

"Where the Digerati Meet the Literati" - Einige Gedanken zur Anwendbarkeit von computerspezifischen Konzepten auf das Schreiben literarischer Texte

Autor, Leser und Text

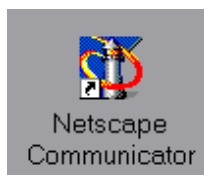
Gleich zu Beginn von *Writing Space*, seinem Grundlagentext zu elektronischer Textualität, formuliert Jay David Bolter eine der zentralen Thesen der Hypertext-Theorie:

"Electronic writing [...] tends to reduce the distance between author and reader by turning the reader into an author. [...] It is changing the relationship of the author to the text and of both author and text to the reader." (Bolter, 2/3)

Die Frage nach dem Verhältnis von Autor, Leser und Text ist aber weder neu noch an die Beschäftigung mit digitaler Literatur gebunden. Schon in der klassischen Moderne begann der Autor, sich aus seinem Werk zurückzuziehen, "invisible, refined out of existence, indifferent, paring his fingernails" - um mit James Joyce zu sprechen. In der Postmoderne - und diesen Begriff muß ich etwas undifferenziert stehen lassen - wird der Tod des Autors ausgerufen, wird seine Bedeutung für Text und Leser hinterfragt, wird seine Person zur Funktion. In den Blickpunkt rücken statt dessen der aktive Leser und der Text, der ein aktives Lesen nicht nur zuläßt, sondern fordert. Nachzulesen ist das bei Roland Barthes, Jacques Derrida, Michel Foucault, auch bei Wolfgang Iser - um hier nur einige zu nennen. Diese Frage nach der Stellung von Autor und Leser bietet sich als Ansatz für die Interpretation digitaler Literatur ganz besonders an, da es sich hier um interaktive Texte handelt, also um Texte, die zumindest vorgeben, daß der formende Einfluß des Autors gegenüber der kreativen Einflußnahme des Lesers zurücktritt.

Sprachmaschine Computer

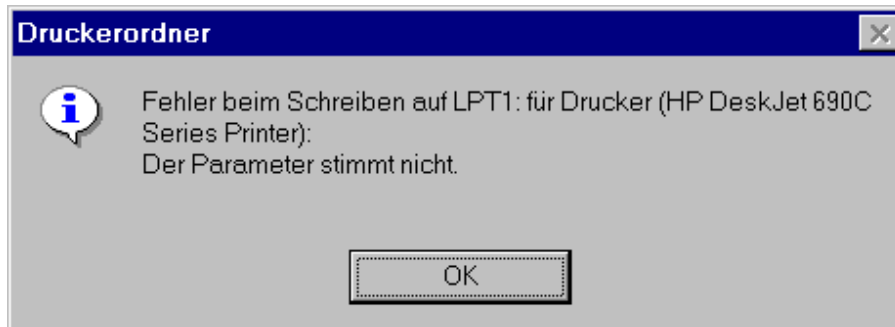
Die Verbindung von Computer und Literatur nun mag nicht ganz naheliegend sein, ist der Computer doch eine relativ prosaische Maschine, die höchstens in der Werbung oder im Film zu metaphorischem Glanz kommt. In der täglichen Anwendung wird er durchaus nicht zu unrecht "number cruncher" und "Rechenknecht" genannt. Wenn wir aber genauer hinschauen, entpuppt sich der Rechenknecht als Symbolmaschine. Geht man von einer Definition von Sprache als System von lautlichen oder visuellen Symbolen aus, die auf eine Realität außerhalb ihrer selbst verweisen (oder eben auch nicht), so erscheint der Computer schnell in einem neuen Licht: nicht als Rechner, sondern sozusagen als Sprecher (oder zumindest als Schreiber). Unser gesamter Umgang mit dem Computer ist sprachlich strukturiert: Wir haben die graphische Oberfläche, wie das MacOS oder Windows 98, darunter liegt beim PC die textbasierte DOS-Oberfläche, wieder darunter, für den Normal-Benutzer jetzt nicht mehr sichtbar, liegt die Ebene des Binärcodes, der Einsen und Nullen, die letztendlich auf Speicheradressen auf Festplatten, Disketten oder im Hauptspeicher verweisen (also auf durchaus reale Entitäten).



