From Content-centered Publishing to a Link-based View of Information Resources

Erik Wilde (hicss33@dret.org)
Swiss Federal Institute of Technology, Zürich

David Lowe (david.lowe@uts.edu.au)
University of Technology, Sydney

Abstract

Influenced by the linking model which is implicit in HTML, today's publishing model on the Web is content-centered, with the emphasis of publishing on content rather than links. With the growing amount of information available on the Web, and the more powerful hypermedia architectures made possible by new Web technologies, putting the content into context will become increasingly important. In this paper, a new way of structuring publishing systems for information providers is presented in an attempt to shift the emphasis in Web-based publishing from content to an improved balance between content and links. After a description of the architecture of a link-based publishing system, a strategy for implementing such a system is described. Finally, a number of challenges associated with such a fundamental transition in the publishing model are described, in the technical as well as in the organizational domain.

1 Introduction

The World Wide Web is the first hypermedia system that makes content available on a truly large scale, accessible to hundreds of millions of globally distributed users. The first hypermedia visions described by Vannevar Bush [6] in 1945 were technically infeasible at that time, but today distributed hypermedia systems can be easily implemented based on the infrastructure provided by the Internet. Surprisingly, the *Xanadu* system described by Nelson [23, 24] in the 1970's was conceptually more powerful than the Web in its early forms (though Xanadu was never actually implemented). Recent advances in Web technology make it possible to use hypermedia features which go far beyond the support provided in the first simple versions of the Web's *Hypertext Markup Language (HTML)*. This paper mainly deals with a focus shift from content to links between resources, including concepts for external links and automatic inclusion of resources, something which in Xanadu was described from the very beginning using concepts such as *transclusion* and *transpublication* [25].

The Web today still uses an easily comprehensible set of core standards [32], but this core is beginning to become much more powerful (and thus, more complex) with a whole new set of recommendations within the context of the *Extensible Markup Language (XML)* [3]. Most information providers today still focus on the HTML model of information publication, where content, presentation, and structure (achieved through links) are inseparably intertwined. Various hypermedia research systems have already demonstrated the power which can be obtained from more sophisticated linking models than is currently supported on the Web. Note that whilst some Web back-end systems (such as Webcosm [14] and HyperWave [22]) provide a richer linking model, we still have to map this to HTML for delivery, and as such are still constrained by the limitations imposed by HTML. However, XML and its companion standards provide a new way to look at providing information, where content and links associated with the content can be cleanly separated using the *XML Linking Language (XLink)*, making it possible to represent, store, and deliver them separately [33].

In this paper, an approach is described which makes it possible to support and manage on the Web a clean separation between document content and links associated with this content. Both units can then be created, maintained, and distributed separately, and it is our prediction that the overall emphasis on information will make a shift from content to links in the next few years. The concepts introduced in this paper also makes it possible to implement a smooth transition of information publication, moving from the combined content/links model of HTML to the XML/XLink model of a cleaner separation of content and links.

1.1 Traditional Publishing

The publishing model of traditional media (such as newspapers, books, etc.) is very much centered around representing and accessing content. In some cases, references

Copyright 2000 IEEE. Published in the Proceedings of the Hawai'i International Conference On System Sciences, January 4-7, 2000, Maui, Hawaii.

to associated resources are supplied, but generally speaking the publishing process is almost exclusively focused on content. This can easily be explained by the limitations of the linear media traditionally used for publishing. All these media provide support for references (consider, for example, the bibliography in this paper) but do not inherently support automated navigation of these references, and therefore make it hard for an author to create easily usable references, and for a reader to follow references given by authors.

The Web has changed this picture radically, being the first widely accessible system which inherently supports references in the form of *hyperlinks*. However, many publishers still do not fully use the potential of the Web, limiting their Web content to the old publishing paradigm of paper-based media. Most magazines and newspapers today use the Web in one way or the other, but only very limited approaches are being made to adapt the whole process of information gathering and its final publication to the new possibilities of a hypermedia environment.

1.2 Future Information Providers

We expect that in the future, the focus of information providers will shift from providing content to providing richly-interlinked content. Links in this context should not be regarded as limited as HTML's links are today. Links in the future will be typed, they will have comments attached to them, there will be ratings (eg, rating how much a resource could be seen as authoritative), links will be multi-ended (as opposed to HTML's 1:1 links), and links will no longer necessarily be embedded into the resources that they refer to. The main reasons we see for these developments are the following:

- *Emergence of new linking methods.* Today, most of the content delivered on the Web uses the *Hypertext Markup Language (HTML)* [27] with its restricted linking model. However, in the future XML in conjuction with the *XML Linking Language (XLink)* [12] and the *XML Pointer Language (XPointer)* [11] will be established as the new language for delivery of Web content.[1] In this paper, we will focus on the greatly enhanced hyperlink features of XLink/XPointer, which are closely connected to XML. In section 3, we will explain in detail how XLink/XPointer can be used to implement a much richer hypermedia model than is used in today's HTML-based Web.

