

Introduction to XML for Decision-Makers Briefing Talking Points

- ☑ Identifies points in the script and slides where you may want to customize the briefing to include case studies and examples from your own experiences, and exercises which may be more appropriate for your audiences.

Customization Index

(listed by slide number)

Ack-1
Page 3
Page 4
Page 6
Page 10
Page 12
Page 13
Page 15
Page 16
Page 17
Page 18
Page 19
Page 20
Page 21
Page 24
Page 30
Appendices

Slide	Talking Points
<p>Briefing Background, Acknowledgements, and Contact Information</p>	<p>Briefing Background, Acknowledgements, and Contact Information</p>
<p>Ack-1 This briefing and all related materials are the direct result of a two-year grant to the State Archives Department of the Minnesota Historical Society (MHS) from the National Historical Publications and Records Commission (NHPRC). Work on the “Educating Archivists and Their Constituencies” project began in January 2001 and was completed in May 2003.</p> <p>The project sought to address a critical responsibility that archives have discovered in their work with electronic records: the persistent need to educate a variety of constituencies about the principles, products, and resources necessary to implement archival considerations in the application of information technology to government functions. Several other goals were also supported:</p> <ul style="list-style-type: none"> • raising the level of knowledge and understanding of essential electronic records skills and tools among archivists, • helping archivists reach the electronic records creators who are their key constituencies, • providing the means to form with those constituencies communities of learning that will support and sustain collaboration, and • raising the profile of archivists in their own organizations and promoting their involvement in the design and analysis of recordkeeping systems. <p>MHS administered the project and worked in collaboration with several partners: the Delaware Public Archives, the Indiana University Archives, the Ohio Historical Society, the San Diego Supercomputer Center, the Smithsonian Institution Archives, and the State of Kentucky. This list represents a variety of institutions, records environments, constituencies, needs, and levels of electronic records expertise. At MHS, Robert Horton served as the Project Director, Shawn Rounds as the Project Manager, and Jennifer Johnson as the Project Archivist.</p>	<p>Ack-1 Explain:</p> <ul style="list-style-type: none"> • slide appearance • space for notes • briefing background in brief • encourage contact with instructors • thank any partners <p><input checked="" type="checkbox"/> <i>Note to instructor: Include your contact information at the bottom of the slide</i></p> <p><input checked="" type="checkbox"/> <i>Note to instructor: It is helpful to handout all the exercises and examples at the beginning of the day, when handing out course books, so that you do not have to interrupt the briefing later. Consider copying them on sheets of different colored paper so that they are easy to distinguish from one another.</i></p>

<p>MHS gratefully acknowledges the contribution of Advanced Strategies, Inc. (ASI) of Atlanta, Georgia, and Saint Paul, Minnesota, which specializes in a user-centric approach to all aspects of information technology planning and implementation. MHS project staff received training and guidance from ASI in adult education strategies and briefing development. The format of this course book is directly based on the design used by ASI in its own classes. For more information about ASI, visit http://www.advstr.com/</p> <p>For more information regarding the briefing, contact MHS staff or visit the briefing web site at http://www.mnhs.org/preserve/records/worksshops/edarchivists.html</p>	
<p>Page-1 This briefing includes:</p> <ul style="list-style-type: none">• Briefing objectives• What do we mean by information resources, digital objects, and electronic records?• Defining digital objects• What is eXtensible Markup Language (XML)?• Why XML?• Marking up a document• Standardizing markup: Document Type Definition (DTD) and XML Schema• eXtensible Stylesheet Language (XSL)• XSL Transformations• Minnesota Electronic Real Estate Recording Task Force	<p>Page-1 Discuss the list on the slide.</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: Strongly consider substituting your own local case study in place of the Minnesota example for greater impact with your audience</i></p>

<p>Page-2 Briefing objectives</p> <p>Upon completion of this briefing, you will be able to:</p> <ul style="list-style-type: none"> • understand basic information technology concepts and terminology • understand what XML is and why it is useful • understand the reasons for the development of XML • recognize XML markup • identify other components of the XML standard • understand how XML may be implemented in a project 	<p>Page-2 Discuss the list on the slide.</p>
<p>Page-3 What do we mean by information resources, digital objects, and electronic records?</p> <p>Information resources: The content of your information technology projects (data, information, records, images, digital objects, etc.)</p> <p>Digital object: Information that is inscribed on a tangible medium or that is stored in an electronic or other medium and is retrievable in perceivable form. An object created, generated, sent, communicated, received, or stored by electronic means.</p> <p>An electronic record is a specific type of digital object with unique characteristics described by archivists and records managers.</p> <p>Types of digital objects:</p> <ul style="list-style-type: none"> • e-mail • Portable Document Format (PDF) files • web pages • PowerPoint presentations • databases • digital images • spreadsheets • word processing documents ...and many more 	<p>Page-3 As we begin to discuss XML, let's make sure we're all on the same page by defining some of the terms we'll be using today.</p> <p>We need to start with a common definition of information resources, digital objects, and electronic records in order to understand how we can use XML to manage them.</p> <p>Define Information resources as: The content of your information technology projects (data, information, records, images, digital objects, etc.)</p> <p>Our definition of digital objects comes from E-Sign</p> <ul style="list-style-type: none"> • The Electronic Signatures in Global and National Commerce Act passed by Congress in 1999 to create a common legal framework for electronic commerce and electronic government in the nation. • Digital object is defined as "Information that is inscribed on a tangible medium or that is stored in an electronic or other medium and is retrievable in perceivable form. An object created, generated, sent, communicated, received, or stored by electronic means." • Electronic record <ul style="list-style-type: none"> ◦ is a specific type of digital object with unique characteristics described by archivists and records managers. ◦ very broad and generic definition, which is exactly why it was chosen ◦ it's not exclusive to anything – records, digital objects, data, information, knowledge, all words we may use interchangeably. <p><input checked="" type="checkbox"/> <i>Note to instructor: You may want to add your own definitions here as they relate.</i></p> <p>This is a good place to start.</p> <ul style="list-style-type: none"> • In a practical sense, we need to break that definition down right away - it's too broad.

