Ann Arbor, Michigan, United States), pp. 47–56, 1988. Available at http://doi.acm.org/10.1145/358922.358931.

[12] F. G. Halasz, T. P. Moran, and R. H. Trigg, "Notecards in a nutshell," in *CHI '87: Proceedings of the SIGCHI/GI conference on Human factors in computing systems and graphics interface*, (Toronto, Ontario, Canada), pp. 45–52, 1987. Available at http://doi.acm.org/10.1145/29933.30859.

[13] R. H. Trigg, *A Network-Based Approach to Text Handling for the Online Scientific Community*. PhD Thesis, University of Maryland, Oct. 2002. Available at http://www.workpractice.com/trigg/thesis-default.html.

[14] B. Shneiderman, "User interface design for the Hyperties electronic encyclopedia," in *Proceedings of the ACM conference on Hypertext*, (Chapel Hill, North Carolina, United States), pp. 189–194, ACM Press, 1987. Available at http://doi.acm.org/10.1145/317426.317441.

[15] A. Pearl, "Sun's Link Service: A Protocol for Open Linking," in *Proceedings of the second ACM conference on Hypertext*, (Pittsburgh, Pennsylvania, United States), pp. 137–146, ACM Press, 1989. Available at http://doi.acm.org/10.1145/74224.74236.

[16] N. Yankelovich, B. J. Haan, N. K. Meyrowitz, and S. M. Drucker, "Intermedia: The Concept and the Construction of a Seamless Information Environment," *IEEE Computer*, vol. 21, Jan. 1988. Available at http://doi.ieeecomputersociety.org/10.1109/2.222120.

[17] A. M. Fountain, W. Hall, I. Heath, and H. Davis, "Microcosm: an open model for hypermedia with dynamic linking," in *Hypertext: concepts, systems and applications*, pp. 298–311, Cambridge University Press, 1992.

[18] C. F. Goldfarb, "HyTime: A Standard for Structured Hypermedia Interchange," *IEEE Computer*, vol. 24, Aug. 1991. Available at http://doi.ieeecomputersociety.org/10.1109/2.84880.

[19] F. Halasz and M. Schwartz, "The Dexter hypertext reference model," *Communications of the ACM*, vol. 37, no. 2, pp. 30–39, 1994. Available at http://doi.acm.org/10.1145/175235.175237.

[20] C. W. Thompson, "Strawman Reference Model for Hypermedia Systems," in *Proceedings of the Hypertext Standardization Workshop*, (Gaithersburg, Maryland, United States), pp. 223–248, Jan. 1990. Available at http://eric.ed.gov/ERICWebPortal/detail?accno=ED345690.

[21] H. Davis, A. Lewis, and A. Rizk, "OHP: A Draft Proposal for a Standard Open Hypermedia Protocol," in *Proceedings of the 2nd International Workshop on Open Hypermedia Systems*, (Washington D.C., United States), 1996. Available at http://users.ecs.soton.ac.uk/hcd/protweb.htm.

[22] D. Millard, *Hypermedia Interoperability: Navigating the Information Continuum*. PhD Thesis, University of Southampton, Dec. 2000. Available at http://eprints.ecs.soton.ac.uk/9234/.

[23] D. Fensel and M. A. Musen, "The Semantic Web: A Brain for Humankind," *IEEE Intelligent Systems*, vol. 16, pp. 24–25, Mar. 2001. Available at http://doi.ieeecomputersociety.org/10.1109/MIS.2001.920595.

[24] T. Berners-Lee, "Information Management: A Proposal," tech. rep., CERN, 1989. Available at http://www.w3.org/History/1989/proposal.html.

[25] T. Berners-Lee and R. Cailliau, "WorldWideWeb: Proposal for a HyperText Project," tech. rep., CERN, Nov. 1990. Available at http://www.w3.org/Proposal.html.

[26] M. Bieber, F. Vitali, H. Ashman, V. Balasubramanian, and H. Oinas-Kukkonen, "Fourth Generation Hypermedia: Some Missing Links for the World Wide Web," *International Journal on Human Computer Studies*, vol. 47, pp. 31–65, 1997. Available at http://ijhcs.open.ac.uk/bieber/bieber-nf.html.

[27] B. C. Ladd, M. V. Capps, and P. D. Stotts, "The World Wide Web: what cost simplicity?," in *Proceedings of the eighth ACM Conference on Hypertext*, (Southampton, United Kingdom), pp. 210–211, ACM Press, 1997. Available at http://doi.acm.org/10.1145/267437.267461.

[28] K. Andrews, F. Kappe, and H. Maurer, "Hyper-G and Harmony: towards the next generation of networked information technology," in *Proceedings of CHI '95: Conference companion on Human factors in computing systems*, (Denver, Colorado, United States), pp. 33–34, ACM Press, 1995. Available at http://doi.acm.org/10.1145/223355.223412.

[29] M. Bernstein, "Hypertext gardens," 1998. http://www.eastgate.com/garden/.