- *Emergence of new user interfaces for link navigation.* One of the main reasons for the Web's success is its extremely simple point-and-click user interface, which was first implemented by the *Mosaic* browser. This simple and easy-to-use interface was a result of the limitations of HTML, where each link is embedded in the document and points to exactly one resource. With the introduction of a more complex hypermedia model using XML and XLink/XPointer, more sophisticated navigation functionality will become possible, which is likely to require new user interfaces within Web browsers in order to take advantage of this functionality. Although XLink and XPointer have been W3C draft recommendations for a while, there are currently no user interfaces implementing easy-to-use XLink/XPointer support. It can be expected that interfaces similar to the *Link Lens* described by Stanyer and Procter [31] will be developed.

- *Increasing availability of information on the Web.* As a general tendency, more and more resources are available on the Web, and the task of finding resources which are almost certainly somewhere on the Web becomes increasingly difficult. One approach to this dilemma is the emergence of more sophisticated search engines. Another is the direct utilization of a known URI[2] (from a bookmark or an external source, for example). Both of these approaches have limitations, and for effectiveness still require the support of associations between information sources, ie require an effective link model.

There are a number of other ways of supporting the effective location of information sources on the Web. For example, there has been significant emphasis on the use of metadata associated with Web resources. Metadata is any kind of information about information, in its simplest form the specification of the author of a resource, in its most complex form the description of a resource in some kind of knowledge representation formalism. Approaches such as W3C's *Resource Description Framework (RDF)*, consisting of a *Model and Syntax Specification* [18] and a *Schema Specification* [4], or alternative representations as described by Martin and Eklund [21], formalize the role and representation of metadata. This is claimed to have the potential to greatly improve the accessibility of information resources by attaching machine readable metadata to these resources. However, it is unlikely that in the short or medium term a significant share of resources on the Web will be described by metadata, since this description in many cases (possibly most) has to be manually created.

A possible alternative approach is to use the linked resources as an implicit source of metadata. The resources which are associated with a given Web resource through the link structures can be viewed as providing both informa-

[1] In some older publications, XLink and XPointer are collectively referred to as *Extensible Linking Language (XLL)*.

[2] We use the general term *Universal Resource Identifier (URI)* instead of the more limited *Uniform Resource Locator (URL)*, but in most cases these two concepts can be seen as synonymous.

tion about that resource (and hence metadata, especially if the conventional concept of metadata is represented in this way), as well as contextualising the resource.

In either case (ie, the lack of widely supported metadata, or the use of linked resources as a context for metadata) the links provided, and hence the link providers who maintain and provide these links, will still be important. The following list is only a fraction of new services which can be implemented using a link-based information model:

- *User-configurable links.* Users can have very different interests when reading a Web page. One user may be interested in links to documents which are related by content (eg, scientific publications about subjects discussed on the Web page), while other users may be interested in links to pictures, or links to biographical information about any person being mentioned on the Web page. Consequently, users may configure their personal preferences, and these preferences may be used by the Web server when responding to the user's request to get the links associated with a particular Web page. Similarly, the link structures (and content) which is relevant to a given user may vary over time, possibly based on aspects such as their navigation history, what other users have been doing, etc. Many of the issues which exist in this area have already been well investigated within the adaptive hypertext literature [5]. A possible method to automate this on the Web is the use of *HTTP cookies* [17].

- *Content-negotiated links.* The *Hypertext Transfer Protocol (HTTP)* [13] supports a mechanism called *content negotiation* [15], which enables clients to communicate their preferences (content types, content languages, and other content properties) to a server, which responds with a response best suited for the client's preferences. Link-based information providers can dynamically create link sets which are based on the client's preferences. This can reduce the number of irrelevant links significantly, since only links appropriate for the client are transmitted to the client.

- *Charging for links.* Conceptually, it may be desirable to charge a user separately for a document (ie, the content in form of a Web page) and the associated links. A variant of this is to provide the content free of charge, but to charge for the links. Furthermore, following the links may also incur a fee, if the links point to resources for which the link provided charges. We believe that many information providers will offer their content for free, but will provide the links as a value-added service for which they charge. This charging may be done individually (for each request for the links related to a particular document) or on a subscription base, where a user pays a fixed amount of money to be able to use an information provider's links for some period of time. Indeed, it is worth pointing out that a model that allows third party to provide links to content which they do not own or manage is a very powerful concept. This will be discussed further shortly.

- *Different levels of expertise.* Links in this new link-based world of publishing are more complex than HTML links, because they may contain additional information such as a general description, a large number of endpoints, rating for all endpoints (eg, rating the relevance of a link or the freshness of the information it is pointing to), classifications of links (eg, differentiating among authoritative references pointing to the official organization for a particular subject, and commentary references pointing to articles and comments about the subject). Using these more complex links may also involve a more complex user interface, and it is desirable to make this user interface configurable, so that it can be used in HTML's point-and-click mode, or in more complex ways, thereby revealing more information pertaining to an individual link.

