UNIT 8 INTRODUCTION TO HTML AND XML

Structure

8.0 OBJECTIVES

In this unit you will be able to :

- Learn what is WWW;
- Understand what is a markup language; and
- Know what is HTML (Hypertext Markup Language) and XML (eXtensible Markup Language) and their use.

8.1 INTRODUCTION

Since it's inception the growth of Internet is very rapid. Earlier it was text based but due to introduction of WWW, Internet has become a channel of

multimedia. Data can be transferred in all the formats. Hypertext Markup language is a markup language to render the information over Internet. It can accommodate audio, video, text and image. The basic feature of today's Internet is Hypertext and Hyper-linking. Internet has become so much dependent on the use of hypermedia and hypertext that WWW has become a synonym for the term Internet.

8.2 WWW

WWW is a standard designed for the access of multimedia information over Internet. It was developed by National Centre for Super-Computing Applications (NCSA). Before WWW, Gopher was another protocol to browse text based information. But with WWW one can read and view text, audio, video and image files.

WWW is a set of protocols which contains several protocols like File transfer protocol (FTP), Telnet (Remote access), Hypertext Transfer Protocol (HTTP) and so on. The browsers support these protocols' work.

The server should support the particular access protocol for example, to support FTP the server must be a FTP server. Similarly to serve the webpage the server must be a webserver.

WWW is used to view the web documents. The web documents are written in a particular language supported by WWW. Hypertext MarkUp Language is used to represent the web content on web page.

In order to use the World Wide Web, one must have:

- Host Server
- WWW Client
- Internet Connectivity

Self Check Exercise

1) What is World Wide Web?

.....

.....

.....

.....

8.3 WHAT CAN BE DONE WITH WWW?

WWW can be used to search for documents, files, pictures, sounds, movies and more on remote computers all over the Internet. Just about every major computer company, and many non-computer companies, have a page on the web containing useful information about their goods and services. There are WWW based shopping malls, banks, libraries, record stores, and even correspondence schools.

Essentially, WWW is designed for the display, distribution and searching, of information, files, and data across multiple machines on the internet.

8.4 UNIFORM RESOURCE LOCATOR (URL)

A URL is a Uniform Resource Locator. A URL is the address of a page by which it is accessed. URL consists of three things, Protocol, server name and

file path, for example, <http://www.isibang.ac.in/drtc/faculty.htm>.

Examining this URL we can see the first part i.e. <http://> is protocol. That means the page is http page. www.isibang.ac.in <http://www.isibang.ac.in> is the address of server. /drtc is the directory under the server's root directory where one file called faculty.htm is present which is being currently displayed.

URL concept is really pretty simple. URLs are of various types:

- File URLs
- Gopher URLs
- News URLs
- HTTP URLs
- Partial URLs

Self Check Exercise

2) What is a Uniform Resource Locator (URL)?

.....

.....

.....

8.5 HYPERTEXT, HYPERLINK AND HYPERMEDIA

The literal meaning of the term Hypertext is a text which is linked with other text. Hypertext is basically the same as regular text - it can be stored, read, searched, or edited - with an important exception.

For example, in the below mentioned example the term *Hypertext* is related to another document which is the Webster's definition of the term *Hypertext*. Once someone selects the Hypertext of the first document he/she will move to the next document which is the dictionary definition of the term. Thus the documents linked like this are known as Hypertext and the linking is known as Hyper-linking, through which it can link to other text.