[30] F. Kappe, K. Andrews, J. Faschingbauer, M. Gaisbauer, M. Pichler, and J. Schipflinger, "Hyper-G: A New Tool for Distributed Hypermedia." Available at http://ftp.iicm.tugraz.at/pub/papers/report388.pdf.

[31] L. Carr, D. De Roure, W. Hall, and G. Hill, "The Distributed Link Service: A Tool for Publishers, Authors and Readers," in *Proceedings of Fourth International World Wide Web Conference*, (Boston, Massachusetts, United States), pp. 647–656, Dec. 1995. Available at http://eprints.ecs.soton.ac.uk/861/.

[32] L. Carr, S. Bechhofer, C. Goble, and W. Hall, "Conceptual Linking: Ontology-based Open Hypermedia," in *Proceedings of Tenth International World Wide Web Conference*, (Hong Kong), pp. 334–342, May 2001. Available at http://www10.org/cdrom/papers/246/index.html.

[33] K. Grønbæk and R. H. Trigg, "Design issues for a Dexter-based hypermedia system," *Communications of the ACM*, vol. 37, no. 2, pp. 40–49, 1994. Available at http://doi.acm.org/10.1145/175235.175238.

[34] "XML Linking Language (XLink) Version 1.0," W3C Recommendation, World Wide Web Consortium, June 2001. Available at http://www.w3.org/TR/xlink.

[35] "Scalable Vector Graphics (SVG) 1.1 Specification," W3C Recommendation, World Wide Web Consortium, Jan. 2003. Available at http://www.w3.org/TR/SVG11/.

[36] "Wiki Wiki Web." http://c2.com/cgi/wiki?WikiWikiWeb.

[37] "Wikipedia, the free encyclopedia." http://en.wikipedia.org/.

[38] "HTML 4.01 Specification," W3C Recommendation, World Wide Web Consortium, Dec. 1999. Available at http://www.w3.org/TR/html4.

[39] T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets, "Tabulator: Exploring and Analyzing linked data on the Semantic Web," in *Proceedings of the 3rd International Semantic Web User Interaction Workshop*, (Athens, Georgia, United States), Nov. 2006. Available at `http://swui.semanticweb.org/swui06/papers/Berners-Lee/Berners-Lee.pdf`.

[40] "SIMILE Project." `http://simile.mit.edu/`.

[41] C. Harris, A. Owens, A. Russel, and D. A. Smith, "mSpace: Exploring The Semantic Web. A Technical Report in Support of the mSpace software framework," tech. rep., University of Southampton. Available at `http://eprints.ecs.soton.ac.uk/10359/`.

[42] N. Shadbolt, N. Gibbins, H. Glaser, S. Harris, and m. schraefel, "CS AKTive Space or how we stopped worrying and learned to love the Semantic Web," *IEEE Intelligent Systems*, vol. 19, pp. 41–47, May 2004. Available at `http://eprints.ecs.soton.ac.uk/8817/`.

[43] "Fresnel—Display Vocabulary for RDF," Technical report, World Wide Web Consortium, Apr. 2005. Available at `http://www.w3.org/2005/04/fresnel-info/`.

[44] J. Nanard and M. Nanard, "Hypertext design environments and the hypertext design process," *Communications of the ACM*, vol. 38, no. 8, pp. 49–56, 1995. Available at `http://doi.acm.org/10.1145/208344.208347`.

[45] F. Anklesaria, M. McCahill, P. Lindner, D. Johnson, D. Torrey, and B. Alberti, "RFC 1436: The Internet Gopher Protocol," Request for Comments, The Internet Society, Mar. 1993. Available at `http://www.ietf.org/rfc/rfc1436.txt`.

[46] Free Software Foundation, *GNU Texinfo Manual*, Apr. 2008. Available at `http://www.gnu.org/software/texinfo/manual/texinfo/`.

[47] "XML Path Language (XPath) Version 1.0," W3C Recommendation, World Wide Web Consortium, Nov. 1999. Available at `http://www.w3.org/TR/xpath`.

[48] "XML Pointe Language (XPointer)," W3C Working Draft, World Wide Web Consortium, Aug. 2002. Available at `http://www.w3.org/TR/xptr/`.

[49] S. Hitchcock, L. Carr, W. Hall, S. Harris, S. Probets, D. Evans, and D. Brailsford, "Linking Electronic Journals: Lessons from the Open Journal Project," *D-Lib Magazine*, vol. 4, Dec. 1998. Available at `http://www.dlib.org/dlib/december98/12hitchcock.html`.

[50] D. E. Millard, D. C. De Roure, D. T. Michaelides, M. K. Thompson, and M. J. Weal, "Navigaional Hypertext Models For Physical Hypermedia Environments," in *Proceedings of the fifteenth ACM conference on Hypertext and Hypermedia*, (Santa Cruz, California, United States), pp. 110–111, ACM Press, Aug. 2004. Available at `http://doi.acm.org/10.1145/1012807.1012839`.