Other issues worth considering include support for user annotation (including links to and from these annotations), provision of third-party linkbases, selling of linkbases, etc. For all these reasons, we believe that providing information on the Web will become more link-based in the future, and that the new genre of media which will emerge around link-based information will be useful for a long time, even if in the very long term metadata-based information will become more important.

2 Link-based Information

The main factors for the emergence of a new form of digital media have been described in section 1.2. The technical foundations, namely XML and XLink/XPointer, are rather new. Although XML as a new way to exchange structured data on the Web receives a lot of attention, XML (and in particular, many of the related standards) is still in its infancy, and the main components of the new XML-based Web, such as XLink/XPointer and the *Extensible Stylesheet Language (XSL)* [10], are still under development.[3] Even if all components of the XML set of recommendations are finalized, it will take a considerable amount of time until XML is established as a generally accepted and widely supported way to deliver information on the Web (as distinct from XML as a technology to manage the content which underpins the Web). The following reasons explain why the ultimate success of XML will take some time:

[3] XLink, XPointer, and XSL are W3C *Working Drafts*, which means that they are believed to be functionally complete, but may undergo major modifications before becoming W3C Recommendations.

- *Slow adoption of new technologies.* Although the main browsers claim to support XML in their latest versions, it will take some more time until they do so reliably and until all other standards required for a truly XML-based Web are completely and correctly implemented.[4] Even if the browsers implement all relevant standards, there will be many users still using older versions, and information providers have to make sure that they can serve both users of old and new browser versions. There is also a substantial volume of existing content which is formatted and stored in HTML. This will either require conversion or replacement.

- *Lack of authoring tools.* Currently, it is not clear how authoring tools could support authors in creating and maintaining large collections of links associated with a large collection of documents, and it will take some time until products appear on the market.

- *Increased user interface complexity.* The adoption of a more complex linking mechanism such as XLink will require more complex user interfaces if we are to take advantage of the increased sophistication of the linking functionality. Even though user interfaces will be configurable, so that users unwilling to deal with the new method can still use simple, HTML-style links, it will take some time until at least a fraction of the users will become accustomed to new user interfaces and demand the increased amount of richly linked information that can be accessed through these interfaces.

Although we believe that the new media type of link-based information will fade slowly into the media distributed over the Web, we think that information providers should start as soon as possible to generate this type of information. It is likely that in the beginning much of the collected link-based information will be stripped to generate plain HTML pages with simple links, but as soon as XML and XLink/XPointer become more widespread, it will be possible to generate truly link-based information resources. In the following sections, we briefly sketch the requirements for the creation, processing, and distribution of link-based information resources.

2.1 Technical Requirements

The technical requirements for creating and processing link-based information resources can be divided into two parts, the first part being the overall architecture as discussed in section 2.1.1, and the second part being the hypermedia presentation, which is described in section 2.1.2.

[4] In this context it is interesting to note that none of the main browsers completely and correctly implements HTML 4.0, although they usually claim to do so.

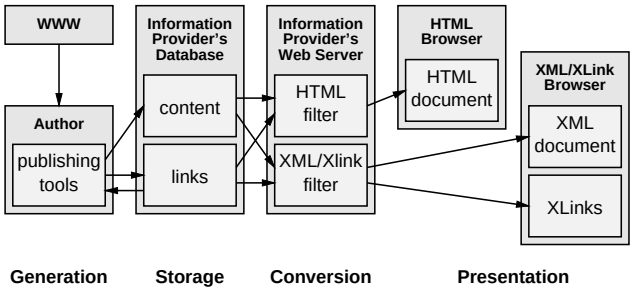

Figure 1. Publishing model of link-based information

It should be noted that both aspects are orthogonal, making it easy to adapt the proposed scheme to different hypermedia environments.

2.1.1 Overall architecture

It is assumed that documents and links are available on information servers (usually HTTP servers), and that all these resources can be referenced by a globally unique identifier (usually URIs). The architecture also is based on the assumption that the documents are using one of the Web's document markup languages (HTML or XML). In case of HTML documents, links are embedded into the document. This can be achieved by a server creating HTML documents on the fly, embedding all relevant links for a particular request. This process is shown in figure 1, where the information provider stores content and links in some kind of database.

Also shown in this figure is the case where the server generates an XML document from the content and XLinks from the links. It should be noted that the content and link information on the server can be stored in some arbitrary format, as long as mappings to HTML and XML/XLink are defined.[5]

Since we regard the transformation to HTML as a means for backwards compatibility, but not as the primary target for creating link-based resources, we will concentrate on the scenario where the request for information is requested by a XML/XLink-capable browser (in section 4.1.2, more information on how to achieve backwards compatibility is provided). The following steps are performed by the user for displaying and then using (ie, following) one of the links provided with a document:

1. *Retrieving the document.* This is the first step of any user interaction, where the document itself is retrieved

[5] However, he formats used on the server could be XML/XLink, making the mapping for XML/XLink browsers trivial and using some XML-based mechanism such as the *XSL Transformation Language (XSLT)* [9] for conversions to HTML.

from the server. The document contains one or more XLink *extended link groups* referencing the XLinks which are associated with it. Any XLink information inside the XML document is identified using the *XML namespaces* [2] recommendation. During this first step, it might also be necessary to retrieve one or more style sheets associated with the document.

