DHQ: Digital Humanities Quarterly
2009
Volume 3 Number 3

XML, Interoperability and the Social Construction of Markup Languages: The Library Example

Jerome McDonough <jmcdonou_at_illinois_dot_edu>, Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign

Abstract

The past decade has seen XML widely adopted within a variety of communities, including the digital library community. While it now plays a critical role in the infrastructure of many digital library operations, XML's promise of interoperability of data across systems and organizations has not been fully realized within digital libraries. The reasons for this are not primarily technical in origin, but social, and relate to the cultures of XML's designers and XML language implementors, and a failure on the part of the digital library community to grapple with the sociotechnical nature of XML and its implementation. Possible strategies for addressing these issues of interoperability might include reduction of the flexibility afforded by specific XML-based markup languages used by the digital library community, and an increased focus on standardizing translations between various communities of practice use of such markup languages.

Introduction: Failures of Interoperability with XML

Eleven years after its endorsement by the World Wide Web Consortium, XML has been widely adopted within numerous, disparate communities and in a vast range of application domains, from standards for electronic filing of federal income tax [Internal Revenue Service 2007] to user interface design [Goodger et al. 2001]. The digital library community has been an active and early adopter of XML, for use in structuring both content and metadata. The reasons for this rapid uptake of XML within the digital library community are familiar to anyone with experience in the world of markup languages: 1

- XML helps ensure platform (and perhaps more critically vendor) independence, simplifying the migration of content between systems;
- XML provides the multilingual character support critical to the handling of library materials;
- XML's extensibility and modularity allow libraries to customize its application within their own operating environments;
- XML helps minimize software development costs by allowing libraries to leverage existing, open source development tools;
- XML, through virtue of being an open standard which enables descriptive markup, may assist in the long-term preservation of electronic materials; and perhaps most importantly
- XML provides a technological basis for interoperability of both content and metadata across library systems.

For all of these reasons, XML-based content standards such as the Text Encoding Initiative (TEI) have seen wide adoption within the library community, and librarians have been actively engaged in the development of a number of XML-based metadata standards, including Encoded Archival Description (EAD), Metadata Object Description Schema (MODS), Metadata Authority Description Schema (MADS), Metadata Encoding and Transmission Standard (METS), Metadata for Images in XML (MIX), MPEG-21 Digital Item Declaration Language (DIDL), Open Archives Initiative Object

Reuse and Exchange (OAI-ORE), Preservation Metadata Implementation Strategies (PREMIS) and many others. XML is now used throughout the research library world, and is a fundamental part of the infrastructure developed within the digital library community over the past decade.

Despite its success, however, XML has not lived up to many librarians' expectations within one area, that of interoperability. Efforts to exchange information employing XML-based metadata standards such as Dublin Core have fallen prey to a number of encoding and semantic inconsistencies [Shreeves et al. 2005]. Variations in the use of namespacing (`<date>` vs. `<dc:date>`), in regional conventions (08-11-2008 in the U.S. vs. 11-08-2008 in the U.K.) and language and culture (11 Août, 2008 vs. 8 Sha'aban 1429 A.H.) confound application developers trying to process XML data. 2

Perhaps more surprising is the failure of XML to ensure interoperability at a syntactic level.[1] Digital library developers have expected that shared use of a XML standard for structuring content and metadata (what is commonly called “structural metadata” within the digital library community) would ensure content interoperability and promote the development of tools and services designed to work with content encoded according to that standard [Hurley et al. 1999]. In practice, however, this goal has proved rather elusive. Experiments conducted by participants in the Library of Congress National Digital Infrastructure for Preservation Program (NDIIPP) to test the exchange of digital objects between repositories failed even when participants were using the same XML-based encoding format [DiLauro et al. 2005], [Shirky 2005]. 3

While some of the failures experienced by the Library of Congress NDIIPP tests were the result of incompatible repository infrastructure, others resulted from mismatched expectations regarding the appropriate use of METS, one of the XML formats employed for the test. DiLauro et al., discussing Johns Hopkins University's experience in the NDIIPP tests, state, 4

> Stanford commented after their ingest of the JHU archive that they had expected one METS object for the entire archive. Because our approach resulted in many METS files – on the order of the number of items in the archive – the Stanford ingest programs experienced out-of-memory conditions. This situation may have been ameliorated had they used the reference code provided by JHU; however, this will be an area that we will look into for future archive ingest projects.
>
> This matter points to a broader issue observed during the various import processes of this phase. Though three of the four non-LC participants (including JHU) used METS as part of their dissemination packages, each of our approaches was different. Clearly there would be some advantage to working toward at least some common elements for these processes. [DiLauro et al. 2005]

As alluded to by [DiLauro et al. 2005], a critical difficulty for achieving interoperability using structural metadata standards such as METS is the level of flexibility they enable in structuring a description of an object. As [Nelson et al. 2005] note in their discussion of using the MPEG-21 Digital Item Declaration Language during the NDIIPP test, it is possible to map a single object into multiple different encodings using MPEG-21, depending on the level of granularity you wish to employ in the description. The same is true of METS and other, similar information packaging standards. They each provide a grammar to describe the structure of complex digital objects. To facilitate description of arbitrarily complex structures, these standards employ a relatively flexible grammar, and document authors can and do find a variety of different ways to describe the structure of a single object using one of those grammars.

To date, the digital library community has treated these interoperability issues surrounding structural metadata standards as a technical problem demanding a technical solution. Most efforts to solve these interoperability problems have focused on the use of a profiling mechanism to further constrain the creation of instance documents conforming with a XML schema, sometimes in conjunction with a validation mechanism (such as Schematron) to test instance documents conformance with the additional requirements established in the profile [Littman 2006], [Keith 2005]. In essence, profiles exist to limit authors' flexibility in the use of a given XML language. If different institutions can agree on using a particular profile of a language, they are far more likely to be able to produce content objects which can be 5 readily exchanged and interoperate with a variety of different systems.