[51] M. Chalmers, B. Brown, S. Benford, R. Conroy, N. Dalton, A. Galani, C. Greenhalgh, I. MacColl, D. Michaelides, D. Millard, C. Randell, A. Steed, T. Rodden, I. Taylor, and M. Weal, "Blurring the Boundaries of the Mackintosh Room," in *CHI 2002: Proceedings of the International Conference on Human Factors in Computing Systems*, (Minneapolis, Minnesota, United States), Apr. 2002. Available at `http://eprints.ucl.ac.uk/1100/`.

[52] M. Weal, D. Michaelides, M. Thompson, and D. De Roure, "Hypermedia in the Ambient Wood," *New Review of Hypermedia and Multimedia*, vol. 9, pp. 137–156, Jan. 2003. Available at `http://eprints.ecs.soton.ac.uk/9514/`.

[53] T. Brody, S. Kampa, S. Harnad, L. Carr, and S. Hitchcock, "Digitometric Services for Open Access Environments," in *Proceedings of European Conference on Digital Libraries*, (Trondheim, Norway), pp. 207–220, 2003. Available at `http://eprints.ecs.soton.ac.uk/7503/`.

[54] Google Inc., "Google Technology: PageRank Explained." `http://www.google.com/technology/`.

[55] Stevan Harnad, "Open Access Scientometrics and the UK Research Assessment Exercise," in *Proceedings of the 11th Annual Meeting of the International Society for Scientometrics and Informetrics*, (Madrid, Spain), June 2007. Available at `http://eprints.ecs.soton.ac.uk/13804`.

[56] T. Berners-Lee, "Semantic Web roadmap," Technical report, World Wide Web Consortium, Sept. 1998. Available at `http://www.w3.org/DesignIssues/Semantic.html`.

[57] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, May 2001. Available at http://www.sciam.com/article.cfm?id=the-semantic-web.

[58] W3C, "Semantic Web 'layer cake'." http://www.w3.org/2007/03/layerCake.png.

[59] S. Luke, L. Spector, and D. Rager, "Ontology-Based Knowledge Discovery on the World Wide Web," in *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, (Portland, Oregon, United States), 1996. Available at http://www.cs.umd.edu/projects/plus/SHOE/pubs/aaai-paper.html.

[60] "PICS Label Distribution Label Syntax and Communication Protocols," W3C Recommendation, World Wide Web Consortium, Oct. 1996. Available at http://www.w3.org/TR/REC-PICS-labels.

[61] "Rating Services and Rating Systems (and Their Machine Readable Descriptions)," W3C Recommendation, World Wide Web Consortium, Oct. 1996. Available at http://www.w3.org/TR/REC-PICS-services.

[62] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, "RFC 2616: Hypertext Transfer Protocol—HTTP/1.1," Request for Comments, The Internet Society, June 1999. Available at http://www.ietf.org/rfc/rfc2616.txt.

[63] "What is Unicode?." http://www.unicode.org/standard/WhatIsUnicode.html.

[64] A. M. Costello, "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications," Request for Comments, The Internet Society, Mar. 2003. Available at http://www.ietf.org/rfc/rfc3492.txt.

[65] M. Düerst and M. Suignard, "Internationalized Resource Identifiers (IRIs)," Request for Comments, The Internet Society, Jan. 2005. Available at http://www.ietf.org/rfc/rfc3987.txt.

[66] T. Berners-Lee, "RFC 1630: Universal Resource Identifiers in WWW," Request for Comments, The Internet Society, June 1994. Available at http://www.ietf.org/rfc/rfc1630.txt.

[67] T. Berners-Lee, "The Original HTTP as defined in 1991," tech. rep., World Wide Web Consortium, 1991. Available at http://www.w3.org/Protocols/HTTP/AsImplemented.html.

[68] "RDF Primer," W3C Recommendation, World Wide Web Consortium, Feb. 2004. Available at http://www.w3.org/TR/rdf-primer/.

[69] "Extensible Markup Language (XML)," W3C Recommendation, World Wide Web Consortium, Feb. 2004. Available at http://www.w3.org/TR/REC-xml.

[70] "ISO 8879:1986 Standard Generalized Markup Language (SGML)," ISO Standard, ISO, Oct. 1986.

[71] "XHTML 1.0 The Extensible HyperText Markup Language (Second Edition)," W3C Recommendation, World Wide Web Consortium, Aug. 2002. Available at http://www.w3.org/TR/xhtml1.

[72] "XML Schema Primer," W3C Recommendation, World Wide Web Consortium, Oct. 2004. Available at http://www.w3.org/TR/xmlschema-0/.

[73] "RELAX NG Specification," Committee Specification, OASIS, Dec. 2001. Available at http://relaxng.org/spec-20011203.html.

[74] "RDF/XML Syntax Specification (Revised)," W3C Recommendation, World Wide Web Consortium, Feb. 2004. Available at http://www.w3.org/TR/rdf-syntax-grammar/.

[75] Dave Beckett, "Turtle—Terse RDF Triple Language," tech. rep., Nov. 2007. Available at http://www.dajobe.org/2004/01/turtle/.

