

Dynamische Verweise in Hypertexten

Zur Verwendbarkeit von Metaphern bei der Hypertextproduktion

Guido Ipsen
Kassel

Hypertext ist ein Informationsmedium, welches über eine äußerst komplexe Struktur verfügt. Obwohl dieses Faktum und die Tatsache, daß Hypertext nonlinear organisiert ist, gemeinhin akzeptiert sind, zögern sowohl Forscher als auch Autoren, die hypertextuelle Struktur mit Metaphern des Raumes oder der Bewegung zu ordnen. Das Problem besteht darin, daß das Verhalten von Nutzern nicht vorhergesehen werden kann und daher Metaphern, die z. B. in gedruckten Texten üblich sind, etwa „siehe oben“ oder „wie Sie bereits gesehen haben“, zu irreführenden Fallen werden können, wenn sie in Hypertexten auftauchen, da der Benutzer das Ziel, auf das sich der Verweis richtet, mangels räumlicher Orientierung nicht versteht, dort noch gar nicht war, etc. Dieser Aufsatz verfolgt die These, daß solche Probleme durch konsequenten Gebrauch der Mittel, die uns Hypertext und darin eingebundene Skriptsprachen an die Hand geben, umgangen werden können. Daher ist der Gebrauch von Metaphern in entsprechend organisierten Hypertextknoten durchaus möglich. Solche Hypertextknoten sollten als dynamische Einheiten erstellt werden, die sich nach dem Verhalten der Benutzer ändern können.

1 Hypertextproduktion – Hypertextverstehen

Das Thema dieses Aufsatzes ist die Verwendung von Metaphern und deiktischen Mitteln, um Hypertextdatenbanken transparenter, leichter nutzbar und besser navigierbar zu gestalten. Im Verlaufe der Arbeit wird auch das Für und Wider der Verwendung genannter Mittel eruiert; zunächst wende ich mich allerdings der generellen Fragestellung nach Produktion und Verstehen von Hypertexten zu. Die Transparenz der Organisation einer Hypertextdatenbank trägt entscheidend zu ihrer Verwendbarkeit bei. Aber auch die Fassung der einzelnen *Textchunks*, d. h., der einzelnen Seiten in der Datenbank, ist von Bedeutung. Beschäftigen wir uns daher zunächst mit der mit Texten befaßten Verständlichkeitsforschung. Verstehen und Produktion von Texten als *Transition* von Sprache in und aus dem transportierenden Medium gehören als Grundthese dieses Aufsatzes zusammen. Meutsch (1989) gibt uns einen guten Überblick über das Feld und formuliert gleichzeitig für uns sehr nützliche Thesen, denn er beschreibt Methoden zur Text- und Bildoptimierung. Nach Meutsch ist Anschaulichkeit und nicht primär die Struktur der

Satz- oder Wortebene entscheidend für die Verständlichkeit eines Textes (ebd., 12). Das Aufnehmen von Informationen aus Texten ist eine konstruktive und aktive Leserleistung (ebd., 13). Meutsch macht deutlich, daß es sich bei dem Objekt der Verständlichkeitsforschung um die interaktionale Beziehung zwischen Leser und Text handeln muß. Im Detail sagt er, es werde deutlich, „daß im Rahmen des kognitiven Konstruktivismus Textverständnis und Textverständlichkeit zwei eindeutig theoretische Frageperspektiven darstellen, die empirisch nicht voneinander zu trennen sind“ (ebd., 14). Dabei spielt der situative Kontext von Texten, also Ort und Zeit des Lesens, ebenso eine Rolle wie die Gedächtnisleistung des Lesers etc. Aber es gibt keine Patentlösung zur Verständlichmachung von Texten:

Es gibt keine Textverständlichkeit an sich! Statt dessen haben wir es mit einer Liste von Faktoren zu tun, die sensibel gegenüber Textsorten, gegenüber unterschiedlichen kommunikativen Absichten, gegenüber unterschiedlichen Lesergruppen und Leserzielen und gegenüber verschiedenen Nutzungszusammenhängen von Textinformation sind. All diese Faktoren können und sollten deshalb differenziert je nach Anlaß bei der konkreten Texterstellung berücksichtigt werden (ebd., 17).

Meutsch spricht zwar nicht über Hypertext, jedoch ist das Simulationsmodell zum Leseverstehen nach Kintsch und van Dijk (1978), welches er beschreibt, auch für das Verstehen von elektronisch vernetztem Text höchst interessant (Meutsch 1989, 17–19). In diesem Modell wird ein Text als eine Liste von *Propositionen* behandelt, wobei eine Proposition eine Bedeutungseinheit darstellt. Propositionen stellen Verbindungen zwischen Konzepten her. Da die Verstehenskapazität des Lesers begrenzt ist, werden die Listen von Propositionen, aus denen ein Text besteht, in Zyklen verarbeitet, die Kintsch und van Dijk interessanterweise *chunks* nennen. Hier besteht eine ungewollte Parallele der Terminologie zum Hypertext. Die Anzahl der in einem Zyklus verarbeiteten chunks variiert zwischen zwei und zwanzig. Die Komplexität der Oberflächenstruktur sowie der Bekanntheitsgrad der Textinhalte spielt eine entscheidende Rolle. Zunächst wird jeder chunk in einem sogenannten Arbeitsgedächtnis auf seine jeweilige Kohärenz überprüft. Da in jedem chunk nur bestimmte Propositionen für die Kohärenz des Textes entscheidend sind, werden die Propositionen zunächst in einem netzwerkähnlichen sogenannten Kohärenzgraph hierarchisiert. In einem zweiten Schritt verbleiben die als kohärenzstiftend erkannten Propositionen im sogenannten Arbeitsspeicher. Der erstellte Hierarchiegraph wird in das Langzeitgedächtnis überführt. Wichtig ist in jedem Fall, daß Menge und Auswahlstrategie der Propositionen immer von der Kapazität des Lesers abhängen. Das gesamte Modell ist also auf den Rezipienten ausgerichtet. Anzunehmen ist aber ein idealer Leser insofern, als daß er an Neuheit und Wichtigkeit der Information interessiert sein sollte. Dies stellt sicherlich auch für Hypertext eine nicht zu verachtende Größe dar. Es ist nämlich fest-

zustellen, daß Versuchspersonen in Tests oft die HTML-Seiten gar nicht komplett durchlesen, sondern recht schnell dem nächsten Link folgen. Dieses Phänomen, das sogenannte *Browsen* oder auch *Surfen* (ähnlich dem Kanalsurfen beim Fernsehen, wo der Zuschauer ebenfalls auf die Inhalte des Gesendeten wenig Gewicht legt), läßt aber auf einen sehr bescheidenen Grad von Interesse beim Leser schließen. Man muß also bei solchen Untersuchungen zwischen Lesern unterscheiden, die einfach nur in das Medium eintauchen, und solchen, die zielgerichtet an bestimmter Information interessiert sind.