However, while these mechanisms may be successful in ensuring interoperability within a narrowly defined local context, they are not in themselves any guarantee of interoperability at the scale envisioned by digital library projects such as Aquifer [Kott et al. 2006], which hope to promote the ready exchange and interoperability of digital library content among a multitude of institutions. The official METS profile registry already contains a variety of mutually incompatible profiles for similar types of objects, with profiles varying in their choices of descriptive metadata (Dublin Core vs. MODS), use of controlled vocabularies in descriptive and administrative metadata sections, and their structure (e.g., requiring the use of a single `<structMap>` element in the case of the Oxford Digital Library METS Profile and mandating the use of two `<structMap>` elements in the case of the Indiana University METS Navigator profile).[2] While profiles may enable localized interoperability, they do not necessarily lead to widespread agreement regarding the best ways of describing objects' structure, and in fact, it is conceivable that to a certain extent they reify differences between institutions. Allowing the specification of local profiles of a XML language may help formalize the problem of interoperability, but it does not solve it. 6

If we are to deal with the issues of interoperability that continually manifest themselves in the realm of structural metadata standards for digital libraries, we need to cease viewing this purely as a technical problem, and acknowledge that it is the result of the interplay of technical *and* social factors. The XML standards for structural metadata employed by the digital library community represent cases of sociotechnical systems, and only when we have analyzed the social, as well as the technical, components of these systems will we be able to suggest how they may be optimized to achieve the goals of interoperability, usability and preservability desired by librarians and their patrons. 7

XML from a Sociotechnical Perspective

One of the fundamental tenets of sociotechnical systems theory is that technological design and technological evolution are not value neutral processes. Technological design is both informed by, and seeks to inform, the social environment in which technology is used, and the work of designers and engineers can be seen as being as much social engineering as technical engineering. By providing a new means of accomplishing a task, a technologist is also prescribing a new set of behaviors centered on the new technology (and possibly proscribing others). This conceptualization of technology was concisely summarized by [Akrich 1992], who argues that 8

> ...when technologists define the characteristics of their objects, they necessarily make hypotheses about the entities that make up the world into which the object is to be inserted. Designers thus define actors with specific tastes, competences, motives, aspirations, political prejudices, and the rest, and they assume that morality, technology, science, and economy will evolve in particular ways. A large part of the work of innovators is " *inscribing* " [emph. original] this vision of (or prediction about) the work in the technical content of the new object [Akrich 1992, 207–8]

All the existing and developing standards for structural metadata[3] within the digital library community are XML-based. Any sociotechnical examination of these standards therefore must start with at least some consideration of XML itself. Our questions concerning XML, then, are what world view have XML's authors inscribed within it and what influence has that had on XML's uptake and use within the digital library community.

We can learn a great deal about the viewpoints of a particular technology's designers from the documents they author which define the goals for the technology (e.g., use cases and user needs analysis), those which help implement the technology (e.g., specification documents), and those which attempt to explain or promote the new technology to potential users. If we look at the original specification document for XML, we find a relatively clear set of goals for the technology enumerated: 9

1. XML shall be straightforwardly usable over the Internet.
2. XML shall support a wide variety of applications.
3. XML shall be compatible with SGML.
4. It shall be easy to write programs which process XML documents.

5. The number of optional features in XML is to be kept to the absolute minimum, ideally zero.
6. XML documents should be human-legible and reasonably clear.
7. The XML design should be prepared quickly.
8. The design of XML shall be formal and concise.
9. XML documents shall be easy to create.
10. Terseness in XML markup is of minimal importance

[Bray et al. 1998]

These goals convey some of the world view that XML's designers brought to bear in creating the technology. They saw XML as a transmission format for communications (hence the requirement that it be usable over the Internet). They believed that XML's success was contingent upon it being flexible enough to “support a wide variety of applications.” They also clearly believed that for XML to succeed it must be easy to use, but they also recognized that the meaning of “ease of use” is contingent upon the use one might make of the technology. Ease of use for a document author (“XML documents shall be easy to create,” “XML documents should be human-legible and reasonably clear”) is a good deal different from ease of use for a software developer (“It shall be easy to write programs which process XML Documents,” “The design of XML shall be formal and concise”).

This relatively small set of goals for the XML language was further elaborated upon by members of the original World 10
Wide Web Consortium (W3C) XML Working Group in a variety of papers they authored to introduce and clarify XML to its potential user community. [Bosak 1998] further defines the goals of XML as supporting the user needs of “extensibility, to define new tags as needed,” “structure, to model data to any level of complexity,” “validation, to check data for structural correctness,” “media independence, to publish content in multiple formats,” and “vendor and platform independence, to process any conforming document using standard commercial software or even simple text tools.” The benefits adhering to XML's providing a standardized format are identified as including “complete interoperability of both content and style across applications and platforms; freedom of content creators from vendor control of production tools; freedom of users to choose their own views into content; easy construction of powerful tools for manipulating content on a large scale; a level playing field for independent software developers; and true international publishing across all media.” Emancipatory language is invoked repeatedly here through the use of the terms *freedom* and *independence*, particularly affording users the freedom “to define new tags” and in so doing “choose their own views into content.” In the designers' world view, a key benefit to XML is the freedom it provides users to define their own structure for documents and data, using their own semantics, and to escape restrictions that software vendors (through their own inscriptions in their own products) might wish to impose on their users. Other articles by members of the XML Working Group (see, for example, [Bosak 1997] and [Bosak & Bray 1999]) reiterate a vision of XML as a technology allowing users to define their own structures while simultaneously supporting interoperability of documents and data on a global scale.

The XML 1.0 Recommendation bears the inscription of its designers' ideological stance towards appropriate 11
mechanisms for data and document structuring as well as appropriate relationships between document creators and software and platform vendors. The effort to promote a metalanguage over any specific markup language, the adoption of Unicode as a basic character set, and the elimination of SGML features which proved difficult to implement (including CONCUR, OMITTAG and SUBDOC) are some of the technological means through which XML's designers sought to normalize and reify a particular set of social and technological relationships. Nor did this process stop with the release of the XML 1.0 recommendation in 1998. The period between February 1998 and October 2001 saw the development and release of a plethora of additional XML specifications, including XML Namespaces, XSLT, XPath, XML Schema, XLink/XBase, XML Information Set and XSL-FO, as well as a variety of XML software tools including parsers, editors and stylesheet engines. All of these various technological objects presented their own opportunities for their designers to further refine the ideological inscription carried within the XML 1.0 Recommendation. One of these objects in particular, the Namespaces in XML Recommendation, deserves further examination due to its significant affect on structural metadata standards developed by the digital library community.

