

Klaus Prator

Topologie und Navigation

Zur Bewegung in elektronischen Editionen

In Texten bewegen

Wir bewegen uns in Texten, in gedruckten wie in elektronischen – und wir bewegen uns in beiden auf verschiedene und doch auch auf ähnliche Weise. Die typische Bewegung in Büchern und Zeitschriften ist das seitenweise Blättern, das erfolgt, nachdem wir eine Seite zeilenweise zu Ende gelesen haben. Aber wir überspringen auch Absätze oder wir blättern zurück, weil uns ein Zusammenhang entgangen ist oder wir etwas vergessen haben. Das zeilenweise Lesen wird unterbrochen, wenn uns ein Fußnotenverweis zum unteren Seitenrand lenkt. Wir suchen nach einer bestimmten Seite, auf die ein Querverweis Bezug nimmt, oder wir blättern zum Ende des Buches, um eine Endnote oder einen Literaturverweis aufzusuchen. Umgekehrt benutzen wir Inhaltsverzeichnisse und Indizes, um uns zu bestimmten Seiten lenken zu lassen. Für viele Druckwerke sind diese zielgerichteten Bewegungen charakteristischer als das serielle Blättern. Man denke an Lexika, Wörterbücher und Rezeptsammlungen. Schließlich gibt es, zum Beispiel in Gestalt von Literaturverweisen, Bewegungen über die Dokumentgrenzen hinweg, die traditionell als Gang zum Bücherregal oder zur Bibliothek realisiert werden.

Alle diese Bewegungsweisen finden sich dem Grundsatz nach auch in elektronischen Texten, freilich werden sie dort anders realisiert. Die Realisierungsweisen sind insgesamt vielfältiger als im Druck und sie verändern sich mit der technischen Entwicklung. Gegenwärtig werden sie hauptsächlich definiert von den Fähigkeiten grafischer Benutzeroberflächen, die genauer vielleicht als symbolische Benutzeroberflächen zu bestimmen sind. Der Übergang von zeichenorientierten zu grafischen Oberflächen bezeichnet den technischen Schritt von der starren Wiedergabe jeweils eines Zeichens in einem Bereich des Bildschirms zu beliebig formbaren Zeichen und Grafiken auf einer frei zu gestaltenden Pixelfläche. Symbolische Handhabung steht im Gegensatz zu kommandoorientierter und zeigt sich zum Beispiel, wenn ein Text nicht mehr, wie in alten oder sehr einfachen Editoren üblich, durch Eingabe von Befehlen in einer eigenen Kommandozeile manipuliert wird, sondern indem Zeichen oder Wörter in Abhängigkeit von der Cursorposition durch Tastenkombinationen gelöscht oder ersetzt werden. Das gab es auch schon in zeichenorientierten Systemen. Aber die grafischen Oberflächen haben die Möglichkeiten symbolischer Handhabung natürlich beträchtlich erweitert. Schaltflächen und Hypertextverknüpfungen haben den Mausclick mittlerweile zur typischen Auslösung für die Bewegung in elektronischen Texten gemacht, vereint mit anderen Mausbewegungen wie dem Scrollen mit

Hilfe eines Balkens, der die Position im Text symbolisiert. Daneben gibt es aber weiterhin auch die Bewegung durch die Betatigung von Tastenkombinationen und auch schon die Moglichkeit, auf beruhungsempfindlichen Displays durch Antippen oder sogenannte Gesten, also Bewegungen auf der Oberflache, Bewegungen im Text auszulosen. Vom Prinzip her mag das als geringfugige Neuerung erscheinen, aber die direkte, nicht durch die Maus vermittelte Manipulierung der Oberflache verandert die Anmutung des Textumgangs betrachtlich.

Nicht nur die Auslosung hat sich verandert, sondern auch die Bewegung selbst, namlich hin zu groerer Unmittelbarkeit. Das Anklicken fuhrt in der Regel direkt zum angezielten Textbereich. Der Unterschied beeindruckt naturlich am starksten bei der Verlinkung zu raumlich entfernten externen elektronischen Dokumenten. Aber auch innerhalb des eigenen Dokuments hat der direkte Sprung nicht nur den Vorteil der Bequemlichkeit. Er entbindet den Benutzer auch von der vermittelten Bezugnahme auf Seitenzahlen oder Zitatkurzeln, die uns im gedruckten Werk zu der gewunschten Stelle fuhren. Realisiert wird das normalerweise durch das Einbringen von Ankeren in den Text, auf die in Verweisen Bezug genommen werden kann.