Der Arbeitsspeicher ist nach dem Transfer des Graphen frei für die Aufnahme der nächsten Sequenz von Propositionen. Der zweite chunk wird allerdings bereits mit Hilfe der bereits aus dem ersten Durchgang gespeicherten Propositionen aufgearbeitet, die Liste der verketteten Propositionen im Arbeitsspeicher wird also länger. Dadurch wird auch über längere Textpassagen ermöglicht, daß wir Zusammenhänge erkennen können. Jeder chunk behält im Langzeitgedächtnis seinen eigenen Hierarchiegraphen, wodurch gewährleistet ist, daß bei Kohärenzlücken auf bereits aufgearbeitete Information zurückgegriffen werden kann. Nach Meutsch liegen die Vorteile dieses Modells in der Möglichkeit,

die Kapazitätsbeschränkungen des menschlichen Kognitionsvorganges, die Sequenti-
alität des Leseprozesses, thematische Wissensbestände der Leser, Ziele der Leser und
damit Verarbeitungsstrategien sowie ihre medialen Fähigkeiten (1989, 19–20)

abzubilden. Das Modell ist gerade daher so interessant für Hypertext, da dort tatsächlich Information bereits in kleineren Einheiten, nämlich den Hypertextchunks, vorliegt. Es erscheint sinnvoll, eine Hypertextdatenbank nach Kriterien zu gliedern, welche die Aufnahme und Verarbeitung der einzelnen Teile von vornherein erleichtern. Auch Meutsch kommt zum Schluß, daß die von ihm referierten Ergebnisse zur Konsequenz haben,

die Gestaltung von Texten bewußt, relativ zu ihren kommunikativen Absichten, relativ
zu ihren dominanten Lesergruppen und relativ zu den Situationen, in denen diese
Texte benutzt werden, hinsichtlich ihrer Verständlichkeit zu verbessern (Meutsch
1989, 20).

Eine Methode zur Verbesserung von Hypertexten soll später in dieser Arbeit vorgestellt werden, besonders was die Erstellung von Kohärenzstrukturen in Hypertexten angeht. Dazu soll uns das Beispiel der Metapher zur Orientierung im textuellen Raum dienen.

Neben der Verständlichkeitsforschung liefert die Textlinguistik interessante Anknüpfungspunkte zum Thema der Organisation von Hypertext. So läßt sich van Dijk's Theorie von den Superstrukturen hervorragend auf Hypertext anwenden. Wir finden bei van Dijk (1980, 128ff.; s. a. van Dijk/Kintsch 1983) die Super- Makro- und Mikrostrukturen, nach denen Texte organisiert sind. Für die

Hypertexttheorie ist jedoch vor allem van Dijks Ansatz zur Beschreibung von Text als System bzw. als Prozeß interessant. Denn während das System Hypertext in seiner gespeicherten Datenstruktur den genannten Kriterien von Superstruktur bis Mikrostruktur entspricht, haben wir es im Gebrauch von Hypertext mit Text als Prozeß zu tun, einem dynamischen Medium (Duhme 1989).

Ich fasse zusammen: die Verständlichkeit eines Textes, also auch eines Hypertextes, hängt entscheidend ab von der Struktur der Einheiten oder Informationschunks. Um dem Leser größtmögliche Transparenz zu bieten, ist eine kohärenzstiftende Organisation der Informationseinheiten vonnöten. Um dieses Ziel darzustellen, gehe ich nun zunächst auf die Vorstellung von Hypertext als Raum ein, um hernach mit einem praktischen Beispiel durch Metaphern erzeugte kohärenzstiftende Hypertextstrukturen darzustellen.

2 Die Räumlichkeit des Mediums Hypertext

Die Wahrnehmung von Hypertext geschieht traditionell als Raum. Wir wollen nun zunächst einmal klären, wie diese Räumlichkeit von Hypertext beschaffen ist. Es gibt durchaus eine Kontroverse, inwieweit der hypertextuelle Raum sprachlich beschrieben werden kann. Interessant wird diese Auseinandersetzung insbesondere, da die Definition von Hypertext an sich bereits die räumliche Konzeption vorgibt und somit für die Beschreibung prägend wirkt. So sagt Theodor H. Nelson, der Vater des Hypertextgedankens: "By 'hypertext' I mean *nonsequential* writing – text that branches and allows choices to the reader" (Nelson 1987, 0/2). Schnupp verweist auf die Informationsstruktur von Hypertext, wenn er sagt, daß Hypertext „die Verknüpfung von Textdokumenten durch hierarchische Relationen und/oder Verweisstrukturen“ (Schnupp 1992, 15) sei. Diese hierarchische Struktur der Hypertextdokumente ist im Gegensatz zu gedruckten Texten nicht linearisiert, sondern in einem Netzwerk organisiert. Wie ist die Dimensionalität dieses Netzwerkes beschaffen? Wenn wir an ein „Netz“ denken, erscheint vor dem inneren Auge ein zweidimensionales Konstrukt wie ein Fischernetz. Ein Informationsnetz ist nun aber dem Beispiel eines Fischernetzes ganz unähnlich, da es ja Knoten und Verknüpfungen nicht lediglich in zwei Dimensionen aufweist, sondern eher dem Beispiel eines neuronalen Netzes ähnelt, in dem viele Zellen auf unterschiedlichste Art und Weise in drei Dimensionen miteinander verwoben sind.

