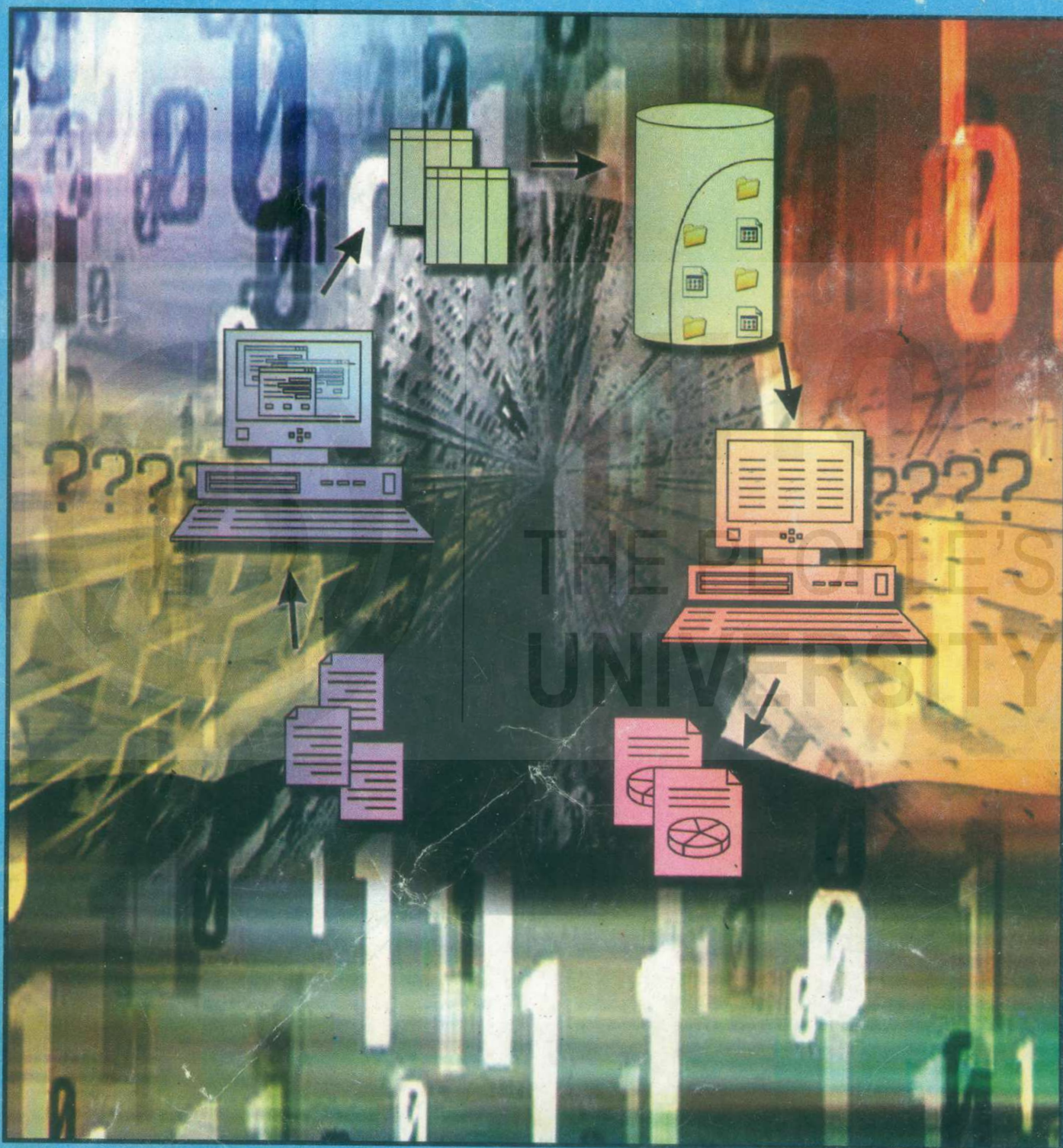




Master's Degree in Library and Information Science (MLIS)



**INFORMATION PROCESSING AND RETRIEVAL
PART II**



MLII-102

Information Processing and Retrieval

Introduction	5
---------------------	---

Part I

Block 1 Organisation of Information

Unit 1: Intellectual Organisation of Information	9
Unit 2: Indexing Languages – Part I: Concepts and Types, Subject Headings Lists and Thesauri	33
Unit 3: Indexing Languages – Part II : Classification Schemes	56
Unit 4: Indexing Systems and Techniques	89
Unit 5: Evaluation of Indexing Systems	177

Block 2 Bibliographic Description

Unit 6: Principles and Evolution of Bibliographic Description	211
Unit 7: Rules for Bibliographic Description	247
Unit 8: Standards for Bibliographic Record Format	277
Unit 9: Metadata : MARC21-856 Field, Dublin Core, TEI	296

Part II

Block 3 Content Development

Unit 10: Norms and Guidelines for Content Development	331
Unit 11: Introduction to HTML and XML	352
Unit 12: Web Based Content Development	378
Unit 13: Multilingual Content Development (Using Unicode)	394

Block 4 Information Storage and Retrieval Systems

Unit 14: ISAR Systems : Objectives, Types, Operations and Design	413
Unit 15: Compatibility of ISAR Systems	433
Unit 16: Intelligent Information Retrieval Systems	448

Block 5 Information Retrieval

Unit 17: Information Retrieval Processes and Techniques	465
Unit 18: Information Retrieval Models and Their Applications	497
Unit 19: Search Strategies, Processes and Techniques	516

Programme Design Committee (Original) (1992)

Prof. Pandav Nayak (<i>Chairman</i>)	Mr. N.M. Malwad	Dr. (Ms.) Neela Jagannathan (<i>Special Invitee</i>)
Prof. (Ms.) A.K. Anand	Dr. S.S. Murthy	Dr. R. Satyanarayana (<i>Convener</i>)
Prof. J.C. Binwal	Prof. K.S. Raghavan	
Prof. M.A. Gopinath	Prof. T.N. Rajan	Internal Members
Prof. B. Guha	Prof. A.P. Srivastava	Dr. Uma Kanjilal
Prof. S.R. Gunjal	Prof. T. Viswanathan	Ms. Neena Talwar Kanungo
Prof. S.G. Mahajan		

Course Preparation Team (1992)

Prof. M.A. Gopinath
Prof. A.A.N. Raju
Dr. S. S. Murthy
Dr. M.P. Satija
Course Editor: Dr. R. Satyanarayana

Programme (Curriculum) Revision Committee (2003)

Prof. S.B. Ghosh (<i>Chairman</i>) Faculty of Library & Information Science School of Social Sciences IGNOU, New Delhi	Prof. C.R. Karisiddappa Head, Dept. of Library & Information Science Karnataka University Dharwar	Prof. J. Sarkhel Head, Dept. of Library & Information Science Vidyasagar University Midnapur W.B.
Dr. Jagdish Arora Librarian Indian Institute of Technology Mumbai	Dr. A. Lahiri Senior Advisor Dept. of Scientific & Industrial Research New Delhi	Prof. R. Satyanarayana Faculty of Lib. & Inf. Sc. (Retd.) IGNOU, New Delhi
Prof. S.R. Ganpule Dept. of Library & Information Science Yaswant Rao Chavan Maharashtra Open University, Nasik	Prof. P.B. Mangla Dept. of Lib. & Inf. Sc. (Retd.) University of Delhi Delhi	Prof. B.K. Sen Head, (Retd.) Education Division INSDOC New Delhi
Prof. B. Guha Professor (Retd.) Dept. of Lib. & Inf. Sc. Banaras Hindu University Banaras, U.P.	Prof. A. Neelameghan UNESCO Expert (Retd.) Sarada Ranganathan Endowment for Library & Information Science Bangalore	Dr. Mahinder Singh Director DESIDOC, Delhi
Prof. Uma Kanjilal Faculty of Lib. & Inf. Sc. IGNOU, New Delhi	Prof. T.N. Rajan Chief Coordinator (Retd.) INSDOC, New Delhi	Dr. N. Vijayaditya Director General National Informatics Centre New Delhi
Dr. Neena Talwar Kanungo Faculty of Lib. & Inf. Sc. IGNOU, New Delhi	Dr. Pravakar Rath Faculty of Lib. & Inf. Sc. IGNOU, New Delhi	Prof. T. Viswanathan Ex-Director, INSDOC New Delhi
		Dr. (Ms) Neela Jagannathan (<i>Special Invitee</i>) IGNOU, New Delhi

Programme Coordinators: Prof. S.B. Ghosh and Prof. Uma Kanjilal

Course Coordinator and Editor: Prof. S.B. Ghosh

Course Preparation Team

Unit	Contributor	Academic Assistance
1	Prof. T.N. Rajan	Mr. Anup Kumar Das, Consultant
2	Dr. Jaideep Sharma	
3	Prof. M.P. Satija (Revised by Prof. S.B. Ghosh)	
4	Prof. J. Sarkhel	
5	Prof. J. Sarkhel	
6	Mr. P.S. Mukhopadhyay	
7	Dr. P.K. Panigrahi	
8	Dr. A.R.D. Prasad	
9	Dr. P. Devika	
10	Dr. P. Devika	
11	Dr. P. Devika	
12	Dr. P. Devika	
13	Dr. A.R.D. Prasad	
14	Dr. Aditya Tripathi (Revised by Prof. S.B. Ghosh)	
15	Dr. M.A. Gopinath, Dr. A.R.D. Prasad	
16	Dr. A.R.D. Prasad	
17	Dr. G.G. Chowdhury	
18	Dr. M.A. Gopinath, Dr. J.P. Singh	
19	Dr. G.G. Chowdhury	

Production

Mr. Jitender Sethi IGNOU, New Delhi	Mr. S.S. Venkatachalam and Mr. Manjit Singh	Secretarial Assistance Shri S. Swaminathan
--	--	--

June, 2006 (Reprint)

© Indira Gandhi National Open University, 2005

ISBN-81-266-2077-3

All rights reserved. No part of this work may be reproduced in any form, by mimeograph or any other means, without permission in writing from the Indira Gandhi National Open University.

Authors are responsible for the academic content of the courses as far as the copyright issues are concerned.

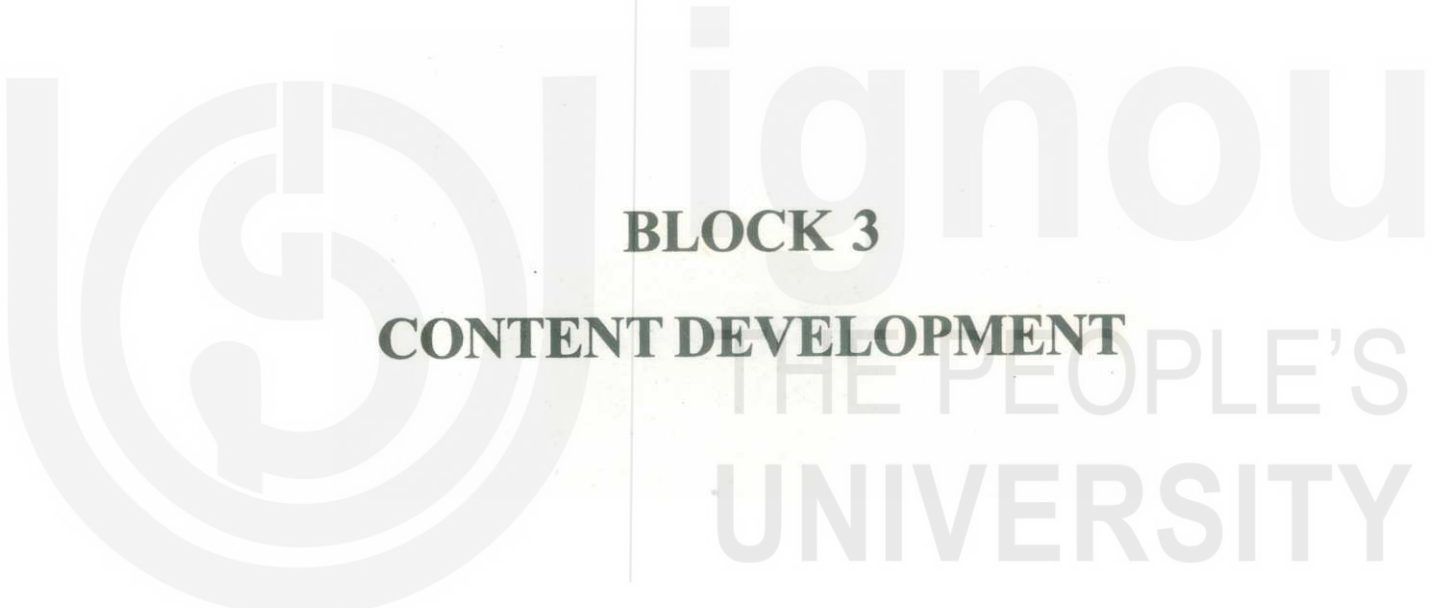
Further information on the Indira Gandhi National Open University courses may be obtained from the University's Office at Maidan Garhi, New Delhi-110 068.

Printed and published on behalf of the Indira Gandhi National Open University, New Delhi, by the Director, School of Social Sciences.

Paper Used : Agrobased Environment Friendly

Laser typeset by : HD Computer Craft, EA 1/75, Inderpuri, New Delhi-110012.

Printed at : Dee Kay Printers, 5/16, Kirti Nagar Indl. Area, New Delhi-110015.

The background of the page features a large, light gray watermark of the Open University logo. The logo consists of a stylized 'U' on the left and the text 'Open University' on the right, with 'THE PEOPLE'S UNIVERSITY' written in smaller letters below it.

BLOCK 3

CONTENT DEVELOPMENT

UNIT 10 NORMS AND GUIDELINES FOR CONTENT DEVELOPMENT

Structure

- 10.0 Objectives
- 10.1 Introduction
- 10.2 Needs and Guidelines
 - 10.2.1 Needs
 - 10.2.2 Some General Guidelines
- 10.3 Standards Related to Electronic Content
 - 10.3.1 ISO Standards
- 10.4 W3C Recommendations
 - 10.4.1 Web Content Accessibility Guidelines and Checkpoints
- 10.5 Electronic Text Encoding and Interchange
 - 10.5.1 Origin
 - 10.5.2 Objectives
 - 10.5.3 Features
- 10.6 Dynamic Content
- 10.7 Summary
- 10.8 Answers To Self Check Exercises
- 10.9 Keywords
- 10.10 References and Further Reading

10.0 OBJECTIVES

Logically, content development is not a new concept. However, in the context of developments in ICT and availability of information in varieties of forms, content development refers to the organisation of web information and, thus, assumes great importance. In the previous Units, you have learnt about norms and standards of record formats of non-web information. In this Unit we will be discussing the norms and standards developed for recording web information.

After reading this Unit, you will be able to:

- understand the need for guidelines for content development; and
- learn about the important guidelines formulated by various bodies such as W3C and ISO.

10.1 INTRODUCTION

The electronic information provides the opportunity of varied style of presentation of information as compared to traditional form of documents – printed form. Due to availability of technology and its options, it is possible to add multimedia components in documents. As a result there is a tremendous onslaught of documents which are graphic intensive. While images and other audio-visual methods are attractive and effective ways of communication, there should be careful analysis and reasoning before adding on many multimedia effects. Due to online accessibility of such documents it is necessary to keep in mind the time taken for transmission of such documents as well as the traffic burden on the

networks. The content needs to be well organized keeping in view the requirements and the level of understanding of the end users. Several guidelines have been provided by bodies such as ISO (International Organization for Standardization) and W3C (WWW Consortium) which specify the content and its presentation.

For the accessibility issues pertaining to web page design, it must be taken into consideration that the end users may be operating in contexts which are very different from the hosts' environment. Also that the users [Neilson, Jacob] may operate in varieties of situations. For example, he may not be able to see, hear, move, or may not be able to process some types of information easily or at all or may have difficulty in reading or comprehending text. Sometimes, he may not have or be able to use a keyboard or mouse, or may have a text-only screen, a small screen, or a slow Internet connection. At times it may be that he may not speak or understand fluently the language in which the document is written or may be in a situation where his eyes, ears, or hands are busy or interfered with (e.g., driving to work, working in a loud environment, etc.). It may so happen also that he may have an early version of a browser – a different browser entirely, a voice browser, or a different operating system.

Content developers must consider these different situations during page design. While there are several situations to consider, each accessible design choice generally benefits the web community as a whole and several disability groups as well. For example, by using style sheets to control font styles and eliminating the 'FONT' element, HTML authors will have more control over their pages, make those pages more accessible to people with low vision, and, by sharing the style sheets, will often shorten page download times for all users. The guidelines address typical scenarios (similar to the font style example) that may pose problems for users and provide accessible design solutions.

10.2 NEEDS AND GUIDELINES

10.2.1 Needs

Information explosion is of concern for information professionals and end users who are trying to harness information for their appropriate use. In the context of online and Internet documents the information available is voluminous and the rate at which resources are added is even more daunting. It is not only the matter of concern for end-users point of view but also of equal concern to the hosts who aim at having a target audience while hosting resources online. The efforts should, therefore, be to develop content and host it, so that it reaches its end-users.

In a distributed uncontrolled system of online resources, retrieval of information is one the main concern. The search facilities available through general Internet search engines are inadequate in harnessing all available information. Reason is not only the bulk of the information but also the non-standard way of authoring and presentation which makes searching difficult. If each electronic resource were to have a 'title page' or the information that occurs in the title pages of the documents then it would have been much easier to organize and locate information in an environment such as the Net. The need for standardised way of hosting electronic documents has lead to the development of few standards and guidelines. Strictly speaking, very few of these guidelines are meant for content aspects, rather they deal more with issues in structuring of documents and presentation aspects. For instance, the Sun Site follows certain premises for provision of content on their site [Neilson and Fox]: firstly that; "Writing for the web is very different from writing for print". Users' interaction with web pages is quite different form print in that:

- 79% of users scan the page instead of reading word-by-word;
- Reading from computer screens is 25% slower than from paper; and
- Web content should have 50% of the word count of its paper equivalent.

Another aspect which is important in content development is what the content itself should consist of, rather than just 'how' it should be presented. This issue has an important bearing especially in the web world where spurious and harmful information may be hosted with ulterior motives. Website hosts, therefore, give a great deal of thought to making adequate policies about the permissible content that will be hosted by them and also the screening procedures before going public.

Electronic resources in the global networked environment has brought new challenges in information processing. With the possibilities of access and usage in diverse locations, standardisation has become a major issue. Standardisation of the ways in which information is stored and represented has become a key in activities like generating secondary services and information locating tools. The main advantages of standardisation of representation facilitates:

- re- usability of information once created;
- true inter-operability with respect to different platforms and applications and multiple languages; and
- global networking and seamless integration of resources.

Self Check Exercise

1) Mention the need for guidelines and norms for 'content development'.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of this Unit.