[76] "RDF Vocabulary Description Language 1.0: RDF Schema," W3C Recommendation, World Wide Web Consortium, Feb. 2004. Available at http://www.w3.org/TR/rdf-schema/.

[77] Thomas R. Gruber, "A Translation Approach to Portable Ontology Specifications," *Knowledge Acquisition*, vol. 5, June 1993. Available at http://tomgruber.org/writing/ontolingua-kaj-1993.htm.

[78] "OWL Web Ontology Language Overview," W3C Recommendation, World Wide Web Consortium, Feb. 2004. Available at http://www.w3.org/TR/owl-features/.

[79] D. Brickley and L. Miller, "FOAF Vocabulary Specification," tech. rep., Sept. 2004. Available at http://xmlns.com/foaf/0.1/.

[80] "DCMI Metadata Terms," DCMI Recommendation, Dublin Core Metadata Initiative, Sept. 2004. Available at http://dublincore.org/documents/dcmi-terms/.

[81] Stevan Harnad, "The Symbol Grounding Problem," *Physica D: Nonlinear Phenomena*, vol. 42, pp. 335–346, 1990. Available at http://cogprints.org/3106/.

[82] D. Beckett, "The Design and Implementation of the Redland RDF Application Framework," in *Proceedings of the 10th International World Wide Web Conference*, (Hong Kong), May 2001. Available at http://www10.org/cdrom/papers/490/index.html.

[83] S. Harris, "SPARQL query processing with conventional relational database systems," in *Proceedings of the International Workshop on Scalable Semantic Web Knowledge base Systems*, 2005. Available at http://eprints.ecs.soton.ac.uk/11126.

[84] Brian McBride, "Jena: Implementing the RDF Model and Syntax Specification," in *Proceedings of the 10th International World Wide Web Conference*, (Hong Kong), May 2001. Available at http://www-uk.hpl.hp.com/people/bwm/papers/20001221-paper/.

[85] Kevin Wilkinson and Craig Sayers and Harumi A. Kuno and Dave Reynolds, "Efficient RDF Storage and Retrieval in Jena2," in *Proceedings of the First International Workshop on Semantic Web and Databases*, (Berlin, Germany), Sept. 2003. Available at http://www-scf.usc.edu/~csci586/paper/Wilkinson_etal.pdf.

[86] Jeen Broekstra and Arjohn Kampman and Frank van Harmelen, "Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema," in *Proceedings of the First International Semantic Web Conference*, (Sardinia, Italy), Springer-Verlag, June 2002. Available at http://www.cs.vu.nl/~frankh/postscript/ISWC02.pdf.

[87] "SPARQL Query Language for RDF," W3C Recommendation, World Wide Web Consortium, Jan. 2008. Available at http://www.w3.org/TR/rdf-sparql-query/.

[88] P. Haase, J. Broekstra, A. Eberhart, and R. Volz, "A comparison of RDF query languages," in *Proceedings of the Third International Semantic Web Conference*, (Hiroshima, Japan), Springer-

Verlag, Nov. 2004. Available at http://www.aifb.uni-karlsruhe.de/Publikationen/showPublikation?publ_id=522.

[89] "RDF Data Access WG Charter," Dec. 2003. Available at http://www.w3.org/2003/12/swa/dawg-charter.

[90] "SPARQL Protocol for RDF," W3C Recommendation, World Wide Web Consortium, Jan. 2008. Available at http://www.w3.org/TR/rdf-sparql-protocol/.

[91] J. Kahan, M.-R. Koivunen, E. Prud'Hommeaux, and R. R. Swick, "Annotea: An Open RDF Infrastructure for Shared Web Annotations," in *Proceedings of Tenth International World Wide Web Conference*, (Hong Kong), pp. 623–632, May 2001. Available at http://www10.org/cdrom/papers/488/.

[92] "Amaya Home Page." http://www.w3.org/Amaya/.

[93] Marja-Riitta Koivunen, "Annotea and Semantic Web Supported Collaboration," in *Workshop on User Aspects of the Semantic Web, ESWC 2005*, (Heraklion, Crete, Greece), May 2005. Available at http://www.annotea.org/eswc2005/01_koivunen_final.pdf.

[94] S. Buckingham Shum, E. Motta, and J. Domingue, "ScholOnto: an ontology-based digital library server for research documents and discourse," *International Journal on Digital Libraries*, vol. 3, pp. 237–248, Oct. 2000. Available at http://kmi.open.ac.uk/projects/scholonto/docs/ScholOnto-IJoDL-2000.pdf.

[95] S. Buckingham Shum, V. Uren, G. Li, B. Sereno, and C. Mancini, "Modelling Naturalistic Argumentation in Research Literatures: Representation and Interaction Design Issues," *International Journal of Intelligent Systems*, vol. 22, no. 1, pp. 17–47, 2007. Available at http://kmi.open.ac.uk/publications/pdf/kmi-04-28.pdf.

[96] B. Sereno, S. Buckingham Shum, and E. Motta, "ClaimSpotter: an Environment to Support Sensemaking with Knowledge Triples," in *Proceedings of the ACM International Conference on Intelligent User Interfaces*, (San Diego, California, United States), ACM Press, Jan. 2005. Available at http://kmi.open.ac.uk/publications/pdf/kmi-04-29.pdf.