2. *Retrieving the associated XLinks.* The XLinks are provided in one or more separate resources, which are either referenced from the document which has been retrieved in the first step, or specified through some other means (such as user preferences for individual linkbases). The retrieval of the XLinks may require some user interaction, because it is possible that this information will be charged for. The XLinks associated with a particular document may vary according to user preferences, browser capabilities, and the service that the user subscribed to.

3. *Retrieving the XLink target resources.* Depending on the types of XLinks, some target resources have to be retrieved immediately (because they use the *auto* value for the *actuate* attribute), and others will only be retrieved after the user requested the resource. In both cases, the *show* attribute determines how the target resources will be displayed or processed. It is possible that some of the target resources will be charged for. In addition to the target resources themselves it may be necessary to retrieve one or more style sheets associated with these resources.

This simple process illustrates the increased complexity of the link-based model, where the retrieval of links is a separate process which is highly dynamic. This is one of the key points the link-based model, where links are stored in a way which makes it possible to select the set of links for some documents based on a number of criteria. This dynamic selection of relevant links, if implemented properly, does not influence the recall, but increases the precision of the set of links that is provided to the user.

2.1.2 Hypermedia Functionality

XML documents and XLinks may be used to distribute content and links over the Web. Adhering to the Web's design philosophy, both the document itself and the links should not prescribe any kind of presentation, they should only provide the information in a way as presentation-independent as possible, with the presentation information deferred to XSL style sheets. We will first discuss how documents and links may be presented, before we give a short overview of the hypermedia functionality of XML documents and XLinks.

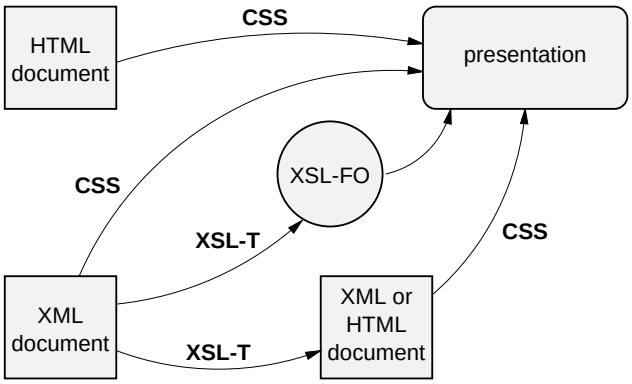

Figure 2. Using style sheet mechanisms for WWW documents

Presentation Aspects. Following the general guidelines of Web design, the presentation of documents should be separated from content and specified using style sheets. The presentation of HTML and XML documents can be specified using *Cascading Style Sheets, level 1 (CSS1)* [20], *Cascading Style Sheets, level 2 (CSS2)* [1] (CSS1 and CSS2 are described in detail by Lie and Bos [19]), or XSL. Figure 2 shows the possible ways from documents to presentation, using XSL's separation into a *transformation language (XSLT)* and *formatting objects (XSL-FO)*.

The presentation of HTML documents will generally only be specified by CSS style sheets, while the presentation of XML documents may be defined by CSS and/or XSL style sheets. XSLT transforms the source XML document (ie, the document to be presented) into another document, which may be a document using XSL-FO, or some other XML or even HTML document, which is then presented using CSS. This variety of style sheet mechanisms makes it possible to select the appropriate style sheet mechanism for each individual document, with CSS being simple and rather limited in its functionality, while XSL is much more complex. The actual style sheet being used for a document is determined using a mechanism for associating stylesheets with XML documents [8].

The presentation of XLinks is not yet part of any standard or implementation (Carr [7] describes a partial implementation of XLinks, but no user interface for link navigation), so we can only speculate how XLinks may be presented to users. We can however gain some interesting insights by looking at various hypermedia systems which have implemented richer linking models that that currently found in the Web.

Hypermedia Aspects. XLink is the key component for the new focus on link-based resources. Prior to the definition on XLink, the distribution of hypermedia links was re-

stricted to using HTML's link mechanism (the well-known <A> element), which implements a very simple hypermedia link. XLink extends this in several dimensions, making it possible to distribute much richer hypermedia links.

