

Interaktive XML-Verarbeitung im Browser mit Saxon-JS und XSLT 3.0

Dr. Thomas Meinike, Hochschule Merseburg

Motivation

Die direkte Nutzung von XSLT in Browsern ist für die Web-Entwicklung mangels Unterstützung aktueller Standards durch die Hersteller meistens keine Option mehr. Dennoch bietet sich die client-seitige Verarbeitung von XML-Daten in unterschiedlichen Kontexten an. Saxon-JS ist das jüngste Produkt aus dem Hause Saxonica unter Federführung von Michael Kay. Es wurde mit dem Ziel entwickelt, die Möglichkeiten von XSLT 3.0 zur XML-Verarbeitung in Web-Browsern nutzen zu können. Dem kam der Vorgänger Saxon-CE mit 2.0-Techniken schon ziemlich nahe, konnte sich aber kaum etablieren. Dieser Beitrag soll einen kompakten Überblick zu den erweiterten Möglichkeiten von Saxon-JS vermitteln.

Einstieg

Saxon-JS wurde 2016 vorgestellt [1-2] und im Februar 2017 in Version 1.0.0 verfügbar gemacht [3a]. Das erste Wartungsupdate 1.0.1 erschien im Juli 2017. Technisch handelt es sich um eine JavaScript-basierte Laufzeitumgebung zur Transformation von XML-Daten und alternativen Datenstrukturen wie JSON. Im Gegensatz zum 2013 präsentierten Vorgänger Saxon-CE [4] werden die erstellten XSLT-Stylesheets zunächst mit dem kommerziellen Prozessor Saxon-EE ab 9.7 in SEF-Dokumente umgewandelt. SEF steht für Stylesheet Export File, ein optimiertes XML-Format für den Transformationsbaum. Abbildung 1 veranschaulicht die Anwendungsarchitektur.

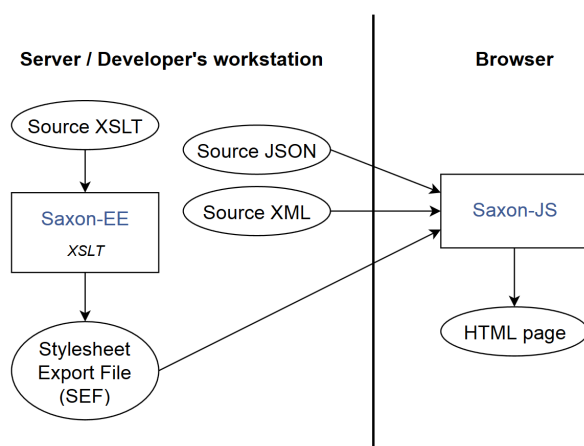


Abb. 1: Architektur-Diagramm aus der Saxon-JS-Dokumentation [3b]

Die zu verarbeitenden Komponenten werden mit der JavaScript-Methode `SaxonJS.transform(...)` aus einem HTML-Dokument heraus prozessiert. In letzterem wird die kompakte 225 KB umfassende Bibliothek `SaxonJS.min.js` referenziert. Spezielle Details zum SEF-Aufbau werden in [5] behandelt.

Praktische Nutzung

Von Saxon-CE zu Saxon-JS

Saxon-JS ermöglicht die Nutzung einer Vielzahl an Techniken aus den kürzlich finalisierten Spezifikationen von XSLT 3.0 und XPath 3.1 [6-7]. Eine Zwischenbilanz zog der Autor bereits 2012 [8]. Einschränkungen benennt die Dokumentation unter „Conformance“. Dazu gehören die neuen Techniken für das Streaming von Dokumenten, die Schema-Validierung von Ausgaben und die Unterstützung von Funktionen höherer Ordnung. Neben bereits aus XSLT/XPath 2.0 stammenden Möglichkeiten wie Sequenzen, Gruppierungen und eigene Funktionen liefern die 3.x-Erweiterungen u. a. nützliche Konzepte für Arrays und Maps, kompakte Text-Value-Templates `<x>{...}</x>` sowie erweiterte mathematische Funktionen, etwa `math:sin(...)` und `math:cos(...)` beim folgenden Beispiel.

Als erster Test wurde die in [4] verwendete CE-Anwendung zur Darstellung der Studierenden-Zahlen nach Jahrgängen problemlos portiert, siehe Abbildung 2. Die Daten sind in einem XML-Dokument gespeichert. Konzeptionell wird weiterhin mit `xsl:template`-Einheiten gearbeitet, die ihre Ergebnisse über `xsl:result-document` an jeweils mit IDs bestückte Blöcke im HTML-Dokument ausliefern.

