Prof. Dr. Stefan Brass
Institut für Informatik
MLU Halle-Wittenberg

07. Februar 2019

XML und Datenbanken
— 14. Übungsblatt: XSLT —

Dieses Übungsblatt für die letzte Semesterwoche enthält keine Hausaufgaben mehr. Das Thema XSLT ist allerdings grundsätzlich noch klausurrelevant (und auch praxisrelevant!) und sollte noch geübt werden.

a) Wie würden Sie in einer mündlichen Prüfung auf folgende Fragen zu XSLT antworten?

- Eine Anwendung von XSLT ist es, XML Daten-Dateien nach HTML zu übersetzen, um die Daten in einem Web-Browser anzuzeigen. Diese Transformation kann im Browser (Client) oder auf dem Server ausgeführt werden. Was sind die Vor- und Nachteile der beiden Möglichkeiten?
- Was kann man mit XSLT noch machen, außer XML nach HTML zu übersetzen?
- Beschreiben Sie den Aufbau eines Templates. Was ist das wichtigste Attribut von `xsl:template`? Welche `xsl`-Elemente werden besonders häufig im Innern von `xsl:template` verwendet?
- Wenn das Attribut `match` eines Templates den Wert `A/B` hat, passt es dann auf ein `A`-Element mit `B`-Kind, oder auf ein `B`-Element mit `A`-Elternteil?
- Falls mehrere Templates auf einen Knoten passen, welches wird angewendet? Wie kann man ggf. die Auswahl steuern?
- Was ist der Zweck der eingebauten Templates von XSLT?
- Was bewirkt `<xsl:apply-templates select="A"/>` im Innern eines Templates, das auf einen Knoten n angewendet wird? D.h. wie wird das entsprechende Stück im Ergebnisbaum bestimmt?
- Was bewirkt `<xsl:value-of select="A"/>` im Innern eines Templates, das auf einen Knoten n angewendet wird?
- Wenn ein Template auf den Knoten n angewendet wird, ist dies ja zunächst der Kontext-Knoten. Wie können Sie in Pfad-Ausdrücken im Innern des Templates darauf zugreifen, wenn der Kontext-Knoten verstellt wurde (siehe Aufgabe)?
- Wenn Sie HTML erzeugen wollen, können Sie im XSLT-Stylesheet dann einfach `<br>` schreiben, oder müssen Sie `<br />` schreiben?
- Wie verweist eine XML-Datei auf ein XSLT-Stylesheet?
- Was ist der Zweck von `xsl:output`?
- Ist es möglich, dass ein Stylesheet nicht terminiert? Geben Sie ggf. ein Beispiel.

- Nennen Sie einige Unterschiede bzw. Einschränkungen von XPath 1.0 gegenüber XPath 2.0.
- Vergleichen Sie XSLT mit XQuery. Wie werden z.B. neue Knoten konstruiert? Wie kann man sortieren?
- Wie kann man Fallunterscheidungen in XSLT formulieren? Z.B. wollen Sie bei CDs mit nur einer Aufnahme “1 Aufnahme” (Singular) ausgeben, bei CDs mit mehreren Aufnahmen dagegen z.B. “3 Aufnahmen” (Plural).

Präsenzaufgaben

Verwenden Sie nochmals die XML-Datensammlung für klassische Musik-CDs:

- Daten:
  [http://www.informatik.uni-halle.de/~brass/xml18/examples/cd/cd.xml]
- XML Schema:
  [http://www.informatik.uni-halle.de/~brass/xml18/examples/cd/cd.xsd]
- DTD:
  [http://www.informatik.uni-halle.de/~brass/xml18/examples/cd/cd.dtd]

Das Dokument enthält die folgenden Elemente:

- `CDDB`: `(komponisten, cds?, solisten?)`
- `komponisten`: `(komponist*)`
- `komponist`: `(stueck*)`
  Attribute: `knr`, `vorname`, `name`, `geboren`, `gestorben`.
- `stueck`: `(aufnahme*)`
  Attribute: `snr`, `titel`, `tonart`, `opus`.
- `aufnahme`: Leerer Inhalt.
  Attribute: `aufnnr`, `orchester`, `leitung`.
- `cds`: `(cd*)`.
- `cd`: `(CDAufnahme*)`
  Attribute: `cdnr`, `name`, `hersteller`, `anz-cds`, `gesamtspielzeit`.
- `CDAufnahme`: Leerer Inhalt.
  Attribute: `aufnnr`.
- `solisten`: `(solist*)`.
- `solist`: `(solist_aufnahme*)`
  Attribute: `name`.
- `solist_aufnahme`: Leerer Inhalt.
  Attribute: `aufnnr`, `instrument`.

Schreiben Sie ein XSLT-Stylesheet zur Umwandlung dieser XML-Daten nach HTML für die folgende Aufgabe.

b) Das Stylesheet soll eine “unordered list” der CDs erstellen, wobei jeweils

- der Name der CD gedruckt wird,
- anschließend in eckigen Klammern die Anzahl der Aufnahmen auf der CD,
- und dann eine geschachtelte Liste mit den Titeln der Stücke, von denen es Aufnahmen auf der CD gibt.

Die Ausgabe soll also ungefähr so aussehen (die Reihenfolge ist beliebig):

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
          "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>CD Verzeichnis</title>
    </head>
    <body>
        <h1>CD Liste</h1>
        <ul>
            <li>Händel: Feuerwerksmusik [4]
                <ul>
                    <li>Concerto grosso op.6 Nr.7</li>
                    <li>Feuerwerks-Musik (Concerto grosso)</li>
                    <li>Concerto grosso op.6 Nr.8</li>
                    <li>Sinfonia e-moll (aus ‘Der Messias’)</li>
                </ul>
            </li>
            ...
        </ul>
    </body>
</html>
```

Hinweis: Die Funktion `current()` liefert den aktuellen Knoten, also den, auf den das Template gerade angewendet wird. Der aktuelle Knoten ist zunächst auch der Kontext-Knoten, so dass man in einfachen Mustern diese Funktion nicht braucht. Wenn Sie aber einen etwas komplizierten XPath-Ausdruck schreiben, ändert sich der Kontext-Knoten innerhalb des Ausdrucks (z.B. nach “/” oder innerhalb von “`[...]`”), während `current()` noch immer den Knoten liefert, auf den das Template angewendet wird. Dies ist z.B. nützlich, wenn Sie von einer `CDAufnahme` zum zugehörigen `stueck`-Element navigieren wollen.