XML Syntax and Rules

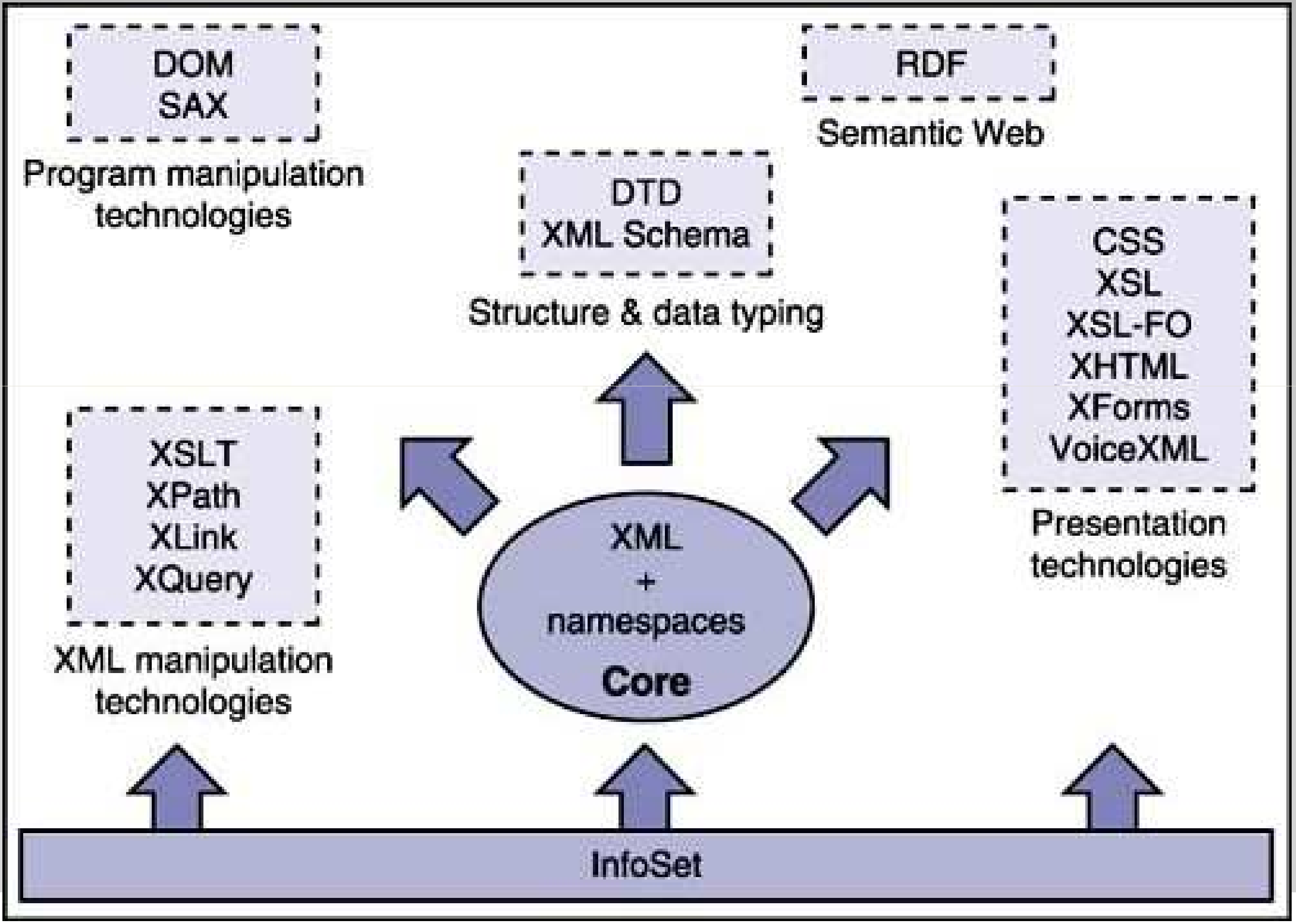
Source

- The slides in this presentation are made from the XML tutorial of W3Schools, available at http://www.w3schools.com/xml/default.asp
- XML, Web Services and the Data Revolution by Frank P.Coyle

XML Technology Family

- XML strength is its simplicity. Primary focus --> data
- Formatting, display, type checking, transform, manipulation, search, and query – are associated with one of the technology families that surround an XML core.
- XML core = XML + namespaces
- Namespaces – spec that allows XML documents from different sources to be combined and yet be able to disambiguate elements with the same name.