.....

.....

.....

.....

.....

10.2.2 Some General Guidelines

The Internet and other electronic documents users often come across problems such as incompatibility of forms and formats, download time, difficulties in comprehending the content, incomplete information, etc. Adherence to some general practices and ethics by the content developers are necessary to assure reasonably well-written and organised resources for the end-users. According to Jacob Neilson, a web content analyst, the content developers should take care of the following points:

- Well-tested and widely used packages in designing web pages should be used. It is better to avoid recently released software, especially if there is use of plug-ins such as Flash or PDF. A good rule is not to use upgraded versions immediately till they mature. Software vendors require time to fix their bugs and offer reasonably reliable versions. It also gives users time to upgrade.
- As far as possible Scripts in web pages should be avoided. Adding code means that there is also a risk of bugs (or faults). If scripts are a must then they should be tested and debugged on all combinations of platforms and browser versions.

- iii) It is advisable to have simple point-and-click web navigation. Special menu controls often fail, especially for users who move their mouse fast or who have motor skill impairments.
- iv) The website should always be made as compact in size as possible. It is good to have spare server capacity so that your site can cope with traffic surges.
- v) Server software must be robust. When selecting vendors, ensuring code quality must be a priority.

According to Mick Wood, following guidelines help content developers to cater for the needs of all Internet users.

i) Screen Layout

Screen layout should be consistent. The more consistent a website is in its design, the easier it will be for users to navigate. Users, especially persistent users, tend to learn and remember the location of key functions and controls.

ii) Logos

Screen layout should include logos, navigation buttons and footer information. Putting the logo in a consistent place on every page (usually top left) ensures that visitors are fully aware that they are on the same site.

iii) Update Information

'Page Last Updated' information needs to be included. Visitors will quickly know if you have added any new content to your site, and it may prompt you to do so. The date format should be understandable, i.e., 1st February 2003 is better than 02-01-03. In the UK 02-01-03 means 2nd January 2003 and in USA it means 1st February 2003.

iv) Links

A link to the home page should be added to every page except the homepage itself. Pages should not link to themselves as this can confuse some users. Making your logo the home page link is considered to be a good practice.

v) Image Sizes

Considerable number of Internet users still use modems with connection speeds of 33K or less. Pages should therefore be kept to a maximum of 35K ensuring download times of less than ten seconds for these users. Visitors may not wait if your pages take too long to download. All images, including spacer images, should include alternative text. The alternative text should describe the function of the image, if applicable.

vi) Use of Frames

Frames should not be used unless absolutely necessary. Frame-based sites can be confusing for the visually disabled, particularly those using screen readers or speech browsers - users can easily become disorientated. Additionally, users cannot easily bookmark individual pages within a frame-based site.

vii) Page Title

Each page should have a descriptive and different title, clear and helpful headings, and a logical structure. Titles are used by search engines to identify web pages. If two or more pages have the same title they cannot easily be differentiated by users.

viii) Font Attributes

The FONT tag should generally be avoided in creating accessible websites.

While attribute specifications like SIZE="+1" or SIZE="-1" are relatively harmless, absolute sizes like SIZE="1" can result in text that is too small to read. Instead, Cascading Style Sheets should be used to format web pages. Style sheets allow authors to suggest relative changes in font size with much greater flexibility than is allowed under FONT.

ix) **Color Attributes**

The COLOR attribute of the FONT tag should always be avoided since many browsers still display the font colour when the user tries to override author-specified colours. The result could be an unreadable document if the font colour does not contrast well against the reader's chosen background.

x) **Text Size and Font**

Text should be the equivalent of size 12 points to enhance reading performance. Research has shown that there is no noticeable difference in reading speed or user preference between Times New Roman, Georgia or other serif fonts and Helvetica, Arial or other sans-serif fonts. The FACE attribute of the FONT tag cannot be overridden by the user in many browsers. This may result in the author choosing a font that is very difficult to read given the user's platform and environment settings. The same font may display strangely on different platforms.

xi) **Link Identification**

Links should be clearly identified and it should make sense when reading out of context. Links should never be designated with the text 'click here'. Some screen readers can be set to read out a list of links on a particular page; a list of 'click here' links is not helpful.

Blue underlined text is the preferred choice for all links. Some users miss links because the text is not underlined. Research shows that users can easily find links which include visual cues, that is, links that are underlined, rather than having to move the mouse to see when the pointer changes to a hand (this is known as mine sweeping). Visited links should be designated using a different colour. Many users use link colours to identify which parts of a site they have already visited.

xii) **Page Display**

Pages should be fluid/flexible because users browse the Internet using a variety of screen resolutions. Pages should be designed to expand or contract according to the user's settings, thus preventing horizontal scrolling. Horizontal scrolling is a particular problem for screen reader users because the screen reader does not automatically scroll horizontally – users may miss important content.

xiii) **Device Independence**

Pages should be device independent. Not all users navigate websites using a mouse – many disabled users use Assistive Technology. All users should be able to navigate the site using the input device of their choice. Keyboard shortcuts, using the 'accesskey' attribute, should be provided for key links.

xiv) **Readability**

Paragraphs and sentences should be kept short. Readability improves when sentences and paragraphs are kept relatively short. Users tend to scan web pages and will often skip over large chunks of text.

These guidelines are not formal guidelines. However, these need to be followed by the host of the electronic documents. The issues relating to the content and its organisation are complex and are under consideration for arriving at standard

practices by organisations such as the International Organization for Standardization (ISO) [www.iso.org]. The ISO 8879 deals with SGML, the Standard Markup Language for the web and has been adopted by Text Encoding and Interchange (TEI) initiative which deals with guidelines for structuring of web documents.

Self Check Exercise

2) Describe the general guidelines to be followed in developing a content.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

10.3 STANDARDS RELATED TO ELECTRONIC CONTENT

There has been an increasing awareness of the importance of the human interface to computer systems which has gained momentum over the past few years. It is no longer sufficient to provide reasonable function at a good price unless the system can be easily and productively used. To codify this usability and provide some consistency from interface to interface, several standards organisations are working to develop User Interface software standards which they perceive will provide benefits such as [www.iso.org] :

- providing enhanced usability which results in increased satisfaction and productivity.
- providing some assurance of the users well being and lack of frustration.
- providing needed consistency within and between systems and applications.
- helping software procurers in their selection of acceptable products.
- helping to protect the employer from complaints and possible litigation.

Self Check Exercise

3) What benefits can be had by adhering to standards?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

10.3.1 ISO Standards

Truly speaking, till date, no set standards exist as such for the content of electronic documents. However, there are related guidelines which deal with the issues of

10.3.1.1 Standards Developed by ISO/TC 46/SC9 for Bibliographic References Including Electronic Documents

- *ISO 690- 2:1997*, Information and documentation – Bibliographic references – Part 2: Electronic documents or parts thereof. *First edition*.
- *ISO 999:1996*, Information and documentation – Guidelines for the content, organisation and presentation of indexes. *Second edition*.
- *ISO 3297:1998*, Information and documentation – International Standard Serial Number (ISSN). *Third edition*.
- *ISO 3901:2001*, Information and documentation – International Standard Recording Code (ISRC). *Second edition*.
- *ISO 5963:1985*, Documentation – Methods for examining documents, determining their subjects, and selecting indexing terms. *First edition*.
- *ISO 10324:1997*, Information and Documentation – holdings statements – summary level. *First edition* [www.nlc-bnc.ca/iso/tc46sc9].

10.3.1.2 Standards Related to Design and Hosting of Electronic Documents with Multimedia Components

These are under consideration and formulation stage by ISO.

ISO 9241:	Ergonomic principles for visual display terminals.
ISO/IEC 10741:	What happens to the cursor control when users interact with text editors.
ISO/IEC 11581:	Usage and appropriateness of icons in the user interface.
ISO/IEC 13251:	Collection of graphical symbols for office equipment.
ISO 13407:	Designing user interfaces with humans in mind.
ISO/IEC 14754:	Defines the basic gesture commands.
ISO 14915:	Recommendations for multimedia controls and navigation.
ISO/IEC 18019:	A standard for the design and preparation of software user documentation.
ISO/IEC 18035:	Icon symbols and functions for multimedia applications.
ISO/IEC 18036:	Icon symbols and functions for World Wide Web browsers.

10.3.1.3 Standards Related to Structuring of Web Documents Using SGML

ISO 8879 SGML: Information Processing – Text and Office Systems – Standardised Generalised Markup Language (SGML), First Edition.

10.4 W3C RECOMMENDATIONS

The World Wide Web Consortium (W3C), an independent, international organisation made up of people from across the Internet and web development communities — from individuals to representatives of major corporations, like, IBM, Microsoft, Netscape, and Sun Microsystems, was formalised on December 14, 1994. Its function is to oversee the formal organisation of HTML, as well as the various web-related protocols and languages, including CSS, XML and XHTML. The Consortium's individual working groups focus on issues within a given technology.

They make recommendations regarding the formal practices which they believe that it should be put into use. The W3C recommendations—while based on those of the ISO—are not true standards. Web markup languages are not ISO standards, but recommendations are often referred to as standards.

10.4.1 Web Content Accessibility Guidelines and Checkpoints

The W3C guidelines include suggestions for content and as well as its presentation on the web and are meant for web content developers both the authors and also the website architects. Though these guidelines basically deal with accessibility issues and more useful to the physically challenged, they also make the content more available to users irrespective of which browser or PC they are using to view the pages. The recommendations also give guidelines for using multimedia components in the pages [W3C, 1999].

The guidelines enlisted below are part of the W3C “Web Content Accessibility Guidelines 1.0”. The guidelines are accompanied by a list of check points which may be used for cross verification and check list for web content developers. These checklist of points for each guideline may be used to review a page or site for accessibility. For each checkpoint the web developer may indicate whether the checkpoint has been satisfied, or has not been satisfied, or is not applicable.

Priorities

Each checkpoint has a priority level assigned by the Working Group based on the checkpoint’s impact on accessibility.

[Priority 1]

A web content developer **must** satisfy this checkpoint. Otherwise, one or more groups will find it impossible to access information in the document. Satisfying this checkpoint is a basic requirement for some groups to be able to use web documents.

[Priority 2]

A web content developer **should** satisfy this checkpoint. Otherwise, one or more groups will find it difficult to access information in the document. Satisfying this checkpoint will remove significant barriers to accessing web documents.

[Priority 3]

A web content developer **may** address this checkpoint. Otherwise, one or more groups will find it somewhat difficult to access information in the document. Satisfying this checkpoint will improve access to web documents.

Some checkpoints specify a priority level that may change under certain (indicated) conditions.

Guideline 1: Provide equivalent alternatives to auditory and visual content.

This guideline emphasises the importance of providing text equivalents of non-text content (images, pre-recorded audio, video). In case where one cannot use images, movies, sounds, applets, etc. directly, still pages that include equivalent information to the visual or auditory content may be used. The equivalent information must serve the same purpose as the visual or auditory content. Thus, a text equivalent for an image of an upward arrow that links to a table of contents could be “Go to table of contents”. In some cases, an equivalent should also describe the appearance of visual content (e.g., for complex charts, billboards, or diagrams) or the sound of auditory content.

Checkpoints

Provide a text equivalent for every non-text element (e.g., via 'alt', 'longdesc', or in element content). This includes: images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ASCII art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video. [Priority 1]

If multimedia is used:

- Until user agents can automatically read aloud the text equivalent of a visual track, provide an auditory description of the important information of the visual track of a multimedia presentation. [Priority 1]
- For any time-based multimedia presentation (e.g., a movie or animation), synchronise equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation. [Priority 1]
- Until user agents render text equivalents for client-side image map links, provide redundant text links for each active region of a client-side image map. [Priority 1]

Guideline 2: Do not rely on colour alone.

It must be ensured that text and graphics are understandable when viewed without colour. If colour alone is used to convey information, people who cannot differentiate between certain colours and users with devices that have non-colour or non-visual displays will not receive the information. When foreground and background colours are too close to the same hue, they may not provide sufficient contrast when viewed using monochrome displays or by people with different types of colour deficits.

Checkpoints

- Ensure that all information conveyed with colour is also available without colour, for example from context or markup. [Priority 1]
- Ensure that foreground and background colour combinations provide sufficient contrast when viewed by someone having colour deficits or when viewed on a black and white screen. [Priority 2 for images, Priority 3 for text].

Guideline 3: Use markup and style sheets and do so properly.

Markup documents with the proper structural elements. Control presentation with style sheets rather than with presentation elements and attributes. Using markup improperly and not adhering to specifications hinders accessibility. Misusing markup for a presentation effect (e.g., using a table for layout or a header to change the font size) makes it difficult for users with specialized software to understand the organisation of the page or to navigate through it. Furthermore, using presentation markup rather than structural markup to convey structure (e.g., constructing what looks like a table of data with an HTML PRE element) makes it difficult to render a page intelligibly to other devices. Content developers may be tempted to use (or misuse) constructs that achieve a desired formatting effect on older browsers. They must be aware that these practices cause accessibility problems and must consider whether the formatting effect is so critical as to warrant making the document inaccessible to some users.

At the other extreme, content developers must not sacrifice appropriate markup because a certain browser or assistive technology does not process it correctly. For example, it is appropriate to use the TABLE element in HTML to mark up tabular information even though some older screen readers may not handle side-

by-side text correctly. Using TABLE correctly and creating tables that transform gracefully makes it possible for software to render tables other than as two-dimensional grids.

Checkpoints

- When an appropriate markup language exists, use markup rather than images to convey information. [Priority 2]
- Create documents that validate to published formal grammars. [Priority 2]
- Use style sheets to control layout and presentation. [Priority 2]
- Use relative rather than absolute units in markup language attribute values and style sheet property values. [Priority 2]
- Use header elements to convey document structure and use them according to specification. [Priority 2]
- Mark up lists and list items properly. [Priority 2]
- Mark up quotations. Do not use quotation markup for formatting effects such as indentation. [Priority 2]

Guideline 4: Clarify natural language usage.

Use markup that facilitates pronunciation or interpretation of abbreviated or foreign text.

When content developers mark up natural language changes in a document, speech synthesizers and braille devices can automatically switch to the new language, making the document more accessible to multilingual users. Content developers should identify the predominant natural language of a document's content (through markup or HTTP headers). Content developers should also provide expansions of abbreviations and acronyms. In addition to helping assistive technologies, natural language markup allows search engines to find keywords and identify documents in a desired language. Natural language markup also improves readability of the web for all people, including those with learning disabilities and cognitive and physical disabilities.

Checkpoints

- Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions). [Priority 1]
- Specify the expansion of each abbreviation or acronym in a document where it first occurs. [Priority 3]
- Identify the primary natural language of a document. [Priority 3]

Guideline 5: Create tables that transform gracefully.

Ensure that tables have necessary markup to be transformed by accessible browsers and other user agents. Tables should be used to mark up truly tabular information ('data tables'). Content developers should avoid using them to lay out pages ('layout tables'). Tables for any use also present special problems to users of screen readers. Some user agents allow users to navigate among table cells and access header and other table cell information. Unless marked-up properly, these tables will not provide user agents with the appropriate information.

Checkpoints

If tables are used:

- for data tables, identify row and column headers. [Priority 1]

- for data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells. [Priority 1]
- do not use tables for layout unless the table makes sense when linearized. Otherwise, if the table does not make sense, provide an alternative equivalent (which may be a linearized version). [Priority 2]
- for layout, do not use any structural markup for the purpose of visual formatting. [Priority 2]
- provide summaries for tables. [Priority 3]
- provide abbreviations for header labels. [Priority 3]

Guideline 6: Ensure that pages featuring new technologies transform gracefully.

Ensure that pages are accessible even when newer technologies are not supported or are turned off. Although content developers are encouraged to use new technologies that solve problems raised by existing technologies, they should know how to make their pages still work with older browsers and people who choose to turn off features. One example is the use of frames. When frames are used a note should also be given that the page should be viewed using which versions of available browsers.

Checkpoints

- Organise documents so that they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document. [Priority 1]
- Ensure that equivalents for dynamic content are updated when the dynamic content changes. [Priority 1]

If applets and scripts are used

- ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page. [Priority 1]
- ensure that dynamic content is accessible or provide an alternative presentation or page. [Priority 2]
- for scripts and applets, ensure that event handlers are input device-independent. [Priority 2]

Guideline 7: Ensure user control of time-sensitive content changes.

Some people with cognitive or visual disabilities are unable to read moving text quickly enough or at all. Movement can also cause such a distraction that the rest of the page becomes unreadable. Screen readers are unable to read moving text. When using utilities like web casters which use push technology to send information to the screens. They offer many features like giving running text (tickers) and various other animations like screen bursts to present information. But these features may distract users in their routine work. In the design of web pages care should be taken when using animations and moving text so that either users could control the movements and/or speed so that users' convenience is achieved.

Checkpoints

- Until user agents allow users to control flickering, avoid causing the screen to flicker. [Priority 1]

- Until user agents allow users to control blinking, avoid causing content to blink (i.e., change presentation at a regular rate, such as turning on and off). [Priority 2]
- Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages. [Priority 2]
- Until user agents provide the ability to stop auto-redirect, do not use markup to redirect pages automatically. Instead, configure the server to perform redirects. [Priority 2]

If applets and scripts are used

- Until user agents allow users to freeze moving content, avoid movement in pages. [Priority 2]

Guideline 8: Ensure direct accessibility of embedded user interfaces.

Ensure that the user interface follows principles of accessible design: device-independent access to functionality, keyboard operability, etc. When an embedded object has its 'own interface', the interface – like the interface to the browser itself – must be accessible. If the interface of the embedded object cannot be made accessible, an alternative accessible solution must be provided.

Checkpoints

- Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]

Guideline 9: Design for device-independence.

Use features that enable activation of page elements via a variety of input devices. Device-independent access means that the user may interact with the user agent or document with a preferred input (or output) device – mouse, keyboard, voice, etc. For example, if a form control is activated with a mouse or other pointing device, someone who is using the page without sight, with voice input, or with a keyboard or who is using some other non-pointing input device will not be able to use the form. Providing text equivalents for image maps or images used as links makes it possible for users to interact with them without a pointing device.

Checkpoints

- Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape. [Priority 1]

If you use applets and scripts:

- ensure that any element that has its own interface can be operated in a device-independent manner. [Priority 2]
- for scripts, specify logical event handlers rather than device-dependent event handlers. [Priority 2]
- create a logical tab order through links, form controls, and objects. [Priority 3]
- provide keyboard shortcuts to important links (including those in client-side image maps), form controls, and groups of form controls. [Priority 3]

Guideline 10: Use interim solutions.

Use interim accessibility solutions so that assistive technologies and older browsers will operate correctly. For example, older browsers do not allow users to navigate to empty edit boxes. Older screen readers read lists of consecutive links as one

link. These active elements are therefore difficult or impossible to access. Also, changing the current window or popping up new windows can be very disorienting to users who cannot see that this has happened.