	<ul style="list-style-type: none"> • To determine what we want to manage and how, we need to be much more precise. • We also need to consider that people often think in terms of types or genres, like e-mail, web pages, databases, word processing documents, and the like. <ul style="list-style-type: none"> ◦ But it's not enough to know what application or file format a digital object is linked to if we want them to be accessible for however long we may need them, especially if we need to share them, re-use them, and/or if they are expected to outlast their original systems <p>Where do we start?</p>
<p>Page-4 Digital objects have three components:</p> <p>Content: Informational substance of the object.</p> <p>Structure: Technical characteristics of the objects (e.g., presentation, appearance, display).</p> <p>Context: Information outside the object which provides illumination or understanding about it, or assigns meaning to it.</p>	<p>Page-4 Archivists and records managers came up with a definition that applies to electronic objects, but we think it applies to all information objects, and it is applicable to anything in a digital format. Content, Structure, and Context was first defined by the Pittsburgh Project in the early 1990s which focused on helping archivists deal with Information Technology and Electronic Records. It's just one of the many ways to describe a record or an object.</p> <ul style="list-style-type: none"> • Content: Informational substance of the object. (What it says) • Structure: Technical characteristics of the objects (e.g.; presentation, appearance, display). (How the record looks) • Context: Information outside the object which provides illumination or understanding about it, or assigns meaning to it. (What it is about) <p><input checked="" type="checkbox"/> Illustration: \$20 bill example [hold one up]</p> <ul style="list-style-type: none"> • Content: basic information about the bill - \$20, serial number, image (2 Jacksons), statement about legal tender. informational substance of records • Structure: what says that this object is authentic - hologram/ghost image, hidden strip, color of the ink, feel of the paper, etc. The structure assures us that it is valid, we doubt it it's validity if any of these components is missing. • Context: information outside the object. foreign currency market – how related to other currency, what's it worth <p>Content, Context, and Structure are the necessary components to help us understand what an object is and what it's worth</p>
<p>Page-5 Defining information objects</p> <p><u>Pittsburgh Project</u> <u>Order of Values</u> <u>Information Technology</u></p>	<p>Page-5 Take those three components and compare it to some other ways to define digital objects.</p> <p>In the first column we have the scheme we just talked about.</p>

<u>Definitio n</u>	Data Informatio n	<u>ogy Architec ture</u>	Data Format
Content	Data		Data
Structure	Informatio n		Format
Context	Knowledg e		Applicat ion

Let's move to the second column and read down. Many people break down information objects in terms of the value they represent to an organization.

- one may say that it's data, information or knowledge.
- look at this column in terms of our \$20 bill again.
 - Data here is the lowest common denominator. \$20 is the equivalent of 2000 pennies, but if you wanted to purchase something for \$14.97, would you be welcome anywhere if you pulled out a bag of pennies and said, "Just wait, I have exact change!" Data is accurate, but it's not necessarily useful. It's the least functional value.
 - Information – here data is structured for more functionality; it's data presented in a practical format. A \$20 bill is more useful than 2000 pennies. It's in a specific format that's designed to be more easy to use.
 - Knowledge – is data available for a wide variety of uses. Think of \$20 in the bank. You have different ways to access it through a check or debit card. You can automatically withdraw money to pay bills. It can earn interest, or the bank can use it to loan to other people. This is the level where you get most value.

Why isn't this enough? Let's look at the third column.

- To use technology, all of these components and values have to be captured in a specific information technology architecture, a configuration of hardware and software that allows us to use computers to manage our information.
- And an architecture has three components:
 - data – the actual stored bits and bytes
 - format – data in a particular format (e.g., Word file, PDF, TIFF image) (how it looks to us)
 - application – a program that takes a particular format and puts the data to use, gives it some functionality.
- The architecture traditionally represents a limitation.
- Applications are subject to rapid obsolescence. And, they often don't do all the things we want them to do
 - we try to take data configured for one application and make it work in another. We lose a lot in the process usually. Because applications and their associated formats are usually proprietary something is almost always lost in the process. Very often, we're lucky to preserve just the data, let alone structure and functionality, given our limitations.
- The \$64,000 question for archivists and for anyone

	<p>trying to preserve an investment in information: How do we preserve the value of our asset, our knowledge, over time, when the tools that help us realize that value (the hardware and software) are so unreliable?</p> <p>That's where metadata and XML come in.</p> <ul style="list-style-type: none">• Metadata allows us to capture the pertinent information, the content, context, and structure.• XML allows us to preserve the information, to wrap records and make them useful across systems. Allows for knowledge and information in an executable format. And the format allows us to take advantage of the metadata.• The key is that XML is infrastructure-independent. You can wrap up your knowledge and apply it in different configurations of hardware and software, protect it from obsolescence and maintain its value over time. It's executable knowledge because we've captured the knowledge with the ability to use it within an architecture that is not tied to specific hardware or software.• XML might not always be the answer for you - that's one thing we will talk about. Something else will undoubtedly come along. Some other choice might be more appropriate or cost-effective for you right now. So we want to discuss when and why XML is the right choice. But it is undoubtedly a viable and effective choice. Even if there's a revolution 5 years from now and YML comes along, you'll be much better placed to take advantage of it, if you're using XML now.• This briefing is designed to give you the information that will help you make appropriate decisions about XML.
<p>Page-6 Exercise: What do you think eXtensible Markup Language (XML) is?</p>	<p>Page-6 Exercise: What do you think XML is?</p> <p>A brainstorming session,</p> <ul style="list-style-type: none">• Feel free to throw out ideas.• Don't expect an ideal definition or perfect understanding right now, because we're going to explain XML to you throughout the day.• We'll revisit this list throughout the day to understand what XML actually is and is capable of and how that fits with the preconceptions we identify here. <p><input checked="" type="checkbox"/> <i>Note to instructor: Use a flip chart to write down responses. Post the sheets so that you can refer back to them throughout the day.</i></p>

<p>Page-7 Language means communication and communication leads to understanding</p> <p>What makes understanding possible?</p> <ul style="list-style-type: none">• vocabulary• dictionary• grammar <p>It's not just semantics. This is the structure of an "unstructured" text. It is executable knowledge.</p>	<p>Page-7 Let's focus in one what you've said about XML, particularly the language and communication aspects of it. Today we're going to explore XML as part of a language. Why do we want to do this? Language means communication and communication, ideally, leads to understanding.</p> <p>What makes understanding possible? When we're talking about a language, it's the following items (think of learning a second language):</p> <ul style="list-style-type: none">• vocabulary - words• dictionary – definitions, rules of pronunciation, words you string together according to a grammar (proper meaning)• grammar – how to form sentences and create meaning and communicate with them in an accepted way <p>In order to have some common understanding, we need consensus about what we're saying. Need to put all of our ideas together in an understandable format.</p> <p>To create meaning...</p> <ul style="list-style-type: none">• It's not just semantics. This is the structure of an "unstructured" text. Turns into executable knowledge. <p>Quotation: "When I say a word, it means exactly what I want it to mean." Humpty Dumpty – Alice in Wonderland</p> <ul style="list-style-type: none">• Humpty Dumpty is one way to look at it.• He didn't care if other people understood him.• He illustrates the good and the bad about XML:• Yes, we have the potential to create our own language• But, if no one understands it and it only makes sense to you, what's the point?• With XML we want to create a language we have control over and that we will use to communicate with, otherwise you're just talking to yourself <p>The implication is XML has to mean something to everyone who needs to understand it, and building a commonly understood language is a part of this.</p>
<p>Page-8 What does eXtensible Markup Language mean?</p> <p>eXtensible: In XML, you create the tags you want to use. XML extends your ability to describe a document, letting you define meaningful tags for your applications. For example, if your document contains many glossary terms, you can create a tag called <glossary> for those terms. If it contains employee identification numbers, you could use an <employeeid> tag. You can create as few or as many tags as you need.</p>	<p>Page-8 So how do we work what we've just discussed with language into a standard definition? Let's explore what the terms eXtensible Markup Language mean</p> <ul style="list-style-type: none">• eXtensible: You get to assign value to the text. Extensibility means independence. While there are a few simple rules to follow, you are free to assign meaning to your documents as you see fit. You determine what's important about your document.<ul style="list-style-type: none">o This is a little disconcerting at first because there is no reference guide to look up what tags to use. But it also gives you great freedom and flexibility because you can define and use tags in a way that makes sense for your documents.