XML Technology Family



DOM
SAX

Program manipulation technologies

RDF

Semantic Web

DTD
XML Schema

Structure & data typing

CSS
XSL
XSL-FO
XHTML
XForms
VoiceXML

Presentation technologies

XSLT
XPath
XLink
XQuery

XML manipulation technologies

XML
+
namespaces

Core

InfoSet

XML Technology Family

- Structure and data types – to define how XML documents should be structured. Tech: DTD and XML Schema.
- XML Presentation technologies – delivering content to users via a variety of devices and presentation media.
- XML Manipulation technologies – provides capability to extract and transform XML in different ways.
- Other related technologies – technology for metainformation

XML – Rules

XML – Syntax Rules

- The syntax rules of XML are very simple and logical.

  # All XML Elements Must Have a Closing Tag
  # XML Tags are Case Sensitive
  # XML Elements Must be Properly Nested
  # XML Documents Must Have a Root Element
  # XML Attribute Values Must be Quoted
  # Entity References
  # Comments in XML
  # With XML, White Space is Preserved

1. Closing tag

- In XML, it is illegal to omit the closing tag. All elements **must** have a closing tag:

```
<p>This is a paragraph</p>

<p>This is another paragraph</p>
```

2. Case Sensitive

- XML tags are case sensitive. With XML, the tag <Letter> is different from the tag <letter>.
- Opening and closing tags must be written with the same case:

```
<Message>This is incorrect</message>

<message>This is correct</message>
```

3. Properly Nested

- In XML, all elements must be properly nested within each other:

```
<b><i>This text is bold and italic</i></b>
```

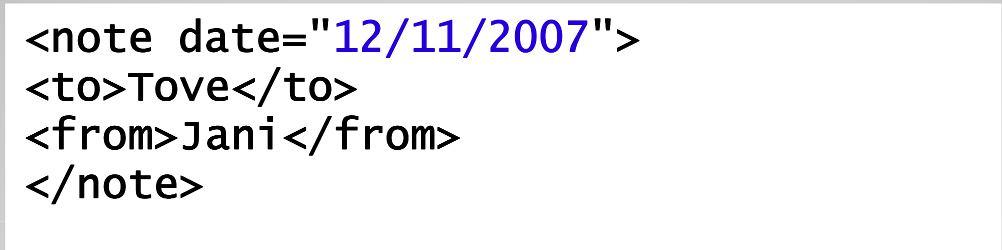
4. Root element

- XML documents must contain one element that is the parent of all other elements. This element is called the root element.

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

5. Attribute values “Quoted”

- In XML the attribute value must always be quoted.



```
<note date="12/11/2007">
<to>Tove</to>
<from>Jani</from>
</note>
```

6. Entity Reference

- This will generate an XML error:

```
<message>if salary < 1000 then</message>
```

To avoid this error, replace the "<" character with an entity reference:

```
<message>if salary &lt; 1000 then</message>
```

There are 5 predefined entity references in XML:

&lt;	<	less than
&gt;	>	greater than
&amp;	&	ampersand
&apos;	'	apostrophe
&quot;	"	quotation mark

7. Comments

- The syntax for writing comments in XML is similar to that of HTML.

```
<!-- This is a comment -->
```