Note: Web Content Guidelines Working Group considers them to be valid and necessary to web accessibility. However, the Working Group does not expect these checkpoints to be necessary in the future, once web technologies have incorporated anticipated features or capabilities.

Checkpoints

- until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user. [Priority 2]

If forms are used:

- until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned. [Priority 2]
- until user agents (including assistive technologies) render adjacent links distinctly, include non-link, printable characters (surrounded by spaces) between adjacent links. [Priority 3]

If tables are used:

- until user agents (including assistive technologies) render side-by-side text correctly, provide a linear text alternative (on the current page or some other) for *all* tables that lay out text in parallel, word-wrapped columns.

If forms are used:

- until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas. [Priority 3]

Guideline 11: Use W3C technologies and guidelines.

Use W3C technologies (according to specification) and follow accessibility guidelines. Where it is not possible to use a W3C technology, or doing so results in material that does not transform gracefully, provide an alternative version of the content that is accessible.

The current guidelines recommend W3C technologies (e.g., HTML, CSS, etc.) for several reasons:

- W3C technologies include 'built-in' accessibility features.
- W3C specifications undergo early review to ensure that accessibility issues are considered during the design phase.
- W3C specifications are developed in an open, industry consensus process.

Many non-W3C formats (e.g., Shockwave, etc.) require viewing with either plug-ins or stand-alone applications. Often, these formats cannot be viewed or navigated with standard user agents. Avoiding non-W3C and non-standard features (proprietary elements, attributes, properties, and extensions) will tend to make pages more accessible to more people using a wider variety of hardware and software. When inaccessible technologies (proprietary or not) must be used, equivalent accessible pages must be provided.

Even when W3C technologies are used, they must be used in accordance with accessibility guidelines. When using new technologies, ensure that they transform gracefully.

Note. Converting documents (from PDF, PostScript, RTF, etc.) to W3C markup languages (HTML, XML) does not always create an accessible document. Therefore, validate each page for accessibility and usability after the conversion process. If a page does not readily convert, either revise the page until its original representation converts appropriately or provide an HTML or plain text version.

Checkpoints

If all else fails

- If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page. [Priority 1]
- Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported. [Priority 2]
- Avoid deprecated features of W3C technologies. [Priority 2]
- Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc.) [Priority 3]

Guideline 12: Provide context and orientation information.

Provide context and orientation information to help users understand complex pages or elements. Grouping elements and providing contextual information about the relationships between elements can be useful for all users. Complex relationships between parts of a page may be difficult for people with cognitive disabilities and people with visual disabilities to interpret.

Checkpoints

If frames are used

- Title each frame to facilitate frame identification and navigation. [Priority 1]
- Divide large blocks of information into more manageable groups where natural and appropriate. [Priority 2]
- Describe the purpose of frames and how frames relate to each other if it is not obvious by frame titles alone. [Priority 2]

If forms are used

- Associate labels explicitly with their controls. [Priority 2]

Guideline 13: Provide clear navigation mechanisms.

Provide clear and consistent navigation mechanisms – orientation information, navigation bars, a site map, etc. – to increase the likelihood that a person will find what they are looking for at a site.

Checkpoints

- Clearly identify the target of each link. [Priority 2]
- Provide metadata to add semantic information to pages and sites. [Priority 2]
- Provide information about the general layout of a site (e.g., a site map or table of contents). [Priority 2]
- Use navigation mechanisms in a consistent manner. [Priority 2]

- Provide navigation bars to highlight and give access to the navigation mechanism. [Priority 3]
- Group related links, identify the group (for user agents), and, until user agents do so, provide a way to bypass the group. [Priority 3]
- If search functions are provided, enable different types of searches for different skill levels and preferences. [Priority 3]
- Place distinguishing information at the beginning of headings, paragraphs, lists, etc. [Priority 3]
- Provide information about document collections (i.e., documents comprising multiple pages.). [Priority 3]
- Provide a means to skip over multi-line ASCII art. [Priority 3]

Guideline 14: Ensure that documents are clear and simple.

Ensure that documents are clear and simple so they may be more easily understood. Consistent page layout, recognizable graphics, and lucid language benefit all users. In particular, they help people with cognitive disabilities or who have difficulty reading.

Using clear and simple language promotes effective communication. Using clear and simple language also benefits people whose first language differs from the language web content is written, including those people who communicate primarily in sign language.

Checkpoints

- Use the clearest and simplest language appropriate for a site's content. [Priority 1]
- Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page. [Priority 3]
- Create a style of presentation that is consistent across pages. [Priority 3]

10.5 ELECTRONIC TEXT ENCODING AND INTERCHANGE

10.5.1 Origin

The Text Encoding Initiative (TEI), an international cooperative research effort, was sponsored and organized by three leading professional associations in the field: the Association for Computational Linguistics (ACL), the Association for Literary and Linguistic Computing (ALLC) and the Association for Computing and the Humanities (ACH) and funded primarily by the US National Endowment for the Humanities and by the European Union 3rd Framework Programme for Linguistic Research and Engineering. It also received with grants from the Mellon Foundation and from the Canadian Social Sciences and Humanities Research Council and the donation of time and expertise by the many members of the wider research community who have served on the TEI's Working Committees and Working Groups [Lou, 1995]. The goal of the project is to define a set of generic guidelines for the representation of textual materials in electronic form.

10.5.2 Objectives

There are many different options available to publish information today which can be in any form. However information in any package is valuable, irrespective of its physical form. As technology emerges which is genuinely adequate to the task of integrating text, graphics and audio into a seamless information-bearing

resource, so the importance of that integrated vision becomes more apparent. TEI scheme enhances the content by facilitating the integration of multimedia technology into electronic resources. This is done by providing a description of information that is independent of realization or media of the resource.

Though TEI is strongly interested in text, it is by no means confined to the use of electronic text as a stage in the production of paper documents. The TEI is equally concerned with both textual and non-textual resources in electronic form, whether as constituents of a research database or components of non-paper publications.

TEI project is basically concerned with two aspects:

- **what** textual features should be encoded (i.e. made explicit) in an electronic text;
- **how** that encoding should be represented for loss-free, platform-independent interchange.

Electronic resources are very heterogeneous in nature. Therefore the content is also in diverse material such as in newspapers, books, office memoranda, play scripts, publicity brochures, letters and diaries, transcribed lectures and interviews, TV and radio broadcasts, and unscripted conversations. Contents from other media can be integrated into a single body of material. Research needs impose that this integration be carried out with minimal loss of information, and at the same time with minimal complexity: in any case, the resulting 'text' is far removed from the conventional notion of a printed work. Electronic texts contain *markup* or encoding, which makes explicit various features of the text, so that they can be efficiently processed. Printed texts adopt a variety of similarly-motivated conventions (use of typeface, organisation of the carrier medium, etc), but these are not so readily processible as the tags of a formal markup scheme [Lou, 1995].

10.5.3 Features

The TEI guidelines mainly deal with any text in electronic form. They are meant for facilitating text processing. TEI explicitly represents the different features of text, which in turn makes it easy to manipulate the processing through computer programs. A set of tags or markers are specified, which may be inserted in the electronic representation of the text, in order to mark the text structure and other textual features of interest. Without such explicit markers, many important features remain difficult to locate by mechanical means such as computer programs, and thus difficult to process effectively. The process of inserting such explicit markers for implicit textual features is often called 'markup' or 'tagging', and the term *encoding scheme* or *markup language* denotes the rules, which govern the use of markup in a set of encodings.

The Guidelines formulated are intended for use in interchange between individuals and research groups using different programs and computer systems over a broad range of applications. Since they contain an inventory of the features most often found useful for text processing, the Guidelines also provide help to those creating texts in electronic form. They can also be used for the local storage of text that is to be processed with multiple software packages requiring different input formats.

The Guidelines apply to texts in any natural language, of any date, in any literary genre or text type, without restriction on form or content. They treat both continuous materials ('running text') and discontinuous materials such as dictionaries and linguistic corpora. Though principally directed to the needs of the

scholarly research community, the Guidelines are not restricted to esoteric academic applications. They should also be useful for librarians who maintain and document electronic materials, as well as for publishers and others creating or distributing electronic texts. Although they focus on problems of representing in electronic form texts that already exist in traditional media, these Guidelines should also be useful for the creation of electronic texts. They are adequate to, but not limited by, existing practices [www.nlc-bnc.ca/iso/tc46sc9].

Standardising text representation helps development of new applications while maintaining upward compatibility between the newer versions and the old ones. By standardising at the level of text representation, it is hoped to retain the flexibility needed to develop new applications, while ensuring that old ones continue to function. TEI *Guidelines* aim to provide not a set of normative rules for particular applications, but rather a modular and extensible framework, within which particular application-specific norms can be defined.

Self Check Exercise

4) State the basic objectives of Text encoding?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

10.6 DYNAMIC CONTENT

In addition to static web content, dynamic content is produced from databases when searched and accessed online. Dynamic content can be generated when page is requested at regular intervals and stored on temporary websites. For example, metadata attaches to web documents may be produced using database facility or other utilities and presented in the search results. Dynamic content causes problems for indexes (search engines) and also catching (by browser or proxy server) [http://www.upriss.org.uk]. However, dynamic content should follow the general norms and guidelines of content development. It calls for more meticulous methods in generating such content, information analysis, metadata design and design of content management system [Rockley, Ann].

Self Check Exercise

5) What is Dynamic Content and the challenges presented by it?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

10.7 SUMMARY

Content development aims at reaching the contents to its target audience which should be comprehended by the end users. To achieve successful and meaningful transfer of information there are basically two factors when electronic sources are considered. One is the technological compatibility which emphasizes on using the mainstream applications which are widely used and available. And other is regarding the content itself and the ideas contained within. While no particular guideline deals with a particular 'thought content', the guidelines described in this Unit will generally give direction to design of electronic documents, its structuring and presentation of information. The guidelines provided by ISO, W3C and TEI are described here.

10.8 ANSWERS TO SELF CHECK EXERCISES

- 1) It is important to follow norms and guidelines as standardization increases the accessibility of information to users. Moreover, the searches on Internet could be made more precise by adhering to standards.
- 2) While using the electronic documents often the user faces problems with forms and formats. Generally the concepts should be made clear in the content and there should be thought flow and continuity in ideas. When structuring, the documents should be well structured and presentation aspects such as using bold fonts, underlines and colour should be taken care of.
- 3) Adhering to standards ensure 'quality' in products. In the parlance of electronic and online resources standards help in providing enhanced usability which results in increased satisfaction and productivity and also provide assurance of the users well being in using the products. They provide consistency within and between systems and applications. The standards also help software customers in their selection of acceptable products. They protect the employer from complaints and possible litigation.
- 4) Text encoding deals with the structuring of web documents using markup tags. The guidelines for text encoding are concerned with both what textual features should be encoded (i.e. made explicit) in an electronic text, and how that encoding should be represented for loss-free, platform-independent, interchange.
- 5) Dynamic Content is the content generated dynamically on the web typically using information in databases. Dynamic content causes problems for indexes (search engines) and also caching (by browser or proxy server).

10.9 KEYWORDS

- | | |
|-------------------|--|
| Accessible | : Content is accessible when it may be used by someone with a disability. |
| Animation | : A technique in which each frame of a film or movie is produced individually, whether generated as a computer graphic, or by photographing a drawn image, or by repeatedly making small changes to a model unit (see claymation and stop motion), and then photographing the result with a special animation camera. When the frames are strung together and the resulting film is viewed at a speed of 16 or more frames per second, there is an illusion of continuous movement (due to the persistence of vision). |

Applet	: A program inserted into a web page.
Assistive Technologies	: Devices used by people with disabilities to access computers. Some assistive technologies include text-to-speech screen readers, alternative keyboards and mice, head pointing devices, voice recognition software, and screen magnification software.
Braille	: Braille uses six raised dots in different patterns to represent letters and numbers to be read by people who are blind with their fingertips.
Computer Animation	: An art of creating moving images via the use of computers. Increasingly it is created by means of 3D computer graphics, though 2D computer graphics are still widely used.
Content Developer	: Someone who authors web pages or designs websites.
Device Independent	: Users must be able to interact with a user agent (and the document it renders) using the supported input and output devices of their choice and according to their needs. Input devices may include pointing devices, keyboards, braille devices, head wands, microphones, and others. Output devices may include monitors, speech synthesizers, and braille devices.
Dynamic Content	: Dynamic Content is automatically or dynamically generated web pages typically produced using information in databases.
Element	: An element is a syntactic construct and more generally means a type of content (such as video or sound) or a logical construct (such as a header or list). An element that causes text characters to be part of the document is called a text element.
IEC	: International Electrotechnical Commission.
Image	: A graphical presentation.
Link Text	: The rendered text content of a link.
Natural Language	: Spoken, written, or signed human languages such as French, Japanese, American Sign Language, and braille. The natural language of content may be indicated with the 'lang' attribute in HTML and the "xml:lang" attribute in XML.
Navigation Bar	: A navigation bar is a collection of links to the most important parts of a document or site.
Navigation Mechanism	: A navigation mechanism is any means by which a user can navigate a page or site. Some typical mechanisms include: navigation bars, site maps and tables of contents.
Plug-Ins	: Software programs that enhance other programs or applications on your computer.

Site Map

There are plug-ins for Internet browsers, graphics programs, and other applications.

- : A site map provides a global view of the organization of a page or site.

Style Sheet

- : A style sheet is a set of statements that specify presentation of a document. Style sheets may have three different origins: they may be written by content providers, created by users, or built into user agents. In CSS, the interaction of content provider, user, and user agent style sheets is called the *cascade*.

Table of Contents

- : A table of contents generally lists (and links to) the most important sections of a document.

Tabular Information

- : When tables are used to represent logical relationships among data – text, numbers, images, etc., that information is called ‘tabular information’ and the tables are called ‘data tables’. The relationships expressed by a table may be rendered visually (usually on a two-dimensional grid), aurally (often preceding cells with header information), or in other formats.

User Agent

- : Software to access web content, including desktop graphical browsers, text browsers, voice browsers, mobile phones, multimedia players, plug-ins, and some software assistive technologies used in conjunction with browsers such as screen readers, screen magnifiers, and voice recognition software.

10.10 REFERENCES AND FURTHER READING

Challenges: dynamic content. <<http://www.upriss.org.uk/db/433121.1.html>>.

Devika, P.M. (2003). *Norms and guidelines for content development*. In: PGDLAN Course material: MLI-006, Unit 7. New Delhi: Indira Gandhi National Open University.

Goldfarb, Charles. (1992). *SGML handbook*. San Jose: IBM Almaden Research Center.

Hackos, Jo Ann. (2002). *Content management for dynamic delivery*. New York: John Wiley & Sons.

IBM. *Ease of use ISO IEC standards*. <http://www-3.ibm.com/ibm/easy/eou_ext.nsf/Publish/583> [accessed on 13/12/2001].

ISO standard for information and documentation- identification and description. <<http://www.nlc-bnc.ca/iso/tc46sc9/>> [accessed on 21/12/2001].

ISO standards search page. <http://www.iso.org/iso/en/standards_Search.StandardsQueryForm> [accessed on 17/01/2002].

Lou, Burnard. (1995). *An Introduction to the Text Encoding Initiative*. Oxford University Computing Services. <<http://www.hcu.ox.ac.uk/TEI/Papers/J31/WHAT.htm>> [accessed on 16/02/2002].

Neilson, Jacob and Fox, Jonathan. (1998). *Writing for the web*. <<http://www.sun.com/980713/webwriting>> [accessed on 26/04/2004].

Neilson, Jacob. (2000). *Content creation for average people*. <<http://www.useit.com/alertbox/20001001.html>> [accessed on 12/01/2002].

Neilson, Jacob. (2001). *Poor code quality contaminates users' conceptual models*. <<http://www.useit.com/alertbox/20011028.html>> [accessed on 14/12/2001].

Rockley, Ann. (2002). *Dynamic content management*. <<http://www.stc.org/confproceed/2002/PDFs/STC49-00015.pdf>>.

Sperberg-McQueen, C.M. and Lou, Burnard. (1994). *Guidelines for electronic text encoding and interchange*. Chicago, Oxford: Text Encoding Initiative.

Tufted, Edward R. (1990). *Envisioning information*. Graphics Press.

W3C. (1999). *Web Content Accessibility Guidelines 1.0. W3C Recommendation*. <<http://www.w3.org/TR/WAI-WEBCONTENT/>>. [accessed on 17/01/2002].

Wood, Mick. *Page design & content guidelines*. <<http://www.mickwood.com/articles/content.shtm>>.

Workman, Richard Saul. (1996). *Information architects*. Graphics Press.



UNIT 11 INTRODUCTION TO HTML AND XML

Structure

- 11.0 Objectives
- 11.1 Introduction
- 11.2 World Wide Web and Markup Languages
- 11.3 Standard Generalized Markup Language (SGML)
- 11.4 HyperText Markup Language (HTML)
 - 11.4.1 Introduction to HTML
 - 11.4.2 Features of HTML
 - 11.4.3 Editor for HTML
 - 11.4.4 Syntax of HTML Commands
 - 11.4.5 Framework of a Web Page
- 11.5 Basic HTML Tags
 - 11.5.1 Linking
 - 11.5.2 URLs
- 11.6 HTML and the Browser
- 11.7 eXtensible Markup Language (XML)
 - 11.7.1 Need for XML
 - 11.7.2 Objectives of XML
 - 11.7.3 Features of XML
 - 11.7.4 How XML is Different from HTML
 - 11.7.5 Advantages of XML
- 11.8 XML Syntax and Semantic Tags
 - 11.8.1 XML Syntax
 - 11.8.2 Semantic Tags of XML
- 11.9 Document Type Definition (DTD)
- 11.10 Implications of XML in Library and Information Activities
- 11.11 Summary
- 11.12 Answers to Self Check Exercises
- 11.13 Keywords
- 11.14 References and Further Reading

11.0 OBJECTIVES

In the previous Unit, you have been acquainted with the guidelines, norms and standards developed by various organisations/ suggested by different experts and organisations for development of content on the Web. The Internet has changed the way the information can be organised therein. It is necessary to know the way or form in which the information can be organised on the net which we will be studying in this Unit.

After reading this Unit, you will be able to:

- understand what is meant by World Wide Web;
- understand the concept and functions of markup language;

- learn structure, tags and syntax of HyperText Markup Language (HTML) and eXtensible Markup Language (XML); and
- know applications of these in creation of web page.

11.1 INTRODUCTION

Today Internet has changed the way of life in all fields. It has created an instant online connection and communication world over. Due to its feasible technology, Internet has grown rapidly in the past few years gained so much popularity. It has been transformed from just a text-based environment to a click-able and link-able world. What has made this possible is the World Wide Web (WWW). Internet today has become a multimedia communication channel where data can be transferred in all the formats.

