

UNIT 1 – Introduction to XML

- Need of XML
- XML Terminology
- XML Standards
- Basic Structure- A simple XML Document
- The Idea of Markup
- Organizing Information in XML
- Creating Well-formed XML Documents
- XML Declaration
- XML Naming rules
- Element Tag- Rules
- Element Attributes- Rules
- Element Content
- PCDATA
- CDATA
- Comments
- Well-formed versus Valid
- HTML versus XML

What is XML?

- XML stands for eXtensible Markup Language
- XML is a markup language much like HTML
- XML was designed to **store and transport data**
- XML was designed to be self-descriptive
- XML is a W3C Recommendation
- XML was designed to be both human and machine-readable.
- XML is a software and hardware-independent tool for storing and transporting data.

What is XML?

- XML Does Not Use Predefined Tags
- The XML language has no predefined tags.
- The tags are "invented" by the author of the XML document.
- HTML works with predefined tags like `<p>`, `<h1>`, `<table>`, etc.
- With XML, the author must define both the tags and the document structure.

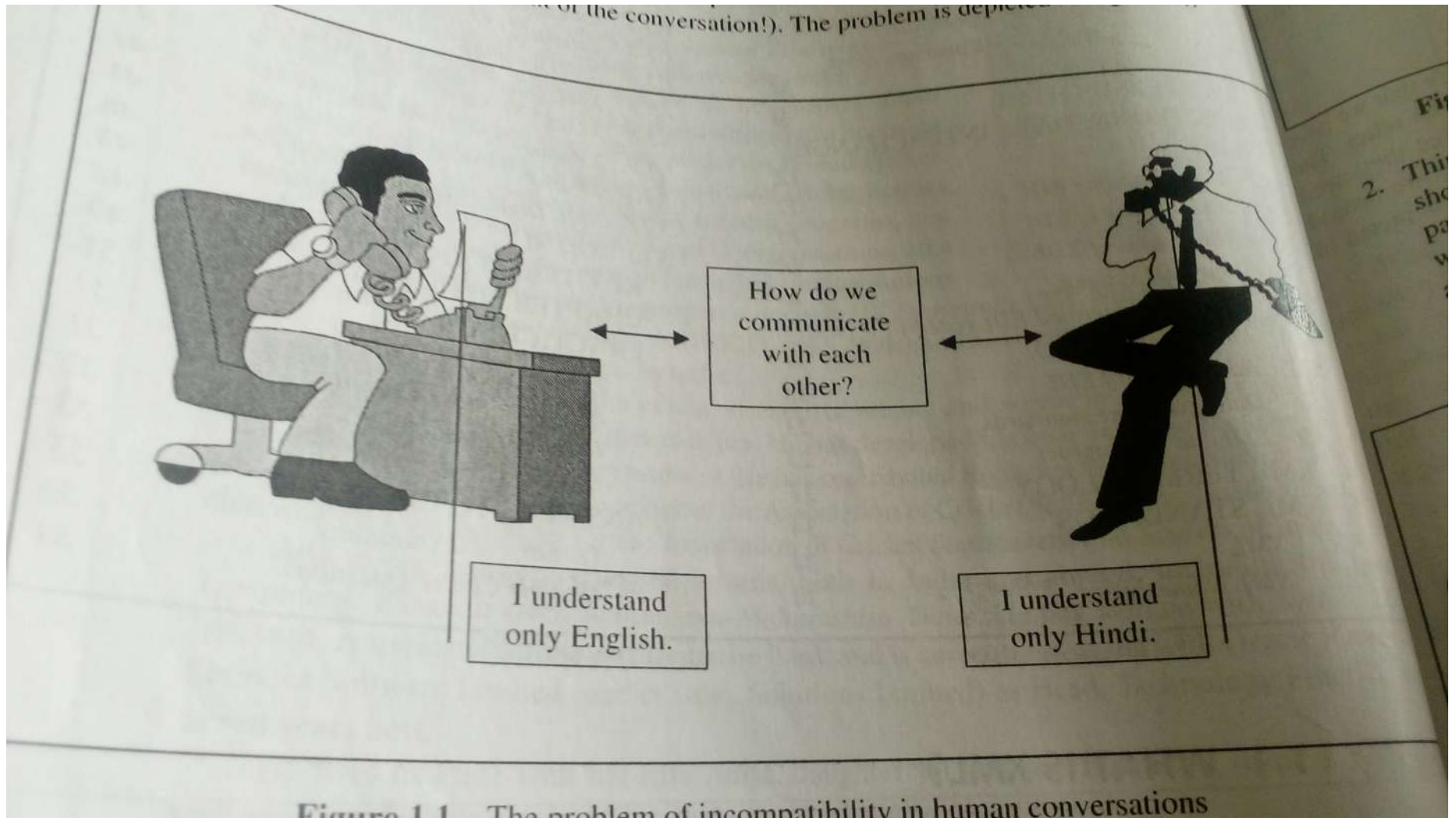
What is XML?

- Extensible Markup Language
- Unlike Programming platform, it is not easy to imagine the end use and applications of XML.
- XML syntax and semantic are well known, but where to use it is usually not clear.
- XML can be used to exchange data between compatible and incompatible applications in Web and non-Web applications.
- XML simlifies the process of data exchange between two or more applications.

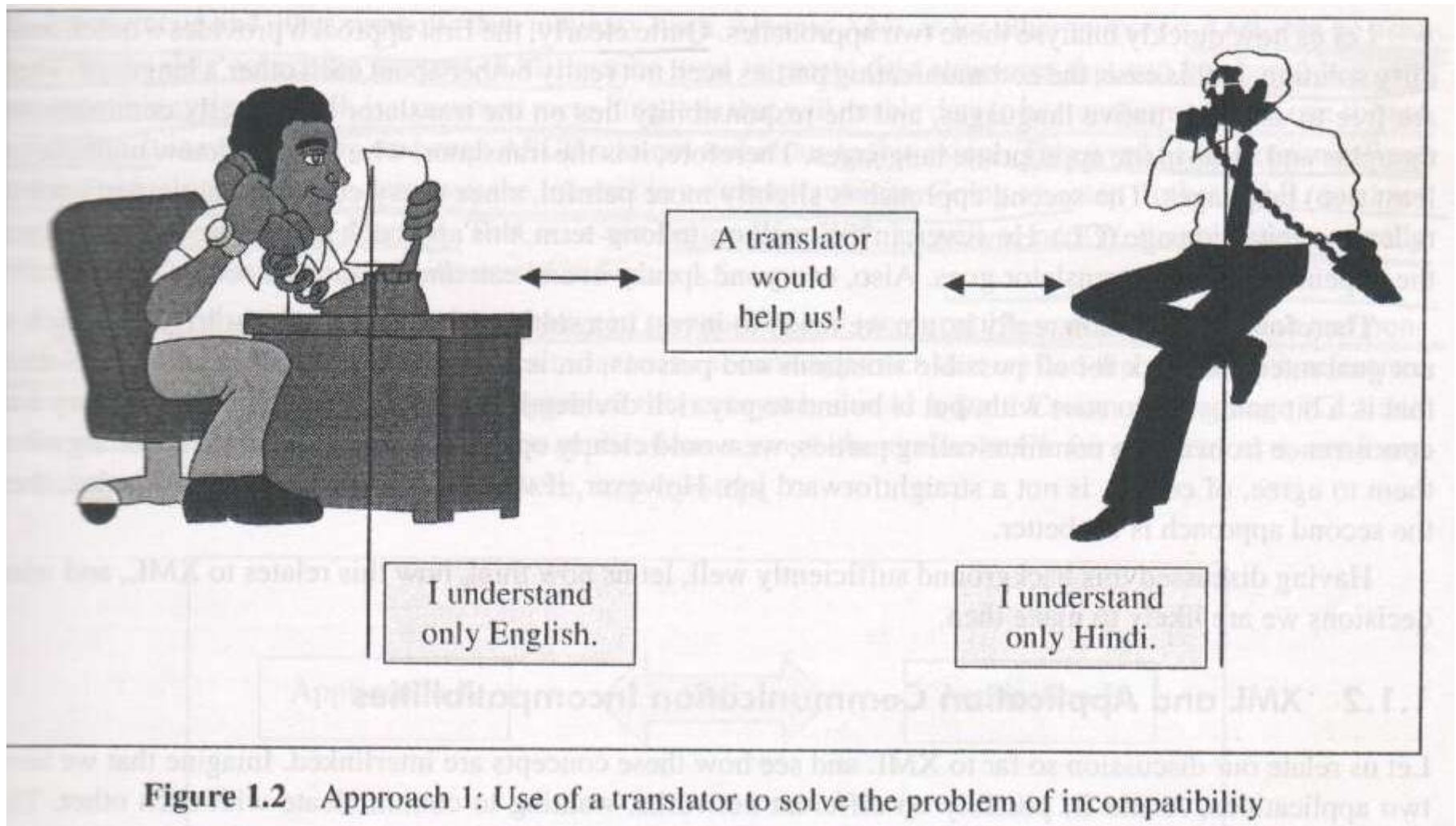
What is XML?

- XML can be used to exchange data across the Internet.
- XML can be used to create data structures that can be shared between incompatible system.
- XML is a common meta-language that will enable data to be transformed from one format to another.
- This would allow organisations and individuals to exchange data over the Internet in a uniform manner.

Why we need XML?



Why we need XML?



Why we need XML?