Wie die Bewegung im elektronischen Text durch mehr Unmittelbarkeit, so ist ihr Resultat durch groere Vielfalt gekennzeichnet. Wo immer meine Bewegung im gedruckten Text auch hinfuhren mag, sie endet immer an einem definierten Stuck eines unveranderten und unveranderbaren Textes. Im elektronischen Fall ist das nicht ausgemacht. Naturlich kann man einen elektronischen Kommentar ganz nach dem Vorbild des gedruckten gestalten und zum Beispiel fest am unteren Rand einer Seite unterbringen. Ich kann aber auch einen von Anmerkungen unbelasteten Text produzieren und diese erst auf Anforderung einblenden, sei es in einem frei gehaltenen Bereich oder in einem eigens geoffneten Fenster (Popup). Sie konnen auch direkt im Flietext auftauchen oder als Sprung in einen speziellen Anwendungsteil realisiert werden. Fur andere Navigationselemente gilt das entsprechend. Angesichts dieser gewaltigen Vielfalt der Realisierungsmoglichkeiten ist es erstaunlich, da die realisierten Funktionalitaten sich weit weniger unterscheiden. Weitgehend sind es die gleichen, die auch in gedruckten Werken, wenn auch in eingeschrankter Weise, zur Verfugung gestellt werden. Eine Neuerung bleibt allerdings zu nennen: Die Volltextsuche in elektronischen Texten hat keine Entsprechung bei den gedruckten. Da dies aber ein ganz eigenes Thema darstellt, lassen die folgenden uberlegungen Suchverfahren auer Betracht und beschranken sich auf Navigationen in einem engeren Sinn, namlich solche, die vom Autor explizit zur Verfugung gestellt werden. Freilich ist zuzugeben, da es zwischen beiden Bereichen funktionale uberschnidungen geben kann.

Schichten elektronischer Texte

In dem einleitenden Vergleich wurde die Bewegung in Texten ausschlielich aus der Sicht des Benutzers betrachtet. Aus Sicht des Produzenten stellt sich die Lage anders dar – namlich komplizierter. Ich will das an einem kleinen Beispiel veranschaulichen, das aber deutlich machen soll, welche Ebenen bei der Produktion elektronischer Texte

generell unterschieden werden müssen. Elektronische Editionen sind dabei das Beispiel, an dem sich die Überlegungen orientieren.

Das wirklich sehr kleine Beispiel ist die Ziffer zu Beginn einer Anmerkung, vielleicht eher eine Reminiszenz an gedruckte Fußnoten als unbedingt erforderlich:

Oberfläche: ³ ...

Was der Benutzer sieht, ist eine Ziffer, in unserem Beispiel¹ eine 3 in einer bestimmten grafischen Gestalt. Die Punkte deuten den nicht wiedergegebenen Inhalt der Anmerkung an.

Der Produzent arbeitet auf einer anderen Ebene, die hier Präsentationsschicht heißen soll. Dort sieht das Element so aus:

Präsentationsschicht (XML)

```
<note id="3">...</note>
```

Die Präsentationsschicht ist nicht mit der Oberfläche identisch. Sie enthält keinen Hinweis auf die grafischen Eigenschaften des Elements. Die Präsentationsschicht basiert auf XML, es könnte auch HTML sein, in jedem Fall auf einer Notation, die durch einen Webbrowser, eventuell auch durch ein anderes Programm, darstellbar ist. Die Ziffer 3 findet sich hier nicht als Inhalt des Tags – der ist die Anmerkung selbst –, sondern lediglich als Wert eines Attributs. An der Oberfläche wird sie sichtbar, weil dem Tag ein Eintrag in einem Cascading Stylesheet zugeordnet ist:

Stylesheet (CSS)

```
note:before {content:attr(id);font-weight:bold; display:block; padding-top:10pt; padding-bottom:10pt;}
```

Dieser Eintrag besagt zunächst, daß vor dem eigentlichen Inhalt des Tags, also der Anmerkung selbst, ein Element einzufügen ist, dessen Inhalt dem Attribut `id` entnommen werden soll – und genau dort findet sich ja in unserem XML-Tag die Ziffer 3. Es folgen dann einige Angaben zur grafischen Wiedergabe des Zeichens.