The use of hypermedia and hypertext is so much ingrained on the Internet that WWW cannot be thought without multimedia and WWW has become a synonym for the term Internet. Hypertext markup language is a language to render the information over Internet. It can accommodate audio, video, text and image. It can be said that basic feature of today's Internet is hypertext and hyper-linking.

11.2 WORLD WIDE WEB AND MARKUP LANGUAGES

The World Wide Web (WWW) is designed for the display, distribution and searching of information, files, and data across multiple locations over the Internet. WWW is used to view the web documents and these web documents are written in a particular language supported by WWW. However, it is not a programming language. Hypertext Markup Language is used to represent the web content on web pages [Slack Inc, 2001].

The term markup means instructions for printing in a particular style, just like, while proofreading editors mark the text (e.g. underlined) to display it in bold while printing. Similarly to display the electronic text in web page on browsers, embedded instructions are given within the text to make the parser understand how text should appear on display [Sol, 1999].

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 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: inder@ignou.ac.in

From: rakesh@ignou.ac.in

Date: Tue, 26 Jan 2005 01:00:58 -0800

Subject: Memo

Kindly inform me the timetable of the term end examination of the MLIS course.

Regards

Rakesh

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 email once the fields are known. This is 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, what markup is required, how markup is to be distinguished from text, and what the markup means. Standard Generalised Markup Language (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 differs 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.

For understanding of all markup languages, when described in SGML terms, knowledge of three basic concepts are fundamental. These are the notions of a markup *entity*, a markup *element* with its associated *attributes*, and a *document type*. As you know at the most basic 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 – for example, they may be included within each other, linked to each other by reference or simply presented sequentially. This level of description ensures 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 Exercises

- 1) What is meant by WWW?
- 2) Distinguish between Hypertext, Hyperlink and Hyper media.
- 3) Define a markup.

Note: i) Write your answers in the space given below.
ii) Check your answers with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

.....

.....

11.3 STANDARD GENERALIZED MARKUP LANGUAGE (SGML)

SGML aims 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. It is not meant for formatting of text. Basically it was meant to preserve the structure of document.

SGML is not a kind of text formatting system (although its origins can be readily traced in the world of electronic text formatting), or is 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 and 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*. Using it becomes easier if one understands the basic principles behind it, and take its limitations into account [Gorman, Dianne].

11.4 HYPERTEXT MARKUP LANGUAGE (HTML)

Hypertext Markup Language (HTML) is a structured markup language that is used to create web pages. A markup language such as HTML is simply a collection of codes called elements that are used to indicate the structure and format of a document. Elements in HTML consist of alphanumeric tokens within angular brackets, such as `<b>`, `<html>`, `<body>`, etc.

Most elements consist of paired tags: a *start tag* and an *end tag*. For example, `<b>` is a start tag and `</b>` is the end tag. The end tag is similar to start tag, except that the symbol is preceded by forward slash. An element's instruction applies to whatever content is contained between its start and end tags:

E.g. `<b>` This text is bold; `</b>` but this text is not.

Element names are not case-sensitive. An element such as `<hTml>` is equivalent to `<html>`. However using either upper or lower case consistently makes HTML documents easier to understand and maintain. Element names cannot contain spaces.

11.4.1 Introduction to HTML

HTML is the language with which Web pages are 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.

11.4.2 Features of HTML

HTML is a content-based or structured 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.

There is some misconception 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 can be seen as both a structural language as well as page layout language. For instance, the tag <H1>, i.e., heading tag is basically a structural tag which says the text embedded is 'Heading of first order'. But similarly HTML has , i.e., bold, <I>, i.e., italics, etc., are formatting or page layout tags.

Some of the very Basic HTML concepts, tags and features are described below.

11.4.3 Editor for HTML

HTML is a plain text file and needs a simple text editor to create the tags. However, it is important that 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.

11.4.4 Syntax of HTML Commands

In general, all HTML commands will have the form:

<COMMAND> text </COMMAND>.

Two points need to be noted here: (1) all commands MUST be enclosed within angular brackets < >; and, (2) all commands are used in pairs wherein the <COMMAND> marks the beginning and </COMMAND> marks the end.

11.4.5 Framework of a Web Page

The framework of a web page is :

<HTML>

<HEAD>

<TITLE> Title of Your Page </TITLE>

</HEAD>

<BODY>

The Body of Your Page

</BODY>

</HTML>

Explanation

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.

Some of the basic HTML tags that are used in developing HTML documents are as follows:

Markup Tags

HTML

This element tells the browser that the file contains HTML-coded information. The file extension .html and .htm also indicates that this is a HTML document.

Head

The head element identifies the first part of your HTML-coded document that contains the title. The title is shown in the title bar of browser's window.

Title

The title element contains your document title and identifies its content in a global context. The title is typically displayed in the title bar at the top of the browser window, but not inside the window itself. The title is also what is displayed on someone's hotlist or bookmark list. So it is better choose something descriptive, unique, and relatively short. A title is also used to identify your page for search engines (such as HotBot or Infoseek). Generally it is advisable to keep titles to 64 characters or fewer.

Body

The second—and largest—part of a HTML document is the body, which contains the document content (displayed within the text area of the browser window). The tags explained below are used within the body of your HTML document.

Headings

HTML has six levels of headings, numbered 1 through 6, with 1 being the largest. Headings are typically displayed in larger and/or bolder fonts than normal body text. The first heading in each document should be tagged <H1>.

The syntax of the heading element is:

<H_y>Text of heading </H_y>

where y is a number between 1 and 6 specifying the level of the heading.

Generally, it is advised not to skip levels of headings in a HTML document. For example, do not start with a level-one heading (<H1>) and then next use a level-three (<H3>) heading.

For example:

```
<html>
```

```
<head>
```

```
<title>IGNOU Homepage</title>
```

```
</head>
```

```
<body> Welcome to the Home Page of IGNOU. IGNOU is one of the open universities  
of India providing distance education courses in different fields.
```

```
</body>
```

```
</html>
```

Paragraphs

Unlike documents in most word processors, carriage returns in HTML files are not significant. In fact, any amount of white space – including spaces, linefeeds, and carriage returns – are automatically compressed into a single space when a HTML document is displayed in a browser. Word wrapping can occur at any point in the source file without affecting how the page will be displayed.

```
<P>Welcome to the world of HTML.
```

```
This is the first paragraph.
```

```
while short it is
```

```
still a paragraph!</P>
```

In the source file there is a line break between the sentences. A Web browser ignores this line break and starts a new paragraph only when it encounters another `<P>` tag.

Important: You must indicate paragraphs with `<P>` elements. A browser ignores any indentations or blank lines in the source text. Without `<P>` elements, the document becomes one large paragraph. (One exception is text tagged as 'preformatted,' which is explained below.) For example, the following would produce identical output as the first example:

```
<H1>Level-one heading</H1>
```

```
<P>Welcome to the world of HTML. This is the  
first paragraph. While short it is still a
```

```
paragraph! </P> <P>And this is the second paragraph.</P>
```

NOTE: The `</P>` closing tag may be omitted. This is because browser understands that when it encounters a `<P>` tag, it means that the previous paragraph has ended. However, since HTML now allows certain attributes to be assigned to the `<P>` tag, it is generally a good idea to include it.

Using the `<P>` and `</P>` as a paragraph container means that you can center a paragraph by including the `ALIGN=alignment` attribute in your source file.

```
<P ALIGN=CENTER>
```

```
This is a centered paragraph.
```

```
</P>
```

This is a centered paragraph.

It is also possible to align a paragraph to the right instead, by including the `ALIGN=RIGHT` attribute. `ALIGN=LEFT` is the default alignment; if no `ALIGN` attribute is included, the paragraph will be left-aligned.

Lists

HTML supports unnumbered, numbered, and definition lists. Nested lists can also be used, but use this feature sparingly because too many nested items can become difficult to follow.

Unnumbered Lists

To make an unnumbered, bulleted list, start with an opening list `<UL>` (for unnumbered list) tag

enter the (list item) tag followed by the individual item; no closing tag is needed to end the entire list with a closing list tag

Below is a sample three-item list:

```
<UL>
<LI> apples
<LI> bananas
<LI> grapefruit
</UL>
```

The output is:

```
apples
bananas
grapefruit
```

The items can contain multiple paragraphs. Indicate the paragraphs with the <P> paragraph tags.

Numbered Lists

A numbered list (also called an ordered list, from which the tag name derives) is identical to an unnumbered list, except it uses instead of . The items are tagged using the same tag. The following HTML code:

```
<OL>
<LI> oranges
<LI> peaches
<LI> grapes
</OL>
```

produces this formatted output:

```
oranges
peaches
grapes
```

Definition Lists

A definition list (coded as <DL>) usually consists of alternating a definition term (coded as <DT>) and a definition description (coded as <DD>). Web browsers generally format the definition on a new line and indent it.

The following is an example of a definition list:

```
<DL>
<DT> IGNOU
<DD> IGNOU, Indira Gandhi National Open University is located in New Delhi.
<DT> IISc
<DD> IISc, the Indian Institute of Science is located in Bangalore
</DL>
```

The output looks like:

IGNOU

IGNOU, Indira Gandhi National Open University is located in New Delhi.

IISc

IISc, the Indian Institute of Science is located in Bangalore.

The <DT> and <DD> entries can contain multiple paragraphs (indicated by <P> paragraph tags), lists, or other definition information.

Nested Lists

Lists can be nested. You can also have a number of paragraphs, each containing a nested list, in a single list item.

Here is a sample nested list:

```
<UL>
  <LI> A few fruits:
    <UL>
      <LI> Apple
      <LI> Grapes
      <LI> Banana
    </UL>
  <LI> Two citrus fruits:
    <UL>
      <LI> Lime
      <LI> Orange
    </UL>
  </UL>
```

The nested list is displayed as:

- A few fruits:
 - Apple
 - Grapes
 - Banana
- Two citrus fruits:
 - Lime
 - Orange

Self Check Exercise

4) What is HTML? Mention some basic HTML tags.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

11.5.1 Linking

The chief ability of HTML comes from its ability to link text and/or an image to another document or section of a document thus weaving 'a web' of resources.

A browser highlights the identified text or image with colour and/or underlines to indicate that it is a *hypertext link* (often shortened to *hyperlink* or just *link*).

HTML's single hypertext-related tag is `<A>`, which stands for *anchor*. To include an anchor in your document:

- a) start the anchor with `<A` (include a space after the `A`)
- b) specify the document you're linking to by entering the parameter `HREF="filename"` followed by a closing right angle bracket (`>`)
- c) enter the text that will serve as the hypertext link in the current document
- d) enter the ending anchor tag: `</A>` (no space is needed before the end anchor tag)

Here is a sample hypertext reference in a file called `US.html`:

```
<A HREF = "hello.html">Hello</A>
```

This entry makes the word 'Hello' the hyperlink to the document `hello.html`, which is in the same directory as the first document.

Relative Pathnames Versus Absolute Pathnames

You can link to documents in other directories by specifying the *relative path* from the current document to the linked document. For example, a link to a file `assam.html` located in the subdirectory *temp* would be:

```
<A HREF = "temp/assam.html">Content </A>
```

These are called relative links because you are specifying the path to the linked file relative to the location of the current file. You can also use the absolute pathname (the complete URL) of the file, but relative links are more efficient in accessing a server.

They also have the advantage of making your documents more 'portable' — for instance, you can create several web pages in a single folder on your local computer, using relative links to hyperlink one page to another, and then upload the entire folder of web pages to your web server. The pages on the server will then link to other pages on the server, and the copies on your hard drive will still point to the other pages stored there.

It is important to point out that UNIX is a case-sensitive operating system where filenames are concerned, while DOS and the MacOS are not. For instance, on a Macintosh, 'DOCUMENT.HTML', 'Document.HTML', and 'document.html' are all the same name. If you make a relative hyperlink to 'DOCUMENT.HTML', and the file is actually named 'document.html', the link will still be valid. But if you upload all your pages to a UNIX web server, the link will no longer work. *Be sure to check your filenames before uploading.*

Pathnames use the standard UNIX syntax. The UNIX syntax for the parent directory (the directory that contains the current directory) is `".."`

If you were in the `assam.html` file and were referring to the original document `INDIA.html`, your link would look like this:

```
<A HREF = "../INDIA.html">India</A>
```

In general, you should use relative links whenever possible because:

- a) it is easier to move a group of documents to another location (because the relative path names will still be valid)
- b) it is more efficient connecting to the server
- c) there is less to type

However, use absolute pathnames when linking to documents that are not directly related. For example, consider a group of documents that comprise a user manual. Links within this group should be relative links. Links to other documents (perhaps a reference to related software) should use absolute pathnames instead. This way if you move the user manual to a different directory, none of the links would have to be updated.

11.5.2 URLs

The World Wide Web uses Uniform Resource Locators (URLs) to specify the location of files on other servers. A URL includes the type of resource being accessed (e.g., Web, gopher, FTP), the address of the server, and the location of the file. The syntax is:

scheme://host.domain [:port]/path/ filename

where scheme is one of the following:

file

a file on your local system

ftp

a file on an anonymous FTP server

http

a file on a World Wide Web server

gopher

a file on a Gopher server

WAIS

a file on a WAIS server

news

a Usenet newsgroup

telnet

a connection to a Telnet-based service

Links to Specific Sections

Anchors can also be used to move a reader to a particular section in a document (either the same or a different document) rather than to the top, which is the default. This type of an anchor is commonly called a *named anchor* because to create the links, you insert HTML names within the document.

You can also link to a specific section in another document. That information is presented first because understanding that helps you understand linking within the same document.

Links between Sections of Different Documents

Suppose you want to set a link from document A (documentA.html) to a specific section in another document (delhi.html). Enter the HTML coding for a link to

a named anchor:

documentA.html:

In addition to the many institute, Delhi is also home to

```
<a href = "delhi.html#JNU">Jawaharlal Nehru University</a>.
```

The characters after the hash (#) mark is using for titling within the delhi.html file. It tells the browser what should be displayed at the top of the window when the link is activated. In other words, the first line in the browser window should be the Jawaharlal Nehru University heading.

Next, to create the named anchor (in this example "JNU") in delhi.html:

```
<H2><A NAME = "JNU"> Jawaharlal Nehru University </a></H2>
```

With both of these elements in place, a reader can go directly to the JNU reference in delhi.html.

NOTE: links cannot be made to specific sections within a different document unless either there is write permission to the coded source of that document or that document already contains in-document named anchors.

Links to Specific Sections within the Current Document

The technique is the same except the filename is *omitted*.

For example, to link to the JNU anchor from within delhi.html, enter:

...More information about

```
<A HREF="#JNU"> Jawaharlal Nehru University </a>
```

is available elsewhere in this document.

Be sure to include the tag at the place in your document where you want the link to jump to (Jawaharlal Nehru University).

Named anchors are particularly useful when you think readers will print a document in its entirety or when you have a lot of short information you want to place online in one file.

Mailto Attribute

You can make it easy for a reader to send electronic mail to a specific person or mail alias by including the mailto attribute in a hyperlink. The format is:

```
<A HREF="mailto:emailinfo@host">Name</a>
```

For example, enter:

```
<A HREF="mailto:pubs@jnu.edu">JNU Publications</a>
```

to create a mail window that is already configured to open a mail window for the JNU Publications Group alias.

5) Mention the different types of links that can be created in a HTML document.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

11.6 HTML AND THE BROWSER

What is typed as HTML tags can be viewed only through a browser to see its actual web display. 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 [Gorman, Dianne]. The popular web browsers available now a days are: Internet Explorer from Microsoft Corporation, Netscape Navigator from Netscape Communication, Firefox from Mozilla Foundation, Opera from Opera Software, and so on. All these web browsers support HTML, SGML and XML tags and elements to display web documents properly and to extract documents' description.

11.7 EXTENSIBLE MARKUP LANGUAGE (XML)

According to the abstract from the XML Specification version 1.0 [World Wide Web Consortium, 2005]:

"The eXtensible Markup Language (XML) is a subset of SGML that is completely described in this document (i.e. XML version 1.0 specification). 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 eXtensible MarkUp Language.
- 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 uses a DTD (**Document Type Definition**) to describe the data.
- XML with a DTD is designed to be **self-descriptive**.

11.7.1 Need for XML

The idea of markup was to format a particular kind of document. The markup languages that carry the instruction for text processing are known as *Procedural markup*. But later on, it was felt that for system-to-system information interchange, markup languages could be used. This was first realized by Charles Goldfarb, Ed Mosher and Ray Lorie when they were working with legal documents. They designed first markup language known as GML (Generalized Markup Language) based on the following observations:

- The document processing programs needed to support a common document format;
- The common format needed to be specific to their domain-for example legal documents; and
- To achieve a high a degree of reliability, the document format would have to follow specific rules.

For example, take an example of memorandum,

From: Akkamahadevi

To: Suchitra Pattanayak

CC: Prasenjit Kar

Date: 27.01.2002

Subject: Appointment order

We are extremely happy to inform you that you are selected as the coordinator of Knowledge management team.

If we look into this document we find that there are six fields in this document.

- Who sent the document (the From: field)
- Who the document is intended for (the To: field)
- Who has been sent a copy of document (the CC: field)
- The date of document written (the Date: field)
- The subject of document (the Subject: field)
- The document body

So, if we make a fixed structure of this document then whoever writes the document has to follow same structure. Thus, for porting information from one system to another it will not be a problem as the structure of document is well defined. The definition of the structure of document is known as DTD (Document Type Definition).

Glodfarb further fine-tuned GML and proposed the SGML (Standardized General Markup Language) which was further approved by ISO (International Organization for Standardization) in 1986. This language was not a language itself but it was a meta language to develop other markup languages. HTML (HyperText Markup Language) is a derivative of SGML. HTML acts more like a formatting language so, it is always difficult to pull out what kind of data is stored inside a HTML document. Once this difficulty was understood, for information interchange the need for domain specific tags was felt. Development of such tags was not possible with HTML. Hence, XML was developed. It is always said that XML is more near to SGML than HTML.

11.7.2 Objectives of XML

The specification for XML has been developed with the following objectives.

- XML shall be straightforwardly usable over the Internet.
- XML shall be compatible with SGML.
- It shall be easy to write programs which process XML.
- The processors could 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.

11.7.3 Features of XML

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

11.7.4 How XML is Different from HTML

- i) XML was designed to attach semantic to data.
HTML has nothing to do with semantics of data. It only defines how the page should be presented (like, font, colour 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.