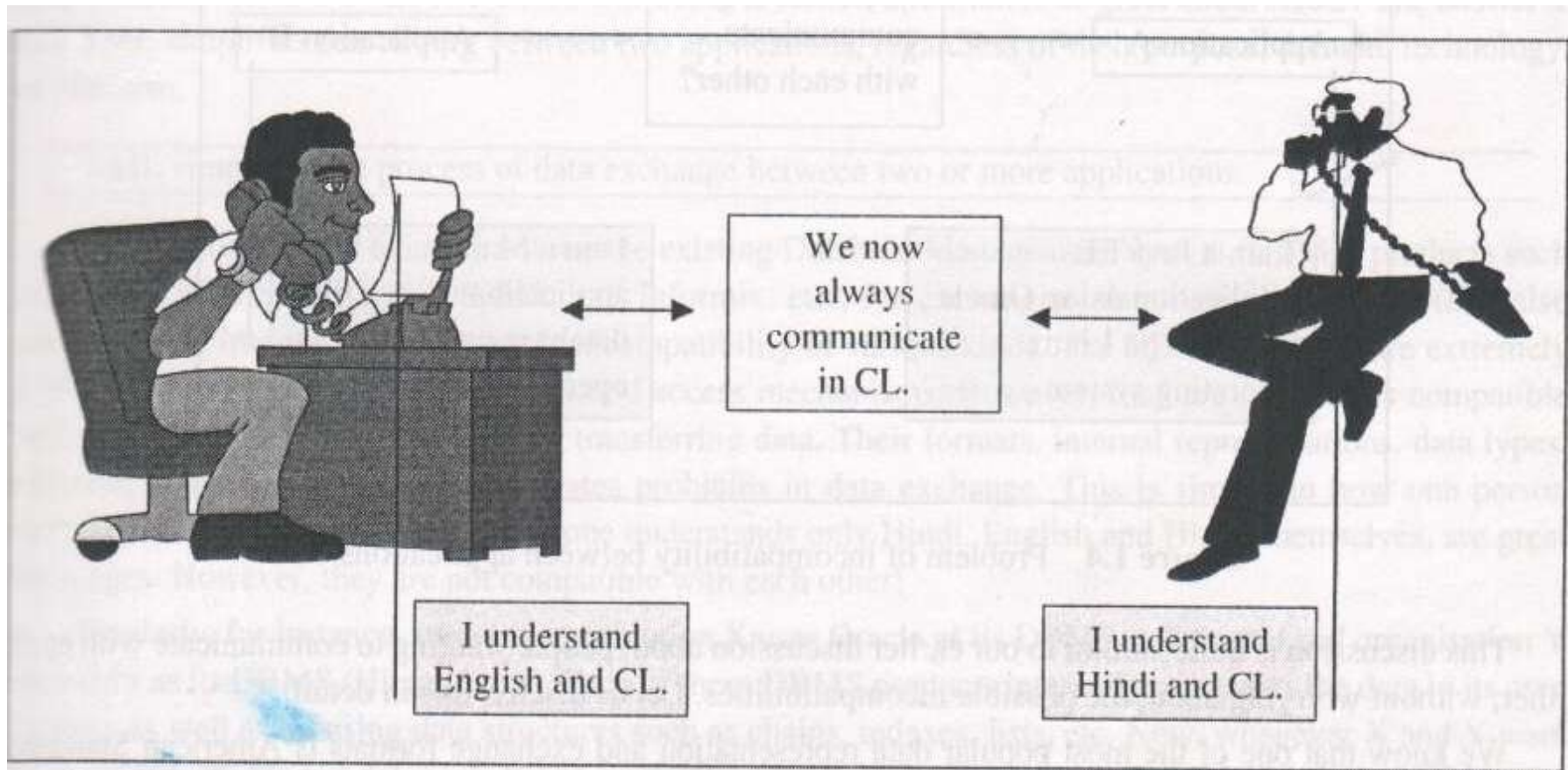


Figure 1.3 Approach 2: Making the communicating parties use a Common Language (CL)

Why we need XML?

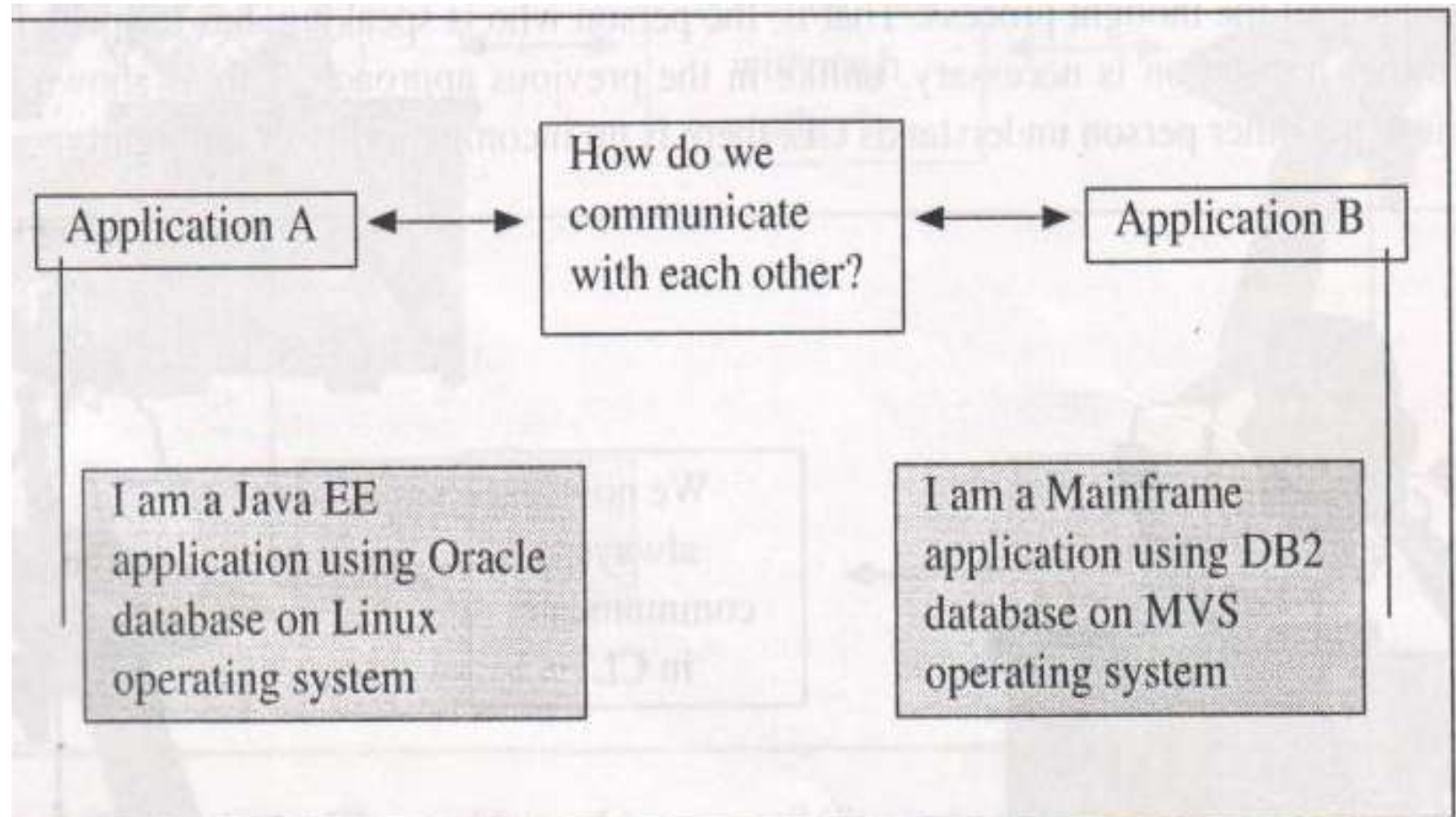


Figure 1.4 Problem of incompatibility between applications

Why we need XML?

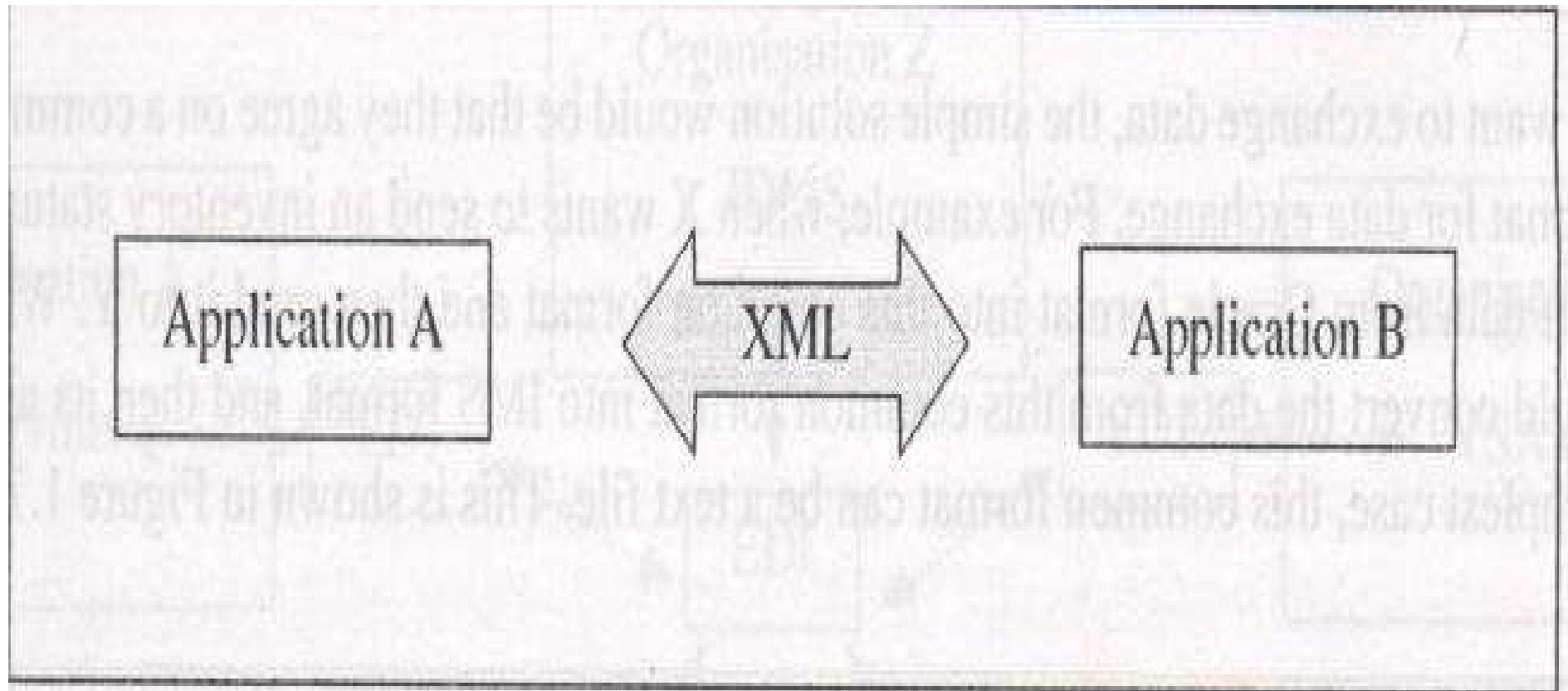


Figure 1.5 XML as the data exchange mechanism between applications

Why we need XML?