Stylesheets sind ein hervorragendes Mittel, die Anforderungen an die sachliche Auszeichnung eines Textes mit den Bedürfnissen nach grafischer Gestaltung zu vermitteln. Dies geschieht gerade dadurch, daß beide Bereiche entkoppelt und damit arbeitsteilig bearbeitbar werden.

Für viele Texte, besonders aber für kritische und wissenschaftliche Editionen ist es wichtig, für unterschiedliche Nutzerinteressen und über lange Zeiträume hinweg, geeignete Ausgaben zur Verfügung zu stellen. Die vorgestellte Präsentationsschicht ist dafür ungeeignet, weil sie zu sehr an eine bestimmte Art der Ausgabe gebunden ist. Sie wird deshalb abgeleitet von einer Archivschicht mit folgender Gestalt – ebenfalls in XML:

¹ Das Beispiel bezieht sich auf die elektronische Variante der Edition der Briefe an Jean Paul der Berlin-Brandenburgischen Akademie der Wissenschaften (Homepage der Arbeitsstelle: <http://jean-paul.bbaw.de>).

Archivschicht (XML)

```
<notep><pg>202</pg><li>6</li><jpzit>Krebse]</jpozit> ...</notep>
```

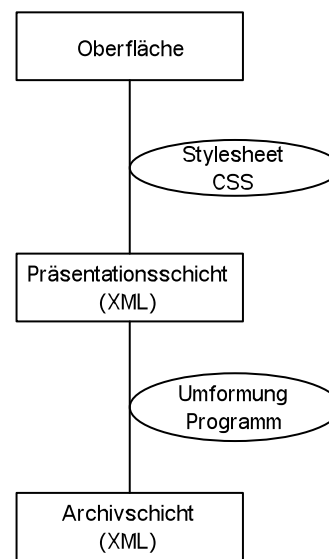
Auf dieser Ebene existiert die Ziffer 3 gar nicht mehr. Sie wurde erst von dem Umsetzungsprogramm erzeugt, das die Archivschicht in die Präsentationsschicht transformiert. Im Fall einer anderen Realisierung der Anmerkungen, z.B. als Popup, wurde sie nicht unbedingt benotigt. Dagegen finden sich auf dieser Ebene Informationen zu Seiten- und Zeilenzahl des Bezugs sowie ein Bezugslemma, die auf der Oberflache nicht auftauchen. Von ihnen wurde kein Gebrauch gemacht, aber es konnte sein, da diese Informationen zum Beispiel fur eine Bezugnahme auf eine Druckfassung gewunscht werden. Deshalb werden sie aufbewahrt und konnen gegebenenfalls sichtbar gemacht werden.

Hier sind die Schichten noch einmal in ihrem Zusammenhang dargestellt. Der Gang der Erlauterung ging von oben nach unten. Die Produktion erfolgt aber naturlich in gegenlaufiger Richtung.

Topologie und Navigation

Unterstellt man einstweilen den Sinn der Unterscheidung dieser Ebenen, so liegt die Frage nahe, auf welcher Ebene denn nun die Navigation anzusiedeln sei. Aus unserem winzigen Beispiel ist sofort zu sehen, da die Realisierung der Navigation gerade beim ubergang von der Archiv- zur Präsentationsebene stattfindet. Andererseits ist intuitiv klar, da die Navigation abhangig ist von den Strukturen, die auf der Archivebene zur Verfugung gestellt werden. Denn gabe es hier beispielsweise keine Auszeichnung von Anmerkungen, konnten sie auf einer daruber liegenden Schicht auch nicht realisiert werden. Dies gibt Anla zu einer Unterscheidung, namlich der von Topologie und Navigation. Beide verhalten sich wie Moglichkeit und Realisierung. Eine Topologie stellt potentielle Verbindungen von Dokumentknoten bereit, eine Navigation realisierte Verknupfungen innerhalb eines Hyperdokuments oder eines anderen Informationssystems. Im Bild eines offentlichen Verkehrssystems gesprochen entspricht die Topologie dem Schienen- oder Straennetz, die Navigation dagegen den tatsachlich existierenden Zug- oder Busverbindungen. Letztere werden immer nur einen Teil der Moglichkeiten ausschopfen. Umgekehrt konnen die Verkehrsmittel nur dort fahren, wo die entsprechende Infrastruktur existiert.

