Prof. Dr. Stefan Brass
Institut für Informatik
MLU Halle-Wittenberg

23. Mai 2016

Grundlagen des WWW
— 6. Übungsblatt: XML DTDs, HTML, Projekt —

Dieses Blatt enthält eine Hausaufgabe. Geben Sie diese bitte als PDF oder reine ASCII-Datei bis Montag, 30.05.2016, 12:15 per EMail an `brass@informatik.uni-halle.de` ab. Schreiben Sie bitte `www16` in die Betreff-Zeile.

Hausaufgabe:

a) Schreiben Sie einen Plan für Ihr Projekt auf 1–3 Seiten. Machen Sie eine (vorläufige) Liste der Webseiten, die Sie erstellen wollen, und schreiben Sie zu jeder Seite, was die Hauptinhalte dieser Seite sein sollen.

b) Wenn Sie schon eine Vorstellung von den Web-Technologien haben, die Sie einsetzen wollen, schreiben Sie das bitte auch.

c) Wenn es ähnliche Webseiten gibt (Konkurrenz oder Vorbilder für Ihre Seite), nennen Sie diese. Eventuell können dies ja auch inhaltlich ganz andere Seite sein, die Ihnen vom Aufbau oder der Optik her gut gefallen.

d) Listen Sie insbesondere auch Webseiten, die Sie früher bereits erstellt haben. Wenn Sie für dieses Projekt eine davon überarbeiten wollen, sagen Sie, was Sie verbessern wollen.

Präsenzaufgaben:
Die restlichen Aufgaben dieses Blattes sind Präsenzübungen, die nicht abgegeben werden müssen. Falls Sie allerdings die Übung verpassen, sollten Sie einen ausreichend großen Teil der Aufgaben bearbeiten und dem Dozenten per EMail schicken (mit “`www16`” in der Betreff-Zeile, bis spätestens 28.05.2016).

e*) Was würden Sie in einer mündlichen Prüfung auf folgende Fragen antworten?

- Was ist die Beziehung zwischen SGML und XML?
- Wie alt ist SGML ungefähr? Von wann ist der ISO-Standard?
- Was ist die Beziehung zwischen HTML und SGML?
- Wann ist die Unterscheidung zwischen Appearance-based Markup und Structure-Based Markup wichtig? Oder: Wozu kann man einen Text noch verwenden, außer ihn auszudrucken?
- Wie sehen SGML/XML-Dokumente aus? Was fällt da als erstes auf?
- Wie ist die genaue Syntax von einem Start-Tag? Von einem End-Tag?
- Was ist der Unterschied zwischen einem Element und einem Tag?

- Wie kann man sich ein SGML/XML-Dokument intern/abstrakt als Datenstruktur vorstellen?
- Müssen Start- und Endtags immer vorhanden sein?
- Warum gibt es in XML keine Tag Minimization?
- Wie schreibt man ein leeres Element in XML?
- Was sind Unterschiede zwischen XML und SGML bei der Attribut-Syntax?
- Wie sehen Zeichen-Referenzen aus? Wie sehen Entity-Referenzen aus?
- Wie sehen Kommentare aus?
- Wie deklariert man Elemente in SGML? Unterschiede in XML?
- Wie deklariert man ein Element, das reinen Text enthält?
- Warum braucht `#PCDATA` das `#`?
- Wie deklariert man ein leeres Element?
- Was ist ein Mixed Content Model? Wie sieht das in XML aus?

f) Sie wollen Ihre Sammlung von CDs mit klassischer Musik in einer XML-Datei erfassen. Schreiben Sie eine XML DTD und ein Beispieldokument mit folgenden Daten:

- Das Dokument soll zwei Komponisten enthalten: Antonio Vivaldi (1678–1741) und Johann Sebastian Bach (1685–1750). Sie wollen auf Vornamen, Nachnamen, Geburtsjahr, und Todesjahr getrennt zugreifen können. Eine Trennung mehrerer Vornamen wie im Fall von Bach ist nicht nötig. Vornamen und Lebensdaten sollen optional sein, aber jeder Komponist muss einen Nachnamen haben.
- Sie wollen folgende Stücke abspeichern: “Die vier Jahreszeiten” von Vivaldi, sowie drei Stücke von Bach: “Violinkonzert Nr. 1” (a-moll), “Violinkonzert Nr. 2” (E-dur), “Doppelkonzert” (d-moll). Zu jedem Stück gibt es einen Titel und optional eine Tonart. Wenn Sie wollen, können Sie einen eindeutigen Bezeichner (Stücknummer oder Ähnliches) einführen.
- Sie haben zwei CDs: “Bach Violinkonzerte” mit den beiden Violinkonzerten und dem Doppelkonzert, “Anne Sophie Mutter spielt Vivaldi und Bach” mit den vier Jahreszeiten von Vivaldi und dem Violinkonzert Nr. 1 von Bach. CDs sind uber Ihren Titel eindeutig identifiziert. Sie wollen zu jeder CD die enthaltenen Stücke erfassen. Es kann sein, dass das gleiche Stück auf mehreren CDs enthalten ist. Optional wollen Sie auch den Kaufpreis abspeichern. Die CD “Bach Violinkonzerte” kostet 9.99 Euro.

Achten Sie darauf, dass Sie die gleiche Information nicht redundant mehrfach erfassen. Grundsätzlich ist es natürlich wünschenswert, dass die XML DTD hilft, Fehler zu vermeiden (z.B. Verweise auf nicht existierende Stücke). Leider sind die Möglichkeiten hierzu sehr begrenzt. Validieren Sie Ihr Beispieldokument bezüglich Ihrer DTD.

g) Gegeben sei folgende stark vereinfachte DTD für XHTML:

```
<!DOCTYPE html [

<!ELEMENT html        (head, body)>
<!ATTLIST html        xmlns CDATA #IMPLIED>
<!ENTITY  % head.misc "(meta|link)*">
<!ENTITY  % block     "p|h1|ul">
<!ENTITY  % inline    "#PCDATA|em|img">
<!ELEMENT head        (%head.misc;, title, %head.misc;)>
<!ELEMENT title       (#PCDATA)>
<!ELEMENT meta        EMPTY>
<!ATTLIST meta        http-equiv NMTOKEN #IMPLIED
                      name       NMTOKEN #IMPLIED
                      content    CDATA   #REQUIRED>
<!ELEMENT link        EMPTY>
<!ATTLIST link        href CDATA #REQUIRED
                      rel (start|next|prev) #IMPLIED>
<!ELEMENT body        (%block;)+>
<!ELEMENT h1          (#PCDATA)>
<!ELEMENT p           (%inline;)*>
<!ELEMENT em          (%inline;)*>
<!ELEMENT ul          (li)+>
<!ELEMENT li          (%inline;|%block;)*>
<!ELEMENT img         EMPTY>
<!ATTLIST img         src     CDATA  #REQUIRED
                      alt     CDATA  #REQUIRED
                      height  CDATA  #IMPLIED
                      width   CDATA  #IMPLIED>
]>
```

Auf der nächsten Seite folgt ein XHTML-Dokument mit (mindestens) 6 Fehlern. Finden Sie diese Fehler! Nehmen Sie dabei an, dass die obige DTD die XHTML-1.1-DTD ist, was natürlich nicht stimmt.


```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
         "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta name="author" content="Gnhild Grndnk" />
    <title>Das Schweigen der Lemmata</title>
    <link rel=next href="autor.html">&Uuml;ber die Autorin</link>
  </head>
<body>
  <h1>Die List</h1>
  <P>
  Ein Bild <img src="held.jpg?width=42&height=23"></img>
  <ul><li>Ein Unterpunkt</li><li/></ul>
</body>
</html>
```

h*) Schauen Sie sich die HTML 3.2 DTD an:

[https://www.w3.org/MarkUp/Wilbur/HTML32.dtd]

Geben Sie mindestens eine `ELEMENT`-Deklaration an, die Sie schon verstehen, und beschreiben Sie in natürlicher Sprache, was sie bedeutet (was ist das Inhaltsmodell?).

i*) Wenn man relationale Datenbank-Zustände in XML codieren will, was sind dann unschöne Einschränkungen von DTDs? Die Datenbank-Community hat später XML Schema als Ersatz für DTDs eingeführt, das wird aber erst in der Vorlesung “XML und Datenbanken” besprochen.