Wenn ich zum Beispiel dieses Symbol auf der graphischen Oberfläche meines Computers anklicke, rufe ich damit meinen Internet-Browser auf, der auf der Festplatte gespeichert ist. Das Symbol verweist über Zwischenschritte auf 1/0-Werte auf einem Datenträger. Konfiguriert man den Browser entsprechend, verweist das Icon nicht nur auf ein Programm, sondern mittelbar auf eine Adresse im Internet, zum Beispiel die Homepage von Mark Amerikas Internet-Literaturprojekt AltX. Bitte klicken Sie. ... Sollten Sie gerade AltX einen kurzen Besuch abgestattet haben, werden sie nun wissen, daß ich den Titel dieses Artikels (zumindest den ersten Teil) bei Mark Amerika entlehnt habe - unter Vorspiegelung falscher Tatsachen, muß ich gestehen, denn während Mark Amerika Internetliteratur macht, behandle ich hier

ausschließlich Offline-Literatur, also nicht-vernetzten Einzelanwendungen.

Man kann also das Klicken, aber auch die Texteingabe in Befehlszeilen (wie in DOS) als sprachliche Anweisung an den Computer verstehen, der diesen Befehl dann parst, also auf seine 'grammatische' Form und seinen Inhalt hin untersucht und dann entsprechend reagiert. Der Computer benutzt natürlich keinen Natursprachen-Parser und was ich hier als 'Grammatik' bezeichne, wird im Kontext von Programmiersprachen eigentlich 'Syntax' genannt. Stimmt diese nicht mit der des jeweiligen Computers, der auf ihm aktuell laufenden Programme oder seiner Netzwerkumgebung überein, reagiert der Computer mit einer Fehlermeldung, wie zum Beispiel wenn ich versuche, die Homepage der Universität Mainz aufzurufen, indem ich <http://www.uni-mainz.de> eingebe. Der Computer hat den Befehl aufgrund formaler Fehler nicht verstanden. Korrekt hieße der Befehl zum Aufsuchen und Anzeigen dieser 'Seite' im Internet nämlich <http://www.uni-mainz.de>.



Aber
auch bei

inhaltlichen Nichtübereinstimmungen mit der Realität des Computers (z.B. wenn der Befehl an eine nicht angeschlossene Hardware geht) 'antwortet' der Computer mit einer Fehlermeldung. Stimmt der Befehl aber inhaltlich und formal mit der spezifischen Realität überein (das hieße in diesem Falle, daß der Drucker eingeschaltet ist) führt der Computer ihn gewöhnlich auch aus.

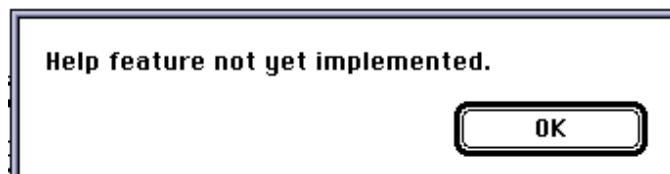
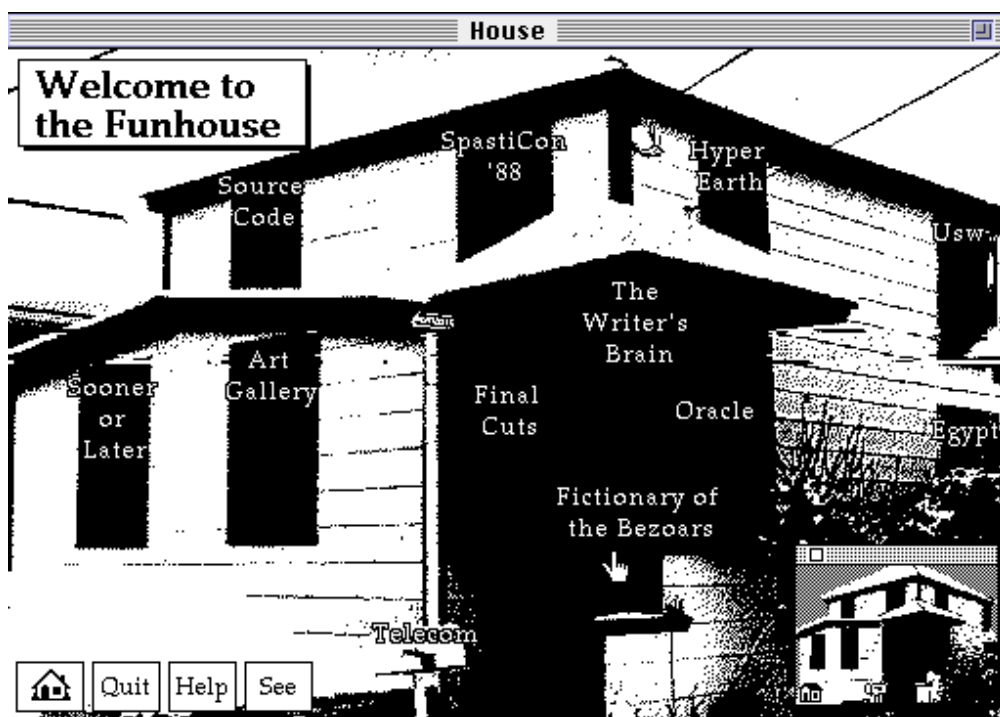
Das heißt, der Computer reagiert auf eine sprachliche Äußerung mit einer Handlung, die durchaus auch sprachlichen Charakter haben kann - wie eben die Ausgabe einer Fehlermeldung, was dann so aussieht, als würde der Computer mit dem Anwender 'reden' oder ihm zumindest eine Nachricht schreiben. Aber auch das Darstellen einer Programmoberfläche, das auf das Anklicken des Programm-Icons oder des Programm-Namens z.B. in der Liste der zuletzt benutzten Programme hin erfolgt, kann als sprachliche Reaktion verstanden werden, denn diese graphische Oberfläche ist wieder nur ein symbolischer Verweis auf die dem Programm zugrundeliegenden Dateien, die wieder auf Speicherstellen verweisen. Unter diesem Blickwinkel ist es eigentlich nicht mehr verwunderlich, daß der Computer für die Produktion von Literatur, also für kreatives Schaffen mit Sprache nicht nur geeignet ist, sondern geradezu dazu aufruft.