11.7.5 Advantages of XML

XML does not DO everything

XML is created as a way to structure, store and send information. XML is not designed to DO everything.

```
<?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.

Customised 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 attaching semantics to the data, data can be exchanged between incompatible systems. In the real world, the data stored in computer systems and databases, usually are in incompatible formats. One of the most time-consuming challenges for developers has been to exchange data among such systems over the Internet. Converting the data to XML greatly reduces complexity, since 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. This 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, a user's data is available to more users. Since XML is independent of hardware, software and application, a user can make his/her data available to more than only standard HTML browsers.

Other clients and applications can access one's XML files as data sources, like they are accessing databases. His/her data can be made available to all kinds of 'reading machines'.

XML can be used to create new languages

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

Self Check Exercise

6) Why XML is needed over HTML?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

11.8.1 XML Syntax

Let us consider the first line of the example at 11.7.5,

```
<?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>
```

a) 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 previously-mentioned example the `<book>` is the root element all the other tags are child to it.

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

b) 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 relationship. 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.

c) Element Naming

XML elements must follow the following 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 does not support them.
- The “:” should not be used in element names because it is reserved for *namespaces*.

d) XML Attributes

Attributes are used to provide additional information about elements. In HTML we often use attribute to get extra effect while formatting. For example,

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

will show the “Hello World” text in 12 font size and red coloured. The size and colour 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 further extends 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.

11.8.2 Semantic Tags of XML

XML was designed to attach semantics to data, i.e., adding context to the data. It does so by allowing to define one's own tags. For example,

```

<?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. These tags provide context to the whole structure of the document, hence these are known as semantic tags.

7) What are semantic tags?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

11.9 DOCUMENT TYPE DEFINITION (DTD)

It is possible to define your own structure of XML document and give others to write the XML document against your own schema to avoid the mistakes. A schema is nothing but the logical structure of document. This schema is called as 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. You may refer to Unit 12 for further discussion on DTD.

11.10 IMPLICATIONS OF XML IN LIBRARY AND INFORMATION ACTIVITIES

In the context of library and information activities, the most important implication is that XML can be used as a common platform for information exchange provided every one agrees to a common set of tags. As we know that many variant versions of MARCs and all 'standard MARCs' have created a kind of non-standardisation. In such a condition XML can be very much useful.

XML can also be used in Digital libraries. It can be used for document surrogate as a catalogue. On the web a great amount bibliographic data exchange is in XML.

With XML one can define the tags. These tags have the semantic value such as – '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 <<http://newsml.org>> 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.

Searching is another area where XML may be of great help. As it provides context to search term, searching becomes efficient. 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 such initiative in this direction.

Finally, it is sometimes felt that formatted display is a tedious job in XML. 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 interface tools to write formatted XML document may be available in the near future.

Self Check Exercise

8) Describe the library applications of XML.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

11.11 SUMMARY

In this Unit the concepts of WWW, Hypertext, Hypermedia and Markup Language are discussed, which are foundation of Internet. XML is another derivative of SGML, which is also used to render the information on the web. It is necessary one should know at least basic HTML tags to put the information on Internet. 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. 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.

11.12 ANSWERS TO SELF CHECK EXERCISES

- 1) World Wide Web (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. A 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) 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 as hyperlink and this can create a complex virtual web of connections.

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

- 3) 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 whereabouts, what markup is required, how markup is to be distinguished from text, and what the markup means.

- 4) HTML is a content-based structured markup language where the codes describe what the contents are. Some the basic tags of HTML are Head, Title, Headings and Body.
- 5) Different types links that can be created are: (i) linking of documents in other directories or websites, (ii) linking to specific sections of documents, (iii) linking between sections of different documents, (iv) linking to specific sections of current documents, etc.
- 6) 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 it is very difficult with HTML
 - XML is also considered as meta-language. Thus, XML can be used to create new languages
- 7) XML was designed to attach semantic to data, i.e., adding context to the data. It does so by allowing to define one's own tags. For example,

```
<?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. These tags provide context to the whole structure of the document, hence these are known as semantic tags.

- 8) XML can have implications in library environment. The first and foremost use of XML can be sought in information exchange. As 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 common platform for information exchange provided at least 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), on the web a great amount of bibliographic data exchange takes place using XML.

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.

11.13 KEYWORDS

Assistive Technologies	: Devices used by people with disabilities to access computers. Some assistive technologies include text-to-speech screen readers, alternative keyboards and mice, head pointing devices, voice recognition software, and screen magnification software.
Attribute	: A setting for a tag, that affects the way the tag is displayed.
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 <i>www.dell.com</i> or <i>www.indiatimes.com</i> .
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.

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 <i>hotlink</i> or <i>hyperlink</i> .
Multimedia	: The combination of several different communications 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 ' <i>quick brown fox</i> ' 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.
Plug-ins	: Software programs that enhance other programs or applications on your computer. There are plugins for Internet browsers, graphics programs, and other applications.
Server	: A program running on an Internet site that makes the web pages at that site available to browsers throughout the Internet.
Site	: Internet website.
Tags	: Tags are metadata which embeds the information in it.
Unicode	: The universal character encoding, maintained by the Unicode Consortium. This encoding standard provides the basis for processing, storage and interchange of text data in any language in all modern software and ICT protocols. It uses two bytes or 16 bits to code each character.
URI	: Uniform Resource Identifier - URIs have been known by many names: WWW addresses, Universal Document Identifiers, Universal Resource Identifiers, and finally the combination

of Uniform Resource Locators (URL) and Names (URN). As far as HTTP is concerned, Uniform Resource Identifiers are simply formatted strings that identify - via name, location, or any other characteristic - a resource.

W3C

- : An international industry consortium which develops common protocols that promote WWW evolution and ensure its interoperability. W3C develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential as a forum for information, commerce, communication, and collective understanding.

11.14 REFERENCES AND FURTHER READING

Blue Book (1988). Volume VIII - Fascicle VIII.8, *Data communication networks directory, recommendations X.500-X.521*, CCITT.

Devika, P.M. (2003). *Introduction to XML and HTML*. In: PGDLAN Course material, MLI-006, Unit 8. New Delhi: Indira Gandhi National Open University.

Gorman, Diane. *Introduction to HTML: understanding HTML*. <<http://www.awpa.asn.au/html>>.

Gorman, Dianne. *SGML and HTML: a guide to resources*. <<http://www.awpa.asn.au/sgml>>.

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

Hu, James H. *A beginner's guide to URLs*. <<http://www.selu.edu/Academics/Depts/Cmps/jhu/urlprimer.htm>>.

Hughes, Kevin (1994). *What is hypertext and hypermedia?* <<http://www.maths.tcd.ie/local/JUNK/guide/guide.02.html>>.

Kantor, B., and P. Lapsley (1986). *Network News Transfer Protocol: a proposed standard for the stream-based transmission of news, RFC 977*. UC San Diego & UC Berkeley.

Lang, R., and Wright, R. (1992). *RFC 1292 - a catalog of available X.500 implementations*. <<http://www.faqs.org/rfcs/rfc1292.html>>.

Lewis, Chris. (2004). *What is a markup language?* <<http://www.faqs.org/faqs/text-faq/section-4.html>>.

National Center for Supercomputing Applications. (2000). *Welcome to SGML on the web*. <<http://www.uv.es/~fores/programa/SGML.html>>.

Schwartz, M., and Tsirigotis, P. (1991). Experience with a semantically cognizant Internet white pages directory tool. *Journal of Internetworking Research and Experience*, 1(2), 23-50. <<http://www.codeontheroad.com/papers/Early.Netfind.pdf>>.

Slack Incorporated. (2001). *What is the World Wide Web?* <<http://www.centerspan.org/tutorial/www.htm>>.

Sol, Selena. (1999). *What is a markup language?* <http://www.wdvl.com/Authoring/Languages/XML/Tutorials/Intro/what_is_markup_language.html>.

Weider, C., and Reynolds, J. (1992). *RFC 1308 - executive introduction to directory services using the X.500 Protocol*. <<http://www.faqs.org/rfcs/rfc1308.html>>.

Weider, C., Reynolds, J., and Heker, S. (1992). *RFC 1309 - technical overview of directory services using the X.500 Protocol*. <<http://www.faqs.org/rfcs/rfc1309.html>>.

Williamson, S. (1993). *RFC 1400 - transition and modernization of the internet registration service*. <<http://www.faqs.org/rfcs/rfc1400.html>>.

World Wide Web Consortium. (2001). *About the World Wide Web*. <<http://www.w3.org/WWW>>.

World Wide Web Consortium. (2005). *Extensible Markup Language (XML)*. <<http://www.w3.org/XML>>.



UNIT 12 WEB BASED CONTENT DEVELOPMENT

Structure

- 12.0 Objectives
- 12.1 Introduction
- 12.2 What can be done with World Wide Web?
 - 12.2.1 Uniform Resource Locator
- 12.3 Hypertext, Hyperlink and Hypermedia
- 12.4 Hypertext Markup Language (HTML)
- 12.5 Introduction to Dynamic HTML
 - 12.5.1 Scripting Languages
 - 12.5.2 Document Object Model (DOM)
 - 12.5.3 Layers
 - 12.5.4 Difference Between HTML and DHTML
- 12.6 Web Interface to Database Linking
 - 12.6.1 Server Pages
- 12.7 Introduction to XML
- 12.8 XML Document Design
 - 12.8.1 Document Type Definition (DTD)
- 12.9 Multimedia Web Resources
- 12.10 Web Servers
 - 12.10.1 Website Hosting
- 12.11 Tools for Web Page Designing
 - 12.11.1 Dream Weaver
- 12.12 Summary
- 12.13 Answers to Self Check Exercises
- 12.14 Keywords
- 12.15 References and Further Reading

12.0 OBJECTIVES

In the previous Unit, you have learnt about markup languages used for web page development. The objective of this Unit is to give overview of technologies used in content development on the Web. After studying this Unit, you will be able to:

- be conversant with different technologies used in content development;
- understand the importance of hyperlinking of multimedia resources on the Web;
- understand the linking of databases can be done through Web interface; and
- know how websites are hosted on the Web.

12.1 INTRODUCTION

World Wide Web (WWW) is a standard designed for the access of multimedia information over Internet. WWW is a set of protocols which consists of several protocols like File Transfer Protocol (FTP), Telnet (Remote access), Hypertext

Transfer Protocol (HTTP) and so on. The Internet browsers support these protocols.

WWW is used to view the web documents. The web documents are written in a particular language supported by WWW that is, Hypertext MarkUp Language [Cluts, 2005].

In order to host documents on the web, one must have:

- Host Server
- WWW Client
- Internet Connectivity

Self Check Exercise

1) What is meant by World Wide Web?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.2 WHAT CAN BE DONE WITH WORLD WIDE WEB?

World Wide Web (WWW) can be used to search and retrieve documents, files, pictures, sounds, movies and more on remote computers all over the Internet. Just about every major company has pages on the web containing useful information about their goods and services. There are WWW based shopping malls, banks, libraries, record stores, and even distance learning schools.

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

12.2.1 Uniform Resource Locator

Uniform Resource Locator (URL) is the address of 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* 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 displayed [Hughes and Birznieks].

The 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)?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.3 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, that it contains 'hyper links'.

As per the Webster's definition, the term "*Hypertext* is basically same as regular text, but contains connections within the text to other documents". The Figure 1 shows how it works. 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 text that links the documents like this are called Hypertext and the linking is called Hyper-linking [Sun Chillisoft].

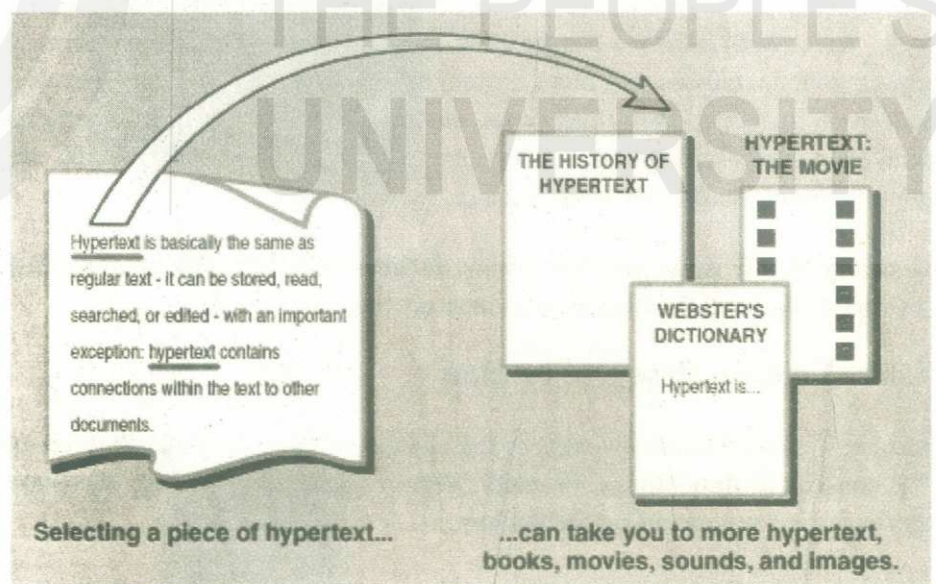


Fig. 12.1: How hypertext works

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 Merriam Webster's website (<http://www.m-w.com>) he/she get the meaning of the text besides one can listen the pronunciation of the word if he/she has sound card installed on machine. That means that when other media besides text are also used, it is known as Hypermedia.

12.4 HYPERTEXT MARKUP LANGUAGE (HTML)

Hypertext Markup Language (HTML) is a structured markup language that is used to create Web pages. A markup language such as HTML is simply a collection of codes called elements that are used to indicate the structure and format of a document. Elements in HTML consist of alphanumeric tokens within angular brackets, such as `<b>`, `<html>`, `<body>`, etc.

Most elements consist of paired tags: a start tag and an end tag. For example, `<b>` is a start tag and `</b>` is the end tag. The end tag is similar to start tag, except that the symbol is preceded by forward slash. An element's instruction applies to whatever content is contained between its start and end tags:

Eg. `<b>` This text is bold `</b>` but this text is not.

Element names are not case-sensitive. An element such as `<hTml>` is equivalent to `<html>`. However using either upper or lower case consistently makes HTML documents easier to understand and maintain. Element names cannot contain spaces.

You have already known about HTML which is described in Unit 11 of this Course.

12.5 INTRODUCTION TO DYNAMIC HTML

Dynamic HTML (DHTML) is a technology that allows a web page to change after it is loaded into the browser – there need not to be any communication with the web server for an update. You can think of it as 'animated' HTML. For example, a piece of text can change from one size or colour to another, or a graphic can move from one location to another, in response to some kind of user action, such as clicking a button. It has been enabled by a number of technologies such as HTML, Scripting languages including Java script, VB Script, document object model, layers and cascading style sheets.

12.5.1 Scripting Languages

Scripts are executed when a document is loaded to modify a document's contents. Scripting enables modifying the document's contents dynamically. The ability to do so depends on the scripting language itself (e.g., the "document.write" statement in the HTML object model supported by some vendors).

JavaScript is often used to create dynamic HTML documents. One of the more practical things that JavaScript does at this time is it increases the aesthetics and friendliness of websites by adding author-specified user events to static pages. For example, JavaScript allows the page author to embed into a page, statements, which allow user response to certain common events such as when you move the mouse cursor over a link.

12.5.2 Document Object Model (DOM)

The Document Object Model refers to mapping the Web page for developers. Broken down into components, the characteristic properties of each element of a Web page is detailed by DOM, thereby detailing how the components might be manipulated and, in turn, the page can be manipulated. The Document Object Model is prescriptive, as it defines what components we may or may not modify, and how it is to be done, as decided by a panel of developers.

The inventory of components which make up a web page are rather arbitrarily

defined. For example, one panel of developers may decide to break the Web page down into a certain Set X of components while another panel may prefer to reduce a page into Set Y of components. Perhaps these sets overlap to some degree, but that, too, is arbitrary.

A particular DOM does not necessarily apply to a particular Web page. Web pages are not necessarily compatible, because web browsers are not. Microsoft and Netscape do not share the same DOM between their two browsers and, as a result, the anatomy of web pages between them differs as well.

12.5.3 Layers

In general, 'layer' refers to elements that can be positioned at exact coordinates on the page. These elements can be defined with the DIV, SPAN, LAYER, or ILAYER tags. Layers created with DIV and SPAN are referred to as CSS layers because their properties are defined by the Cascading Style Sheets specification by the World Wide Web Consortium. This specification defines style properties (e.g., font, colour, padding, margin, word-spacing) in addition to the positioning properties associated with layers (top, left, z-index, visibility).

Microsoft Internet Explorer 4.0, allows users to change style properties on the fly with a scripting language such as JavaScript or VB Script. Netscape Navigator 4.0 cannot change style properties after the page has been loaded, but positioning properties can be changed with JavaScript.

Both Microsoft and Netscape support CSS layers in their 4.0 browsers. Only Netscape Navigator 4.0 and later version supports layers created with the LAYER and ILAYER tags.

Self Check Exercise

3) What is DHTML?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.5.4 Difference Between HTML and DHTML

As stated earlier, DHTML is a combination of various technologies like scripting languages, CSS, layers, DOM, including HTML. It can be said to be the superset of HTML. Dynamic HTML allows a web page to change after it is loaded into the browser. Any communication with the web server for an update need not be there. Whereas HTML pages are static and have to be communicated to the web server if changes are made.

12.6 WEB INTERFACE TO DATABASE LINKING

Servlet technology is one of the most used technologies for web interface and server pages are an extension of it. Introduction of this technology has hyped the web environment. Lot of its applications is seen with e-commerce. This technology can be exploited very well for the users convenience in libraries. Servlet was

initially developed by Sun Microsystems, afterwards Microsoft also brought Active Server Pages (ASP) – a more user-friendly environment of developing Servlets. Later, Sun also introduced Java Server Pages (JSP). These act as database connectivity languages, which help in creating Web Interfaces to databases; convert the user queries into the database query languages; and retrieve the results and display on the browser.

12.6.1 Server Pages

Server pages are nothing but codes embedded in HTML tags to perform specific routine on certain request. An excellent example is, running a SQL query on request and fetching the results in a particular format or in a customised format. There are two server page technologies presently accepted and widely in use, i.e., Java Server Page (JSP) and Active Server Page (ASP), developed by Sun and Microsoft respectively.

The features of these server pages i.e., ASP and JSP are described below:

12.6.1.1 Overview of Active Server Page

Developed by Microsoft, Active Server Page (ASP) is one of the most powerful server pages. It runs on the *Internet Information Server (IIS)*, which comes with *Windows 2000* and *Windows NT 4.0*. For *Windows NT 4.0* one requires to load *Windows NT Option Pack drivers* downloadable free from Microsoft site. ASP applications can be developed on *Windows 9x* and on *Windows Me* using *Microsoft's Personnel Web Server (PWS)* [Cluts, 2005]. Efforts have been made to use ASP on *Linux* also. *Sun Chilli!Soft ASP* provides full ASP support for *Apache*, *Lotus*, and also *Microsoft*, running on *HP-UX*, *Linux*, *Sun Solaris* and *IBM AIX* [Hughes and Birznicks].

How ASP works

An ASP file normally contains HTML tags, just as a standard HTML file. In addition, an ASP file can contain server scripts, surrounded by the delimiters `<%` and `%>`. Server scripts are executed on the server, and can contain any expressions, statements, procedures, or operators that are valid for the scripting language. One can use any scripting language – VBScript, JScript, PERL or PYTHON. VBScript is default script language for ASP while JScript needs to be defined in program. To use *PYTHON* or *PERL* one needs to load the scripting engines.

There are Objects defined in ASP to perform different tasks. Here, you have to write there is an object called as 'Response' which may further uses a function 'write' within the class 'Response'.

For example, to write name of a person, code should be as follows and the file name as `<filename>.asp` extension,

```
<html>
<body>
<%
response.write("S.R.Ranganathan")
%>
</body>
</html>
```

This code uses VBScript and finally writes on the browser S. R. Ranganathan. If one is using JScript he has to define at the beginning of the document the Scripting language he is using. For example,

```
<%@ language="Jscript" %>
```

12.6.1.2 Overview of Java Server Page

Java Server Page (JSP) is developed by Sun Microsystems, based on Java technology. There are a number of servers which supports JSP. Sun offers the Java Web Server to run JSP pages. But still there are servers available free either as standalone or add ons. *Tomcat* which is developed under *Jakarta project*, is the server which can be used as standalone as well as add on for the Apache web server, only a slight modification is required in Tomcat to connect to Apache.

How JSP works

The biggest advantage with JSP is the use of Java as scripting language. It also has the feature of Tag library where one can define his own tags and use it like XML (Extensible Markup Language) (Java Server page) [<http://java.sun.com>]. Another important feature with JSP is use of Java beans and more advanced version is EJB (Enterprise JavaBeans). JSP by default uses JavaScript. But one can also use PHP.

