

Markup-Sprachen und semi-strukturierte Daten, SS 2008
 Übungsblatt 6

Abgabe: bis Do 05.06.2008 14 Uhr

Besprechung am Do/Fr 05./06.06.2008

Hinweis: Auf den Rechnern des Instituts können Sie mit dem Kommando `xsltproc stylesheet.xml daten.xml` ein Stylesheet in der Datei `stylesheet.xml` auf die XML Datei `daten.xml` anwenden. Auch XPath-Ausdrücke lassen sich mit `xsltproc` testen, indem man sie in ein Template einbindet.

Aufgabe 6-1 schriftlich bearbeiten
Viele (X)Pfade führen zum Ziel

Das folgende XML-Dokument repräsentiert einen Ausschnitt aus dem biblischen Stammbaum:

```
<?xml version="1.0" ?>
<mann><name> Abraham </name>
    <mann><name> Isaak </name>
        <mann><name> Jakob </name>
            <mann><name> Dan </name></mann>
            <mann><name> Naphthali </name></mann>
            <mann><name> Benjamin </name></mann>
        </mann>
        <mann><name> Esau </name></mann>
    </mann>
    <mann><name> Ismael </name>
        <mann><name> Nebajoth </name></mann>
        <mann><name> Kedar </name></mann>
        <mann><name> Kedma </name></mann>
    </mann>
    <mann><name> Jokschan </name>
        <mann><name> Saba </name></mann>
        <mann><name> Dedon </name></mann>
    </mann>
    <mann><name> Simran </name></mann>
    <mann><name> Nedan </name></mann>
    <mann><name> Midian </name>
        <mann><name> Epha </name></mann>
        <mann><name> Ephher </name></mann>
        <mann><name> Eldaa </name></mann>
    </mann>
    <mann><name> Jischbak </name></mann>
    <mann><name> Schuach </name></mann>
</mann>
```

Geben Sie 10 wesentlich verschiedene XPath-Ausdrücke an, deren Ergebnis die Menge der beiden `<mann>`-Elemente ist, die Saba und Dedon repräsentieren.

Aufgabe 6-2 schriftlich bearbeiten
Tabelle mit XSLT erzeugen

Gegeben Sei ein Dokument mit der folgenden DTD (vgl. Aufgabe 4-3):

```
<!ELEMENT tabelle      ( Student* ) >
<!ELEMENT Student      ( MatrNr, Name, Studienfach ) >
<!ELEMENT MatrNr       ( #PCDATA ) >
<!ELEMENT Name         ( #PCDATA ) >
<!ELEMENT Studienfach  ( #PCDATA ) >
```

- Schreiben Sie ein XSLT-Stylesheet, das die (Informationen über) Studenten aus dem Dokument in eine XHTML-Tabelle umwandelt. Die Tabelle soll Überschriften "MatrNr", "Name", "Studienfach" haben.
- Schreiben Sie ein XSLT-Stylesheet, das ebenfalls die Informationen in eine XHTML-Tabelle umwandelt, wobei diesmal jedoch nur solche Studenten in der Tabelle erscheinen sollen, die das Studienfach "Informatik" haben.

Aufgabe 6-3 schriftlich bearbeiten
XSLT (abc)

```
<?xml version="1.0" ?>
<xsl:stylesheet version="1.0 "
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match='//a'>
    <bc>
        <b><xsl:apply-templates/></b>
        <xsl:apply-templates select='//b' />
    </bc>
</xsl:template>

<xsl:template match='//b'>
    <c><xsl:apply-templates/></c>
</xsl:template>

</xsl:stylesheet>
```

- Welches Ergebnis entsteht, wenn dieses XSLT-Stylesheet auf das XML-Dokument
`<a>inhalt</a>`
angewandt wird?
- Erläutern Sie, wie das Ergebnis zustande kommt.

Aufgabe 6-4 XSLT (bib)

Veranschaulichen Sie die Entstehung der Ausgabe, wenn das untenstehende XSLT-Stylesheet auf das untenstehende XML-Dokument angewendet wird. Illustrieren Sie dabei insbesondere schrittweise (1) welche Templates (2) auf welche Knoten angewandt werden und (3) welches Ergebnis dabei entsteht.