Topologie und Navigation unterscheiden sich nicht nur durch den Modus von Potentialitat und Aktualitat. Zur Erstellung von beiden sind unterschiedliche Fahigkeiten vonnoten. Bei der Erstellung einer Topologie stehen begriffliche und sachstrukturelle



Kenntnisse im Vordergrund. Hier geht es zum Beispiel um eine Fachtaxonomie oder um die inhaltliche Struktur einer Edition. Die Topologie beruht auf sachlicher Auszeichnung des Dokuments sowie auf Metainformation. Die Navigation hat die technische Realisierung der Nutzerinteraktion und die ästhetischen Anforderungen an sie zum Gegenstand. Die entsprechenden Anforderungen erstrecken sich von der Programmierung bis zur Erforschung von Mensch-Computer-Interaktion. Und beide sind auf verschiedenen Ebenen anzusiedeln. Die Topologie ist auf der dauerhaften Archiv-ebene zu verankern, die Navigation wird erst auf der Präsentationsebene realisiert.

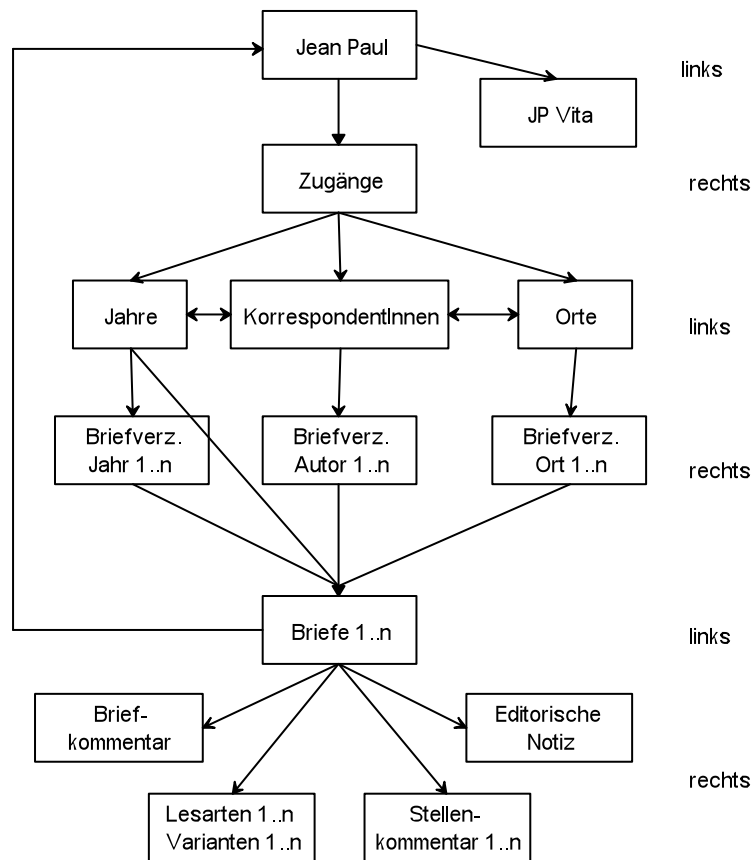
Die Idee der sachlichen Textauszeichnung ist zweifellos eine ganz wesentliche Voraussetzung für das Zusammenspiel der Ebenen unter den Randbedingungen von Dauerhaftigkeit und Flexibilität. Wie unschwer zu sehen, folgt auch der hier verfolgte Ansatz diesem Paradigma, die Textgrundlage unabhängig von Fragen der endgültigen Gestaltung vorzunehmen und die grafische Oberfläche erst im letzten Schritt der Realisierung hinzuzufügen. Gleichwohl seien zwei Fragen aufgeworfen zu Punkten, die in der allgemeinen Diskussion allzu selbstverständlich vorausgesetzt werden.

Zum einen: Gibt es *die* sachliche Auszeichnung eines Textes? Die Frage stellen heißt sie verneinen. Zweifellos hatte diese Rede in den Anfangszeiten der Textauszeichnung einen recht problemlosen Sinn. Der Autor eines Textes sollte die grafischen Hervorhebungen, die er in seinem Manuskript anzubringen gewohnt war, durch sachliche Textauszeichnungen ersetzen. Mittlerweile werden Unmengen von Texten im Nachhinein unter den unterschiedlichsten Interessen mit Textauszeichnungen versehen und die Rede von der sachlichen Auszeichnung ist nun weit weniger selbstverständlich. Es gibt das berechtigte Interesse an linguistischer Auszeichnung und bei kritischen Editionen kann es vorkommen, daß gerade die grafischen Eigenschaften eines Manuskripts (Tintenfarben etc.) Teil der sachlichen Textauszeichnung werden.