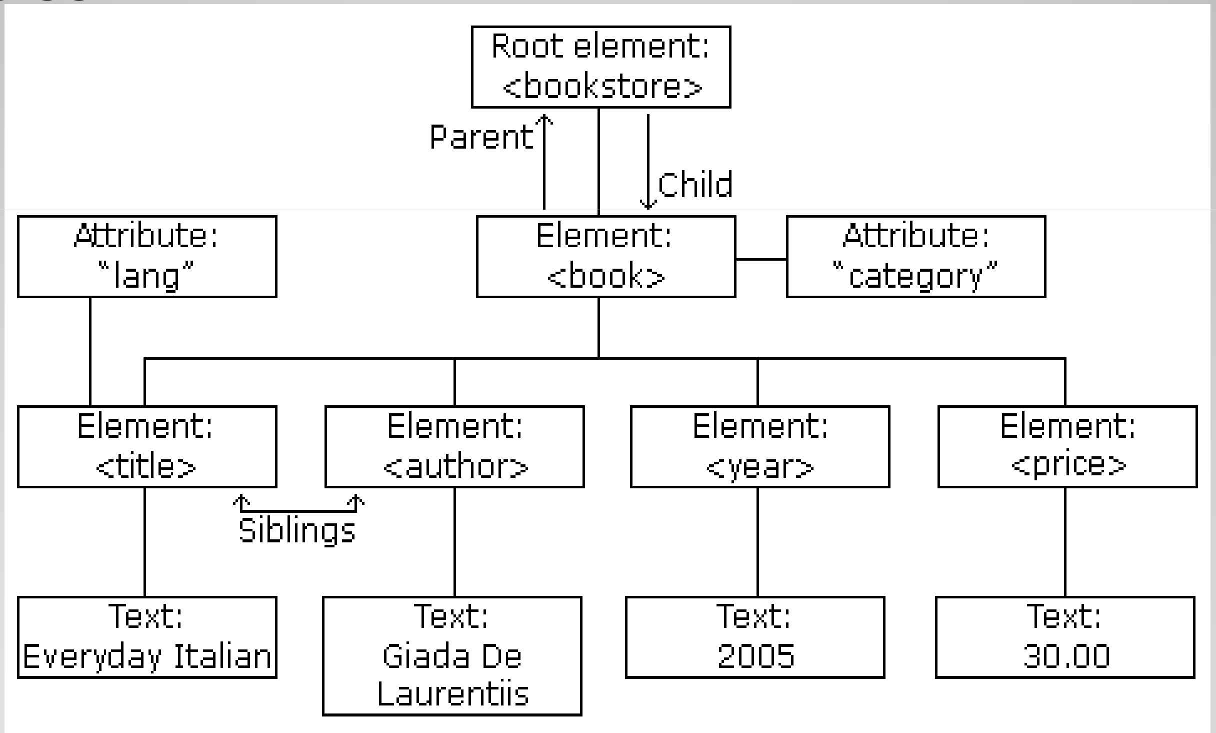
XML Tree

- The first line is the XML declaration. It defines the XML version (1.0) and the encoding used.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<bookstore>  ---> Root
<book category="COOKING"> ---> Child
      <title lang="en">Everyday Italian</title>
      <author>Giada De Laurentiis</author>
      <year>2005</year>
      <price>30.00</price>
</book>
<bookstore>
-------------------------------------------------------------
title, author, year, price ------> Siblings
```

XML Tree

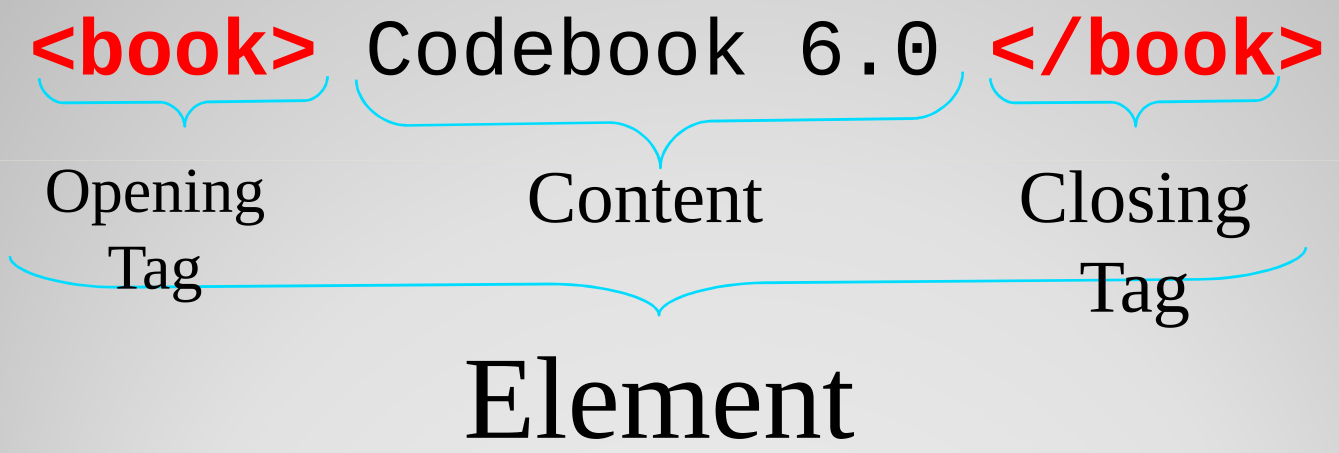
- The elements in an XML document form a document tree. The tree starts at the root and branches to the lowest level of the tree.



XML Elements

What is an XML Element?

- An XML element is everything from (including) the element's start tag to (including) the element's end tag.



```
<book> Codebook 6.0 </book>
```

Opening Tag

Content

Closing Tag

Element

What is an XML Element?

- An element can contain other elements, simple text or a mixture of both. Elements can also have attributes.

```
<bookstore>
<book category="CHILDREN">
  <title>Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
</bookstore>
```

- <bookstore> and <book> has element content and <author> has text content. <book> has an attribute (category="CHILDREN")

XML Naming Rules

- XML elements must follow these naming rules:
  * Names can contain letters, numbers, and other characters
  * Names must not start with a number or punctuation character
  * Names must not start with the letters xml (or XML, or Xml, etc.,)
  * Names cannot contain spaces
- Any name can be used, no words are reserved.

Best Naming Practices

- Make names descriptive
- Names with underscore separator are nice: <first_name>
- Name should be short and simple: <book_title>
- Avoid – character: first-name, subtract name from first?
