

Mediennutrales Publizieren mit XML und DocBook

Im Zuge der zunehmenden Verflechtung verschiedener, auch weltweit verteilter Fertigungsstätten bei der Herstellung eines Produktes fällt der Austauschbarkeit von Dokumenten und Daten aller Art immer größere Bedeutung zu. Hinzu kommt noch der Wunsch, die Dokumente auch nach einem Wechsel des Softwareherstellers oder dessen eventuellem Verschwinden vom Markt weiterhin benutzen bzw. weiter entwickeln zu können. Nicht zuletzt ist es oft wünschenswert, einmal erfasste Daten in verschiedenen Medien, also z.B. als gedruckten Katalog oder im Internet ohne besonderen Aufwand veröffentlichen zu können.

Problematik und heutige Situation

Allgemein üblich ist es, sich auf einen so genannten Industriestandard zu einigen, wobei die meisten Beteiligten unter Industriestandard die Datenformate des jeweiligen Marktführers verstehen. Bei Textdokumenten ist dies das Microsoft-Word.DOC-Format, bei Tabellen Microsoft-Excel.XLS und bei Zeichnungen meist das AutoCAD.DWG-Format.

Leider gestaltet sich trotz der Einigung auf die Software der Marktführer der Datenaustausch nicht immer unproblematisch. Durch die relativ kurzen Innovationszyklen bei Software kommt es hin und wieder zu einer Änderung des Datenformats, so dass vor allem ältere Programmversionen die Dateien der neueren Versionen nicht mehr lesen können. Da die Software der Marktführer nicht gerade billig ist und die Umstellung meist einen gewissen Schulungsaufwand bedeutet, ist nicht jeder am Datenaustausch Beteiligte bereit, jedes Programm-Update mitzumachen. Hinzu kommt das nicht sehr erhebende Gefühl, von einem einzigen Hersteller abhängig zu sein und auf dessen Befehl hin größere Investitionen zu tätigen. Besonders Schulen mit ihren

zurzeit eher mageren Haushaltsmitteln könnten diese in vielen Fällen für dringendere Anschaffungen gebrauchen.

Selbst bei kostenlosem Bezug der Software (z.B. Versuchsschulen) bleibt immer noch das Problem der Langzeitarchivierung und der Mehrfachverwendung in verschiedenen Ausgabemedien.

Lösungsansätze

Standardisierung

Schon mehr als zehn Jahre ist ein Gremium, das so genannte W3C¹ mit der Standardisierung des World Wide Web betreffender Techniken beschäftigt. Folgende de-facto-Standards u. v. a. m. wurden mit Hilfe des World Wide Web Consortium entwickelt²:

- Hypertext Markup Language (HTML)
- Extensible Markup Language (XML)
- Extensible Hypertext Markup Language (XHTML)
- Cascading Style Sheets (CSS)
- Portable Network Graphics (PNG)
- Scalable Vector Graphics (SVG)
- Synchronized Multimedia Integration Language (SMIL)
- Mathematical Markup Language (MathML)

Allen Standards des W3C gemeinsam ist, dass sie frei von Patenten und kostenlos für jedermann zugänglich sind. Mitglieder des W3C sind u.a. die Firmen Novell, HP, IBM und Microsoft.

¹World Wide Web Consortium

²aus wikipedia.org

Textauszeichnungssprachen

Bei den Standards HTML, XML und XHTML handelt es sich um Textauszeichnungssprachen (ML für Markup Language), die alle im Klartext, z.B. mit einem einfachen Texteditor lesbar sind. Je nach eingebetteten Befehlen (Tags) wird der Text dann von einem Internetbrowser bzw. einem anderen Programm nach bestimmten Vorschriften formatiert und dargestellt. Diese Vorschriften, z.B. in einem Cascading Style Sheet (CSS) formuliert, sind so kostenlos und frei zugänglich wie die deutsche oder französische Grammatik. Stellen Sie sich vor, Sie müssten für den Gebrauch der deutschen Grammatik bezahlen!?

