



XML-Technologien

Tutorium 5

Thema:
XSLT

von:
Kain Kordian Gontarska
kainkordian@gmail.com

XSLT

- XSL - Extensible Stylesheet Language:
 - Xpath
 - XSLT
 - XSL-FO
- XSLT - XSL Transformation:
 - 'Programme' zur Transformation von XML-Dokumenten
 - Layout oder Inhalt kann transformiert werden
 - In XML-Syntax gehalten
 - Erlaubt XML-Dokumente in beliebige Textformate zu transformieren
 - Eigenständiges Dokument
 - Kann mit XML-Dokument verknüpft werden:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="file.xsl"?>
<root>
<!-- ... -->
</root>
```

XSLT - Paradigma

- Ein XSLT – Stylesheet wird von einem XSLT-Prozessor auf ein XML-Dokument angewandt (Xalan, Saxon, ...)
- Ein XSLT-Stylesheet besteht aus einer Menge von Transformationsregeln (Templates)

```
<xsl:template match="XPath expression">  
  ...  
  <p><xsl:value-of select="."/>  
  ...  
</xsl:template>
```

- Alle Templates (genauer: XPath Ausdrücke) beziehen sich auf das Eingangsdokument.
- Funktionale Sprache: Variablen können nicht überschrieben werden, es gibt keine Seiteneffekte.

XSLT - Regelauswahl

- Alle Regeln werden betrachtet, eine wird angewandt.
- Sollte beim Patternmatch mehr als eine Regel gelten, dann wird die konkreteste gewählt. (Die Priorität wächst mit der Spezialisierung des Xpath Ausdrucks.)
- Bei gleicher Priorität wird die Reihenfolge im Dokument beachtet. (Sollte nach Möglichkeit vermieden werden.)
- Priorität der Regel mit dem Attribut priority bestimmbar. 0 ist der Standardwert.
- Prioritätenverteilung

Direkter Pfad / Priorität	Namensnennung	Mit *-Präfix (*/*foo)	Nur Wildcard (*/*@/node())
0.5	0	-0.25	-0.5

XSLT – Basic Funktionalität

- `<xsl:value-of select="foo"/>`- Gibt den Wert der Auswahl zurück (Element, Attribut, XPath, Variable)
- `<xsl:copy-of select="bar"/>`- Kopiert alle Knoten der Auswahl
- `<xsl:variable name="var" select="something"/>`- Kann als Variable im Template verwendet werden.
- `<xsl:template match="bla"/>`- Wendet folgende Regel auf alle Knoten an, die gematched werden. (Optional mit Attribut `name="template1"` für call-template)
- `<xsl:apply-templates/>`- Wendet Templates auf alle Kinder des Kontextknotens an.
- `<xsl:call-template name="myTemplate">`- Ruft Regel mit gegebenem Namen auf (innerhalb einer Regel), Optional können Parameter übergeben werden.
- `<xsl:with-param name="myPara" select="foo"/>`- Übergibt den Parameter und kann von Template aufgerufen werden.
- `<xsl:param name="myPara"/>`- Erwartet Parameter myPara, kann ab hier als Variable verwendet werden.

XSLT – Programmsteuerung

- Zu apply-templates und call-template kommen einfachere Strukturen hinzu.
- `<xsl:if test="someBooleanXPath">bla</xsl:if>`- Fragt eine Bedingung ab, bevor der Codeteil 'bla' ausgeführt wird, sonst wird hinter dem schließenden Tag weiter gemacht. (Kein Else)
- `<xsl:choose>` - Eröffnet eine Mehrfachauswahl (Von Oben nach Unten)
 - `<xsl:when test="someBoolean">bla</xsl:when>`- Analog zu if
 - `<xsl:otherwise>boo</xsl:otherwise>`- Bei keiner when Übereinstimmung`</xsl:choose>`
- `<xsl:for-each select="./*">Do Something</xsl:for-each>`- Führt für alle durch den Ausdruck erreichten Knoten eine Anweisung aus, Nacheinander.