Developing Web Applications

Lecture : XML

Internet Technology CSC457

Dr. Iehab AL Rassan

Today …

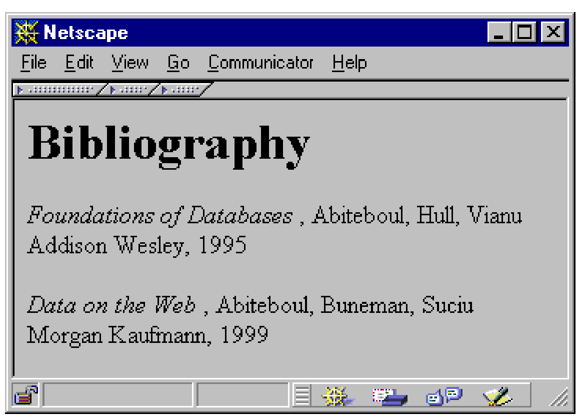
- XML
- XSLT
- WEB SERVICES

XML-

- eXtensible Markup Language
- Not actually a markup language but a way of defining Markup languages
- Tags are not for layout but meaning (semantics)

•A good example of XML? XHTML!

From HTML to XML



HTML describes the presentation

HTML

<h1> Bibliography </h1>

<p> <i> Foundations of Databases </i>

Abiteboul, Hull, Vianu

 Addison Wesley, 1995

<p> <i> Data on the Web </i>

Abiteoul, Buneman, Suciu

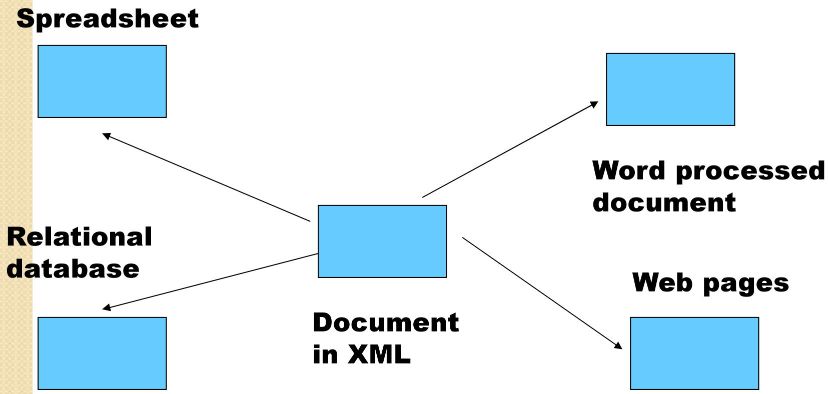
 Morgan Kaufmann, 1999

XML

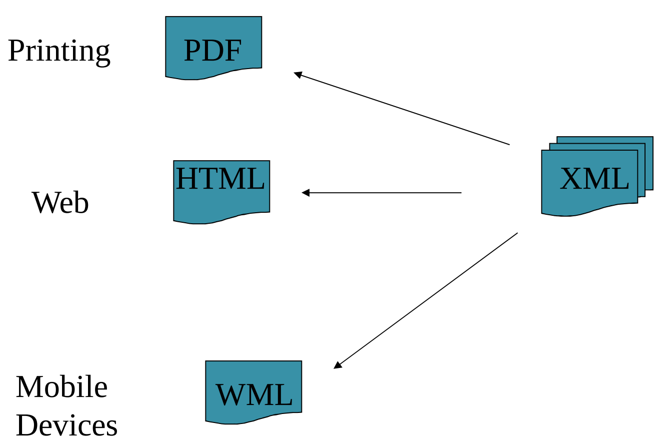
```
<bibliography>
    <book>   <title> Foundations… </title>
             <author> Abiteboul </author>
             <author> Hull </author>
             <author> Vianu </author>
             <publisher> Addison Wesley </publisher>
             <year> 1995 </year>
    </book>
    …
</bibliography>
```

XML describes the content

XML as a central repository of data



XML is independent from the medium



XML Terminology

- tags: book, title, author, …
- start tag: <book>, end tag: </book>
- elements: <book>…<book>,<author>…</author>
- elements are nested
- empty element: <red></red> abbrv. <red/>
- an XML document: single *root element*

well formed XML document: if it has matching tags

XML parts

XML declaration (processing instruction)

example1.xml

```
<?xml version = "1.0"?>
```

Comment

```
<!-- Simple XML document -->
```

Root element

```
<myDocument language="English">
  <message>Hello, World!</message>
</myDocument>
```

- Elements define structure; they may or may not have content.
- Attributes describe elements. An element may have zero, one or more attributes placed within the start tag. Attribute values must be enclosed in quotes (single or double).

