


Week 1 Lecture : What is XML?

Recorded Lecture (2022):

- https://sjeccd-edu.zoom.us/rec/play/_/mEQXC0cCB1AB4YSMTOHvQaPgzsBg7FrXS_W21jIBvqGzh8z5dBR712lx-xH-RXG7AQ_t_wUpNA5SRcgMK.Q-n-Do9SKftbGchh  (https://sjeccd-edu.zoom.us/rec/play/_/mEQXC0cCB1AB4YSMTOHvQaPgzsBg7FrXS_W21jIBvqGzh8z5dBR712lx-xH-RXG7AQ_t_wUpNA5SRcgMK.Q-n-Do9SKftbGchh)

Recorded Lecture (2021):

- https://sjeccd-edu.zoom.us/rec/play/_/RcUR6zCtmakFgwjzuapQJbp4P44uAlxT2wmFXrSVYmOOEy6G_x_65gkILj7SHga3DoSNuAQIAfUjS4PE.TqkYXQbb0yuqYytP  (https://sjeccd-edu.zoom.us/rec/play/_/RcUR6zCtmakFgwjzuapQJbp4P44uAlxT2wmFXrSVYmOOEy6G_x_65gkILj7SHga3DoSNuAQIAfUjS4PE.TqkYXQbb0yuqYytP)

Welcome to our class. I'll start off with some general information about XML and then get into much more detail in future lessons. This is a small lesson as we are just getting started, future lessons will be much longer, cover more material, and will be full of lots of XML examples. I hope you find the following interesting. There is an associated homework assignment that goes along with this lesson so be sure to do it.

What is XML?

XML stands for eXtensible Markup Language. It was developed by the W3C (World Wide Web Consortium). The W3C is the organization in charge of the development and maintenance of many Web standards. Markup is used to convey some information about text or other data. XML is actually a set of rules for defining semantic tags that break a document into parts and identify the different parts of the document

XML is a meta-markup language

A meta language can be thought of as a set of grammar rules. The application languages that follow the specified set of rules can be thought of as a vocabulary. XML isn't just another markup language like Hypertext Markup Language (HTML). HTML defines a fixed set of tags (element type names) that describe a fixed number of elements. If the markup language you use doesn't

contain the tag you need, you're out of luck.

XML, however, is a meta-markup language. It's a language in which you make up the tags (element type names) you need as you go along. These tags must be organized according to certain general principles, but they're quite flexible in their meaning. For instance, if you're working on genealogy and need to describe family names, personal names, dates, births, marriages, and so on, you can create tags for each of these. You don't have to force your data to fit into paragraphs, list items, table cells, and other very general categories. The tags you create can be documented in a Document Type Definition (DTD), which we will talk about later. For now, think of a DTD as a vocabulary and syntax for certain kinds of documents.

XML means you don't have to wait for browser vendors, such as Microsoft (Edge), Google (Chrome), and Mozilla (Firefox), to catch up with what you want to do. You can invent the tags you need, when you need them, and tell the browsers how to display these tags.

XML describes structure not formatting

Another fact about XML is that XML markup describes a document's structure and meaning. It does not describe the formatting of the elements on the page. Formatting can be added to a document with a style sheet. The document itself only contains tags that say what is in the document, not what the document looks like.

XML is self-describing data

Much computer data from the last 60 years is lost, not because of natural disaster or decaying backup media but simply because no one bothered to document how one actually reads the data media and formats. A Lotus 1-2-3 file on a 40 year-old 5.25-inch floppy disk may be irretrievable in most corporations today without a huge investment of time and resources. Data in a less known binary format such as Lotus Jazz may be gone forever.

XML is an incredibly simple data format. It can be written in 100 percent pure ASCII text as well as in a few other well-defined formats. ASCII text is reasonably resistant to corruption. The removal of bytes or even large sequences of bytes does not noticeably corrupt the remaining text. This starkly contrasts with many other formats, such as compressed data or serialized Java objects, in which the corruption or loss of even a single byte can make the entire remainder of the file unreadable.

XML is self describing. Suppose you live in the twenty-third century and encounter the following chunk of XML code on an old floppy disk that has survived the ravages of time:

```
<PERSON ID="p1100" SEX="M">
  <NAME>
    <FIRST>John</FIRST>
    <LAST>Doe</LAST>
  </NAME>
```

```
<BIRTH>
  <DATE>21 March 1903</DATE>
</BIRTH>
<DEATH>
  <DATE>14 April 1972</DATE>
</DEATH>
</PERSON>
```

Even if you're not familiar with XML you've got a pretty good idea that this fragment describes a man named John Doe, who was born on March 21, 1903 and died on April 14, 1972. Even with gaps in, or corruption of the data, you could probably still extract most of this information. The same could not be said for a proprietary binary spreadsheet or word-processor format.

There is price to pay for the simple ASCII format of XML. It means larger file sizes and increased transmission time on the Web. We'll talk about these issues at a later time.

Some more general info

XML is, at its root, a document format. It is a series of rules about what XML documents look like. There are two levels of conformity to the XML standard. The first is well-formed and the second is validity. We'll talk about both of these later.

XML documents are most commonly created with an editor. Simple editors can be used and are usually the easiest and best to use.

My favorite text editor these days is Atom. It's free, and works on both Mac and Windows systems:

<https://atom.io/>  (<https://atom.io/>)

A pretty good free text editor for Windows is Notepad++ and can be obtained at:

<https://notepad-plus-plus.org/>  (<https://notepad-plus-plus.org/>)

And if you don't mind spending a few dollars, Sublime is, well, sublime:

<https://www.sublimetext.com/>  (<https://www.sublimetext.com/>)

An XML parser (also known as an XML processor) reads the document and verifies that the XML document is well formed and can even check that the document is valid. There are a number of parsers around, most of which are not free.

- Saxon: <http://www.saxonica.com/html/documentation/about/whatis.html> 

(<http://www.saxonica.com/html/documentation/about/whatis.html>)

- Stylus Studio: <http://www.stylusstudio.com> ➦ (<http://www.stylusstudio.com/>)
- Xerces: <https://xerces.apache.org/> ➦ (<https://xerces.apache.org/>)

There are, however, a few free, online parsers that we will be using in class:

- <http://xmlvalidator.new-studio.org/> ➦ (<http://xmlvalidator.new-studio.org/>)
- <https://www.freeformatter.com/xml-validator-xsd.html> ➦ (<https://www.freeformatter.com/xml-validator-xsd.html>)
- <http://videlibri.sourceforge.net/cgi-bin/xidelcgi> ➦ (<http://videlibri.sourceforge.net/cgi-bin/xidelcgi>)

To summarize, an XML document is created in an editor (like Atom). The XML parser reads the document and converts it into a tree of elements. The parser passes the tree to the browser or exports it to a file that the user can then use.

In this class we will be creating many types of XML documents and use them to analyze and extract useful information.