XML

BASIS

- *Extensible Markup Language (XML)*
- It is a restricted form of the older *SGML*, a much more complex language.
- Original purpose of XML: to provide a universal format for the representation and sharing of structured data on the Web in a textual semi-formal format.
- Different application-specific concretizations of XML. E.g., a XML language for representing spreadsheets, another one for graphics, …

XML

Importance for enterprise programmers

- In the IT world, XML became a ubiquitous format for the storing of data as documents (not just on the web) and for the transfer of data over networks
- In enterprise computing, XML is mainly used for
  - Configuration files
  - Data transfer
- Platform-independent and based on international standards.
- Text-based format readable both for machines and for humans (well, theoretically…)

XML

Example XML document

```
<?xml version="1.0" encoding="UTF-8"?>
    <!--XML version and encoding (optional)-->
<bookstore>
   <book category="COOKING">
     <title lang="en">Everyday Italian</title>
     <author>Giada De Laurentiis</author>
     <year>2005</year>
     <price>30.00</price>
   </book>
   <book category="CHILDREN">
     <title lang="en">Harry Potter</title>
     <author>J K. Rowling</author>
     <year>2005</year>
     <price>29.99</price>
   </book>
</bookstore>
```

XML

Importance for enterprise programmers

- Human legibility is relative:

```
xmlns:wne="http://schemas.microsoft.com/office/word/2006/wordml"><w:body><w:p
w:rsidR="00000000" w:rsidRDefault="0060676B"><w:pPr><w:pStyle
w:val="Titel"/><w:jc w:val="left"/></w:pPr><w:r><w:t>Example (1): A simple
view for an Indefinite XML (IXML) document</w:t></w:r></w:p><w:p
w:rsidR="00000000" w:rsidRDefault="0060676B"><w:pPr><w:rPr><w:rFonts
w:ascii="Arial" w:hAnsi="Arial"/></w:rPr></w:pPr><w:r><w:rPr><w:rFonts
w:ascii="Arial" w:hAnsi="Arial"/></w:rPr><w:t>(the bold element ensemble
denotes the set of all persons with green or blue yes)</w:t></w:r></w:p><w:p
w:rsidR="00000000" w:rsidRDefault="0060676B"><w:pPr><w:rPr><w:rFonts
w:ascii="Arial" w:hAnsi="Arial"/></w:rPr></w:pPr></w:p><w:p w:rsidR="00000000"
w:rsidRDefault="0060676B"><w:pPr><w:rPr><w:rFonts w:ascii="Arial"
w:hAnsi="Arial"/></w:rPr></w:pPr><w:r><w:rPr><w:rFonts w:ascii="Arial"
w:hAnsi="Arial"/><w:noProof/></w:rPr><w:pict><v:shapetype id="_x0000_t202"
coordsize="21600,21600" o:spt="202" path="m,l,21600r21600,l21600,xe"><v:stroke
joinstyle="miter"/><v:path gradientshapeok="t"
o:connecttype="rect"/></v:shapetype><v:shape id="_x0000_s1028"
type="#_x0000_t202" style="position:absolute;margin-left:-6pt;margin-
top:113.1pt;width:65.25pt;height:19
```

XML

Importance for enterprise programmers

- All important programming languages such as Java and C++ support reading, processing and writing XML documents
- Lots of important enterprise-relevant technologies use XML, in particular for configuration files and for data transfer over the internet
- Also Web programming technologies like *Ajax* (= Asynchronous JavaScript + XML)
- XML also has shortcomings, such as its verbosity (→ JSON)
- Current trend: add relevant meta-information to Java source code directly in form of *annotations*. But still lots of XML around…

XML

Core properties of XML

- *Markup*: Adding meta-information to text or other markup

```
<markup>text</markup>
<markupX><markupY>text</markupX></markupY>
```

- XML is a language for the representation of *hierarchically structured* data (e.g., books, eMails, pictures, web sites, rathe simple databases…)
- An XML document is a *tree*
- XML documents consist mainly of so-called *elements*

XML

Core properties of XML

- An element consists of a *start-tag*, the *content* of the element, and an *end-tag*
- The start-tag has the form `<elementName>` and the end-tag has the form `</elementName>`, where `elementName` is the *name* or *type* of the respective element. Different elements can have the same name/type
- Optionally, an element can contain *attributes:*

  `<elementName attr1="value1" attr2="value2"> … </elementName>`
- In addition to elements and plain text, an XML document might contain `<!--comments-->` and declarations

XML

Core properties of XML

- The contents of an element can be…
  - other elements (i.e., elements can be nested):

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

  - plain text

```
…
<someElement>
  This is some text
</someElement>
```

  - both (so-called “*mixed content*”)

