Universität Freiburg
Institut für Informatik
Prof. Dr. Peter Fischer
Lecuture XML and Databases
Winter Semester 2011/12

11.11.2011

Solution Sheet 2

XML and DTD validation

Exercise 1: Validation of XML with a DTD

1.1.

- title and xml are not allowed as children of movies
- the id may not begin with a digit
- comment may not have an attribute

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE movies [
<!ELEMENT movies (Movie+) >
<!ELEMENT Movie ( title, year, _director, (comment | newcomment)+)>
<!ATTLIST Movie id ID #REQUIRED>
<!ELEMENT title (#PCDATA)>
<!ELEMENT year (#PCDATA) >
<!ELEMENT _director (#PCDATA)>
<!ATTLIST _director name CDATA #IMPLIED>
<!ELEMENT comment (#PCDATA)>
<!ELEMENT newcomment (#PCDATA)>
<!ATTLIST comment lang CDATA #IMPLIED>
]>
<movies>
    <Movie id="y56225">
        <title>Love Story</title>
        <year>1980</year>
        <_director name="Coppola"/>
        <comment lang=""/>
        <newcomment >Oscar</newcomment>
        <comment lang="de">1980 Warner Bros.</comment>
        <!-- Famous movie of the 80s -->
    </Movie>
</movies>
```

1.2.

PCDATA means "parsed character data". It means that this character data is to be parsed. In particular:

- Entity references (&lt; &gt; &apos; &quot; and &amp;) will be resolved (to < > ' " and & respectively), as well as any additional entities defined in the DTD.
- It may not contain any unencoded < or & characters, because they would be confused with an opening tag or an entity reference.

In the DTD, PCDATA is used to say that an element may only contain parsed character data, without child elements.

CDATA is more confusing, because it can have several meanings in the XML world:

- In the DTD, it is used to give the most general type for an attribute: an attribute of type CDATA may contain any attribute value. Note, however, that entity references **are** resolved, and that & and < must be encoded as well. In addition, the single quote ' must be encoded to &apos; if the attribute value is single-quoted, and the double quote " must be encoded to &quot; if the attribute value of double-quoted.

- Where PCDATA is expected in an element, one can explicitely use a CDATA construct to **escape** the special XML characters like < or >, which will not be recognized as markup. The only sequence recognized as markup in a CDATA section is]]>, which is interpreted as the end of the CDATA section.

```
<![CDATA[
if (a<2) { // notice the use of < without needing to encode it
as &lt;
  Writeline("The number is too low");
}
]]>
```

1.3.
This is a big debate between programmers.
- Some say that attributes are for metadata whereas elements contain information
- In general, an element is better if there is a lot of data inside
- One has to use an element if one wants to nest children
- Attributes are in a set (so two attributes of an element may not have the same name), whereas two sibling elements may have the same name.

Exercise 2: Namespaces

2.1.
- eth had a wrong closing tag, so had Rektor
- a / was missing for the president tag
- the DOCTYPE should be introduced with <! and the root element should not be quoted.

```
<?xml version="1.0"?>
<!DOCTYPE eth SYSTEM "eth.dtd">
<eth xmlns="http://www.ethz.ch"
    xmlns:xmldb="http://www.dbis.ethz.ch"
    date="11.11.2006"
    xmldb:date="12.11.2006">
    <date>13.11.2006</date>
    <president number="1">Empty</president>
    <Rektor>Name 2</Rektor>
</eth>
```

2.2.

```
<eth xmlns="http://www.ethz.ch"
```

eth is in the namespace http://www.ethz.ch

```
     xmlns:xmldb="http://www.dbis.ethz.ch"
     date="11.11.2006"
```

data does not inherit the namespace from the root element, so it is in no namespace.
Unlike elements, an attribute with no prefix is in no namespace, even if there is a default namespace.

```
     xmldb:date="12.11.2006">
```

this attribute is in the http://www.dbis.ethz.ch namespace - it is allowed to have two attributes with the same local name if their namespaces are different

```
  <date>13.11.2006</date>
  <president number="1">Empty</president>
```

the number attribute is in no namespace

```
  <Rektor>Name 2</Rektor>
```

All children elements are in the namespace of the root, i.e. http://www.ethz.ch

```
</eth>
```

2.3.

DTDs are not aware of namespaces. They see prefix bindings as normal attributes. This means that the bindings in the document have to be explicitely declared in the DTD (as #FIXED attribute values).

The following DTD could be used:

```
<!ELEMENT eth (date, president, Rektor)>
<!ATTLIST eth xmlns CDATA #FIXED "http://www.ethz.ch"
              xmlns:xmldb CDATA #FIXED
                                   "http://www.dbis.ethz.ch"
              date CDATA #IMPLIED
              xmldb:date CDATA #IMPLIED>
<!ELEMENT date (#PCDATA)>
<!ELEMENT president (#PCDATA)>
<!ATTLIST president number CDATA #IMPLIED>
<!ELEMENT Rektor (#PCDATA)>
```

2.4.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<html xmlns="http://www.ethz.ch">
    <head>
        <title>Untitled</title>
    </head>
    <body>
        <div> Dear jane <br/>
            <p>You are invited at the weekly meeting</p>
            <p>Yours sincerely, <br/> John</p>
        </div>
    </body>
</html>
```

First, you can check that oXygen tells you this document is well-formed.

To validate a document, you need a DTD or a schema. Modifying the namespace also modifies the expanded name of the root element: this is no longer XHTML. As we did not define any schema for the namespace "http://www.ethz.ch", one would expect an error message about a missing DTD or schema for this namespace.
But remember: DTDs are not namespace-aware. oXygen knows about an XHTML DTD and it recognizes the html tag. In "DTD validation mode", only the tag name "html" matters, regardless of the xmlns declaration which is handled like any other attribute.

If you look at the XHTML DTD used by oXygen (probably frameworks/xhtml/dtd /xmtml1-strict.dtd in oXygen's directory), everything becomes clear at line 236:

```
<!ELEMENT html (head, body)>
<!ATTLIST html
  %i18n;
  id     ID     #IMPLIED
  xmlns  %URI;  #FIXED   'http://www.w3.org/1999/xhtml'
  >
```

This DTD defines a fixed attribute value for the attribute xmlns in the html tag. This is the reason why oXygen wants you to use this value.

You can tell oXygen to forget about the XHTML DTD by going to the preferences, in "Document Type Association" and unchecking the XHTML document type. If you now try to validate the document, you will get the expected error message (no DTD or schema to validate against). You are now free to define a new schema or a new DTD for your document.