

# XML-Technologie: Übungsblatt 3

## Version 1.0

**Ausgabe am 16. November 2016, Abgabe (optional) bis zum 27. November 2016**

**Vorbereitung:** Installieren Sie einen XSLT-Prozessor, der XSLT 2.0 unterstützt, z.B. Saxon. Alternativ können Sie eine XML-Entwicklungsumgebung benutzen, die diese Sprachen unterstützt, z.B. oXygen. Arbeiten Sie sich, etwa an Hand des Skripts oder von Online-Tutorials, in XSLT ein, bevor Sie die Aufgaben dieses Blatts bearbeiten. Studieren Sie insbesondere die Serie von Transformationsprogrammen für das Buchbeispiel (Skript und Folien).

**Aufgabe 1:** Welche Zustandsdaten benötigen Sie für die Benutzungsoberfläche Ihres Mancala-Spiels (Verteilung der Steine, wer ist am Zug, hat schon jemand gewonnen)? Beschreiben Sie diese Daten in einem UML-Klassendiagramm. Übersetzen Sie dann das Klassendiagramm in eine XML DTD. Kodieren Sie dann einige Instanzen der DTD / Zustandsdaten für Mancala in XML.

Dokumentieren Sie Ihre Lösung mit DocBook.

**Aufgabe 2:** Überlegen Sie, wie Sie die in Aufgabe 1 entwickelten Zustandsdaten nach SVG übersetzen können, um die graphische Oberfläche Ihres Mancala-Spiels zu realisieren. Dazu greifen Sie auf Ihre Lösung von Aufgabenblatt 2 zurück. Eventuell müssen Sie Ihre SVG-Graphik noch besser explizit parameterisieren, damit die systematische Übersetzung möglich ist.

Entwerfen und implementieren Sie dann ein XSLT-Programm, das aus Ihren Zustandsdaten in XML-Kodierung eine HTML/SVG-Datei für ihre graphische Oberfläche generiert.

Eine Anregung zur Vorbereitung Ihrer SVG-Datei: Wenn Sie vorher feste Werte als Entitäten eingebunden haben, können Sie nun als Ersetzungstext dieser Entitäten leere Elemente als Markierungen definieren, die dann, gesteuert durch die Zustandsdaten, in einer XSLT-Transformation durch die passenden festen Werte ersetzt werden.

Dokumentieren Sie Ihre Lösung mit DocBook.

**Ausblick:** In diesem Aufgabenblatt generieren Sie aus den XML-kodierten Zustandsdaten mit Hilfe von SVG die graphische Oberfläche. Im nächsten Schritt des Praktikums werden Sie mit Hilfe von XQuery die Spielzüge implementieren, die auf diesen Zustandsdaten arbeiten. In einem letzten Schritt binden Sie dann die entwickelten Komponenten zu einer Web-Anwendung zusammen.

**Präsentation:** Zwei Gruppen können ihre Lösung zu Aufgaben 1 und 2 am 23. November 2016 im Praktikum präsentieren und dazu einen Notenbonus erhalten. Ein Slot ist bereits vergeben. Es gelten die folgenden Spielregeln:

1. Die Gruppen, die sich zuerst per E-Mail bei mir melden, bekommen den Zuschlag.
2. Es gilt ein striktes Zeitlimit von 3 Minuten pro Gruppenmitglied für die Präsentation.
3. Die Lösung muss 36 Stunden vor der Präsentation per E-Mail vorzeitig abgegeben werden.
4. Nur Studierende, die sich an der Präsentation aktiv beteiligen, bekommen den Notenbonus.
5. Konzentrieren Sie sich in der Präsentation auf die Organisation Ihrer Lösung, wie in der Aufgabenbeschreibung angegeben.

**Abgabe:** Geben Sie Ihre Lösung der Aufgabe bis zum 27. November, Mitternacht, in einem Archiv-Format in Moodle ab. Geben Sie Ihrer Archiv-Datei einen unverwechselbaren Namen, z.B. ein Kürzel aus den Namen Ihrer Gruppenmitglieder. Schreiben Sie außerdem Ihre Namen als Kommentar in jede einzelne Datei. Die Abgabe ist optional. Ich werde stichpunktweise reinschauen und Ihnen Feedback geben. Wenn Sie konkrete Fragen haben, diskutieren Sie die im Moodle-Forum oder sprechen Sie mich explizit an.

**Leistungsstand:** Die folgenden Lernziele sollten Sie inzwischen erreicht haben:

- Verständnis für den Hintergrund von XML.
- Beherrschen der Sprache XML: Instanzen mit Elementen, Attributen, Referenzen und Text; Definition von Entitäten in DTD.
- Parsen von XML-Dokumenten.
- Visualisieren von XML-Dokumenten in Web-Browsern mit CSS.
- Einsatz von Kodierungsformaten in XML.
- Einsatz von Namensräumen in XML.
- Umgang mit DocBook.
- Strukturierung und Implementierung von SVG-Dokumenten.

**Lernziele:** Konkrete Lernziele für die aktuelle Arbeit sind:

- Weitere Vertrautheit mit XML DTD; Modellierung von XML-Anwendungen; Deklaration von Elementen und Attributen.
- Praktischer Umgang mit XPath.
- Entwicklung modularer und konfigurierbarer XSLT-Programme.

**Ausblick:** In den nächsten drei bis vier Wochen werden wir uns mit folgenden Themen beschäftigen:

- XQuery und XML-Datenbanken.
- XML-Datenbanken als serverseitige XQuery-Prozessoren.
- 
- GUIs mit XForms
- Mapping von HTTP-Requests und HTTP-Responses auf XQuery-Funktionen mit RestXQ.