

Text und Textgestaltung

C. M. Sperberg-McQueen

Festvortrag, 25. Jahrestag der ITUG, Potsdam, 4. Oktober 2018.

I.

Es ist eine Ehre, eingeladen zu werden, vor Ihnen an diesem Abend anlässlich des 25. Jahrestages der Internationalen TUSTEP User-Gruppe vorzutragen, in dem Jahr, in dem Prof. Wilhelm Ott seinen 80. Geburtstag feiert. Ich gratuliere.¹

Ich sollte gleich am Anfang bekennen, dass ich kein TUSTEP-Experte bin. Meine TUSTEP-Kenntnisse zu verbessern steht seit Jahren auf der privaten Aufgabenliste, und ich bedaure, dass bis jetzt andere Obliegenheiten es unmöglich gemacht haben, dazu zu kommen. Von TUSTEP habe ich schon früh in meiner Arbeit in den Digital Humanities gelernt, und das, was TUSTEP-Köner mit diesem Werkzeug erreichen können, hat mich immer tief beeindruckt. Auch die partielle Kenntnis von TUSTEP hat meine Arbeit an Infrastrukturen für die Digital Humanities und andere wissenschaftliche Gebiete beeinflusst.

Unter anderem zeigt TUSTEP (wie mir scheint), was man erreichen kann, wenn man bei der Herstellung von Software zuallererst versucht, den Anforderungen des wissenschaftlichen Umgangs mit Text gerecht zu werden. Software für wissenschaftliche Textverarbeitung steht in einer interessanten und intimen Beziehung mit unserem Verständnis von Texten als linguistische, historische und kulturelle Objekte: Um Textarbeit effektiv zu unterstützen, muss Software ein wissenschaftliches Verständnis von Text zugrunde liegen. Umgekehrt kann das Nachdenken darüber, wie unsere Software funktioniert und wie es textuelle Strukturen modelliert, uns zu einem tieferen Verständnis von Text verhelfen.

¹ Für Diskussion und Hilfe bei der Vorbereitung des Vortrags danke ich auch Frau Dr. Sabine Bartsch, Frau Deborah A. Lapeyre, und Herrn Dr. Felix Sasaki. Herrn Dr. Hans-Werner Bartz danke ich für Hilfe mit der TUSTEPkodierung des Griechischen (und bei der Vorbereitung der PDF-Version dieses Texts).

Hieraus ergibt sich mein heutiges Thema: der Text, die Textgestalt, und die Textgestaltung.

Warum sollten wir uns heute, in dieser Zeit des Multimedialen, des Vollfarb-Scannens in höchster Auflösung, der automatischen Bilderkennung, noch mit Text beschäftigen?

Der Informatiker Tim Bray, mit dem ich vor einigen Jahren eng zusammenarbeitete, hatte einmal eine Start-up-Firma mit dem Motto »Knowledge is a text-based application.« (»Wissen ist eine textbasierte Anwendung.«)

Als sich Alan Turing die Frage stellte »Wann wird es fair sein zu sagen, dass Computer denken können?«, war seine Antwort das, was uns jetzt als der Turing-Test bekannt ist. Dieser Test beruht auf dem Austausch von geschriebenen Nachrichten – also Texten – zwischen einem Computer und einem Menschen. Turing wurde dafür kritisiert, das Denken mit bloßer verbaler Fertigkeit gleichzusetzen – es gibt ja das visuelle Denken und andere nichtverbalen Formen der Intelligenz! – aber Turing hat gar nicht angenommen, dass alle Gedanken verbal sind. Er schlug nur vor, dass es fair sei, wenn eine Maschine eine ausreichende verbale Fähigkeit aufweist, das von der Maschine Geleistete »Denken« zu nennen. (Schließlich sind unsere Bibliotheken voll mit Werken von Menschen, die wir die Denker der Vergangenheit nennen, und der Hauptbeweis, dass sie gedacht haben, sind die Texte, die sie hinterlassen haben.)

Wenn das Wissen und das Denken beide so eng mit Text zu tun haben, scheint mir die Natur von Text eine zentrale Frage für alle, die die Wissenschaft fördern wollen. Dazu gehört auch die Art, wie wir Text gestalten. Ich hoffe, diese Themen zu diskutieren ist eine geeignete Art, TUSTEP, eines der avanciertesten und komplexesten Werkzeuge, die wir für die Arbeit mit Texten haben, und den Mann, dem wir seine Existenz verdanken, zu würdigen.

II.

Wenn wir wissen wollen, was Text ist, welche Gestalt er hat, ist es vielleicht sinnvoll damit zu beginnen, dass wir schauen, in welcher Gestalt man Texte vom Sender zum Empfänger sendet.

Die Linearität der Sprache

Wenn wir mündliche Texte in Betracht ziehen (die ja die Sprachwissenschaftler gemeinhin als primär betrachten), werden wir wahrscheinlich schnell darauf kommen, dass Text, was auch immer er sonst sein mag, eine eindimensionale Abfolge grundlegender Einheiten ist. Wir können als grundlegende Einheiten Sätze oder Wörter oder Töne ansetzen. Die Details der Analyse werden danach variieren. Eine Eigenschaft aber bleibt invariant: Für jedes Element im Text, ob Satz, Wort oder Phonem, gibt es genau ein nächstes Element. (Eine Ausnahme: wenn der Text endlich ist, wie bisher alle menschlichen Texte, dann hat das letzte Element im Text kein nächstes Element.) Und umgekehrt gibt es für jedes Element (außer dem ersten) genau ein sofort vorhergehendes Element.

Diese Idee scheint natürlich, fast unvermeidbar zu sein: Alle natürlichen Sprachen scheinen aus eindimensionalen Tonfolgen zu bestehen, schon allein deshalb, weil der Mensch nur einen Mund hat und das Tonsignal der menschlichen Sprache sich in Echtzeit entfaltet.

Gelegentlich ist diese Eindimensionalität Anlass zur Klage, oder Gegenstand kritischer Attacken: George Landows Darstellung der Eigenschaften von Hypertext kontrastiert diesen mit konventionellem Text, der eindimensional ist und in einer einzigen vorbestimmten Reihenfolge gelesen werden muss, von Anfang bis Ende, ohne Abweichung vom vorbestimmten Pfad. Eine vergleichbare Feindseligkeit gegenüber dem eindimensionalen Text findet sich in der Arbeit Ted Nelsons. Nelson scheint häufig die Schuld an der Eindimensionalität beim Papier zu suchen.² Und

2 Aber beileibe nicht immer: in *Literary Machines*, schreibt er an einer Stelle »The sequentiality of text is based on the sequentiality of language and the sequentiality of printing and binding. These two simple and everyday facts have led us to thinking that text is intrinsically sequential.« (1/14, »Hypertext«.

Landow scheint wiederum den Kodex verantwortlich zu machen, obgleich in Wirklichkeit die Art der Eindimensionalität, über die sie sprechen eine Funktion der menschlichen Sprache ist und nicht das Resultat der Aufzeichnung auf Papier.³

Die Nichtlinearität der Seite

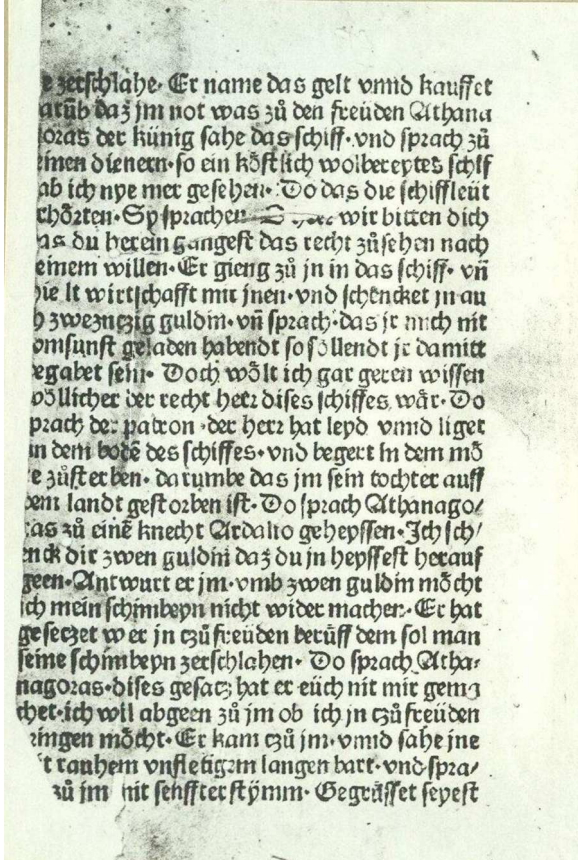
Ganz im Gegenteil: das Papier ist ja nicht ein- sondern zweidimensional.⁴ Papier bietet mehr Möglichkeiten, als die menschliche Stimme, mehrere Äußerungen »gleichzeitig« zu tätigen. Der Kodex ist dreidimensional und (Nelson und Landow zum Trotz) zwingt den Leser nicht zum eindimensionalen Lesen. Der Kodex unterscheidet sich als Textträger grundsätzlich von der Schriftrolle gerade dadurch, dass er einen direkten Zugriff auf beliebige Stellen des Inhalts bietet, während die Schriftrolle prinzipiell nur sequentiellen Zugriff bietet. Die Schriftrolle mag ein lineares Lesen erfordern, der Kodex aber nicht.

3 Es wären durchaus Sprachen vorstellbar, die nicht komplett linear sind — wenn zum Beispiel die Vokale und Konsonanten einen Signalkanal darstellten und Intonation und Tonhöhe oder Lautstärke einen anderen, davon unabhängigen Kanal. Oder man könnte sich eine Zeichensprache vorstellen bei der die linke und die rechte Hand voneinander unabhängige Bedeutungen repräsentieren. Aber natürlich vorkommende Zeichensprachen sind nun einmal linear genug, um schriftlich abgebildet zu werden. Und in Tonsprachen notiert die linguistische Analyse einfach die Tonhöhe als phonemische Eigenschaft des Vokals, der den Nukleus der Silbe bildet.

Intonation mag näher an der Unabhängigkeit und der Nicht-Linearität in nicht-tonalen Sprachen wie Deutsch und Englisch liegen. Jedoch ist der Informationswert der Intonation dieser Sprachen vergleichsweise gering, und die konventionelle Orthographie beschränkt sich auf die Unterscheidung nichtmarkierter Pitchprofile, die mit Punkt angegeben werden, und der bedeutungstragenden Profilen, die durch Fragezeichen oder Ausrufezeichen am Satzende markiert werden. Sprachen wie das Spanische hingegen, in denen Frage und Imperativsätze nicht syntaktisch unterschieden werden, markieren diese an Anfang und Ende, zeigen jedoch bislang keinen erhöhten Bedarf nach einem reicheren Inventar an Interpunktionszeichen für Pitchprofile.

4 Von der real gegebenen Dreidimensionalität des Papiers (und jeder anderen Schreibfläche) sehe ich hier der Einfachheit halber ab.