[World Wide Web Consortium 1999a] provides the following justification for the introduction of a formal namespace mechanism into XML: 12

> We envision applications of Extensible Markup Language (XML) where a single XML document may contain elements and attributes (here referred to as a “markup vocabulary”) that are defined for and used by multiple software modules. One motivation for this is modularity; if such a markup vocabulary exists which is well understood and for which there is useful software available, it is better to re-use this markup rather than re-invent it.
>
> Such documents, containing multiple markup vocabularies, pose problems of recognition and collision. Software modules need to be able to recognize the tags and attributes which they are designed to process, even in the face of “collisions” occurring when markup intended for some other software package uses the same element type or attribute name.[World Wide Web Consortium, 1999b]

A strong motivating force for the “Namespaces in XML” recommendation, then, was a desire to promote modularity in the design of markup languages. Interoperability was also cited as a motivating factor by the World Wide Web Consortium in the introduction of “Namespaces in XML” [World Wide Web Consortium, 1999b]. Fundamentally, the authors of the “Namespaces in XML” recommendation wanted to simplify XML document authors' lives by ensuring that they did not need to reinvent markup languages which already existed, and that they could readily mix elements and attributes conforming to disparate schemas within a single document instance without worrying about collisions between element and attribute names. Again, modularity and flexibility in design of markup languages would give users the freedom they need while also insuring interoperability.

XML's designers have inscribed two overarching messages within the technology they have created. The first is that XML is about establishing a new *social* relationship between content creators and software vendors. By putting control of data formats into the hands of the content creation community via an open standard, XML provides that community with signicant political leverage. They can avoid the proprietary data formats that software vendors have used to lock them into continuing use of a particular software package. XML thus represents the path to freedom. The second message is that XML enables *easy* communication and interoperability. XML will not only allow you to control your content, it will make it easier for others to use your content. Freedom and interoperability are the two underlying themes running through the complete set of XML specifications, with modular design embraced as the means for achieving these ends. 13

Structural Metadata Standards and the Digital Library Community

Libraries' exploration of the use of markup languages for encoding of library data predates the origin of XML by several years. Collaborations with the digital humanities community on the development of the TEI Guidelines, the development of Encoded Archival Description, and early efforts to apply SGML to bibliographic data[4] provided the library community with experience in the use of markup languages and demonstration of the benefits they could provide. When the XML 1.0 Recommendation was released, many digital library projects were already using SGML, and libraries were quick to embrace XML. XML's simpler design meant that software tools for processing XML data were readily available, and the new capabilities for data typing introduced by XML schema languages made XML even more attractive for certain uses than its predecessor, SGML. Early projects which employed XML, such as the Making of America II project [U.C. Berkeley Library 1997], were rapidly followed by a number of XML-based markup languages intended for use in the library community. A significant focus of much of the library community's work with XML has been developing languages which can serve to structure all the metadata and data comprising a digital library object into a coherent whole. Examples of languages developed and explored for this purpose in the library community include the Metadata Encoding and Transmission Standard (METS), the Fedora Object XML (FOXML) language, MPEG-21 Digital Item Declaration Language, and the new Open Archives Initiative Object Reuse and Exchange (OAI-ORE) specification. 14

Most of these languages employ a similar pattern for structuring content and metadata. They provide an encoding 15 mechanism which allows the author to record a hierarchical structure defining the object, and then associate both content and metadata with various nodes within that structure. Figure 1 depicts a very simple version of such a structure, a book with a single chapter; metadata and content files (and metadata *for* the content files) are associated with appropriate nodes.