- *Directionality.* With HTML, links are always unidirectional, pointing from one resource to another. It is not possible to follow a link backwards, only when using the browser's "back" button, which is not a true link traversal but a request to display a previously cached resource. XLinks can be bidirectional, which makes it possible to follow the links in both directions.
- *Number of resources.* XLinks can link any number of resources, thus generalizing HTML's 1:1 model of hypermedia links. A possible example is a link associating a document with all its authors.
- *Resource semantics.* XLink implements resource semantics, so that a link associating a document with all its authors may differentiate the roles "document", "main author", and "co-author", and the usual definition of such a link would require that there is exactly one document, at least one main author, and co-authors would be optional.
- *How to present a resource.* Linking one resource to another resource does not specify in which context a resource is to be displayed once the link to that resource is traversed. XLink defines three different possibilities:
  - *Embedding.* In this case, the resource is embedded into the resource from where the link is traversed. This has the effect of a transclusion, including a resource (eg, a part of a Web page) into the resource where the traversal started. This functionality rises important questions about copyright issues, which have to be addressed when using embedding.
  - *Replacement.* This is the functionality as known from most HTML links, the new resource replaces the resource from where the link traversal was initiated. For graphical user interfaces, this means that the window content changes from one resource to the other.
  - *New context.* In this case, the new resource should be presented in a new context, not affecting the context of the resource where the traversal started. For graphical user interfaces, this means that the a new window is opened in which the new resource is displayed.

  Embedding is a new functionality of XLink which was not possible with HTML's linking model, while replacement and new context could be achieved with HTML links, in case of new context by using the <A> element's TARGET attribute. We assume that embedding (enabling the transclusion of resources) will have a large impact how resources are used and re-used on the Web.
- *When to present a resource.* Links can be traversed automatically or on user request. HTML only allows links to be traversed on user request, but with the new functionality (especially embedding) it may be necessary to retrieve a resource a link is pointing to in order to be able to completely present the resource containing the automatic link.

Much of the sophisticated linking functionality is a consequence not of the link, but of the way in which the link anchors are represented (effectively, this is managed by XPointer rather than XLink). XPointer allows us to define complex link sources and link end points – the anchors can be single words or phrases, or complex sets of sub-resources inside a resource. The anchors can also be computed rather than hard-coded. Effectively, this allows us to define hypermedia concepts such as generic links and dynamic links.

It should be noted that these features of XLink are not unique or new, in fact XLink's linking model is based on *HyTime* [16], HTML's link mechanism, and the linking mechanisms defined by the *Text Encoding Initiative Guidelines (TEI P3)* [30]. This means that implementations of these linking concepts already exist and have been proven useful. However, XLink's goal to be used in for the Web introduces a a degree of complexity which is an order of magnitude greater than any which has been tested with earlier systems.

We assume that XML and XLink/XPointer will be successful and that advanced techniques for exchanging documents and links on the Web will be based on these standards. However, even if for some not foreseeable reason one of these standards will not become generally accepted, the basic mechanisms for link-based information such as the separation of content and links and advanced link properties such as bidirectionality and link semantics will still be important and finally be used in some other format. The important issue is to use a model for information storage which supports all these aspects. Even if the information is at first being used to generate plain HTML pages, it should not be forgotten that at some later point in time it would be much better to generate XML/XLink pages for requests for the same information.

Consequently, the technical realization of link-based information has to include two stages, at first a data model for in-house storage, supporting advanced hypermedia features and providing tools for easy creation and maintenance of links, and as a second step filters which transform this in-house information of content and links to formats

which are being requested from customers, HTML today, XML/XLink in the near future, and maybe yet some other format in the long term.

It is worth noting that this approach has been adopted in a number of existing systems. For example, the HyperWave server maintains it's content and links (stored in a separate linkbase) in a form which retains the richness of the model. HyperWave browsers can be used to access this material in a sophisticated fashion, or converters can be applied which convert the content into a form suitable for use on the Web. (ie, HTML pages). Webcosm uses a similar approach, though it allows the use of a set of filters which can select and adapt links before they are merged with the content in the creation of HTML pages. However, in both cases the representation is inconsistent with the evolving XML standard.

2.2 Organizational Requirements

For an organization making the transition from content-centered to link-based publishing, it is important to realize that this transition will change many things. First of all, the authors have to integrate link creation and maintenance into their daily tasks, and this will be very easy for some authors who already use the Web a lot and believe that putting their content into context with links is a good idea. However, there will also be authors who are not using the Web a lot or at all and who disagree with the idea of making links a foundation of the publishing process. Integrating all these different opinions in a reasonable way in many cases will be a very challenging organizational task. The main decisions to be taken in the process towards link-based publishing are the following:

1. *Strategic decision for resource production.* The shift from link-poor content to link-rich content should be seen as a strategic decision, which has a large number of implications, such as new tasks for resource production, new business models for providing information, and the utilization of new technology and tools. It is therefore important to be informed about all implications of such a decision, technically, organizationally, and economically.

2. *Assigning sufficient resources.* Creating and in particular maintaining a high-quality database of links is a very time-consuming task. It is therefore necessary that sufficient resources are allocated to this task. This means that link maintenance must become an accepted and required activity in the work process. Furthermore, some kind of quality assurance of links should be implemented, possibly involving an easy-to-use feedback mechanism for customers who can report any problems that they may have with a link.