Die erste ist aber mehr die Vorbereitung der zweiten Frage, die für unseren Zusammenhang noch wichtiger ist und die ich die nach der Ausschließlichkeit der Dichotomie zwischen sachlicher Auszeichnung und grafischer Gestalt nennen möchte. Wenn diese Dichotomie gilt, dann sind alle Erzeugnisse auf der Basis einer dauerhaften XML-Textgrundlage nur verschiedene grafische Gestaltungen. Dies entspricht in der Tat gegenwärtiger Redeweise. Auch die Ausgaben in unterschiedlichen Medien werden als solche Varianten betrachtet und auch Systeme mit vollkommen unterschiedlichen Zugangs- und Handhabungsweisen erscheinen als solche. Damit werden komplexe, sinntragende Strukturen, teilweise auch ganz eigene inhaltliche Elemente mit Farbgebungen und Schriftarten gleichgesetzt. Dieser Denkart wird jede Umformung zum Stylesheet und dementsprechend sind auch die entsprechenden Werkzeuge für XML als XSL-Tools (XSL für extended stylesheet language) bezeichnet wurden. Erst während der Arbeit bemerkten die Entwickler, daß sie eigentlich eine Umformungssprache für Markupssprachen geschaffen hatten, benannten sie in XSLT um (T für Transformation) und lieferten mit XSLFo das Produkt nach, das mit größerem Recht als Stylesheet betrachtet werden kann. Demgegenüber wird hier dafür plädiert, zwischen der dauerhaften Textauszeichnung und der grafischen Oberflächengestaltung

im engeren Sinn einen eigenen Bereich von textlichen Strukturen anzunehmen. Navigationsstrukturen bilden ihren wesentlichen Kern.

Nun sind solche Fragen abstrakt schwer zu entscheiden, und man mute sich nun eigentlich zumindest in den praktischen Umgang mit elektronischen Texten begeben oder noch besser in die Praxis ihrer Erstellung, um sie zu diskutieren. Das ist auf Papier nicht moglich, aber es soll doch versucht werden, dem Phanomen etwas naher zu kommen. Als Beispiel diene der Zugang zu einem Briefband aus dem in der ersten Funote erwahnten Jean-Paul-Projekt. Dem sehr fluchtigen Benutzer erscheint die Navigation ja nur als eine Folge von Mausklicks, die irgendwohin fuhren oder die neue Elemente auftauchen lassen. In der Regel folgen diese Wege aber einem Plan, zum Beispiel dem, der in der nachsten Abbildung zum Ausdruck kommt.



Ziel ist die Bahnung des Zugangs zu Briefen an *einen* Autor, und so werden bereits auf der Startseite die Zugangsarten, namlich nach Jahren, Orten und Korrespondenz-

partnern angeboten, als ein Seitenweg auch zu einer Vita Jean Pauls. Man gelangt dann zu einer Liste der Namen, Jahre oder Orte, zwischen denen direkt gewechselt werden kann. Die nächste Stufe sind dann die Briefverzeichnisse zum jeweiligen Bereich. Dies geschieht jeweils im Wechsel zwischen linker und rechter Hälfte des Fensters, so daß die übergeordnete Ebene immer noch im Zugriff bleibt. Weiter höher liegende Ebenen sind durch zusätzliche Verlinkungen erreichbar. Der nächste Schritt führt dann zu den Briefen und der übernächste zu Briefkommentar, editorischer Notiz, Stellenkommentaren, Lesarten und Varianten.

Nun könnte die gleiche Textgrundlage auch in anderen Zusammenhängen verwendet werden. Archivebene und Topologie blieben dieselbe, die Navigation müßte sich aber sinnvollerweise beträchtlich verändern. Man denke an

- einen (in Auswahl bereits realisierten) erweiterten Briefwechsel mit Gegenbriefen und Zusatzmaterial. Hier müssen natürlich die Gegenbriefe in die Navigation einbezogen und die Materialien den jeweiligen Personen beziehungsweise Briefen zugeordnet werden.
- Briefe im Rahmen einer Darstellung von Person und Werk Jean Pauls. In diesem Fall könnte der zentrale Zugang über die Vita führen und von dort weiter zu Briefen und Werken, deren wechselseitige Bezüge auch zu modellieren wären.
- Für literarische Korrespondenzen um 1800 wäre der eben genannte Ansatz ganz ungeeignet, weil er einen Autor ins Zentrum stellt. Für einen multizentrischen Ansatz müßte die Navigation so verändert werden, daß sie mehr ist als die Addition verschiedener monozentrischer Autorenkorrespondenzen.