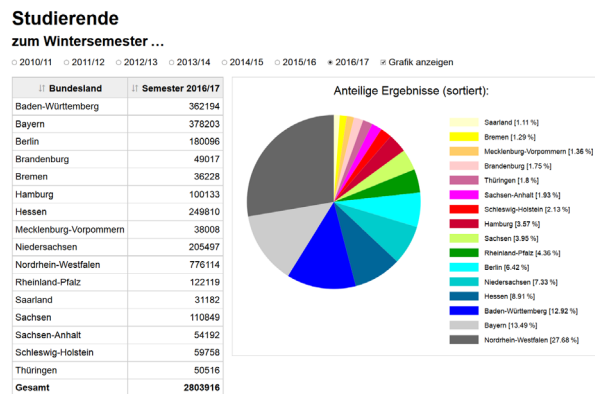


Abb. 2: Saxon-JS-Anwendung „Studierende“ [9] mit Datenbasis [10]

Verarbeitung von JSON-Daten

Ansätze zur Nutzung des zunehmend populären JSON-Formates wurden ebenfalls erprobt. Mit der Funktion `fn:json-doc(...)` ließ sich eine Datenstruktur als Map einlesen und in ihre Teile zerlegen, hier Jahreszahlen und Mengenangaben. Abbildung 3 zeigt die resultierende SVG-Balkengrafik.

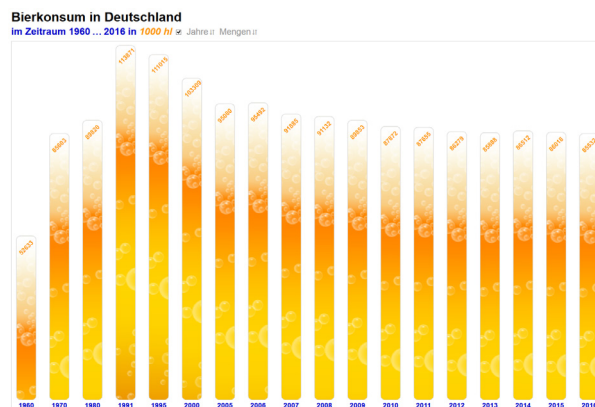


Abb. 3: Saxon-JS-Anwendung „Bierkonsum“ [11] mit Datenbasis [12]

Praxistest: Das Wahlinfo-Projekt

Abbildung 4 zeigt die Browseransicht eines umfangreicheren Projektes, das unter [13] online verfügbar ist. Im August 2017 stellte der in unserer Region ansässige Verein „wahlinfo+ e. V.“ [14] detaillierte Informationen zur Bundestagswahl bereit. Dazu gehören allgemeine Informationen der antretenden Parteien und spezielle zu ihren Kandidaten. Eine zusätzliche Bereitstellung von XML-Schnittstellen zu den jeweiligen Datensätzen als Open Data weckte sofort das Interesse des Autors.

Die Parteien zur Bundestagswahl 2017

von deutschlandwaehlt.de – [wahlinfo+ e.V.](http://wahlinfo+e.v.de)

Auswahl

Partei für Arbeit, Rechtsstaat, Tierschutz, Elitenförderung und basisdemokratische Initiative

Informationen

Partei:	Partei für Arbeit, Rechtsstaat, Tierschutz, Elitenförderung und basisdemokratische Initiative
Kurzform:	Die PARTEI
Adresse:	Kopischstraße 10 10965 Berlin
Programm:	 PDF-Version
Kontakte:	 E-Mail  Facebook  Instagram  Twitter  Web

Kandidaten (1)

Name:	Serdar Somuncu
Kontakte:	 Facebook  Twitter  Web

Änderungsdatum: 19.07.2017

Abb. 4: Saxon-JS-Anwendung „Wahlinfo“ [13] mit Datenbasis [14]

Die entwickelte Anwendung arbeitet zweistufig. Zunächst wird aus einer XML-Quelle die Liste aller (42) verfügbaren Parteien in eine Auswahlliste überführt. Dabei sind die Datensätze der einzelnen Parteien mit IDs ausgestattet. Somit lassen sich nach Auswahl der gewünschten Partei die zugehörigen Details nachladen. Derartige Interaktionen werden über die ixsl-Erweiterungen (Interactive XSLT) gesteuert, hier mit `ixsl:onchange`. Ausgegeben werden letztlich Adressen, Links zu PDF-Partei-programmen und die verfügbaren E-Mail-, Web- und Social-Media-Kontakte. Sofern es Spitzenkandidaten gibt, werden die dazu hinterlegten Daten ebenfalls analysiert und präsentiert.