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="bib">
    <html><body><ul>
      <xsl:apply-templates select=".*[position()>2]" />
    </ul></body></html>
  </xsl:template>

  <xsl:template match="paper">
    <li>
      <xsl:apply-templates />
    </li>
  </xsl:template>

  <xsl:template match="paper/title">
    <em>
      <xsl:apply-templates />
    </em>
  </xsl:template>

  <xsl:template match="author">
    by <xsl:apply-templates />.
  </xsl:template>
</xsl:stylesheet>
```

```
<?xml version="1.0" encoding="utf-8" ?>
<bib>
  <uri>http://www.example.com/bibliography.xml</uri>

  <paper>
    <title>On the criteria to be used in decomposing systems
      into modules</title>
    <author>D. L. Parnas</author>
    <uri>http://doi.acm.org/10.1145/361598.361623</uri>
  </paper>

  <paper>
    <title>Implementing Quicksort programs</title>
    <author>Robert Sedgewick</author>
    <uri>http://doi.acm.org/10.1145/359619.359631</uri>
  </paper>

  <paper>
    <title>Ueber formal unentscheidbare Saetze der Principia
      Mathematica und verwandter Systeme I</title>
    <author>Kurt Goedel</author>
  </paper>
</bib>
```

Aufgabe 6-5 XPath, ein steiniger Pfad

Das folgende XML-Dokument repräsentiert einen Ausschnitt aus einer Netzwerkdarstellung:

```
<!DOCTYPE net SYSTEM "net.dtd">
<net>
  <node id="n0" links="n1 n3" />
  <node id="n1" links="n0 n2 n4" />
  <node id="n2" links="n1 n4" />
  <node id="n3" links="n0 n4" />
  <node id="n4" links="n2 n3" />
  <node id="n5" links="" />
</net>
```

Offensichtlich müssen die *id* Attribute vom Typ *ID* und die *links* Attribute vom Typ *IDREFS* sein, wie in der Datei *net.dtd* deklariert. Geben Sie XPath-Ausdrücke an, deren Ergebnis

- alle Knoten, die von dem Knoten mit Attributswert 'n0' aus auf der zweiten Ebene (also folgen Sie nicht einem sondern zwei Referenzen ausgehend von 'n0') erreichbar sind, repräsentiert.
- alle Knoten, die eine Referenz auf sich selbst haben, repräsentiert.
- alle Knoten und die nicht referenziert werden, repräsentiert.
- die Anzahl aller Knoten, die nicht referenziert werden.
- die Konsistenz des Netzwerks testet, d.h. wenn ein Knoten A von einem Knoten B aus erreichbar ist, so ist der Knoten B auch von dem Knoten A aus erreichbar.

Aufgabe 6-6 Vorwärts-Anfragen in XPath

XPath bietet die Möglichkeit, Anfragen mit Vorwärts- und Rückwärts-Achsen zu spezifizieren. Mit einer Vorwärts-Achse kann man von einem Knoten aus andere Knoten auswählen, die in der Dokument-Reihenfolge nach diesem Knoten vorkommen, z.B. **child**, **descendant**, **following-sibling**, **following**. Mit einer Rückwärts-Achse kann man von einem Knoten aus andere Knoten auswählen, die in der Dokument-Reihenfolge vor diesem Knoten vorkommen, z.B. **parent**, **ancestor**, **preceding-sibling**, und **preceding**. Eine Anfrage, die nur Vorwärts-Achsen enthält, ist eine Vorwärts-Anfrage. Zwei Anfragen sind äquivalent, wenn sie für alle XML-Dokumente jeweils die selben Knoten auswählen.

- Geben Sie ein paar Beispiele von zwei äquivalenten XPath-Anfragen an, wobei die eine nur Vorwärts-Achsen und die andere auch Rückwärts-Achsen enthält.
- Gibt es immer für eine Anfrage, die auch Rückwärts-Achsen enthalten kann, eine äquivalente Vorwärts-Anfrage ?