Markup: Any means of making explicit an interpretation of a text. In this instance, a notation for writing text with tags. The tags may indicate the structure of the text, they may have names and attributes, and they enclose a part of the text.

Language: This states that XML follows a firm set of rules. It may let you create an extensible set of markup tags, but its structure and syntax remain firm and clearly defined.

- o Extensibility means you get more options and more power - but with those capabilities comes a need for planning. You will need to plan what tags to create. (You have to use that power for good)
 - Markup: Structure of text within tags. Markup helps you keep track of that meaning. It's structure applied to unstructured data that helps you to capture the value of your information and documents. Make explicit your interpretation of a particular text.
 - Language: Based on a set of rules – define your values and markup in ways that others understand. Remember the goal of XML is communication. Language is what you use when you want to talk to someone, or tell people about your information.
- eXtensible Markup Language gives you flexibility and creativity, all within a dictionary and grammar that have been defined so that everyone understands the language you are using.

<p>Page-9 Why XML?</p> <p>Share data: Different organizations rarely use the same tools to create and read data. XML can be used to store any kind of structured information, and to enclose or encapsulate it in order to pass the information between different computing systems which would otherwise be unable to communicate.</p> <p>Reuse data: XML documents can be moved to any format on any platform - without the elements losing their meaning. This means you can publish the same information to a web browser, or a personal digital assistant (PDA), and each device would use the information appropriately. XML can be designed in such a way that fragments or chunks can be pulled out of any given context and reused. So, when a chunk is updated, the resources that use the chunk are updated also.</p> <p>Customize data: XML allows for the development of user-defined document types. Users define the XML tags they want to encapsulate their data. XML also allows groups of people or organizations to create their own customized markup languages for exchanging information in their domain.</p>	<p>Page-9 So, why do people want to use XML? Three reasons come to mind: to share, reuse, and customize data.</p> <ul style="list-style-type: none">• Share data: We've all made significant investments of time, money, and resources in our technological environments and in maintaining our data. So, how do we break down our silos of information to share data when we want to and when we have to? How do we share data without changing our legacy systems?<ul style="list-style-type: none">◦ Because different organizations (or even different parts of the same organization) rarely standardize on a single set of tools, it takes a significant amount of work for two groups to communicate. XML makes it easy to send structured data across the web/or network so that nothing gets lost in translation. When using XML, I can receive XML-tagged data from your system, and you can receive XML-tagged data from mine. Neither of us has to know how the other's system is organized. If another partner or supplier teams up with my organization, I don't have to write code to exchange data with their system. I simply require them to follow the document rules defined in the Document Type Definition, which describes how my XML documents work.• Reuse data: Technical obsolescence occurs at an amazing rate, yet we want the value of our data to remain consistent and constant. How do we best translate this data, and migrate it from system to system? Also, how do we make data available now on all the different systems we currently use: computer, phone, PDA? XML helps us to structure information so it's useful on different devices.• Customize data: We want to use data in a particular way, and we want to use it, enhance it, and package it with other data in common applications. XML allows us to get the most from our information resources that we can. <p>XML is an infrastructure-independent markup.</p> <ul style="list-style-type: none">• Used to mark up records so they can be used by different systems and not dictated by particular hardware and software configurations.• Not tied to one particular application, which is why it's so valuable for breaking down those silos of information and allowing for the long-term preservation of records, because it allows for it's use by a variety of different applications.
<p>Page-10 Marking up a document?</p> <p>Declaration: Declares what version of XML you are using. Appears first in an</p>	<p>Page-10</p> <ul style="list-style-type: none">• What do you need to know about marking up documents to make this possible? Here are the basic components.

<p>XML document.</p> <pre><?xml version="1.0" standalone="yes"?></pre> <p>Elements: The most basic unit of an XML document. The name of the element (defined by you) should assign some meaning to the content.</p> <pre><recipe> <title>Original Nestle Toll House Chocolate Chip Cookies</title> <background> <author>Ruth Wakefield</author> </background> </recipe></pre> <p>Attributes: Additional data elements that help to more accurately describe an element. Attributes have quotation-mark delimited values that further describe the purpose and content of an element. Information contained in an attribute is generally considered metadata.</p> <pre><ingredients> <item quantity="1" unit="12 oz pkg.">Nestle Toll House semi-sweet chocolate morsels</item> </ingredients></pre> <p>The decision of whether to present your information as attributes or sub-elements will depend on your business needs.</p>	<p><input checked="" type="checkbox"/> <i>Note to instructor: If you chose to substitute another example in place of the recipe, you should consider making the examples in this slide match the XML handout.</i></p> <p>All XML documents begin with a declaration that declares what version of XML you are using.</p> <ul style="list-style-type: none">• The declaration is always the first thing to appear in your XML document. Currently this is easy since there is only version 1.0, but could become more relevant in the future as the standard evolves.• Standalone in this reference means that your document does not have an accompanying DTD. Should it have an accompanying DTD, you would reference the location of that DTD here.• Notice also that the declaration is within angle brackets, this is how all XML notation works, within angle brackets.• the question mark?=processing instruction <code><? ?></code>. In this case the processing instruction tells the computer we will be using XML version 1.0 <p>Elements: the most basic unit of an XML document. The name of the element (defined by you) should assign some meaning to the content.</p> <ul style="list-style-type: none">• Elements are what go in the tags that surround your content. Add structure to the content. You create and define elements, assign meaning and structure to the text.• Notice elements may contain other elements (recipe/title), which are called sub-elements. Hierarchical relationships.• We also see here an example of the root element. Every XML document must start with a single element that holds the entire document content. In this case the root element is recipe. The recipe element may not be repeated again, and all other elements must be placed between the opening and closing root element. The root element is one descriptive word for the entire document. It's a wrapper around the rest of the tags. <p>Attributes are additional data elements that help to more accurately describe an element. Attributes have quotation-mark delimited values that further describe the purpose and content of an element. Information contained in an attribute is generally considered metadata. Here we see attributes as item quantity and unit.</p> <ul style="list-style-type: none">• Attributes go within the tag, providing further description of the content <p>The decision of whether to present your information as attributes or sub-elements will depend on your business needs. What is it you want to do, and what is your ultimate goal in using XML?</p>
--	--

Go over the XML recipe handout.