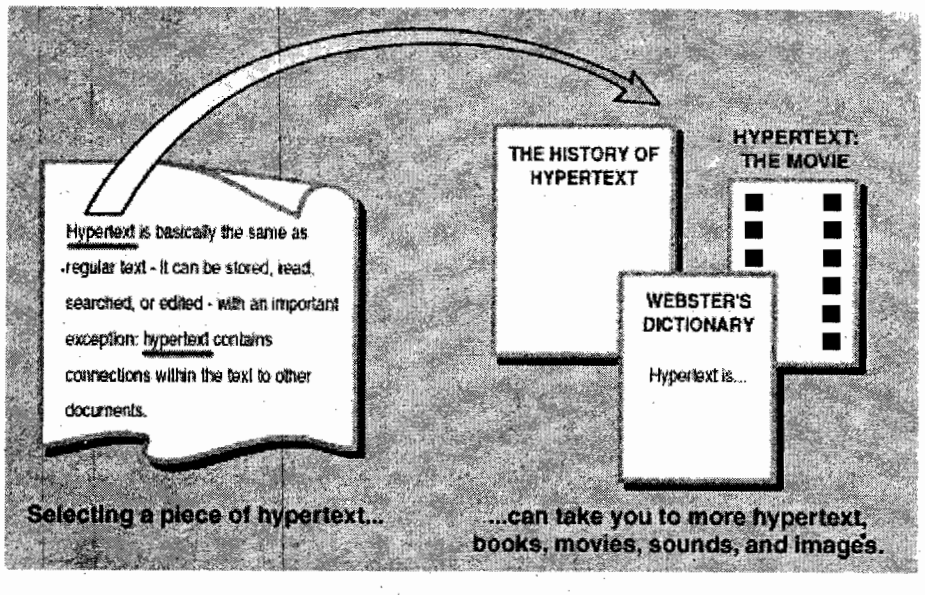

Fig.1: How hypertext works

Source: *http://www.maths.tcd.ie/local/JUNK/guide/guide.02.html*

Hypermedia is something more than hypertext. These documents not only contain the links but also use different media. For example, if one searches the meaning of a word at Merrium Webster's website (*http://www.m-w.com/*) . one get the meaning of the text besides one can listen to the pronunciation of the word if one has sound card installed on the machine. That means when besides text other media is also used it is known as Hypermedia.

Self Check Exercise

3) Distinguish between Hypertext, Hyperlink and Hypermedia.

.....

.....

.....

.....

8.6 MARKUP AND MARKUP LANGUAGES

The meaning of markup means instructions for printing in a particular style, for example, while proofreading, editors mark the text (e.g. underlined) to write the text in bold while printing. Similarly to display the electronic text in webpage the embedded instructions are given within the text to make the parser understand how the text should appear on display.

But MarkUp is also used for data retrieval, particularly in the library and information field. Once the structure of a document is fixed one can easily find out which part of the document contains which kind of data. For example, an email has a fixed structure. That means it will look like:

To: rama@ignou.ac.in

From: sita@ignu.ac.in

Date: Tue, 26 Jan 2002 01:00:58 -0800

Subject: Memo

From the time of the receipt of this letter you are promoted to chief librarian of University library.

Regards

Sita

If we observe this email we will find the following fields,

To:

From:

Date:

Subject:

Body:

It is very easy to fetch the data from an email once the fields are known. This a typical example of MarkUp.

A "markup language", may be no more than a loose set of markup conventions used together for encoding texts. A markup language must specify what markup is allowed and the whereabouts, what markup is required, how the markup is

to be distinguished from text, and what the markup means. SGML provides the means for doing the first three of these only; it allows one to describe a markup language independently of what the markup is intended to do. To understand and act upon the markup, additional semantic information is needed, which will differ in different situations. An SGML-aware processor can analyze the structure of an SGML-encoded document with no sense of its meaning. This independence is necessary, given the open-ended nature of electronic textual applications. It does not, of course, imply that the intentions of the encoder of a text are unimportant or vacuous; only that they are formally distinct from the encoding itself.

Three basic concepts are fundamental to an understanding of all markup languages, when described in SGML terms. These are the notions of a markup *entity*, a markup *element*, with its associated *attributes*, and a *document type*. At the most primitive level, texts are composed simply of streams of symbols (characters or bytes of data, marks on a page, graphics, etc.): these are known as *entities* in SGML. At a higher level of abstraction, a text is composed of representations of objects of various kinds, linguistically or functionally defined. Such objects do not appear randomly within a text: coherence demands that particular types of object appear in specifiable relationship to other objects — they may be included within each other, linked to each other by reference or simply presented sequentially, for example. This level of description sees texts as composed of structurally defined objects, known as *elements* in SGML. The grammar defining how elements may legally be combined in a particular class of texts is known as a *document type*. These three fundamental concepts together are, it is claimed, adequate to describe all the complexities of marked-up texts, of whatever kind and for whatever purposes.