```
<%@ page language="java" %>
```

```
<html>
```

```
<body>
```

```
<%
```

```
    out.print("S.R. Ranganathan");
```

```
%>
```

```
</body>
```

```
</html>
```

JSP is object-oriented that means once an object is defined it can be called many times. To call a class JSP provides a method called as page property named 'Import'. One can call any number of classes in one page. For example,

```
<%@ page import="java.sql.*" %>
```

But use of many classes reduce the speed of loading of webpage in browser. Most of the work can be done without defining Tags or using Beans.

Self Check Exercise

4) What are server pages? Name different server pages available.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.7 INTRODUCTION TO XML

Internet today has progressed from just text-based browsers to completely GUI (Graphical User Interface) browser. The representation of information has also changed. We have seen the development of HTML in 1980s and then its various versions in 1990s, then came the era of XML (eXtensible Markup Language). XML facilitates domain specific information interchange systems through semantic tagging.

Various features of XML have been discussed in Unit 11. In this section we will be discussing those aspects necessary for Web Content Development. You are advised to go through this Unit in conjunction with Unit 11.

12.8 XML DOCUMENT DESIGN

As it has been mentioned in the previous Unit, one can define a structure of XML document and give others to write the XML document against his own schema to avoid the mistakes. A schema is nothing but the logical structure of document. This schema is called as 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.

12.8.1 Document Type Definition (DTD)

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. Further extension of *book.xml* as a catalog for the description of book is as follows:

```

1—<?xml version="1.0" encoding="UTF-8"?>
2—<!DOCTYPE catalog SYSTEM "C:\hydra\book.dtd">
3—<catalog>
4—  <book>
5—    <title>Mastering XML</title>
6—    <author authorship="primary">
7—      <f_name>Ann</f_name>
8—      <l_name>Navaroo</l_name>
9—    </author>
10—   <author authorship="secondary">
11—     <f_name>Chuck</f_name>
12—     <l_name>White</l_name>
13—   </author>
14—   <author authorship="secondary">
15—     <f_name>Linda</f_name>
16—     <l_name>Burman</l_name>
17—   </author>
18—   <edition>First edition</edition>
19—   <place>New Delhi</place>
20—   <publisher>BPB</publisher>
21—   <physical_desc>xxxiv, 882p.</physical_desc>
22— </book>
23—</catalog>

```

In this example, the second line,

```
<!DOCTYPE catalog SYSTEM "C:\hydra\book.dtd">
```

indicates that this XML file should use *book.dtd* for the validation of document. The absolute path of the DTD file is mentioned, even the relative path can be mentioned as the case may be. The tag starting from <! > contains processing instruction for parser. Comments are always given inside <!-- -->.

For example, <!--Your comments -->

The listing of *book.dtd* is as follows:

```
1—<?xml version="1.0" encoding="UTF-8"?>
2—<!ELEMENT catalog (book+)>
3—<!ELEMENT book (title, author+, edition, place, publisher, physical_desc)>
4—<!ELEMENT title (#PCDATA)>
5—<!ELEMENT author (f_name, l_name?)>
6—<!ELEMENT edition (#PCDATA)>
7—<!ELEMENT place (#PCDATA)>
8—<!ELEMENT publisher (#PCDATA)>
9—<!ELEMENT physical_desc (#PCDATA)>
10—<!ELEMENT f_name (#PCDATA)>
11—<!ELEMENT l_name (#PCDATA)>
```

First line shows the version of XML specification used for the generation of file. The second line means the element *catalog* will have at least one or more occurrence of element *book*. That means the element *book* is mandatory and repeatable field in the terms of CDS/ISIS language. The sign plus (+) represents the field in question is mandatory and repeatable. There are other notations for different kind of repeatability.

No sign	Element is mandatory and non-repeatable
Plus (+)	Element is mandatory and repeatable
Star (*)	Element is optional and repeatable
Question mark (?)	Element is optional and non-repeatable

Third line means the element *book* has the child elements *title*, *author* (which is repeatable and mandatory), *place*, *publisher* and *physical_desc* (physical description). The element which contains data is known as *Leaf element*. A *Leaf element* is represented by either #PCDATA, i.e., parsed character data (the data will be parsed by XML parser) or #CDATA, i.e., character data (XML parser will not parse the data). Parsed character data or Character data are nothing but the actual data content of the element. For example, we look the corresponding xml file to this DTD,

