HyperText, Metadata and XML

Hypertext - establishes the conceptual principles of content linking

Metadata - provides a means to label content

XML - provides mechanisms to formally describe content.

Hypertext

Consists of content elements which contain **links** to other content elements. These links generally have some type of *topical relationship*.

- WWW essentially an eg of hypertext..a *hypermedia*, combining text, images, audio, video. The Web is an implementation of a subset of hypertext concepts.
- Hypertext browsers eg Chrome, Safari encourage reading in a **non-linear manner**
- Non-linear differs to print media's sequential presentation, with contents page for jumping, article splitting etc determined by an editor. Readers still probably read eg magazines in a unique order.
- Hypertext enables computer-supported non-linear view where readers can jump ro necessary info and follow links in the hyperext
- Also enables a flexible way to organise info; can use traditional hierarchial chapters, sections etc, but can adopt other structures or nothing formal at all
- Hypertext has **no physical size constraint on links**

A Hyperext contains:

- **Nodes**: chunks of info that can be broken into smaller units eg HTML web page.
- **Links**: connects two or more nodes eg a web link between HTML web pages. Links in HTML are *unidirectional, one to one* and mayy point at entire pages, files of various media or sections of the same document(using the # fragment identifier), mail addresses
- **Anchors**: Represent the source or destination of an information link. persistent selection in documents/nodes, can be highlighted words, phrases, strings in text, icons, tgraphic/image.
  - anchors should always be *instantly identifiable*
  - Anchortext can provide a link to another node which gives a detailed indication of the contend of the link item
    - Anchortext can also label images, represent a doc not yet crawled by a web spider and can augment contents of a document which has been indexed.

..Where do you begin with no start point or end point in hypertext? ROle of search engine in htext is to provide a list of potential entry points

A **landmark/index node** is typically maintained by an authority on a subject and provides links to reliable resources on a topic.

The **Search Engine Results Page** or SERP provided by SE could be viewed as a *dynamically created landmark node*.

Easy to get disoriented in Htext.. use Graphical browsers to enable traversal, a "Go Back" command, History lists and Bookmarks. Search Facilities provide good starting points.

- FIsheye views present local data in details and remote data is abstract.
- Margin Notes are personalised notes which should be iconised and individualised
- Breadcrumbs and coffee stains indicate that a reader was here recently

When creating a Hypetext, it is advisable to adopt these 3 rules..

- oganise source data into **fragments** that can be linked together.
- ensure these fragments *relate* to one another
- ensure that the user only needs a *small fraction* of the fragments at any one time

Metadata

Provides a way of **describing the contents of items**, which can be used to support search for these items or filtering using constraings. Machine readable information about media content. Metadata can describe other metadata.

- Real world examples:
  - Library index: keywords, title, author, ISBN, subject
  - video movie file: direcotr, actors, genres
  - Golden Pages business directory: company name, product section, address

Metadata can be created automatically or manually

Metadata can use a controlled or fixed vocab, standardised features (dates), free text

SGML: Standardised Generalised Mark-up Language

- Metadata for an info element neeeds to be stored in a form tha_makes its purpose clear_ and make the individual info elements easily accessible.
- It needs to be in a **standardised format** that a computer can be programmed to process reliable - become machine readable
- SGML is an international standartd that describes a generalized markup scheme for representing doc structure and content in a system and platform independant way. It is not a language, it is a **meta-language** for defining mark up languages. tags are stored in a **Document Type Definition**(DTD) file.
- Special sw must be produced in order to visualise SGML
- SGML is the basis for defining markup languages including HTML

XML - eXtensible Markup Language.

This is a simplified subset of SGML. It's a **protocol for designing mark-up languages**. Also:

- A family of technologies that can do everything from formatting documents to aiding the filtering of data and transferring of data between applications.
- A philosophy for info handling that seeks maximum usefulness and flexibility for data by refining it to its purest and most structued form. A defined markup language is used to contain and manage the info
- Based on the Unicode character set which enables it to be used with numerous language.

XML is **not** HTML nor a replacement for HTML.

- XML to *describe data* and focus on what data *is*

- HTML to *display data* and forcus on how data *looks*

```
XML                                 HTML
must have closing tags <></>        Does not need closing tags to function
Tags are case sensitive             Tags not case sensitive
```

to display XML encoded data a **stylesheet** can be used to turn tags into formatting instructions.

XML is **extensible**: new tags can be defined for specific domains or data

Freeform XML is well formed XML, and conforms to the following

- all elements properly nested
- all attributes quoted
- all elements with empty content are *self identifying*
- case sensitive
- No use of characters with *reserved meanings*

Document modelling "valid" XML conforms to

- rules as above -A valid DTD file. Document instance is compared to a document model in a process called validation.

```
<!-- course.dtd ->

<! DOCTYPE course
[
    <!ELEMENT course (title,author)>
    <!ELEMENT title (modcode,modtitle)>
    <!ELEMENT modcode (#PCDATA)>
    <!ELEMENT modtitle (#PCDATA)>
    <!ELEMENT author (firstname,surname)>
    <!ELEMENT firstname (#PCDATA)>
    <!ELEMENT surname (#PCDATA)>
]>
```

Cascading Style Sheets and XSLT are useful for displaying XML, providing tags with formatting characteristics. Display rules are defined in a stylesheet

XML DTDs can be defined for objects to be entered into a S.E.

- XML markup scheme can be defined which captures attributes of eg an image with time of capture, GPS location, quality and automatic content analysis
- Structured metadata of thys type can be used in various ways by a S.E.
  - Search against individual elements (search images based on the description entered upon uploading, or ones taken by a certain model of camera etc)
  - Combine *all* metadata into a single unstructured file, index like a standard free text doc, and then search using a standard IR system.
  - See *MediAssist* system to see how both uses of metadata are incorporated.