Self Check Exercise

4) What is a markup?

.....

.....

.....

.....

8.7 SGML (STANDARD GENERALIZED MARKUP LANGUAGE)

SGML is not meant for formatting of text. Basically it was meant to preserve the structure of a document. SGML efforts to give a general structure for other MarkUp languages. Thus it is a meta-language which gives rise to other MarkUp languages, for example, XML (eXtensible MarkUp Language) is a derivative of SGML. It basically preserves the semantics of the text through the embedded text.

SGML is not a kind of text formatting system (although its origins can be readily traced in the world of electronic text formatting), or as a competitor for such languages as TeX or PostScript. These languages define how the text should appear on screen or over print. SGML by contrast is decidedly unhelpful about how texts are to be reproduced but it binds one to a specific structure of document and the sequence of elements in the text.

HTML is a relatively simple language, but using it becomes even easier if one understands the basic principles behind it, and take its limitations into account.

HTML stands for HyperText Markup Language. An HTML "page" is a plain text document with *markUp* inserted into it. This markUp includes codes for forming *hypertext links*. There is a need to understand the underlying principles.

8.8 HTML (HYPERTEXT MARKUP LANGUAGE)

HTML is a content-based or structural markUp language, where the codes describe what the contents of the document are. This means that the codes are used to indicate the various parts of the document, such as headings, paragraphs, lists, etc.

It is platform-independent. This means that HTML documents are portable from one computer system to another.

8.8.1 What it isn't?

There are some misconceptions about HTML:

- HTML is not a programming language. The markup in an HTML document describes the contents - it does not contain processing instructions.
- HTML is not a page layout language. With only a few exceptions, HTML tags are concerned with the structure of a document rather than its appearance.

Basically HTML cannot be called a strict structural language and the same is true when it comes to describing it as a page layout language. The tag <H1> i.e. heading tag is basically a structural tag which says which the text embeded is Heading of first order. But similarly HTML has <B> i.e. bold, <I> i.e. itallics etc/are formatting or page layout tags. That means it doesn't fall strictly in any category.

8.8.2 Introduction to HTML

HTML is the language with which Web pages may be designed. HTML allows web documents to be created with ease. The primary objective of using HTML would be to build a web page that communicates readily and effectively to make the document on the web most compelling to access and read.

This following section describes to the user some of the very basic HTML concepts, tags and features.

8.8.3 Editor for HTML

HTML is a plain text file and needs a simple text editor to create the tags. However, it is important that all HTML documents have the extension .html which is a four letter extension. As most editors allow only three letters, it is important to select an editor that allows four letters as the file extension. MS-DOS "edit" may be used as an editor for writing the HTML files.

8.8.4 Syntax of HTML Commands

In general, all HTML commands will take the form:

<COMMAND> text </COMMAND>.

Two points need to be noted:

All commands MUST be enclosed within angular brackets < >.

All commands are used in pairs wherein the <COMMAND> marks the beginning and </COMMAND> marks the end.

8.8.5 Frame Work of a Web Page

The framework of a web page is this:

```
<HTML>
<HEAD>
<TITLE> Title of Your Page </TITLE>
</HEAD>
<BODY>
The Body of Your Page
</BODY>
</HTML>
```

The <HTML> </HTML> tells the browser that your page is in HTML code.

The <HEAD> </HEAD> encloses the header of your page.

The <BODY> </BODY> is that part of your page that will actually be displayed.

8.8.6 HTML and the Browser

What is typed as HTML tags can be viewed only through a browser. It is hence necessary to constantly view the web page by switching into the browser mode as and when necessary. A windows based version allows you to keep both the editor window and the browser window open, thus making it easier to use.

8.9 XML (EXTENSIBLE MARKUP LANGUAGE)

HTML has problem of storing semantics of data. The gravity of the problem can be understood when someone searches the Internet for *Books on Ranganathan*, the results fetched by the search engines will have books on Ranganathan as well as books by Ranganathan. The problem of preserving the semantics can be easily addressed by XML.