- Avoid . Character: first.name, name is a property of object first
- Avoid : charater: Colons are reserved for namespaces
- Non-english letters are prefectly legal in XML.

XML Attributes

- XML elements can have attributes in the start tag.
- Attributes provide additional information about elements.
- An attribute consists of a name-value pair.
- Attribute values must always be enclosed in quotes, but either single or double quotes can be used.

```
<productname prodid=”P001”>Barbie Doll</productname>
```

```
<gangster name="George &quot;Shotgun&quot; Ziegler">
```

Elements Vs. Attributes

- Attributes cannot be further subdivided into subelements, but elements can always be subdivided. If there's need to break down data into more detail, then choose elements.
  1. <note date="10/01/2008">
  2. <date>10/01/2008</date>

```
<note>
<date>
        <day>10</day>
        <month>01</month>
        <year>2008</year>
</date>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Avoid Attributes?

- Attributes cannot contain multiple values
- Attributes can not contain tree structures
- Attributes are not easily expandable
- Attributes are difficult to read and manipulate with programs.
- Use elements for data. Use attributes for information that is not relevant to the data.

XML Attributes for Metadata

- The ID is for identifying the differnet notes. It is not part of the note itself.
- The metadata should be stored as attributes, and data itself should be stored as elements

```
<messages>
  <note id="501">
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
  </note>
  <note id="502">
<to>Jani</to>
<from>Tove</from>
<heading>Re: Reminder</heading>
 <body>I will not</body>
  </note>
</messages>
```

XML Validation

- XML with correct syntax is "Well Formed" XML.
- XML validated against a DTD is "Valid" XML.