Ebenen-Verschiebung

Ein Charakteristikum der Sprachmaschine Computer ist die Schachtelung der oben erwähnten Ebenen - von der Benutzeroberfläche bis zu den On/Off-Zuständen in den Mikroprozessoren. Im folgenden möchte ich einige Methoden aufzeigen, wie mit dem Strukturkonzept der Ebenen und Ebenenverschiebung der Autor digitaler Texte ver- und entortet wird. Einerseits können nicht alle Anwender, also Autoren und Leser, gleichermaßen auf alle Ebenen zugreifen, andererseits dringen aus den 'Tiefen' des Computers aber auch Meldungen zum Benutzer der obersten Schnittstelle. Es findet also eine Überlagerung der sichtbaren Ebenen statt. Am einfachsten läßt sich das anhand der allgegenwärtigen Fehlermeldungen zeigen: Fehlermeldungen wie diejenigen, die ich eben demonstriert habe, kommen eindeutig vom laufenden Programm; andere Fehlermeldungen kommen wohl eher vom Betriebssystem selbst, wie die berühmte Macintosh-'Bombe', die ich hier lieber nicht provoziere - sie signalisiert schließlich einen vollständigen Systemabsturz. Tatsächlich (und natürlich) 'kommen' alle diese Fehlermeldungen ursprünglich von den Programmieren der einzelnen Programme, also von deren Autoren.

Error

Bei digitaler Literatur gibt es nun drei Textebenen: den 'eigentlichen' Text, dessen Zustände heute schon bei Printliteratur nicht mehr als direkte Äußerungen des Autors gelesen werden; den *Reader* mit Standardmeldungen und solchen, die der Autor für die Bedürfnisse seines Textes selbst programmieren kann; und die Betriebsumgebung mit Standardmeldungen und -zuständen, die vom Autor gewöhnlich nicht zu beeinflussen sind. Der Hyperfiction-Autor John McDaid arbeitet in seiner HyperCard-Fiction *Uncle Buddy's Phantom Funhouse* unter anderem mit Fehlermeldungen, um die Rolle des Autors im Text zu thematisieren. In *Uncle Buddy* entsteht aus unterschiedlichen, zum Teil auf dem und für den Computer konzipierten Dokumenten, aber auch aus Photokopien und Musik-Kassetten das Bild einer Epoche - die späten 70er und frühen 80er Jahre, die Zeit der ersten Personalcomputer und Computernetzwerke - aber auch die Geschichte eines Mannes, des Musikers Arthur 'Buddy' Newkirk und die Geschichte der Liebe zwischen Arthur und Emily Keane. Einer der zentralen Aspekte von *Uncle Buddy* ist dabei die Frage nach Autorschaft und Repräsentation - wer schreibt die Geschichte, wer interpretiert die Songtexte, wie sehen wir Emily und durch wessen Augen? Das geschieht zum Teil inhaltlich, zum Teil perspektivisch, zum Beispiel über die Häuser (neben dem 'Funhouse' gibt es auch noch das 'Haunted House'), zum Teil aber auch mit Hilfe von Fehlermeldungen und Dialogfeldern.