According to the abstract from the XML Specification version 1.0:

"The eXtensible Markup Language (XML) is a subset of SGML that is completely described in this document (i.e. XML version 1.0 specificatoin). Its goal is to enable generic SGML to be served, received, and processed on

the Web in the way that is now possible with HTML. XML has been designed for ease of implementation and for interoperability with both SGML and HTML."

- XML stands for e**X**tensible **M**arkUp **L**anguage.
- XML is a **markUp language** much like HTML, structurally.
- XML was designed to **describe data**.
- XML tags are not predefined. You must **define your own tags**.
- XML might use a DTD (**Document Type Definition**) to describe the data.
- XML with a DTD is designed to be **self-descriptive**.

XML is still under development, and the following goals are kept in mind while developing the specification for XML.

i) XML shall be straightforwardly usable over the Internet.

ii) XML shall be compatible with SGML.

iii) It shall be easy to write programs which process XML.

iv) The processors can read the XML document easily.

v) XML document should be human-legible and reasonably clear.

vi) The XML design should be prepared quickly.

vii) The design of XML should be formal and concise.

viii) XML document shall be easy to create.

ix) Terseness in XML is of minimum importance.

8.9.1 How XML is different from HTML

i) XML was designed to attach semantic to data.

HTML has nothing to do with semantics data. It only defines how the page should be presented (like, font, color etc.).

ii) XML is not a replacement for HTML.

Many have a misconception that XML will replace HTML but whatever the case finally the actual representation is done in HTML format.

iii) XML is about describing information.

HTML is about displaying information.

8.9.2 When Can be done with XML?

XML does not DO everything

XML is not designed to DO everything. Maybe it is a little hard to understand, XML is not made to DO everything. XML is created as a way to structure, store and send information.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <book>
  <title>Prolegomena to library classification</title>
- <author>
  <f_name>Ranganathan</f_name>
  <l_name>S.R.</l_name>
  </author>
  <edition>3rd reprint</edition>
  <place>Bangalore</place>
  <publisher>Sarada Ranganathan Endowment</publisher>
  <physical_desc>640 p.</physical_desc>
  </book>
```

The example shows the structure of a document, which describes a book, titled *Prolegomena to library classification.* The book has a title, author, edition, place, publisher, physical description elements. Author is further divided into first name (f_name) and last name (l_name). Inside these tags the actual data is stored. If one sees the document in the web browser, data will appear embedded in the tags without having any kind of formatting.

Customized tags

In the above-mentioned example, <book> tag is defined by the person who is describing the document. Thus one can see that XML provides the facility to define user-customized tags. It is contrary to HTML where the tags are fixed and predefined. So the XML is used to create domain specific tag set which facilitates the information interchange within a specific domain. For example, NewsML is developed for information interchange among the news agencies like Reuter and others.

Data exchange

As XML allows to attach semantics to the data, data can be exchanged between incompatible systems. In the real world, the data stored in computer systems and databases, are usually in incompatible formats. One of the most time-consuming challenges for developers has been to exchange data between such systems over the Internet.

Converting the data to XML greatly reduces complexity. Many applications can easily read such data.

Share data

With XML, plain text files can be used to share data. Since XML data is stored in plain text format, XML provides a software- and hardware-independent way of sharing data.

This makes it much easier to create data that different applications can work with. It also makes it easier to expand or upgrade a system to new operating systems, servers, applications, and new browsers.

XML can make data more useful

With XML, your data is available to more users. Since XML is independent of hardware, software and application, you can make your data available to more than only standard HTML browsers.

Other clients and applications can access your XML files as data sources, like they access databases. Your data can be made available to all kinds of "reading machines".

XML can be used to create new languages

XML is the mother of WAP (Wireless Application Protocol) and WML (Wireless Markup Language), the Wireless Markup Language, used to markup Internet applications for handheld devices like mobile phones, is written in XML.

Self Check Exercise

5) Why is XML needed over HTML?

.....

.....

.....

.....

8.9.3 XML Syntax

Let us consider the first line of above mentioned example,

```
<?xml version="1.0" encoding="UTF-8" ?>
```

This line opens and closes with an angular bracket and a question mark, which suggests to XML parser that this document follows XML version 1.0 specification given by W3C and the character encoding system is used for data representation is *UNICODE Transformation Format-8*. The second line is - <book>, which is nothing but collapsible tags which shows that this tag has child elements. For each starting tag there is a closing tag e.g. the tag <book> ends with closing tag </book>. <book> has several child element like <title> <author>, <edition>, <place>, <publisher> and <physical_desc>. A child can have further sub-children as in case of <author>.