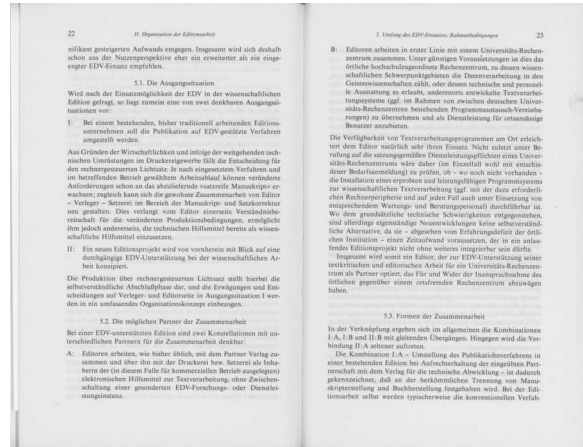
In diesem Zusammenhang mag es informativ sein, einen geschriebenen Text zu betrachten, der mehr oder weniger direkt die Idee einer eindimensionalen Abfolge von Zeichen widerspiegelt, wie in diesem 1494er Druck einer deutschen Übersetzung des Romans von Apollonius von Tyrus.⁵



Die Schrift auf der Seite ist gut lesbar, am Layout ist nichts unkonventionell, und dennoch sieht die Seite gar nicht wie eine moderne Seite aus (auch wenn man den Einsatz von Fraktur außer Acht lässt). Wenn Text im Grunde eine eindimensionale Folge von Zeichen ist, warum fühlt sich diese Seite, die den Text genau in diesem Sinne wiedergibt, nicht vertrauter an? Es ist vielleicht nicht sofort klar, aus welchen Gründen die Seite uns fremd erscheint. Die Quelle unseres Gefühls kann deutlicher werden, wenn wir eine konventionelle gedruckte Seite betrachten.⁶

5 Apollonius von Tyrus. Historia Apollonii regis Tyri, deutsch von Heinrich Steinhöwel. [Augsburg: Johann Schönsperger, um 1494.] Bl. [4]^b. (Gesamtkatalog der Wiegendrucke [GW] 0227610N.)

6 Hier S. 22–23 aus Wilhelm Ott, Hans Walter Gabler,

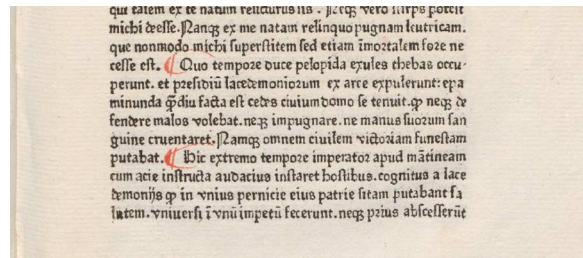


Das moderne Auge vermisst in dem Wiegendruck unter anderem

- die Seitenzahlen
- die lebenden Kolummentitel
- die Gestaltung von Absatzumbrüchen durch Zeilenbruch und Einrückung (o. ä.)

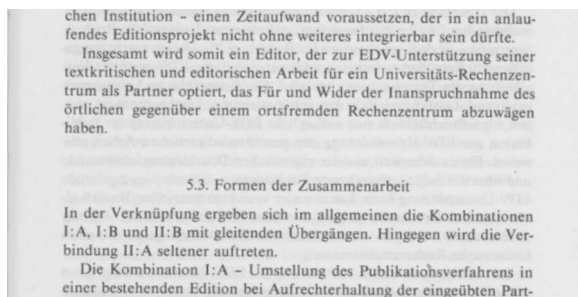
Absatz und Textunterteilung

In modernen Büchern werden Absätze mit Leerzeichen gestaltet, was bei Inkunabeln häufig nicht der Fall ist: Die letzte Zeile des einen Absatzes wird nicht ausgefüllt, und der neue Absatz beginnt in der nächsten Zeile nach einer Einrückung.⁷ We-



und Paul Sappeler, *EDV-Fibel für Editoren*, im Auftrag und in Zusammenarbeit mit der Arbeitsgemeinschaft Philosophischer Editionen der Allgemeinen Gesellschaft für Philosophie in Deutschland (Stuttgart: frommann-holzboog; Tübingen: Niemeyer, 1982).

7 Das erste Bild zeigt Absatzangaben mittels Rubrizierung in GW M34393, Plutarchus: Apophthegmata, lat. Übers. Franciscus Philephus. Daran: (Pseudo-)Petrarca, Francesco [Nicolaus de Cusa]: De vera sapientia. Utrecht: Nicolaas Ketelaer und Gerard Leempt, [um 1473]. 2^o. Das zweite Bild zeigt moderne Absatzumbrüche, S. 22–23 der *EDV-Fibel* (s. o.).



gen der Verschiebung in die nächste Zeile ist der hier gezeigte moderne Absatzumbruch ein zweidimensionales Phänomen – das gleiche gilt für den abgebildeten Abschnittskopf noch stärker. Wir können diese auch in einen linearen Zeichenstrom einfügen, indem wir Funktionen wie »Rücklauf«, »Zeilenvorschub«, »vertikaler Sprung« und »horizontaler Sprung« als spezielle Arten von Zeichen behandeln, die in den Textfluß eingebettet sind wie alle anderen. Auf diese Weise haben Teletypemaschinen die Textformatierung verwaltet. Und in der Tat reservieren alle standardisierten codierten Zeichensätze für den Computereinsatz Platz für »Steuerzeichen« wie diese.

Dass Wiegendrucke keine modernen Absatzumbrüche oder Abschnittsumbrüche haben, dient als nützliche Erinnerung daran, dass unsere modernen typografischen Erwartungen genau das sind – eine junge Entwicklung, die nicht mehr als einige Jahrhunderte alt ist. Die Layout-Konventionen, die wir heute als selbstverständlich empfinden, erfordern zu einem großen Teil das Denken in zwei Dimensionen, nicht nur in einer.

Wir gestalten nicht nur in modernen Büchern Absatzumbrüche anders, wir haben auch, glaube ich, mehr Absatzumbrüche, genauso wie wir moderne Texte feiner untergliedern. Literarische Texte sind jetzt fast immer in Kapitel und Absätze unterteilt; lange Gedichte sind seit Dante und Tasso und Vergil in Cantos und Bücher aufgeteilt. Homer ist auch in Bücher unterteilt, aber diese Gliederung ist eine posthomerische (alexandrinische) Neuerung, ebenso wie die Aufteilung von Beowulf in Fittes und die Aufteilung des Nibelungenliedes in Aventiuren. Wie kommt es, dass wir unsere Texte nun feingranular in Kapitel und Abschnitte und Unterabschnitte unterteilen? Ist der regelmäßige, ungebrochene Textblock so unwillkommen geworden, dass wir strukturelle Brüche in unsere Texte einführen, um einen Anlass für

die Einführung von Leerzeichen auf der Seite zu haben?

Die Seitenzahlen

Seitenzahlen illustrieren überzeugend eine wichtige Eigenschaft gedruckter Bücher: wenn Sie und ich Kopien der selben Edition besitzen, kann ich Sie auf eine bestimmte Passage verweisen, indem ich deren Seitenzahl benenne.⁸ Das vereinfacht die Diskussion (und damit die Untersuchung) von Texten dramatisch. Da eine Handschrift in der Regel die Seiteneinteilung der Vorlage nicht beibehält, sind solche Seitenreferenzen in der Schreiberkultur eher selten. Um präzise auf Textstellen verweisen zu können, ist es notwendig⁹ ein System zur Identifizierung der Stellen zu schaffen. Die in West-Europa zur Referenzierung von Textstellen in den Evangelien und zur Erstellung von Evangelienharmonien und rudimentären Konkordanzen verwendeten Eusebischen Einteilungen sind ein solches Verweissystem. In einer Druckkultur, wo alle Seiten nummeriert sind, hat jeder Text ein gut benutzbares Referenzsystem.¹⁰

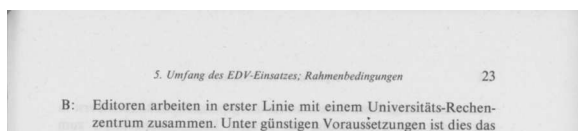
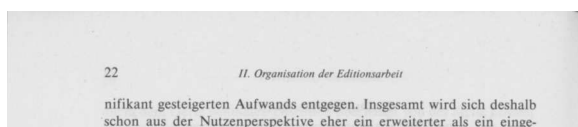
In Büchern, die heutzutage gedruckt werden, identifizieren lebende Kolummentitel oft sowohl das Buch als auch das Kapitel des Buches. Dies ist selten (wenn nicht gar unerhört) in den Handschriften der Antike und des Mittelalters, und es

⁸ Hier wie in anderen Punkten folge ich der überzeugenden Analyse von Elisabeth Eisenstein, *The printing press as an agent of change: communication and cultural transformation in early modern Europe* (Cambridge: Cambridge University Press, 1985), später gekürzt und erweitert als *The printing revolution in early modern Europe* (Cambridge: Cambridge University Press, 2005); dt. als *Die Druckerpresse: Kulturrevolutionen im frühen modernen Europa* (Wien [u. a.]: Springer, 1997).

⁹ Oder zumindest ist mir kein alternativer Weg bekannt.

¹⁰ Selbstverständlich führt die Tatsache, dass die Paginierung von Edition zu Edition variiert dazu, dass je wichtiger der Text ist (und je mehr verschiedene Ausgaben und Übersetzungen vorliegen), desto weniger nützlich werden die Seitenzahlen sein. Zudem sind Seitenzahlen in E-Books selten nützlich, da diese sich selbst neu paginieren, wenn der Nutzer die Schriftgröße verändert. Aber hier ist nicht der Ort, um in eine detaillierte Diskussion der technischen und sozialen Auswirkungen von Referenzsystemen im elektronischen Zeitalter einzugehen.

ist verlockend zu spekulieren, dass es die relative Knappheit oder Fülle von Büchern widerspiegelt. In einer klösterlichen Gemeinschaft, in der man ein einzelnes Buch in der Zelle für die stille Lektüre hat, wird es relativ selten der Fall sein, dass man den Wunsch verspürt durch einen Blick auf die Seite sagen zu können, welches Werk das Buch enthält. Wenn man nur ein Buch in der Zelle hat, wie schwer wird es wohl sein, sich daran zu erinnern, welches Buch es ist? Es ist der Leser, dessen Schreibtisch mit offenen Büchern übersät ist, der solche Metadaten benötigt. Wenn man leicht Zugang zu vielen Büchern hat, ist es natürlich bequem, auf einen Blick sehen zu können, welches Buch der Aristoteles ist und welches Plutarch und welches Augustinus.



Die allmähliche Übernahme der Titelseite spiegelt, so glaube ich, die selbe Entwicklung wider: Titelseiten sind wichtig für die schnelle Identifizierung von Büchern, wenn viele davon herumliegen. Die Drucker selbst waren mutmaßlich die ersten Nutznießer von Titelseiten: sobald Drucken zum Geschäft wird, wird ein Drucker, der mehrere Bücher zum Verkauf anbietet, es einfacher finden seinen Bestand zu verwalten, wenn es einfach ist zu sehen, welches Buch welches ist. Seitenzahlen dienen ebenfalls einem Zweck in der Druckerei; es wird so einfacher zu überprüfen, dass die Einrichtung der Form korrekt ist und die Bögen in der richtigen Reihenfolge zusammengestellt worden sind. Die Verschiebung von den Blattbezeichnungen wie Ai, Aii, Aiii, Aiiii zu Seitenzahlen wie 1, 2, 3, 4 steht im engen Zusammenhang mit der Erkenntnis, dass Seitenzahlen auch für andere Leser von Nutzen sind.