Was ist XML

Die Extensible Markup Language, abgekürzt XML, ist ein Standard zur Erstellung

maschinen- und menschenlesbarer Dokumente in Form einer Baumstruktur. XML definiert dabei die Regeln für den Aufbau solcher Dokumente (siehe auch wikipedia.org). Diese Regeln eines XML-Dokuments (Zulässige Elemente, Art der Daten usw.) werden in einer so genannten DTD (Document Type Definition) oder einem XML-Schema ebenfalls wieder im Klartext nach entsprechenden weiteren Regeln beschrieben. Besonders wichtig ist bei XML, dass jede Auszeichnung (z.B. ein Abschnitt, zwischen ein öffnendes und ein schließendes Element eingebettet wird).

Hervorzuheben ist vor allem, dass im Gegensatz zu HTML nicht das Aussehen (groß, fett ...) sondern die logische Struktur (Autor, Produktname, Anmerkung...) durch die so genannten Tags beschrieben ist.

Hier ein Beispiel für ein wohlgeformtes XML-Dokument:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<aufsatz>
  <ueberschrift>Soll man XML lernen?</ueberschrift>
  <einleitung>
    Warum schon wieder etwas Neues, jammert der geplagte Computerfreak.
    Ist es denn sinnvoll, sowas wie XML zu lernen, wo ich doch so mit
    Word zufrieden bin?
  </einleitung>
  <hauptteil>
    <pro>
      Lernen ist gesund fuer die grauen Zellen. Mein XML ist in 50 Jahren
      noch lesbar.
    </pro>
    <contra>
      Soll ich diesen immensen Aufwand mit dem neumodischen Zeug betreiben?
      Die Belastungen steigen doch immer weiter und
      die Schueler kapieren das sowieso nicht so leicht.
    </contra>
  </hauptteil>
  <schluss>
    Wenn du wissbegierig bist und dich gerne mit Logik und Strukturierung
    beschaefstigst ist es gut, sonst Finger weg lassen, oder halt ein
    Programm nehmen, welches automatisch so was macht!
  </schluss>
</aufsatz>
```

Die Auszeichnungen wie z.B. „aufsatz, ueberschrift, einleitung ...“ sind frei gewählt und in der Dokumenttyp-Definition (DTD) erklärt, damit das Dokument gültig ist. Natürlich wäre es

lästig, für jedes Dokument eine eigene Definition zu schreiben. Deshalb muss eine Definition für eine bestimmte Dokumentenart, hier z.B. die Dokumentart „Erörterung“ lediglich einmal als separate Datei erstellt und abgespeichert werden. Anschließend kann sie jederzeit wieder verwendet werden. Noch einfacher ist die Verwendung bereits fertig erhältlicher DTDs wie z.B. die DocBook-DTD für technische Dokumente.

Die DocBook-DTD

Ein Beispiel für ein DocBook-Dokument:

```
<?xml version="1.0" standalone="no"?>

<!DOCTYPE article SYSTEM "docbookx.dtd">

<article lang="de">

  <articleinfo>

    <author>
      <firstname>Walter</firstname>
      <surname>Schlenker</surname>
      <affiliation>An LS-Abgeordneter</affiliation>
    </author>
    <title>DocBook</title>
    <subtitle>XML: mit DocBook und OpenOffice</subtitle>

    <abstract>
      <para>Den SGML- und XML-Dokumenttyp DocBook
        nutzen Programmierer und andere fuer die Erstellung technischer Dokumente
      </para>
    </abstract>

  </articleinfo>

  <!-- ..... ein Kommentar ..... -->

  <section>
    <title>DTDs m"ussen lokal vorliegen</title>

    <para>Dies alles selber zu schreiben ist furchtbar, ...
      deshalb verwendet man eine <acronym>DTD</acronym>.
    </para>