3. *Using appropriate tools.* Shifting the focus from content to links involves the introduction of new tools, such as authoring tools and databases. Using these tools requires new skills, and it may take some time until all people are sufficiently educated. Since these new tools are the foundation of the new focus on links, it is crucial that the introduction of new tools is planned and executed carefully.

When making the transition, it should always be realized that producing link-rich content is the key issue. Whether these resources are only available in-house, published in a limited form using HTML, or published with an XML/XLink-based model, is not the main issue. Ultimately, there will be a demand from the consumer side to get high-quality link-rich content, and it is important that this information will be available. XML/XLink is only one possible form of exporting link-based resources, and currently it looks as if it will be the Web's hypermedia format of the future, but the most important aspect is to make sure that all information relevant to links (such as classifications) can somehow be captured and stored for later retrieval, analysis, and understanding.

3 Implementing Link-based Resources

The technical and organizational requirements presented in section 2 illustrate that implementing link-based resources is a complex process. The transition from a content-centered way of providing information to a link-based way can be separated into five phases (figure 1 can be used to identify these phases in the overall publishing model):

1. *Creating a new data model.* In a first step, it is necessary to create a data model that supports link-based publishing. In particular, links must represent a key foundation of the data model. To support the efficient creation and maintenance of link information, the data model must be accompanied by powerful publishing tools, making it easy to generate link-based information.

   The data model not only defines the functional aspects of links (which should be identical or a superset of the functionality of XLinks), but also the semantics assigned to links and individual resources within links, making it easy for authors to reuse, update, or extend existing links. Efficient and powerful support for the usage of links by authors is the key factor for the successful implementation of link-based publishing.

2. *Selecting appropriate publishing tools.* Although the data model is the foundation on which everything else

is based, the quality of the information depends entirely on the level of support for authors. An environment meant to implement link-based publishing also has to provide link-based publishing tools, making it easy for authors to access link information. Furthermore, the publishing tools should enforce link reuse (by using updates and extensions) whenever possible, thus ensuring that existing links are properly maintained and reused instead of creating many redundant links which are very hard to unify afterwards.

Furthermore, some kind of quality control for links has to be established. This quality control monitors the generation of content and links, is responsible for making suggestions for link reuse, may compile link directories for relevant topics, should monitor any tools for automatic link maintenance (such as testing for broken links), and implement an internal linking schema (eg, linking all resources for specific topics or by specific authors or any other internal relations which may be of interest).

3. *Creating filters.* The first two steps have the goal of creating an environment supporting the generation of link-based information. In a third step, the goal is to export this information in a format which is mainly dictated by the available technology. An information provider could implement many filters, configurable through content negotiation or user preferences. It is the goal of these filters to present the link-based information in a way which is optimally adapted to the context of the customers, including their technology, history, current needs, etc.

   For example, an HTML filter could have two modes of operation, in the "simple" mode creating an HTML page containing all the links relevant to this page and pointing to the resources which has been marked as most relevant for these links, while the "complete" mode generates an HTML where each link points to a separate, link-specific HTML page containing the complete link information, pointing to more than one resource and including link semantics such as a description of the link and the resources it is linking. Depending on the user preferences, the HTML filter could automatically generate "simple" or "complete" HTML pages.

   Later developments for filters include exporting XML and XLinks, distributing style sheets (which can also be selected using content negotiation or user preferences), and the adaptation to new technologies beyond XML and XLinks as soon as they become available. A significant body of work already exists within the information retrieval and hypermedia literature on the use of link filtering which could be invaluable in guiding the adaptation of the concepts of XML and the Web.

4. *Designing a scheme for accounting, charging, and billing.* After designing an infrastructure for the creation, maintenance, and distribution of link-based information, the question has to be addressed if and how this information should be charged for. One possible approach is to provide the content for free (eg, newspapers making their articles available on the Web without any charge), but to charge for the links (eg, a newspaper charging a reader for the links associated with an article). Different schemes for charging can be thought of, in the simplest case cookie-based subscription services[6], in more elaborated scenarios the integration of integrated schemes for accounting, charging, and billing on the Web.

   Currently, most models for financial transactions on the Web are designed for macro-payments, while generally accepted schemes for making micro-payments, which would be more appropriate for a pay-per-link scheme, do not exist. It is not yet clear which way is the best to go for a service providing link-based information, the two general directions are subscription schemes based on generally accepted macro-payments (such as credit card transactions), or a micro-payment scheme based on a payment method which is not universally accepted and/or available.

   Similarly, we will need to determine charging models for when linkbases are provided by third-party vendors who are not directly affiliated with the content provider.

5. *Adaptation of filter mechanisms.* The final step, which has to be iterated infinitely, is the adaptation of filtering mechanisms to technical developments and customer requirements. Filtering is based both on user preferences and on supported hypermedia mechanisms on the user side, and both elements are likely to change over time. Consequently, an information provider must continually monitor technical developments and customer requirements and thus make sure that the provided service is always appropriate for the fast-changing environment on the Web. This continuous task should not be underestimated, both in expenses and in effects regarding customer satisfaction and ultimately success of the service.