Bei diesem Projekt ließen sich viele technologische Facetten von Saxon-JS ausprobieren, angefangen vom externen Datenzugriff mittels `fn:doc(...)` über die alphabetisch sortierte und nach Anfangsbuchstaben gruppierte Auswahlliste bis hin zur Einzeldatenabfrage und deren Aufarbeitung. Am Ende der Ausgabe erscheint das Datum der letzten Änderung des jeweiligen Datensatzes auf dem angefragten Server. Hierzu wurden die Zeitstempel mit geeigneten Funktionen umgeformt.

Fazit und Ausblick

Saxon-JS bereichert die Web-Entwicklung mit Fokus auf XML- oder auch JSON-Daten. Bei entsprechender Akzeptanz können interessante Anwendungen entstehen. Der breite Unterbau in Form der aktuellen Spezifikationen von XSLT und XPath konnte hier nur angerissen werden. Ein Überblicksartikel unter [15] zeigt zahlreiche neue Möglichkeiten auf und der Vortrag widmet sich weiteren Code-details der genannten Beispiele. Die Saxonica-Plattform stellt aktuelle Informationen und Problemlösungen bereit und ermöglicht Entwicklern einen aktiven Austausch [16].

Literaturangaben und Links

- [1] Lockett, D. und Kay, M.: Saxon-JS: XSLT 3.0 in the Browser. In: Proceedings of Balisage: The Markup Conference 2016, DOI: 10.4242/BalisageVol17.Lockett01. <https://www.balisage.net/Proceedings/vol17/html/Lockett01/BalisageVol17-Lockett01.html>
- [2] Kay, M.: Transforming JSON using XSLT 3.0. In: XML Prague 2016 – Conference Proceedings, S. 167-183. <http://archive.xmlprague.cz/2016/files/xmlprague-2016-proceedings.pdf>
- [3] Saxonica: Saxon-JS. (a) <http://saxonica.com/html/saxon-js/> (b) <http://www.saxonica.com/saxon-js/documentation/>
- [4] Meinike, T.: XSLT 2.0 im Browser mit Saxon-CE. In: tekomp, Gesellschaft für technische Kommunikation e. V., Tagungsband zur Jahrestagung 2013, S. 338–341.
- [5] Lumley, J., Lockett, D. und Kay, M.: XPath 3.1 in the Browser. In: XML Prague 2017 – Conference Proceedings, S. 1-18. <http://archive.xmlprague.cz/2017/files/xmlprague-2017-proceedings.pdf>
- [6] W3C: XSL Transformations (XSLT) Version 3.0. <http://www.w3.org/TR/xslt-30/>
- [7] W3C: XML Path Language (XPath) Version 3.1. <http://www.w3.org/TR/xpath-31/>
- [8] Meinike, T.: 3.0-Updates von XSLT und XPath auf einen Blick. In: tekomp, Gesellschaft für technische Kommunikation e. V., Tagungsband zur Jahrestagung 2012, S. 341–343.
- [9] Meinike, T.: Studierende in Deutschland. <http://datenverdrahten.de/xslt3/saxon-js/studis/>
- [10] Statistisches Bundesamt (Destatis): Studierende – Insgesamt nach Bundesländern und tiefer gegliederten Angaben. <https://www.destatis.de/DE/ZahlenFakten/GesellschaftStaat/BildungForschungKultur/Hochschulen/Tabellen/StudierendeInsgesamtBundeslaender.html>
- [11] Meinike, T.: Bierkonsum in Deutschland. <http://datenverdrahten.de/xslt3/saxon-js/bierkonsum/>
- [12] Statista: Deutscher Brauer-Bund – Konsum von Bier in Deutschland in den Jahren 1960 bis 2016 (in 1.000 Hektoliter). <https://de.statista.com/statistik/daten/studie/29630/umfrage/bierverbrauch-bierkonsum-in-deutschland/>
- [13] Meinike, T.: Die Parteien zur Bundestagswahl 2017. <http://datenverdrahten.de/xslt3/saxon-js/wahlinfo/>
- [14] wahlinfo+ e. V.: Deutschland wählt – Bundestagswahl 2017. <http://deutschlandwaehlt.de/>
- [15] Cagle, K.: Why You Should Be Using XSLT 3.0. <https://www.xml.com/articles/2017/02/14/why-you-should-be-using-xslt-30/>
- [16] Saxonica: Developer Community. <https://saxonica.plan.io/projects/saxon-js/>

für Rückfragen:
thomas.meinike@hs-merseburg.de