An example of a simple XML-based markup language

<PRODUCT>
<PRODUCTNAME> CoatBlue</PRODUCTNAME>
<PRODUCTPRICE> 34000</PRODUCTPRICE>
..
</PRODUCT>

Similar to HTML but contains semantic markers

More XML: Comments

- Syntax <!-- .... Comment text... -->
- Yes, they are part of the data model !!!

More XML: Namespaces

- Namespace is used to provide a unique name for a document.
- One way to do it by using a prefix to the element :
  - Animal:name , person:name
- Namespaces usually take the form of a URL, beginning with a domain name, an optional namespaces label in the form of a directory name and finally a version number, which is also optional :
- Xmlns="http://www.mydomain.com/ns/animals/1.1"

16/11/2013 13

More XML: Attributes

```
<book price = "55" currency = "USD">
   <title> Foundations of Databases </title>
   <author> Abiteboul </author>
    …
   <year> 1995 </year>
</book>
```

attributes are alternative ways to represent data

16/11/2013 14

Document Type Definitions (DTD and Schemas

- An XML document is *well formed* if it is syntactically correct, i.e.
  1. There is a single root element.
  2. Each element has a start and an end tag.
  3. Elements are nested properly.
  4. Attribute values are in quotes.
- Document Type Definitions (DTD and Schemas are used to check whether a document follows the order and structure or not.
- DTD can be declared in the XML document it self, or as an external file.
- Examples Listing 8-2, Listing 8-3 and Listing 8-4

Document Type Definitions (DTD and Schemas

- Schemas performs the same function as a DTD. It defines what is legal in an XML document.
- Schemas are largely replacing DTDs as they are extensible, richer and more useful. Became official W3C recommendation.
- Examples Listing 8-5 and Listing 8-6

Well Formed XML Documents

- XML element and attribute names are *case sensitive* (unlike HTML).
- XML *parser* (or XML *processor*) is required to process XML documents (e.g. msxml built in Internet Explorer).

16/11/2013 17

Processing XML documents (i)

- Done via parsers
- Parser can be conforming or non-conforming
- Conforming checks everything
- Non-conforming just makes rudimentary checks
- Most parsers conforming

16/11/2013 18

Processing XML documents(ii)

- Many parsers written in Java
- Parsers can be in-memory based: they build up a tree
- Parsers can also be event-based: they trigger processing when some XML element is encountered

DTD Example

```
<?xml version="1.0"?>
<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to      (#PCDATA)>
  <!ELEMENT from    (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body    (#PCDATA)>
]>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend</body>
</note>
```

DTD - attributes

<!ELEMENT TOWN (COUNTY, POPULATION)>
<ATTLIST TOWN NAME CDATA #REQUIRED>

The element TOWN has an attribute NAME which contains char data and is always required

XML Schema – the successor to DTDs

• An XML schema fulfils the same criteria as the DTD -- it defines what is legal in an XML document
• A parser or some other processing tool can then use it as a guide
• If data is being exchanged between systems the schema can be used as a guide to the receiver to understand what is expected
• Not only a syntactical check but semantic too