```
- <author>
  <f_name>Ranganathan</f_name>
  <l_name>S.R.</l_name>
 </author>
```

Inside the tags actual data is stored for example,

```
<title>Prolegomena to library classification</title>
```

XML tags are case sensitive and should be properly nested

Unlike HTML, XML tags are case sensitive. With XML, the tag <Author> is different from the tag <author>. Opening and closing tags must therefore be

written with the same case. All XML elements must be properly nested. Improper nesting of tags makes no sense to parser. For example,

<edition>**3rd reprint**</edition>

<place>**Bangalore**

<publisher></place>**Sarada Ranganathan Endowment**</publisher>

All XML documents must have a root tag

The first tag in an XML document is the root tag. All XML documents must contain a single tag pair to define the root element. All other elements must be nested within the root element. All elements can have sub elements (children). Sub elements must be correctly nested within their parent element. In the above mentioned example the <book> is the root element and all the other tags are child to it.

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

XML elements

An element is a component of a document. Elements can be made up of other elements, other types of data, or a descriptive representation that tells the XML parser about a resource that exists in document. Thus,

- XML Elements have simple naming rules.
- XML Elements are Extensible. XML documents can be extended to carry more information.
- XML elements have relationships. All the elements inside the <book> are child elements for <book>. This relationship indicates that <title> <author>, <edition>, <place>, <publisher> and <physical_desc> are describing an element *book*.

Thus the tags used like <book>, <author>, <place>, <publisher> etc. are elements.

Element naming

XML elements must follow these naming rules:

- Names can contain letters, numbers, and other characters. For example, <author1> ...</<author1>
- Names must not start with a number or other punctuation characters. For Example, it is illegal to have tags like, <856> ... </856> or <:856> ... </ 856>
- Names must not start with the letters xml (or XML or Xml ..).
- Names cannot contain spaces. For Example, it is illegal to have tags like, <first author> ... </first author>

- Any name can be used, no words are reserved, but the idea is to make names descriptive. Names with an underscore separator are nice.
- Examples: <f_name>, <l_name>.
- Avoid "-" and "." in names. It could be a mess if your software tried to subtract name from first (f-name) or think that "name" is a property of the object "first" (f.name).
- Element names can be as long as you like but names should be short and simple, for example, <book_title>

  not like,

  <the_title_of_the_book>
- Non-English letters like éòá are perfectly legal in XML element names, but watch out for problems if your software vendor doesn't support them.
- The ":" should not be used in element names because it is reserved for namespaces.

XML attributes

Attributes are used to provide additional information about elements. In good old HTML we often use attributes to get extra effects while formatting. For example,

```
<font size= "12" color= "red">Hello World</font>
```

will show the "Hello World" text in 12 font size and red colored. The size and color used are nothing but pre-defined attributes to the <font>.

Similarly, in XML also one can define the attributes. Attribute values must be quoted and it is illegal to omit quotation marks. XML elements can have attributes in name/value pairs just like in HTML. It can further extend file *book.xml* as:

```
1—<?xml version="1.0" encoding="UTF-8" ?>
2—— <book>
3— <title>Prolegomena to library classification</title>
4—— <author authorship="primary">
5— <f_name>Ranganathan</f_name>
6— <l_name>S.R.</l_name>
7— </author>
8— <edition>3rd reprint</edition>
9— <place>Bangalore</place>
10— <publisher>Sarada Ranganathan Endowment</publisher>
11— <physical_desc>640 p.</physical_desc>
12— </book>
```

NOTE: Here 1, 2, 3..... represents the line number of program.

Line 4 - <author authorship="**primary**"> has an attribute called as *authorship* which has value "*primary*". One can have any number of attributes associated with a single element.

There are some problems associated with using attributes:

- attributes cannot contain multiple values (child elements can)
- attributes are not easily expandable (for future changes)
- attributes cannot describe structures (child elements can)
- attributes are more difficult to manipulate by program code
- attribute values are not easy to test against a DTD

So it is always good to use child elements in spite of using attributes to describe an object.

8.9.4 DTD (Document Type Definition): Well-formed and Valid Document

One can define his own structure of XML document and give others to write the XML document against his own to avoid schema mistakes. A schema is nothing but the logical structure of document. This schema is called DTD (Document Type Definition). When the XML document is prepared against DTD it is called a *Valid document* and when there is no DTD for the document and the syntax of document is correct it is known as *Well-formed* document.

A DTD can be defined for a Valid-document. The declaration of DTD used for the validation is given in the processing tag of XML file.

8.9.5 XML and LIS

XML can have implications in library environment. The first and foremost use of XML can be sought in information exchange. We know that we are sitting on a heap of MARCs, and ironically this heap of standard MARCs has created a kind of non-standardization. In such a condition XML can be used as a common platform for information exchange provided atleast everyone will have acceptance to a common set of tags.

XML can also be used in Digital libraries. It can be used for document surrogate as a catalogue. It will be still an ambitious statement to make that XML can beat DBMS (Database Management Systems) and can be a solution for BDBMS (Bibliographic Database Management Systems) but one day it can happen.

Searching is another area where XML is of great help. As it provides context to search term searching becomes efficient particularly when we are agreed to follow a set of tags. XML can improve the search efficiency of current search engines. There are projects under development to identify schemas to perform search. RDF (Resource Description Framework) is one initiative in this direction.

With XML we can define the tags. These tags have the semantic value such

that author tag contains the name of author. Once we define a set of tags in a particular subject field, it becomes easy to transport data from one machine to other. For example, NewsML is a very good initiative in this direction as lot of news information have to be transferred from one place to other. The NewsML tag set provides a standard for data interchange among the news agencies. Currently Reuter is taking care of NewsML. More information regarding NewsML is available at *http://newsml.org*.

Finally, many think that with XML, formatted display is a tedious job. This is because currently we are in the world of HTML and the objective of XML is not the display in browser but to store data in a more meaningful manner. But the technology is so fluid that it would be no wonder tomorrow if we get an interface tool to write formatted XML document.

Based on the discussion on XML attributes, self check exercises may be given here which the student has to write in XML.

8.10 SUMMARY

After study of this chapter we are able to answer,

What is WWW, Hypertext, Hypermedia.

To put the information on Internet one should know atleast basic HTML tags. Though HTML has certain problems associated with it for example, inability to handle efficient search, but still it is widely used for web page design. XML is another derivative of SGML, which is also used to render the information on the web. The XML preserves the context of the term as well as its semantics.

An XML file also like HTML is a plain ASCII file, where one can define his/her own tags.

Self Check Exercise

6) Describe the library applications of XML.

.....

.....

.....

.....

8.11 ANSWERS TO SELF CHECK EXERCISES

1) WWW is actually a collection of traditional Internet access methods (FTP, Gopher, Telnet, etc.) and a new communications method called Hyper Text Transport Protocol (HTTP).