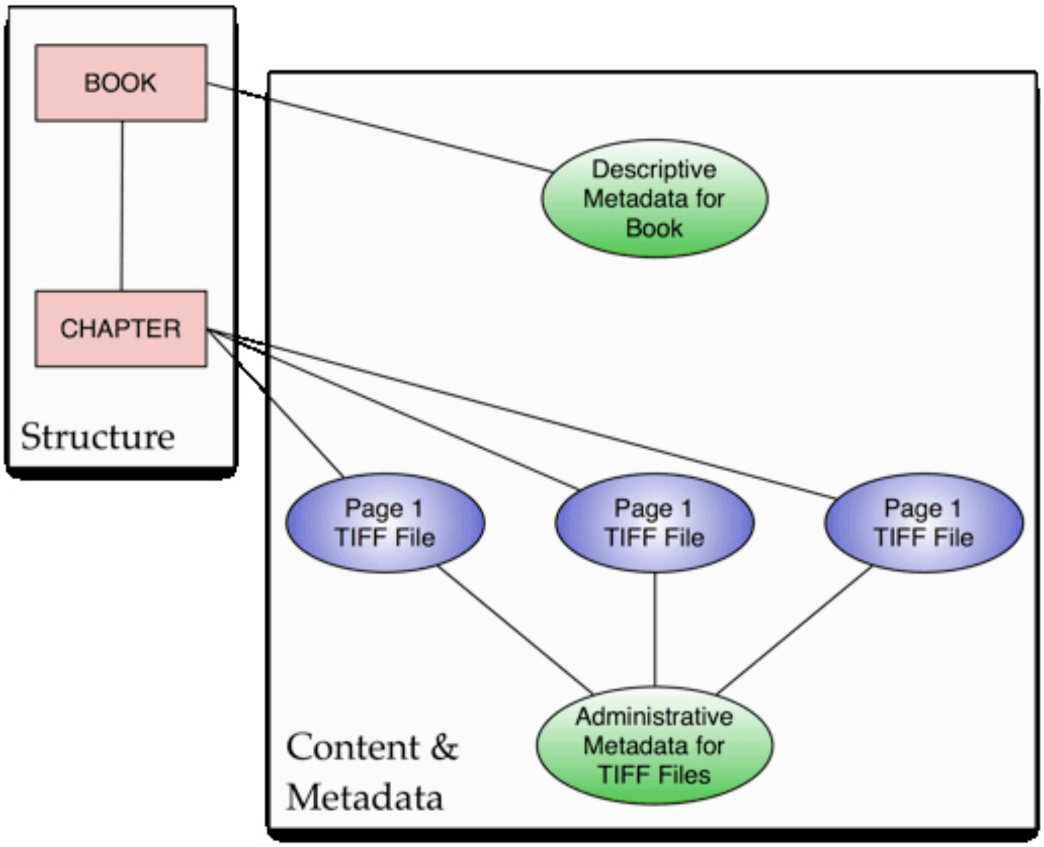

Figure 1. A simple digital object

A METS encoding for such an object can be seen in Example 1. The hierarchical structure for the object is defined within the `<structMap>` element as a set of recursive `<div>` elements. Subsidiary `<fptr>` elements within a `<div>` are used to associate that `<div>` element with content files described in separate `<file>` elements, and ID/IDREF linking attributes are used to associate the root `<div>` element with a descriptive metadata record, and the individual `<file>` elements with an administrative metadata record. A TYPE attribute on the `<div>` elements allows the METS document author to indicate the type of subobject represented by each node in the structural hierarchy.

```xml
<mets xmlns="http://www.loc.gov/METS/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xsi:schemaLocation="http://www.loc.gov/METS/
http://www.loc.gov/standards/mets/mets.xsd">
    <dmdSec ID="DM1">
        <mdWrap MDTYPE="OTHER">
            <xmlData>
                <meta>Descriptive Metadata for Book</meta>
            </xmlData>
        </mdWrap>
    </dmdSec>
    <amdSec>
        <techMD ID="AM1">
            <mdWrap MDTYPE="OTHER">
                <xmlData>
                    <meta>Administrative metadata applicable to TIFF files</meta>
                </xmlData>
            </mdWrap>
        </techMD>
    </amdSec>
    <fileSec>
        <fileGrp>
            <file ID="F1" ADMID="AM1">
                <FLocat LOCTYPE="URN" xlink:href="urn:x-
mets:Location_of_Page_One_TIFF_Image"/>
            </file>
            <file ID="F2" ADMID="AM1">
                <FLocat LOCTYPE="URN" xlink:href="urn:x-
mets:Location_of_Page_Two_TIFF_Image"/>
            </file>
            <file ID="F3" ADMID="AM1">
                <FLocat LOCTYPE="URN" xlink:href="urn:x-
mets:Location_of_Page_Three_TIFF_Image"/>
            </file>
        </fileGrp>
    </fileSec>
    <structMap>
        <div TYPE="book" DMDID="DM1">
            <div TYPE="chapter">
                <fptr FILEID="F1"/>
                <fptr FILEID="F2"/>
                <fptr FILEID="F3"/>
            </div>
        </div>
    </structMap>
</mets>
```

Example 1.

The goal of representing the structure of a work as a hierarchy of nested `<div>` elements with TYPE attributes was to have a relatively simple, abstract hierarchical structure that could be readily applied to a variety of materials. This was intended to promote the adoption of the standard (a single, simple standard is more likely to be adopted than a variety of complex ones), which in turn was seen as promoting interoperability. Having all of the digital library community using a single standard for structuring content and metadata was seen by METS' designers as preferable to the community adopting a disparate set of standards. 16

It should be noted that this move towards abstraction was a relatively significant break from the SGML design practices that many research libraries had been using to date. While it is true that the notion of using nested `<div>` elements was derived from the TEI text apparatus, TEI does not rely on pure abstraction; one does not expect to encounter encoding such as `<div type="figure">` in a TEI document, when a `<figure>` element is available to use. Just as XML itself, 17 the METS schema carries an inscription of its designers' world view, that it was preferable to develop a single, simple, generalizable, highly abstract model and encoding mechanisms to structure content and metadata for digital library objects, rather than to pursue the development of a variety of highly specific schemas (one for photographs, one for journals, etc.), or a grand encompassing schema that contained elements appropriate to different genres that could be combined as needed (e.g., the TEI model). Despite its use of the abstract `<div>` element with a TYPE attribute to represent the structural components of a digital library object, however, the METS schema insisted on the use of more specific concrete elements to identify different forms of metadata, with the `<dmdSec>` element used for descriptive metadata and the `<amdSec>` element used for administrative metadata, along with a series of subelements for different forms of administrative metadata (technical, rights, provenance and source). This typification of different forms of metadata was itself an effort to promote both modularity in further metadata schema development and the creation of certain types of metadata schema. By identifying specific subclasses of metadata within the METS schema, METS' designers hoped to encourage XML developers in the digital library world to create discrete, specialized metadata standards that would align with those subclasses, and that those creating digital library objects could then select from a set of such modular XML metadata standards in composing a particular object. Through METS' design, its implementers consciously sought to encourage the adoption of modular schema design practices within the digital library community.

Other XML-based markup languages adopted by the digital library community have taken a similar approach. The MPEG-21 Digital Item Declaration Language also employs a rather abstract hierarchical structural mechanism for ordering content and metadata. It differs inasmuch as non-structural metadata (`<Descriptor>` elements in MPEG-21 parlance) are not typed, and structural metadata elements, while still rather abstract, are of three different types: `<Container>`, `<Item>` and `<Component>`. The Open Archives Initiative Object Reuse and Exchange specification is perhaps the most abstract of all the structural metadata standards adopted within the digital library community; while it has multiple serialization syntaxes, all of them employ a single mechanism to link an abstract aggregation with a set of aggregated resources (although the specific linking mechanism varies according to the serialization syntax employed). Those aggregated resources may in turn be aggregations, and any aggregation may be associated with a variety of additional metadata. 18

If we examine these other standards to determine what inscriptions their designers have placed within them regarding their use, we find messages very similar to that of METS. Structural metadata should be highly abstract, so that a very small set of elements can be employed to structure widely disparate content genres. While METS was perhaps more vocal in trying to push the message that further development of metadata schemas should try to create small, focused and modular metadata sets that could be drawn upon as needed to encode a particular object, the other standards convey the same message (through the use `<Descriptor>` elements to associate metadata with other elements in the case of MPEG-21, and through RDF mechanisms in the case of OAI-ORE). Other structural metadata standards of interest to the digital library community employ similar mechanisms. The XFDU standard for data archiving uses hierarchies of `<ContentUnit>` elements that may be associated with `<dataObjects>` and `<metadataObjects>`. The IMS Content Packaging standard for learning objects uses hierarchies of `<item>` elements that may be associated with `<resources>` and `<metadata>`. While implementations differ in details, the pattern is similar and widespread across the various structural metadata standards of interest to the digital library community. Again and again we see designers seeking to achieve wide adoption of their standard in order to promote interoperability between differing institutions; to secure this goal, they favor a highly abstract structural mechanism which can be applied to a wide variety of content, and mechanisms to allow a variety of additional metadata schemas to “plug and play” within the larger structural framework. 19

While of perhaps some limited interest to social researchers of technology, none of the preceding seems particularly surprising or problematic. That the designers of XML itself, and the designers of encoding standards for digital library metadata and content, should favor flexibility, extensibility, modularity and the use of abstraction to support the generalizability of their standard, and hence promote its widespread adoption to help achieve interoperability, would not be a great shock to anyone who has spent more than five minutes in the company of computer scientists. These are all considered almost innate goods among software engineers in general and markup language enthusiasts in particular. 20

Yet the NDIIPP tests cited previously would seem to indicate that flexibility, extensibility, modularity and abstraction are not in and of themselves sufficient to achieve interoperability. So what, specifically, is the problem that METS and other structural metadata standards are encountering?