This road map for the implementation of link-based resources is only a very rough description and ignores some

[6]However, cookies are not well-suited for this, because instead of a *user*, they effectively identify a particular *browser installation*. For one browsing session, these two concepts are identical, but in most cases this relationship is not reliable enough to be used for automatic billing schemes.

important questions, for example the whole copyright issue. If resources can be included automatically, it is not yet clear how these documents are protected by copyright issues, especially because a document including resources from different providers appears to the customer as one uniform resource, if no special measures are taken to mark included resources. The legal problem of copyright issues can only be resolved if technical provisions for signing and therefore protecting resources are taken, but work on related issued such as digitally signed XML resources using the *XML-Signature* [28] approach has only just started.

4 Challenges

In the previous section we have pointed out that link-based information will become more important in the future and that it is possible to implement a strategy which enables the smooth transition from today's HTML Web pages to a more sophisticated model of XML documents and associated XLinks. However, the implementation of link-based resources is a complex issue and involves many difficult tasks. In the following section, some of the challenges which have to be resolved when implementing a link-based information model are discussed.

4.1 Technical Challenges

The technical challenges for implementing link-based information are not a question of the link model. We assume that taking XLink as an example of how links will be distributed in the foreseeable future is a good starting point for designing the in-house linking model. The in-house linking model, which is being used for creating, maintaining, and exporting links should be designed in a way which optimally supports the reuse of links, thus making sure that the link-based approach is supported properly. However, even if the in-house linking model is sufficiently powerful for creating link-based information, there still remain some questions to be solved.

4.1.1 Authoring tools

Today's authoring tools developed for the Web tend to be very content-centered. Most of them are basically word processing tools for producing content without putting it into context. Many of them even ignore the underlying document structure, focusing only on the layout of an HTML document. Future authoring tools are supposed to work more closely together with an already existing database of content and links, thus enabling authors to produce content and put it into context using links. The presentation of the content should then be defined in a separate step. Because for link-based information it is essential that authoring tools support the reuse of links, authoring tools must work in close cooperation with the database of content and links. This architecture results in new requirements for the authoring tools, which no longer only store information in the database, but also must provide comfortable and powerful access to the database's content.

Furthermore, the authoring tools of the future must include Web access, including functionality for creating links to Web resources and sub-resources (XPointers can be used in countless ways to make references to the same sub-resource, and selecting the most robust XPointer can be difficult). Only if the authoring tools provide seamless and comfortable integration with the Web and the in-house database of content and links, authors will be able to efficiently create link-based information.

4.1.2 Transition process

Although we believe that XML and XLink/XPointer are the appropriate foundations of link-based information, it will take some time until resources on the Web can be distributed exclusively in these formats. For the transition period, it is important that an information provider uses an adaptable mechanism for exporting resources. Such a process could be improved by a mechanism such as the *Composite Capability/Preference Profiles (CC/PP)* [29], which define a way for clients to communicate their capabilities and preferences to servers. However, in the simplest case, the information provider's server could analyze the HTTP request's `Accept` and `User-Agent` header fields and then generate the appropriate response, based on the client's capabilities.

4.2 Organizational Challenges

One of the biggest organizational challenges clearly is the shift from content-centered thinking to a more link-based view of information. Many traditional publishers (such as newspapers and magazines) will find it difficult to have all authors and editors accept this change. It will be necessary to make a strategic decision to view links as an important factor for providing useful information, and it will furthermore be necessary to make sure that this strategic decision is adhered to. This means that some sort of quality control must be established, making sure that the task of creating and maintaining is taken seriously. These changes within an organization can be very hard to implement, and the transition process from a traditional, content-centered publisher to a link-based view of information provision will therefore be time- and resource-intensive.

5 Conclusions

In this paper, a new view of information resources has been proposed. There are still many technical and organiza-

tional challenges to tackle before this new view can be applied without any restrictions, but the existence of a smooth transition between the older content-centered view and the new link-based view makes it possible to extend the transition process. Consequently, information providers should start as soon as possible to rethink and rework their ways of producing and providing information. However, there still remain many open issues in the areas of authoring tools and user interface design for browsing link-based information, in the domain of charging and billing mechanisms for pay-per-link schemes, and it will take quite a while until these issues are resolved and the software implementing these solutions is widely deployed.

References

[1] B. Bos, H. W. Lie, C. Lilley, and I. Jacobs. CSS2 Specification. World Wide Web Consortium, Recommendation REC-CSS2-19980512, May 1998.

[2] T. Bray, D. Hollander, and A. Layman. Namespaces in XML. World Wide Web Consortium, Recommendation REC-xml-names-19990114, Jan. 1999.