Eine andere beliebte Metapher auf den hypertextuellen Raum ist die des Weltalls oder des Universums. In der Tat bietet sich der Vergleich an, denn ähnlich den Galaxien oder Sternsystemen im tatsächlichen Weltraum sind die Informationen im *Hyperspace* hierarchisch gruppiert, lassen sich aber beliebig verknüpfen, etwa

in der Art des freien Fluges eines Raumschiffes von einem Stern zum anderen. Das in diesem Zusammenhang gebrauchte Wort von den „virtuellen Textuniversen“ stammt von Wallmannsberger (1994). Ihm zufolge handelt es sich allerdings nicht nur um ein textuelles oder informationelles Raumgefüge, sondern regelrecht um einen Arbeitsraum, in den der Computerphilologe eintaucht. Wir haben es also bei der Wahrnehmung von Hypertext durchaus mit einem virtuellen Phänomen zu tun. Tatsächlich ist Hypertext als Mittel zur Textproduktion ein Nebenprodukt der Forschung im Bereich *virtual reality* (VR), wie Howard Rheingold (1995, 96) erwähnt. In diesen Zusammenhang gehört auch die bekannte Navigationsmetapher (Rheingold 1995, 113). Sie und das gefürchtete Syndrom des *getting lost* im Cyberspace, d. h., des Sich-Verlierens im virtuellen Textuniversum aufgrund zu komplexer Strukturen, machen uns einmal mehr deutlich, daß der textuelle Hyperraum durchaus erfahrbar erscheint.

Allerdings wird diese Konzeption nicht uneingeschränkt vertreten. Im Gegensatz zum allgemeinen Sprachgebrauch und den bisher zitierten Quellen vermutet Miller (1995), daß die herkömmlichen räumlichen Metaphern, die für den elektronischen Raum Verwendung finden, tatsächlich eine Täuschung verursachen. Diese besteht in der Vorgaukelung eines faßbaren Raumes; alles im virtuellen Universum sei scheinbar an einem quasivisuell faßbaren Platz. Die visuellen Metaphern des virtuellen Raumes sollen einen nicht vorhandenen Raum erfahrbar machen. Aber in seiner ganzen Komplexität sei das Internet, auf das sich Miller hier bezieht, nicht räumlich definierbar. Es besteht aus vernetzten Computern, aus Leitungen, anderen Medien und ständig übertragenen Daten. Daher bezeichnet Miller die Metaphern des Internets als Katachrese (1995, 31). Im Gegensatz zu Autoren wie Wallmannsberger, Rheingold oder Stoll sagt Miller, das Internet sei keineswegs ein Raum, sondern etwas nicht durch räumliches Wissen Beschreibbares oder Wahrnehmbares, wenn Beschreibung hier Beschreibung mit den Sinnen bedeutet. Ein realer Raum habe bestimmte Grenzen, die einzelnen Regionen sind voneinander getrennt. Das Internet sei dagegen ein „nichträumlicher“ Raum. Überspitzt formuliert: im virtuellen Raum ist alles immer überall, alles ist allem immer gegenübergestellt. Die Verwirrung nimmt konkrete Formen an, wenn es darum geht, sich bestimmte Wege und Orte im Internet zu merken. Es gibt zu viele Wege von hier nach dort, zu wenige Anhaltspunkte, um Strukturen im Gedächtnis zu behalten.

Miller verneint auch die Metapher der Bewegung, die wir im Begriff der Navigation im Internet bereits kennengelernt haben. Tatsächlich bewegt man sich natürlich nicht wirklich, wenn man durch das Internet streift. Die einzige Bewegung spielt sich auf dem Bildschirm ab, verursacht nur von einer geringen körperlichen Anstrengung beim Tippen oder der Bewegung der Maus. Dennoch können wir

feststellen, daß die Identifizierung der Benutzer mit Positionen im virtuellen Raum nicht nur durchaus gegeben ist, sondern in der Wahrnehmung des Computers überhaupt eine wesentliche Rolle spielt (vgl. Ipsen 1997). So existieren räumliche (und auch temporale) Metaphern nicht nur für die Bewegung in Hypertexten, sondern für alle möglichen Applikationen von der Eingabe von Befehlen an der Kommandozeile bis hin zu vierdimensionalen Programmstrukturen, die Raum- und Zeitkomponenten beinhalten. Gewiß stellt sich außerdem die Frage, inwieweit der Begriff der Katachrese nicht auch für Orientierungsmetaphern in nichtelektronischen Texten zutrifft, denen ja auch eine Räumlichkeit jenseits der realen Ausdehnung des Buches zugemessen wird (vgl. Nöth 1994). Insgesamt läßt sich also zwar rhetorisch die Nicht-Räumlichkeit des Internets oder der Hypertexte im allgemeinen formulieren, allerdings bleibt eine solche Diskussion angesichts der Tatsache der Bindung menschlicher Kognition an Raumvorstellungen fragwürdig.

3 Die Verwendung räumlicher und temporaler Metaphern

Obwohl der Gedanke an das Internet und also auch an den Hypertext mit der Vorstellung von Räumen verquickt ist, sind die Auffassungen zur Verwendung räumlicher Metaphern negativ geprägt (Stanton 1994). Zwar spricht man davon, *in das Internet zu gehen, sich im Hyperspace zu verlieren, einen site zu besuchen*; diese und viele andere Beispiele von Raum- und Bewegungsmetaphern belegen die räumliche Vorstellung. Das Paradigma von der Nichtverwendbarkeit räumlicher Metaphern entstand demgegenüber durch einen direkten Vergleich mit nichtelektronischen Texten. Obwohl selbst in gedruckten Werken kein wirkliches Oben oder Unten, Vorn oder Hinten im Sinne eines dreidimensionalen Raumes existiert, sind diese Metaphern dort so akzeptiert, daß ihre Verwendung keine Probleme bereitet. Den Widerspruch zwischen mentalem Konzept von Raum als Text und seiner physischen Erscheinung nennt Nöth die semiotische Paradoxie vom Text als Raum (1994, 163). Diese Paradoxie bietet sich nicht nur im Vergleich mit gedruckten Texten. In der Tat ist menschliche Rede ein lediglich lineares Konstrukt. Dennoch ist die Bezugnahme auf die Rede oder Texte aller Art von räumlichen Metaphern geprägt. Nöth gliedert die Dimensionalität des textuellen Raumes in Punkte, Linien, Flächen und den dreidimensionalen Raum (ebd., 164-168). Zusammenfassend läßt sich sagen, daß eine dezidierte Vorstellung von oben, unten, vorn, hinten, aber auch von bestimmten Orten oder Bewegungen zwischen Orten im Text existiert. Die Mehrdimensionalität des Textes als Konzept ist auch auf Hypertexte in virtuellen Textuniversen anzuwenden (Ipsen 1997, 570). Es bleibt die Einschränkung, daß die Wahrnehmung dieses Universums ins-