Die Verwendung von zweidimensionalen, visuell auffälligen Absatzumbrüchen, die Verwendung von lebenden Kolummentiteln zur Identifikation des Kapitels oder Abschnitts, und die Verwendung von Seitenzahlen helfen tendenziell dem

Leser, der etwas im Buch sucht. Dem systematisch das Buch durcharbeitenden Leser helfen sie weniger, wenn überhaupt.

Wesentlich oder unwesentlich?

In das Textmodell als eindimensionaler Wortstrom ohne weitere interne Struktur lassen sich weder Seitennummern noch die lebenden Kolummentitel leicht einfügen. Wir können einen solchen Text nur dann mechanisch verarbeiten, wenn wir die Seitenzahlen und the Kolummentitel irgendwie von dem anderen Material im Textstrom unterscheiden können. Dazu dient die Textauszeichnung oder Inline-Markup: XML, HTML, Wiki-Markup oder andere.

Um das eindimensionale Modell zu retten, könnten wir vielleicht Seitenzahlen und Kolummentitel und den modernen Stil von Absatzumbrüchen und Abschnittsüberschriften als Randthemen abtun. Ja, natürlich, wenn wir einen Text auf mehreren Seiten drucken, kann es sinnvoll sein, die Seiten zu beschriften. Aber dann wird nicht der Text, sondern es werden die Seiten nummeriert (oder so könnten wir argumentieren): die Seitenzahlen und die lebenden Kolummentitel sind Nebeneffekte der materiellen Darstellung des Textes und nicht Teil des Textes selbst. (Wir bestehen z. B. nicht darauf, dass die Seitenzahlen in einer neuen Ausgabe eines Textes gleich bleiben.)

Ein logischer Fehler ist diese Denkweise wohl nicht; es geht nur um die Frage, wo man eine Grenze ziehen will zwischen Text und den kulturellen Praktiken, mit denen man Texte schafft, speichert, verbreitet, und gebraucht. Es kann nützlich sein, klar zwischen Texten und Büchern zu unterscheiden. Aber wenn wir uns für Texte interessieren und wenn wir Software bauen wollen, die uns hilft, Texte zu produzieren, zu verwalten oder zu untersuchen, werden wir uns auch für Bücher und andere Formen interessieren, in denen Texte dargestellt werden können. Unsere Software muss sowohl über Bücher wie auch über Texte Bescheid wissen. Wenn nicht, dann wird die Gestaltung des Texts durch diese Software immer etwas fremd anmuten, ähnlich wie der 1494 erschienene Apollonius von Tyrus. Wenn die Seitenzahlen nicht zum eigentlichen Text gehören, dann darf unsere Aufmerksamkeit nicht nur dem eigentlichen Text gelten, sondern wir müssen eine breitere

Denkkategorie finden, die den eigentlichen Text sowohl wie auch die Randerscheinungen wie Seitenzahlen, lebenden Kolummentitel, und verünftigte (d. h. moderne) Absatzumbrüche beinhaltet.

III.

Anmerkungen, Paralleltex-te, apparatus criticus

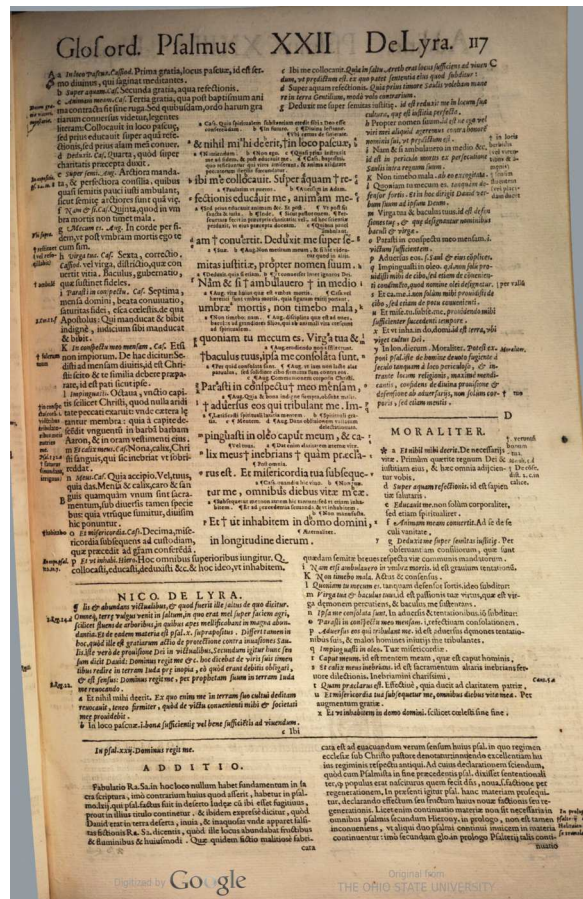
Die Vorstellung, dass Text eine einfache Abfolge von Wörtern oder Zeichen sei, wird noch weniger haltbar, wenn wir wichtige oder zentrale kulturelle Objekte ins Betracht ziehen. In fast jeder Kultur werden wichtige Texte kommentiert. Oft erhalten sie mehrere Kommentare und Annotationsstufen, wie auf dieser Seite einer venezianischen Ausgabe der Bibel (lateinisch), zusammen mit einer interlinearen Glose, der Glossa ordinaria, die früher Walafrid Strabo zugeschrieben wurde, und der Postilla von Nicolaus de Lyra, einem der beliebtesten mittelalterlichen Bibelkommentare.¹¹

Wenn man eine beliebige Stelle dieser Seite nimmt, wie groß ist dann die Wahrscheinlichkeit, dass das nächste Wort im Text gleich rechts auf derselben Zeile steht? Das Textmodell als eine eindimensionale Abfolge hat schon mit Seitenzahlen seine Schwierigkeiten; mit diesem Werk wird das eindimensionale Modelle gar nicht fertig.

Auf dieser Seite der Glossa ordinaria können wir nicht einen, sondern sieben Text-Ströme unterscheiden, die Seite für Seite synchronisiert werden müssen:

- der Grundtext der Bibel
- eine Glose zwischen den Zeilen, die die Wörter oder Satzkonstruktion des Textes erklärt
- In den inneren Rändern stehen Buchstaben, die den Bibeltext mit entsprechenden Passagen in der Glossa ordinaria oder in dem Kommentar von Nicolaus de Lyra verknüpfen.
- die Glossa ordinaria selbst
- der Kommentar von Nicolaus de Lyra

11 *Biblia sacra cum glossis, interlineari, et ordinaria / Nicolai Lyrani postilla, ac moralitatibus, Burgensis additionibus, & Thoringi replicis.* Venetiis: [s.n.], 1588.



- In den äußeren Rändern stehen Verweise auf Parallelstellen in anderen Büchern der Bibel (gekennzeichnet durch intratextuelle Symbole im Bibeltext).
- Auch in den äußeren Rändern stehen bibliographische Hinweise auf Autoren und Werke, die in der Glossa ordinaria oder von Nicolaus de Lyra erwähnt werden.

Der Bibeltext hat einen klaren Vorrang der Glossa ordinaria und dem Kommentar von Nicolaus de Lyra gegenüber: der Bibeltext ist sowohl intellektuell als auch topographisch zentral, die anderen sind peripher. Aber in anderen Fällen (wie z. B. unten die Complutensische Polyglotte von 1522) handelt es sich nicht um Basistext und Kommentar, sondern um mehrere Versionen desselben Texts (hier das IV. Buch Mose, Deuteronomium), ohne, dass einer von ihnen notwendigerweise den Vorrang hätte.¹²

12 *Vetus testamenti multiplici lingua nūc primo impressum, Et imprimis Pentateuchus Hebraico Greco*



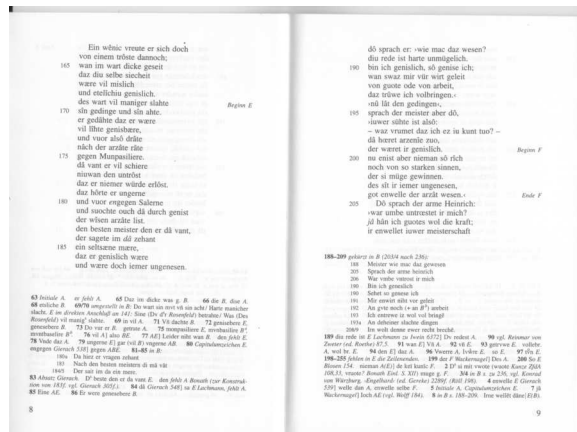
COMPLUTENSIS POLYGLOT. DEUT. XXXII, 33-46
 UPPER PART: HEBREW, VULGATE, SEPTUAGINT (WITH INTERLINEAR LATIN TRANSLATION)
 LOWER PART: CHALDAIC, WITH 7 LATIN TRANSLATIONS. IN THE LEFT-HAND MARGIN ARE GIVEN HEBREW AND PHALGIAN ROOTS, RECORDED INTO FIVE ONE OF FOUR COLUMNS OF ORIGINAL HEBREW (AND COPIES) IN THE NEW YORK PUBLIC LIBRARY

Diese Beispiele komplexer typographischer Arbeiten legen nahe, Text nicht als eindimensionale Zeichenabfolge, sondern als Bündel aus einer oder mehreren solcher Abfolgen zu betrachten, die an verschiedenen Punkten synchronisiert sind. Um solches Material auf die Seite zu bringen, muss man sicherzustellen, dass jeder gegebene Synchronisationspunkt für alle parallelen Ströme auf dieselbe Seite fällt. Dies ist eine komplexe Aufgabe, was ein milde auszudrücken, was ein Grund dafür sein mag, dass es so wenige Bücher gibt, die mit so vielen parallelen Strömen arbeiten wie die Complutensische Polyglotte.

Wenn selbst die einfachsten konventionellen modernen Drucke kaum eindimensional sind, sollten wir vielleicht nicht überrascht sein, dass komplexere Texte noch weniger eindimensional sind,

atq[ue] Chaldaico idiomate: adiūta unicuiq[ue] sua latina interpretatione Im[pre]ssa in ... copluti vniversitate ... : sumptibus ... F. Francisci Ximenez de Cisneros ... : industria & solertia ... Arnaldi Guilielmi de Brocario artis impressorie magistri ... Gedr. 1514; veröffentlicht 1520.

noch dass wissenschaftliche Ausgaben, die die Naharbeit an dem Text ermöglichen sollen, auch typographisch komplex sind.¹³



Herkömmliche gedruckte wissenschaftliche Ausgaben bieten neben dem Basistext eine Reihe verschiedener Arten von Informationen und unterscheiden typographisch verschiedene Informationsarten:

- Zeilennummern
- Anmerkungen
- apparatus criticus (manchmal gibt es mehrere Apparate, jeder einer anderen Informationsart gewidmet)

Ich beschreibe diese Seite im Vergleich zu anderen modernen Büchern als komplex; nach den Maßstäben von TUSTEP ist es natürlich keine besonders komplexe Seite.

