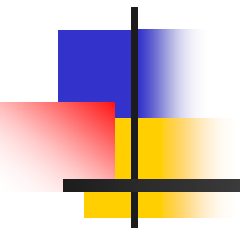
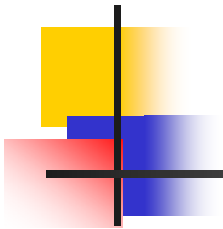
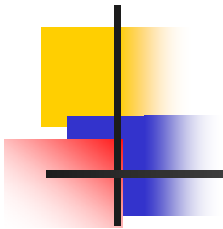


XML Basics



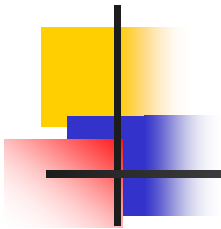
Lecture objectives

- To introduce the basic components of XML.



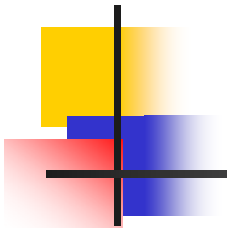
Lecture Outline

- Introduction
- The anatomy of XML document
- Components of XML document
- XML validation
- Rules for well-formed XML document
- XML DTD
- More XML components
- References
- Reading list



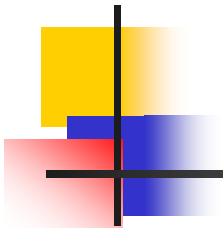
- Introduction

- What is XML
- How can XML be used
- What does XML look like
- XML and HTML
- XML is free and extensible



-- What is XML

- XML stands for Extensible Markup Language.
- XML developed by the World Wide Web Consortium (www.W3C.org)
- Created in 1996. The first specification was published in 1998 by the W3C
- It is specifically designed for delivering information over the internet.
- XML like HTML is a markup language, but unlike HTML it doesn't have predefined elements.
- You create your own elements and you assign them any name you like, hence the term extensible.
- HTML describes the presentation of the content, XML describes the content.
- You can use XML to describe virtually any type of document: Koran, works of Shakespeare, and others.
  - Go to http://www.ibiblio.org/boask to download



-- How can XML be Used?

- XML is used to Exchange Data
- With XML, data can be exchanged between incompatible systems
- With XML, financial information can be exchanged over the Internet
- XML can be used to Share Data
- XML can be used to Store Data
- XML can make your Data more Useful
- XML can be used to Create new Languages

-- What does XML look like

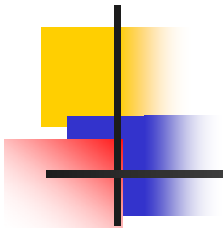
Book

Title	Author	year
Java	Mustafa	1995
Pascal	Ahmed	1980
Basic	Ali	1975
Oracle	Emad	1973
….	….	

Relation

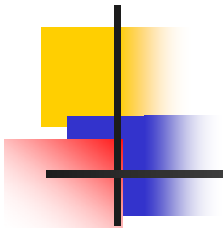
```
<Bibliography>

    <Book>
        <Title>   Java    </Title>
        <Author>  Mustafa  </Author>
        <Year>    1995     </year>
    </Book>
    …
    …
    …
    <Book>
        <Title>   Oracle  </Title>
        <Author>  Emad    </Author>
        <Year>    1973     </Year>
    </Book>
    ….
     ….
</ Bibliography>
```

XML document



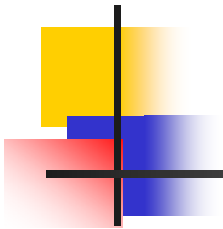
-- XML and HTML ...

