Basics of XML and HTML

Gaston Sanchez
Creative Commons Attribution Share-Alike 4.0 International (CC BY-SA)
About

Large amounts of data and information are stored, shared and distributed using HTML and XML-dialects.

They are widely adopted and used in many applications.

The goal is to give you a crash introduction to XML and HTML so you can work with so-called web technologies.
What is XML?
XML
eXtensible Markup Language
"XML is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable"
<xml version="1.0"?>
<!DOCTYPE movies>
<movie mins='126' lang='en'>
    <!-- this is a comment -->
    <title>
        Good Will Hunting
    </title>
    <director>
        <first_name>Gus</first_name>
        <last_name>Van Sant</last_name>
    </director>
    <year>1997</year>
    <genre>drama</genre>
</movie>
Markup Language?
A markup language is a system for **annotating** (i.e. *marking*) a document in a way that the content is **distinguished** from its representation.

- LaTeX
- HTML
- Markdown
In XML the structure markers are defined by angle brackets:

<mark>Text marked with a tag</mark>
Extensible?
The concept of extensibility means that you can create NEW marks.
So what is XML?
XML is NOT

A programming language
A network transport protocol
A database
XML is

More than a markup language

A generic language that provides structure and syntax for defining many markup dialects

A standard for the semantic, hierarchical representation of data

It is particularly useful as a format for sharing information between various software systems
Examples
<movie>
  Good Will Hunting
</movie>
<movie>
  Good Will Hunting
</movie>

one single element 'movie'
<movie>Good Will Hunting</movie>
<movie>

Good Will Hunting

content

</movie>
Simple Example
<movie>
  <title>Good Will Hunting</title>
  <director>Gus Van Sant</director>
  <year>1997</year>
  <genre>drama</genre>
</movie>
<movie>
  <title>Good Will Hunting</title>
  <director>Gus Van Sant</director>
  <year>1997</year>
  <genre>drama</genre>
</movie>
<movie> child elements

<title>Good Will Hunting</title>
<director>Gus Van Sant</director>
<year>1997</year>
<genre>drama</genre>
</movie>
XML Tree Structure
<Root>
   <child_1>...</child_1>
   <child_2>...</child_2>
      <subchild>...</subchild>
   <child_3>...</child_3>
</Root>
must have exactly ONE root element

<child_1>...</child_1>
<child_2>...</child_2>
  <subchild>...</subchild>
<child_3>...</child_3>

</Root>
<Root>
  <child_1>...</child_1>
  <child_2>may contain child elements</child_2>
  <subchild>...</subchild>
  <child_3>...</child_3>
</Root>
<Root>
  <child_1>... </child_1>
  <child_2>... </child_2>
    <subchild> may contain subchild elements </subchild>
  <child_3>... </child_3>
</Root>
Another Example
<movie>
  <title>
    Good Will Hunting
  </title>
  <director>
    <first_name>Gus</first_name>
    <last_name>Van Sant</last_name>
  </director>
  <year>1997</year>
  <genre>drama</genre>
</movie>
<movie>
  <title>
    Good Will Hunting
  </title>
  <director>
    <first_name>Gus</first_name>
    <last_name>Van Sant</last_name>
  </director>
  <year>1997</year>
  <genre>drama</genre>
</movie>
<movie>
  <title>Good Will Hunting</title>
  <director>
    <first_name>Gus</first_name>
    <last_name>Van Sant</last_name>
  </director>
  <year>1997</year>
  <genre>drama</genre>
</movie>

'movie' has 4 child elements
<movie>
  <title>Good Will Hunting</title>
  <director>
    <first_name>Gus</first_name>
    <last_name>Van Sant</last_name>
  </director>
  <year>1997</year>
  <genre>drama</genre>
</movie>

'director' has 2 children
Tree Diagram
movie

- title
  - Good Will Hunting

- director
  - first_name: Gus
  - last_name: Van Sant

- year
  - 1997

- genre
  - drama
*Movie: Good Will Hunting*

- **Title:** Good Will Hunting
- **Director:** Gus Van Sant
- **Year:** 1997
- **Genre:** Drama
a well-formed XML document has exactly one root element

movie

title
- Good Will Hunting

director
- Gus Van Sant

year
- 1997

genre
- drama
XML Attributes
<movie mins="126" lang="en">
  Good Will Hunting
</movie>
XML elements can have attributes