Schema example

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.w3schools.com"
xmlns="http://www.w3schools.com"
elementFormDefault="qualified"><xs:element name="note">
    <xs:complexType>
      <xs:sequence>
            <xs:element name="to" type="xs:string"/>
            <xs:element name="from" type="xs:string"/>
            <xs:element name="heading" type="xs:string"/>
            <xs:element name="body" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
</xs:element></xs:schema>
```

16/11/2013 23

Using XML with Application)

- Listing 8-7 shows an example of using a **data island**.
- **Data islands :** away of including information stored in an XML file inside an HTML page, although this does not work with all browsers.
- First the XML file with the data island is loaded using <XML> element, then the <table> element is bound to the data island using **datasrc** attribute.
- This code will output all entries in the file that are in the fields ARTIST and TITLE.

16/11/2013 24

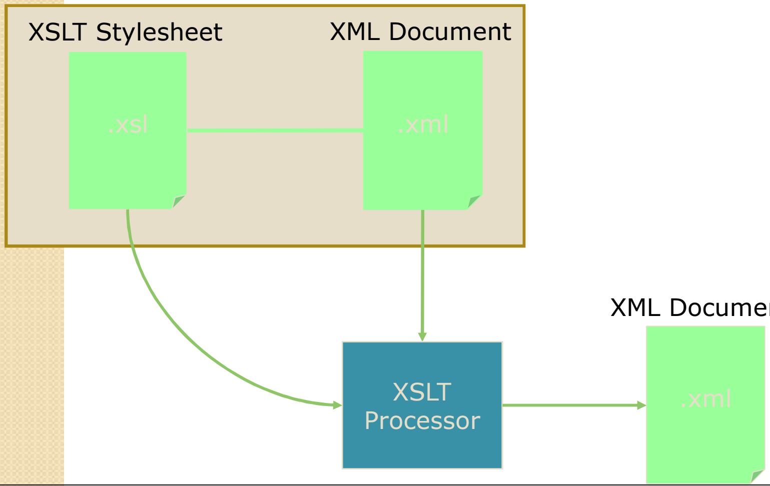
XSL/T
XSL Transformation

An XML-based language used for transforming XML document

What is XSL?

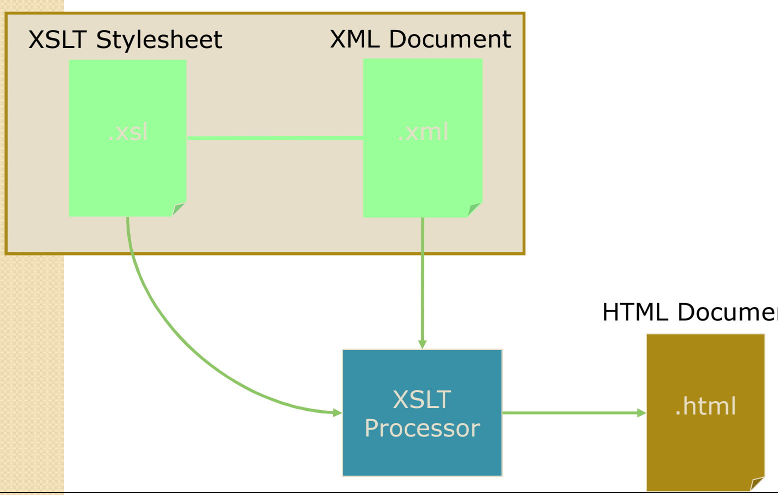
- Stylesheets are typically used to specify how something should be displayed/rendered.
- They lead to separation of document's content from presentational information.
- The Extensible Stylesheet Language (XSL) includes two independent parts: a transformation language (XSLT) and formatting objects language (XSL:FO).
- XSL defines rules for how one XML document is transformed into another XML document

XSLT Processor



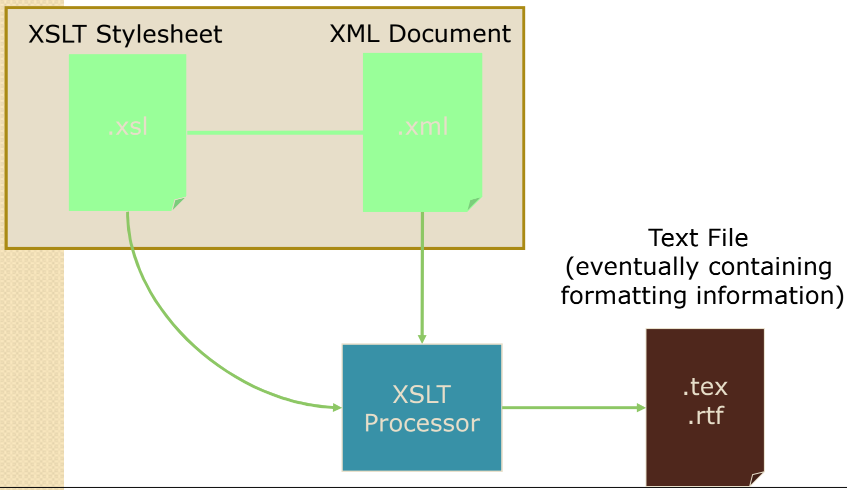
16/11/2013 27

XSLT Processor



16/11/2013 28

XSLT Processor



16/11/2013 29

XSLT Stylesheet (1)

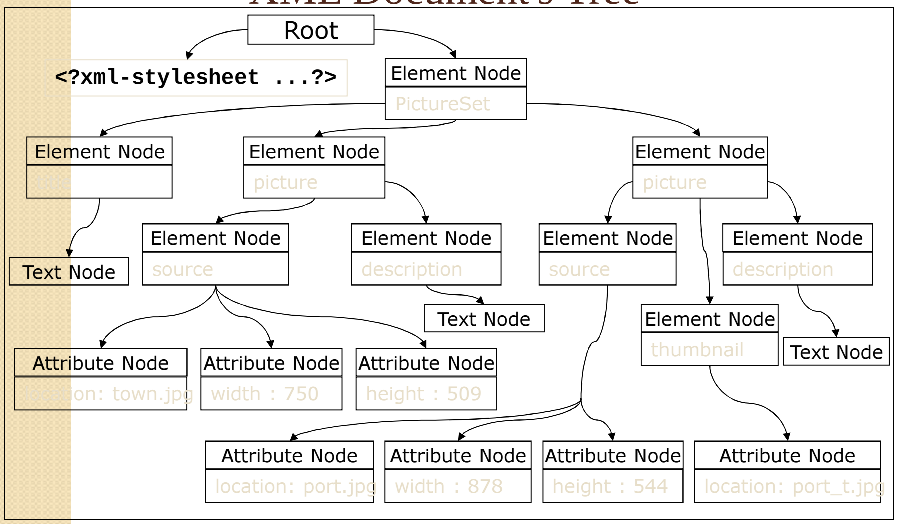
XML document

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xml" href="picturegallery.xsl"?>
<PictureSet>
  <title>Pictures from My Holiday</title>
  <picture>
     <source location="town.jpg" width="750" height="509"/>
     <description>Seaside town</description>
  </picture>
  <picture>
     <source location="port.jpg" width="878" height="544"/>
     <thumbnail location="port_t.jpg"/>
     <description>Sea port</description>
  </picture>
</PictureSet>
```