[97] B. Sereno, S. Buckingham Shum, and E. Motta, "Formalization, User Strategy and Interaction Design: Users' Behaviour with Dis-

course Tagging Semantics," in *Workshop on Social and Collaborative Construction of Structured Knowledge, WWW2007*, (Banff, Canada), May 2007. Available at `http://kmi.open.ac.uk/projects/hyperdiscourse/docs/WWW_CKC2007_Sereno_Final.pdf`.

[98] J. Domingue, M. Dzbor, and E. Motta, "Collaborative Semantic Web Browsing with Magpie," in *Proceedings of the First European Semantic Web Symposium (ESWS)*, (Hersonissos, Crete, Greece), 2004. Available at `http://kmi.open.ac.uk/people/dzbor/public/2004/ESWS-domingue-dzbor-motta-final.pdf`.

[99] B. Popov, A. Kiryakov, A. Kirilov, D. Manov, D. Ognyanoff, and M. Goranov, "KIM—Semantic Annotation Platform," in *Proceedings of the second International Semantic Web Conference, ISWC 2003*, vol. 2870 of *Lecture Notes in Computer Science*, (Sanibel, Florida, United States), pp. 834–849, Springer-Verlag, Oct. 2003. Available at `http://www.springerlink.com/link.asp?id=31l2tk95lcdv76py`.

[100] L. Carr, T. Miles-Board, G. Wills, A. Woukeu, and W. Hall, "Towards a Knowledge-Aware Office Environment," in *Proceedings of the 5th International Conference on Practical Aspects of Knowledge Management (PAKM 2004), LNAI 3336*, (Vienna, Austria), pp. 129–140, ACM Press, 2004. Available at `http://eprints.ecs.soton.ac.uk/9847/`.

[101] M. Dzbor, E. Motta, and J. Domingue, "Opening Up Magpie via Semantic Services," in *Proceedings of the third International Semantic Web Conference, ISWC 2004*, vol. 3298 of *Lecture Notes in Computer Science*, (Hiroshima, Japan), pp. 635–649, Springer-Verlag, Nov. 2004. Available at `http://www.springerlink.com/link.asp?id=n1hyrvet4jvql95d`.

[102] A. Dingli, F. Ciravegna, and Y. Wilks, "Automatic Semantic Annotation Using Unsupervised Information Extraction and Integration," in *Proceedings of the K-CAP 2003 Workshop on Knowledge Markup and Semantic Annotation*, (Sanibel, Florida, United States), Oct. 2003. Available at `http://ceur-ws.org/Vol-101/Alexiei_Dingli-et-al.pdf`.

[103] H. Alani, S. Dasmahapatra, K. O'Hara, and N. Shadbolt, "Identifying Communities of Practice through Ontology Network Analysis," *IEEE Intelligent Systems*, vol. 18, pp. 18–25, Mar. 2003. Available at `http://eprints.ecs.soton.ac.uk/7397/`.

[104] AKT Project, "AKT Reference Ontology." Available at http://www.aktors.org/publications/ontology/.

[105] "ISWC2006 Core Metadata Set." http://iswc2006.semanticweb.org/program/tech_links.php.

[106] "Flink." http://flink.semanticweb.org/.

[107] M. Smith, M. Bass, G. McClellan, R. Tansley, M. Barton, M. Branschofsky, D. Stuve, and J. H. Walker, "DSpace: An Open Source Dynamic Digital Repository," *D-Lib Magazine*, vol. 9, Jan. 2003. Available at http://dlib.org/dlib/january03/smith/01smith.html.

[108] "Welkin." http://simile.mit.edu/welkin/.

[109] "Longwell." http://simile.mit.edu/longwell/.

[110] D. Huynh, S. Mazzocchi, and D. Karger, "Piggy Bank: Experience the Semantic Web Inside Your Web Browser," in *Proceedings of the fourth International Semantic Web Conference, ISWC 2005*, vol. 3729 of *Lecture Notes in Computer Science*, (Galway, Ireland), pp. 413–430, Springer-Verlag, Nov. 2005. Available at http://hdl.handle.net/1721.1/29466.

[111] D. A. Quan, *Designing end user information environments built on semistructured data models*. PhD Thesis, Massachusetts Institute of Technology, 2003. Available at http://hdl.handle.net/1721.1/29750.

[112] D. Smith, A. Owens, m. schraefel, P. Sinclair, P. André, M. Wilson, A. Russell, K. Martinez, and P. Lewis, "Challenges in Supporting Faceted Semantic Browsing of Multimedia Collections," in *Proceedings of the Second International Conference on Semantic and Digital Media Technologies*, (Genova, Italy), Dec. 2007. Available at http://eprints.ecs.soton.ac.uk/14507.

[113] R. Crow, "The Case for Institutional Repositories: A SPARC Position Paper," tech. rep., Scholarly Publishing and Academic Resources Coalition, 2002. Available at http://www.arl.org/sparc/repositories/readings.shtml.

