

XSLT

Lordick, Harald; lordick@steinheim-institut.org

XSLT (Extensible Stylesheet Language Transformations) ist mit XPath und XSL-FO Teil der in XML notierten Extensible Stylesheet Language (XSL) und gehört zu den Web-Standards des W3C. Die Turing-vollständige und dokumentenorientierte Programmiersprache dient insbesondere der Transformation von XML-Daten und findet häufig als Verarbeitungskomponente digitaler Workflows von Editionen Verwendung, etwa in Frameworks wie Apache Cocoon, Kiln, TEI-Publisher, TEI Boilerplate oder Versioning Machine (siehe auch Web-Architektur für DSE).

XSLT benutzt die Abfragesprache *XPath*, um durch XML-Dokumente zu navigieren und Teile davon zu adressieren, und XSLT-Elemente- (Programmieranweisungen) eröffnen die Möglichkeit, XML-Eingabedokumente in eine andere Struktur bzw. ein anderes Ausgabeformat zu überführen: Text (Plain Text, Markdown, JSON, CSV, CSS, MIF, Programm / Script-Code ...), XML (XML, XSLT, SVG, PDF via XSL-Fo ...) sowie HTML.

Eine originäre Eigenschaft von XSLT-Skripten ist die Dokumententransformation durch Templates. Das folgende Beispiel adressiert alle `<div>`-Elemente eines XML-Eingabedokuments und erzeugt dafür jeweils ein `<p>`-Element im Ausgabedokument. Die Anweisung `<xsl:apply-templates/>` stößt dann die Verarbeitung gegebenenfalls enthaltener Kind-Knoten des `lt;divgt;`-Elements an, soweit dafür entsprechende (weitere) Templates vorliegen.

```
<xsl:template match="div">
<p>
<xsl:apply-templates/>
</p>
</xsl:template>
```

Im mächtigen *Deep-Copy-* bzw. *Shallow-Copy-Modus* werden dabei seit XSLT 3.0 Dokumentstrukturen, für die kein Template formuliert ist, unverändert übernommen. Dies erlaubt die gezielte Überarbeitung von Details komplexer XML-Dokumente bei nur geringem Programmieraufwand.

Die Dokumentenzentrierung von XSLT – und damit auch die besondere Eignung für Editionsdaten – zeigt sich nicht nur in diesem Template-Ansatz, sondern auch etwa in der Konzeption der *For-Each*-Schleife, die anders als in anderen Programmiersprachen üblich, ausschließlich auf (durch XPath oder XSLT-Funktionen adressierte) Dokumentstrukturen iteriert.

Die Entwicklung des XSLT-Sprachumfangs bis hin zu Version 3 hat aber auch Eigenschaften gestärkt, die eine allgemeinere Anwendung von XSLT ermöglichen: erweiterte Unterstützung von Datentypen, Verarbeitung des verbreiteten JSON-Datenformats, Funktionen zur String-Manipulation (einschließlich regulärer Ausdrücke), Funktionen und Operatoren für Datum, Zeit, Zeitverlauf sowie erweiterte Möglichkeiten der Sortierung, Filterung und Gruppierung von Dokumentstrukturen. Die direkte Einbindung (d. h. die clientseitige Ausführung) in Webbrowsern blieb allerdings auf Version 1 beschränkt.

XSLT unterstützt konzeptbedingt die klare Trennung von Editionsdaten von ihrer Präsentation und hat über XML-strukturierte Dokumente hinaus mittlerweile breite Einsatzmöglichkeiten im Editionsworkflow: Bereinigung, Strukturierung und Anreicherung von Daten, Splitten und Zusammenführen von Dokumenten, Aufbereitung für Datenaustausch sowie Transformation in Darstellungs- und Publikationsformate.

Literatur:

- Xsl transformations (xslt). URL: <http://new.renderx.com/files/demos/xmlspec/xslt/REC-xslt-19991116.pdf>
- Bia, Alejandro; Sanchez-Quero, Manuel; Deau, Regis: Multilingual markup of digital library texts using XML, TEI and XSLT. In: XML Europe 2003 Conference and Exposition XML Europe 2003 Conference and Exposition. London: 2003, S. 63-68.
- Mangano, Sal: XSLT Cookbook: Solutions and Examples for XML and XSLT Developers. Köln: 2006.
- Kay, Michael: XSLT programmer's reference. New York: 2002.
- W3C Recommendation: XSL Transformations (XSLT) Version 3.0. URL: <https://www.w3.org/TR/2017/REC-xslt-30-20170608/>
- Tennison, Jeni: Beginning XSLT 2.0: from novice to professional Beginning XSLT 2.0. Berkeley, CA : New York: 2005, URL: <http://it-ebooks.info/book/1789/>.
- Tidwell, Doug: XSLT: XML-Dokumente transformieren. Köln: 2002.
- Wadler, Philip: A formal semantics of patterns in XSLT and XPath. In: Markup Languages 2: 2000, S. 183-202.
- XSL Transformation. URL: https://de.wikipedia.org/wiki/XSL_Transformation

Software:

Oxygen, xsl-tokenizer, Wippets - XSLT/HTML/JS-Fragmente für GAMS

Verweise:

XML

Themen:

Einführung, Datenanalyse, Software und Softwareentwicklung

Zitiervorschlag:

Lordick, Harald. 2021. XSLT. In: KONDE Weißbuch. Hrsg. v. Helmut W. Klug unter Mitarbeit von Selina Galka und Elisabeth Steiner im HRSM Projekt "Kompetenznetzwerk Digitale Edition". URL: <https://gams.uni-graz.at/o:konde.86>