Defining the Problem, or, Why is XML like a rope?

One of the earliest discussion points in the development of the METS standard was which of the various elements within the schema should be declared mandatory and which optional. After some discussion among the members of the working group that established METS' original design, it was decided that the `<structMap>` element, which records the basic tree structure on to which content files and metadata are mapped in METS, would be the only required element. METS, in the group's opinion, was fundamentally a *structural* metadata standard; it existed to provide a framework into which other descriptive and administrative metadata, as well as content, could be placed. The `<structMap>` element provided the tree upon which all the other structural components of METS where hung, where the logical and physical structure of a work could be delineated, and so was really the only section that needed to be mandatory. As the `<structMap>` was the only mandatory portion of a METS file, it was also expected that any structural description of a work should reside there; software that would process a METS file would expect to find logical or physical descriptions of the structure of a work residing within a structural map, and not elsewhere in the METS file. 21

It was a matter of some surprise for many in the METS community, then, when the Library of Congress, which serves as the maintenance agency for the METS standard, began to produce METS files for digital versions of certain kinds of audio recordings which placed the logical structure of the works in MODS records contained within the METS descriptive metadata section (`<dmdSec>`) rather than in the structural map, and registered a profile of METS establishing this as their formal internal practice for “recorded events” [Library of Congress 2006]. The MODS record within a METS file would provide a logical structure for the work using a hierarchical arrangement of the MODS `<relatedItem>` element, while the METS `<structMap>` would contain the physical structure, with ID/IDREF links used to draw connections between the two structural descriptions. A recorded concert, for example, might have a MODS record containing a hierarchy such as this:[5] 22

```
<mods:mods xmlns:mods="http://www.loc.gov/mods/v3"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="http://www.loc.gov/mods/v3
                  http://www.loc.gov/standards/mods/mods.xsd">
     <mods:titleInfo>
         <mods:nonSort>The </mods:nonSort>
         <mods:title>1946 Library of Congress recital</mods:title>
     </mods:titleInfo>
     <mods:relatedItem type="constituent" ID="DMD_disc01_tr001">
         <mods:titleInfo type="uniform">
             <mods:partName>Chaconne von Vitali</mods:partName>
         </mods:titleInfo>
     </mods:relatedItem>
     <mods:relatedItem type="constituent" ID="DMD_disc01_tr002_005">
         <mods:titleInfo><mods:title>Sonata in G minor, BWV 1001</mods:title>
         </mods:titleInfo>
         <mods:relatedItem type="constituent" ID="DMD_disc01_tr002">
             <mods:titleInfo><mods:partName>Adagio</mods:partName>
             </mods:titleInfo>
         </mods:relatedItem>
         <mods:relatedItem type="constituent" ID="DMD_disc01_tr003">
             <mods:titleInfo><mods:partName>Fuga : allegro</mods:partName>
             </mods:titleInfo>
         </mods:relatedItem>
         <mods:relatedItem type="constituent" ID="DMD_disc01_tr004">
             <mods:titleInfo><mods:partName>Siciliano</mods:partName></mods:titleInfo>
         </mods:relatedItem>
         <mods:relatedItem type="constituent" ID="DMD_disc01_tr005">
             <mods:titleInfo><mods:partName>Presto</mods:partName></mods:titleInfo>
         </mods:relatedItem>
     </mods:relatedItem>
     <mods:relatedItem type="constituent" ID="DMD_disc01_tr006">
         <mods:titleInfo><mods:title>Paganiniana : variations</mods:title>
</mods:titleInfo>
     </mods:relatedItem>
     <mods:identifier type="lccn">99594334</mods:identifier>
</mods:mods>
```

Example 2.

Meanwhile, the structural map would record the physical structure of the work as follows: 23

```
<div TYPE="cd:compactDiscObject" DMDID="MODS1">
     <div TYPE="cd:disc">
         <div DMDID="DMD_disc01_tr001" TYPE="cd:track"></div>
         <div DMDID="DMD_disc01_tr002" TYPE="cd:track"></div>
         <div DMDID="DMD_disc01_tr003" TYPE="cd:track"></div>
         <div DMDID="DMD_disc01_tr004" TYPE="cd:track"></div>
         <div DMDID="DMD_disc01_tr005" TYPE="cd:track"></div>
         <div DMDID="DMD_disc01_tr006" TYPE="cd:track"></div>
     </div>
</div>
```

Example 3.

This represents a *valid* use of METS (in the technical XML sense), but is a departure from expected practice, which would be to include both the logical and physical structural information within one or more <structMap> elements. 24

This example demonstrates two problems that have impeded the development of interoperable content within the digital library community. The first is that the implementation of highly abstract elements for the definition of structure provides 25 a tremendous amount of flexibility to document encoders; there are a vast number of potential encodings of any given object in METS, with variations possible in depth of structure (do I limit my structure to musical movements or do I provide structural information down to the measure level?), labeling (you say `TYPE="book"`, I say `TYPE="monograph"`), and arrangement (should the Lord of Rings film trilogy be encoded as a single METS file? Three METS files? Three METS files for the individual films and a fourth representing the abstract notion of the Trilogy?). This can lead to significant variation in encoding practices, even between two institutions dealing with remarkably similar material and using the same metadata standards, as noted by [DiLauro et al. 2005].