- Note to instructor: The recipe example was chosen because it is non-technical and generic enough for everyone to understand. If you feel it's more appropriate, you may create and present your own example.*
-

<p>Page-11 Standardizing Markup</p> <p>Document Type Definition (DTD)</p> <p>The document which holds the rules that govern what makes an XML document valid. A standard mechanism for defining what elements and attributes may be used in an XML document, where they may appear, and indicating their relationship to one another within the document. In other words, a DTD is the grammar of an XML document.</p> <p>XML Schema</p> <p>Specifies the structure of an XML document and constraints on its content. A schema defines the grammar of an XML document and is for validation.</p> <p>What are the benefits of XML Schemas?</p> <ul style="list-style-type: none">• XML Schema is expressed in well-formed XML. DTDs are not expressed in XML language.• XML Schema gives you all the functionality of XML for sharing, re-using and customizing the grammar and dictionary of your mark-up language. XML Schema allows you to change schemas easily and without affecting the already formatted documents in XML.• Offers an extensive system of datatypes that you can specify for a given element. For example, an element may be an integer, contain a period of time, contain a string, boolean, a language code, etc. DTDs are unable to restrict character data to a pattern.	<p>Page-11</p> <p>On the previous slide we said that if the XML declaration contains a standalone reference, that means it is entirely self-contained. Well, what are the other options? Your XML document may also refer to a Document Type Definition (DTD), or XML Schema.</p> <p>We've been referring to the need to have some standard for the markup we are using. Why?</p> <ul style="list-style-type: none">• Because of the need to share information. In order to share information, someone else has to have a way to understand the system your are using, and your systems and computers have to be able to speak to each other – using a common language. <p>With XML, there have been two methods designed to do this.</p> <ul style="list-style-type: none">• They function similarly to a dictionary and grammar, because they contain agreed upon rules that define the meaning of your XML documents and describe how they should be used.• The two methods are document type definitions (DTDs) and XML Schema.<ul style="list-style-type: none">◦ They serve the same purpose, to define elements and attributes and how they are used, and to define what may be contained in elements.◦ They serve the common purpose of communication. <p>How do they function?</p> <ul style="list-style-type: none">• A computer doesn't understand strings of text, like it would get in a XML document, but with a DTD or XML Schema that defines allowed parameters, a computer knows what to do with the XML document.• In most applications, if your XML documents do not conform to a stated DTD or XML Schema you will receive an error and your system will be unable to work with this XML document. <p>DTD and XML Schema have a similar function, but they have different potentials.</p> <ul style="list-style-type: none">• DTD was inherited by XML because it was used with XML's predecessor, Standard Generalized Markup Language (SGML) (for more about SGML, please see Appendix A). And, DTDs work, but they have limitations.• XML Schema is an alternative approach. With schemas, you are able to further specify allowed content and constraints on elements. It's a more robust grammar. And, schemas offer things DTDs do not<ul style="list-style-type: none">◦ expressed in XML◦ further control over allowed datatypes and the content of elements◦ the ability to bring in more than one schema• Both DTDs and XML Schema allow you to define the dictionary and grammar for your XML documents. You
---	---

	<p>reap more benefits from XML if you use one or the other, but you will have to decide which one you want to use. You have to decided whether DTDs or Schemas are going to meet your needs. You will also be influenced by which our stakeholders and partners are working with.</p>
<p>Page-12 Example: Document Type Definition of a recipe</p> <pre><!DOCTYPE recipe[<!ELEMENT recipe (title, background, recipe_info, nutritional_info, comments, ingredients, directions)> <!ELEMENT title (#PCDATA)> <!ELEMENT background (author, history)> <!ELEMENT author (#PCDATA)> <!ELEMENT history (#PCDATA)> <!ELEMENT recipe_info (prep_time, cook_time)> <!ELEMENT cook_time (#PCDATA)> <!ELEMENT prep_time (#PCDATA)> <!ELEMENT nutritional_info (calories, fat, protein, carbohydrates, cholesterol, sodium, fiber)> <!ELEMENT protein (#PCDATA)> <!ELEMENT calories (#PCDATA)> <!ELEMENT carbohydrates (#PCDATA)> <!ELEMENT sodium (#PCDATA)> <!ELEMENT cholesterol (#PCDATA)> <!ELEMENT fat (#PCDATA)> <!ELEMENT comments (#PCDATA)> <!ELEMENT fiber (#PCDATA)> <!ELEMENT ingredients (item+)> <!ELEMENT directions (directions_standard, directions_variation+)> <!ELEMENT item (#PCDATA)> <!ELEMENT directions_standard (step+)> <!ELEMENT directions_variation (variation_name+, step+, variation_comment?)> <!ATTLIST item quantity CDATA #REQUIRED unit CDATA #REQUIRED> <!ELEMENT step (#PCDATA)> <!ELEMENT variation_comment (#PCDATA)> <!ELEMENT variation_name (#PCDATA)>]></pre>	<p>Page-12 Exercise: Document Type Definition of a recipe to which our XML recipe document complies.</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you choose to replace the recipe example earlier with one of your own, you should customize this example to match.</i></p> <ul style="list-style-type: none"> • Each element and attribute has been declared. They must be declared or the machine won't recognize the elements or attributes in the XML document when it is processing it. DTDs are necessary for a computer to make use of an XML document. DTDs state what's allowable in XML document, and tells the computer what to expect. DTDs contain the grammar and the rules for our XML. • PCDATA and CDATA mean the same thing only the former is applied to elements and the latter is applied to attributes. They stand for parsed character data and character data, respectively, and mean that additional element tags are not allowed. In general these elements and attributes will be filled in with text. • Required: means what it says, that attribute is required. Other modifiers that exist are: <ul style="list-style-type: none"> ◦ default value, meaning there is a default value that is passed on unless told otherwise ◦ implied, meaning that the element can omit the attribute even when no default value is provided ◦ fixed, meaning that if that attribute is supplied than it must match whatever the fixed definition is. • Also take a look at ingredient or step, we see a little + means 1 or more. <ul style="list-style-type: none"> ◦ A * means 0 or more. ◦ A ? means 0 or 1. • All of these things constrain or define your data.
<p>Page-13 Example: Schema of a recipe</p>	<p>Page-13 Exercise: Schema of a recipe</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you choose to replace the recipe</i></p>

example earlier with one of your own, you should customize this example to match.

Look at hand out. What we see is a schema for a recipe to which our XML recipe document conforms.

- root element of schema, declares the schema (<schema> is the root element of EVERY XML schema)
 - namespace: source that the elements used in the schema are from. Also lets you know that the elements and data types from this 'namespace' should be prefixed with xsd:, which you can see throughout the rest of the document
 - elementFormDefault="qualified" means that any elements used in the XML document that were declared in the Schema must be qualified in the namespace being used

Elements and attributes

- Elements:
 - simple – contains data (text, number or date). string
 - complex – contains other elements or attributes. xsd:complexType
- attributes xsd:element ref="item"
- name- unique name of element
- content
 - empty (remain empty)
 - textOnly (only text, no sub-elements)
 - eltOnly (only sub-elements, no text)
 - mixed (text or sub-elements)
- order
 - one (only one set of elements is allowed)
 - seq (elements must appear in the sequence defined by the schema)
 - many (elements may appear in any order, any number of times, includes may not appear at all).
- minOccurs –
 - 0 (optional, not required)
 - 1(at least one is required),
- max Occurs
 - 1 (only one)
 - * (unlimited)
- dt:type – optional, one of the many data types available , such as entity, entities, enumeration, id/idref/idrefs, nmtoken/nmtokens, notation, string

This specific schema is just an example, there are many ways to do it.

