Fachbereich Mathematik und Informatik
Prof. Dr. K. Ostermann
Paolo G. Giarrusso, pgiarrusso@informatik

Philipps Universität Marburg

Nov 16, 2011

Web Technologies
Assignment 3

Please hand in your homework by email to `pgiarrusso@informatik`. The deadline for this assignment is Nov 21.

If at all possible, please bring your laptop to the following exercises.

G3.1 XPath

1. Find your own examples of XPath expressions violating the algebraic axioms from slide 39, i.e. reflexivity, symmetry and transitivity for equivalence relation, and anti-symmetry for orderings.

2. Consider the Recipe XML document from the lecture or from IXWT `http://www.brics.dk/ixwt/examples/recipes.xml`, and assume the document's root node is the current context node. Apply to it the XPath expressions `//rcp:ingredient[1]` and `/descendant/rcp:ingredient[1]`. Do they produce the same result? Try explaining why in terms of their desugarings.

3. Consider again the Recipe XML document and do (the corrected version of) exercise 3.6 of IXWT, that is, what are the possible results of evaluating the following XPath expressions?

   (a) `//(rcp:ingredient)[40] eq //(rcp:ingredient)[53]`

   (b) `//(rcp:ingredient)[40] = //(rcp:ingredient)[53]`

   (c) `//(rcp:ingredient)[40] is //(rcp:ingredient)[53]`

   Note: XPath Explorer has a bug, which means that to reference the intended nodes one needs to use as positions, instead of 40 and 53, 52 and 56.

4. Consider again the Recipe XML document. State XPath expressions selecting the following nodes:

   (a) all recipes with more than 10 preparation steps

   (b) all recipes containing milk and sugar

   (c) the number of eggs being used in recipes that also use milk

please turn the page!

G3.2 Schema Languages

1. Given the following Document Type Definition, write a valid document using all declared elements at least once (IXWT 4.3).

```
<!ELEMENT store (division,location?,product*)>

<!ELEMENT division EMPTY>
<!ATTLIST division code CDATA #REQUIRED>

<!ELEMENT location (#PCDATA)>
<!ATTLIST location sector (s1|s2|s3) #IMPLIED>

<!ELEMENT product (name,description,quantity)>
<!ATTLIST product code CDATA #REQUIRED
                  id   ID     #REQUIRED>

<!ELEMENT name (#PCDATA)>
<!ELEMENT description (#PCDATA|br|em)*>
<!ELEMENT quantity (#PCDATA)>

<!ELEMENT br EMPTY>
<!ELEMENT em (#PCDATA|br|em)*>
```

2. Given the following XML Schema, write a valid document using all declared elements at least once (IXWT 4.3).

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:m="http://movies.example"
        targetNamespace="http://movies.example"
        elementFormDefault="qualified">

  <element name="movies">
    <complexType>
      <sequence>
        <element ref="m:movie" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="m:person" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
    </complexType>
    <unique name="movies-unique">
      <selector xpath="m:movie"/>
      <field xpath="m:title"/>
      <field xpath="m:year"/>
    </unique>
    <key name="cast-key">
      <selector xpath="m:person|m:star"/>
      <field xpath="@id"/>
    </key>
    <keyref refer="m:cast-key" name="cast-keyref">
      <selector xpath=".//m:role"/>
      <field xpath="@ref"/>
    </keyref>
  </element>

  <element name="movie">
    <complexType>
      <sequence>
        <element name="title" type="string"/>
        <element name="length" type="nonNegativeInteger"/>
        <element name="year" type="gYear"/>
        <element name="cast">
          <complexType>
            <sequence maxOccurs="unbounded">
```

```
              <element name="role">
                <complexType mixed="true">
                  <attribute name="ref" type="NCName"/>
                </complexType>
              </element>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </complexType>
  </element>

  <element name="person" type="m:personType"/>

  <complexType name="personType">
    <sequence>
      <element name="name" type="string"/>
      <element name="birth" type="date"/>
    </sequence>
    <attribute name="id" type="NCName"/>
  </complexType>

  <element name="star" substitutionGroup="m:person">
    <complexType>
      <complexContent>
        <extension base="m:personType">
          <sequence maxOccurs="unbounded">
            <element name="award" type="string"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
  </element>

</schema>
```

H3.1 XML for Library Management

1. Write a DTD specifying your XML library management language (leave out the description column for now). Is your hand-in from last week valid?

2. Write an XML Schema specifying your XML library management language (leave out the description column for now). Is your hand-in from last week valid?

3. Compare the two schema languages using the specifications you just wrote. Among other criteria, consider the difficulty of extending your XML library management language with the XHTML description column again.