The second problem is what we might call the problem of standards independence. Many of the XML metadata schemas that have been developed with the help of the digital library community have been created with the understanding that ensuring their usefulness in a variety of application environments requires that they not contain inherent dependencies on other schemas; they need to be able to express all the relevant information within their particular domain on their own. In many of these XML standards, the designers recognized a need to be able to account for relationships between various content objects being described, whether the description being applied was the more traditional form of intellectual description you would expect in a library catalog, or a technical description of the composition of a TIFF image. The result has been that a number of common metadata schemas within the digital library field contain elements for expressing structural metadata, even schemas that are not primarily intended for recording structural metadata. Dublin Core has its `<relation>` element, MODS has its `<RelatedItem>` element, the PREMIS schema for preservation metadata has a `<relation>` element, even the MIX standard for still image technical metadata contains an element for referencing previous image metadata. As the standards' developers felt they should not make their efforts dependent on structural metadata mechanisms in other standards, they implemented their own. Unfortunately, with the addition of each new metadata standard containing structural metadata capabilities, the potential for difficulties with our first problem increases. Every new metadata standard created within the digital library community seems to add another mechanism for describing the structural relationships between content objects, and hence greater potential for variation in object encoding practices. 26

The irony is that both these problems derive from the flexibility, extensibility, modularity and use of abstraction to create structural metadata elements that the designers of the metadata schemas hoped to promote. The potential range of variation in encoding structural metadata is the result of each of these factors. The use of abstraction in METS (i.e., the `<div>` element) was an attempt to make the standard flexible in application; however, it opens up a tremendous degree of play in encoding practice. If you ask two different individuals how many page breaks there are in a text, the likelihood that they will give the same answer is a good deal greater than if you ask them how many divisions there are in a text. The use of abstraction opens up encoding to a much greater degree of personal interpretation, and hence variation. The extensibility of METS, and the hope to promote a modular system of metadata schema reuse that its authors inscribed within it, opens up the possibility of using other metadata schemas to encode structural metadata. And it was this same desire for flexibility and modularity that has led other metadata schema designers to include structural metadata components in their own schemas; they wanted to ensure that their own efforts were flexible enough to be applied in a variety of settings, and with a variety of others. But having to design their own schemas without knowing the specific supporting capabilities to be found in other schemas with which their own might be used, they are inevitably forced to create structural metadata capacities of their own within their schemas. The designers of metadata schemas (structural or otherwise) within the digital library community have sought to adhere to a particular set of design practices, seeking to create flexible, extensible, modular and generalized tools, and to promote like practice in others through inscription of their view of appropriate XML design within their technological artifacts. Unfortunately, promoting such good practices has been a death blow to one of the principle reasons for adopting XML in the first place: to ensure interoperability of digital library materials across systems. Wide-scale interoperability requires wide-scale adoption, but the design practices of schema implementers intended to promote wide-scale adoption run directly counter to wide spread interoperability. 27

Hence XML's similarity to a rope. Like a rope, it is extraordinarily flexible; unfortunately, just as with rope, that flexibility makes it all too easy to hang yourself. 28

Strategies for Interoperability in a World of Multilingual Markup

The digital library community seems to face a dilemma at this point. Through its pursuit of design goals of flexibility, extensibility, modularity and abstraction, and its promulgation of those goals as common practice through their inscription in XML metadata standards, it has managed to substantially impede progress towards another commonly held goal, interoperability of digital library content across a range of systems. How then, should the community respond? 29

One possible response to this situation would be to say that perhaps our community cares less about interoperability than we thought. Despite projects intended to promote interoperability, such as the Digital Library Federation's Aquifer, it may be that interoperability is actually a lower priority for the digital library community than it likes to believe, and the adoption of metadata standards that impede interoperability is merely a reflection of that underlying reality, and not a major problem to resolve. There is at least some reason to suspect this may be the case. Research libraries typically have a clearly defined local clientele, and while voices within the digital library community have been calling for some time for the liberation of content from local silos to enable their use by a larger community [Seaman 2003], libraries' primary responsibility will always be to their local communities. The first sentence in the mission statement for the University Library at the University of Illinois at Chicago exemplifies the priorities present at most research libraries: "The University of Illinois at Chicago (UIC) Library strives to meet the information needs of UIC students, faculty, and staff." [6] Prioritizing service to the local community is endemic to the social structure of library systems, and if systems developed to deliver digital library content to that community are successful in that context, and if the costs associated with achieving much more widespread interoperability are high, then many libraries may decide that interoperability, while desirable, is a goal which may have to wait. 30

If libraries do wish to make progress on the issue of interoperability of structural metadata, they will need to recognize that, as [Renear & Golovchinksky 2001] observed, "every significant information processing standardization effort must skillfully negotiate competing and apparently irreconcilable objectives, [and] serve a wide variety of stakeholders with many different interests." In the case of structural metadata, the particular competing objectives that the digital library community does not seem to have successfully reconciled to date are what [Kendall 2007], in a discussion of blogging practices, has labeled the problem of "control vs. connection." The structural metadata standards which have been developed to date, with their emphasis on flexibility, extensibility and modularity have sought to afford local institutions the greatest degree of control possible in their encoding practices. The standards are designed to allow any given institution to do what it wants. This has clear benefits in terms of easing adoption of the standard in any given context, and as a result insuring the standard's widespread adoption (obviously a good thing in a standard). However, increasing the amount of local control over the ways in which a language is used and developed is fundamentally at odds with a language's ability to serve as a means for connection with others outside the local context. It is, in essence, promoting the development of regional dialects at the expense of mutual intelligibility. The particular case of structural metadata standards reveals that sufficient local variation in syntax, the ways in which people structure their objects using a markup language, can be as fatal to communication as variation in semantics. 31

Given this fundamental tradeoff between internal control and external connection, libraries wishing to promote interoperability of digital library content have two possible strategies. The first, and most obvious, is to attempt to alter the balance currently struck between connection and control to more strongly favor connection. There are several mechanisms which the library community might employ in pursuit of this strategy, including the design and use of schemas which more significantly restrict both the means for recording the structure of objects and the ability to employ arbitrary additional schemas within instance documents (or developing profiles of existing schemas to achieve the same ends), establishing formal rules of structural description (equivalent to rules of description used in cataloging for creating bibliographic records) dictating aspects of object encoding not susceptible to enforcement through XML's validation mechanisms, and mandating the use of particular controlled vocabularies and ontologies within document instances to record information such as a `<div>` element's TYPE attribute in METS. 32

Decreasing the possibility for local variation in encoding of structural metadata will certainly help improve digital libraries' capability to interoperate with each other. However, removing local capacity for variation will also tend to reduce the number of institutions who are willing to use such a markup language. If the digital library community, for instance, was to revise the METS standard to forbid any use of a `<relation>` or `<RelatedItem>` element in a descriptive metadata 33 section to express the logical structure of a work, it would assist in insuring interoperability of digital content, but it might also very well mean losing the Library of Congress's support for the standard. More importantly, however, such an approach overlooks one of the fundamental realities of the web environment: communities of practice no longer operate in isolation from each other (if indeed, they ever did). Even if libraries could agree on a structural metadata standard that enabled a significantly greater degree of support for interoperability than we find with today's standards, libraries must now interact with a variety of other communities (publishers, museums, archives, educational technology companies, etc.) that are also creating their own structural metadata standards. This is not to say that pursuit of this strategy is futile or even inappropriate in many instances; libraries' previous experience with standard efforts such as MARC demonstrate that with sufficient time and effort a particular community of practice can achieve widespread interoperability of metadata. However, the library community's interactions with other communities clearly indicates that this strategy by itself is insufficient to resolve the interoperability problems that libraries confront today.