[3] T. Bray, J. Paoli, and C. M. Sperberg-McQueen. Extensible Markup Language (XML). World Wide Web Consortium, Recommendation REC-xml-19980210, Feb. 1998.

[4] D. Brickley and R. V. Guha. Resource Description Framework (RDF) Schema Specification. World Wide Web Consortium, Proposed Recommendation PR-rdf-schema-19990303, Mar. 1999.

[5] P. Brusilovsky, A. Kobsa, and J. Vassileva, editors. *Adaptive Hypertext and Hypermedia*. Kluwer Academic Publishers, Boston, Massachusetts, June 1998.

[6] V. Bush. As We May Think. *The Atlantic Monthly*, 176(1):101–108, July 1945.

[7] L. Carr. A Simple XLink Package. *InterChange – Newsletter of the International SGML/XML Users' Group*, 4(4):17–22, Oct. 1998.

[8] J. Clark. Associating stylesheets with XML documents. World Wide Web Consortium, Recommendation REC-xml-stylesheet-19990629, June 1999.

[9] J. Clark. XSL Transformations (XSLT) Version 1.0. World Wide Web Consortium, Working Draft WD-xslt-19990709, July 1999.

[10] S. Deach. Extensible Stylesheet Language (XSL) Specification. World Wide Web Consortium, Working Draft WD-xsl-19990421, Apr. 1999.

[11] S. DeRose and R. Daniel. XML Pointer Language (XPointer). World Wide Web Consortium, Working Draft WD-xptr-19990709, July 1999.

[12] S. DeRose, D. Orchard, and B. Trafford. XML Linking Language (XLink). World Wide Web Consortium, Working Draft WD-xlink-19990726, July 1999.

[13] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. Internet draft standard RFC 2068, Nov. 1998.

[14] W. Hall, H. Davis, and G. Hutchings. *Rethinking Hypermedia: The Microcosm Approach*. Kluwer Academic Publishers, Boston, Massachusetts, May 1996.

[15] K. Holtman and A. H. Mutz. Transparent Content Negotiation in HTTP. Internet experimental RFC 2295, Mar. 1998.

[16] International Organization for Standardization. Information technology – Hypermedia/Time-based Structuring Language (HyTime). ISO/IEC 10744, 1997.

[17] D. M. Kristol and L. Montulli. HTTP State Management Mechanism. Internet proposed standard RFC 2109, Feb. 1997.

[18] O. Lassila and R. R. Swick. Resource Description Framework (RDF) Model and Syntax Specification. World Wide Web Consortium, Recommendation REC-rdf-syntax-19990222, Feb. 1999.

[19] H. W. Lie and B. Bos. *Cascading Style Sheets*. Addison-Wesley, Harlow, England, second edition, July 1999.

[20] H. W. Lie and B. Bos. Cascading Style Sheets, level 1. World Wide Web Consortium, Recommendation REC-CSS1-19990111, Jan. 1999.

[21] P. Martin and P. Eklund. Embedding knowledge in Web documents. In Proceedings of the Eighth International World Wide Web Conference [26], pages 325–341.

[22] H. Maurer. *HyperWave – The Next Generation Web Solution*. Addison-Wesley, Reading, Massachusetts, 1996.

[23] T. H. Nelson. *Computer Lib/Dream Machines*. Mindful Press, Sausalito, California, 1976.

[24] T. H. Nelson. *Literary Machines*. Mindful Press, Sausalito, California, 1982.

[25] T. H. Nelson. The Heart of Connection: Hypermedia Unified by Transclusion. *Communications of the ACM*, 38(8):31–33, Aug. 1995.

[26] *Proceedings of the Eighth International World Wide Web Conference*, Toronto, Canada, May 1999. Elsevier.

[27] D. Raggett, A. Le Hors, and I. Jacobs. HTML 4.0 Specification. World Wide Web Consortium, Recommendation REC-html40-19980424, Apr. 1998.

[28] J. Reagle. XML-Signature Requirements. World Wide Web Consortium, Draft Note xml-dsig-requirements-990601, June 1999.

[29] F. Reynolds, J. Hjelm, S. Dawkins, and S. Singhal. Composite Capability/Preference Profiles (CC/PP): A user side framework for content negotiation. World Wide Web Consortium, Note NOTE-CCPP-19981130, Nov. 1998.

[30] C. M. Sperberg-McQueen and L. Burnard. Guidelines for Electronic Text Encoding and Interchange (TEI P3). Technical report, Text Encoding Initiative, Chicago, Illinois, Apr. 1994.

[31] D. Stanyer and R. Procter. Improving Web usability with the link lens. In Proceedings of the Eighth International World Wide Web Conference [26], pages 455–466.

[32] E. Wilde. *Wilde's WWW – Technical Foundations of the World Wide Web*. Springer-Verlag, Berlin, Germany, Nov. 1998.

[33] E. Wilde and D. Lowe. *Transcluding the Web*. Addison-Wesley, Reading, Massachusetts, Mar. 2000.