Database Management System (DBMS) are incompatible with each other, when it comes to data exchange.

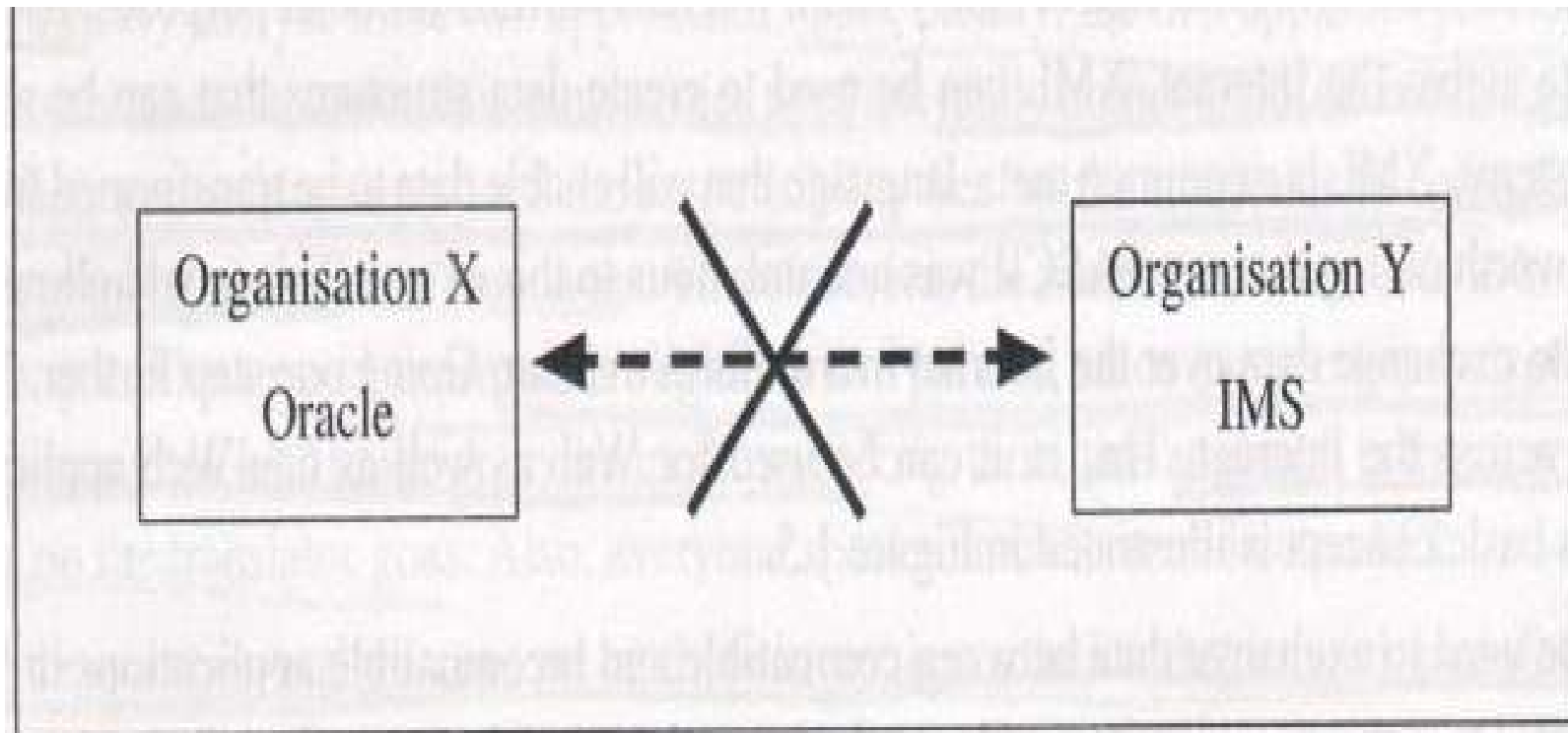


Figure 1.6 Incompatible data formats

Why we need XML?

Solution for data Exchange Management System (DBMS) between Database

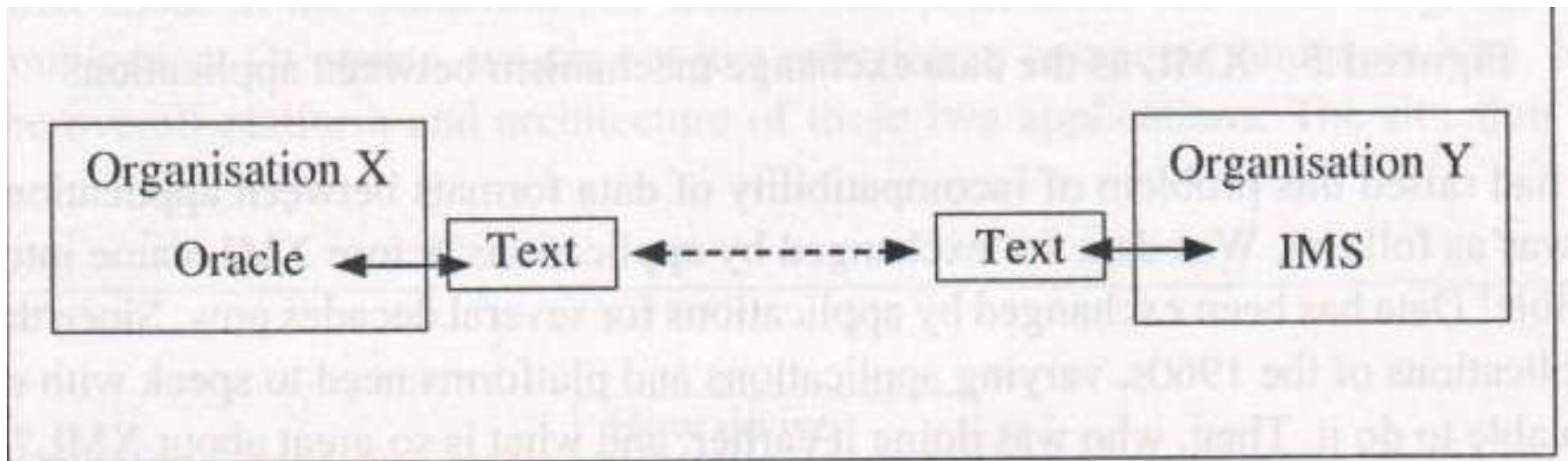
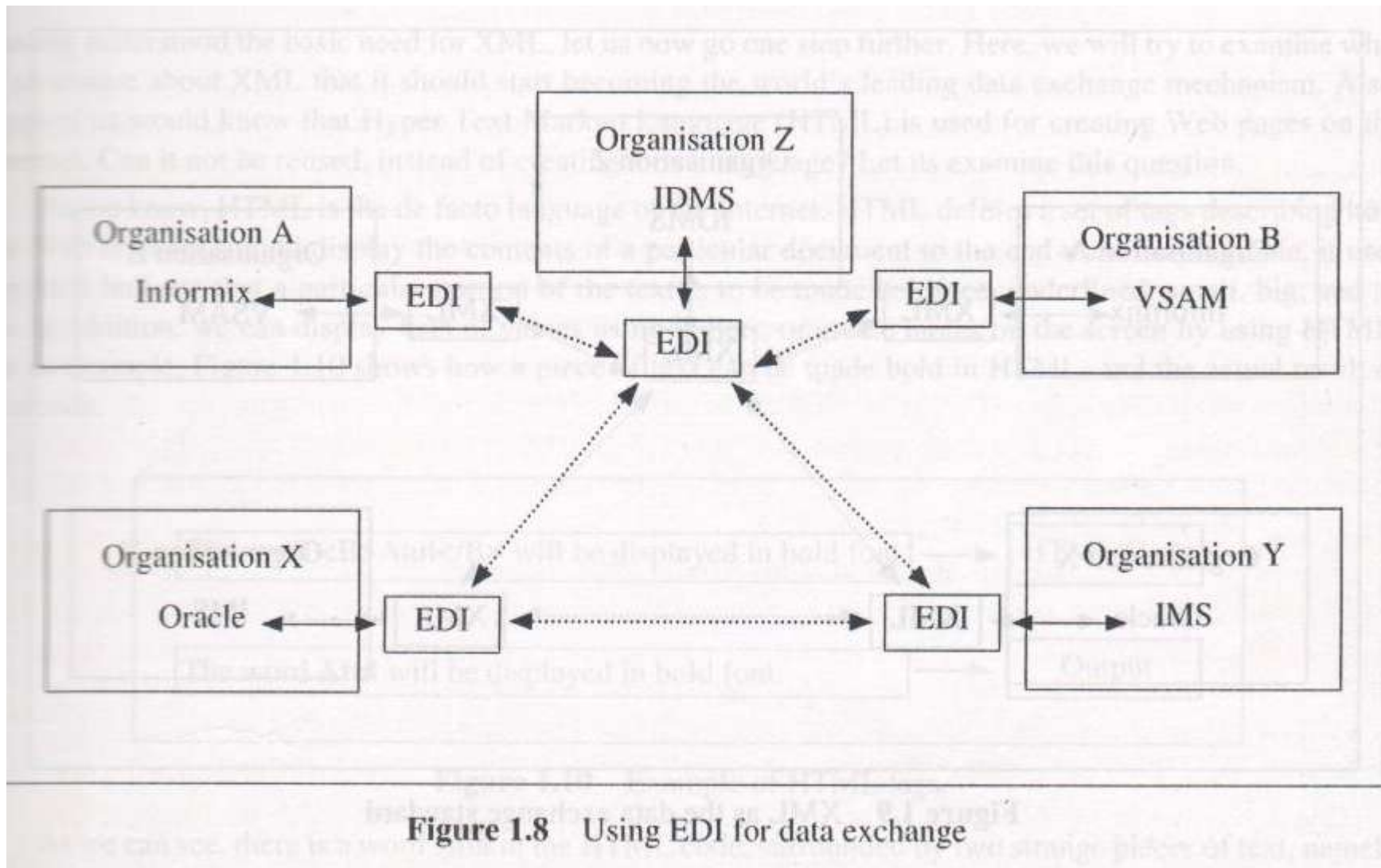


Figure 1.7 Data exchange in a text format

Why we need XML?