Textvariation

Eine äußerst reiche Komplexitätsquelle bietet die unausweichliche (wenn auch oft unwillkommene) Tatsache, dass Texte variieren. Sie verändern sich, sie bleiben sich nicht gleich, sie sitzen nicht fest. Kein Text, der in mehreren Textzeugen vorhanden ist, kann auf eine einzelne Wortfolge reduziert werden, aus dem einfachen Grunde, dass bei solchen Texten alle Textzeugen praktisch nie in der Wortfolge übereinstimmen.

13 Hartmann von Aue, *Der arme Heinrich*, hsg. Hermann Paul, 16., neu bearbeitete Auflage besorgt von Kurt Gärtner (Tübingen: Niemeyer, 1996).

Wie zu erwarten, gibt es verschiedene Ansätze, um die Textvariation zu modellieren oder zumindest in elektronischer Form darzustellen. Die Zeit würde keinen vollständigen Überblick erlauben, selbst wenn mir alle Techniken bekannt wären, die bei der Herstellung von wissenschaftlichen Ausgaben, im juristischen Verlagswesen und in der technischen Redaktion verwendet wurden. Ich beschränke mich darauf, einige der Ansätze zu erwähnen, denen man in den Digital Humanities begegnet.

Eine stets beliebte Lösung für das Problem der Textvariation ist, so zu handeln, als ob es kein Problem der Textvariation gäbe. Man gibt z. B. ein Werk nach einer einzigen Handschrift heraus (das ist seit dem Aufstand Joseph Bediers gegen die lachmannsche Stammbaumkritik in den 20er Jahre des 20. Jhs. die übliche Praxis in der altfranzösischen Philologie; auch in der Altgermanistik hat diese Editionsmethode seit Wilhelm Grimm immer wieder Anhänger). Oder man gibt vielleicht einen kritischen Text ohne Handschriftenapparat aus; das ist in Texten für das breite Lesepublikum üblich (und kommt auch gelegentlich vor, wenn der Herausgeber mit dem Apparat nie fertig wird). (Hier der kritische Text einer Strophe des mittelalterlichen Dichters Reinmars des Alten, in der Ausgabe von Moser und Tervooren.¹⁴)

Ich wirbe umbe allez, daz ein man
ze wêltlichen vrôiden iemer haben sol.
daz ist ein wîp, der ich enkan
nâch ir vil grôzem werde niht gesprechen wol.
Lobe ich si, sô man ander vrouwen tuot,
daz engenîmet sie niemer tac von mir vûr guot.
doch swer ich des, si ist an der stat,
dâs ûz wîplichen tugenden nie vuoz getrat.
daz ist *in mat!*

Aber auch wenn wir dem Leser die Varianten der Handschriftenüberlieferung vorlegen wollen, können wir das schaffen, ohne die Textvariation in das Textmodell zu integrieren. Wir können jeden Text einzeln abschreiben, ohne zu versuchen, die variablen Stellen des Texts explizit anzugeben. Stattdessen geben wir alle Zeugen des Textes parallel

¹⁴ *Des Minnesangs Frühling* unter Benutzung der Ausgaben von Karl Lachmann und Moriz Haupt, Friedrich Vogt und Carl von Kraus, bearbeitet von Hugo Moser und Helmut Tervooren, 36., neugestaltete und erweiterte Auflage (Stuttgart: S. Hirzel, 1977), Bd. I, S. 305.

an und überlassen es dem Leser zu sehen, wo die Unterschiede liegen. Bezogen auf die Reinmarstrophe würde das Ergebnis in etwa so aussehen:

A:

Ich wirbe umbe alles dc ein m̄.
ze weltlichen froids iemer haben fol
dc ift ein wip *der* ich enkan.
nach ir wol grozer werdeheit gesprechen wol/
fobich fi, fo man and frowen tuot.
dc engenimet eth fi v̄ mir niht fvr guot.
doch fwer ich def fiu stet noch hiute *ander* stat.
dc uzer wiplichen tugendē nie fuoz getrat.
dc ist iv mat.

b:

Ich wirbe umbe alles das ain man.
ze weltlichen vroiden iemer haben fol.
das ift ain wip der ich enkan.
nach ir vil groffen werde niht gesprechen wol.
lobe ich fi fo man *ander* vrowen tuot.
das engenimet fi niemer tag von mir verguot.
doch fwer ich des fi ift an der stat.
das us wiplichen tugende nie fuos getrat.
da ift div mat.

C:

Ich wirbe umbe alles das ein man.
ze *wertlichen* frôidē iemer habē fol.
dc ift ein wib der ich enkan.
nach ir vil groffem werde niht gesprechē wol.
lob ich fi fo man *ander* frowē tuot.
dc genimt fi niemer tac v̄ mir verguot.
doch fwer ich def fift an der stat
da fi us wiplichē tugēden nie fuos getrat.
da ist div mat.

E:

Ich wirbe umme allez daz ein man
ze wereltlichen frauden haben fol.
daz ift ein wip der nieman kan.
noch ir grozzen werdekeit gesprechē wol.
lobe ich fie fo man *ander* frauwen tuot.
daz en nimet fie von mir niht fuer guot.
doch fwuere ich wol fie ift *ander* stat,
do fie uz wiplichē tugenden nie fuoz getrat.
dar ist in mat!

Der übliche wissenschaftliche apparatus criticus wurde bereits erwähnt; weiter unten sieht man eine Nahaufnahme der Apparate in der Minneangs-Frühling-Ausgabe von Moser und Tervooren. In dem ersten Apparat geben Moser und Tervooren Handschriftenlesarten, im zweiten Apparat die Lesarten der früheren Herausgeber des Gedichts.¹⁵

X. 1 35] 43 C. — 2 *wereitlichen f. haben* E. 3 *der nieman kan* E. 4 *ir wol grozer werdeheit* g. A. *ir grozzen werdeheit* g. E. *grossen b.* 5 *Lob C.* *Sobich (?) A.* *and frowen A.* 6 *genimt C.* *De e. etb (Daz en nimet sie E) si vō mir niht fur g.* *AE.* *verguot bC.* 7 *sist C.* *sū stet noch hōte andē s.* *A.* *swāre ich wol sie ist E.* 8 *Da si uz CE.* *De uzer A.* *sie E.* *tugende b.* 9 *De ist iv A.* *Da ist dō bC.* *Dar ist E.*

2 36] 44 C. — 1 *So A.* *Als CE.* *etsuenne E.* *der muot E.* 2 *Dvr AC.* *sin C.* *grozze vntete raten E.* 3 *on ich gefriunde mir ein A.* 4 *Son wil daz hertze eht niendert denne dar E.* 5 *So wol A.* *im C.* *des das es bC.* *dc ez A.* *der E.* *so reine AE.* *weln ACE.* 6 *Daz er mir E.* *sōze A.* *arbeit C.* *erebeitte E.* 7 *Des E.* *ein wip E.* 8 *Dem i. z. dienste AC.* *were AC.* *Der ich diene wer ez al der wēlede E.* 9 *Mōz AE.* *gebor C.*

X ¶ Str.-Folge nach E, d. i. 1, 4, 2, 5, 3 K(LV). — 1, 2 *wereitlichen* K(LV). 4 *grōzen werdeheit* g. K(LV). 6 *Dazn nimet eht si von mir niht f.* g. K(V). 7 *sist* K(LV). 8 *tugenden* K(LV). 9 nach K(LV); vgl. auch L Anm., V Anm. sowie Kralik, Walther gegen Reinmar 1955, 10, Wapnewski Euph. 60, 17 ff.

2, 5 *deiz so reine* K(LV). 6 *arebeitte* K(LV). 7 *Doch L* Des K(V). 9 *Muoz* K(LV).

Als die Gelehrten des 16. und 17. Jahrhunderts die moderne textkritische Tradition begründeten, mussten sie eine Methode finden, auf einer gedruckten Seite einen umfassenden oder auswählenden Überblick über die Lesarten zu geben, die in der Überlieferung von Belang sind. Die gesuchte Methode musste wirtschaftlich sein (so sollte sie auch vorzugsweise kompakt und typographisch nicht allzu schwierig sein). Ich glaube, der herkömmliche kritische Apparat ist eine brillante Lösung dieser Herausforderung. Aber man muss zugeben, dass den allerwenigsten Lesern die Darstellung der Varianten mittels eines solchen Apparats einleuchtet.

Viele Leser finden die Darstellungstechnik der Zeilensynoptik intuitiver: Sie besteht darin, den Text verschiedener Zeugen zeilenweise untereinander zu geben. Das lässt sich leicht machen in poetischen Texten, wo die Verszeile eine natürliche Einheit für die Organisation der Information bildet.

Bedingung für die zeilensynoptische Wiedergabe eines Texts ist nur, dass man weiß, welche Zeilen (oder allgemeiner welche Textsegmente) in jedem Textzeugen einander entsprechen. Formal bedeutet dies, eine Äquivalenzrelation über die Textsegmenten zu definieren; in der Praxis kann es die relativ einfache Form der Zuordnung kanonischer Bezeichner zu den Zeilen (oder Segmenten) der Textüberlieferung annehmen, so dass Software wissen kann, dass Zeilen in verschiedenen Doku-

menten, die den gleichen Bezeichner teilen, Varianten voneinander sind. (Hier die letzten drei Zeilen der Reinmarstrophe.)

- A: doch fwer ich def, fiu stet noch hiute ander stat.
- b: doch fwer ich des fi ift an der stat.
- C: doch fwer ich def list an der stat
- E: doch fwuere ich wol fie ift ander stat,

- A dc uzer wiplichen tugenden nie fuoz getrat.
- b das us wiplichen tugende nie fuos getrat.
- C da fi us wiplichē tugēden nie fuos getrat.
- E do fie uz wiplichē tugenden nie fuoz getrat.

- A dc ist iv mat.
- b da ift div mat.
- C da ist div mat.
- E dar ist in mat!

Eine Methode, die ich von Manfred Thaller gelernt habe (der sie vielleicht erfunden hat), ist, jeden Textstrom parallel zu segmentieren, so dass entsprechende Teile des Textes nebeneinander angezeigt werden können. Dies ist effektiv die Zeilensynoptik, aber bei einer feineren Gliederung des Texts.

doch om.	swer <i>AbC</i>	ich om.	des <i>AbC</i>
	swuere <i>E</i>		wol <i>E</i>

siu stet noch hiute <i>A</i>	ander <i>AE</i>	stat om.
si ist <i>bE</i>	an der <i>bC</i>	
sist <i>C</i>		

dc <i>A</i>	uzer <i>A</i>	wiplichen <i>A</i>
das <i>b</i>	us <i>bC</i>	
da si <i>C</i>	uz <i>E</i>	
do sie <i>E</i>		

tugenden <i>ACE</i>	nie om.	fuoz <i>AE</i>	getrat om.
tugende <i>b</i>		fuos <i>bC</i>	

dc <i>A</i>	ist om.	iu <i>A</i>	mat om.
da <i>bC</i>		diu <i>bC</i>	
dar <i>E</i>		in <i>E</i>	

Diese tabellarische Darstellung zeigt die Anwendung dieser Methode auf den Reinmartext. Jedes

15 Moser/Tervooren, S. 305.

selbstverständlich in einer guten Druckausgabe vor der Fertigstellung des Wörterbucheintrags erneut manuell geprüft werden. Man sah daher keinen Bedarf für einen elektronischen Text, der zwischen Groß- und Kleinschreibung unterscheiden musste.