- This is a basic standard form that is customizable to your needs.
- Not necessarily plugable into anything else with the expectation that it will work immediately.

	<p>You can see why some prefer DTDs over schemas, they are very verbose, but they use XML syntax.</p>
<p>Page-14 Presenting XML</p> <p>eXtensible Stylesheet Language (XSL)</p> <p>A language for expressing stylesheets.</p> <p>Stylesheet: A definition of a document's appearance or layout in terms of such elements as default typeface, size, and color of headings and body text, how sections are laid out in terms of space, line spacing, margin widths on all sides, spacing between headings, etc. Typically expressed at the beginning of an electronic document. May be embedded in or linked to a document.</p> <p>XSL Transformations (XSLT)</p> <p>A language for transforming XML documents. A tool which uses XSL to act on XML documents. XSLT is used to transform XML document contents into something else more suitable for a particular task.</p> <p>Why would we want to transform a document from one format into another?</p> <ul style="list-style-type: none"> • store in one format, display in another • convert to a more useful format 	<p>Page-14</p> <p>XML only represents potential, it allows you to assign meaning to text. But, it doesn't really do anything. You need something else in order to share and use your content. Which is why a family of applications and extensions has been developed for XML. We're going to look at one of these applications, one example of a function, used for presentation, particularly over the web. They provide functionality.</p> <p>XSL or eXtensible Stylesheet Language is used to create stylesheets, which define how a particular document will appear.</p> <ul style="list-style-type: none"> • A language that enables stylesheets to be attached to XML documents. • The great potential of XSL is translating XML to HTML (Hypertext markup language, the code that builds all of the web pages we see) and for presentation over the web. • A stylesheets describes how various elements in our XML document will appear online. <p>XSLT or XSL Transformations provides the language for actually doing the transformations.</p> <ul style="list-style-type: none"> • XSLT combines the XML document and the stylesheet to produce something useful, for example our new HTML web page. • XSLT will transform our XML document in accordance with the rules in our stylesheet. <p>XML has a lot of potential, but you need something else to help you realize that potential.</p>
<p>Page-15</p> <p>Example: eXtensible Stylesheet Language (XSL) of a recipe</p> <pre><?xml version="1.0" encoding="UTF-8"?> <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/ Transform"> <xsl:template match="/"> <html> <head/> <body><p>Shopping List for: <xsl:value-of select="recipe/title"/></p> <xsl:for-each select="recipe/ingredients/item"> <p><xsl:value-of select="@quantity"/> <xsl:text> </xsl:text> <xsl:value-of select="@unit"/></pre>	<p>Page-15</p> <p>Exercise: XSL of a recipe</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you choose to replace the recipe example earlier with one of your own, you should customize this example to match.</i></p> <ul style="list-style-type: none"> • What we see is a mixture of XML syntax and HTML coding • Templates are applied to describe how various parts of the recipe should appear, and they are modified with HTML code. <p>In this example we want to extract a shopping list from our recipe example, and then display it in a browser.</p> <ul style="list-style-type: none"> • We can see that it uses some HTML. The document calls in templates which define what happens to different parts of the documents, but instructions for display are described in HTML. The templates say

<pre> <xsl:text> </xsl:text> <xsl:value-of select="."/></p> </xsl:for-each> </body></html> </xsl:template> </xsl:stylesheet> </pre>	<p>which parts of the XML document are called in</p> <ul style="list-style-type: none"> o for example, the title element, which we make bold o for example, the item element o for example, the attributes quantity and unit o each are displayed in a particular order
<p>Page-16 XSLT example</p>	<p>Page-16 How does it show up in a browser</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you choose to replace the recipe example earlier with one of your own, you should customize this example to match.</i></p> <ul style="list-style-type: none"> • Shopping list of ingredients for our recipe • Header/Title is a particular size and color • Ingredients list quantity and item in paragraph form
<p>Page-17 Exercise: A Markup exercise</p> <p>A joke.</p> <p>Two North Dakotans come into a bar, slapping each other on the back, laughing, clearly happy as clams. One says to the bartender, "We're celebrating! Give everybody a round on us!"</p> <p>The bartender says, "So what's the big deal? What are you celebrating?"</p> <p>And the North Dakotan says, "We just finished a jigsaw puzzle and it only took us four days."</p> <p>The bartender says, "A jigsaw puzzle? Two people? Four days? That doesn't sound like much reason to celebrate."</p> <p>And the other North Dakotan says, "Are you kidding? The box said '2-3 Years.'"</p>	<p>Page-17 Exercise: Markup of a joke XML by doing our own markup language.</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: A joke was chosen for this exercise since it is less intimidating than a technical example. Edit or replace it, if appropriate, to suit your audience's sensibilities and needs.</i></p> <p>{Read joke}</p> <p>Think about what it is that you want to do with XML, what's important, what markup do you want to capture?</p> <p>Split into groups of 3-4 people (or a manageable number of groups)(work with those around you) and take about 10-15 minutes to mark-up the joke as you see fit.</p> <ul style="list-style-type: none"> • What would you mark-up and why? • Remember that you can make up your own tags, and that there is no one right answer. • When you're finished we'll discuss what you did, and then share some examples of our own. <p>[After the participants have presented their markups] There are different ways to do the markup. What have we seen? By semantic values, by categorizing funny things, such as type of joke, ethnic subjects, punchline</p> <p>What functions or business purposes are you trying to provide with your markup? Why did you mark-up the joke the way you did? What do you want to do with the data? What meaning are you assigning to the data?</p> <p>As you can see, it's a challenge to get different groups to completely agree on what they feel is important to capture. This illustrates how you get to the next step – building consensus, at which point you need to hammer out some consensus and agreement so your new</p>

