



Universität Freiburg
Institut für Informatik
Prof. Dr. G. Lausen
Alexander Schätzle
Martin Przyjaciel-Zablocki

Georges-Köhler Allee, Geb. 51
D-79110 Freiburg
lausen@informatik.uni-freiburg.de
schaetzle@informatik.uni-freiburg.de
zablocki@informatik.uni-freiburg.de

Data Models and Query Languages Summerterm 2013

4. Exercise Sheet: XML, DTD & XPath

Discussion: 07.06.2013

Exercise 1 (XML)

Explain the difference between *well-formed* and *valid* XML documents.

Exercise 2 (DTD)

Given the following DTD D_1 (from the XQuery Use Cases).

```
<!ELEMENT bib      (book*)>
<!ELEMENT book     (title, (author+ | editor+ ), publisher, price)>
<!ATTLIST book     year CDATA #REQUIRED >
<!ELEMENT author   (last, first)>
<!ELEMENT editor   (last, first, affiliation)>
<!ELEMENT title    (#PCDATA)>      <!ELEMENT last      (#PCDATA)>
<!ELEMENT first    (#PCDATA)>      <!ELEMENT affiliation (#PCDATA)>
<!ELEMENT publisher (#PCDATA)>      <!ELEMENT price     (#PCDATA)>
```

- Provide the smallest XML document that is valid according to D_1 and contains at least `bib` nodes.
- Provide the smallest XML document that is valid according to D_1 and contains at least `book` nodes.
- Provide the smallest XML document that is valid according to D_1 and contains at least `bib`, `author` and `editor` nodes.

Size is measured in terms of characters, excluding whitespaces. Check your result with the tool `xmllint`¹ that is part of `libxml2`. To check if an XML file `x.xml` is well-formed, use `xmllint -format x.xml`. With `xmllint -dtdvalid d.dtd x.xml` you can validate the file against the DTD `d.dtd`.

- The rule for `book` nodes is now changed. We get DTD D_2 with

```
<!ELEMENT book     (title, author*, price, publisher, editor*)>
```

Give a XPath query that can be used to distinguish XML documents that are valid according to D_1 and D_2 , respectively, and contain at least one `book` node. That means, the query should return an empty result for documents valid according to D_1 and a non-empty result for documents valid according to D_2 or vice versa.

¹<http://www.xmlsoft.org/downloads.html>

Exercise 3 (XPath)

Write the unabbreviated versions of the following XPath expressions.

- a) `./@foo`
- b) `/foo/bar/../../baz[7]`

Exercise 4 (XPath)

Consider the following XPath query where `n` is an element name.

`//n[parent::n and child::n]`

- a) Are the following XPath queries equivalent to the given query? If not, give an XML document where the query results are different.
 - (a) `//n[ancestor::n and child::n]`
 - (b) `//n[child::n/child::n]/child::n`
 - (c) `//n[preceding::n and parent::*]/child::n`
- b) Give an equivalent XPath query that neither uses the parent axis nor its abbreviation `..`.

Exercise 5 (XPath)

Consider the XML document *mondial.xml*² and the corresponding DTD *mondial.dtd*³.

- a) Is the document valid according to the DTD? If not, give an updated DTD such that it is valid.
- b) Formulate the following queries in XPath. You can check your solutions with the online demo of the Zorba XQuery engine⁴.
 - (a) Which are the names of the countries that have more than 10 million inhabitants and an area less than 200.000 km²?
 - (b) Which are the names of the countries that are smaller (area) than the Netherlands?
 - (c) Which are the names of the countries that have a border to Germany and higher population growth than Germany?
 - (d) Which are the names of the cities that are located at a lake?
 - (e) Return the "leaf nodes" of Germany, i.e. all elements within the country element of Germany that have no child elements.

²<http://tinyurl.com/mondial-xml>

³<http://tinyurl.com/mondial-dtd>

⁴<http://www.zorba-xquery.com/html/demo>