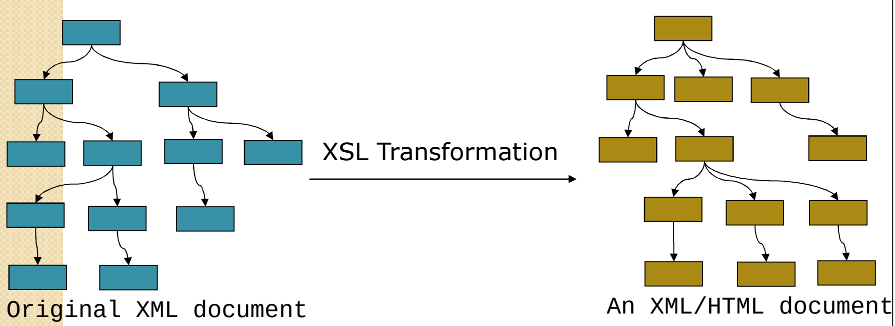
16/11/2013 30

XSLT Stylesheet (2)

XML Document's Tree



XSLT Stylesheet (3)



- XSLT processor models an XML document as a tree that contains seven kind of nodes: the root, element, text, attribute, namespace, processing instruction, comment.
- XSLT operates by transforming one XML tree into another XML tree by employing operators for selecting nodes from the tree, reordering the nodes, and outputting nodes.