<p>Page-18 Markup examples</p> <pre> <text> <paragraph><sentence type="expository">Two North Dakotans come into a bar, slapping each other on the back, laughing, clearly happy as clams.</sentence> <sentence type="exclamation">One says to the bartender,<quotation> "We're celebrating! Give everybody a round on us!"</quotation></sentence></paragra ph> <paragraph><sentence type="question">The bartender says, <quotation>"So what's the big deal? What are you celebrating?"</quotation></sentence> </paragraph> <paragraph><sentence type="expository">And the North Dakotan says, <quotation>"We just finished a jigsaw puzzle and it only took us four days."</quotation></sentence></parag raph> <paragraph><sentence type="other">The bartender says, <quotation>"A jigsaw puzzle? Two people? Four days? That doesn't sound like much reason to celebrate."</quotation></sentence></p aragraph> <paragraph><sentence type="other">And the other North Dakotan says, <quotation>"Are you kidding? The box said '2-3 Years.'"</quotation></sentence></par agraph> </text> </pre>	<p>community and partners can all use XML the way you want to.</p> <p>Page-18 Different ways to markup this joke.</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you chose to alter or replace the joke on the previous slide, edit this slide to match.</i></p> <p>Straightforward structural markup.</p> <ul style="list-style-type: none"> • start by letting you know that it's text. • identify all the paragraphs and sentences and what types of sentences there are. • identify quotations.
<p>Page-19 Markup examples</p> <pre> <story> <setting>Two North Dakotans come into a bar, slapping each other on the back, laughing, clearly happy as clams.</setting> <dialogue><character1>One</character1> says to the <character2>bartender</character2>, "We're celebrating! Give everybody a round on us!" </pre>	<p>Page-19 A story.</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you chose to alter or replace the joke on the previous slide, edit this slide to match.</i></p> <ul style="list-style-type: none"> • identify the setting. • identify dialogue. • identify all our characters

<p>The bartender says, "So what's the big deal? What are you celebrating?" And the North Dakotan says, "We just finished a jigsaw puzzle and it only took us four days." The bartender says, "A jigsaw puzzle? Two people? Four days? That doesn't sound like much reason to celebrate." And the <character3>other North Dakotan</character3> says, "Are you kidding? The box said '2-3 Years.'" </dialogue> </story></p>	
<p>Page-20 Markup examples</p> <p><humor> <joke taste="questionable"> Two <ethnic subject>North Dakotans</ethnic subject> come into a bar, slapping each other on the back, laughing, clearly happy as clams. One says to the bartender, "We're celebrating! Give everybody a round on us!" The bartender says, "So what's the big deal? What are you celebrating?" And the North Dakotan says, "We just finished a jigsaw puzzle and it only took us four days." The bartender says, "A jigsaw puzzle? Two people? Four days? That doesn't sound like much reason to celebrate." <punchline>And the other North Dakotan says, "Are you kidding? The box said '2-3 Years.'"</punchline> </joke> </humor></p>	<p>Page-20 A compendium of funny things (as opposed to tragic things)</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: If you chose to alter or replace the joke on the previous slide, edit this slide to match.</i></p> <ul style="list-style-type: none"> • jokes set apart from puns, • able to identify the ethnic subjects • the tone and to search through the punch lines, since that's a significant element of the structure - minimal markup. • start by letting you know this is something humorous. Specifically a joke of questionable taste. • identify an ethnic subject to further clue you into the type of joke this will be. • identify the punchline. <p>Why did we do it this way? Well people often remember, say, I heard a great joke about a North Dakotan, or, I know a joke and the punchline went like this...</p> <p>I wanted to keep a record of jokes, these may be some of the things you would search on.</p> <p>The joke exercise illustrates the different needs, breakdowns, and potentials that arise with XML; it demonstrates what language does, and what you need to make a language work (a vocabulary, a dictionary); and it demonstrates the different functions we will use XML to fulfill.</p>
<p>Page-21 Using XML in a program</p> <ul style="list-style-type: none"> • A common language needs a: <ul style="list-style-type: none"> ◦ Vocabulary ◦ Dictionary ◦ Grammar ◦ And an educational system • A successful XML project needs a: <ul style="list-style-type: none"> ◦ Compelling business need 	<p>Page-21 Now we will move from talking about XML to talking about XML as implemented in a program, finally wrapping up with a look at a specific program the Minnesota Electronic Real Estate Recording Task Force (ERERTF).</p> <p><input checked="" type="checkbox"/> <i>Note to instructor – Strongly consider using a local example appropriate for your audience</i></p>

<ul style="list-style-type: none">o Collaborative communityo Practical applicationo And a very large investment in people, time, money and knowledge	<p>XML is standard and a tool, and a technology for data sharing and use. In order to implement an XML project, you need to start with business needs.</p> <p>Language is a necessity. Essentially XML is a second language, which requires</p> <ul style="list-style-type: none">• a vocabulary,• dictionary,• grammar,• education system for people to learn it themselves and teach others how to put it to use. <p>Similarly, a successful markup language project needs the following components:</p> <ul style="list-style-type: none">• Compelling business need – to be persuasive• Collaborative community - to work together• Practical application – to put XML to use• A very large investment in people, time, money and knowledge – and this isn't easy, the larger the scale of the project, the larger the investment will need to be
--	---

<p>Page-22 Business needs</p> <p>Data sharing</p> <p>Infrastructure independent applications</p> <p>Web-based transactions</p> <p>Improved business processes</p> <p>Legal mandates</p> <p>Preservation</p> <p>The first concern is having a real application or business need that XML may help fulfill. The second step is developing the appropriate XML language.</p>	<p>Page-22 What business needs have been used to justify investments in XML?</p> <ul style="list-style-type: none"> • Data sharing - silo metaphor: how does the FBI share data with the CIA and local law enforcement? We know that all those groups are not going to buy the same hardware and software, so we need something that will work independently of any specific architecture. XML is designed to share data across applications and systems. • Infrastructure independent applications – We cannot change already existing investments, so XML may be the best way to connect across technologies. If technology changes, we want to be able to move from one application to another without data loss and without losing our investment in our information, our knowledge. • Web-based transactions - The most logical starting point for sharing data across organizations is use of the web - it's already there and people already have access to it. XML is designed to work with the web. Because of e-government and e-commerce there is an interest in using the web for automated transactions which will allow more people to do more things online. • Improved business processes – There is a lot of information out there, and many communities that need to process more information. • Legal mandates – Expectations in meeting our legal mandates may lead us to choose to work with XML. • Preservation - And, finally, near and dear to our hearts is preservation. The archives and records management communities have often talked about migration and conversion of records as the way to ensure access and preservation in the face of rapidly changing technology. We must preserve digital information and our investment over time, in spite of obsolescence. XML will certainly foster migration and conversion. <p>The first concern is having a real application or business need that XML may help fulfill. The second step is developing the appropriate XML language.</p>
<p>Page-23 Legal mandates</p> <p>E-Government Act of 2002</p> <p>“4) enterprise architecture (A) means</p> <p>(i) a strategic information asset base, which defines the mission;</p> <p>(ii) the information necessary to perform the mission;</p>	<p>Page 23 One of the items noted under business needs was “legal mandates.” In government, our business needs are very often defined by legal mandates, in the statutes that define our missions and functions.</p> <p>Various legislative concerns about developing e-government, the cost of information technology projects, the problems of “doing more with less,” and so on, have resulted in more business needs being translated into specific mandates.</p> <p>We can see how XML ties into some specific examples of</p>

