XML Tutorial

Thomas G. Habing

Grainger Engineering Library Information Center

E-Text Working Group
2004-01-15

Presenters

Tom Habing

Research Programmer, Grainger Engineering Library Information Center

thabing@uiuc.edu

Agenda – XML Tutorial

1. Well-formed XML (basic markup)
2. DTDs and schemas
3. Stylesheets
4. XML and databases

Topics include illustrations, demos, and exercises.

1. Well-formed XML

XML 1.0 Specification

- Markup
  - Prolog & Document Type Declaration
  - Elements
  - Attributes
- Content
  - Entities
  - Encoded (Unicode) characters

Prolog & Document Type Definition

- XML documents should begin with an XML Declaration which specifies version
  - Optionally may also include:
    - Encoding (recommended)
    - Stand-alone declaration
- Document Type Definition is typically next

```
<?xml version="1.0" encoding='UTF-8' standalone='no' ?>
<!DOCTYPE root SYSTEM "myDocs.dtd" >
```

Elements

Elements are markup that enclose content

- <element_name>...</element_name> or <element_name />
- Content models
  - Parsed Character Data Only
  - Child Elements Only
  - Mixed
  - Empty

```
<author>Cole, T</author>
```

Attributes

Associate a name-value pair with an element

- <tag name1="value1" name2='value2'>...</tag>
  - Can be used to embellish content...
  - or to associate added content to an element
- Attributes beginning XML are reserved

```
<author order='1'>Cole, T</author>
<author name='Habing, T' />
```

Entities

- Placeholders for internal or external content
  - Placeholder for a single character...
  - or string of text...
  - or external content (images, audio, etc.)
- Implementation specifics may vary

```
<!ENTITY copyright "&#xA9;" >
&copyright; is replaced by ©
<!ENTITY pic SYSTEM "mugshot.gif" NDATA gif >
&pic; is replaced by graphic image
```

Character Encoding Issues

- XML Parsers must accept UTF-8 & UTF-16
- Also must accept &#nnnn; or &#xhhhh;
- MARC-8 encodings must be converted to Unicode for use in XML
- Special rules for EOL and control characters
  - Changes will be effective version 1.1

http://lcweb.loc.gov/marc/specifications/specchartables.html

Namespaces

- Qualify element and attribute names
- Allows modularization of schemas
  - Mix and match elements from multiple schemas in document instances
  - Import or include from one XML Schema into another

```
<oai:metadata xmlns:oai='http:…' xmlns:oai_dc='…'
   xmlns:dc='…'>
   <oai_dc:dc>
       <dc:title>…</dc:title>
       <dc:creator>…</dc:creator>
```

ZVON Tutorial

http://www.zvon.org/xxl/XMLTutorial/General/contents.html

Simple Illustrations

- A POEM

- Unqualified Dublin Core

Simple Authoring Tools

- MS Notepad (Plain Text Editor)
- MS XML Notepad Beta 1.5

XML Markup Exercise

- Markup one or more of the sample Bib records
- Use <BibRecord> for your root element
- Use Dublin Core element names where applicable:

title	creator	subject
description	publisher	contributor
type	format	date
coverage	relation	identifier
rights	language	source

Advanced Examples

- MARC
- MODS
- OAI (version 2.0)
- DLI/DLIB Test Suite Journal Article
- RDF Qualified Dublin Core
  - Guidelines for DC & RDF DC in XML
- SOAP (Primer)

Advanced Authoring Tool Demos

- YAWC - Microsoft Word
- Extensibility TurboXML

2. DTDs and Schemas

- Formal descriptions of document structure
  - Set expectations
  - Maximize reusability
  - Enforce business rules
- Validation ensures that documents conform
  - Could be multiple schemas for different purposes or different points in a documents lifecycle
- Primary: DTD or XML Schema Language
- Others: Schematron or Relax NG

Document Type Definitions (DTD)

- Legacy from SGML; part of XML standard

```
<!DOCTYPE Book SYSTEM 'http://…'>
<!ELEMENT Book (Front, Chapter+, Back?)>
<!ATTLIST Book
     type (series|monograph) #REQUIRED>
```

ZVON DTD Tutorial

http://www.zvon.org/xxl/DTDTutorial/General/contents.html

Sample DTDs

- A simple DTD for poems
- XHTML 1.0 Strict
- How to validate
  - In Internet Explorer
  - Using XML Notepad

DTD Exercise

- Create a DTD for the previous XML Bib Files
  - Name your DTD file BibClass.dtd
- Insert at the top of each of your XML Bib Files

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE BibRecord SYSTEM "BibClass.dtd" >
```

- Validate using XML Notepad or Internet Explorer

More Examples of DTDs

- OASIS DocBook
- TEI
- Gutenberg

XML schema language

- New in XML
  - Uses XML syntax
  - Supports datatyping
  - Richer and more complex