gesamt eine andere ist, wie wir bei Miller gesehen haben. Die physische Greifbarkeit des Mediums ist nicht gegeben und daher die Transposition räumlicher Begrifflichkeiten wie im Falle Raum – Buch nicht ohne weiteres möglich. Vielleicht ist tatsächlich in einer Überspitzung der Universumsmetapher der Vergleich zu einem Astronauten angebracht, der frei im Raum schwebt und also kein Oben oder Unten erfährt, es sei denn, er definiert diese Richtung nach seinem subjektiven Standpunkt (wie wir es tatsächlich auch auf der Erde tun, denn letztlich sind unsere Antipoden „unter“ uns auf der anderen Seite der Erdkugel). Man kann sich aber des Gebrauchs räumlicher Metaphern zur Orientierung im Raum nicht entziehen. Lakoff und Johnson (1980, 14) verweisen auf die physische und kulturelle Erfahrung, die dieses Phänomen begründet. Sie schließen, daß die meisten unserer fundamentalen Konzepte durch räumliche Metaphern organisiert seien (1980, 17). Kann man sich also im Hypertext den Metaphern entziehen?

Immer noch bleibt das Problem der Unklarheit der Richtungen im virtuellen Raum des Hypertextes. Bleiben wir bei dem Beispiel des Weltraumreisenden: zehn freischwebende Astronauten im Weltraum können in zehn verschiedene Richtungen zeigen, wenn sie ein „Unten“ beschreiben sollten. Ähnlich verhält es sich in Hypertextuniversen. Räumliche und auch zeitliche Relationen sind der individuellen Erfahrung des einzelnen Benutzers unterworfen. Eine festdefinierte Formel wie „siehe weiter vorn“ stellt also zunächst einen Verweis ins Ungewisse dar, da ein Benutzer die Reihung von Informationsstücken in der Hypertextsituation (der Zeit zwischen Ein- und Ausstieg in das Informationsnetz) anders erfahren haben kann als ein anderer. Detaillierte Tests zu dem Problem der Navigation unter Verwendung von Metaphern haben im Zusammenhang mit elektronischen Lernumgebungen, englisch *Computer Assisted Learning* (CAL), stattgefunden. Eine ausführliche Studie findet sich bei Stanton (1994). Der Autor betont, daß die räumliche Metapher zwar nützlich für die *Designer* von nonlinearen Textwelten sei, aber für *Benutzer* hinderlich werden könne, wenn die Sicht auf das Material eine andere ist und somit die Raumkonzepte von Designer und Benutzer sich unterscheiden. Besonders erschwerend erscheint in diesem Zusammenhang die Tatsache, daß in einem Hypertextdokument mehrere Verknüpfungen, englisch *links*, von einem Knoten, englisch *node*, zu anderen und zum selben Knoten hin führen können. Das Verfolgen solch verzwickter Wege wird mühsam, so daß ein *mapping* seitens des Benutzers erschwert wird (Stanton 1994, 289). Die Folge ist zwangsläufig eine Desorientierung des Nutzers. Allerdings scheint das Problem darin zu bestehen, ein sinnvolles semantisches Netzwerk für alle Benutzer zu schaffen, die möglicherweise die Hypertextdatenbank besuchen könnten. Stanton (1994, 290) bietet die Lösung an, erstens die Notwendigkeit zur Navigation für den Benutzer möglichst zu reduzieren, zweitens eine Nutzerführung einzubauen,

die es erlaubt, das Fortkommen jederzeit nachzuprüfen und drittens Knoten einzuführen, die als Markierungen oder Orientierungspunkte dienen können. Als wichtigstes Ziel sieht der Autor allerdings eine neue Art des Designs von Informationsnodes, um deren Zugehörigkeit zu bestimmten Pfaden oder größeren Informationsverbänden darzustellen. Die Verwendung räumlicher oder temporärer Metaphern zur Beschreibung solcher *nodes* schließt Stanton allerdings aus den bekannten Gründen aus.

Wir stehen also vor einem Dilemma. Einerseits stellen wir fest, daß unser Konzept von Text *per se* ein räumliches ist, welches uns kulturgeschichtlich geprägt hat. Der elektronisch vernetzte Text erscheint dabei geradezu als Paradebeispiel für den verräumlichten Text. Andererseits besteht gerade in der Wahrnehmung des hypertextuellen Raumes das Problem der Individualisierung seitens der Benutzer, d. h., die räumliche Organisation wird immer wieder neu erfahren, ja bei der Benutzung konstruiert. Somit versagen die überkommenen textuellen oder sprachlichen Beschreibungsmethoden offenbar, denn jeder im Text fest gefügte metaphorische Verweis muß schließlich fehlgehen und seine Funktion verlieren. Dennoch erscheinen sprachliche Mittel zur Beschreibung einer textuellen Struktur nach wie vor geeignet.