- XML is not a replacement for HTML
- XML was designed to carry data
- XML and HTML were designed with different goals
  - XML was designed to describe data and to focus on what data is
  - HTML was designed to display data and to focus on how data looks.
- HTML is about displaying information, while XML is about describing information



... -- XML and HTML

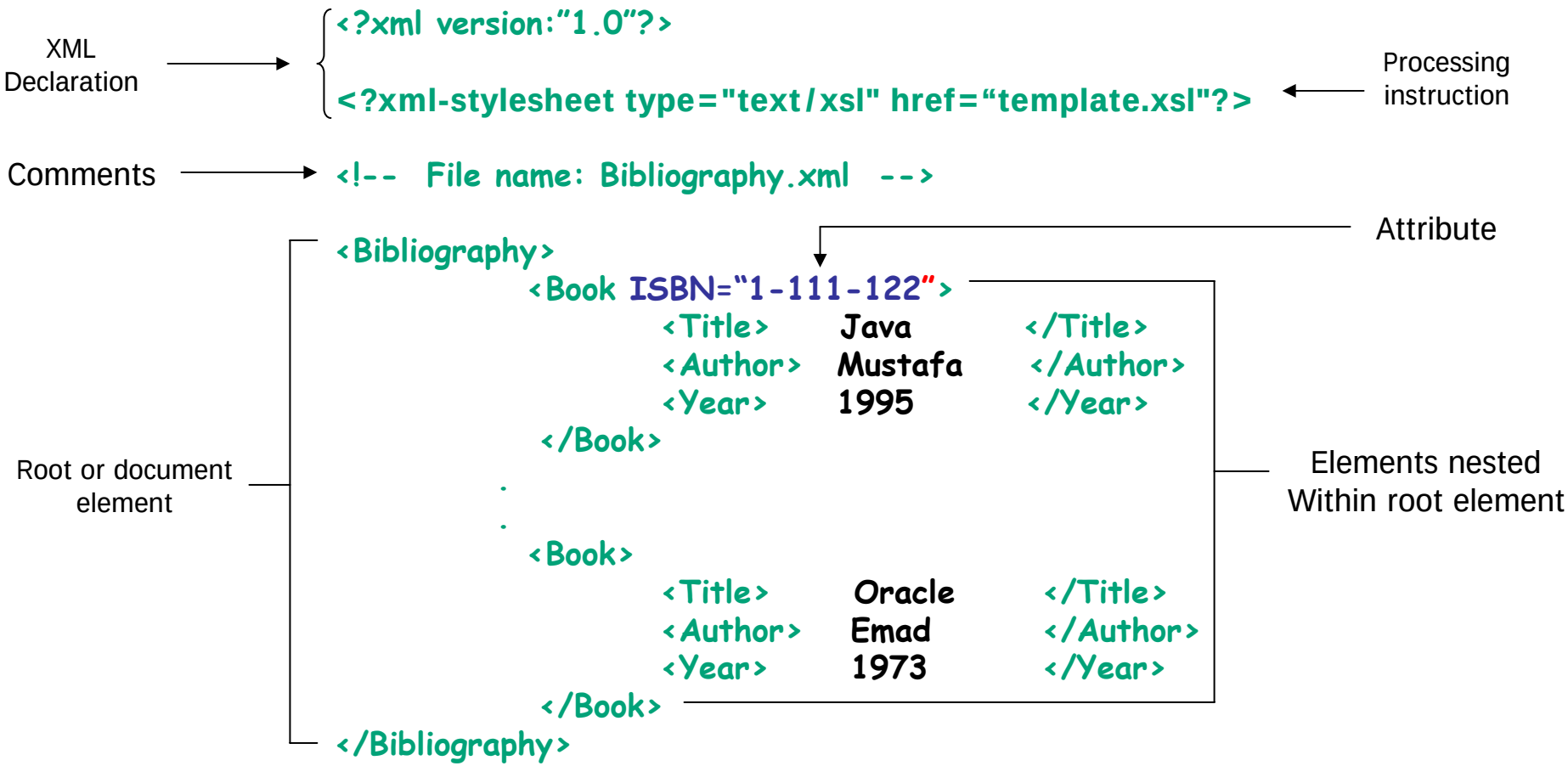
- HTML is for humans
  - HTML describes web pages
  - You don't want to see error messages about the web pages you visit
  - Browsers ignore and/or correct as many HTML errors as they can, so HTML is often sloppy
- XML is for computers
  - XML describes data
  - The rules are strict and errors are not allowed
    - In this way, XML is like a programming language
  - Current versions of most browsers can display XML