Solution for data Exchange Management System
(DBMS) between Database.
Electronic Data Interchange



Why we need XML?

Solution for Data Exchange Management System (DBMS) between Database

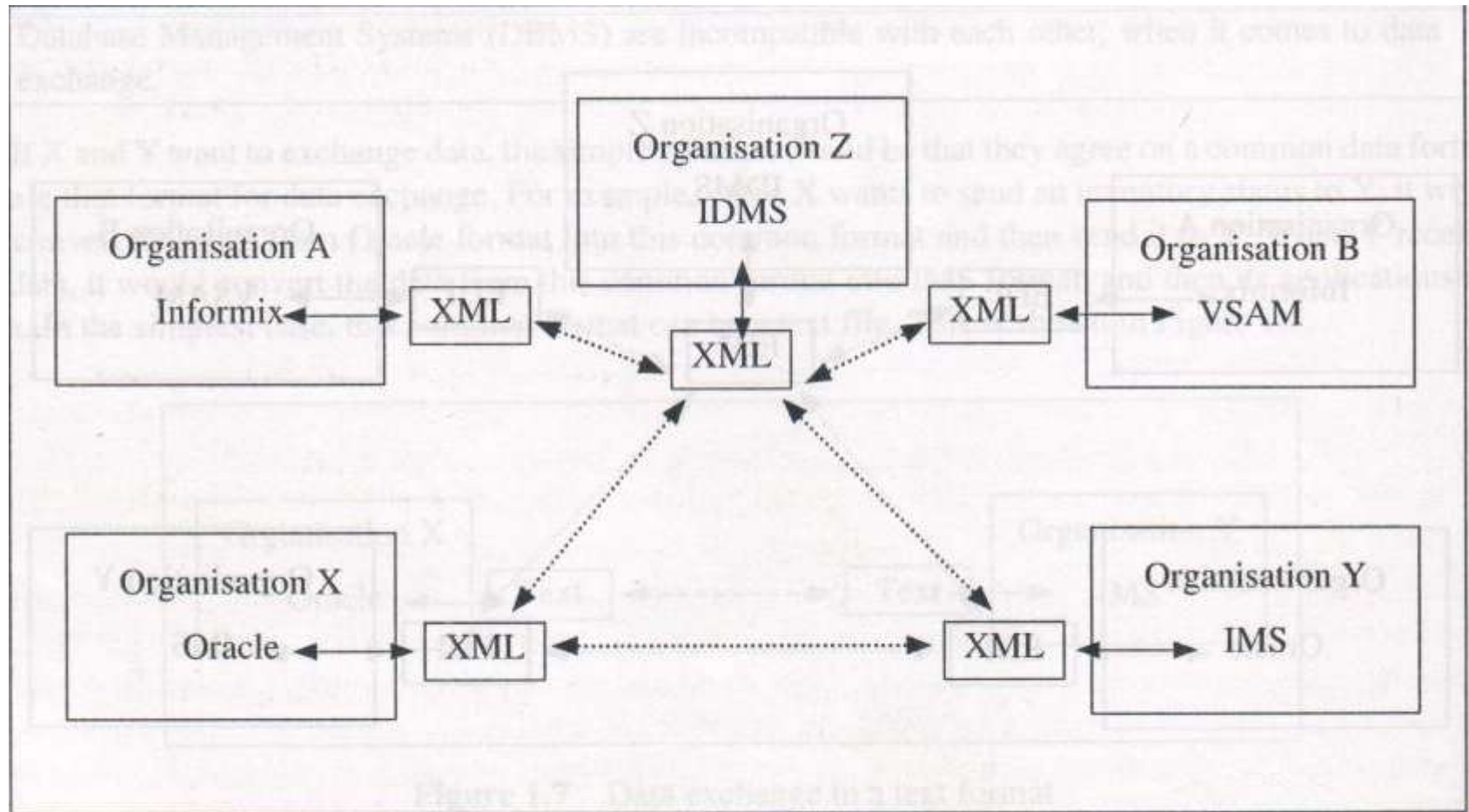


Figure 1.9 XML as the data exchange standard

XML Terminology

```
<?xml version="1.0"?>  
<?xml-stylesheet type="text/xsl" href="books_list.xsl"?>
```

XML tag

```
<BOOK pubyear="1929">  
  <BOOK_TITLE>Look Homeward, Angel</BOOK_TITLE>  
  <AUTHOR>Wolfe, Thomas</AUTHOR>
```

Start element indicator

```
</BOOK>
```

End element indicator

```
<BOOK pubyear="1973">  
  <BOOK_TITLE>Gravity's Rainbow</BOOK_TITLE>  
  <AUTHOR>Pynchon, Thomas</AUTHOR>
```

```
</BOOK>
```

Element value

```
<BOOK pubyear="1977">  
  <BOOK_TITLE>Cards as Weapons</BOOK_TITLE>  
  <AUTHOR>Jay, Ricky</AUTHOR>
```

Element name

```
</BOOK>
```

Attribute name

```
<BOOK pu byear="2001">  
  <BOOK_TITLE>Computer Networks</BOOK_TITLE>  
  <AUTHOR>Tanenbaum, Andrew</AUTHOR>
```

Attribute Value

```
</BOOK>
```

```
</BOOKS>
```


XML Standards

- The World Wide Consortium (W3C) has developed a number of standards for XML.
- Standards are:
 - XML
 - XML Namespaces
 - Document Type Definitions (DTD)
 - Cascading Style Sheets (CSS)
 - Extensible Stylesheet Language (XSL)
 - XML Schemas
 - XML Query Language (Xquery)
 - XLink
 - Xpointer
 - XPath
 - XML Digital Signatures
 - XHTML

XML Standards - XML

- The first version of XML Standard was 1.0 finalised in feb, 1998.
- XML shall be easily implemented on the Internet
- XML shall be usable in a wide variety of applications
- XML shall comply with SGML (Standard Generalized Marked up Language)
- XML documents shall be readable to humans
- XML document design shall be formal and compact
- XML document design shall be done quickly
- XML documents shall be easy to create
- XML markup is not required to be abrupt

XML Standards - Namespace

Namespace help to resolve the confusion when there are duplications in terms used.

Problem:

Suppose,

User A create <book> tag in Book1.xml

User B create <book> tag in Book2.xml

If Book1.xml and Book2.xml going to merge this would be an ambiguity.

Solution:

```
<BOOK_ISSUE xmlns:user_a="urn:book1.xml"
             xmlns:user_b="urn:book2.xml">
```

```
<user_a:book>XML & Related Technology</user_a:book>
```

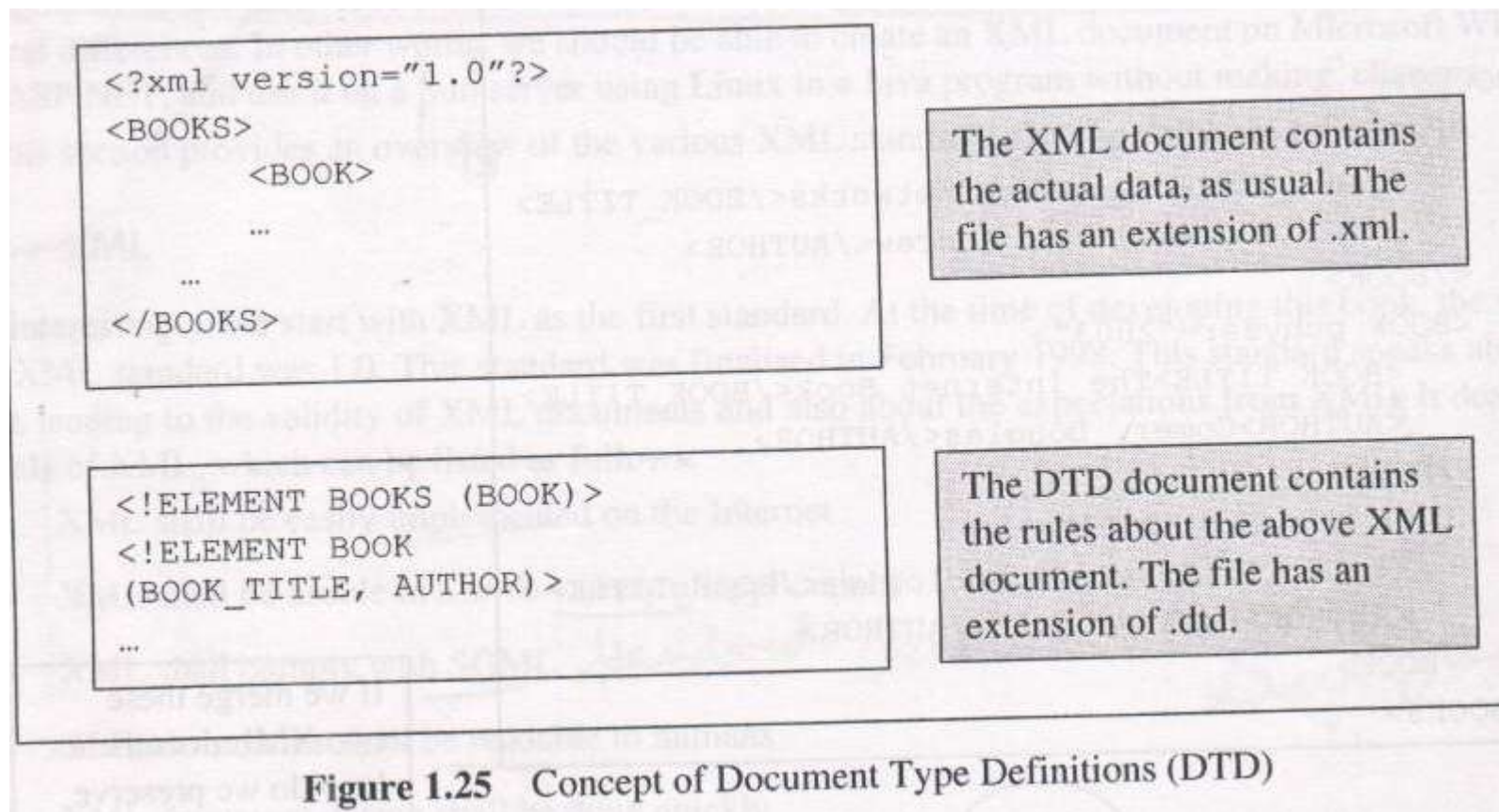
```
<user_b:book>Introduction to XML </user_b:book>
```

XML Standards - DTD

A DTD (Document Type Definitions) file allows us to specify the rules associated with an XML file.