Die Reihe ließe sich fortsetzen. Vielleicht ist aber schon deutlich geworden, daß es sich bei diesem Geschäft doch um etwas ganz anderes handelt als die grafische Gestaltung der Oberfläche, so wichtig diese auch sein mag. Ich widerstehe der Versuchung, weitere Beispiele anzuführen (Die Darstellungsmöglichkeiten varianter Texte wären sehr reizvoll.) und auch der, diese gar systematisieren zu wollen. Stattdessen beschränke ich mich auf zwei unverzichtbare Ergänzungen vor einem abschließenden Wort zur praktischen Umsetzung.

Metainformation

Am Beginn des letzten Abschnitts war es ganz kurz schon erwähnt worden: Die Topologie beruht auf sachlicher Auszeichnung des Dokuments *und* auf Metainformation. Das Markup allein reicht nicht aus. Im angeführten Beispiel erfolgte der Zugang zu den Briefen über die Verfasser, über Zeitpunkte und Orte des Briefschreibens. Das sind aber Zusatzangaben über die Briefe – Metadaten oder, wenn sie Kontexte bilden, Metainformationen. Zwar werden sie häufig auch im Brief erwähnt, aber nicht in allen Fällen, nicht immer vollständig und richtig und nicht in normierter Form. Häufig finden sich Metadaten gar nicht im Dokument, zum Beispiel Manuskriptbeschreibungen oder Schlagwörter aus bibliothekarischen Klassifikationen. Im Beispiel werden die Metadaten in einer Datenbank gehalten. Es gibt aber auch gute Gründe sie jeweils mit dem elektronischen Dokument abzuspeichern, zum Beispiel im Header-Bereich

von XML. Seit einiger Zeit gibt es auch ein eigenes Format fur Metadaten, das Resource Description Format RDF. Es ist XML-kompatibel, was bedeutet, da es in XML darstellbar ist, und das ist auch die gebrauichlichste Form. Es lat sich aber auch in anderen Formaten umsetzen. In RDF werden nicht so sehr einzelne Objekte reprasentiert, sondern Sachverhalte, d.h. Zusammenhange zwischen Objekten oder Konzepten.

Metadaten sind keine Spielerei, sondern werden seit langem benutzt, wenn man sich Zugang zu Informationen verschaffen und diese dann verarbeiten will. Ein klassisches Beispiel sind die bibliographischen Informationen, mit denen bibliothekarische Bestande verwaltet und suchbar gemacht werden. Nun konnte man sich beruhigt zurucklehnen und feststellen, da wieder mal ein modisches Wort fur eine alte Sache gefunden wurde. Das ist aber nicht ganz richtig.

Ursprunglich waren Metadaten ausschlielich fur die Benutzung durch den Menschen vorgesehen, wahrend jetzt ihre Auswertung durch Computer im Zentrum der Aufmerksamkeit steht. Computergestutzt verarbeitet werden die Metadaten schon seit einiger Zeit, aber immer noch im Hinblick auf die menschliche Nutzung. Der angestrebte neue Schritt besteht darin, da – wie vielfach gesagt wird – der Computer die Daten versteht, oder – wie ich lieber sage – ein Teil der bisher vom Menschen erbrachten Systematisierungs- und Ordnungsleistungen nun vom Computer geleistet wird. Anwendungsbereiche liegen in der Informationswiedergewinnung (information retrieval), der Katalogerstellung, Programmierung von Agenten etc. Dafur sind besondere Formen der Darstellung und Speicherung dieser Daten notwendig. RDF wird haufig im Zusammenhang mit der Idee des Webgrunders Berners-Lee fur ein Netz des Wissens unter dem Schlagwort semantic web genannt. In dieses konnen auch Ontologien, also Systeme begrifflicher Zusammenhange, einbezogen werden, die die Navigationsmoglichkeiten in elektronischen Dokumenten wesentlich erweitern konnten.