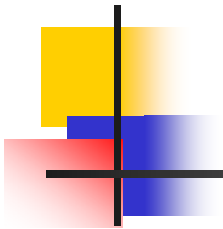


-- XML is free and extensible

- XML tags are not predefined
  - You must "invent" your own tags
  - The tags used to mark up HTML documents and the structure of HTML documents are predefined
  - The author of HTML documents can only use tags that are defined in the HTML standard
- XML allows the author to define his own tags and his own document structure, hence the term extensible.

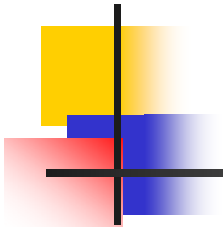
-The Anatomy of XML Document





- Components of an XML Document

- Elements
  - Each element has a beginning and ending tag
  - <TAG_NAME>…</TAG_NAME>
  - Elements can be empty (<TAG_NAME />)
- Attributes
  - Describes an element; e.g. data type, data range, etc.
  - Can only appear on beginning tag
    - Example: <Book **ISBN = "1-111-123**">
- Processing instructions
  - Encoding specification (Unicode by default)
  - Namespace declaration
  - Schema declaration

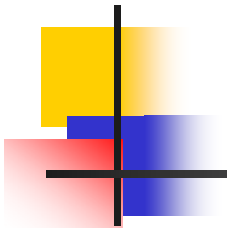


-- XML declaration

- The XML declaration looks like this:

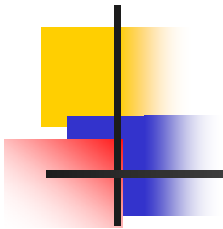
  ```
  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  ```

  - The XML declaration is not required by browsers, but is required by most XML processors (so include it!)
  - If present, the XML declaration must be first--not even white space should precede it
  - Note that the brackets are <? and ?>
  - version="1.0" is required (I am not sure it is the only version so far)
  - encoding can be "UTF-8" (ASCII) or "UTF-16" (Unicode), or something else, or it can be omitted
  - standalone tells whether there is a separate DTD



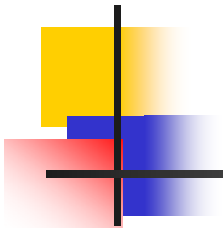
-- Processing Instructions

- PIs (Processing Instructions) may occur anywhere in the XML document (but usually in the beginning)
- A PI is a command to the program processing the XML document to handle it in a certain way
- XML documents are typically processed by more than one program
- Programs that do not recognize a given PI should just ignore it
- General format of a PI: <?target instructions?>
- Example: <?xml-stylesheet type="text/css" href="mySheet.css"?>



-- XML Elements

- An XML element is everything from the element's start tag to the element's end tag
- XML Elements are extensible and they have relationships
- XML Elements have simple naming rules
  - Names can contain letters, numbers, and other characters
  - Names must not start with a number or punctuation character
  - Names must not start with the letters xml (or XML or Xml ..)
  - Names cannot contain spaces

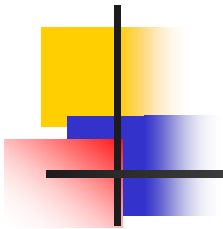


-- XML Attributes

- XML elements can have attributes
- Data can be stored in child elements or in attributes
- Should you avoid using attributes?
  - Here are some of the problems 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 Document Type Definition (DTD) - which is used to define the legal elements of an XML document

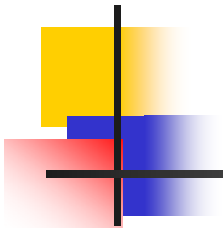
-- Distinction between subelement and attribute

- In the context of documents, attributes are part of markup, while subelement contents are part of the basic document contents
- In the context of data representation, the difference is unclear and may be confusing
  - Same information can be represented in two ways
    - <Book … Publisher = "McGraw Hill" … </Book>
    - <Book>
      …
      <Publisher> McGraw Hill </Publisher>
      …
      </Book>
- Suggestion: use attributes for identifiers of elements, and use subelements for contents



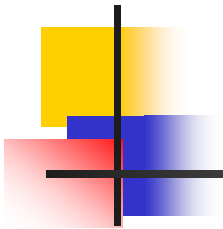
- XML Validation

- Well-Formed XML document:
  - Is an XML document with the correct basic syntax
- Valid XML document:
  - Must be well formed plus
  - Conforms to a predefined DTD or XML Schema.



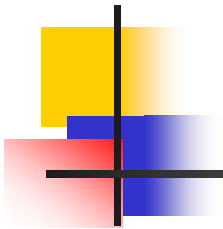
- Rules For Well-Formed XML

- Must begin with the XML declaration
- Must have one unique root element
- All start tags must match end-tags
- XML tags are case sensitive
- All elements must be closed
- All elements must be properly nested
- All attribute values must be quoted
- XML entities must be used for special characters



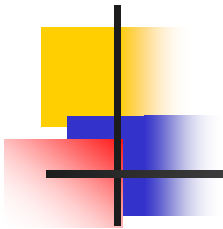
- XML DTD

- A DTD defines the legal elements of an XML document
  - defines the document structure with a list of legal elements and attributes
- XML Schema
  - XML Schema is an XML based alternative to DTD
- Errors in XML documents will stop the XML program
- XML Validators



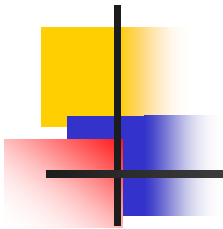
- More XML components

- Namespace
- Entities
- CDATA



-- Namespace

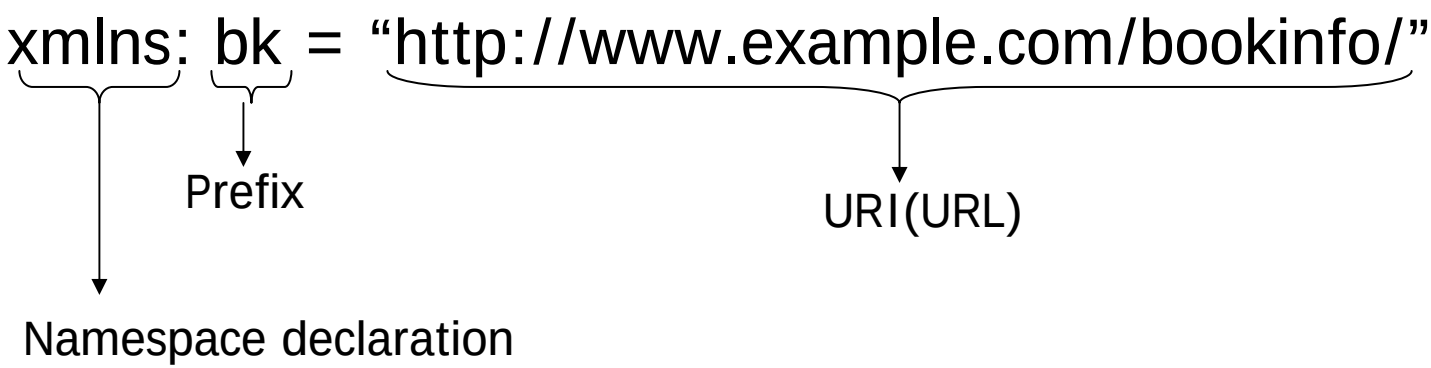
- Overview
- Declaration
- Default namespace
- Scope
- attribute



--- Namespaces: Overview

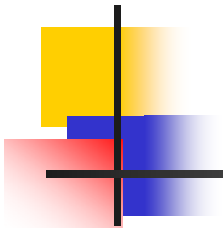
- Part of XML's extensibility
- Allow authors to differentiate between tags of the same name (using a prefix)
  - Frees author to focus on the data and decide how to best describe it
  - Allows multiple XML documents from multiple authors to be merged
- Identified by a URI (Uniform Resource Identifier)
  - When a URL is used, it does NOT have to represent a live server

--- Namespaces: Declaration



Example:

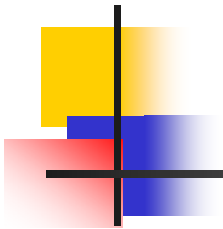
```
<BOOK xmlns:bk="http://www.bookstuff.org/bookinfo">
        <bk:TITLE> All About XML </bk:TITLE>
        <bk:AUTHOR> Joe Developer </bk:AUTHOR>
        <bk:PRICE currency='US Dollar'> 19.99 </bk:PRICE>
</BOOK>
```



--- Namespaces: Default Namespace

- An XML namespace declared without a prefix becomes the default namespace for all sub-elements
- All elements without a prefix will belong to the default namespace:

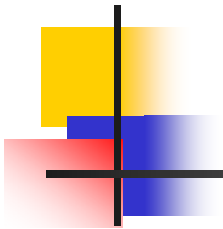
```
<BOOK xmlns="http://www.bookstuff.org/bookinfo">
    <TITLE> All About XML </TITLE>
    <AUTHOR> Joe Developer </AUTHOR>
</BOOK>
```



--- Namespaces: Scope

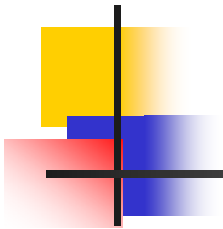
- Unqualified elements belong to the inner-most default namespace.
  - BOOK, TITLE, and AUTHOR belong to the default book namespace
  - PUBLISHER and NAME belong to the default publisher namespace