To deal with these wider issues of interoperability, the library community must adopt a second strategy based on accepting that the need for community control over encoding practices is a valid one, that community "dialects" of markup languages are inevitable, and that we must find ways to facilitate information exchange across the boundaries of different communities' markup vernacular. However, this will require a significant shift in the digital library community's relationship to the notion of standards. Specifically, the library community needs to shift from its current singular focus on schema development to a dual focus on both schema development *and* translation between schemas. 34

This is certainly not the case today, as can be seen if we examine the work of some of the major agencies involved in metadata standardization in libraries such as the Library of Congress. The Library of Congress currently serves as maintenance agency for a variety of XML standards developed within the library community; if you examine the list of standards that they are maintaining [Library of Congress 2008], however, you will find that while there are several metadata standards listed, standardized stylesheets to enable conversion between formats are not listed here. Such stylesheets do exist in some cases. The Library of Congress has, for example, provided stylesheets to enable conversion of MODS descriptive metadata records into MARC/XML format and back. These efforts to formalize prior work that established crosswalks between different descriptive metadata standards are not, however, seen by the community as having the status and importance of standards, as exhibited by their omission from the "Standards at the Library of Congress" web page. If the digital library community wishes to support interoperability while simultaneously affording institutions localized control over encoding practices, that situation needs to change. We can no longer view the creation of translations between standard formats as an ancillary activity; instead, we must regard it as a form of standards activity in its own right, as important, if not more important, than the creation of schemas for metadata sets. 35

A heightened emphasis on standardizing translation between markup languages will mean further work on formalizing translations between markup languages using XSLT, and treating those with the level of attention and care that the community has lavished on metadata schemas. However, it might also be worth considering whether the notion of formal rules of structural description mentioned earlier might be of benefit in trying to achieve greater translatability between different markup languages. As an example of what this might mean, consider the example of the 1:1 principle in Dublin Core [Hillmann 2005], that a single Dublin Core record should describe one and only one resource. The 1:1 principle provides guidance on the relationship between a metadata record and a described resource that is applicable outside the realm of Dublin Core; in fact, several other descriptive metadata standards developed since Dublin Core refer to the 1:1 principle as a guide to usage. We could easily envisage similar principles being developed for structural metadata that could guide usage of a variety of different structural metadata standards, and by working to insure similar use practices, would help insure ease of translation between different structural markup languages. We might, for instance, take as a working principle that any given structural metadata document should never contain more than two levels of structural hierarchy. Our METS example above passes muster with this rule; if, however, we modified it so that a third level of `<div>` elements was needed (of TYPE "subchapter," for example), then we would be in violation of this principle. To fix this problem, we could employ METS' `<mptr>` element to allow the `<div>` elements for each chapter to reference separate METS files containing the structural descriptions for the individual chapters. Through the establishment of common principles of structural encoding and standardized stylesheets for translation, we might be able to improve our ability to interoperate while simultaneously retaining some flexibility for local encoding practice 36 (although obviously adoption of common principles of structural encoding may impede local control in favor of connection to some degree).

The rise of the network information society is presenting libraries with a variety of new challenges. Perhaps the most significant of these is the heightened degree of interaction with communities of practice that do not share libraries' standards, practices or values. If libraries are to survive and thrive in this new information society, they must alter their own value structure to prioritize communication with other communities to an equal, if not greater, extent than internal communication between libraries. If they pursue this course, they may find that issues of internal interoperability of library systems are more tractable than they have appeared to date. 37

Notes

[1] As a reviewer for this article noted, the meaning of *interoperability* is vague and its interpretation highly variable depending on the community and context in which it is used. Within the discussions among the members of the digital library community upon which this article is based, interoperability is interpreted primarily as the ability to reuse data and metadata outside of the technological system in which they were originally instantiated as well as outside of the community they were originally intended to serve. Interoperability thus has both a technological and a social component.

[2] See http://www.loc.gov/standards/mets/mets-registered-profiles.html for the set of formally registered METS profiles.

[3] I am including within this set the Metadata Encoding & Transmission Standard (METS, see http://www.loc.gov/standards/mets/), the MPEG-21 Digital Item Declaration Language (DIDL, see http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm), the Open Archives Initiative Object Reuse & Exchange standard (OAI-ORE, see http://www.openarchives.org/ore/) and the Fedora Object XML specification (FOXML, see http://www.fedora-commons.org/documentation/2.2.2/userdocs/digitalobjects/introFOXML.html). There are a variety of other structural metadata standards that are of particular interest to the digital library community, although not being developed within it, including standards for data archiving such as the XML Formatted Data Unit specification (XFDU, see http://sindbad.gsfc.nasa.gov/xfdu/) being developed by the Consultative Committee on Space Data Systems and standards for structuring content for e-learning systems such as the IMS Content Packaging specification (IMS-CP, see http://www.imsglobal.org/content/packaging/) developed by the IMS Global Learning Consortium and the Shareable Content Object Reference Model (SCORM, see http://www.adlnet.gov/Technologies/scorm/default.aspx) developed by the United States Department of Defense.

[4] See http://www.loc.gov/marc/marcsgmlarchive.html for information regarding the development of an SGML version of the MARC 21 record format.

[5] Complete MODS and METS records for this example can be found at the Library of Congress webpage for the 1946 Library of Congress recital at http://lcweb2.loc.gov/diglib/ihas/loc.natlib.ihas.200003790/default.html

[6] For the complete mission state of the University of Illinois at Chicago Library, see http://www.uiuc.edu/lib/about/libmission.shtml

Works Cited

Akrich 1992 Akrich, Madeleine. “The De-Scription of Technical Objects”. In John Law and Wiebe E. Bijker, eds., *Shaping Technology/Building Society: Studies in Sociotechnical Change*. Cambridge, MA: MIT Press, 1992.