Einige Jahre später führte IBM ein neues Lochkartengerät (die 029) ein, das über mehrere neue Zeichen – darunter auch Klammern! — verfügte. Der neue Zeichensatz:

```
0123456789
ABCDEFGHIJKLMN
OPQRSTUVWXYZ
&¢.<(+|-!$*);-/_ ,_>?:@'="
```

Wie sich einige von Ihnen erinnern werden und andere sich vorstellen können, war die Situation für Zeichen mit Diakritika (wie ä, ö, ü) oder für nicht englische Buchstaben der lateinischen Schrift wie etwa Esszett oder isländisch Edh (ð) und Thorn (þ), noch schwieriger. Die Entwicklung standardisierter Zeichensätze für griechische, hebräische, arabische und andere Schriften dauerte noch länger.

Die Entwicklung kodierter Zeichensatzstandards war langsam, und kommerzielle Interessen widerstanden der Standardisierung von Zeichen, die sie für kommerziell belanglos hielten. Sie beanstandeten zum Beispiel Vorschläge, die kyrillischen Schriftzeichen in ISO 8859 einzugliedern, die kurz nach der bolschewistischen Revolution in Russland abgeschafft wurden.

Unter diesen Umständen mussten Wissenschaftler, die mit älteren Texten und minderheitssprachigen Texten arbeiteten, ihre eigenen Wege gehen. Man hat z. T. die benötigten Zeichen durch Umschrift (wie im Thesaurus Linguae Graecae) oder durch Sonderzeichfolgen nachgebildet. Hier z. B. ein kurzer griechischer Text in Beta Code und in der TUSTEP-Kodierung.

```
γνῶθι σεαυτόν
GNW=QI SEAUTO/N
#g+gn%?vui seayt%/on#g-
```

TUSTEPs Behandlung der griechischen, hebräischen, und der selteneren lateinischen Zeichen benutzt diese zweite Methode; TUSTEP entwickelte früh einen größeren Bestand an vordefinierten Zeichen als jede andere wissenschaftliche Textverarbeitungssoftware, mit der ich vertraut bin.

Die Standardisierung von Zeichensätzen ist wichtig, und die Entwicklung des Universal Character Set (UCS) von ISO 10646 / Unicode ist ein großer Fortschritt für alle, die sich für maschinelle Textverarbeitung interessieren. Aber wenn unsere Werkzeuge auf Texte aus allen Sprachen, aus allen Perioden und für alle wissenschaftlichen Zwecke angewendet werden sollen – und für die Wissenschaft sind dies die Ziele – dann muss man schnell einsehen, dass keine endliche Aufzählung von Zeichen vollständig in diesem Sinne sein kann. Es gibt Schriftsysteme, die wir noch nicht vollständig verstehen.

In der Paläographie interessiert man sich z. B. nicht nur für die Buchstaben, sondern auch für deren Form. Informationen dieser Art gehen normalerweise in elektronischen Darstellungen verloren, da sie in der Normalschrift keine Bedeutungsträger sind. Die im Mittelalterlichen Nordischen Textarchiv (MeNoTA) verwendeten Zeicheninventare unterscheiden nicht nur zwischen dem gekrümmten insularen (d. h. irischen) d und dem aufrechten (oder kontinentalen) d, sondern man unterscheidet zudem vier verschiedene Formen des Kleinbuchstabens e, sechs Formen des Kleinbuchstabens a, neun Formen des Kleinbuchstabens f.



Wenn unsere Textverarbeitungstools fundiert sind, sollten wir sehr selten neue Zeichen für unsere Arbeit definieren müssen, weil die Zeichen, die wir brauchen, bereits Teil des Systems sind. (Zumindest, wenn wir TUSTEP benutzen.) Aber Werkzeuge, die für den wissenschaftlichen Gebrauch gedacht sind, müssen es uns auch erlauben, neue Zeichen hinzuzufügen, wann und wie wir sie brauchen. Aus diesen Gründen müssen XML-Prozessoren Zeichen im Unicode-Bereich »Private Use Area« akzeptieren, auch wenn dieser Gedanke manche Leute empört. Aus diesen Gründen erlaubt auch TUSTEP Geschichte eine so breite Abdeckung des historischen die Definition neuer Zeichen, wenn dies notwendig ist.

OHCO

Was die Organisation des Textes oberhalb der Zeichenebene betrifft, so wurden im Laufe der Jahrzehnte eine Vielzahl von Datenformaten entwickelt, die auf ihre Art eine bestimmte Konzeption von Text und die entsprechenden Textoperationen widerspiegeln, die unterstützt werden müssen. Ich werde keine historische Übersicht darüber versuchen, obwohl ich glaube, dass die Hermeneutik von Datenformaten eine lohnende Übung ist.

Stattdessen möchte ich einige Jahrzehnte überspringen und über den bisher ernsthaftesten Versuch nachdenken, ein allgemeines Textmodell prägnant zu beschreiben. Ich meine das Modell, das Steven DeRose, David Durand, Elli Mylonas und Allen Renear, damals alle an der Brown University, unter dem Titel »What is text, really?« (»Was ist eigentlich Text?«) vorgelegt haben.¹⁶

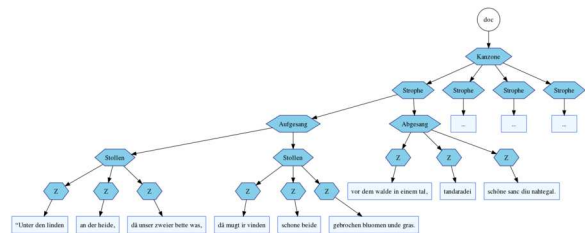
Wie viele von Ihnen vielleicht wissen, definieren die Autoren ein Textmodell als »geordnete Hierarchie von Inhaltsobjekten« (engl. »Ordered Hierarchy of Content Objects«), das seither als »OHCO-Modell« diskutiert wird. Die Autoren beschreiben kurz eine Reihe alternativer Textmodelle und argumentieren für die Überlegenheit ihres Modells, bevor sie einige verbleibende Herausforderungen für die Zukunft beschreiben. Die entscheidende Behauptung, die mit dem Papier verbunden ist (und weithin »OHCO-These« oder »OHCO-Hypothese« genannt wird) lautet: »Wir meinen, dass man es nicht deutlich genug sagen kann: Text ist eine geordnete Hierarchie von Inhaltsobjekten.« Das OHCO-Modell von Text ist eindeutig ein Versuch der Autoren, die Tugenden zu erklären, die sie in der Standard Generalized Markup Language (SGML) – und nun im Nachfolger von SGML, genannt Extensible Markup Language (XML) – sehen, und das OHCO-Modell wird häufig als die philosophische Grundlage von SGML und XML gesehen.

Die Idee, dass ein Text beliebiger Komplexität auf eine einfache hierarchische Anordnung von Teilen reduziert werden kann, erscheint vielen Textwissenschaftlern lächerlich, weshalb das OH-

CO-Modell und die SGML- und XML-Spezifikationen, die darauf basieren, weithin kritisiert wurden als: einfach, uninformativ, sogar autoritär. (Manche Geisteswissenschaftler haben ein reflexives Misstrauen gegenüber allem, was Hierarchie genannt wird.) Jedes zufällig ausgewählte Buch zeigt Beispiele von Struktureinheiten, die nicht richtig verschachtelt sind und somit keine Hierarchie bilden: Texte sind in Seiten unterteilt (deren kulturelle Bedeutung wir bereits bemerkt haben) und auch in Absätze, die sich oft über Seitengrenzen erstrecken. Wenn die Elemente jedoch nicht verschachtelt sind, bilden sie keine Hierarchie.

Ich möchte sowohl das OHCO-Modell als auch die SGML- und XML-Spezifikationen gegen diese Kritik verteidigen.

Es ist wahrscheinlich richtig, dass das OHCO-Modell eine ziemlich gute Übereinstimmung mit dem Ansatz zur Textcodierung und -verarbeitung bietet, der von den meisten frühen Nutzern von SGML und XML verwendet wird. Die Entwickler von SGML- und XML-Vokabularen identifizieren verschiedene Arten von Textobjekten und stellen diese (im einfachen Fall) als SGML- oder XML-Elemente dar, die ineinander verschachtelt sind und somit eine Verschachtelungs-Hierarchie bilden. In einer Dokumentherstellungsumgebung sind Seiten das Ergebnis der Verarbeitung und nicht die Eingabe für die Verarbeitung. Wenn sich das Dokument von Version zu Version ändert, kann und sollte sich die Seitennummerierung ändern. Die frühen Anwender von SGML markierten daher regelmäßig Absätze, aber keine Seiten als Struktureinheiten des Textes.¹⁷



Aber das eine bedingt nicht notwendigerweise das andere. Um zu glauben, dass SGML und XML in der Praxis nützlich sind, ist es nicht notwendig zu behaupten, dass Text von Natur aus eine geord-

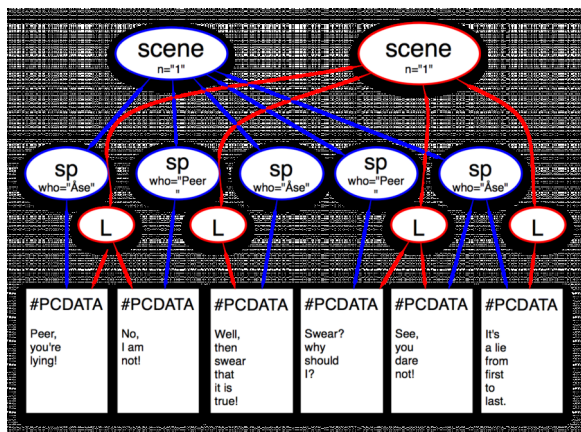
¹⁶ DeRose, Steven J., David G. Durand, Elli Mylonas, and Allen H. Renear. What is text, really? *Journal of Computing in Higher Education* 1, no. 2 (1990): 3–26. doi:<https://doi.org/10.1007/BF02941632>.

¹⁷ Das folgende Bild zeigt eine idiomatische Darstellung in XML der Strophenstrukturen in einer Kanzone von Walther von der Vogelweide.

nete Hierarchie von Inhaltsobjekten ist. Nichts in den XML- oder SGML-Spezifikationen macht einen solchen Anspruch explizit oder implizit geltend. Um diese Technologien nützlich zu finden, genügt es zu glauben, dass Text unabhängig von der zugrundeliegenden platonischen Textwirklichkeit (vorausgesetzt, er hat eine, an dem natürlich Aristoteliker und postmoderne Konstruktivisten zweifeln) als geordnete Hierarchie von Inhaltsobjekten modelliert werden kann.