[114] C. A. Lynch, "Institutional Repositories: essential infrastructure for scholarship in the digital age," *Association of Research Libraries bimonthly report*, vol. 226, Feb. 2003. Available at http://dspace.uniroma2.it/dspace/handle/2108/261.

[115] "Cogprints Cognitive Science eprint archive." http://cogprints.org/.

[116] "RePEc: Research Papers in Economics." http://repec.org/.

[117] "arXiv.org e-Print archive." http://arxiv.org/.

[118] "Reference Model for an Open Archival Information System (OAIS)," CCSDS Recommendation, Consultative Committee for Space Data Systems, 2002. ISO 14721:2003. Available at http://public.ccsds.org/publications/RefModel.aspx.

[119] J. Bekaert and H. Van de Sompel, "Access Interfaces for Open Archival Information Systems based on the OAI-PMH and the OpenURL Framework for Context-Sensitive Services," in *Proceedings of PV2005*, (Edinburgh, Scotland), Nov. 2005. Available at http://www.ukoln.ac.uk/events/pv-2005/pv-2005-final-papers/032.pdf.

[120] J. Allinson, "OAIS as a reference model for repositories: an evaluation," tech. rep., UKOLN, Nov. 2006. Available at http://www.ukoln.ac.uk/repositories/publications/oais-evaluation-200607/.

[121] S. Harnad, "For Whom the Gate Tolls?," 2001. Available at http://www.ecs.soton.ac.uk/~harnad/Tp/resolution.htm.

[122] S. Harnad, "The Invisible Hand of Peer Review," *Exploit Interactive*, vol. 5, Apr. 2000. Available at http://www.exploit-lib.org/issue5/peer-review/.

[123] A. S. Relman, "Peer Review in Scientific Journals—What Good Is It?," *Western Journal of Medicine*, vol. 153, pp. 520–522, Nov. 1990. Available at http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1002603.

[124] S. Morris, "Learned journals and the communication of research," *Learned Publishing*, vol. 11, pp. 253–258, Oct. 1998. Available at http://www.ingentaconnect.com/content/alpsp/lp/1998/00000011/00000004/art00002.

[125] S. Morris, "Open publishing," *Learned Publishing*, vol. 16, pp. 171–176, July 2003. Available at http://www.ingentaconnect.com/content/alpsp/lp/2003/00000016/00000003/art00003.

[126] "Budapest Open Access Initiative," Dec. 2001. Available at http://www.soros.org/openaccess/read.shtml.

[127] H. V. de Sompel and C. Lagoze, "The Santa Fe Convention of the Open Archives Initiative," *D-Lib Magazine*, vol. 6, Feb. 2000. Available at http://www.dlib.org/dlib/february00/vandesompel-oai/02vandesompel-oai.html.

[128] C. Lagoze and H. V. de Sompel, "The Open Archives Initiative: Building a low-barrier interoperability framework," in *Proceedings of the First ACM/IEEE Joint Conference on Digital Libraries*, (Roanoke, Virginia, United States), pp. 54–62, ACM Press, June 2001. Available at http://www.openarchives.org/documents/jcdl2001-oai.pdf.

[129] H. V. de Sompel, T. Krichel, M. L. Nelson, P. Hochstenbach, V. M. Lyapunov, K. Maly, M. Zubair, M. Kholief, X. Liu, and H. O'Connell, "The UPS Prototype: An Experimental End-User Service accross E-Print Archives," *D-Lib Magazine*, vol. 6, Feb. 2000. Available at http://www.dlib.org/dlib/february00/vandesompel-ups/02vandesompel-ups.html.

[130] "OAIster." http://www.oaister.org/.

[131] "CiteSeer." http://citeseer.ist.psu.edu/.

[132] "Google Scholar." http://scholar.google.co.uk/.

[133] "CiteBase Search." http://citebase.eprints.org/.

[134] "The Open Archives Initiative Protocol for Metadata Harvesting," tech. rep., Open Archives Initiative, June 2002. Available at http://www.openarchives.org/OAI/openarchivesprotocol.html.

[135] "ORE User Guide," tech. rep., Open Archives Initiative, June 2008. Available at http://www.openarchives.org/ore/0.9/primer.

[136] R. Tansley and S. Harnad, "Eprints.org Software for Creating Institutional and Individual Open Archives," *D-Lib Magazine*, vol. 6, Oct. 2000. Available at http://www.dlib.org/dlib/october00/10inbrief.html#HARNAD.

[137] M. Maxwell, "Technical Evaluation of Selected Open Access Repositories in New Zealand," tech. rep., Open Access Repositores in New Zealand, Sept. 2006. Available at https://eduforge.org/docman/view.php/131/1062/Repository%20Evaluation%20Document.pdf.

[138] "EPrints Free Software." Available at http://www.eprints.org/software/.

[139] S. Harnad, "Free at Last: The Future of Peer-Reviewed Journals," *D-Lib Magazine*, vol. 5, Dec. 1999. Available at http://www.dlib.org/dlib/december99/12harnad.html.