```
<title>Mastering XML</title>
```

the data "*Mastering XML*" is parsed character data according to our DTD. The use #CDATA is to represent the formulae or data which contains restricted characters for example, angular brackets (<) and so on. So in our example, all the leaf elements are represented as #PCDATA. One can use DTD in the same XML file but it is always good to use a separate DTD file for an XML file.

Once you define DTD any non-conformity with the structure during the generation of XML document will be complained at the time of validation.

Further details about XML have been discussed in Unit 11.

5) What is DTD?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.9 MULTIMEDIA WEB RESOURCES

Multimedia technology implies using a computer to present simultaneously multiple types of media, in an integrated manner. These can include sound, graphics, video, text, animation, or any other form of information representation. The popularity of the Internet as an information resource is largely due to the ease of navigating through various websites effortlessly. But another equally influencing factor is its multimedia presentation. The ability to incorporate of images, animations, sounds to web content has made tremendous impact on the end-users and information providers alike. The components of multimedia systems are:

Text: Text can be plain ASCII text as in case of Notepad or formatted text as in MS-Word and WordPerfect like applications. Text can also be Hypertext (written in HTML) as in the case of web documents.

Audio: Audio can be either speech or music. There are various ways of storing audio in computers. The popular formats are the .wav and .mp3 formats.

Images: Images form a major part of the web as pictures, logos, banners, etc. Image files are large in size. Hence it should considered when web resources are designed how many images to incorporate and what resolution as huge file sizes hinder transmission of data on the Net. The popular image formats are .jpg, .gif and .tiff.

Video: Video is another media format being used on web. Video files again are very large and take a lot of bandwidth. Hence caution should exercised on what clips are included and how much video is downloaded. The popular formats for video are the .avi, .mov and .mpeg.

Animation: Animation is another type of popular multimedia content on the web. Animation is a technique in which each frame of a film or movie is produced individually, whether generated as a computer graphic, or by photographing a drawn image, or by repeatedly making small changes to a model unit, and then photographing the result with a special animation camera. When the frames are strung together and the resulting film is viewed at a speed of 16 or more frames per second, there is an illusion of continuous movement (due to the persistence of vision). Generating such a film is very labour intensive and tedious, though the development of computer animation has greatly sped up the process. Graphics file formats like .gif, .mng and Flash allow animation to be viewed on a computer or over the Internet. Increasingly computer animation is created by means of 3D computer graphics, though 2D computer graphics are still widely used.

Self Check Exercise

6) What do you mean by Multimedia resources? What are its components?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.10 WEB SERVERS

A Web server is a program which serves the files that form Web pages to Web users (whose computers contain HTTP clients that forward their requests) using the client/server model and the World Wide Web's Hypertext Transfer Protocol (HTTP). Every computer on the Internet that contains a Web site must have a Web server program. Two leading Web servers are Apache, the most widely-installed Web server, and Microsoft's Internet Information Server (IIS). Other Web servers include Novell's Web Server for users of its NetWare operating system and IBM's family of Lotus Domino servers, primarily for IBM's OS/390 and AS/400 customers.

Web servers often come as part of a larger package of Internet- and intranet-related programs for serving e-mail, downloading requests for File Transfer Protocol (FTP) files, and building and publishing Web pages. How well it works with the operating system and other servers, its ability to handle server-side programming, security characteristics, and publishing, search engine, and site building tools that may come with it are the consideration for choosing a Web server.

Self Check Exercise

7) What is a Web Server?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

12.10.1 Website Hosting

Every webpage, email, file, or online service is stored ('hosted') on a computer (called a 'server') that is connected to the Internet. Putting the resource on a server to provide access to it on Web is called 'Website Hosting'.

When Web servers were first prototyped, they served simple HTML documents and images. Today, they are frequently used for many other applications. A Web site's success or failure is due to its content and functionality rather than the server used to power it. However, the choice of the correct server, and understanding its capabilities and limitations is an important step in hosting web based collections or services.

Web servers work in several steps to facilitate information hosting on web. It serves static content to a Web browser at a basic level. This means that the Web server receives a request for a Web page such as,

<http://www.webcompare.com/index.html>

and maps that Uniform Resource Locator (URL) to a local file on the host server.

In this case, the file *index.html* is somewhere on the host file system. The server then loads this file from disk and serves it out across the network to the user's Web browser. This entire exchange is mediated by the browser and server talking to each other using Hypertext Transfer Protocol (HTTP). This workflow is shown in the figure below:

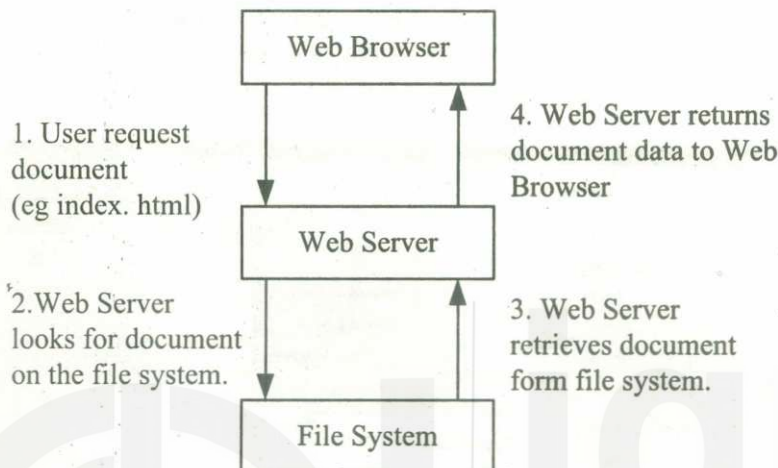


Fig. 12.2: Work flow of web hosting

12.11 TOOLS FOR WEB PAGE DESIGNING

Many software used for web page designing, like FrontPage, Dream Weaver, Coffee cup HTML editor, etc are available. Each software has its own advantages and disadvantages. The most popular ones are Dream Weaver and FrontPage.

The following section shows you how to use Dream Weaver to create and edit Web documents.

12.11.1 Dream Weaver

The different steps involved in designing a Web page are:

- Create a document
- Tables
- Attach a behavior to an image
- Import a Microsoft Word in HTML document
- Format text using HTML styles
- Create links
- Apply a template
- Create a jump menu

Create a Document

Suppose you are to create a home page for a site. As you build this document, you will add a page title, layers, images, text, and links; desirably all the pages will contain the same design components. The Object palette (choose Window

> Objects), is used to add objects to the document. The Property inspector (choose Window > Properties), is used to set properties or attributes for objects in the document.

Save Your Document

Save the blank document created when you launched Dream Weaver. It is better to save all the files in one folder and the index.html or index.htm file in a separate folder to avoid cluttering of files.

Define the Title of the Page

Defining a page title for an HTML document helps users identify and keep track of a page that they are browsing. The page title appears in the title bar of the browser when a page is viewed in it. When a page is bookmarked, the page title appears in the bookmark list. If a document is created without a page title, the document appears in the browser with the title 'Untitled Document'.

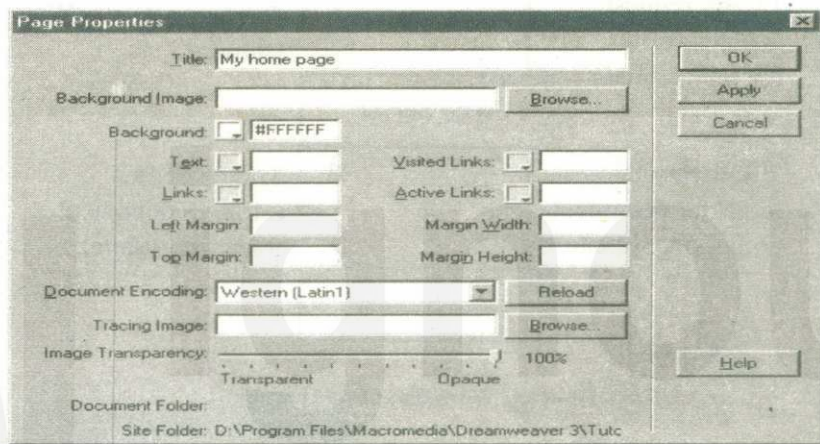


Fig. 12.3: Page properties box

In the Page properties the text colour, background colour, hyperlinked colour, active link and visited links colour also can be set according to user's requirements.

Adding an Image

Adding an image in Dream Weaver is a very simple. For this, the cursor is placed on the screen where the image is wanted and the image can be selected by using the Insert Image option. Mouse-over images can be inserted with the Rollover option.

Formatting Text

Text can be formatted in the Document window by setting properties in the Property inspector. First, select the text you want to format, and then apply the changes. You can change the font, colour and size of the text.

Before formatting the text, a couple of paragraphs should be created. Add a paragraph return after the first sentence and after the second sentence.

- ★ If the Property inspector is not open, choose Window > Properties.
- ★ With the insertion point anywhere in the layer, press Control+A (Windows) to select all the text in the layer.
- ★ In the Property inspector's second Format pop-up menu, which currently reads Default Font, *select* Arial, Helvetica, sans-serif.
- ★ In the Size pop-up menu, *select* 3.
- ★ The text in your document automatically updates to reflect the changes.

Importing a Word Document to a HTML Document

The word document is to be imported first into Dream Weaver, and then any nonstandard HTML code in the document should be cleaned using a Dream Weaver feature called Clean Up Word HTML. The Clean Up Word HTML feature provides options for cleaning up or fixing HTML tags, defining CSS (Cascading Style Sheets) elements, setting a page background colour, removing Word-specific markup tags, and converting Word font sizes and headings to HTML size attributes.

Making Font Style

There are two methods of changing the font style:

Method 1: Select the text, right click and select from Properties.

Method 2: Select the text, and choose from the Text menu.

We can also prepare the ordered list, unordered list, paragraph or general text, text size, etc. through the Text menu.

Creating Hyperlinks

Hyperlinking provides link from one hypertext to another or text to image or image to text or image to image. It can be within the same document or can be linked to other documents. Hyperlinks lead to related information on other pages within that document itself. Hyperlinks can be created by using the Link option of the Properties menu.

By default, the hyperlink opens the linked page in the same window. If you want the hyperlink to open the page in a separate window, then Select the Target option.

12.12 SUMMARY

The presentation and display of websites are of importance in popularising and publicising information. Libraries can also develop and design their own Web pages and host them on the Internet to gain visibility as well as provide access to their local OPAC. Technologies like HTML, DHTML, and XML help in creating and designing Web pages. Especially XML is gaining a lot of interest in the library community, as it allows for semantic tags. It is in the interest of the librarians to keep abreast of emerging technologies, so that they can provide the user better services. Advanced tools for web page designing are now available. As these are user friendly, using them is simple as remembering of syntax of all tags are not required. To incorporate various multimedia elements into the resources has become simpler with those tools. However, one aspect should be kept in view that not to overload the network capacity by adding unnecessary elements which make the file size large.

12.13 ANSWERS TO SELF CHECK EXERCISES

- 1) WWW is a standard designed for the access of multimedia information over Internet. 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 Internet browsers support these protocols. WWW is used to view the web documents. The web documents are written in a particular language supported by WWW that is, Hypertext MarkUp Language.
- 2) Uniform Resource Locator (URL) is the address of 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 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 displayed.

- 3) Dynamic HTML (DHTML) allows a web page to change after it is loaded into the browser - there does not have to be any communication with the web server for an update. DHTML is named so because even after it is loaded on the browser it can be changed, without having to send a request to the browser. The different technologies that make up DHTML are DOM (Document Object Model), Layers, CSS (Cascading Style Sheets), and HTML.
- 4) Server pages act as database connectivity languages, which help in creating Web Interfaces to databases, which convert the user queries into the database query language and retrieve the results and display on the browser. Some of the available server pages, today, are ASP (Active Server Pages) from Microsoft, JSP (Java Server Pages) from Sun Microsystems Inc. and PHP (Hypertext Preprocessor).
- 5) One can define his own structure of XML document and give others to write the XML document against his own schema to avoid the mistakes. A schema is nothing but the logical structure of document. This schema is called as DTD (Document Type Definition)
- 6) Multimedia resources are those that contain more than one medium of content. The components are text, audio, images, video, animation, etc.
- 7) A Web server is a program that, using the client/server model and the World Wide Web's Hypertext Transfer Protocol (HTTP), serves the files that form Web pages to Web users (whose computers contain HTTP clients that forward their requests).

12.14 KEYWORDS

Browser	: User's software program for viewing & browsing information on the Internet.
DTD	: Document Type Definition-this is the formal specification of a markup language, written using SGML
Element	: An <i>element</i> is a fundamental component of the structure of a text document. Some examples of elements are heads, tables, paragraphs, and lists. Elements can contain plain text, other elements, or both.
Home Page	: Originally, the web page that a browser is set to use when it starts up. The more common meaning refers to the top level document of a set of web pages.
HTML	: HyperText Markup Language-HTML is an SGML DTD. In practical terms, HTML is a collection of platform-independent styles (indicated by markup tags) that define the various components of a World Wide Web document.
Linux	: A free Unix-type operating system originally created by Linus Torvalds with the assistance of developers around the world. Developed under the GNU General Public License, the source code for Linux is freely available to everyone.
Plug-In	: This is a program that the browser uses to manipulate a downloaded file. It differs from a Helper Application in that the plug-in works inside the browser window.

- SGML** : Standard Generalized Markup Language – a standard for describing markup languages. A mark-up language alters the presentation of the content of a document. A generalised mark-up language can deal with many different document types, and a standard generalized mark-up language deals with all these documents in a standardised way.
- Server** : A main computer that serves the other computers attached to it.

12.15 REFERENCES AND FURTHER READING

A beginner's guide to HTML. <<http://archive.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimerAll.html>>.

A beginner's guide to HTML. <<http://archive.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimerAll.html>>.

Beginner's guide to DHTML. <<http://www.wdvl.com/Authoring/Tutorials/dhtml.html>>.

Cluts, Nancy Winnick. (2005). *ASP from A to Z*. <<http://msdn.microsoft.com/library/default.asp?URL=/library/en-us/dnasp/html/aspatoz.asp>>.

Comparing JavaServer Pages and Microsoft ActiveServer Pages. <<http://java.sun.com/products/jsp/jsp-asp.html>>.

Devika, P.M. (2003). *Web based content development*. In: PGDLAN course material, MLI-006, Unit 9. New Delhi: Indira Gandhi National Open University.

Hughes, Chris and Birznies, Gunther. *Serving up web server basics*. <<http://webcompare.internet.com/webbasics/index.html>>.

Introduction to JSP. <http://www.w3schools.com/asp/asp_intro.asp>.

Introduction to web multimedia. <http://www.w3schools.com/media/media_intro.asp>.

JavaServer pages. <<http://java.sun.com/products/jsp/keyfeatures.html>>.

Linking style sheets to HTML. <<http://www.htmlhelp.com/reference/css/style-html.html>>.

Marchal, Benoit. (2000). *XML by example*. New Delhi: Prentice-Hall.

Navarro, Ann; White, Chuck and Burman, Linda. (2000). *Mastering XML*. New Delhi: BPB.

Powell, Thomas A. (2000). *The HTML complete reference*. 2nd Edition. New Delhi: Tata McGraw Hill.

Savio, Nadav. (1997). *What is Dynamic HTML?* <<http://hotwired.lycos.com/webmonkey/geektalk/97/39/index3a.html?tw=authoring>>.

Sun ChilliSoft ASP homepage: products. <<http://www.chilisoft.com/products/default.asp>>.

The Jakarta Project. *Jakarta Tomcat*. <<http://jakarta.apache.org/tomcat>>.

The XML Bible. *XSL Transformations*. (Chapter 17). 2nd Edition. <<http://www.ibiblio.org/xml/books/bible2/chapters/ch17.html>>.

Web Developer's Virtual Library. <<http://www.wdvl.com/Authoring/DHTML>>.

UNIT 13 MULTILINGUAL CONTENT DEVELOPMENT (USING UNICODE)

Structure

- 13.0 Objectives
- 13.1 Introduction
- 13.2 Character Representation in Computer
- 13.3 American Standard Code for Information Interchange (ASCII)
 - 13.3.1 Problems with ASCII
- 13.4 Indian Scenario and Indian Standard Code for Information Interchange (ISCII)
 - 13.4.1 The GIST Card
 - 13.4.2 Web Applications and GIST
 - 13.4.3 Constraints with ISCII
- 13.5 UNICODE
 - 13.5.1 Design Principles of UNICODE
 - 13.5.2 Structure of UNICODE
 - 13.5.3 UNICODE and Its Features
- 13.6 Web Content Development Through UNICODE
- 13.7 Applications of UNICODE
 - 13.7.1 Multilingual Encoding
 - 13.7.2 Multilingual Transliteration
 - 13.7.3 Multilingual Works
- 13.8 Applying UNICODE to the Libraries
- 13.9 Problems Associated with UNICODE
- 13.10 Summary
- 13.11 Answers to Self Check Exercises
- 13.12 Keywords
- 13.13 References and Further Reading

13.0 OBJECTIVES

In the previous Unit, you have learnt about various aspects of technology, tools and techniques of Web content development. Such contents may be in different scripts. It will be worthwhile to know how the characters are represented in a computer and whether the standard international codes are able to accommodate other scripts particularly Indian characters. In this Unit we will be discussing the standard codes which are available for handling the scripts with particular reference to Indian scripts.

After reading this Unit, you will be able to:

- understand method of Character Representation in Computers;
- learn the international standard codes such as ASCII;
- apprehend the need for and challenges in non-Latin character coding;
- understand the Indian languages representation in computers;

- know the techniques and standards for handling Indian languages in computers;
- understand the features of UNICODE; and
- learn about the possible applications of UNICODE and its limitations.

13.1 INTRODUCTION

Information communication has come of age with IT providing it the necessary boost and tools. One universal form of communication today is over networks. But these information are mostly in English – mostly because until recently, representing scripts of different languages in computer lacked a standard for representation. To represent thought content in written form one needs a script. Representing the idea through signals on a surface is Scripting. There are many theories regarding the invention of scripts. Many believe that it was born in Mesopotamia. Few believe that it was born separately and parallel in Indus, Mesopotamia, Mesoamerica and China [Ancientscripts.com]. In India, Sanskrit was written in different scripts from time to time. The ancient scripts known are Bramhi around 7th BC and Kharosti during 5th BC. Of which Bramhi became the mother script for many Indian scripts. Meanwhile many scripts came into being like Grantha and later on Nagari, i.e., Devanagari, during 5th and 7th century respectively. In due course of time, many languages developed due to the migration of civilization and accordingly different scripts in the Indian sub-continent were developed from Bramhi [Encyclopedia Britannica, 1982].

With the development of Internet as information super-highway, it is felt that information should also flow in source language script rather than translations and should be accessible to the rest of the world. Representation of characters of a script in computer applications in computer applications is a complex process and there are standards for it. ASCII (American Standard Code for Information Interchange) is one such standard, which was developed by a committee of software giants in 1962 and revised in 1967. But it could hardly address the problem of representing more than one script thus, failing to become a global character encoding system. ASCII is a one-byte code for character representation, which can represent only 256 (0–255) characters in extended form at a stretch but it uses only 7-bits, that means only 128 characters are accommodated. On the lines of ASCII, other countries drafted their standards for their individual language script. On the same guidelines, to address the problem in India, Bureau of Indian Standards (BIS) has developed ISCII (Indian Standard Code for Information Interchange).

13.2 CHARACTER REPRESENTATION IN COMPUTER

Within computers, each character is designated by a single number, because computer can only understand numbers and, that too, only in the form of electrical current flow. But actual process is even more complex as computer is an electronic device and it only understands voltage ON and OFF. That is why a notation is developed to represent voltage ON and OFF as '1' and '0' respectively. This particular system of notation is known as 'Binary system' and coding of characters using such system is known as 'character encoding'. In ASCII, when 'A' labeled key is pressed, computer understands it as '01000001' (or decimal 65) and prints 'A' on the screen.

Character on the screen	Binary value used to process it	Character on the screen	Binary value used to process it
1	0110001	A	1000001
2	0110010	B	1000010
3	0110011	C	1000011
4	0110100	D	1000100
5	0110101	E	1000101

Self Check Exercise

1) Explain how are characters represented in computers?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

13.3 AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII)

ASCII- 1963

ASCII was developed by a working committee known as X3.4 constituted by IBM, AT&T and its subsidiary Teletype. Basically, this code was 8-bit character coding system and the arrangement was logical, i.e., if one wants to write sorting programs he/she can manage using ASCII value. But this particular version of code considered only the upper case characters and there, a lot of blank positions were left assuming that it could be used for Roman smalls.

ASCII- 1967

Same committee in 1967 came out with a modified version of ASCII code. This version included Roman smalls and few more additional characters, which were absent in the ASCII 1963 version, like {, }, |, ~, etc. Strictly speaking, ASCII is not an 8-bit code rather it is a 7-bit code, that means it uses only 128 character positions. It should be noted that the numbers '0-9' are given 48 to 57; Capital letters 'A-Z' are given 65 to 90 and small letters 'a-z' are given 97 to 122, to get correct results when words or lines are sorted.

Extended ASCII (Latin - +)

Before networking of personal computers became common font and software developers for languages written in non-Roman scripts laid out non-standard character maps, usually employing extended ASCII positions 128-256, to accommodate non-Roman characters. The non-standard character maps made it possible to use non-roman characters with many stand-alone applications and, after loading special fonts, even on the Web. But interoperability remains problematic with non-standard schemes [Monotype Imaging Inc].

The increasing popularity of the Internet has made interoperability a major issue. HTML (RFC1866) has relied for character encoding on ISO-8859-1, known as

Latin-1, a version of extended ASCII, which is appropriate only for English and Western European languages. However, ISO-8859 has made it possible to standardise character sets for some widely used scripts and it now includes a series of 10 standardised character sets for writing in occidental alphabetic languages:

- 1) Latin1 (West European)
- 2) Latin2 (East European)
- 3) Latin3 (South European)
- 4) Latin4 (North European)
- 5) Latin-Cyrillic
- 6) Latin-Arabic
- 7) Latin-Greek
- 8) Latin-Hebrew
- 9) Latin5 (Turkish)
- 10) Latin6 (Nordic)

Compared to Unicode, these ISO-8859 character sets are extremely limited but they are usable on the Internet, for instance, with MIME software, and they are a major improvement over the 7-bit US-ASCII. Characters 0 to 127 are always identical with US-ASCII and the upper positions hold characters for other scripts [McMohan, 1997].

13.3.1 Problems with ASCII

No doubt ASCII was a great invention. After the invention of Internet, the world started talking about global village and then it seemed less desirable to have English as imposed language on the Internet and, thus, everybody started talking about multilingualism.

When ASCII uses 1-byte character encoding it can only handle 256 characters that means after 128 characters it can accommodate 128 more. So it can hardly handle one or at most two languages and that shows its limitation to handle all world languages. So definitely at this juncture ASCII fails and UNICODE seems to be a promising solution which covers all the world languages scripts.

Self Check Exercise

- 2) Discuss the ASCII standard for character coding.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

13.4 INDIAN SCENARIO AND INDIAN STANDARD CODE FOR INFORMATION INTERCHANGE (ISCII)

The BIS standard IS 13194:1991 is 8-bit code. It covers 10 Indic languages derived out of Bramhi. ISCII (Indian Standard Code for Information Interchange) is an extended ASCII. It uses last 128 characters position for characters representation in Indic scripts. It uses the same Keyboard as of ASCII with Inscription of other 10 languages over it. It uses control characters SO (Shift out) and SI (Shift in) for selection of ASCII and ISCII respectively. The arrangement of characters in ISCII is phonetic. Switching from one language to other transliterates the text [<http://tdil.mit.gov.in>].

ISCII is no doubt a solution for the Indian scenario but when it comes to global representation of characters both ASCII and ISCII fail to represent the characters at a stretch, as both use single byte code.

In ISCII, ASCII characters are placed in the lower half (0-127) of the 8-bit code table while Indian script characters are in the upper half (160-255). ISCII caters to the following 10 Indian scripts – Devanagari, Gujarati, Punjabi, Bengali, Assamese, Oriya, Telugu, Tamil, Malayalam, Kannada. The ISCII code table is a superset of all the characters required for the above mentioned scripts. First version was released in 1983 and adopted by the Bureau of Indian Standards (BIS) in 1991 after revisions in 1986 and 1988.

13.4.1 The GIST Card

Graphics and Intelligence based Scripting Technology (GIST), developed by C-DAC (India), supports major Indic languages. GIST technology uses ASCII code for character representation using the extended ASCII table. GIST product uses underlying standards like ISCII, ISFOC (for font representation on screen) and INSCRIPT (common keyboard layout for Indian scripts). The Indic languages can be divided in two parts – Consonants and Vowels. The lay out of script of all the Indic languages are same. GIST is developed on 8-bit encoding system, i.e., it is an extended ASCII system. It uses same code for same vowel and consonant in different Indic language. Thus transliteration becomes very handy. Only it is required to change the mode (or switches) of language. Indian languages (multilingual) word-processing software, like iLeap, also developed by C-DAC, work on the same principle and enables transliteration.

Hex	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Hex	Dec	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
0	0	NUL	DLE	SP	0	@	P	·	p								EXT
1	1	SOH	DC1	!	1	A	Q	a	q			ँ	औ	ण	ल	े	०
2	2	STX	DC2	"	2	B	R	b	r			ं	ऑ	त	ळ	ै	१
3	3	ETX	DC3	#	3	C	S	c	s			ं	क	व	ळ	ै	२
4	4	EOT	DC4	\$	4	D	T	d	t			अ	ख	ह	य	ौ	३
5	5	ENQ	NAK	%	5	E	U	e	u			आ	ग	घ	ज	ी	४
6	6	ACK	SYN	&	6	F	V	f	v			इ	घ	न	व	ी	५
7	7	BEL	ETB	'	7	G	W	g	w			ई	ड	न	स	ी	६
8	8	BS	CAN	(	8	H	X	h	x			उ	ष	प	ह	ी	७
9	9	HT	EM	)	9	I	Y	i	y			ऊ	छ	फ	INV	ी	८
A	10	LF	SUB	*	:	J	Z	j	z			झ	न	ब	ा	।	९
B	11	VT	ESC	+	;	K	[	k	{			ऐ	झ	ष	ि		
C	12	FF	FS	.	<	L	\	l				ए	ज	म	ी		
D	13	CR	GS	-	=	M	]	m	}			ऐ	ट	थ	ी		
E	14	SO	RS	.	>	N	^	n	~			ऐ	ठ	ड	ी		
F	15	SI	US	/	?	O	_	o	DEL			औ	ड	र	ी	ATR	

Fig. 13.1: ISCII representing Devanagari characters

GIST helps in development of web based solutions. The software called iPlugin gives facility to develop the website in Indic languages. It provides facility to create chat room and email facility in any Indic language. The biggest advantage with iPlugin is that now one need not download the language font on the local machine. Necessary plugins directly get loaded on the machine when website is accessed, which is known as Dynamic fonts. Many of the websites in Indian languages are using GIST.

13.4.3 Constraints with ISCII

The main disadvantage with ISCII is that it is impossible to represent more than one Indian scripts in the same screen as ISCII uses the same name space, i.e., 160-255 for all the Indian scripts. With ISCII, one can have content in English and one the Indian languages, but not two or more Indian languages at a time. The major problem with ISCII is its character encoding. It uses 8-bit character. Even to convert from English to any Indic language, it requires a lot of labour and from other language to any Indic language it is very difficult. Translingual operations in two or more Indian languages is not possible using ISCII.

Self Check Exercise

3) What is the Indian Standard for character coding in Indian scripts? Explain.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

13.5 UNICODE

UNICODE is developed by UNICODE consortium which is a group of software organisations like IBM, Microsoft, Xerox, Oracle, etc. They came out with a 16-bit code and called it UNICODE, as it promises to cover all the world's scripts. The promising feature of UNICODE is that it can represent 65,536 characters. The first version of UNICODE came in 1991. Since then two more versions came in and the present one is version 3.0.

Unicode, which is identical with ISO 10646 BMP, is based on 16-bit encoding that permits 65,535 characters instead of the 256 characters of Latin-1 (and in the 32-bit version a capacity of 2,080,000,000 characters). The BMP stands for Basic Multilingual Plane, sometimes referred to as Plane Zero. While Latin-1 assigns a value of 1-256 to a character, Unicode assigns the value U+nnnn to a character where nnnn is a four digit number in hexadecimal notation; this value is referred to as a code point [Johnson, Dave].

Unicode can be represented using 7-bit, 8-bit, and 16-bit format standards. UTF-7 (Universal Character Set Transformation Format-7) is a format that breaks the Unicode code point into 7-bit values; these can be transferred through email (which uses the 7-bit ASCII encoding) and on the Internet. This format presents difficulties because some values are ambiguous, but it is still usable. The 8-bit UTF-8 (Universal Character Set Transformation Format-8) format breaks Unicode

values into 8-bit sequences, which work well on the Web. UCS-2 (Universal Character Set -2) stores each character as a 16-bit value and each value corresponds directly to its value (code point) in the Unicode standard.

Unicode encodes text by script, not by language. This avoids duplication of letters. For instance, the Latin A is used without distinction for text in Catalan, English, Indonesian, Swedish, or Swahili. Similarly a letter in Devanagari script could be used in Sanskrit, Hindi or Nepali. Character set standards relates to basic interchange of text. They do not include attributes regarding language, display format, color, or typeface. Unicode characters are made visible through a distinct rendering process that maps characters (entities used in data interchange that generically specify a particular symbol) to glyphs (the particular shapes of a given characters as they are displayed) [Monotype Imaging Inc.]. For example, in English lower-case 'c' is the character (a code for a generic lower-case 'c') transmitted. When the code is received, it is converted into a glyph, a letter with a particular shape that will be displayed on the screen or sent to a printer.

Unicode is fully compatible with ASCII, i.e., the first 128 characters are ASCII characters. Thus, any software using ASCII is compatible with UNICODE without any specific changes. UNICODE has defined different language zones for their corresponding scripts. For example, Devanagari characters start at 2304 and end at 2431. Thus, it includes all the scripts of the world.

13.5.1 Design Principles of UNICODE

The general design principles on which UNICODE is developed are:

- It is a 16-bit code.
- For future expansion surrogate area has been defined which can generate combinations of two 2-bytes packets.
- UNICODE is concerned about characters not glyphs.
- It renders logical ordering for writing, i.e., from left to right and in case of Hebrew, Urdu, etc. right to left.
- Combining characters also follow logical order. Combining characters are non-spacing characters, which form combination to represent a new character.

13.5.2 Structure of UNICODE

UNICODE initially was the interest of software organisations. Earlier, to produce a different language software, it was required to use ASCII code in the same language. But once UNICODE came up, it was really not needed to rewrite the software for different language. So it has become a boon for software developing organisations; side-by-side it became popular as it supports multilingual communication, which ASCII fails to do.

UNICODE is fully compatible to ISO/IEC 10646 set of standard developed by International Organization for Standardisation and International Electrotechnical Commission in 1993. This is another very interesting venture, which also promises 16-bit code, but it does not restrict to 16-bits. It promises to give even 32-bit code, if it really required for character representation. The use of term multiple octet is used because multiples of 8-bits are used in system. This code is called as Universal Multiple-Octet Code (UCS). Since this character code system uses 16-bit and 32-bit, it is known as UCS-2 and UCS-4 respectively.

13.5.3 UNICODE and Its Features

It is known that UNICODE is a 16-bit character code and is an effort to tame all the existing word scripts. But there are few challenges also sought with it:

- It has to take care of existing ASCII coding system (with respect to Indian scripts).
- There are characters, which use a combination of characters; so such combinations have to be managed. (Strictly speaking this problem has not been solved yet, because it is the property of the fonts which are used in writing UNICODE characters).

13.5.3.1 Sixteen Bit Character Code

It is known that UNICODE is 16-bit character code. It can accommodate 65,536 characters. But, the consortium has not exhausted all the character positions, 2048 character positions are left for the future characters. This region of UNICODE is known as Surrogate area. Surrogate area covers 55,296-57,343 character positions. There are two types of Surrogates:

a) High Level Surrogate

It starts from character position 55,296 and ends at 56,319, i.e., total 1024 character positions.

b) Low Level Surrogate

It starts from character position 56,320 and ends at 57,343, i.e., total 1024 character positions.

High level surrogates and Low level surrogates make a pair and represent a character. That means these characters are 4-byte characters, and that way total 10,48,576 characters can be represented which can be used for future. This combination of High level surrogate and Lower level surrogate makes a Surrogate pair.

13.5.3.2 Efficiency

UNICODE is no doubt a very efficient code. All like characters are not repeated. They can be simply borrowed from some other group where they actually occur. For example, Hindi characters are from 2304 to 2431, Bengali is from 2432 to 2559, Telugu is from 3072 to 3199 and so on.

For Carriage Return (CR) the code value is 14 (good old ASCII value) and is retained in UNICODE along with English alphabets. While encoding other scripts where CR is required, UNICODE still uses the value 14. After all, Kannada CR is not different from English CR. This is uniformly applied to all the special characters of ASCII. Thus UNICODE avoids the repetition and adds efficiency to the coding system.

13.5.3.3 Characters not Glyphs

In a sentence the smallest unit is a character, which can be written and can have semantic value. Glyphs are nothing but exposition of characters in graphical form, i.e., how a character appears on display, for example, bold, italics, or whatever. Glyphs are decided by the fonts and the actual value of UNICODE character will remain the same. To display the character glyph, it is necessary that a font must have a glyph in its source, that is why, it is said that it is the property of font used for writing.

If one tries to write in the word 'JAVATM' the appearance of TM on the head of JAVA is due to glyph which shows superscript and is supported by the font. But the value of T will be 84 and M will be 77. That is why it is said that UNICODE is a code for characters not for glyphs.

13.5.3.4 Plain Text

As it has been mentioned above that UNICODE has nothing to do with glyphs, that means it only works out the plain text. For example, if a character is written '65' it is represented as 'A' (which has value in binary system 0000000001000001) irrespective of the format added with the character, which is known as 'Rich character'. To produce a Rich text any markup language can be used like HTML or XML, or even one can use software like MS-Office, etc.

13.5.3.5 Logical Order

UNICODE stores characters in logical order. Logical order means the order in which characters are stored in memory is same as the order characters are keyed in using the keyboard. At certain places it has to be violated because the representation of the text is reverse. For example, Urdu.

Urdu is written from right to left and same is the case with Hebrew too. So when a character is typed it displays from right to left, but the order in which the character is represented in the memory is same. And UNICODE does it automatically. This particular feature of UNICODE is known as bi-directional ordering.

13.5.3.6 Unification

UNICODE has taken care to identify the same character within different language and put it at one place in UNICODE table. For example, Punctuation marks, common letter, etc.

The characters of East Asian scripts are taken in a separate table called as Han table. It covers Chinese, Japanese and Korean (CJK) characters because these scripts have a lot of overlapping characters, many a time the same character is used in all languages. Similarly, UNICODE has 'Combining characters table' which covers non-spacing characters that can be combined with other characters.

13.5.3.7 Compatibility Characters

Compatibility characters are the characters, which actually would not have been encoded because these are the variant form of some characters, which are already encoded.

For example,

Colon (:) {character position 58} has two Compatibility characters - one at character position 847, i.e., 'Ratio' and other is 1417, i.e., Armenian full stop. There is only formatting variance between Colon and Ratio. In 'ratio' the distance between the points is more. These characters reside in Compatibility area as one can see the index of character positions, but it is not necessarily true that all characters reside in this area only. For example, the variant of Latin character 'A' (character position 65) has another variant Full length Latin character 'A' which is given a position at the end of UNICODE table.

13.5.3.8 Dynamic Composition

Dynamic composition is used to generate different accent forms. There is a block in UNICODE character set map called as 'Combining Characters'.

To generate, Å, the characters A and ' are used.

Å → A + '

Of course, the glyph Å is permitted in the font set used. In Indic languages, both vowels and consonants can be used as combining characters (non-spacing characters).

13.5.3.9 Equivalent Sequence

These are pre-imposed character codes available in the UNICODE character set. But same characters may be generated using composed characters. For example,

Å → A + '
192 65 768

So, in this case, 192 is equivalent to 65 and 768. That means value 65 + 768 is canonical to 192. It is the responsibility of the system to convert the values in any one form, i.e., either 192 or 65 + 768. This is known as 'Normalization'.

13.5.3.10 Convertibility

UNICODE promises conversion to any standard code, because each character is given a distinct number. This particular feature of UNICODE had emerged in 1993 version of code.

Self Check Exercise

4) What are the important features of the UNICODE standard?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

13.6 WEB CONTENT DEVELOPMENT THROUGH UNICODE

The most popular use of UNICODE is multilingual content development over Internet. Ideally one should be able to write in any language. Presently, Windows 2000 and further Windows XP are UNICODE compatible. Still lot more has to be done on Linux platform. Windows 2000 supports only two Indic scripts, i.e., Devanagari and Tamil. The representation of character in browsers can be done in two ways. First is by changing the mode of language and second, by using the codes of characters as HTML supports UNICODE. The following example shows that the page can be written in both modes.

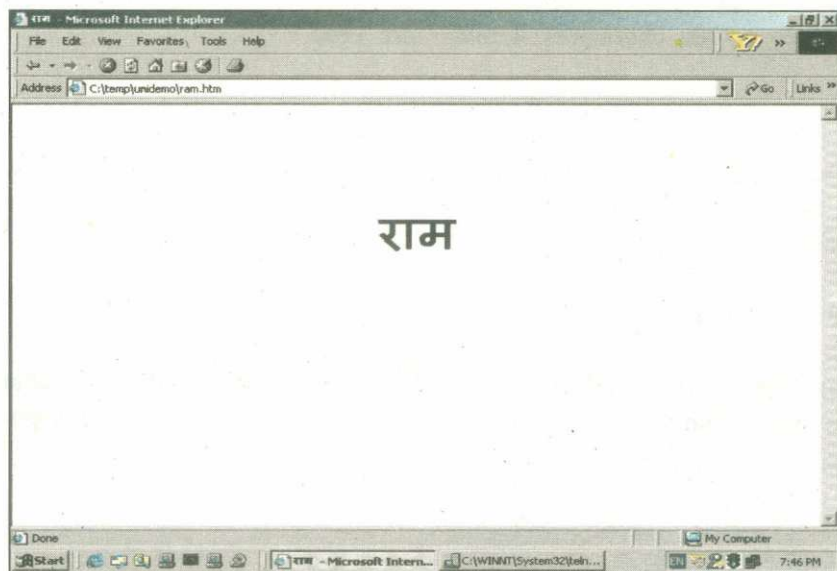


Fig. 13.2: Content Representation in Devanagari

Code using Devanagari script in Windows 2000