Und dass Texte mit Hilfe von SGML- und XML-Elementen sinnvoll modelliert werden können, zeigt sich vielleicht an dreißig Jahren erfolgreicher SGML-Anwendungen. *Se non è vero, è ben trovato.*

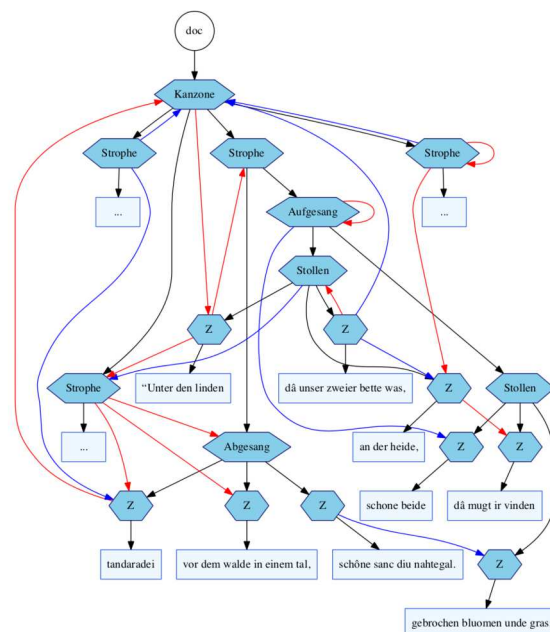
Die SGML-Spezifikation definiert darüber hinaus einige Konzepte, die der Behauptung, dass ein Text eine einzelne Objekthierarchie ist, direkt widersprechen: Mit dem optionalen Feature CONCUR können mehrere Elementhierarchien in einem Dokument definiert werden. Wenn wir nicht nur die Spiel- / Akt- / Szenen- / Zeilenhierarchie von Shakespeares Dramen aufzeichnen wollen, sondern auch die Seitengrenzen des Ersten Folio (da die korrekte Interpretation der Rechtschreibung und des Textlayouts im ersten Folio von Seite zu Seite variiert) Seite), können wir dies mit zwei parallelen Hierarchien tun.¹⁸



Auch ohne CONCUR wäre es eine zu starke Vereinfachung zu glauben, dass SGML (oder

18 Das hierauf folgende Bild zeigt die ersten Zeilen von Ibsens Versdrama *Peer Gynt*, mit sowohl die dramaturgische Struktur (Akt, Szene, Rede) wie auch die Versstruktur (Akt, Szene, Verszeile). Wie Akt und Szene erscheinen die Bühnenanweisungen in beiden Strukturen.

XML) die Struktur nur unter Verwendung von Verschachtelungselementen definiert. Beide Spezifikationen definieren einen Mechanismus zum Zuweisen von Bezeichnern (IDs) zu Elementen und zum Verweisen auf diese Bezeichner von anderer Stelle. (HTML-Hypertext-Anker sind vielleicht die am weitesten verbreitete Verwendung dieser Idee.) Die grundlegende Datenstruktur, die am besten geeignet ist, um über SGML- und XML-Dokumente zu argumentieren, ist nicht der Baum, sondern der gerichtete Graph. Die Elementstruktur bildet eine oft nützliche Unterstruktur dieses Graphen.¹⁹



Wenn SGML auf der Annahme basiert hätte, dass ein Text von Natur aus einen hierarchischen Satz von Inhaltsobjekten umfasst, dann wäre es unmöglich, die Existenz des CONCUR-Features und der ID / IDREF-Links in SGML zu erklären.

Es sollte auch darauf hingewiesen werden, dass die Autoren des OHCO-Papiers zwar von Text als einer geordneten Hierarchie von Objekten sprechen, ihre Schlussbemerkungen jedoch die Beobachtung enthalten, dass »viele Dokumente mehr als eine nützliche Struktur haben«. Da sie diese Beobachtung offensichtlich nicht als Untergrabung ihrer früheren Behauptung ansehen, dass »Text

19 Das Bild zeigt dieselben Zeilen aus der Kanzone Walters als das vorhergehende, und berücksichtigt auch Verweise, die mittels ID/IDREF bewerkstelligt werden.

eine geordnete Hierarchie von Inhaltsobjekten« ist, dürfen wir sicher annehmen, dass sie mit »einer« geordneten Hierarchie »mindestens eine« geordnete Hierarchie andeuteten, nicht »höchstens eine« oder »genau eine« geordnete Hierarchie.

Während SGML und XML sowohl für die Autoren des OHCO-Papiers als auch für die Kritiker dieses Papiers im Mittelpunkt des Interesses stehen, weist das Modell des Textes als geordnete Objekthierarchie eine Reihe weiterer Instanzierungen auf, die hier zumindest im Vorbeigehen erwähnt werden sollten.

Das in den 1960er Jahren von Douglas Engelbart bei Xerox PARC entwickelte Softwaresystem (NLS für »oNLine System«, später »Augment«) strukturierte Dokumente anhand einer generischen Drei-Ebenen-Hierarchie: Im Unterschied zu SGML benannte Engelbarts System keine Knoten in der Baum mit einem Typnamen (wie »Kapitel«, »Kapitelüberschrift« oder »Absatz«), und Knoten konnten nicht beliebig tief geschachtelt werden. Aber es implementierte ein Modell von Text als eine geordnete Hierarchie von anonymen Objekten. Eines der ersten weitverbreiteten PC-basierten Konkordanzsysteme, Word Cruncher, verwendete auch eine dreistufige Organisation für Texte. Dies erleichterte es Word Cruncher, mit Texten wie Shakespeare-Stücken umzugehen, wo Passagen normalerweise durch Handlung, Szene und Zeilennummer identifiziert werden, oder mit der Bibel, wo Passagen durch Buch, Kapitel und Vers identifiziert werden.

Aber vielleicht ist die Instanzierung dieses Modells, das diesem Publikum am vertrautesten ist, TUSTEP selbst: Die Organisation einer TUSTEP-Datei in nummerierte Seiten, Sätze, und Sätze mit Unterscheidungsnummer ist ein gutes Beispiel für das Modell. Und wie TUSTEP mit Seiten- und Satznummer arbeiten kann, zeigt, wie ein leistungsfähigeres Textmodell es ermöglicht, leistungsfähigere Software zu schreiben.

OHCO-Variante

Es ist vielleicht nicht uninteressant, jeden Teil der OHCO-Hypothese zu betrachten und zu fragen, was aus der OHCO-These würde, wenn man diesen Teil modifizieren würde.

Nichtgeordnete Hierarchie?

Was wäre, wenn wir eine ungeordnete Hierarchie von Inhaltsobjekten annehmen würden? Die Antwort ist überraschend klar: Das Ergebnis ist ein Datenbankverwaltungssystem, das ein hierarchisches Modell unterstützt. Die objektorientierten Datenbanken der 1980er und 1990er Jahre kommen mir in den Sinn. Das Modell ist interessant, aber es ist ziemlich eindeutig kein Textmodell. Seit mindestens 40 Jahren haben Datenbanken mit dem Aufstieg des relationalen Datenbankmodells explizit die Reihenfolge der Datensätze in einer Tabelle und die Reihenfolge der Spalten in einer Zeile außer Acht gelassen.

Aber Texte ignorieren die Reihenfolge nicht: Wir können die Kapitel eines Romans nicht neu ordnen, ohne den Roman zu verändern. Wir können die Worte eines Gedichts nicht neu ordnen, ohne das Gedicht zu zerstören. »Heide auf Röslein Röslein rot Röslein der Röslein« ist nicht die erste Zeile eines Goethe-Gedichtes.

Es gibt einige Romane mit mehreren Lesesequenzen, wie die *Rayuela* (»Himmel-und-Hölle« oder »Hickelkasten«) des argentinischen Modernisten Julio Cortázar, die das Prinzip der Mehrfachzuordnung von Kapiteln verkörpern: wenn die Handlungen in einer anderen Reihenfolge erzählt werden, ergeben sie eine andere Geschichte.

Geordnete Hierarchie von ... ?

Was wäre, wenn wir annehmen, dass der Text nicht aus Inhaltsobjekten besteht? Wäre der Text anders? Woraus würde ein Text bestehen?

Es ist schwierig, diese Frage zu beantworten, weil die Autoren keine Definition des Begriffs »Inhaltsobjekt« anbieten, und man weiß nicht mit Sicherheit, was sie mit diesem Satz meinen.

Vielleicht meinen sie »Elemente der logischen Struktur« des Textes, im Gegensatz zu den Elementen der Layoutstruktur (also: Absätze, nicht Seiten, für das typische Prosadokument). Dies würde der Rhetorik der meisten Befürworter von SGML in den 80er Jahren entsprechen.

Wenn Text keine Hierarchie von Inhaltsobjekten wäre, könnte es sich im Gegensatz dazu um eine Hierarchie von Seitenlayout- oder Renderingobjekten handeln. Dies ist, glaube ich, das Modell von Text, das in PDF oder in den geräteunabhän-

gigen Dateien vom TeX-System implizit enthalten ist und möglicherweise in den internen Datenstrukturen von Software für diese Formate sichtbar wäre.

Es ist explizit das Modell der XSL-Formatierungsobjekte-Spezifikation des World Wide Web Consortiums. Die XSL FO-Spezifikation definiert eine Menge von XML-Elementen, die Regionen auf einer Seite entsprechen, die in einer bestimmten Weise mit formatiertem Text gefüllt werden sollen.

In den letzten Jahrzehnten hat es mehrere Wissenschaftler gegeben, die argumentiert haben, dass man literarische Texte nur in Hinblick auf ihre ursprüngliche physische Präsentation interpretieren darf bzw. kann. In der Erstausgabe von William Butler Yeats findet man vielleicht eine Bedeutung in der Eigenart der Seitengestaltung oder in der Papierwahl. Man argumentiert energisch, dass die Gedichte von William Blake nicht von der Schrift und den Zeichnungen der Stiche Blakes getrennt werden dürfen. Diese Schule interessiert sich sozusagen für die Materialität des Textes. Das Textmodell als Hierarchie von Layoutobjekten – quasi als Seitenfolge – wird diejenigen ansprechen, die sich für Materialität in diesem Sinne einsetzen.

Als Mediävist interessiert mich auch die Materialität des Textes: Ein gutes Verständnis der materiellen Bedingungen der Textübertragung ist sowohl in der Textkritik als auch im Verständnis der Entstehung und des Konsums von Literatur hilfreich. Aber der Mediävist wird zögern, Text nur als eine typographische (oder handschriftliche!) Struktur zu identifizieren. Wenn wir keine Autographen haben, wenn wir stattdessen vierunddreißig Manuskripte mit verschiedenen Seitenlayouts haben. Welche dieser vierunddreißig Handschriften wird »der Text« sein? Ist das Nibelungenlied ein literarisches Werk, oder vierunddreißig? Die Materialität ist für den Mediävisten wie für andere von großem Interesse. Aber gerade weil es fast nie ein einziges, maßgebliches Zeugnis für ein antikes oder mittelalterliches Werk gibt (sie können singular, aber selten maßgeblich sein), finden es Mediävisten – und Altphilologen nützlich, systematisch zwischen Text (einem abstrakten Objekt) und Textträgern (physisch) zu unterscheiden. Es ist hilfreich, Seiten und andere materielle Objekte zusätzlich zu den abstrakteren und rein textuellen

Objekten wie Cantos, Strophen oder Absätzen kodieren zu können.