XML Standards - DTD

A DTD (Document Type Definitions) file allows us to specify the rules associated with an XML file.



XML Standards - DTD

A DTD (Document Type Definitions) file allows us to specify the rules associated with an XML file.

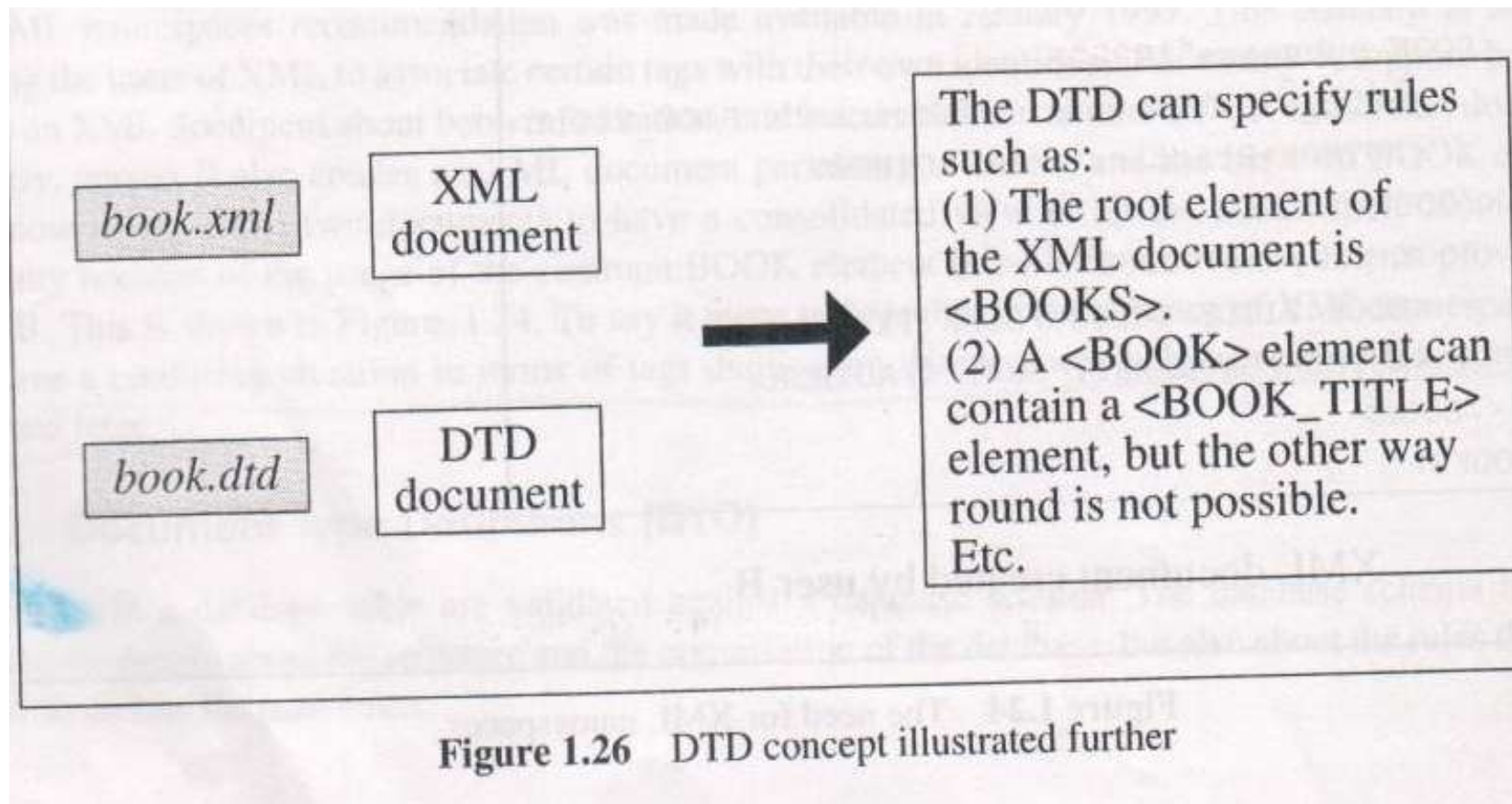


Figure 1.26 DTD concept illustrated further

XML Standards - CSS

Cascading Style Sheets (CSS)

The Cascading Styling Sheets technology allows the formatting of HTML documents in a standard uniform manner.

XML Standards - XSL

The Extensible Stylesheet Language standard provides means for formatting and transforming XML documents in the desired manner.

XSL technology consists of two parts.

- XSL Transformation (XSLT)
- XSL Formatting Objects (XSL - FO)

XML Standards - XSL

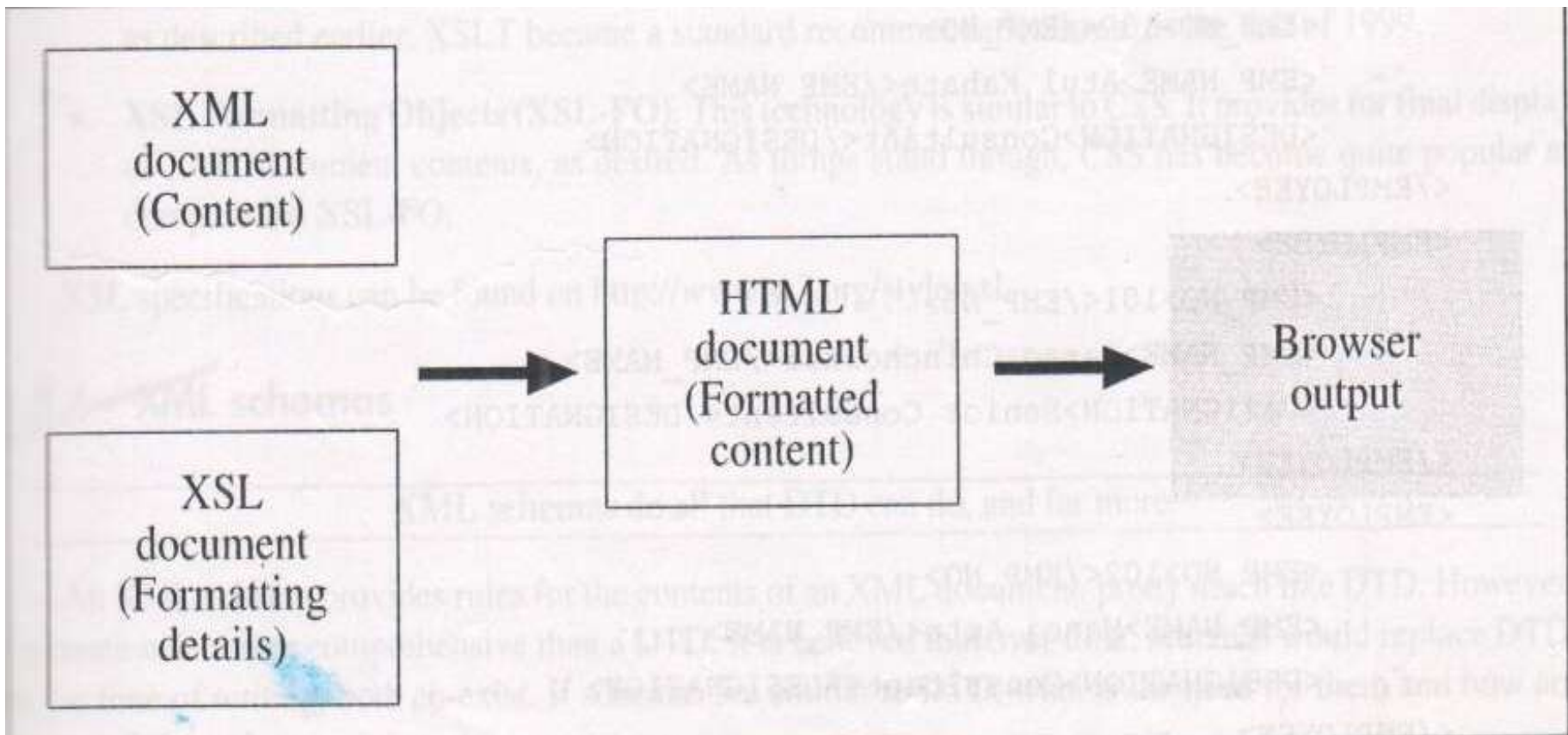
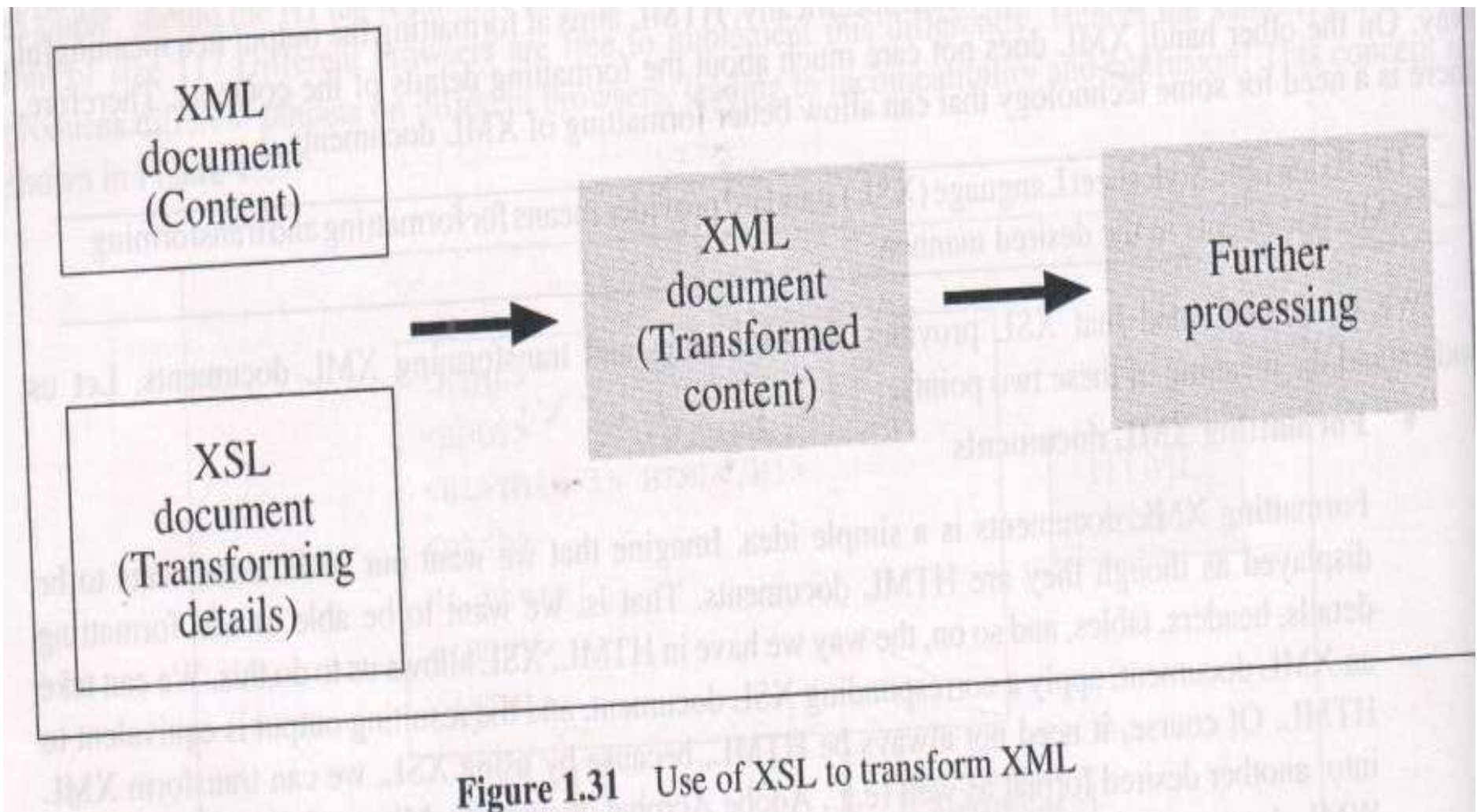