Wie ich zeigen will, ist das beschriebene Dilemma ein scheinbares. Es entsteht aus dem Eindruck, ein Hypertext bestehe ähnlich einem gedruckten Buch aus einem fixierten Text, der zwar in einer anderen Informationsstruktur organisiert ist, ansonsten aber den gleichen Prinzipien unterliegt wie ein Text in jedem anderen Medium. Dieser Eindruck ist, wie wir sehen werden, falsch. Tatsächlich besteht das Potential eines Hypertextes darin, daß er in seiner elektronischen Natur *stets veränderbar* ist, da wir ihn mit Programmen und geeigneten Softwarewerkzeugen beeinflussen können. Dazu besteht die Notwendigkeit, den Gedanken vom Netzwerk beim Hypertext aufzugeben und sich statt dessen auf die einzelnen Informationseinheiten zu konzentrieren, sowie das Design von Verweisen auf einzelnen Informationsseiten zu thematisieren.

Diese noch sehr abstrakte Forderung mag den Anwender von Hypertexten befremden, da man es ja z. B. beim Surfen durch das Internet lediglich mit auf Bildschirmoberflächen abgebildetem Hypertext zu tun hat und somit die erwähnte Flexibilität, die im Hypertextskript verankert ist, verborgen bleibt. Für unsere Analyse ist der auf dem Bildschirm dargestellte Text eben nur eine *Repräsentation*, ein bereits *vermittelter, medialisierter* Text. Der Begriff der Repräsentation soll hier auf die Problematik der Darstellung von Daten, d. h., dem Inhalt eines Hypertextskriptes, für einen Anwender mit Hilfe eines Interfaces, d. h., in unse-

rem Falle des Bildschirms, Verwendung finden (vgl. aber zum generellen Problem der Repräsentation und Computer im semiotischen Sinne Nöth 1997).

Zur Frage der Repräsentation von Daten kann uns ein Einblick in die kognitive Analyse der Repräsentation im Computer weiterhelfen (vgl. Woods/Roth 1988; Ipsen 1997, 562-564), wobei wir die eben erwähnten Begriffe weiter klären können. Die „Welt“ aus Daten im Computer wird dem Benutzer durch ein Interface, eine Schnittstelle, eröffnet oder repräsentiert. Natürlich sind die „Welten“ im Computer keine selbstkonstituierenden Welten. Die Datenwelten sind ihrerseits bereits Repräsentationen für Probleme, Konzepte oder Vorgänge in der realen Welt. Für uns interessant also ist die Datenwelt als Text, besser noch als Feld des Textschaffenden. Wir wenden uns an dieser Stelle daher endgültig von der Betrachtung der Hypertexte als graphischer Darstellung auf dem Bildschirm ab und konzentrieren uns ganz auf die Skripte, die allein uns Einblick in die Praxis der Hypertextproduktion erlauben.

4 Skript und dynamischer Hypertext

Unser Ziel, Varianten der Hypertextproduktion unter Verwendung metaphorischer Mittel aufzuzeigen, bedarf einer kurzen Übersicht über die Grundprinzipien des Aufbaus von Hypertextdokumenten. Im folgenden Abschnitt werden diese Mechanismen geschildert, auf die dann in einem praktischen Beispiel Bezug genommen wird.

4.1 HTML und JavaScript

Ein unter Verwendung der *Hypertext Markup Language* (HTML) geschriebenes Skript besteht aus dem darzustellenden Text und einer Vielzahl von Informationen, die zur Darstellung der HTML-Seite durch den Browser dienen. Diese Informationen sind in sogenannten *tags* gefaßt, die, in spitzen Klammern geschrieben, alle Anweisungen zur Formatierung und Verlinkung einer Hypertextseite enthalten. Außerdem regeln sie die Einbettung von Objekten wie Grafiken, Ton oder bewegten Bildern in der Hypertextseite. Zum Beispiel weist der *tag* `` den Browser an, den nachfolgenden Text fett zu formatieren. Das Ende der Formatierungsanweisung oder anderer Funktionen wird mit einem sogenannten *endtag* beschrieben, der wie der *starttag* aussieht, aber zusätzlich einen Schrägstrich oder *slash* aufweist: ``. Alle *tags* werden vom Autoren des Hypertextes eingegeben und abgespeichert. Ein solcher Hypertext besteht dann aus reiner Textinformation, die seitens des Benutzers nicht verändert werden kann und folglich bei jedem Ladevorgang durch den Browser gleich aussieht. Es ist leicht ersichtlich, daß ein metaphorischer Verweis von einer Seite auf die andere unter diesen

Umständen tatsächlich wenig Sinn macht. Ein Hinweis wie: „siehe weiter vorn“ oder „das können Sie später an Stelle XY lesen“ entbehrt jeder Grundlage, denn weder gibt es eine eindeutige zeitliche noch eine räumliche Ordnung im Hypertextgewebe.

Allerdings stellt eine solche eindeutig fixierte Hypertextseite nur die Basis dessen dar, was sich mittlerweile mit Hypertexten machen läßt. Viel mehr Funktionen lassen sich durch die Einbindung sogenannter Skriptsprachen erzielen, mit denen kleine Programme in das HTML-Skript eingebettet werden können. Eine dieser Skriptsprachen ist JavaScript. Um JavaScript hinreichend zu erklären, wollen wir einen kurzen Blick auf die Programmiersprache Java werfen. Java ist eine neuere Programmiersprache für das Web (Lemay/Moncur 1997, 23). Man schreibt mit Java kleinere Programme, sogenannte *Applets*, die auf einer HTML-Seite geladen und durch den Java-Interpreter ausgeführt werden, welcher in dem Browser enthalten ist. Hier liegt ein großer Vorteil dieses Konzeptes entgegen anderen Programmiersprachen, deren Produkte erst kompiliert werden müssen, d. h. durch den Programmierer mit Hilfe der Entwicklungsumgebung in den betriebsfertigen Zustand versetzt werden müssen. Die Kompilierung eines komplexen Programmes kann lange Zeit in Anspruch nehmen. Ein Java-Applet bleibt bis zu seinem Aufruf gewissermaßen „halbfertig“ und wird erst durch den Interpreter in einen durch den Computer ausführbaren Code verwandelt. Dennoch ist Java immer noch sehr komplex (es ist eng mit der Sprache C++ verwandt) und es bedarf einiger Übung, gute – und vor allem elegante – Ergebnisse damit zu erzielen. JavaScript dagegen richtet sich auch an den ungeübten Benutzer. Es hat eine einfache Syntax, und der Quellcode einer JavaScript-Applikation kann ohne weiteres in eine HTML-Seite eingebunden werden (Lemay/Moncur 1997, 24).