Bezugnahme und Zitierbarkeit

Wahrend schon erwahnt wurde, da die Bezugnahme innerhalb von elektronischen Texten sehr einfach ist, weil man an beliebigen Stellen Anker setzen kann, die dann das Ziel von Links bilden konnen, wurde verschwiegen, da es in diesem Zusammenhang auch groe Schwierigkeiten gibt. Ein Grund liegt darin, da dieses Verfahren nur moglich ist, wenn ich selbst auf das Dokument zugreifen kann. In anderen Fallen hat das HTML-Web den Nachteil, da es von auen nicht adressierbar ist. Im Zusammenhang mit der Etablierung von XML hat man in den XPath-Ausdrucken eine Losung gefunden. Da XML-Elemente immer vollstandig in anderen enthalten sein mussen, bilden sie eine Baumstruktur, den sogenannten DOM-tree. Damit lat sich jede Stelle im Dokument als ein Ast dieses Baumes beschreiben. Der Ast ist nichts anderes als die Kette der jeweils ubergeordneten Markierungen (Tags). Der praktische Umgang damit ist nicht ganz so bequem wie mit den Ankern. Computer konnen es besser als Menschen.

Dem Menschen bleibt das Problem der Zitierbarkeit. Genau genommen verbergen sich dahinter zwei Probleme. Wir sind es gewohnt, wissenschaftliche oder literarische Werke nach Seiten und Zeilen zu zitieren. In elektronischen Dokumenten sind diese häufig nicht vorhanden und wenn doch, stimmen sie nicht mit einer parallelen Druckausgabe überein. Mit Zitierbarkeit ist dann auch meistens gemeint, daß der Bezug zu dieser Druckausgabe nicht herstellbar ist, in der vorurteilsvollen, aber möglicherweise richtigen Annahme, dem gedruckten Text gehöre die Priorität. Es bleibt aber auch die Frage, wie ich denn auf eine Stelle eines elektronischen Textes von außerhalb Bezug nehmen kann. Mit der gängigen Antwort auf die erste ist die zweite dann allerdings auch mit beantwortet. Die Antwort heißt: Ernstzunehmende wissenschaftliche elektronische Editionen müssen seiten- und zeilenidentisch mit der Druckfassung sein.

Auch wenn man die Frage der Zitierbarkeit so ernst nimmt, wie sie es wirklich ist und die Bezugnahme auf die Druckfassung akzeptiert, kann das nicht die zukunftsweisende Lösung sein. Zum einen hieße es, die Leistungen einer avancierten Technik an die Grenzen einer beschränkteren zu binden. Zum anderen würde man gerade an einer so wichtigen Stelle wie der der Zitierbarkeit und Bezugnahme die ansonsten hoch gehaltenen Prinzipien der sachlichen Textauszeichnung aufgeben. Denn das Zitieren, zweifellos eine inhaltliche Angelegenheit, wird an ein Merkmal der grafischen Ebene gebunden, mit dem Zeilenfall darüber hinaus noch an ein zufälliges. Damit müssen wir doch nach einer Antwort auf die zweite Frage, nämlich die nach der Bezugnahme in elektronischen Texten, suchen. Die Einschränkung, daß es sich um eine inhaltliche Struktur handeln muß, weist auch schon den Weg zur Lösung. Eine denkbare wäre, den XPath-Ausdruck zu benutzen und soweit das um den automatisierten Umgang mit den Texten geht, ist das auch in Ordnung. Für das menschliche Zitieren ist er etwas unhandlich. Dafür gibt es aber eine andere, sehr alte Lösung. Statt nach Seiten und Zeilen sollte man nach Kapiteln, Abschnitten und Sätzen zitieren, wie es mit Paragraph und Satz bzw. Kapitel und Vers in Gesetzen und in der Bibel schon der Fall ist. Übergangsweise wären Seitenzählungen parallel zu halten, während umgekehrt künftige Druckwerke die Satzzählung mit anzugeben hätten.