Bosak & Bray 1999 Bosak, John, and Tim Bray. “XML and the Second-Generation Web”. *Scientific American* 280: 5 (1999). http://search.ebscohost.com/login.aspx?direct=true&db=aph&an=1775198&site=ehost-live.

Bosak 1997 BosakJ. *XML, Java, and the future of the Web*. March 10, 1997. http://www.ibiblio.org/bosak/xml/why/xmlapps.htm.

Bosak 1998 Bosak, John. “Media-Independent Publishing: Four Myths about XML”. *Computer* 31: 10 (1998). http://ieeexplore.ieee.org/iel4/2/15590/00722303.pdf.

Bray et al. 1998 Bray, Tim, Jean Paoli, C.M. Sperberg-McQueen, Eve Maler and François Yergeau, eds. *Extensible Markup Language (XML) 1.0*. *W3C Recommendation 10-February-1998*. World Wide Web Consortium, 1998. http://www.w3.org/TR/1998/REC-xml-19980210.

DiLauro et al. 2005 DiLauro, Tim, Mark Patton, David Reynolds and G. Sayeed Choudhury. “The Archive Ingest and Handling Test”. *D-Lib Magazine* 11: 12 (2005). http://www.dlib.org/dlib/december05/choudhury/12choudhury.html.

Goodger et al. 2001 Warning: Biblio formatting not applied. GoodgerB. HicksonI. HyattD. WatersonC. DavidHyatt. *XML User Interface Language (XUL) 1.0*. Mozilla Foundation. 2001. http://www.mozilla.org/projects/xul/xul.html.

Hillmann 2005 HillmannD. *Using Dublin Core*. Dublin Core Metadata Initiative, 2005. http://dublincore.org/documents/usageguide/index.shtml.

Hurley et al. 1999 HurleyB. J., Price-Wilkin J., Proffitt M. and Besser H. *The Making of America II Test Bed Project: A Digital Library Service Model*. Washington, DC: Digital Library Federation, 1999. http://www.clir.org/pubs/reports/pub87/pub87.pdf.

Internal Revenue Service 2007 Internal Revenue Service. *Current Valid XML Schemas for 11/20/1120-F/1120S/7004 Modernized e-File (MeF)*. Washington, DC: United States Department of the Treasury, 2008. http://www.irs.gov/efile/article/0,,id=128360,00.html.

Keith 2005 KeithC. *Making METS Profiles Machine-Actionable to Support Validation and Facilitate Interoperability*. *Digital Library Federation Spring Forum 2005, April 13-15, 2005, San Diego, CA*. 2005. http://www.diglib.org/forums/spring2005/presentations/keith-2005-04.pdf.

Kendall 2007 Kendall, Lori. “Shout Into the Wind, and It Shouts Back: Identity and Interactional Tensions on LiveJournal”. *First Monday* 12: 9 (2007). http://www.firstmonday.org/issues/issue12_9/kendall/index.html.

Kott et al. 2006 Kott, Katherine, Jon Dunn, Martin Halbert, Leslie Johnston, Liz Milewicz and Sarah Shreeves. “Digital Library Federation (DLF) Aquifer Project”. *D-Lib Magazine* 12: 5 (2006). http://www.dlib.org/dlib/may06/kott/05kott.html.

Library of Congress 2006 Warning: Biblio formatting not applied. Library of Congress. *Library of Congress METS Profile for Recorded Events*. January 12, 2006. http://www.loc.gov/standards/mets/profiles/00000009.html.

Library of Congress 2008 Library of Congress. *Standards (Library of Congress)*. May 20, 2008. http://www.loc.gov/standards/.

Littman 2006 Littman, Justin. “A Technical Approach and Distributed Model for Validation of Digital Objects”. *D-Lib Magazine* 12: 5 (2006). http://www.dlib.org/dlib/may06/littman/05littman.html.

Nelson et al. 2005 Nelson, Michael L., Johan Bollen, Giridhar Manepalli and Rabia Haq. “Archive Ingest and Handling Test: The Old Dominion University Approach”. *D-Lib Magazine* 11: 12 (2005). http://www.dlib.org/dlib/december05/nelson/12nelson.html.

Renear & Golovchinksky 2001 Renear, Allen, and Gene Golovchinsky. “Content Standards for Electronic Books: The OEBF Publication Structure and the Role of Public Interest Participation”. *Journal of Library Administration* 35: 1/2 (2001). http://www.tandf.co.uk/journals/titles/01930826.asp.

Seaman 2003 Seaman, David. “Deep Sharing: A Case for the Federated Digital Library”. *Educause Review* 38: 4 (2003). http://net.educause.edu/ir/library/pdf/erm0348.pdf.

Shirky 2005 Shirky, Clay. “AIHT: Conceptual Issues from Practical Tests”. *D-Lib Magazine* 11: 12 (2005). http://www.dlib.org/dlib/december05/shirky/12shirky.html.

Shreeves et al. 2005 Shreeves, Sarah L., Ellen M. Knutson, Besiki Stvilia, Carole L. Palmer, Michael B. Twidale and Timothy W. Cole. “Is 'Quality' Metadata 'Shareable' Metadata? The Implications of Local Metadata Practices for Federated Collections”. Presented at *ACRL 2005*. *Proceedings of the Twelfth Annual National Conference of the Association of College and Research Libraries* (2005), pp. 223-237. http://hdl.handle.net/2142/145.

U.C. Berkeley Library 1997 The Library, University of California at Berkeley. *The Making of America II Test Bed Project Proposal*. 1997. http://sunsite.berkeley.edu/moa2/moaproposal.html.

World Wide Web Consortium 1999a Warning: Biblio formatting not applied. World Wide Web Consortium. TimBray. DaveHollander. AndrewLayman. *Namespaces in XML*. Jan. 14, 1999. http://www.w3.org/tr/1999/rec-xml-names-19990114/.

World Wide Web Consortium, 1999b World Wide Web Consortium. *The World Wide Web Consortium Issues “Namespaces in XML”” as a W3C Recommendation*. *Press Release*. January 14, 1999. http://www.w3.org/press/1999/xml-namespaces-rec.