```
<BOOK xmlns="www.bookstuff.org/bookinfo">
    <TITLE> All About XML </TITLE>
    <AUTHOR> Joe Developer </AUTHOR>
    <PUBLISHER xmlns="urn:publishers:publinfo">
            <NAME> Microsoft Press </NAME>
    </PUBLISHER>
</BOOK>
```



--- Namespaces: Attributes

- Unqualified attributes do NOT belong to any namespace
  - Even if there is a default namespace
- This differs from elements, which belong to the default namespace

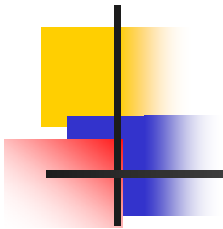


-- Entities

- Entities provide a mechanism for textual substitution,
- e.g.

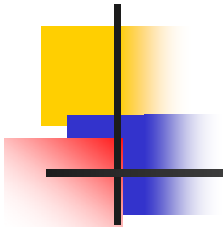
Entity	Substitution
&lt;	<
&amp;	&

- You can define your own entities
- Parsed entities can contain text and markup
- Unparsed entities can contain any data
  - JPEG photos, GIF files, movies, etc.



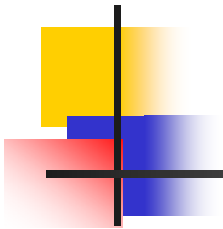
-- CDATA

- By default, all text inside an XML document is parsed
- You can force text to be treated as unparsed character data by enclosing it in <![CDATA[...]]>
- Any characters, even & and <, can occur inside a CDATA
- White space inside a CDATA is (usually) preserved
- The only real restriction is that the character sequence]]> cannot occur inside a CDATA
- CDATA is useful when your text has a lot of illegal characters (for example, if your XML document contains some HTML text)



- References

- W3 Schools XML Tutorial
  - http://www.w3schools.com/xml/default.asp
- W3C XML page
  - http://www.w3.org/XML/
- XML Tutorials
  - http://www.programmingtutorials.com/xml.aspx
- Online resource for markup language technologies
  - http://xml.coverpages.org/
- Several Online Presentations



- Reading List

- W3 Schools XML Tutorial
  - http://www.w3schools.com/xml/default.asp

END