attributes attached to element's start tag

<movie mins="126" lang="en">Good Will Hunting</movie> attributes MUST be quoted!
Additional XML elements
<?xml version="1.0" encoding="UTF-8" ?>
<![CDATA[ a > 5 & b < 10 ]]>  
<!DOCTYPE Movie>  
<!-- This is a comment -->  
<movie mins="126" lang="en"> 
  <title>Good Will Hunting</title>  
  <director>  
    <first_name>Gus</first_name>  
    <last_name>Van Sant</last_name>  
  </director>  
  <year>1997</year>  
  <genre>drama</genre>  
</movie>
<table>
<thead>
<tr>
<th>Markup</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;?xml &gt;</code></td>
<td>XML Declaration: identifies content as an XML document</td>
</tr>
<tr>
<td><code>&lt;?PI &gt;</code></td>
<td>Processing Instruction: processing instructions passed to application PI</td>
</tr>
<tr>
<td><code>&lt;!DOCTYPE &gt;</code></td>
<td>Document-Type Declaration: defines the structure of an XML document</td>
</tr>
<tr>
<td><code>&lt;![CDATA[ ]]&gt;</code></td>
<td>Character Data: anything inside a CDATA is ignored by the parser</td>
</tr>
<tr>
<td><code>&lt;!-- --&gt;</code></td>
<td>Comment: for writing comments</td>
</tr>
</tbody>
</table>
XML Dialects
XML

It is useful as a format for sharing information between different software systems.

Allows the analyst to think about data in new ways because of the metadata on the structure for complex data.
XML is used for

- traditional data sets (i.e. data tables)
- spreadsheets (i.e. excel)
- visual graphical displays such as SVG
- social network structures
- text documents
- descriptions of user interfaces
- RSS feeds
- data sent to and from web services
- XML databases
Some XML dialects

- **KML (Keyhole Markup Language)** for describing geo-spatial information used in Google Earth, Google Maps, Google Sky
- **SVG (Scalable Vector Graphics)** for visual graphical displays of two-dimensional graphics with support for interactivity and animation
- **PMML (Predictive Model Markup Language)** for describing and exchanging models produced by data mining and machine learning algorithms
Some XML dialects

- **RSS (Rich Site Summary)** feeds for publishing blog entries
- **SDMX (Statistical Data and Metadata Exchange)** for organizing and exchanging statistical information
- **SBML (Systems Biology Markup Language)** for describing biological systems.
What is HTML?
HTML

HyperText Markup Language
Hypertext

Hypertext is text that contains links to other texts. By clicking on a link in a hypertext document, a user can quickly jump to different content. The term was coined by Ted Nelson (1965).
HTML is

The standard markup language for creating web pages and web applications
<html>
<head>
    <title>Page title</title>
</head>
<body>
    <h1>Big Header</h1>
    <p>This text is a paragraph.</p>
    <h2>This is a sub-heading</h2>
    <p>Here’s another paragraph. Just another dummy sentence.</p>
    <h2>A second sub-heading</h2>
    <p>Final paragraph.</p>
</body>
</html>
Big Header

This text is a paragraph.

This is a sub-heading

Here’s another paragraph. Just another dummy sentence.

A second sub-heading

Final paragraph.
HTML elements
Basic element structure

- Tag name

- Left-angle bracket
- Right-angle bracket

- Opening tag

- Slash

- Left-angle bracket
- Right-angle bracket

- Closing tag
Attributes provide additional information about the contents of an element. They appear on the opening tag of the element and are made up of two parts: a **name** and a **value**, separated by an equals sign.
Head (with Title) and Body

<html>
<head>
  <title>Page title</title>
</head>
<body>
  <h1>Big Header</h1>
  <p>This text is a paragraph.</p>
  <h2>This is a sub-heading</h2>
  <p>Here’s another paragraph. Just another silly dummy sentence.</p>
</body>
</html>
Body, Head, and Title

**<body>**
Everything inside the body element is shown inside the main browser window

**<head>**
Before the body there’s usually a head element. This contains information about the page.

**<title>**
Inside the head we usually find a title element. Its contents are shown in the top of the browser or on the tab for the page.
Summary

HTML pages are text documents

HTML uses tags, which are characters that sit inside angled brackets. They act like containers and tell you something about the information that lies between them.

To learn HTML you need to know what tags are available for you to use
Summary

Tags usually come in pairs. The opening tag denotes the start of a piece of content; the closing tag denotes the end.

Opening tags can carry attributes, which tell us more about the content of that element.

Attributes require a name and a value.

To learn HTML, you need to know what tags are available for you to use, what they do, and where they can go.