Well Formed XML

- A "Well Formed" XML document has correct XML syntax.
- The syntax rules were:
- * XML documents must have a root element
- * XML elements must have a closing tag
- * XML tags are case sensitive
- * XML elements must be properly nested
- * XML attribute values must be quoted

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Valid XML

- A "Valid" XML document is a "Well Formed" XML document, which also conforms to the rules of a Document Type Definition (DTD)

Viewing XML

- Type the XML contents in a notepad and save the file with **.xml** extension.
- Open the file with any browser – IE, Netscape, Firefox, Opera
- To view the raw XML source, select "View Page Source" or "View Source" from the browser menu.

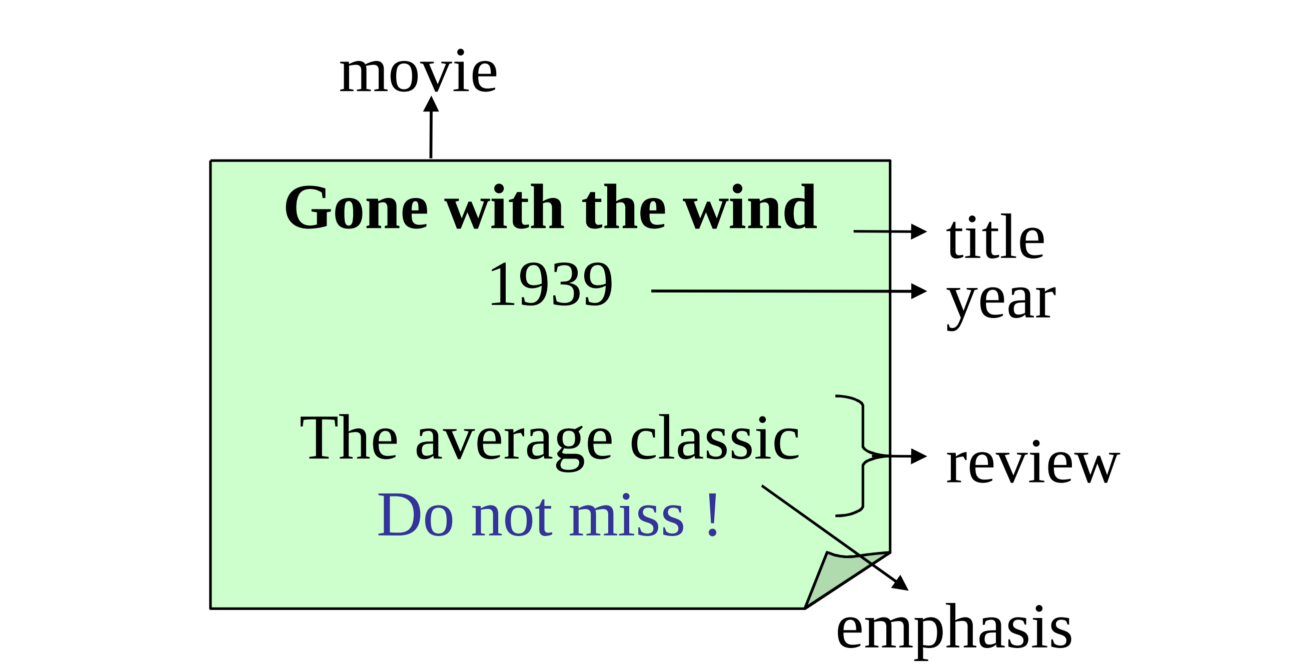
Problem Statement -1

- The details of books sold by CyberShoppe need to stored and made available to various branches regardless of platforms used. The book details consist of the title, the first and the last names of the author and the price of the book. Each book is uniquely identified by a book ID.

- Create a “WellFormed” XML document and view it in a browser

Problem Statement - 2

- Create a “WellFormed” XML document and view it in a browser



What is DTD ?