XML

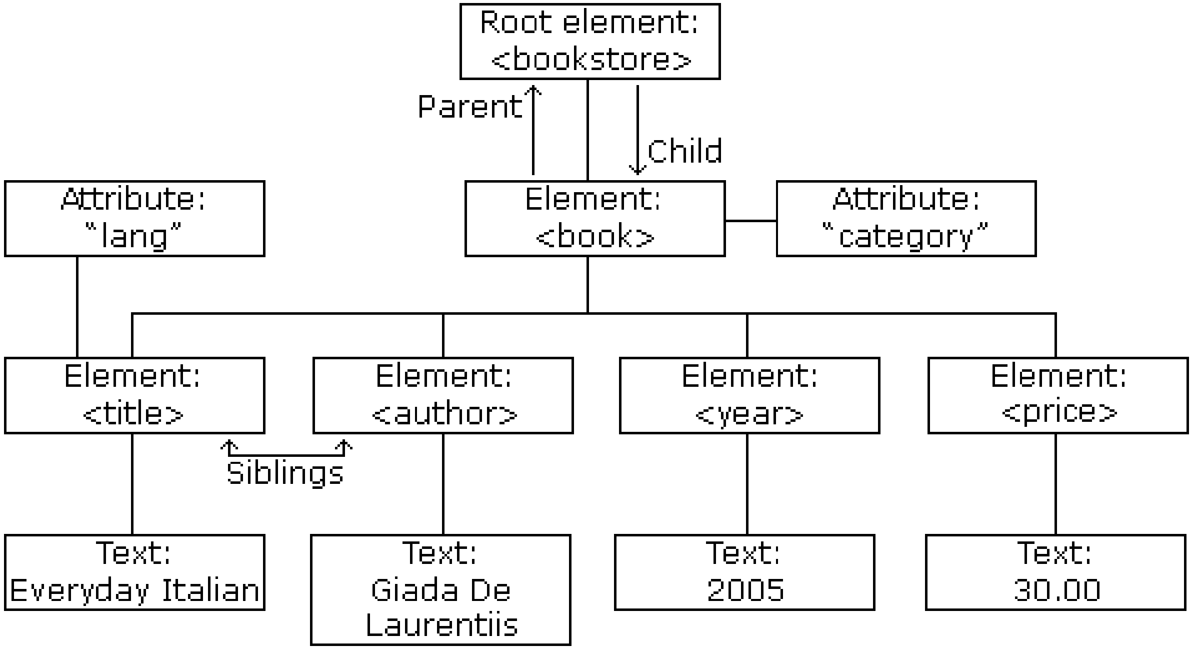
Core properties of XML

```xml
<?xml version="1.0" encoding="UTF-8"?>
     <!--XML version and encoding (optional)-->
<bookstore>
    <book category="COOKING">
      <title lang="en">Everyday Italian</title>
      <author>Giada De Laurentiis</author>
      <year>2005</year>
      <price>30.00</price>
    </book>
    <book category="CHILDREN">
      <title lang="en">Harry Potter</title>
      <author>J K. Rowling</author>
      <year>2005</year>
      <price>29.99</price>
    </book>
</bookstore>
```

XML

Core properties of XML

- There must be one *root element* (also called the *document element*) in each XML document, corresponding to the root node of the tree
- The other elements correspond to nodes (sub-trees, resp.)



XML

Core properties of XML

- The sequence of sibling-elements is important. E.g.,

  ```
  <parent>
     <child> 11111 </child>
     <child> 22222 </child>
  <parent>
  ```

  is not equivalent to

  ```
  <parent>
     <child> 22222 </child>
     <child> 11111 </child>
  <parent>
  ```

- Tags are case-sensitive: `<myElement>` ≠ `<myelement>`

XML

Core properties of XML

- Elements must be properly nested (i.e., hierarchically).
- E.g., the following is invalid:

```
<myElement>
     <mySubElement>
</myElement>
    </mySubElement>
```

- Elements can be empty. Abbreviated syntax: `<nothing/>`
- Comments are just like in HTML: `<!--` a comment `-->`

XML

Core properties of XML

- Elements may have *attributes* (in the start-tag). Attributes are used to add further information to elements. E.g.,

```
<appointment date="12/11/2007">
  …
</appointment>
```

- Elements can be duplicate, but not the attributes of a certain elements
- *Value* of an attribute: plain text in quotation marks

XML

Core properties of XML

- Special characters must be *escaped*
  - `&` as `&amp;`
  - `<` as `&lt;`
  - `>` as `&gt;`
  - `'` as `&apos;`
  - `"` as `&quot;`
- You can have element-like plain text content *ignored* using *CDATA* sections:

```
<![CDATA[
    <greeting>Hello, world!</greeting>
]]>
```

XML

Core properties of XML

- An XML document which observes the previously described constraints (plus a few minor things) is called *well-formed*
  - One root element
  - Closing end-tag for each element
  - No attribute appears more than once at the same start-tag
  - …
- Being well-formed is important to ensure that an XML document can be parsed properly

XML

Core properties of XML

- An XML document can be stored as a file (*XML file*), but it can also be stored in memory, e.g. as a Java object.
- In both cases, it is called "document"

XML

Core properties of XML

- XML syntax looks quite like HTML syntax. But there are certain differences:
  - XML is case sensitive. `<head>` is not the same as `<Head>`
  - Tags need to be *balanced*: For each start-tag `<element>`, there needs to be a corresponding end-tag `</element>`
  - Attribute values need to be enclosed in quotation marks
  - All attributes must have values. `<myElement myAttr>` is not allowed
  - Whitespace is preserved!
- The designated HTML successor *XHTML* observes these restrictions

XML

DTDs and XSDs

- You can restrict what is allowed inside an XML document using a *DTD (Document Type Definition)* or a *XSD (XML Schema Definition)*
- So-called *schema(s)* (not:“scheme(s)”) which specify a class of XML documents.
- A schema contains rules for elements and attributes which these XML documents need to observe. It specifies the syntax of these XML documents.
- Usually, XML documents come with a DTD or XSD which is typically provided as a separate document (shared by multiple XML documents of the same "type")
- DTD: pretty old, XSD: newer + more powerful