4.2 Zielsetzung: Das Skript „CustomizeText“

HTML und JavaScript im Verein geben uns die Werkzeuge in die Hand, um die Problematik zu umschiffen, welche sich uns eingangs darstellte. Das Problem ist aus Kapitel 1 ersichtlich: Ein Hypertext soll sprachlich kartographiert werden, um dem Benutzer mit Hilfe temporaler oder räumlicher Metaphern die Orientierung zu erleichtern. Gleichzeitig muß diese metaphorische Lokalisierung bestimmter Textstücke individuell auf das Verhalten verschiedener Benutzer reagieren, um irreführende Verweise zu vermeiden.

Ich möchte hier ein Beispiel für ein JavaScript vorstellen, welches den Text einer HTML-Seite verändern kann. Mit seiner Unterstützung lassen sich *dynamische Verweise* herstellen. Räumliche oder auch zeitliche Metaphern werden dann flexibel und verweisen nutzerindividuell auf bereits besuchte oder noch unbekannte

Seiten in einer Form, die der räumlichen Varianz gerecht wird und dennoch die textuelle Integrität des einzelnen Chunks wahrt. Nennen wir das betreffende Skript transparenterweise „CustomizeText“. Was soll es können? Wir benötigen ein Mittel, um:

1. festzustellen, wo ein Benutzer sich im Hypertextcluster befindet;
2. die Adresse dieses Ortes zu speichern, um auf die Lokalität des chunks wieder zugreifen zu können, also stets zu wissen, wo der Nutzer bereits gewesen ist und wo nicht;
3. gemäß einem Abgleich der so erstellten Liste mit dem aktuellen Standort des Benutzers die räumlichen oder zeitlichen Verweise, die sich auf dieser Seite befinden, zu gestalten, d. h. den Gehalt der Bildschirmrepräsentation des Hypertextdokumentes dynamisch der Benutzersituation anzupassen.

Diese Anpassung bedeutet nichts weiter, als Verweise oder Text aus einem im Skript vereinbarten Pool von Möglichkeiten gemäß bestimmter Bedingungen auszuwählen. War also Benutzer X auf Seite 1, soll auf dem Bildschirm Textvariante Alpha erscheinen. War er noch nicht dort, soll Textvariante Beta abgebildet werden.

4.3 Ein Beispielcluster zum Skript CustomizeText

Ich will dieses noch abstrakte Problem an einem einfachen praktischen Beispiel demonstrieren. Unser HTML-Cluster besteht aus lediglich vier Seiten:

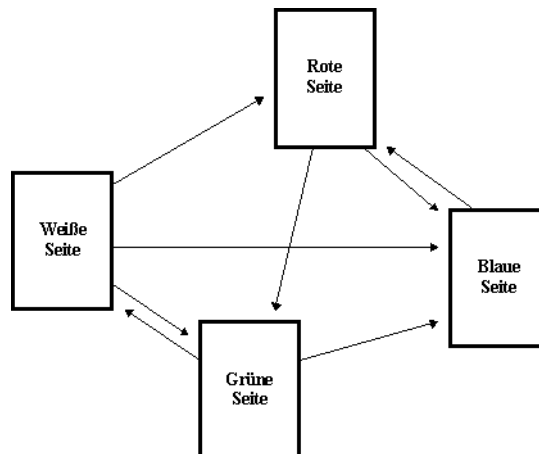


Abb. 1: Der Beispielcluster zum Skript „CustomizeText“

Ich benenne die Seiten der Einfachheit halber mit Farben. Unter den einzelnen Seiten existieren willkürliche Verknüpfungen, die aber auch beliebig anders aussehen könnten.

Unsere Aufgabenstellung erfordert es nun, Angaben über das Userverhalten zu speichern, d. h., den genauen Pfad, den er oder sie durch unser Netzwerk nimmt, aufzuzeichnen. Wir können dies erreichen, indem wir eine Variable vereinbaren, die allen HTML-Seiten in unserem Cluster zur Verfügung steht. Eine Variable ist eine Größe in einem Programm, die beliebig veränderbar ist. In unsere Variable sollen die Namen und Adressen aller Seiten, die ein Nutzer besucht hat, gespeichert werden. Leider kann JavaScript keine Werte von Variablen von einer Seite an die andere übergeben: die Variable würde bei jedem Ladevorgang einer anderen Seite wieder auf den ursprünglichen Wert zurückgesetzt. Daher müssen alle Seiten im Cluster einer Elternseite untergeordnet sein. Das läßt sich in HTML sehr einfach durch Implementierung eines Framesets erreichen (eine detaillierte Anleitung zum Erstellen von Framesets und für alle weiterführenden Detailfragen siehe z. B. die ausgezeichnete Online-Einführung von Münz 1998). Alle vier Seiten sind also einem Elternframe untergeordnet. Den HTML-Quellcode zeigt Abbildung 2.

```
<HTML>
<HEAD>
<TITLE>Frameset Farbseiten</TITLE>
</HEAD>
  <frameset rows="50,*" border=0 frameborder=2>
    <frame src="top.html" name="menu" scrolling="no" border=0 noresize
marginwidth=0 marginheight=0 bordercolor=#000000>
    <frame src="bottom1.html" name="main" border=0 noresize marginwidth=0 margin-
height=0
bordercolor=#000000>
  </frameset>
<script language="JavaScript">
var visitPage=""
</script>
</HTML>
```