Mit »Inhaltsobjekten« meinen DeRose et al. vielleicht lediglich »die Gegenstände, die im Inhalt vorkommen«, oder »die Gegenstände, die den Inhalt des Textes tragen oder verwirklichen«. Das würde, so scheint es, offen lassen, welche Objekte und welche Arten von Objekten das sind. Dies wäre konsistent mit der Tatsache, dass in SGML und XML der Benutzer (oder der Designer des SGML / XML-Vokabulars), nicht die Spezifikation oder die Software, bestimmt, welche Arten von Objekten aufgezeichnet werden, so wie sie im Text vorkommen. Wenn wir an der Materialität unseres Exemplars interessiert sind, müssen wir nur seine Inhaltsobjekte als materiell definieren. Aus diesem Grund kann XML für die Beschreibung von Seitenlayouts oder anderen spezialisierten Arten von Strukturen verwendet werden, und wird tatsächlich so verwendet.

Eine Nichthierarchie?

Was wäre, wenn der Text nicht in einer hierarchischen Struktur angeordnet wäre, nicht in einem Baum? Was ist, wenn es eine andere Form der Organisation hat? Welche anderen Organisationsformen sind plausible Kandidaten?

Der Linguist Igor Mel'čuk beobachtet in einer Diskussion über die Modellierung der menschlichen Sprache, dass hierarchische Strukturen – das heißt Bäume – eine interessante Variante zwischen völlig unbeschränkten gerichteten Graphen und den am engsten eingeschränkten gerichteten Graphen darstellen.²⁰

Eine [semantische Repräsentation] ist ein (fast) willkürliches Netzwerk, ein Graph, dem praktisch keine formalen Beschränkungen auferlegt sind. Im Gegensatz dazu ist eine [phonetische Darstellung] offensichtlich eine Kette von phonetischen Symbolen, d. h. eine Kette, die einfachste aller möglichen Graphen, mit dem Maximum an Beschränkungen, die ihr auferlegt sind. Um eine Viele-zu-Viele-Abbildung zwischen beliebigen Netzwerken zu etablieren und Ketten wird eine zweckmäßige Brücke benötigt, d. h. ein Diagramm, das formell

20 Igor Mel'čuk, *Dependency syntax: Theory and practice* ([Albany]: State University of New York Press, 1988), S. 48.

auf halbem Wege zwischen beliebigen Netzwerken einerseits und Ketten andererseits liegt. Dies ist ein Baum, eine formale Entität, die traditionell verwendet wird, um Satzstrukturen darzustellen ...

Wenn wir über Alternativen zu Baumstrukturen für unser Textmodell nachdenken wollen, können wir unsere Aufmerksamkeit zunächst auf die Familie von Modellen lenken, die strengere Einschränkungen auferlegen, indem wir Text als Sequenz modellieren (Kette, in Mel'čuks Terminologie).

GEORDNETE FOLGE VON INHALTSOBJEKTEN?

Wir haben schon die Vorstellung von Text als eine Sequenz kennengelernt.

Einige Linguisten argumentieren, dass die Behandlung von gesprochenen Äußerungen als eindimensionale Folge von Phonemen eine übermäßige Vereinfachung ist: unter anderem fehlen Pausen, Tonhöhenänderungen und Geschwindigkeitsänderungen. (Im herkömmlichen Schreiben liefert die Interpunktion teilweise Informationen über diese Dinge, aber nicht annähernd genug, um die Prosodie einer Äußerung vollständig darzustellen. Natürlich vorkommende Schriften sind nur so eindeutig, wie sie sein müssen, damit geschriebene Nachrichten zuverlässig entschlüsselt werden können, und wollen typischerweise keine vollständige Aufzeichnung einer Äußerung überhaupt anbieten.) Gebärdensprachen werden unterdessen nicht mit einem einzigen Mund gesprochen, sondern mit zwei Händen, einem Körper und einem Gesicht.

Eine Vielzahl von Sequenzmodellen wurde definiert oder implementiert; Sie modellieren alle Text als »eine Sequenz von ... X«, aber sie stellen unterschiedliche Werte von X bereit.

Das einfachste Modell dieser Klasse beschreibt Text als eine Folge von Zeichen. Dieses Modell ist das Kernstück von Texteditoren wie emacs oder vi auf Unix-Systemen und von »Programmiereditoren« verschiedener Art auf anderen Betriebssystemen.

Es ist auch in den String-Datentypen unserer Programmiersprachen sichtbar, in Standard-E-Mail (ich meine E-Mail ohne HTML-Markup) und in vielen Unix-Werkzeuge (wo es mit dem alternativen Textmodell als eine Abfolge von Zeilen konkurriert).

Vielleicht stellt das älteste Modell historisch gesehen Text als eine Folge von Lochkarten dar, mit so vielen Wörtern auf jeder Karte wie möglich, oder für Verse, mit einer Verszeile pro Karte.²¹

```
BO101011SIGNORI E CAVALLIER CHE VE ADUNATI
BO101012PER ODIR COSE DILETTOSE E NOVE,
BO101013STATI ATTENTI E QUI&ETI, ET ASCOLTATI
BO101014LA BELLA ISTORIA CHE 'L MIO CANTO MUOVE;
...
```

Im Pfeffer-Corpus des modernen gesprochenen Deutschen zeigte jede Karte den Gebrauch eines Wortes im Zusammenhang mit einem Satz; Wenn der Satz mehr als 80 Zeichen lang war, wurden Auslassungen verwendet, damit der Satz auf die Karte passte.

Auch als Lochkarten nicht mehr gebräuchlich waren, blieb das Textmodell als Zeilenfolge erhalten, wenn auch nur in den Datenstrukturen der Texteditoren, die Textdateien als Zeilenzeigerarrays darstellen. Es ist auch in vielen Unix-Werkzeugen wie grep sichtbar, das nach Zeichenketten in einer Datei sucht und alle Zeilen zurückgibt, die dem gegebenen Muster entsprechen.

In den frühen 1960er Jahren begann man, Programme zu schreiben, die Texte für das Drucken formatierten – dem TUSTEP FORMATIERE-Kommando sehr ähnlich. Viele solcher Programme behandelten ihre Eingabe als eine Abfolge von Zeilen, die das Programm in zwei Gruppen unterteilt hat: Datenzeilen, mit dem zu druckenden Text, und Steuerzeilen, die die Behandlung des Textes vorschrieben. In den Programmen Runoff, roff, nroff, troff und Script, von denen Versionen für im wesentlichen jeden Mainframe-Computer verfügbar waren, wurden Formatierungsanweisungen durch einen Punkt in der ersten Spalte der Zeile identifiziert (nach der Theorie, dass in natürlicher Sprache, keine Wörter mit einem Punkt beginnen; das ist für Englisch nicht ganz richtig, wie sich herausstellt, aber der Benutzer kann leicht vermeiden, eine Eingabezeile mit einem solchen Wort zu beginnen).

Das jüngste der Batch-Formatierungsprogramme, Donald Knuths TeX, behält die Aufteilung des Textes in Daten und Befehle bei, macht jedoch Befehle auf andere Weise erkennbar, wo-

²¹ Diese Art der Kodierung war weit verbreitet, aber das hier gegebene Beispiel ist ein erfundenes.

durch die strukturelle Bedeutung von Zeilenumbrüchen in der Eingabe eliminiert wird. Sein Modell ist (etwas vereinfachend), dass Text eine Folge von Zeichen ist, durchsetzt mit Formatierungsbeehlen.

Eine interessante Variante des Karten- oder Linienmodells wurde schon früh von Roberto Busa entwickelt. In ihr ist jede Karte einem einzelnen Wort gewidmet, das in den ersten Spalten der Karte (in seiner Oberflächenform) aufgezeichnet wird. Spätere Spalten der Karte, in einer Art tabellarischem Format, liefern weitere Informationen über das Wort: vielleicht sein Lemma oder seine Nachschlageform, vielleicht seine grammatischen Merkmale, vielleicht seine syntaktische Funktion oder die Wortnummer des Wortes, von dem es abhängt (Adjektive zeigen auf ihre Substantive, Subjekte und Objekte zu ihrem Verb, etc.). Da der herkömmliche Text vertikal am Anfang der Zeilen gelesen werden kann, wird dieses Format manchmal als »vertikaler Text« bezeichnet,²² aber weil die zusätzliche Information auf jeder Karte eine zweite Dimension orthogonal zu der ersten bietet (was Busa den »internen Hypertext« des Wortes genannt hat), könnte man es vielleicht besser als zweidimensionaler Text bezeichnen. Wie hier gezeigt kann der Text auch waagrecht, statt senkrecht gestaltet werden.²³

1 **Mejit Axun:**
 Barāt ajniñ on ikkinçi kächeſi »Barāt« boſurtar.
 bara:tʰ ajniñ on ikkʰinʃi kʰeʃesi bara:tʰ boſurtar

baraat	ay-ning	onikki-nchi	keché-si	baraat	bol-ur-lar
Npr	N-GEN	NU-ORD	N-POSS3	Npr	Vi-AOR-3p2
Baraat.month	month-GEN	twelve-ORD	night-POSS3	Baraat.month	be-AOR-3p2

On the night of the 12th day of the Barāt month, the Barāt festival begins.
 In der Nacht des 12. im Monat Barāt findet das »Barāt« statt.

Heutzutage ist das weitaus verbreitetste sequenzielle Textmodell dasjenige, das die meisten Menschen für die Erstellung und Überarbeitung von Dokumenten verwenden, das Modell von Word, WordPerfect, usw. In diesem Modell wird Text als Abfolge von Absätzen (oder absatzartigen Objekten) betrachtet, und Absätze wiederum als Abfolgen von Zeichen. In diesem Modell können ver-

22 S. z. B. John R. Abercrombie, *Computer programs for literary analysis* (Philadelphia: Univ. of Pennsylvania Press, 1984), S. 53 u. 86.

23 Das Beispiel stammt aus dem »Uyghur Light Verbs Project«, (geführt von Arianne M. Dwyer); die Daten des Projekts sind im Web (<http://uyghur.ittc.ku.edu>) frei zugänglich.

schiedene Absätze unterschiedlich formatiert sein (d. h. unterschiedliche Stilanweisungen tragen): Sie können verschiedene Ränder haben, verschiedene Arten von Ausrichtungen (linksbündig, rechtsbündig, zentriert), verschiedene Durchschüsse und verschiedene Standardschriftarten. Innerhalb eines Absatzes können verschiedene Zeichen (oder Zeichenfolgen) einer bestimmten Schriftart, Schriftgröße oder Behandlung zugewiesen werden.

Obwohl das Stilmenü in einem typischen Textverarbeitungsprogramm Einträge wie Kapiteltitle, Abschnitttitle, Unterabschnitttitle usw. enthalten kann, verstehen Word usw. nicht, dass Kapitel Abschnitte enthalten oder dass Abschnitte Unterabschnitte enthalten. Sie können also keine Regel erzwingen, dass (z. B.) ein Kapitel nicht direkt einen Unterabschnitt enthalten sollte. Tatsächlich weiss das typische Textverarbeitungsprogramm überhaupt nichts über Kapitel, Abschnitte und Unterabschnitte, nur über Kapiteltitle, Abschnittsüberschriften und Unterabschnitttitle.

