

# MARKUP LANGUAGES

## A brief history of Markup languages

The first standardised structured information technology of any importance was SGML (**Standard Generalised Markup Language**) developed by IBM. It was originally created to provide a way of formatting legal documents. It was subsequently expanded as an all purpose information standard and in 1986 emerged as an ISO (International Standards Organisation) standard

SGML is extremely powerful and as a consequence is also quite complex. For this reason in the early days of the Internet the search was on for something simpler. In 1989, Tim Berners-Lee and Anders Berglund, two researchers at CERN (the European Laboratory for Particle Physics) created a tag based language for marking up technical documents which could then be shared across the internet. This language we now know as HTML (Hypertext Markup Language) was a simplified version of SGML.

As the popularity of the web increased, so did the demands placed on it and HTML underwent a number of changes in order to keep up. Remember it was originally conceived as a way of presenting static information. The information presented on the web needed to be supported by databases with **HTML front ends**.

HTML was evolving to cope with dynamic information (DHTML), supported by other technologies, such as Java applets and other plug-ins. In changing however it started to expose its weaknesses. The most obvious being its fixed number of tags. It would be nice if custom tags could be added to suit the particular needs of industry. SGML supported the use of

custom tags and offered three significant benefits that were missing from HTML, namely

- Extensibility
- Structure
- Validation

Technically HTML does have structure, but most browsers, and most notable MSExplorer, did not enforce it. Good marketing perhaps, selling a browser that doesn't complain when users write sloppy code, but a bad idea.

In 1996 the W3C (the World Wide Web consortium) set out to find a way of adding the benefits of SGML but retain the simplicity of HTML. The result of their work was released two years later in February 1998 as XML 1.0 (eXtensible MarkupLanguage). XML is a sub set of SGML, the original XML specification was about a tenth the size of the SGML specification. **So isn't XML just another version of HTML?**

No, it is radically different. HTML is a **specific markup**

language for encoding information and displaying displaying information in a web browser. **XML is a specification for designing markup languages.** I.e. its a **meta language**, (meta meaning information about information). In this case information about how to design your own markup language. So if you think about it, XML can be used to design HTML.

In fact this has already been done, and is reflected in the current HTML specifications. The new version of HTML naturally is called XHTML. But hold on a minute, HTML has a fixed number of tags, to markup text etc. for display. The browser knows how to display each tag. (Which is why different browsers from different manufacturing, display things in a slightly different way).

**Whats a browser going to do if I make up my own set of tags?**

OK remember **XML is about structuring information, not about how it should be displayed.**

There are XML viewers which let you view the structure of the HTML markup but this doesn't show how the information is to be displayed. If you have used HTML its a problem you are perhaps already familiar with. Along with the content of the document itself you must embed tags to tell the browser how to display the document. Consequently, you end up with a document thats littered with <font> tags for example. **I.e. the content and format information is all mixed up together.** That is until **Cascading Style Sheets** (CSS) came along. Now the content and format information can be kept separate.

So you use XML to structure the content, and CSS to describe how it is to be displayed. Problem solved then. Well, in the same way as people started to realise the limitations of HTML, the spotlight has turned on CSS. It to has its shortcoming, so along with XML comes XSL (eXtensible Style sheet Language).

Currently either CSS or XSL can be used to display an XML document. Eventually though CSS may be gobbled up by XSL in the same way HTML will probably be swallowed up by XML.

In focusing on how information is to be displayed is straying from the real point of XML. HTML and **the Internet is currently designed for use by humans, not machines.** How many times have you typed a query in your favourite search engine, and get a million replies, on all sorts of things you never intended.

The classic story perhaps is the dear old ladies of the Womens Institute in Norfolk, who got their first computer. As a group they were very keen on lace making, so naturally enough typed lace into their search engine. What they didn't know, but very quickly

found out, is that lace is American slang for prostitution, so no guesses what the majority of search results were.

Now some of the later generation of search engines will take the million or so results and categorise them for you. By selecting the category you are interested in the search engine can use this information to filter out all the unwanted stuff. What you are doing is provided **metadata** i.e. information about information (your query) by putting it into context.

It is this more fundamental problem, that XML is setting out to address; because once the machines understand what we are looking for, and can manipulate the data for themselves (using **Artificial Intelligence** techniques), maybe in the not to distant future when you enter your query you will actually get back just what you what

Source : <http://webstyle.soslug.org/>