Projektarbeit: Dokumentsichten und Hyperlinks für die Datenmodellierung

Aufgabenstellung:

Ursprünglich sollte eine XML-DTD entwickelt werden mit der Vorlesungsskripten in verschieden Sichten dargestellt werden können, z.B. für Leser mit unterschiedlichem Kenntnisstand (Anfänger, Fortgeschrittene, Experten...), verschiedenen Zielen (Überblick, systematische Aufarbeitung,...), oder für verschiedene Verwendungszwecke (Papierversion zum Austeilen, Online-Version für die Präsentation, usw.).

Im Laufe der Arbeit stellte sich heraus, dass die Aufgabenstellung sehr allgemein und schwer zu konkretisieren war.
Deshalb wurde die Aufgabenstellung neu definiert:

- Das in HTML vorliegende Skript der Vorlesung "Techniken der Logikprogrammierung" sollte in XHTML umgewandelt werden.

- Dann sollten 2 konkrete Sichten realisiert werden: Vorlesungsskript, dass es erleichtert mit dem Skript zu arbeiten und Folien für die Präsentation der Vorlesung.

Vorgehensweise:

Als erstes haben wir das Skript mit Hilfe eines Umwandlungsprogramms (tidy) in XHTML umgewandelt. Das Ergebnis war nicht ganz so perfekt, denn an vielen Stellen im Dokument wurden end-Tags falsch gesetzt, so daß wir diese per Hand korrigieren mussten.
Um das Lesen oder das Bearbeiten des Quellcodes zu erleichtern, versuchten wir die semantische Bedeutung einzelner Textabschnitte zu klassifizieren. Da so gut wie nirgendwo auf Anhieb im Text zu erkennen war wo es sich, z.B. um eine Definition oder einen Satz handelte. War diese Aufgabe an manchen Stellen nicht ganz so einfach.
Dann haben wir div-Tags verwendet um die einzelnen Abschnitte von einander unterscheiden zu können und gaben dem class-Attribut einen Wert entsprechend ihrer semantischen Bedeutung, z.B. ...
Mittels eines CSS2-Stylesheets definierten wir unterschiedliche Eigenschaften für die verschiedenen Abschnitte z.B. eine Definition umrandet und blau hinterlegt.
Mit den Pseudo-Elementen :before und :after haben wir Textbestandteile im Stylesheet definiert, die an bestimmten Stellen und in immer der selben Form vorkommen sollen. Dadurch wir den Quellcode etwas reduzieren.
An einigen Stellen im Text gab es Verweise auf bestimmte Programmcodes, auf die wir dann einen Link gesetzt haben, so dass das Programm noch einmal in einem separaten Fenster angezeigt werden kann.
Dann kamen die Folien dran. Diese haben wir zunächst mit dem Slide-Maker vom W3C erstellt, da wir uns bei der Gestaltung an W3C-Folien orientieren sollten. Aber die so erstellten Folien waren nur nach HTML 4-Standard und außerdem der

Quellcode, der die Elemente zur Navigation zwischen den Folien enthielt, sehr umfangreich. Darum entschieden wir uns auf den Slide-Maker zu verzichten und die Folien mit Hilfe der Pseudo-Elemente :before und :after zu gestalten, was erhebliche Einsparungen im Quellcode ergab.

Als nächstes suchten wir nach einer Möglichkeit um das Wechseln zwischen den beiden Sichten zu ermöglichen.
Zuerst haben wir zwei Frames verwendet. In dem einen sollte der eigentliche Inhalt der jeweiligen Sichten angezeigt werden und das andere sollte ein Navigationsfenster für den Inhalt bereitstellen, mit Buttons für das Wechseln der Sichten. Dafür wären zwei getrennte Dateien für die beiden Sichten nötig gewesen.
Aus Gründen der besseren Wartbarkeit (z.B. bei Änderungen des Inhalts der Vorlesung), deshalb haben wir uns dafür entschieden, daß ein Dokument beide Sichten enthalten soll.

So dass ein Dokument folgende Form hat:

```
<div class="chapter">
        <div class ="section">
                <div class="fulltext">…</div>

                <div class="slide">…</div>
        </div>
</div>
```

Die nächste Idee die wir hatten war, dass wir Teile bzw. Tags des Dokuments ausblenden. Wir versuchten zuerst dies mit Java-Skript zu verwirklichen. Befehle hierzu waren aber nur im Internet Explorer implementiert. Um das selbe Ergebnis mit anderen Browsern zu erreichen müßten wieder andere Befehle in der HTML-Datei implementiert werden. Eine Skript-Version, die jedoch nur auf einem Browser dargestellt werden kann bzw. für jeden Browser eine eigene Version erschien uns nicht sinnvoll.

Im Standard von XHTML ist die Verwendung von unterschiedlichen Stylesheets für ein Dokument vorgesehen. Dies ermöglicht es über einen Menüpunkt im Browser den Stylesheet und somit die Sicht zu wechseln. Diese Variante ist zwar nicht so komfortabel wie wir es uns gewünscht hätten und wird außerdem bisher nur von den Browsern Netscape Navigator ab Version 6.0 und Mozilla ab Version 5.0 unterstützt.

Wünschenswert für die Zukunft wäre eine breitere Unterstützung dieses W3C-Standards durch andere Browser. Desweiteren wäre es vorteilhaft den Standard dahingehend zu erweitern, dass für das Zieldokument eines Links der zu verwendende Stylesheet angegeben werden kann. In der momentanen Form wird jede neu aufgerufene Seite immer mit dem Standard-Stylesheet angezeigt.