[140] S. Sun, L. Lannom, and B. Boesch, "RFC3650: Handle System Overview," Request for Comments, The Internet Society, Nov. 2003. Available at http://www.ietf.org/rfc/rfc3650.txt.

[141] "Fedora Open Source Repository Software," White Paper, Fedora Development Team, Oct. 2005. Available at http://www.fedora-commons.org/pdfs/WhitePaper.10.28.05.pdf.

[142] S. Payette and C. Lagoze, "Flexible and Extensible Digital Object and Repository Architecture," in *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries*, (Heraklion, Crete, Greece), Sept. 1998. Available at http://www.cs.cornell.edu/payette/papers/ECDL98/FEDORA.html.

[143] T. Gill, A. J. Gilliland, and M. S. Woodley, *Introduction to Metadata*. Getty Research Institute, 1998. Available at http://www.getty.edu/research/conducting_research/standards/intrometadata/index.html.

[144] "Southampton Crystal Reports." http://ecrystals.chem.soton.ac.uk/.

[145] "Worldwide Protein Data Bank." http://www.wwpdb.org/.

[146] B. Furrie, *Understanding MARC Bibliographic: Machine-Readable Cataloguing*. Library of Congress, 2003. Available at http://www.loc.gov/marc/umb/.

[147] A. Sale, "Researchers and institutional repositories," in *Open Access: Key Strategic, Technical and Economic Aspects*, pp. 87–100, Chandos Publishing, 2006.

[148] H. Atkins, "The ISI Web of Science - Links and Electronic Journals," *D-Lib Magazine*, vol. 5, Sept. 1999. Available at http://www.dlib.org/dlib/september99/atkins/09atkins.html.

[149] "Fact sheet: Medline," tech. rep., National Library of Medicine, Apr. 2008. Available at http://www.nlm.nih.gov/pubs/factsheets/medline.html.

[150] "Medical subject headings." http://www.nlm.nih.gov/mesh/.

[151] H. Van de Sompel and O. Beit-Arie, "Open Linking in the Scholarly Information Environment Using the OpenURL Framework," *D-Lib Magazine*, vol. 7, Mar. 2001. Available at http://www.dlib.org/dlib/march01/vandesompel/03vandesompel.html.

[152] "Connotea." http://www.connotea.org/.

[153] "Interoperable Repository Statistics." http://trac.eprints.org/projects/irstats.

[154] T. Brody, S. Harnad, and L. Carr, "Earlier Web Usage Statistics as Predictors of Later Citation Impact," *Journal of the American Association for Information Science and Technology (JASIST)*, vol. 57, no. 8, pp. 1060–1072, 2006. Available at http://eprints.ecs.soton.ac.uk/10713/.

[155] "Architecture of the World Wide Web, Volume One," W3C Recommendation, World Wide Web Consortium, Dec. 2004. Available at http://www.w3.org/TR/webarch/.

[156] International DOI Foundation, *DOI Handbook*, Sept. 2007. Available at http://dx.doi.org/10.1000/182.

[157] "Functional Requirements for Bibliographic Records," tech. rep., Internation Federation of Library Associations and Institutions, Sept. 1997. Available at http://www.ifla.org/VII/s13/frbr/frbr.htm.

[158] N. Paskin and L. Lannom, "From one to many," tech. rep., International DOI Foundation, Aug. 2000. Available at http://dx.doi.org/10.1000/190.

[159] A. Apps and R. MacIntyre, "Why OpenURL?," *D-Lib Magazine*, vol. 12, May 2006. Available at http://www.dlib.org/dlib/may06/apps/05apps.html.

[160] "Library of Congress Authorities." http://authorities.loc.gov/.

[161] "Dewey Decimal Classification." http://www.oclc.org/dewey/.

[162] "Library of Congress Classification." http://www.loc.gov/aba/cataloging/classification/.

[163] "ACM Computing Classification Systems." http://www.acm.org/class/.

[164] J. Kunze, "Towards Electronic Persistence Using ARK Identifiers," in *Proceedings of the 3rd EDCL Workshop on Web Archives*, (Trondheim, Norway), Aug. 2003. Available at http://www.cdlib.org/inside/diglib/ark/arkcdl.pdf.

[165] M. J. Bass, D. Stuve, R. Tansley, M. Branschofsky, P. Breton, P. Carmichael, B. Cattey, D. Chudnov, and J. Ng, "DSpace Technology and Architecture," tech. rep., Hewlett-Packard/MIT, Mar. 2002. Available at http://www.dspace.org/technology/architecture.pdf.

[166] D. Booth, "URIs and the Myth of Resource Identity." Available at http://www.dbooth.org/2006/identity/, May 2006.

[167] A. Jaffri, H. Glaser, and I. Millard, "Managing URI Synonymity to Enable Consistent Reference on the Semantic Web," in *Proceedings of ISRW2008: Identity and Reference on the Semantic Web 2008*, (Tenerife, Spain), June 2008. Available at http://eprints.ecs.soton.ac.uk/15614/.