   WWW uses the concept of a page for viewing information. Each page is actually a single text files written in something called HyperText Markup Language (HTML). This HTML file is retrieved from a remote computer, known as the HTTP Server, by a WWW browser, and is used to determine the appearance of that particular WWW page. An HTML document can contain pointers to other HTML documents, graphics, files, sounds, and

even descriptions for buttons and other on-screen elements for displaying data. This interconnection of HTML documents on computers all over the Internet, each containing pointers to other HTML documents on other computers on the internet has created a kind of web of virtual documents and that is why, the term "web" came.

2) A URL is a **Uniform Resource Locator**. It is nothing but the unique address of the web document. One can think of it as a networked extension of the standard *filename* concept: not only can one point to a file in a directory, but that file and that directory can exist on any machine on the network, can be served via any of several different methods, and might not even be something as simple as a file: URLs can also point to queries, documents stored deep within databases, the results of a *finger* or *archie* command, or whatever.

   URLs are of different kinds:

   - **File URLs**

     file:// ftp.ignou.ac.in /pub/files/foobar.txt

   - **Gopher URLs**

     gopher://gopher.ignou.ac.in /

   - **News URLs**

     news: ignou.ac.in

   - **HTTP URLs**

     http://www. ignou.ac.in /pub/files/foobar.html

   - **Partial URLs**

   Once you are viewing a document located somewhere on the network (say, the document http://www. ignou.ac.in /pub/afile.html), one can use a *partial*, or *relative*, For example, if another file exists in that same directory called "anotherfile.html", then anotherfile.html is a valid partial URL at that point.

3) **Hypertext** is basically the same as regular text - it can be stored, read, searched, or edited - with an important exception: hypertext contains connections within the text to other documents.

   When on selection any specific part of document gives access to other document this is known called **Hyperlinks** and this can create a complex virtual web of connections.

   Hypermedia is hypertext with a difference - hypermedia documents contain links not only to other pieces of text, but also to other forms of media - sounds, images, and movies.

4) The word *markup* was originally used to describe annotation or other marks within a text intended to instruct a compositor or typist how a particular passage should be printed or laid out.

   A "markup language", may be no more than a loose set of markup conventions used together for encoding texts. A markup language must

specify what markup is allowed and its whereabouts, what markup is required, how markup is to be distinguished from text, and what the markup means.

5) eXtensible Markup Language is a kind of markup language.

It has certain advantages over HTML.

- XML can carry data.
- XML was designed to describe data and to focus on what data is.
- HTML is about displaying information. XML is about describing information
- XML is extensible. One can define own tags
- XML is used to exchange data while the same is very difficult with HTML
- XML is also considered a meta-language. Thus XML can be used to create new languages.

6) XML can have implications in library environment. The first and foremost use of XML can be sought in information exchange. We know that we are sitting on the heap of MARCs, and ironically this heap of standard MARCs has created a kind of non-standardization. In such a condition XML can be used as a common platform for information exchange provided atleast everyone will have acceptance to a common set of tags.

XML can also be used in Digital libraries. It can be used for document surrogate as a catalogue. It will still be an ambitious statement to make, that XML can beat DBMS (Database Management Systems) and can be a solution for BDBMS (Bibliographic Database Management Systems) but one day it can happen.

Searching is another area where XML is of great help. As it provides context to search term searching becomes efficient particularly when we are agreeable to following a set of tags. XML can improve the search efficiency of current search engines. There are projects under development to identify schemas to perform search. RDF (Resource Description Framework) is one initiative in this direction.

8.12 KEYWORDS

ANSI : American National Standards Institute, an organization that sets many standards for the computer industry.

ASCII : American Standard Code for Information Interchange, the mapping of ordinary letters and numbers to standard numerical representations. Often used to refer to plain text that does not contain word-processing codes.

Attribute : A setting for a tag, that affects the way the tag is displayed.

Boolean : Refers to something that can be True or False. A checkbox is a good way for a form to get a true or false answer from a user.