```
<book xsi:noNamespaceSchemaLocation='HTTP://…'>

<xsd:element name='Book'>
  <xsd:complexType>
      <xsd:sequence>
            <xsd:element name='Front' minOccurs='1'
                  maxOccurs='1' type='frontType'/>…
```

XML Schemas

- BibClass.xsd
- Simple Dublin Core

- Datatypes and Facets
  - Unions
  - Enumerations
  - Lists

More examples of XML Schemas

- Qualified Dublin Core
- OAI-PMH

Alternatives: Schematron & RelaxNG

- Schematron based on XPath (XSLT)
  - Doesn't support datatyping as well
  - Supports additional content models
  - May become an ISO standard
- RelaxNG
  - Returns some of the power of SGML DTDs back to XML (mixed and unordered content)
  - Uses datatyping from the XML Schema spec
  - Does not support inheritance
  - Developed by an OASIS Technical Committee chaired by James Clark

DTD and Schema Tools

- Extensibility TurboXML
- Corel/SoftQuad XMetaL
- XSV for validating

XML & Cascading Style Sheets

- Attach styling instructions directly to XML files
  - `<?xml-stylesheet href=“http:…" type="text/css" ?>`
  - Supported by newest browsers: IE5+, Mozilla, Opera
- Can style but not rearrange elements
  - Block or inline style
  - Bold, italic, underline, font, color, etc.
  - Margins, positioning
  - Generated content (browser support not good)

```
front author {color:red; font-weight:bold; font-family:serif;}
```

XSLT — Transforming Stylesheets

Language for transforming XML documents

  - Into HTML, Text, or other XML documents
  - Supported in new browsers (IE5+, Mozilla; not Opera)
  - Usually applied on the server or in batch mode

- Valuable for interoperability or reusability

```
<xsl:template match='//author'>
   <xsl:element name='dc:creator'>
       <xsl:value-of select='lastname'/>
       <xsl:text>, </xsl:text>
       <xsl:value-of  select='firstname'/>
   </xsl:element >
</xsl:template>
```

CSS & XSLT Examples

- A CSS for poems in XML

- XSLT for Dublin Core XML to XHTML
- XSLT for MARC21 XML to HTML

- XSLT for MARC21 XML to DC XML

ZVON Tutorial

http://www.zvon.org/xxl/XPathTutorial/General/examples.html

http://www.zvon.org/xxl/XSLTutorial/Output/contents.html

XSLT Exercise

- Create a simple XSLT to transform your previous Bib XML files into HTML
  - Name the file simple.xsl
- Either attach the XSLT to your files using

  ```
  <?xml-stylesheet href="simple.xsl" type="text/xsl"?>
  ```

  And display it in IE
- Or use MSXSL from the command line to create the HTML file to display

  msxsl.exe source.xml simple.xsl –o source.htm

XSL Tools

- TIBCO XMLTransform
- MSXSL
- EXSLT

- XSL Formatter

XML & databases

Discussion Points

- Suitability of XML for databases
- XML Query Language
- XML & relational databases
- Demonstrations
  - XML and MS SQL Server
  - A simple XML search using XSLT

Suitability for databases

- Documents are well-structured (well-formed syntax requirement)
- Content is well-labeled (markup semantics)
- Ubiquitous, cross-application, cross-community
  - Open, non-proprietary standard
- Validated XML potentially “machine-understandable”

XML Query Language

Intended to facilitate “interaction between the web world and the database world.”

- Follow-on to Quilt (Don Chamberlin, et al.)
- Another special-purpose programming language for processing XML (& XML-like database structures)
  - Expressions, data types, functions, etc.
- Access (search) functionality – e.g., not for update
- Like XSLT, it relies on XPath
- Borrows data types from XML schema language
- Template Processor (like XSLT, PHP, JSP, ASP)

E.g., List books published by Addison-Wesley after 1991:

XQuery:

```
<bib> { for $b in document("http://www.bn.com")/bib/book
where $b/publisher = "Addison-Wesley"
and $b/@year > 1991
return <book year="{ $b/@year }"> { $b/title } </book> } </bib>
```

Result:

```
<bib>
  <book year="1994">
    <title>TCP/IP Illustrated</title>
  </book>
  <book year="1992">
    <title>Advanced Programming in the Unix environment</title>
  </book>
</bib>
```

XML Query Implementations

- W3C Grammar Test Page
- SoftwareAG Tamino / QuiP
  - Queries XML files directly or XML in Tamino
- Microsoft's XQuery Demo
  - Online demo queries W3C use cases documents
- Oracle prototype XQuery implementation
  - Queries XML files
  - Same site has Oracle SQL/XML implementation
- Xindice (apache.org)
  - Uses XPath instead of XQuery

XML & Relational DBMS

- SQLX
  - Mapping data types, character sets, ... from SQL to XML
  - Searching XML using XPath-based SQL style queries
- Import / Export of XML from relational database management systems now common
  - Specifics vary by implementation
- Microsoft implementation allows:
  - Query relational DB using XPath
  - Query with SQL, return XML
  - Load XML directly into Microsoft SQL
  - OLEDB provider for XML files

Demonstrations

- Using XSLT to query multiple XML docs
  - XSLT to search multiple DC XML files
- Fetching XML from Microsoft SQL

Contact information

- Presentation
  - http://dli.grainger.uiuc.edu/ETEXT/
- Presenters
  - thabing@uiuc.edu