[168] A. Jaffri, H. Glaser, and I. Millard, "URI Identity Management for Semantic Web Data Integration and Linkage," in *3rd International Workshop on Scalable Semantic Web Knowledge Base Systems*, (Vilamoura, Algarve, Portugal), Nov. 2007. Available at http://eprints.ecs.soton.ac.uk/14361/.

[169] H. Alani, S. Dasmahapatra, N. Gibbins, H. Glaser, S. Harris, Y. Kalfoglou, K. O'Hara, and N. Shadbolt, "Managing Reference: Ensuring Referential Integrity of Ontologies for the Semantic Web," in *Proceedings of 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW'02)*, (Sigenza, Spain), pp. 317–334, Sept. 2002. Available at http://eprints.ecs.soton.ac.uk/6649/.

[170] T. Lewy, H. Glaser, and N. Shadbolt, "A Framework for Reference Management in the Semantic Web," tech. rep., University of Southampton, Nov. 2005. Available at http://eprints.ecs.soton.ac.uk/11539/.

[171] T. Berners-Lee, "What do HTTP URIs identify?." Available at http://www.w3.org/DesignIssues/HTTP-URI.html, July 2002.

[172] T. Berners-Lee, "What do HTTP URIs identify?." Available at http://www.w3.org/DesignIssues/HTTP-URI2, June 2005.

[173] "ECS URI System Specification." Available at http://id.ecs.soton.ac.uk/docs/.

[174] "Kultur." http://kultur.eprints.org/about.htm.

[175] "Relator Terms and Dublin Core elements," May 2005. Available at http://lcweb2.loc.gov/cocoon/loc.terms/relators/dc-contributor.html.

[176] "Research Assessment Exercise (RAE) 2001." http://www.hero.ac.uk/rae/.

[177] "A Guide to the 2001 Research Assessment Exercise," tech. rep., HEFCE / SHEFC / HEFCW / DELNI, 2001. Available at http://www.hero.ac.uk/rae/Pubs/other/raeguide.pdf.

[178] "ISO/IEC 8859-1:1998 Information technology—8-bit single-byte coded graphic character sets—Part 1: Latin alphabet No. 1," ISO Standard, ISO, 1998.

[179] M. Jewell, "ParaCite: An Overview," tech. rep., EPrints.org, Dec. 2002. Available at http://paracite.eprints.org/docs/overview.html.

[180] "WWW2006 Conference RDF." Available at http://www2006.org/programme/dynamic/rdf/.

[181] "ECS EPrints Repository." http://eprints.ecs.soton.ac.uk/.

[182] "DSpace at MIT." http://dspace.mit.edu/.

[183] J. A. Konstan, "Introduction To Recommender Systems: Algorithms and Evaluation," *Transactions on Information Systems*, vol. 22, no. 1, 2004. Available at http://doi.acm.org/10.1145/963770.963771.

[184] Emmanuel Pietriga, "IsaViz Overview." Available at http://www.w3.org/2001/11/IsaViz/.

[185] T. Berners-Lee, "Linked Data—Design Issues," Technical report, World Wide Web Consortium, July 2006. Available at http://www.w3.org/DesignIssues/LinkedData.html.

[186] A. Hill, "What's in a Name? Prototyping a Name Authority Service for UK Repositories," in *Proceedings of the 10th International Conference of the International Society for Knowledge Organization (in press)*, (Montréal, Canada), Aug. 2008. Available at http://names.mimas.ac.uk/documents/Names_ISKO2008_paper.pdf.

[187] C. E. Rudder, "Copyright, Libraries, and the Financial Viability of Scholarly Society Journals," *Copyright, Public Policy, and the Scholarly Community*, 1995. Available at http://academy.eserver.org/rudder.html.

[188] M. Waltham, "Learned society business models and open access: overview of a recent JISC-funded study," *Learned Publishing*, vol. 19, pp. 15–30, Jan. 2006. Available at http://dx.doi.org/10.1087/095315106775122529.

[189] L. Horwood, S. Sullivan, E. Young, and J. Garner, "OAI compliant institutional repositories and the role of library staff," *Library Management*, vol. 25, no. 4, pp. 170–176, 2004. Available at http://repository.unimelb.edu.au/10187/1527.

[190] "Report of the Working Group on Virtual Research Communities," OSI e-Infrastructure Report, Office of Science and Innovation. Available at http://www.nesc.ac.uk/documents/OSI/vrc.pdf.

[191] A. Woukeu, L. Carr, and W. Hall, "WiCKEd: A Tool for Writing in the Context of Knowledge," in *Proceedings of the fifteenth ACM conference on Hypertext and Hypermedia*, (Santa Cruz, California, United States), pp. 93–94, ACM Press, Aug. 2004. Available at http://doi.acm.org/10.1145/1012807.1012835.

[192] *Fedora Metadata for Object-to-Object Relationships*, 2005. Available at http://www.fedora.info/download/2.0/userdocs/digitalobjects/introRelsExt.html.

[193] *FAQ: DSpace*. Available at http://www.dspace.org/faqs/index.html#standards.