XSLT Stylesheet (4)

The Stylesheet picturegallery.xsl

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
       xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="picture">
              <html>
                <body>
                  <xsl:value-of select="description"/>
                </body>
              </html>
       </xsl:template>
</xsl:stylesheet>
```


Result of Applying picturegallery.xsl to the example XML Document

```
Seaside town Sea port
```

Template Rules

- An XSLT processor transforms an XML document starting from the root of the document tree.
- At each turn, a node from the document tree is compared to the template rules.
- When the XSLT processor finds a template rule such that the current node matches its pattern then the XSLT processor outputs the template rule’s substitution part to the result document.
- If the substitution part contains
  `<xsl:apply-templates/>`
  then at that point the child (element and text) nodes of the current node are processed one by one. This is a recursive process.

Template Rules

General structure of an XSLT rule:

```
<xsl:template match="XPath pattern">
    Substitution part
</xsl:template>
```

- The XPath expression specifies
  a pattern to be matched by a node in the document's tree.
- The substitution part specifies what should be inserted instead of the matched pattern:
  - XSL elements, markup and text

Default Template Rules

Default Rule

```
<xsl:template match="/ | *">
    <xsl:apply-templates/>
</xsl:template>
```

- Matches the document tree's root node (**/**) and any element node (*****) and applies template to their child nodes.

What we did:

- Introduction to XML
- Basic XML language
- Structure
- Parsing
- Introduction to XSL/T
- Simple transformations

WEB SERVICES

Web Services

- A service that is platform independent, will work between systems that are distributed and can communicate through firewalls without raising security issues.
- Simple Object Access Protocol (SOAP)
  - Provide a mechanism that allows access to objects across the NET.
    - Cross platform boundaries
    - Go through firewall setup for normal web browsers (port 80)
    - Post little security risk
  - SOAP is a text file using XML.
  - Remote Procedure Call (RPC)
    - One of the most common messaging pattern.
    - Client node sends a request to another node (usually server), which then responds.

Web Services - SOAP

- SOAP = XML + HTTP
- The main idea behind SOAP is to wrap the message you want to send to the remote application in XML and then transport it over HTTP.
- SOAP shares the same port as any other Web communication over port 80.
- SOAP is using the same HTTP request/response protocol.
- The Content-Type header for SOAP request and response states:
  - **POST / item HTTP/1.1**
  - **Content-Type : application/soap+xml ; charset=utf-8**
- The mime type is **application/soap+xml**

16/11/2013 41

Web Services - SOAP

- The Content-length header for SOAP request and response specifies the number of bytes in the body.
- Using SOAP with XML contains several elements :
  - Envelop – identifies the XML document as SOAP message (required)
  - Header – contains header information (optional)
  - Body – contains call and response information (required)
  - Fault – provides information about errors occurred while processing the message (optional)

16/11/2013 42

Web Services - SOAP

- A SOAP message must use the SOAP Envelop and Encoding namespaecs :

<?xml version="1.0"?>

<soap:Envelope
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding"
xmlns:soap="http://www.w3.org/2001/12/soap-envelope">

- Listing 10-1 shows a request is being made for information from remote application about a product whose stock code is 289387.
- Listing 10-2 shows a response back from the remote application, where the server-based application has responded with the answer of ‘DVD recorder’

16/11/2013 43

WEB FEED - RSS

- A Web feed is a document based on XML, which provides content such as news items, weather and blogs.
- Sharing the data with other sites (publishing a feed or syndication)
- The main feed format used are Really Simple Syndication(RSS) and Atom.
- Web feed is usually retrieved using an aggregator, which will read the stream.
- It is easy to subscribe and unsubscribe from streams. For example, news feeds for BBC.
- RSS uses XML.

16/11/2013 44

WEB FEED - BLOGS

- Publication of regular articles over time in some area of personal or professional interest.
  - Business
  - Cultural
  - Political
  - Science
  - Social

- Blogs can be authored manually, simply by writing entries to a Web site with an editor, or automated using a software on sites.