Figure 1.30 Use of XSL with XML to produce HTML

XML Standards - XSL



XML Standards - Schemas

- An XML schema provides rules for the contents of an XML document, pretty much like DTD.
- It is far more than DTD.

XML Standards - xQuery

- The XML query language, also called as Xquery, is still being developed.
- When completed, it is expected to provide a complete set of query language and operators to work with XML documents.

XML Standards - xLink

- Xlink defines a standard mechanism for creating hyperlinks in XML documents.
- Xlink is a language for creating hyperlinks in XML doc.
- Xlink is similar to HTML links.
- Any elements in an XML document can behave as an Xlink
- Xlink supports two types of links
- Simple link (like HTML)
- Extended links (for linking multiple resources together)

XML Standards - XPath

- XPath is used to refer to specific portions of an XML document using XSLT and Xpointer.
- The SQL of XML.

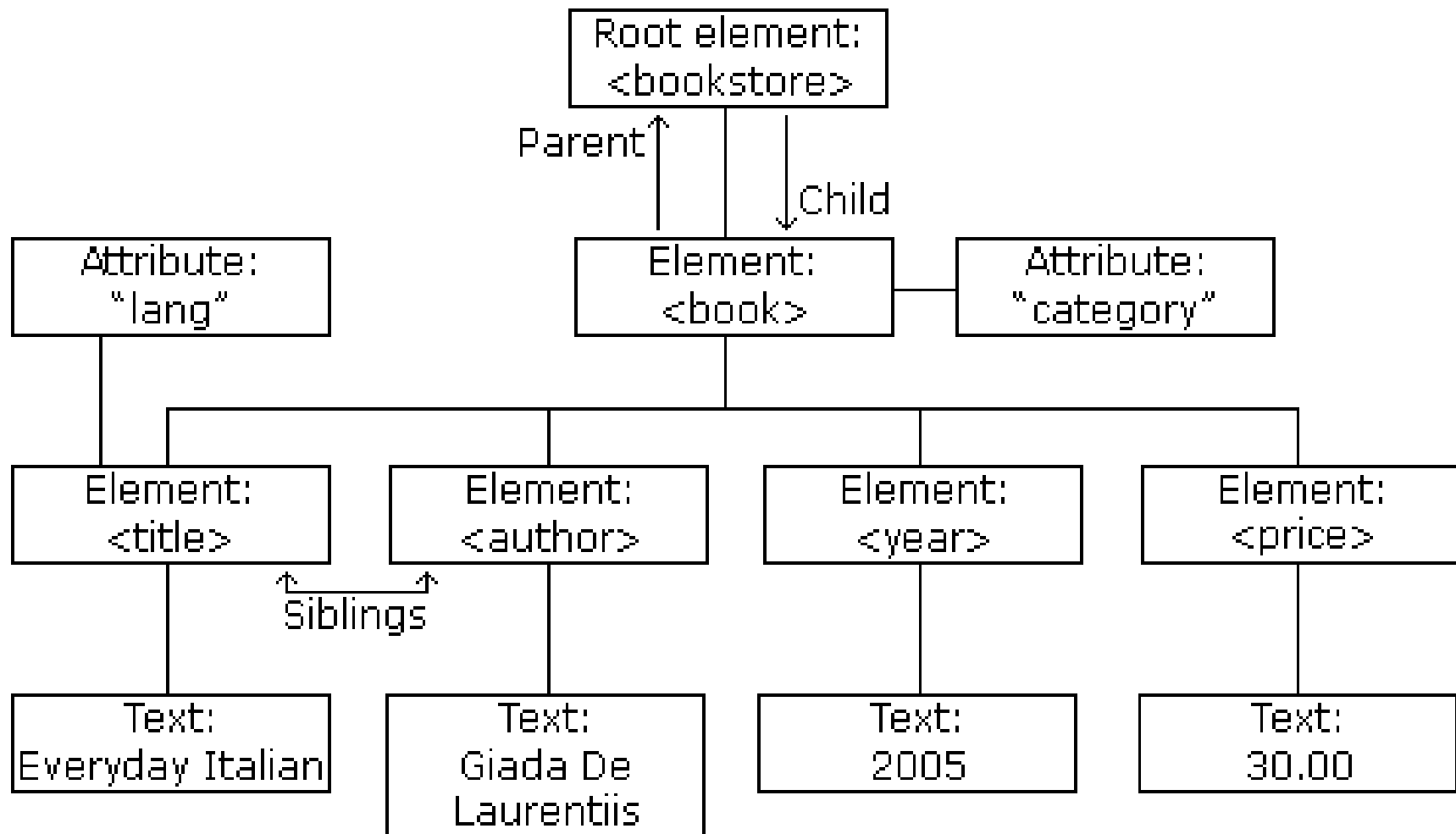
XML Standards – Digital Signature

- XML digital signature provide a means of message integrity and non- repudiation for XML documents.

XML Standards – XHTML

- HTML + XML = XHTML
- It is a revised version of HTML with rules from XML.