Browser : A program used to access and display web pages. Graphical browsers can display images and many different text fonts; non-graphical browsers cannot.

CGI : Common Gateway Interface is a way to allow users to provide information to scripts attached to web pages, usually through forms.

Cyberspace : The imaginary space users of the web move around in. A metaphor that many people take almost literally.

Domain Name : The name of an Internet site, for example www.dell.com or www.indiatimes.com.

Font : A font, strictly speaking, is a set of characters that all belong to the same size and style of a typeface. For example, Courier.

Forms : The mechanism by which web pages become interactive, allowing users to supply input to CGI or other scripts.

FTP : File Transfer Protocol, a way to exchange files with other sites on the Internet.

Gopher : A protocol that is older than HTTP and serves a similar purpose, allowing users to tunnel through cyberspace in search of information.

Graphic : A picture or illustration, also called image.

HTML : HyperText Markup Language, the language in which web pages are written.

HTTP : HyperText Transfer Protocol, the conventions used by web browsers and servers to transfer web pages.

Hypermedia : A combination of hypertext and multimedia that allows users to move in a non-linear fashion through text, images, sounds, and other information.

Hypertext : A collection of documents joined by links so that users can read it in a variety of different orders.

Image File : A file containing an image.

Indexers : Programs that read pages throughout the web and add a description of their contents to a database that can be searched by users looking for specific information.

Link : The anchor tag (<A>) is used to define both anchors and links. A link is a directive to a browser: when a user selects a link a new page is loaded. Some people call a link a *hotlink* or *hyperlink.*

Multimedia : The combination of several different communication techniques: for example sound, written text, still pictures, and moving pictures.

Nested : An element that is entirely contained within another element. For example, the phrase "*the **quick** brown fox*" contains a bold element (the word "quick") nested within an italic element (the entire phrase.) Some browsers will display the word "quick" only as bold, others will display it as both bold and italic.

Server : A program running on an Internet site that makes the web pages at that site available to browsers throughout the Internet.

SGML : Standard Generalized Markup Language. HTML is a derivative of SGML.

Site : Internet website.

Tags : Tags are metadata which embeds the information in it.

URL : Uniform Resource Locator, a description of the location of a link or image file. It specifies the *protocol* (http:// for a web page,) *site name, path and file name* to the resource.

8.13 REFERENCES AND FURTHER READING

What is the World Wide Web? *http://www.swpco.com/internet/alg_web.html*

A Beginner's Guide to URLs.
http://www.selu.edu/Academics/Depts/Cmps/jhu/urlprimer.htm

What is hypertext and hypermedia?
http://www.maths.tcd.ie/local/JUNK/guide/guide.02.html

Sol, Selena. What is a Markup Language?
http://www.wdvl.com/Authoring/Languages/XML/Tutorials/Intro/what_is_markup_language.html

Gorman, Dianne SGML and HTML: A Guide to Resources.
http://www.awpa.asn.au/sgml/

Welcome to SGML on Web.
http://archive.ncsa.uiuc.edu/SDG/Software/Mosaic/WebSGML.html

Introduction to HTML: Understanding HTML.
http://www.awpa.asn.au/html/html.html

Blue Book (1988). Volume VIII - Fascicle VIII.8, Data Communication Networks Directory, Recommendations X.500-X.521, CCITT.

GN3. What is a markup language? *http://www.faqs.org/faqs/text-faq/section-4.html*

Horton, M., and R. Adams (1987). Standard for interchange of USENET messages, RFC 1036, AT&T Bell Laboratories, Center for Seismic Studies, December 1987.

Kantor, B., and P. Lapsley (1986). Network News Transfer Protocol: A Proposed Standard for the Stream-Based Transmission of News, RFC 977prop, UC San Diego & UC Berkeley, February 1986.

Lang, R., and R. Wright (1992). A Catalog of Available X.500 Implementations, FYI 11, RFC *1292*(-> *1632*(-> 2116fyi11)), SRI International, Lawrence Berkeley Laboratory, January 1992.

Schwartz, M., and P. Tsirigotis (1991). Experience with a Semantically Cognizant Internet White Pages Directory Tool, *Journal of Internetworking Research and Experience*, March 1991, pp. 23-50.

W3C.org. *http://www.w3.org/XML/*

Weider, C., and J. Reynolds (1992). Executive Introduction to Directory Services Using the X.500 Protocol, FYI 13, RFC 1308fyi13, ANS, ISI, March 1992.

Weider, C., Reynolds, J., and S. Heker (1992). Technical Overview of Directory Services Using the X.500 Protocol, FYI 14, RFC 1309fyi14, ANS, ISI, JvNC, March 1992.

Williamson, S. (1993). Transition and Modernization of the Internet Registration Service, RFC 1400, Network Solutions, Inc., March 1993.

World Wide Web. *http://helpdesk.uvic.ca/resource/network/www.html*