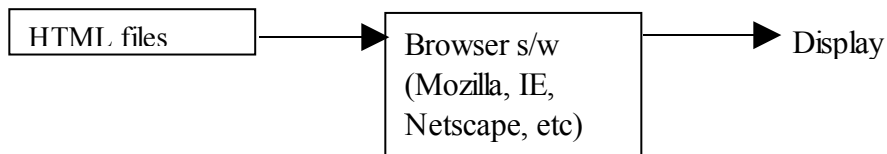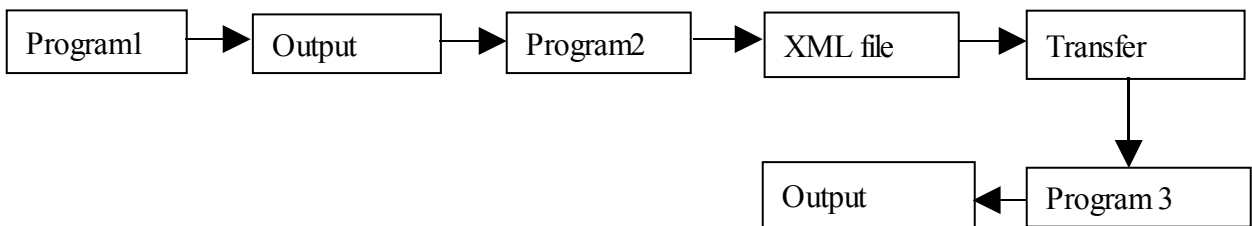Module1 Introduction of XML

INTRODUCTION

[HTML files and XML files cannot do anything. If we have a HTML file say f1.html we need browser software like Internet Explorer, mozilla etc to display that file. So to display an HTML file we need another program. So without this second program we cant use HTML file. Also in XML we need second software.]

HTML files is used by Browser s/w only
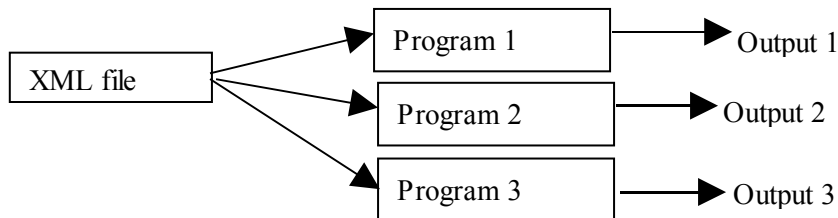
XML file may fed to more than one program and get different outputs

XML file is used to transfer output of one program from machine to another program in another machine

Html file1
```
<HTML>
<BODY> <B> Introduction to XML</B> XML focusses on structure of the XML documents</BODY>
</HTML>
```

Html file2
```
<HTML><BODY><TABLE>
<TR> <TD> Shilpa </TD><TD>21</TD></TR>
<TR> <TD> Varun </TD><TD>22</TD></TR>
</TABLE></BODY></HTML>
```

Anil C.B.   Sree Narayana Gurukulam College of Engineering, Kdayiruppu

Module1 Introduction of XML

Xml file3
```
<XML>
<CLASS>
<STUDENT>
<NAME> Shilpa </NAME>
<AGE>21</AGE>
</STUDENT>
<STUDENT>
<NAME> Varun </NAME>
<AGE>22</AGE>
</STUDENT>
</CLASS>
</XML>
```

The HTML file1 and file2 are displayed in only one format by browser software. Displaying of first file is

Introduction to XML XML focusses on structure of the XML documents
Displaying of second file is

Shilpa	21
Varun	22

The Xml file3 can be dispyled in different format. By creating different programs for each kind of output. Let, we created three programs Program1, program2, Program3 and this three progrms take the input xml file3. We can write programs program1, program2 and program 3 using any language like C, C++, JAVA etc.
Program1 may generate output like

Shilpa	21
Varun	22

Program2 may generate output like
Shilpa age 21
Varun age 22

Program3 may generate output like
Name   Age
Shilpa   21
Varun   22

Markup Language

HTML (hyper Text Markup Language), SGML (Standard Generalized Markup Language) and XML (E**X**tensible **M**arkup **L**anguage) are **examples for markup languages**. The **word 'Markup'** **refers to the sequence of characters or other symbols that you insert at certain places in a text** or word processing file. Markup **indicates how the file should look when it is displayed**. Markup may also **used to describe the document's logical structure**. Markup is a text that is added to the data of a document (file) in order to **convey information about it**. The markup indicators are often **called**

Module1 Introduction of XML

"tags".

By *markup language* **we mean a 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.**

A markup language is a mechanism **to identify the document structures.** If you are familiar with HTML, you have some concept of markup language. If you write a plain text file, it is composed of simple ASCII characters. When a program (like notepad) is used to display the file, all characters in the text file will be displayed using the same font size, type, and boldness. There are no special characteristics to present such type of file.

Markup languages, like HTML or XML, allow special markup to be embedded with the text of a given file. Markups or tags in HTML files will enable the program that displays the file to determine how to show the text. In this way, special text like paragraph may be justify, have a larger and bolder font, or specific display colors may be set. Also additional elements may be added to the file such as numbered lists and tables.

Generalizing from that sense, we define markup, or (synonymously) *encoding*, as any means of making explicit an interpretation of a text. At a banal level, all printed texts are encoded in this sense: punctuation marks, use of capitalization, disposition of letters around the page, even the spaces between words, might be regarded as a kind of markup, the function of which is to help the human reader determine where one word ends and another begins, or how to identify gross structural features such as headings or simple syntactic units such as dependent clauses or sentences. Encoding a text for computer processing is in principle, like transcribing a manuscript from *scriptio continua*, a process of making explicit what is conjectural or implicit, a process of directing the user as to how the content of the text should be interpreted.