Basic Structure – A Simple XML Document



Basic Structure – A Simple XML Document

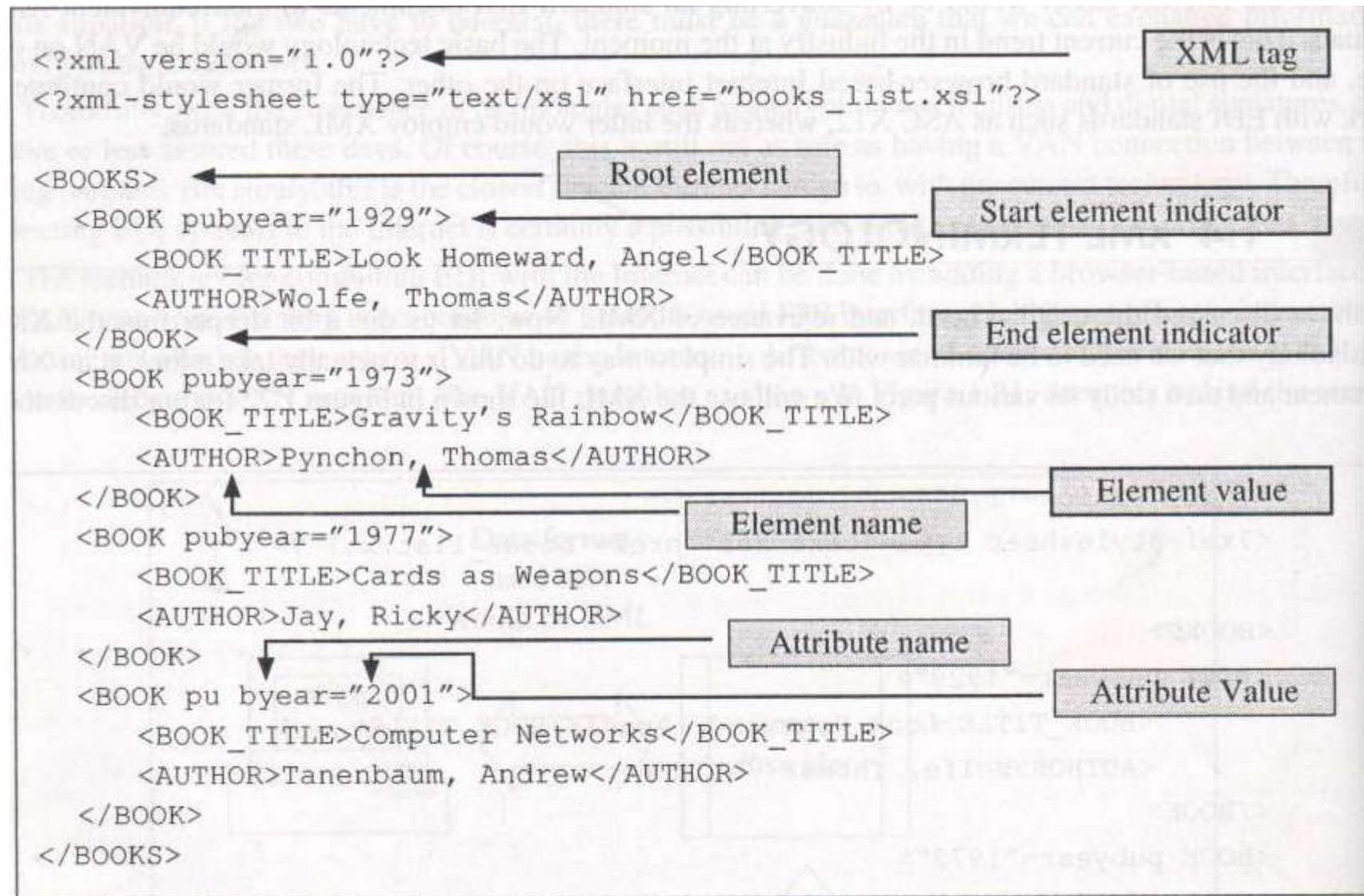


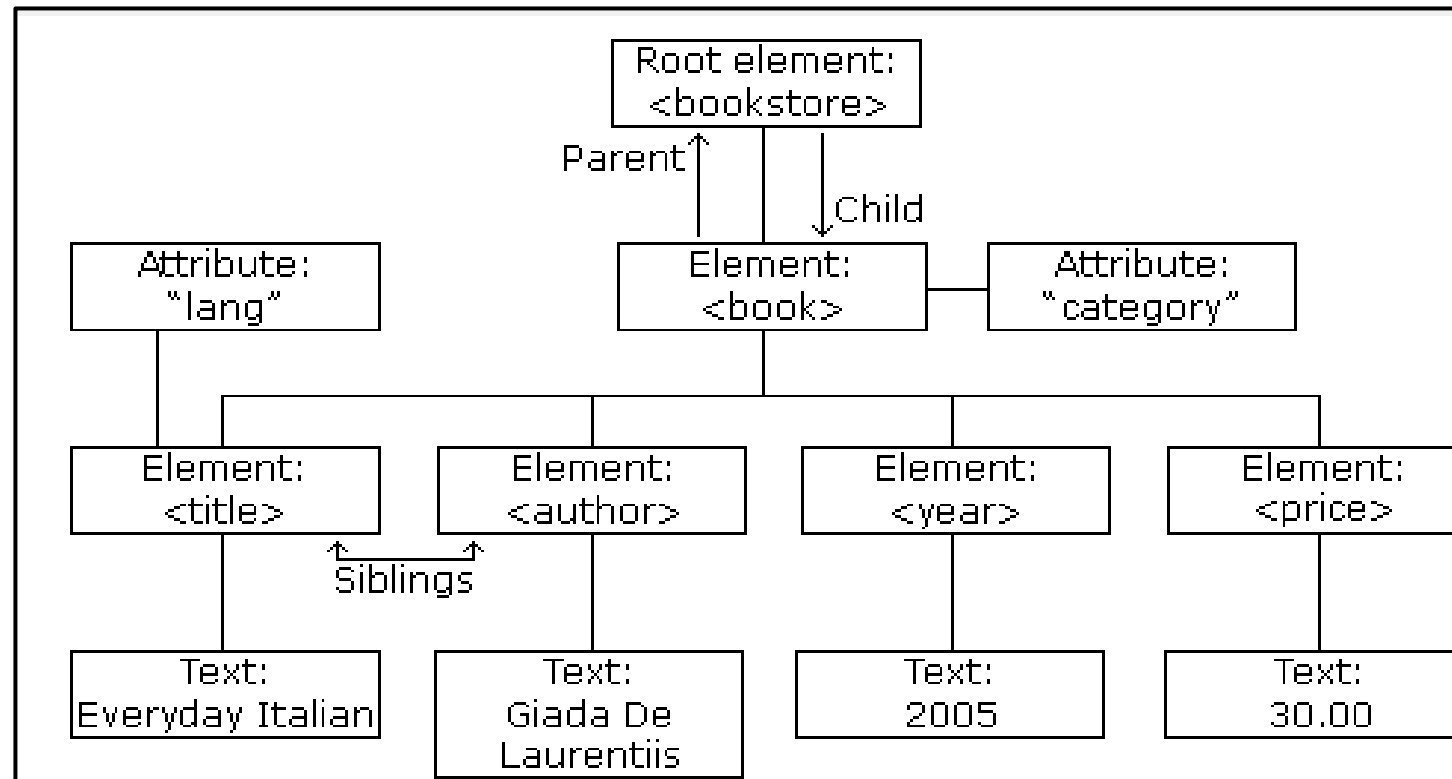
Figure 1.23 Terminology in XML – High-level overview

XML – As a tree structure

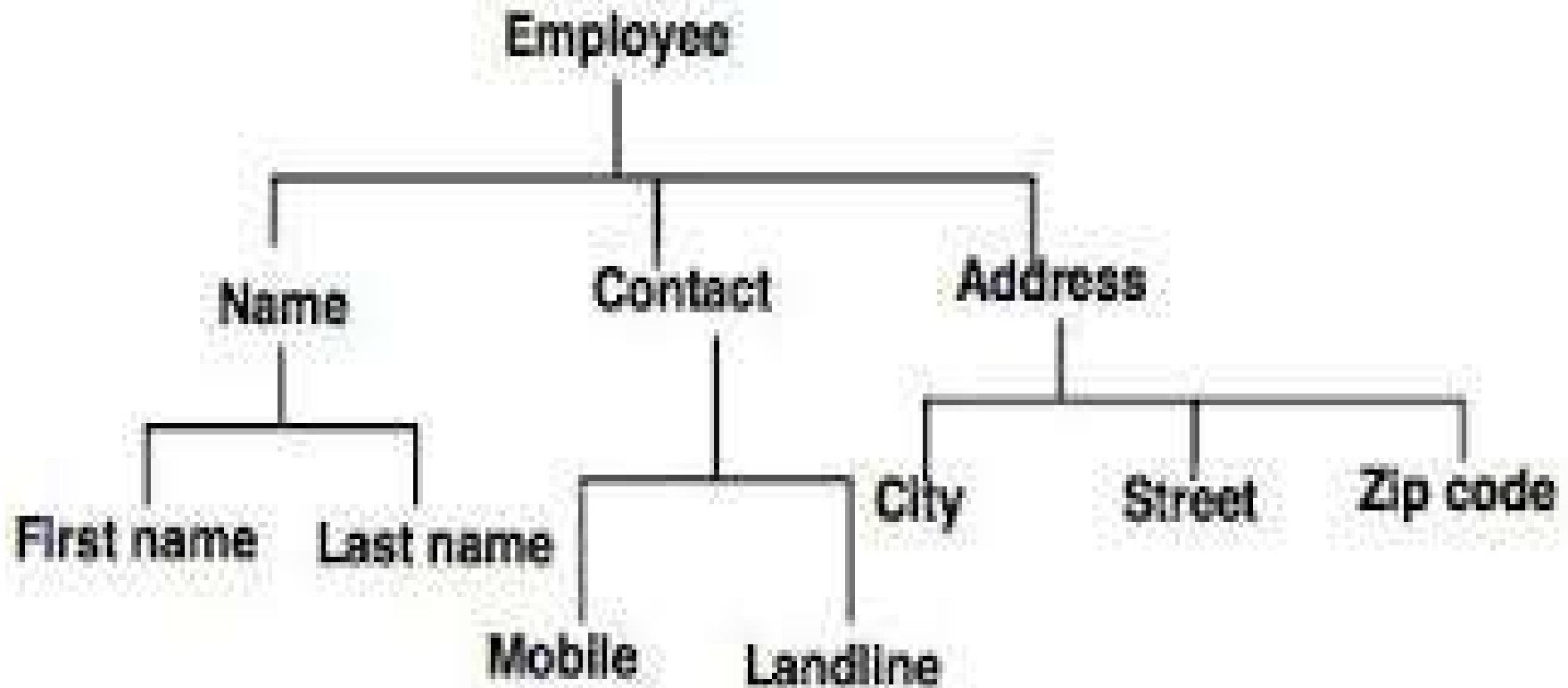
- All XML documents can be described as tree-structures. They all have exactly one top element.
- The top element can contain any number of attributes and simple elements and it can contain any number of complex elements, which again can contain sub-structures.
- When you process an existing XML document, you will often do this by traversing the tree-structure so that you traverse the entire structure, extracting information as you go through the elements and attributes that contain data.
- When you create an XML document, you will create the different elements and attributes that make up the final document and combine them into a tree-structure.
- To be able to do either of these things effortlessly, you need to understand how an XML document maps to a tree-structure.