Abb. 2: Das Frameset Farbseiten

In diesem Frameset wird nicht nur die Bildschirmaufteilung für die Tochterframes vereinbart, sondern auch – unbemerkt für den Benutzer – die Variable „visitPage“ durch ein kurzes Skript initialisiert, wie dem in der Abbildung fett gedruckten Textbereich zu entnehmen ist. Die Anführungsstriche zeigen, daß die Variable

noch leer ist. Nun befindet sich in jedem der Quellcodes der HTML-Seiten ein weiteres Skript, welches dieser Variable die Adresse der gerade aktuellen Seite hinzufügt, wie in Abbildung 3 ersichtlich.

```
<HTML>
<HEAD>
<TITLE>Java-Skript-Testseite</TITLE>
</HEAD>
<body bgcolor="#8080FF">
<h2>Dies ist die Blaue Seite.<br>
Der Text auf dieser Seite ist interaktiv dynamisch, je nachdem, ob Sie andere Seiten bereits besucht
haben.<p>
Wie Sie auf der <a href="bottom3.html">Roten Seite</a>
<script language="JavaScript">
parent.visitPage=parent.visitPage+document.location.href+"\n"
var wasThere=parent.visitPage.indexOf("bottom3")
if(wasThere != -1)
    {
        document.write("<u>bereits gesehen</u>")
    }
else
    {
        document.write("<u>noch sehen werden</u>")
    }
</script>
, verändert sich der Text auf den Farbseiten je nach Ihrer Navigation.</h2>
</BODY>
</HTML>
```

Abb. 3: Quellcode der Blauen Seite

Wie man der zweiten Zeile des fettgedruckten Skriptes in Abbildung 3 entnehmen kann, wird jeweils der Wert der Variable um die aktuelle Adresse erweitert. So ergibt sich im Laufe des Besuches des Clusters eine exakte Reihung der besuchten Seiten. Im weiteren Teil des Skriptes kann man sehen, wie das Skript die Variable nach bestimmten Seitennamen durchsucht. Je nachdem, ob die in Frage kommende Seite besucht worden ist oder nicht, wird ein anderer Text auf dem Bildschirm ausgegeben. Auch für den Laien ist die Variabilität im Skript, die durch die Bedingung *if-else* gesteuert wird, leicht erkennbar. Im Detail wird durch die Bedingung geregelt, was geschieht, wenn der Name der eventuell besuchten Seite in der Variable auftaucht: *wenn* (if) das Ergebnis der Prüfung positiv aus-

fällt, soll das Programm die erste Textvariante ausgeben, *sonst* (else) die zweite Variante. Wie diese Seite auf dem Bildschirm aussehen kann, zeigen die Abbildungen 4 und 5:

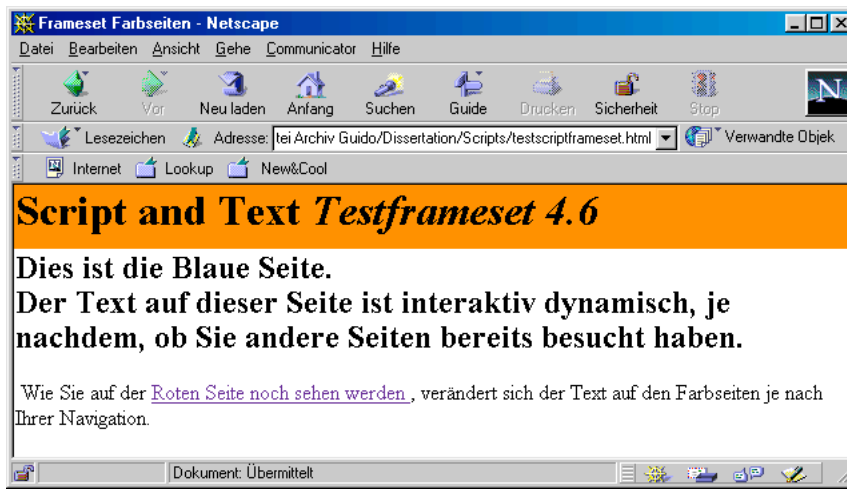


Abb. 4: Die Blaue Seite vor Besuch der Roten Seite



Abb. 5: Die Blaue Seite nach Besuch der Roten Seite

Wenn man auf den unterstrichenen Bereich achtet, so wird man feststellen, daß sich der Text vor und nach dem Besuch der Roten Seite verändert hat. Diese Veränderung wird dadurch bewirkt, daß das Skript auf der Blauen Seite in der Variablen „visitPage“ die Adresse der Roten Seite feststellt.

In unserem Beispiel werden temporale deiktische Ausdrücke verwendet, um korrekte Bezüge zwischen den einzelnen Seiten herzustellen. Allerdings kann man sich auch vorstellen, daß Ausdrücke wie *eingangs*, *zu Beginn*, *vorn*, *hinten* etc. in Betracht kommen können, um eine transparente Verweisstruktur zu erzeugen. Sicherlich ist zu diskutieren, ob der hier verwendete Ausdruck „sehen werden“ in jeder Situation verwendbar ist, da ja nicht garantiert werden kann, ob der Benutzer tatsächlich dem angegebenen Pfad folgt. Das Beispiel versteht sich hier lediglich als weiterführende Anregung, verbale Nutzerführung durch *Versprechen*, *Aufforderungen* oder *Befehle* zu implementieren, eine Möglichkeit, deren Nutzen durch eine pragmatisch orientierte Analyse überprüft werden muß. In seiner Verwendung ist das Skript unbegrenzt zu nutzen. Der veränderte Text kann von beliebiger Länge sein, in unterschiedlichen Sprachen verfaßt sein oder ein unterschiedliches Format aufweisen. Die Variation kann aber auch Graphiken umfassen oder ein beträchtliches Stück Quellcode, so daß die Navigation des Benutzers auch das Layout der Seite beeinflussen kann. Nützlich wäre dies insbesondere für eine farbgesteuerte Benutzerführung.

Die Nachteile dieses Verfahrens sind, daß die Variable bei jedem neuen Laden des gesamten Framesets für den Benutzer wieder zurückgesetzt ist, d. h., alle Werte aus ihr gelöscht werden. Dieses kleine Skript läßt leider keine Speicherung der Variablenwerte auf der Festplatte zu. Auch benötigt man immer ein Frameset, eine Methode, die bei einigen Webdesignern durchaus unbeliebt ist. Diese Klippen lassen sich jedoch umschiffen, indem das JavaScript Skript durch ein echtes Java-Applet ersetzt wird. In diesem Falle ist aber eine Client-Server-Kommunikation aufzubauen; eine Möglichkeit, über die viele Hypertextautoren nicht verfügen, wenn sie nicht gleichzeitig Netzwerkadministratoren sind. JavaScript bietet also durchaus Alternativen, wenn auch seine Kapazitäten teilweise begrenzt sind.