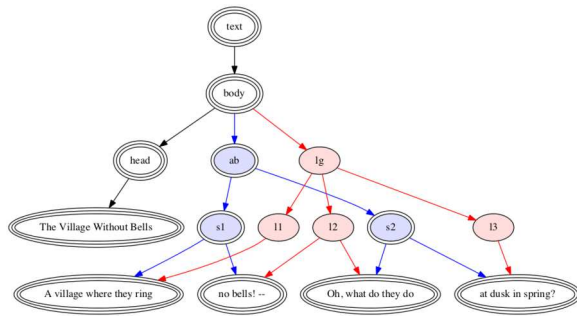
Aus wissenschaftlicher Sicht oder zu Modellierungszwecken ist dies vielleicht der größte Nachteil aller dieser flachen Sequenzmodelle: In diesen Modellen kann man nicht davon ausgehen, dass der erste Absatz des ersten Kapitels eines Romans Bestandteil jenes Kapitels ist: Das Kapitel ist kein Element in der Abfolge von Absätzen, die den Text bildet, und das Kapitel erscheint daher nicht im Modell. Wir können das Kapitel natürlich als Abfolge von Absätzen definieren, aber das bedeutet, das Modell zu erweitern.

MEHRERE HIERARCHIEN VON INHALTSOBJEKTEN?

Wenn wir weniger eingeschränkte Strukturen betrachten, kommen zwei Möglichkeiten leicht in den Sinn.

Erstens: Statt einer einzelnen Hierarchie von Inhaltsobjekten können wir mehrere haben: zwei, drei oder beliebig viele. Das folgende Bild zeigt ein Haiku als einfaches Beispiel: ein Haiku mit drei Zeilen und zwei Sätzen. Blaue Knoten markieren die linguistische Struktur, Rosa Knoten die Versstruktur; Weiße Knoten werden geteilt und treten in beiden Strukturen auf.

Wie bereits erwähnt, definiert SGML bereits mehrere Hierarchien als eine Möglichkeit, und obwohl die CONCUR-Funktion selten implementiert



wurde und als unnötige Komplikation aus XML entfernt wurde, ist es nicht konzeptionell schwierig, XML so zu erweitern, dass dasselbe unterstützt wird. Oliver Schonefeld und Andreas Witt haben vor einigen Jahren einen XCONCUR vorgeschlagen.²⁴ Kürzlich hat mindestens ein Projekt berichtet, dass eine reine XML-Notation verwendet wird, um mehrere Baumstrukturen in ihren Dokumenten aufzuzeichnen: Die »dominante« Struktur verwendet herkömmliche XML-Elemente und die »rezessiven« Strukturen verwenden leere Meilensteinelemente.²⁵ Für verschiedene Arten von Arbeit sind verschiedene dominante Strukturen geeignet, so dass das Projekt bei Bedarf automatisch Dokumente von einer XML-Struktur in die andere umwandelt. Dadurch kann das konzeptionelle Modell von CONCUR in einem reinen XML-Kontext angewendet werden, so dass Standard-XML-Tools zur Verarbeitung verwendet werden können.

GERICHTETER GRAPH VON INHALTSOBJEKTEN?

Zweitens können wir zulassen, dass die Textstruktur nicht ein Baum ist und nicht mehrere Bäume,

24 Oliver Schonefeld und Andreas Witt, Towards validation of concurrent markup (Proceedings of Extreme Markup Languages, Montréal, 2006; auch Oliver Schonefeld, XCONCUR and XCONCUR-CL: A constraint-based approach for the validation of concurrent markup, *Data Structures for Linguistic Resources and Applications: Proceedings of the Biennial GLDV Conference 2007*, hsg. Georg Rehm, Andreas Witt, und Lothar Lemnitzer (Tübingen: Gunter Narr, 2007).

25 Darüber habe ich berichtet in: Representing concurrent document structures using Trojan Horse markup, *Proceedings of Balisage: The Markup Conference 2018. Balisage Series on Markup Technologies*, vol. 21 (2018).

sondern ein Graph oder ein gerichteter Graph, der kein Baum sein muss. Vor einigen Jahren schlugen Claus Huitfeldt und ich eine gerichtete Graphenstruktur für Text unter dem Namen Goddag (generalized ordered-descendant directed acyclic graph, »verallgemeinerter gerichteter azyklischer Graph mit geordneten Nachkommen«, auch norwegisch für »Hallo«) vor;²⁶ das Modell war nützlich für Gedankenexperimente, aber wir hatten weder die Ressourcen noch den Wunsch, für Goddags die gleiche Infrastruktur aufzubauen, die bereits für XML existiert: Parser, Validatoren, Transformationssprachen und Transformationsmaschinen, Datenbanken und Datenbankabfragesprachen, und so weiter.

Ein Team am Huygens-Institut in den Niederlanden unternimmt nun ein so ehrgeiziges Projekt, um sein eigenes neues Textmodell unter dem Namen »Text as Graph« zu unterstützen. In TAG wird Text nicht als einzelner Baum, sondern als Hypergraph modelliert, aus dem verschiedene Bäume mit unterschiedlichen Strukturen projiziert werden können.²⁷

EINE PRAGMATISCHE LÖSUNG?

Ein wesentlicher Nachteil von uneingeschränkten Graphen ist, dass sie ziemlich schwierig effizient zu verarbeiten sind. Mehrere gleichzeitige Bäume sind einfacher zu verarbeiten, aber immer noch komplexer als die Behandlung eines einzelnen Baumes. Selbst wenn wir die zugrundeliegende Realität unserer Texte als ungezwungene Netzwerke betrachten, oder als Äquivalent, können wir

26 C. M. Sperberg-McQueen und Claus Huitfeldt, GODDAG: A data structure for overlapping hierarchies, in *DDEP-PODDP 2000*, hsg. P. King und E. V. Munson [= LNCS 2023] (Berlin: Springer, 2004), S. 139–160.

27 Ronald Haentjens-Dekker und David J. Birnbaum, It's more than just overlap: Text As Graph, Presented at Balisage: The Markup Conference 2017, Washington, DC, August 1–4, 2017. In *Proceedings of Balisage: The Markup Conference 2017*. Balisage Series on Markup Technologies, vol. 19 (2017). Ronald Haentjens Dekker u.a., TAGML, A markup language of many dimensions, Presented at Balisage: The Markup Conference 2018, Washington, DC, July 31 – August 3, 2018. In *Proceedings of Balisage: The Markup Conference 2018*. Balisage Series on Markup Technologies, vol. 21 (2018).

Bäume als Werkzeuge für die Arbeit mit Text attraktiv finden: Sie ermöglichen uns, einen Teil der Realität in einer einfacheren Form darzustellen, und sie bieten eine bequeme Möglichkeit, zwischen einer unbeschränkten Graphenstruktur für Text und der linearen Reihenfolge von Bytes zu vermitteln, die wir benötigen, um ein Dokument in einer Datei unter Verwendung eines Dateisystems zu speichern, das Dateien als flache Sequenzen von Bytes modelliert.

In diesem Zusammenhang sollte ich TUSTEPs Verwendung eines hierarchischen Modells für die Verwaltung und Manipulation von Text erwähnen, verbunden mit der Fähigkeit, ein Dokument von einer Hierarchie in eine andere Hierarchie zu transformieren (unter Beibehaltung von Spuren der ursprünglichen Struktur) das Dokument in die andere Richtung). Dies ermöglicht eine bemerkenswerte Flexibilität bei der Verwaltung von Dokumenten: Es gibt immer eine hierarchische Struktur von »Seiten« und »Zeilen«, aber Markup kann auch im Dokument bereitgestellt werden und verwendet werden, um das Dokument unter Benutzerkontrolle in eine andere Struktur umzuwandeln. Für einen erfahrenen Benutzer bietet Tusstep daher die Möglichkeit, entweder mit einem Textmodell als Baum oder als Wald mit mehreren gleichzeitigen Bäumen oder als Graph zu arbeiten. Die Macht, die den Benutzern gegeben wird, ist meiner Meinung nach ein Grund für den Erfolg von TUSTEP.

V.

Wir sehen, der Text ist nicht so einfach, wie man es denken könnte. Wenn man genau auf die Seitengestaltung schaut, sieht man, dass unser Textmodell viel komplexer sein muss, als eine eindimensionale Abfolge von Wörtern oder Zeichen, wenn wir dem Kulturobject Text gerecht werden wollen. Wenn man den Definitionsversuch von DeRose u. a. etwas genauer betrachtet, sieht man dass auch dieser Ansatz eher zu einfach als zu komplex ist.

Dabei haben wir nicht einmal angefangen, von der sprachlichen Natur des Text zu reden, und davon, wie man die Modelle der Sprachwissenschaftler in ein integriertes Textmodell einbauen könnte. Es liegt in der Natur von Text, dass er nicht nur die Strukturen aufweist, die wir gerade

besprochen haben, sondern auch eine phonetische Struktur, eine morphologische, eine syntaktische, und eine semantische. Es wäre sicher eine lohnende Aufgabe, alle diese Strukturen auf einen gemeinsamen Nenner zu bringen, aber es wäre wohl eine sehr harte Prüfung Ihrer Geduld, das hier und jetzt zu versuchen.

VI.

Stattdessen möchte ich meine Betrachtungen mit der Beobachtung zu Ende führen, dass fast alle die Strukturen, die wir in Text finden, entweder explizit oder ansatzweise schon in TUSTEP präsent sind, und dass fast alle Anforderungen, die die wissenschaftliche Arbeit mit Text uns und unserer Software stellt, in TUSTEP schon Spuren hinterlassen haben.

Vor einigen Jahren pflegte der prominente Digital Humanist Willard McCarty (Herausgeber des Diskussionsforum *Humanist*) humorvoll zu sagen, dass Software oft nicht nur das Denken, sondern auch die Persönlichkeit ihrer Hersteller und Programmierer widerspiegeln. In dem Maße, in dem Software ein Werkzeug ist, spiegelt sie die Überzeugungen seiner Entscheidungsträger wider, welche Aufgaben sich lohnen. Nicht nur das: Software spiegelt auch notwendigerweise die Überzeugungen ihrer Schöpfer darüber wider, wie man diese Aufgaben ausführen kann (oder sollte) oder (etwas weniger direkt) über den Menschen, die die Software benutzen werden: was sie wichtig finden, was sie weniger wichtig finden, wie sie über ihre Aufgabe denken und wie viel Verantwortung sie dafür übernehmen wollen.

Es gibt nicht viele Software-Programme, deren implizitem Benutzerbild ich *würdig sein* möchte. TUSTEP ist ein solches Programm. Der fleissige, kreative, wissenschaftliche Benutzer, dem TUSTEP zugeschnitten ist, der möchte ich sein. Der möchten vielleicht alle Wissenschaftler sein.

Um mit diesem Gedanke fortzufahren: jede Textverarbeitungssoftware verkörpert eine Vorstellung von Text, von dem, was man mit Text machen kann, von den Gestaltungsmöglichkeiten von Text, wie man den Text auf der Seite gestalten kann. Bei mancher Software sind die verkörperten Ideen einfach, leichtfertig, sogar simpel. Nicht viele Programme bieten ein Konzept von Text und von der Gestaltung von Text so detailliert, so tief

empfunden, so zusammenhängend durchdacht als das von TUSTEP. Wie immer jedoch reflektieren Programme die Gedanken derer, die die Software geschaffen haben, und es scheint fair zu sagen, dass sich Wilhelm Ott mit TUSTEP als einer der großen Denker unseres Tages in Betreff auf Text, Textgestalt, und Textgestaltung erwiesen hat.

Bitte danken Sie mit mir Wilhelm Ott für seine Arbeit.