Example XML

```
<bookstore>
  <book category="technology">
    <title lang="en">Intorduction to XML</title>
    <author>Atul Kahate</author>
    <year>2015</year>
    <price>300.00</price>
  </book>
</bookstore>
```

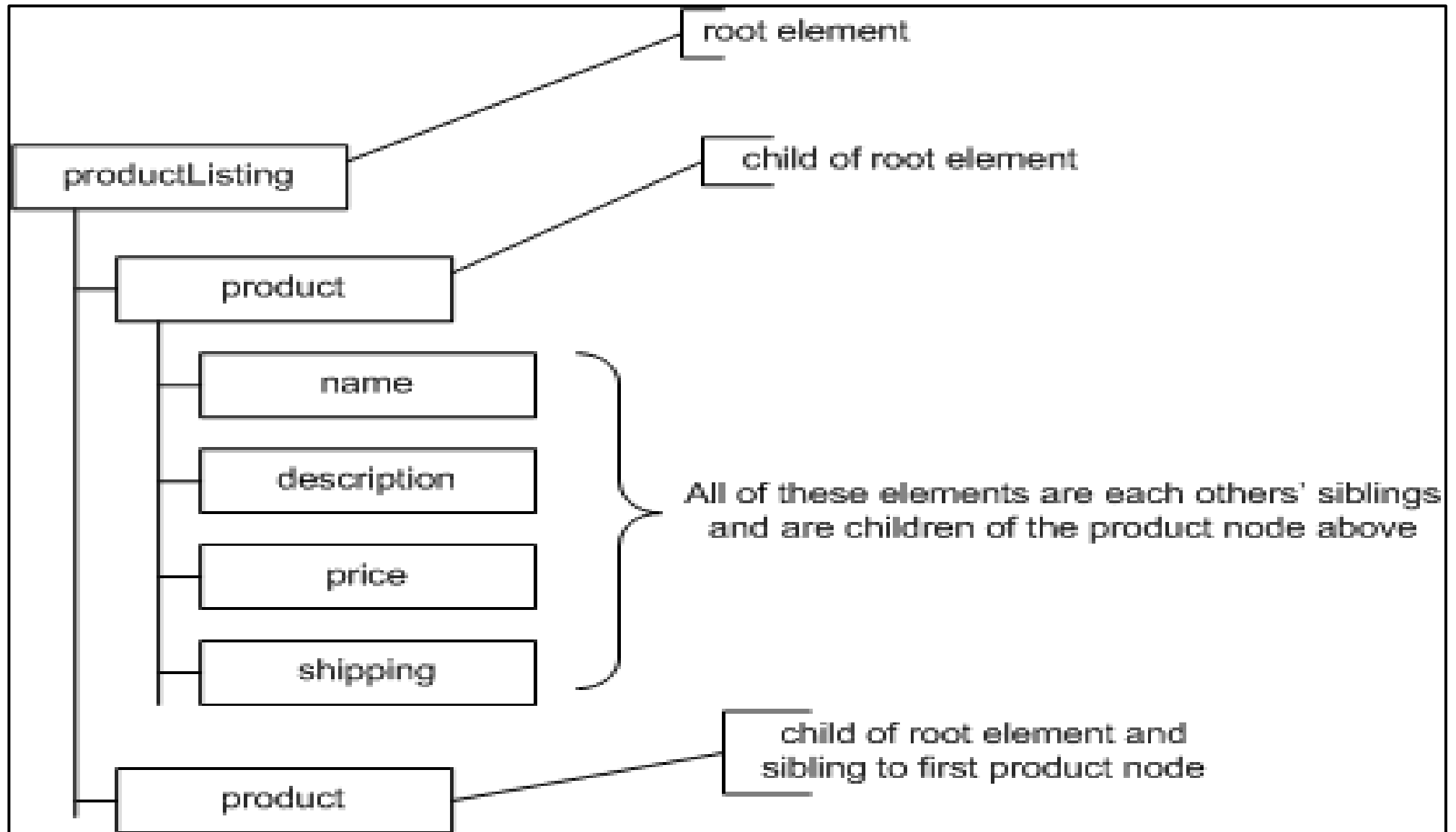


Basic Structure – A Simple XML Document



XML tree structure

Basic Structure – A Simple XML Document



The Idea of Markup

- Another Name for tags is markup. Hence, we also call XML as a markup language.
- **SGML** is the parent of almost all important modern markup language.

The Idea of Markup

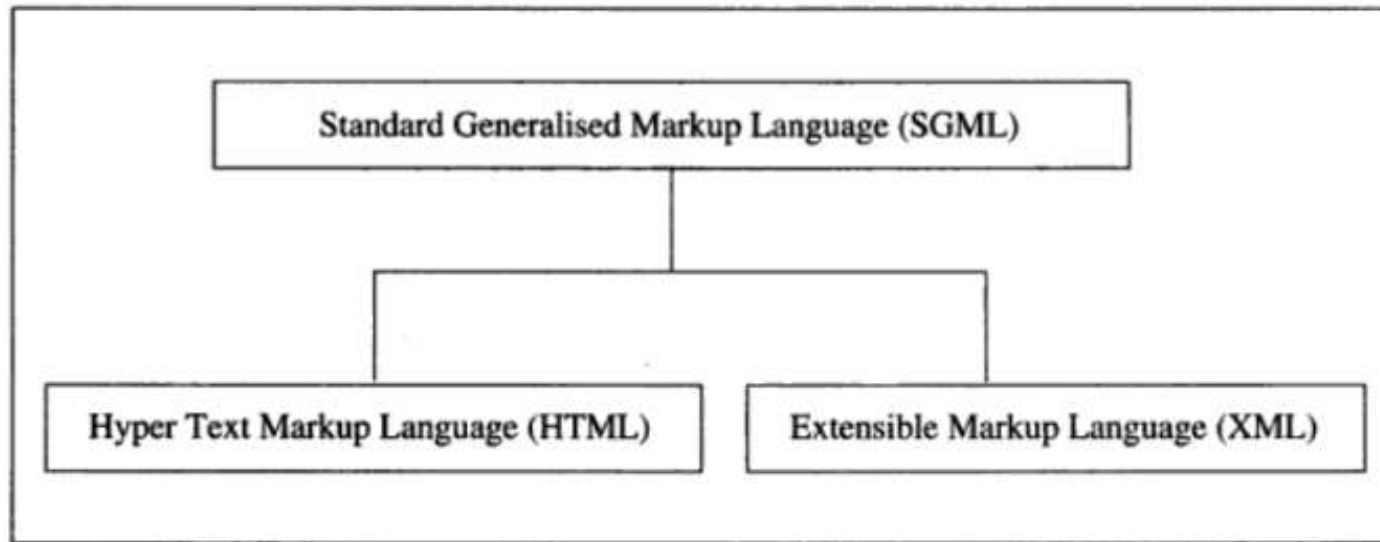


Figure 2.10 Positioning of SGML, HTML, and XML

What does the markup portion of XML mean to us, then? It brings the following features of value:

- Easy to read for humans
- Easy to use
- Easy to use for computers
- Easy to debug
- Easy to modify suitably for any industry or domain
- Works with all leading programming languages, databases, and formats such as spreadsheets and drawings

We have mentioned that XML is based on SGML.

SGML was created to provide a means for identifying the portions and content of a document, not by line numbers or the actual content, but by the type of the content.

Organising Information in XML

Process of organising information into an XML format in 3 steps.

Step 1. Classifying information as per its importance.

Step 2. Adding the details

Step 3. Transforming information into XML format.

a. Identifying elements

b. Identifying attributes

Creating Well Formed XML Documents

<?xml> Tag

- This tag identifies document as XML document.
- It must always be first statement in an XML document.
- It specifies XML specifications.
- Eg: <?xml version="1.0"? >

The Root element

- There must be one and only one root element in a document.
- All other contents are sub part of this root element.
- Root Element must always be the first

Creating Well Formed XML Documents

Opening and Closing of Tags

- The opening and closing tags signify position of element in XML document.
- All elements have an opening tag.
- Opening tag : `<>`, closing tag : `</>`

Empty elements

- Empty tags are `<>` and `</>` without content in between.
- Eg: `<BOOK_NAME>` `</BOOK_NAME>`
OR
- `<BOOK_NAME/>`

Creating Well Formed XML Documents

Entities

- Entity is text to be represented repeatedly without writing it everyday.
- There are three types of entities:
 - **Character Entities** – are special character codes that assign a different meaning to special symbols.
 - Eg: &

<hr/>	
<	<
>	>
&	&
"	"
'	'

Text Entities – allow to define data as entity. Used to associate large or repeated blocks of text with name and replace the text with the entity name.

Syntax: <! ENTITY name “value”>

Eg: <! ENTITY country “India”>

INPUT

```
<?xml version="1.0"?>
<!DOCTYPE college
[
<!ENTITY course "Integrated
MSc(IT)">
<!ENTITY sem "sem 3">
]>
<college>
<C>&course;</C>
<S>&sem;</S>
</college>
```

```
<college>
  <C>Integrated MSc(IT) </C>
  <S>sem 3 </S>
</college>
```

Binary Entities – Used to associate name with binary data (image or video) with the entity name.

Syntax: `<!ENTITY city SYSTEM "c.html" NDATA html>`

```
<?xml version="1.0"? >
<!DOCTYPE college
[
<!ENTITY city SYSTEM "c.html"
NDATA html>
]>
<college>
<C>&city;</C>

</college>
```

Element Naming and Conventions

Rules for Naming Elements:

- Should contain atleast one letter: a-z or A-Z
- Can start with an alphabet or an underscore
- Can cotain letters, digits, hyphens, underscores, full stops
- Elements are case sensitive
- Child elements must be nested completely inside parent element.
- Elements and sub elements cannot overlap each other.

Attributes Naming and Conventions

- Attributes allow us to specify more for elements.
- They are optional

- Syntax:

<ELEMENT attribute1, attribute 2,..., attribute n>

</ELEMENT>

- Example:

<BOOK author="Atul Kahate" edition="2">

</BOOK>

XML Naming Rules

- Element names are case-sensitive
- Element names must start with a letter or underscore
- Element names cannot start with the letters xml (or XML, or Xml, etc)
- Element names can contain letters, digits, hyphens, underscores, and periods
- Element names cannot contain spaces
- XML elements can be extended to carry more information.

Attribute values must always be quoted. Either single or double quotes can be used. Eg. `<person gender="female">` or `<person gender='female'>`

XML Comments

The syntax for comments in XML is exactly the same as for HTML. That is, the comments should be enclosed inside the tag boundaries `<!--` and `-->`.

Consider the example shown in Figure 2.17.

```
<?xml version="1.0"?>

<BOOK>

    <!-- This is a comment. It would be ignored by the browser.
    --> Computer Networks

</BOOK>
```

Difference between well-formed and valid" XML document

Well-Formed Document	Valid XML Document
Well-formed documents follow all the syntax rules of XML.	An XML document that conforms to DTD or an XML Schema is said to be valid XML document.
Not all well-formed XML documents are valid XML document.	A valid XML document is also well-formed.
Well-formed XML documents are designed for use without any constraints	Valid XML documents explicitly require constraint mechanisms
It does not support advanced feature	It support advanced features such as linking mechanisms, value and range bounding, and data typing

HTML vs XML

HTML vs XML

HTML	XML
HTML document formats and displays web page data.	XML document carry data along with their description.
Predefined tags (Predefined markup language).	Not have predefined tags.You can create and define new tags as per your needs.(Meta Language).
May not have closing tag.	Must have closing tag.
Not Case Sensitive.	Case Sensitive.
Directly Viewable in browser.	Viewable if proper Stylesheet provided.

UNIT 1

COMPLETED