```
<html>
<title>राम</title>
<body><center><font size="7">
राम</font></center>
</body>
</html>
```

HTML page using the UNICODE values in Windows 2000:

```
<html>
<title> &#2352 &#2366 &#2350 </title>
<body> <center> <font size = "7">
&#2352 &#2366 &#2350 </font> </center>
</body>
</html>
```

Both approaches are good but since Windows 2000 gives the facility of virtual keys to type Devanagari fonts, it is easier. Using UNICODE facilitates writing in more than one script in a single document.

13.7 APPLICATIONS OF UNICODE

UNICODE promises to cover all the world scripts including extinct scripts. It also takes challenge of future addition of scripts.

13.7.1 Multilingual Encoding

Most popular use of UNICODE is in multilingual editing. One can write in any language, if he/she knows the language and codes. But the problem of such writing is to know codes for all characters. So definitely some multilingual editor is needed, with which we can write in any language. The advantage of it is that each script can be read in any UNICODE compatible editor.

13.7.2 Multilingual Transliteration

Once being aware of the character codes, the approach towards transliteration can be taken. This feature is particularly advantageous with regard to Indian languages because all Indian languages use more or less the same consonants and vowels. Though some languages have some additional characters but in

UNICODE these are arranged in such a way that there is fixed difference in the value of same phonetic character belonging to different scripts. For example, the difference between all Telugu and Devanagari characters is 768.

13.7.3 Multilingual Works

In Unicode multilingual works like polyglots can be composed and also searched online with the text in different languages.

13.8 APPLYING UNICODE TO THE LIBRARIES

India is a country of diversity. Many languages are spoken and many scripts are used. According to one survey, English is the language spoken by less than 10%. Besides, India holds voluminous information in many of the native languages. Libraries particularly Digital libraries can record information in the language and script of the source document. Even the bibliographic information can be represented in the language of the document. Once UNICODE is adopted, one can exploit the advantages of transliteration so that data can be transliterated wherever and whenever it is required. It is not uncommon that much of the Sanskrit literature is available in different scripts of India. In such cases, even if the Sanskrit text is keyed in using Devanagari, one can read the same text in his script (for example Telugu). In other words, it is not required to enter the text in Telugu, rather the Devanagari script can be transliterated on the fly.

In case of metadata (bibliographic information), the semantics of information can be given using XML (Extensible Markup Language), provided that libraries participating in the program should have consensus on a definite set of tags. Thus catalogue of one library can be seen and read in other language.

Self Check Exercise

5) Discuss the possible applications of UNICODE.

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of the Unit.

.....

.....

.....

.....

.....

13.9 PROBLEMS ASSOCIATED WITH UNICODE

UNICODE is a 16 bit code to represent all the world script. However in case of Chinese, the figure of 65,536 is insufficient. Chinese is an ideographic script which has more than 40,000 characters. But only 6000 characters are in day-to-day use. Japanese is a little different from Chinese and thus politically they demanded for separate zone, which creates a political dilemma with code [Goundry, 2001].

UNICODE is used in Windows OS. There are developments in the Linux environment but the problem of glyph still exists. Though Windows 9x, Me, XP are shipped with USP10.DLL which is also known as *Uniscribe* but some of the features of this program do not function for representation of Complex Scripts. There are text editors developed in Linux that support UNICODE. Yudit is one such editor which supports European languages. UNICODE compatible 'emacs'

is also available for download. Besides there are efforts to make Linux in the Indic languages but still now these projects are mostly a sort of preliminary work.

Browsers alone cannot display multilingual text without the support of operating system. In machine with Windows 9x and Me one can't get combining characters for Indic languages. The primary reason is not having the glyphs for corresponding code value. That is, these operating systems do not have character rendering engines for Indic languages.

One advantage with ISCII is that transliteration can be achieved on the fly by simply changing the mode, as ISCII uses the same name space. But it proves costly and makes it impossible to represent more than one Indian language on the same screen. Transliteration can be achieved in UNICODE also, but it requires programs to compute the differences in the values of codes of two different languages. This is a trade off.

13.10 SUMMARY

Today there is a shift in rendering information on the web in different languages. Internet has brought a movement for the globalisation of information. Since the world has great deal of diversity in cultures, languages and scripts, UNICODE has taken up the challenge for the globalisation of world scripts in IT as well as Virtual world (Internet). There are several technical and political constraints in its development. Updations of it are coming from time to time, which include additional and modified rules. Though this standard is an effort of software companies, it has a variety of applications in today's world and libraries stand a better chance for resource sharing.

13.11 ANSWERS TO SELF CHECK EXERCISES

- 1) Within computers, each character is designated by a single number, because computer can only understand numbers and that too only in the form of electrical current flow. But actual process is even more complex as computer is an electronic device and it only understands voltage ON and OFF. That is why a notation is developed to represent voltage ON and OFF as '1' and '0' respectively. This particular system of notation is known as 'Binary system' and coding of characters using such system is known as 'character encoding'. In ASCII, when 'A' labelled key is pressed, computer understands it as '01000001' (or decimal 65) and prints 'A' on the screen.
- 2) ASCII was developed by a working committee known as X3.4 constituted by IBM, AT&T and its subsidiary Teletype. Basically this code was 8-bit character coding system and the arrangement was logical, i.e., if one wants to write sorting programs he can manage using ASCII value. But this particular version of code considered only the uppercase characters and there were a lot of blank positions were left assuming that it can be used for Roman smalls. (read further details under section 3)
- 3) ISCII (Indian Standard Code for Information Interchange) is extended ASCII. It uses last 128 characters position for characters representation in Indic scripts in the range 0-255 provided by ASCII. It uses the same Keyboard as of ASCII with Inscription of other 10 languages over it. It uses control characters SO (Shift out) and SI (Shift in) for selection of ASCII and ISCII respectively. In ISCII, ASCII character are placed in the lower half (0-127) of the 8-bit code table while Indian script characters are in the upper half (160-255). ISCII caters to the following 10 Indian scripts - Devanagari, Gujarati, Punjabi, Bengali, Assamese, Oriya, Telugu,

Tamil, Malayalam, Kannada. The ISCII code table is a superset of all the characters required for the above mentioned scripts. First version was released in 1983 and adopted by the Bureau of Indian Standards (BIS) in 1991 after revisions in 1986 and 1988. (for read section 4 above).

4) Unicode is efficient in character coding with no redundancies. It is a 16 bit code (allows for 65,536 characters) and has the following features:

- Plain text
- Dynamic composition
- Logical Order
- Unification
- Compatibility characters
- Equivalent sequence
- Convertibility.

5) One of the basic applications of Unicode is the ability to provide computer based services in languages other than English also. It has several other applications such as:

- Multilingual encoding enables information to be presented in different regional languages.
- Multilingual searches
- Multilingual works such as Polyglot
- Transliteration
- Cross Lingual Information Retrieval (CLIR)

13.12 KEYWORDS

ASCII	: American Standard Code for Character Interchange.
Bit	: When information is digitised, it is turned into ones and zeros. So all digital information is made up of bits.
Byte	: A group of data bits that are processed together. Typically, a byte consists of 8 bits. There are kilobytes, Megabytes, Gigabytes, Terabytes, etc. 1 Byte = 8 bits 1 kilobyte = about 1,000 bytes 1 Megabyte = about 1,000,000 bytes 1 Gigabyte = 1,000,000,000 bytes 1 Terabyte = 1,000,000,000,000 bytes.
C-DAC	: Centre for Development of Advanced Computing, India
Code	: A set of symbols for representing something. E.g., most computers use ASCII codes to represent characters.
File	: A format for encoding information in a file. Different file types have different formats. File formats specify first whether the file is a binary or ASCII file, and second, how the information is organised.

GIST

: Graphics and Intelligence based Scripting Technology (GIST) is developed by C-DAC, India. It supports major Indic languages.

Glyphs

: Particular shapes of a given characters as they are displayed/rendered on screen

Hexadecimal

: A numbering system which uses a base of 16. The first ten digits are 0-9 and the next six are A-F. Hexadecimal numbers are used to colour web pages. For example, the hexadecimal equivalent for the colour white is #FFFFFF.

ISCII

: Indian Standard Code for Character Interchange.

Linux

: A free Unix-type operating system originally created by Linus Torvalds with the assistance of developers around the world. Developed under the GNU General Public License, the source code for Linux is freely available to everyone.

Multilingual data

: Data in more than one language in the same document or file.

UNICODE

: Universal code for representing world's scripts in computers

13.13 REFERENCES AND FURTHER READING

Ancientscripts.com. *Origin of writing*. <http://www.ancientscripts.com/ws_origins.html>.

Bhatt, Shashank. (2001). *Character encoding standard for Indian scripts - a report*. <<http://www.cicc.or.jp/english/hyoujyunka/mlit4/7-3India/India.htm>>.

Bureau of Indian Standards. (1999). *Indian Standard Code for Information Interchange (ISCII)* (IS 13194:1991). <<http://varamozhi.sourceforge.net/iscii91.pdf>>.

Czyborra, Roman. (1997). *The ISO 8859 alphabet soup*. <<http://www.wbs.cs.tu-berlin.de/~czyborra/charsets>>.

Encyclopedia Britannica. (1982). *Development of script of India*. <<http://www.geocities.com/Athens/Parthenon/2104/scripts.html>>.

Goundry, Norman. (2001). *Why Unicode won't work on the Internet: linguistic, political, and technical limitations*. <<http://www.hastingsresearch.com/net/04-unicode-limitations.shtml>>.

Internet Service Providers Association of India. (2005). *Internet services industry - an overview*. <<http://www.ispai.com/overview.html>>.

Johnson, Dave. *Concepts of C/Unix internationalization: Unicode*. <http://cns-web.bu.edu/pub/djohnson/web_files/i18n/unicode.html>.

Malaiya, Yashwant K. *Language and scripts of India*. <<http://www.cs.colostate.edu/~malaiya/scripts.html>>.

McMahon, Suzanne. (1997). *Beyond transliteration: digitizing south Asian scripts using the Unicode standard: Phase 1, The Review*. <<http://www.lib.berkeley.edu/SSEAL/SouthAsia/Review.html>>.

Monotype Imaging Inc. *A Unicode font solution: a white paper from Monotype Imaging Inc.* <http://www.monotypeimaging.com/isv/uni_whitepaper.asp>.

Raje, Chandrashekhar. (2001). *Text processing of national languages (India)*.

<<http://www.cicc.or.jp/english/hyoujyunka/af08/8-05.html>>.

Searle, Steven J. (2004). *A brief history of character codes in North America, Europe, and East Asia*. <<http://tronweb.super-nova.co.jp/characcodehist.html>>.

Technology Development for Indian Languages. (2005). *Standards – ISCII*. <<http://tdil.mit.gov.in/standards.htm#iscii>>.

Unicode. (2005). *Frequently asked questions: basic questions*. <http://www.unicode.org/faq/basic_q.html>.

Unicode. (2005). *Frequently asked questions: Indic scripts and languages*. <<http://www.unicode.org/faq/indic.html>>.

Unicode. (2005). *Versions of the Unicode standard*. <<http://www.unicode.org/versions>>.

Wood, Alan. (2005). *Alan Wood's Unicode resources: Unicode and multilingual support in HTML, fonts, web browsers and other applications*. <<http://www.alanwood.net/unicode/>>.