Stylesheets und Templates

Diese Überlegungen sollen zum einen das Verhältnis von Topologie und Navigation in ihrem Zusammenhang mit dauerhafter Textgrundlage und grafischer Gestaltung grundsätzlich abklären. Zum anderen soll dies in einer Weise geschehen, daß die Art der technischen Realisierbarkeit erkennbar wird. Für die grafische Gestaltung waren Cascading Stylesheets empfohlen worden. Die Gründe lagen in der dadurch ermöglichten Arbeitsteilung und daneben auch in dem Prinzip „Keep it simple“. Wenn etwas mit CSS machbar ist – und mit CSS ist sehr viel zu machen –, sollte man es damit realisieren. Entsprechend legt sich für die Navigation der Einsatz von Templates nahe. Templates sind Schablonen einer Markupsprache, in die über Variablenausdrücke einzelne Textobjekte oder auch Listen von solchen eingebunden werden können. Auch hier ist der Vorteil, daß die Hauptaufgabe, nämlich eine Seitenstruktur mit den erforderlichen Navigationselementen zu erstellen, klar separiert ist von grafischer Gestal-

tung auf der einen Seite und erforderlicher Programmierung auf der anderen. Als Erluterungsbeispiel diene die Template Attribute Language, ursprunglich fur Zope und die Programmiersprache Python entwickelt, mittlerweile aber auch fur Java, Perl und PHP verfugbar. Eine Liste von Ortsnamen, wie sie oben im Beispiel fur die Zugangsnavigation benutzt wurde, sieht in dieser Darstellung so aus:

```
<ul>
  <li tal:repeat="Orte">
    <a tal:content="item" href="datei.html"
      tal:attributes="href string:$(item).html"
      class="name" target="right">EinOrt</a>
  </li>
</ul>
```

„Orte“ ist eine Liste, die der Datenbank mit Metadaten entnommen ist, und fur jedes Element der Liste wird nun ein Ankerelement <a> erzeugt. Der jeweilige Ortsname wird als Inhalt anstelle des Platzhalters „EinOrt“ eingefugt. Dieser Inhalt des Items wird daruber hinaus mit der Endung html verkettet und als Wert des Attributs href eingefugt. Das Charakteristikum dieser Darstellung ist das Einbetten der Variablen in Attribute des Namensraums tal. Der Vorteil ist die konfliktfreie Bearbeitung des Codes in gangigen HTML-Attributen und darauf aufbauend wiederum eine problemlose Arbeitsteilung zwischen dem Ersteller der Seitenstruktur und einem Programmierer.

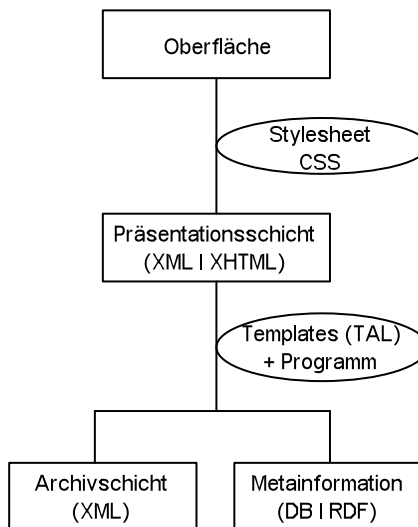
Die „Programmierung“ der Ortsliste beschrankt sich hier auf eine einfache Datenbankabfrage, die in Prolog zum Beispiel so aussahe:

```
setof(Ort,briefe(_,_,Ort,_),Orte).
```

Vielfach wird die Seitenstruktur und teils auch die grafische Gestaltung mit in die Programmierung einbezogen. Das hat den Nachteil, da Grafiker und Layouter nicht unabhangig vom Programmierer nachtragliche anderungen vornehmen konnen. Templates konnen auch direkt in Serverpages eingesetzt werden, so da eine Erzeugung der Webseiten zur Laufzeit ohne zusatzlichen Aufwand moglich ist.

Insgesamt ergibt sich damit der rechts dargestellte Gesamtzusammenhang.

Die Aufteilung in Templates und eventuelle Programmierung einerseits und Stylesheets andererseits unterstutzt auch die Einteilung gem dem



hier vorgeschlagenen Schichtenmodell optimal. Die ersteren erzeugen die Präsentationsschicht einschließlich Navigation und orientierenden Elementen, die zweiten beschränken sich auf die im engeren Sinn grafische Gestaltung.

Diese Art der Schnittstellenbildung dient nicht nur der besseren Handhabung eines einzelnen Projekts. Sie unterstützt auch die Integration verschiedener Projekte sowie die Herausbildung entsprechender Standards und weckt die Hoffnung, daß elektronische Editionen zu einer ganz neuen Art von Informations- und Kommunikationsplattform zusammenwachsen.