<p>(iii) the technologies necessary to perform the mission; (iv) the transitional processes for implementing new technologies in response to changing mission needs”</p> <p>“(6) interoperability means the ability of different operating and software systems, applications, and services to communicate and exchange data in an accurate, effective, and consistent manner;”</p> <p>“(7) integrated service delivery means the provision of Internet-based Federal Government information or services integrated according to function or topic rather than separated according to the boundaries of agency jurisdiction”</p>	<p>that by looking at a federal law, the E-Government Act of 2002, and a federal law, E-sign, that has a counterpart in many states, UETA. Let’s look at some clauses from each law that are worthy of note.</p> <p>Take a look at this definition of “enterprise architecture.” There are four components and three of them immediately call XML to mind.</p> <ul style="list-style-type: none"> • “A strategic information asset base” refers to information being our most important asset. <ul style="list-style-type: none"> ◦ In fact, Minnesota’s technical enterprise architecture starts with the phrase “information is our most important asset.” • Additionally, we have “the information necessary to perform the mission.” <ul style="list-style-type: none"> ◦ With XML we’re assigning some meaning, some context to our information. ◦ XML allows for continued access to our information. ◦ It’s additional information tied to performing the mission - let’s just call it executable knowledge. ◦ With XML we turn information into knowledge, we add meaning and structure to our knowledge to make it executable knowledge. • And finally, “transitional processes for implementing new technologies” – <ul style="list-style-type: none"> ◦ what’s the bridge, the middleware to get from one system to another? XML
<p>Electronic Signatures in Global and National Commerce Act (E-Sign)</p> <p>“A Federal regulatory agency shall not adopt any regulation, order, or guidance described in paragraph, and a State regulatory agency is preempted by section 101 from adopting any regulation, order, or guidance described in paragraph, unless--</p> <p>(iii) the methods selected to carry out that purpose do not require, or accord greater legal status or effect to, the implementation or application of a specific technology or technical specification for performing the functions of creating, storing, generating, receiving, communicating, or authenticating electronic records or electronic signatures.”</p>	<p>That’s underscored when we look at interoperability and integrated service delivery. We could have an architecture based on the principle of every agency buying the same hardware and software and using it all the same way, but that’s clearly an impossible dream. So these two principles - interoperability and integrated service delivery - assume that we’ll be working with different operating and software systems, etc., but we’ll still want to share and exchange information and provide services across agencies.</p> <ul style="list-style-type: none"> • XML is the best option right now for that. <p>See it again in E-Sign, the federal equivalent, and UETA, which has been enacted by many states. These laws basically legitimate the use of electronic records. They say that government agencies can specify certain performance measures or functionalities as long as these “do not require ... a specific technology or technical specification.”</p> <ul style="list-style-type: none"> • This seems to point to XML. • It’s a standard that doesn’t require any specific hardware or software, because, again, it’s infrastructure independent. • It’s knowledge executable in any number of configurations of hardware and software. <p>What’s the conclusion?</p> <ul style="list-style-type: none"> • Business needs and legal mandates are pointing to

	<p>XML.</p> <ul style="list-style-type: none"> • Have to remind ourselves that this is a dynamic situation, but XML is right now the best bet to realize the goals these laws describe. <p>So let's see how this plays out in real life. We'll quickly look at a case study ...</p>
<p>Page-24 Case study: MN electronic real estate recording task force</p> <p>Task force formed 2000 Project to end 2004 Funded by filing fee surcharge Private-public partnership Entirely voluntary</p>	<p>Page-24 Case study: Minnesota Electronic Real Estate Recording Task Force (ERERTF). Let's take a look at a case study for some practical lessons in XML development</p> <p><input checked="" type="checkbox"/> <i>Note to instructor – Strongly consider using a local example appropriate for your audience.</i></p> <p>Real estate recording has tremendous potential if it's automated.</p> <p>Minnesota is one of the first states to try a statewide standard.</p> <ul style="list-style-type: none"> • The task force formed in 2000: A state senator with an interest in IT and the head of the governor's election campaign (a former Real Estate Lawyer who hated the paper work) got the Secretary of State to champion the project, and approval from the legislature to form the task force to explore how to automate the system. They did different studies. And, by levying a \$1 filing fee surcharge on all closings has made up the \$1.2 million funding for the project. • The project will end in 2004. • A private-public partnership – everybody who's interested is invited to join the project (lawyers, archivists, banks, secondary mortgage market, title companies, counties recorders and treasurers) • Entirely voluntary – can't force anyone to be at the table

<p>Page-25 Case study: Minnesota Electronic Real Estate Recording Task Force</p> <p>What do we mean by recording and electronic recording?</p> <p>Recording:</p> <ul style="list-style-type: none"> Recording is the act of entering deeds, mortgages, easements, and other written instruments that affect title to real property into the public record. The purpose of recording is to give notice, to anyone who is interested, of the various interests that parties hold in a particular tract of land. Recording determines the legal priority of instruments that affect title to a particular tract of land. <p>Electronic recording:</p> <ul style="list-style-type: none"> A publicly owned and managed county system, defined by statewide standards, that does not require paper or “wet” signatures, and under which real estate documents may be electronically: Created, executed, and authenticated; Delivered to and recorded with, as well as indexed, archived, and retrieved by, county recorders and registrars of title; and <ul style="list-style-type: none"> Retrieved by anyone from both on- and off-site locations. 	<p>Page-25 What do we mean by recording?</p> <ul style="list-style-type: none"> Recording – as it is now, there is a general standard across the country with minimum variations. In Minnesota, the County Recorder functions as an archivist; they keep the information permanently. Electronic recording will take paper, as the current medium of exchange and electronically connect everyone involved. Has to be publicly owned. In Minnesota they can't get rid of the county recorders because the county system is in the state constitution, but there has to be a standard so that there aren't 87 (the number of counties in Minnesota) different ways to do it in Minnesota. Need to develop one statewide standard And give everyone a piece of the action so the everyone potentially benefits from the project.
<p>Page-26 Case study: Minnesota Electronic Real Estate Recording Task Force</p> <p>What are the business needs?</p> <ul style="list-style-type: none"> Huge and increasing volume of filings Highly inefficient paper workflow between automated activities Secondary mortgage market demand for digital records Increasing complexity of property rights and descriptions Legislative mandate to develop common technical and information architectures 	<p>Page-26 What are the business needs?</p> <ul style="list-style-type: none"> Huge and increasing volume of filings – different property is passed back and forth, and if interest rates fall there are increased filings Highly inefficient paper workflow between automated activities – no way to manage more paper more effectively Secondary mortgage market demand for digital records - Fannie Mae is a \$400 billion/year industry. Time is money. If they can turn over mortgages faster, they can make more money. Time is a big deal, particularly in large counties where they're unable to deal with the large amounts of paper Increasing complexity of property rights and descriptions – transactions take place over property all the time, but you also have to be able to keep track of what rights have applied and do apply over time Legislative mandate to develop common technical and information architectures – the state legislature is annoyed that they are continually asked to fund