  <!-- ..... weitere Kommentare ..... -->

  </section>

</article>
```

Sicherlich sehen Sie, dass der Aufbau dieses Dokuments dem XML-Beispiel sehr ähnlich sieht, allerdings englische Ausdrücke verwendet werden. Die gültigen Elemente sind in der so genannten DocBook-DTD definiert und müssen nicht mehr extra aufgeschrieben werden.

Erstellen eines DocBook-Dokumentes

Export aus Textverarbeitungen

Word 2003 Professional, OpenOffice ab 1.1 und Abiword unterstützen teilweise den Export als DocBook-XML, wobei aber die logische Strukturierung mithilfe der Absatzformate vorgenommen werden muss und nicht immer zum gewünschten Ergebnis führt. Die besten Ergebnisse erhält man meist mit dem relativ unbekanntem und kostenlosen Abiword, da es eine logische Gliederung des Textes recht gut in DocBook-Tags umsetzen kann.

Einfache Editoren

Prinzipiell können mit jedem Texteditor alle XML- bzw. DocBook-Texte geschrieben werden, was jedoch höchstens für Übungszwecke oder ausgesprochene Kommandozeilen-Liebhaber sinnvoll ist.

Spezielle DocBook- und XML-Editoren

Die beste Lösung sind die so genannten XML-Editoren, welche die strukturierte Texteingabe auf verschiedenen Arten unterstützen (Anzeige der Baumstruktur, farbige Hervorhebungen bzw. Einrückungen ...). Sie reichen vom kostenlosen und schlanken Eingabetool bis zum hochprofessionellen Werkzeug wie z.B. XMetal, Altova XMLSpy oder Framemaker von Adobe.

Online-Anzeige mit Stylesheets

Da in XML lediglich die logische Struktur eines Dokuments festgelegt ist und noch keine Aussagen über eine Formatierung bestehen, müssen den Elementen, z.B. einem Titel, vor der Ausgabe am Bildschirm ihre Eigenschaften wie z.B. groß, fett oder kursiv zugewiesen werden. Dies kann durch eine Formatierungsanweisung, ein so genanntes Cascading

Style Sheet (CSS) geschehen, welches als separate Datei abgespeichert wird und worauf dann im XML-Dokument verwiesen wird.

Umwandlung in Ausgabeformate

Etwas mehr Aufwand ist notwendig, wenn ein XML-Dokument in ein anderes Format wie z.B. Text, HTML oder auch \LaTeX umgewandelt werden soll. Dies geschieht im allgemeinen mit der Transformationssprache XSL-T (eXtensible Stylesheet Language-Transformation). Das Formatieren als PDF oder PS geschieht dann mit Hilfe von XSL-FO (Extensible Stylesheet Language - Formatting Objects).

Automatisierung des Arbeitsablaufs

Prinzipiell können natürlich sämtliche Arbeitsschritte nach Installation der notwendigen Werkzeuge „von Hand“ auf der Kommandozeile (Eingabeaufforderung) ausgeführt werden, was aber doch etwas mühselig ist, für Übungszwecke aber durchaus seinen Sinn geben mag. Komfortabler und effektiver für den Profieinsatz sind meist komplette und weitgehend benutzergelieferte Entwicklungsumgebungen, die Sie z.B. unter dem Stichwort „XML-Entwicklungsumgebung“ im Internet finden können.

Bedeutung für den Unterricht

In erster Linie ist eine Beschäftigung mit dem Thema XML und DocBook für die Bereiche Digital- und Printmedien bzw. Informatik interessant. Aber auch bei der Auswahl einer Textverarbeitung, eines Office-Pakets oder einer beliebigen Anwendung zur Datenhaltung bzw. Datenpublikation sollte zumindest ein Blick auf zukunftsfähige Ausgabemöglichkeiten geworfen werden, auch wenn sie sich dezent im Hintergrund halten (siehe AbiWord und OpenOffice).

Walter Schlenker

□