5 Zusammenfassung und Ausblick

Wir haben gesehen, daß die Aufgabe der Organisation eines Hypertextdokumentes keineswegs bei der Hierarchisierung und Vernetzung der Informationschunks enden darf. Die Aufgabe des Autors fängt dort erst an, denn eine sinnvolle Textgestaltung findet vor allem innerhalb der einzelnen *chunks* statt. Die Verwendung von Skripten dazu erscheint essentiell, denn das flexible Medium Hypertext sollte nicht durch erstarrte Textstrukturen innerhalb der einzelnen Informationscluster

behindert werden. Zusätzlich gewinnen wir ein neues Bild der räumlichen Metapher im Text: die Metapher im Hypertext kann als *dynamische Metapher* definiert werden. Diese Erkenntnis von grundlegend neuen Möglichkeiten der Textualität gibt Anlaß, auch über weitere textuelle Phänomene neu nachzudenken. So erscheinen die Bemerkungen Kuhlens (1991, 27–39) zur Hypertextkohärenz in einem neuen Licht. Wenn er beschreibt, daß herkömmliche kohärenzerzeugende Mittel in Hypertexten nicht greifen, so beruht diese Ansicht auf oberflächlicher Betrachtung des Phänomens Hypertext als einem nichtdynamischen, fixen Medium. Tatsächlich offeriert die Makrostruktur eines Hypertextes mit seinen nur logisch verbundenen Chunks geringe Möglichkeiten, die einzelnen Teile kohärent zu verbinden. Wir sehen aber, daß bei der Einbindung von Skripten auf der Ebene der Mikrostruktur einzelner *chunks* interessante Möglichkeiten zur Realisation einer Hypertextkohärenz, zur Verwendung räumlicher Metaphern und sogar zur Erzeugung einer virtuellen ich-Origo des Benutzers existieren. Weitere Beispiele zum dynamischen Hypertext wären erforderlich, um andere Spielräume veränderlicher Verweise zu studieren. So erscheint es – mit entsprechend komplexeren Skripten – realistisch, kausale Verweisketten auf der Grundlage des Benutzerpfades herzustellen. Es könnte jeder Nutzer je nach Vorliebe oder Schwerpunkt durch einen intelligenten *guide* durch die Hypertextdatenbanken geführt werden. Die Möglichkeiten sind flexibel, und ihre Grenzen sicherlich schwer abzuschätzen. Erforderlich ist aber auch ein neues Bild des Autors, nicht etwa geprägt von seinem Verschwinden, wie reichlich nachzulesen, sondern von seiner Rolle als Informationsmanager und Programmierer.

Literatur

- Duhme, Michael (1989): Der Text als System und Prozess: Textlinguistische Strukturmodelle in der Anwendung. Essen: Die Blaue Eule [Sprache und Theorie in der Blauen Eule; 4]
- Ipsen, Guido (1997): Linguistic orientation in computational space. In: Nöth, Winfried (Hrsg.): Semiotics of the media. Berlin: Mouton de Gruyter, 559–573
- Kuhlen, Rainer (1991): Hypertext: Ein nicht-lineares Medium zwischen Buch und Wissensbank. Berlin: Springer
- Lakoff, George/ Johnson, Mark (1980): Metaphors we live by. Chicago: University of Chicago Press
- Lemay, Laura/ Moncur, Michael (1997): JavaScript: Interaktive Webseiten mit JavaScript. Haar b. München: Markt&Technik
- Meusch, Dietrich (1989): Text- und Bildoptimierung. In: Antos, Gerd/ Augst, Gerhard (Hrsg.): Textoptimierung. Frankfurt/M. u. a.: Lang, 8–37
- Miller, J. Hillis (1995): The ethics of hypertext. In: Diacritics: A review of contemporary criticism 25 (3), 27-40
- Münz, Stefan (1998): HTML-Dateien selbst erstellen: Das Kompendium für Entwickler von WWW-Seiten. Online in Internet. URL: <http://www.uni-essen.de/www/selfhtml/selfhtml.htm> (Stand Version 7.0 vom 27.04.1998)
- Nelson, Theodore H. (1987): Literary machines. Version 87.1. Selbstverlag

-
- Nöth, Winfried (1994): Der Text als Raum. In: Halwachs, Dieter W./ Penzinger, Christine/Stütz, Irmgard (Hrsg.): Sprache, Onomatopöie, Rhetorik, Namen, Idiomatik, Grammatik: Festschrift für Karl Sornig. Graz: Institut für Sprachwissenschaft der Universität Graz [Grazer Linguistische Monographien; 11], 163–173
- Nöth, Winfried (1997): Representation in semiotics and computer science. In: *Semiotica* 115 (3-4), 203-213
- Rheingold, Howard (1995): Virtuelle Welten: Reisen im Cyberspace. Reinbek: Rowohlt
- Schnupp, Peter (1992): Hypertext [Handbuch der Informatik; 10.1] München: Oldenbourg
- Stanton, N. A. (1994): Explorations into hypertext: Spatial metaphor considered harmful. In: *Educational and training technology international* 31 (4), 276-294
- Van Dijk, Teun A. (1980): Textwissenschaft: Eine interdisziplinäre Einführung. Tübingen: Niemeyer
- Van Dijk, Teun A./ Kintsch, Walter (1978): Toward a model of text comprehension and production. In: *Psychological Review* 85, 363–394
- Van Dijk, Teun A./ Kintsch, Walter (1983): Strategies of discourse comprehension. Amsterdam: Orlando
- Wallmannsberger, Josef (1994): Virtuelle Textwelten. Heidelberg: Winter
- Woods, David D./ Roth, Emily M. (1988): Cognitive systems engineering. In: Martin Helander (Hrsg.): Handbook of human-computer interaction. Amsterdam: North Holland, 3–43