	<p>hardware and software implementations year after year without seeing any tangible results. They want state agencies and functions to break down the silos and develop common technologies that foster information sharing and reuse.</p>
<p>Page-27 Case study: Minnesota Electronic Real Estate Recording Task Force</p> <p>What are the options?</p> <ul style="list-style-type: none"> • Level 1: images and minimal metadata (scan paper documents and send them in an e-mail message that is manually processed) • Level 2: images, metadata, digital or digitized signature (scan documents with some substantive metadata that can be automatically processed) • Level 3: so-called “smart” documents in XML format, following recognized standards (create documents in an XML format which can be entirely processed automatically, populating tract and grantor/grantee indexes, calculating taxes and fees, validating legal descriptions, forwarding approvals and authorizations to different offices, etc.) 	<p>Page-27 XML is not the only answer</p> <p>Minnesota looked at three different options.</p> <ul style="list-style-type: none"> • Someone can just scan paper documents and send them in an email message that is manually processed (level 1); <ul style="list-style-type: none"> ◦ Speeds up the transmission process and real benefits are gained primarily in how items are exchanged. It’s cheap, and we have not really changed any business processes. • Scan documents with some substantive metadata that can be automatically processed (level 2); <ul style="list-style-type: none"> ◦ Automate more processes and benefits gained in terms of time • These two are not technically demanding and can be done without a great deal of investment and re-engineering. Very easy to implement • Option #3: Create documents in an XML format which can be entirely processed automatically, populating tract and grantor/grantee indexes, calculating taxes and fees, validating legal descriptions, forwarding approvals and authorizations to different offices etc (level 3). <ul style="list-style-type: none"> ◦ “smart documents” are only as smart as the people who design it and the applications that put it to use ◦ XML to entirely automate ◦ Has the most potential, but requires the most investment of time, people, and money
<p>Page-28 What are the problems?</p> <p>Ordinary challenges to re-engineering Extraordinary political challenges to re-engineering Connectivity to existing systems Resources Setting standards Role of the archivist</p>	<p>Page-28 What are the problems?</p> <ul style="list-style-type: none"> • Ordinary challenges to re-engineering – resistance to change, learning something new, doing something different • Extraordinary political challenges to re-engineering – the political framework: laws, system, elected officials. <ul style="list-style-type: none"> ◦ political implications: they can’t cut counties out of middle-management; also elected officials: how many individuals will be driven out of work by efficiency? • Connectivity to existing systems – XML ‘s potential is realized by having different applications all work with XML. Currently 87 counties do 87 different processes • Resources - money, and minimum number of people who understand XML • Setting standards – agreement is impossible. This is a voluntary process. There are people at the table who don’t work well together and who don’t like each other • Role of the archivist – a State Archivist is involved because the senator, governor’s campaign manager, and secretary of state were aware that the State

	<p>Archives was working with technology. They were also aware that the task force is talking about permanent records, even if they don't come to the State Archives. State Archives has a special expertise, and is the sole party with nothing at stake and nothing to gain – which makes them the honesty broker - the sole objective source in the entire process. The Archivist can represent the public good, provide information on the long-term preservation of records, and take on new roles of honesty broker, translator, and interpreter. However, it's tiring, and State Archivist expected to fill roles most staff have never been trained for.</p>
<p>Page-29 How are we using XML?</p> <p>Business and workflow analysis Data and process models National standards Schemas XSLT</p>	<p>Page-29 How are we using XML?</p> <ul style="list-style-type: none"> • Didn't have to use XML, but it offered the most potential • Business and workflow analysis - \$200000 to do this, huge upfront cost. Studied who creates documents, what do they create, how exchanged • Data and process models – developed from the business and workflow analysis. How the analysis was represented, what was important to capture • National standards – MISO and PRIJTF – standards developed outside of any practical application and are not used anywhere. PRIJTF also copyrighted and charges for use. Minnesota standards will be as close to the national standard as possible • Schemas – not DTDs More effectively able to move between different types of markup • XSLT – translate documents from XML into TIFF • Case study conclusions: <ul style="list-style-type: none"> ◦ changed our practices. Talked about XML as a way of becoming independent of technology • However, XML makes Minnesota's ERER absolutely dependent on collaboration, on a very large scale. • ERERTF uses XML Spy (mentioned in Appendix C)

<p>Page-30 You will have to make choices. One person or organization cannot do it all</p> <p>The first choice is XML right for me?</p> <p>Components: X + Y + Z = resources, tools, standards/methodologies, education, technology, partners</p> <p>Variables for the Electronic Real Estate Recording Task Force:</p> <ul style="list-style-type: none"> • Resources: consultants with business analysis and XML expertise, a special surcharge on filing to generate sufficient funds, volunteers with subject matter expertise. • Tools: business analysis, communication, negotiation, web sites, facilitated meetings, pilots, subcommittees. • Standards/methodologies: XML, XML Schemas, national standard with local extensions. • Education: not enough of it, no common basis of understanding of the choices. • Technology: web-based transactions, digital signatures, imaging, XML as middleware between legacy applications. • Partners: banks, realtors, title companies, Fannie Mae, archivists, county officials, state agencies, legislators ... 	<p>Page-30 How do you actually apply XML in your work environment?</p> <p>Given the sheer amount of electronic records created by any one person or agency/organization you are going to have to make choices. You will have to decide how best to balance the following variables and understand what options are available to you.</p> <p>First you need to decide at the beginning if XML is right for you.</p> <p>In order to bridge XML from a concept, you're going to need to be able to implement it as a program.</p> <ul style="list-style-type: none"> • Technology within a context. • To do this you need a sustainable plan. Something which moves you from the conceptual to the actual, and sets you up for long-term success. This is an outline of the variables you will have to address in getting you from here to there. <p>Every organization will place varying levels of importance on the following components, but in general you need to have all of the following, to some degree, in order to have a successful, sustainable plan.</p> <ul style="list-style-type: none"> • You will need to ask yourself, what is appropriate for you and your organization. • You start to fill in the variables, which are all dependent on your environment and your needs: <ul style="list-style-type: none"> ◦ resources, tools, standards/methodologies, education, technology, and partners • We have filled in our variables here with the components from the Minnesota Electronic Real Estate Recording Task Force (ERERTF) <p>{Go over the components.}</p> <p><input checked="" type="checkbox"/> <i>Note to instructor: Fill in the variables to reflect your case study</i></p> <p>The plan is very scaleable. This will fit any size organization. All archivists will have to make decisions about these components and how they fit into their program.</p> <ul style="list-style-type: none"> • XML is a choice, and just one of the many choices you have to make • Keep in mind that one size does not fit all. • You can't take one model and set it up –what worked in Minnesota may not work for you. • You need to make decisions about what's appropriate for you and your partners and your environment.
<p>Page-31 Thank you</p>	<p>Page-31 Thank you! Any questions or Discussion?</p>
	<p><input checked="" type="checkbox"/> <i>Note to instructor: You may choose to use the current appendices or change them for your audience</i></p>

Appendix A: Comparing SGML, HTML, and XML	Appendix A: Comparing SGML, HTML, and XML
Appendix B: List of Acronyms	Appendix B: List of Acronyms
Appendix C: XML Tools and Editors	Appendix C: XML Tools and Editors that are free and that are available for purchase
Appendix D: Bibliography	Appendix D: Bibliography