Die erste Fehlermeldung taucht schon im empfohlenen Einstiegsdokument auf. Klickt man in "Read Me First" auf 'Help on Using the

Macintosh', erscheint diese Meldung, die zunächst aussieht, wie ein Standard-Dialog. Andererseits ist es unwahrscheinlich, daß auch *Uncle Buddy* (wie so viel kommerzielle Software heute) nur als Beta-Version ausgeliefert wurde und die fehlende Hilfefunktion ein noch nicht behobener Bug, also ein veritabler Fehler ist. An einer anderen Stelle erscheint auf den oberflächlich vorgesehenen Druckbefehl hin die Fehlermeldung: "Don't even think of it!" Auch dies könnte eins der für den Macintosh üblichen umgangssprachlichen Dialogfelder sein. Allerdings entsteht hier ein Bruch zwischen der Programmrealität und der Form des Dialogs: Wenn an dieser Stelle die Druckoption zumindest potentiell vorgesehen (wenn auch nicht realisiert) ist, paßt der ethisch-verbotende Tonfall des "wie kannst Du an sowas denken?" nicht. Noch deutlicher als bei "Help function not yet implemented" scheint es sich hier um einen Kommentar des Autors zu handeln, der aus der Leseumgebung in den Text eindringt.

Egypt



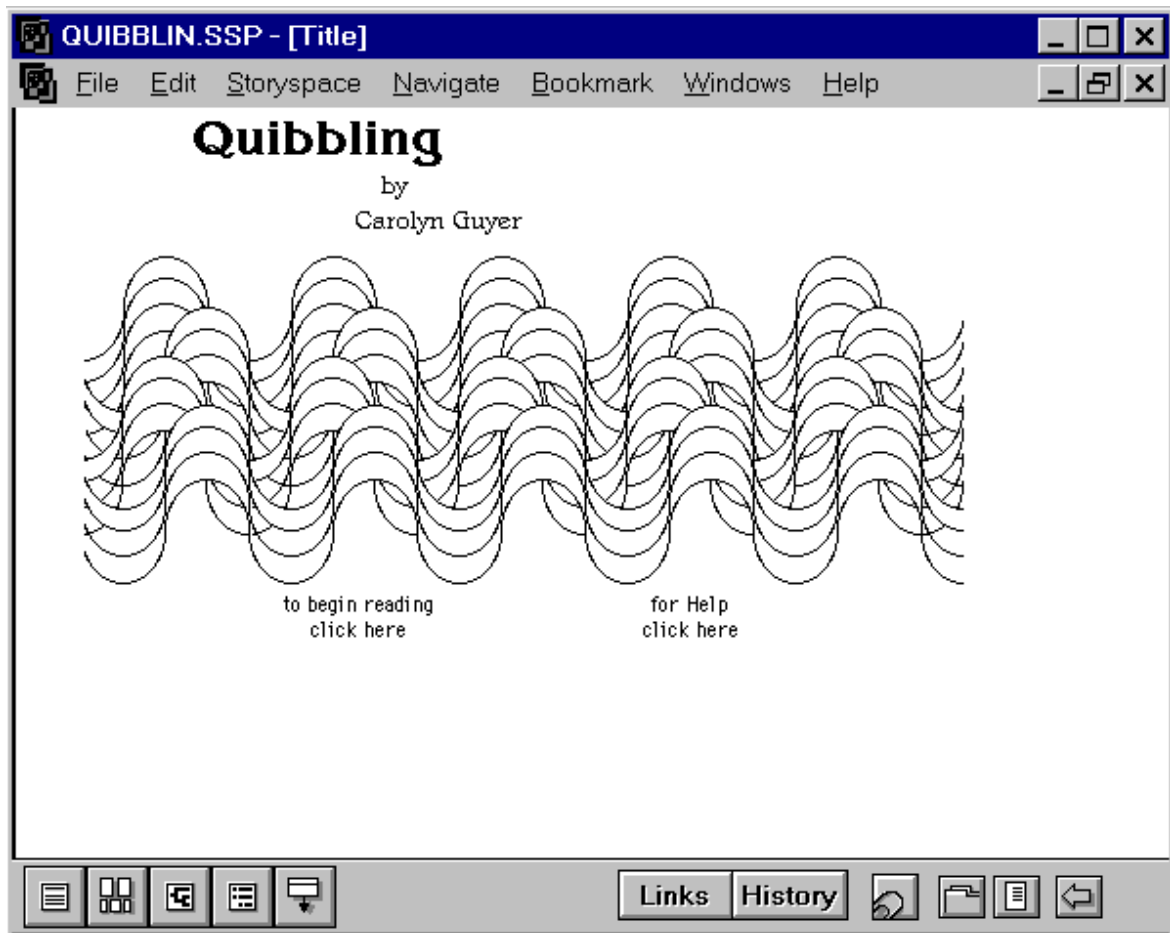
Es gibt da eine noch deutlichere Stelle, an die der Leser erst nach längerem Lesen gelangt. Ich habe ca. ein Jahr gebraucht, um die Paßwörter im Text zu finden, die mich auf die dritte Ebene bringen. In diesem linearen Pfad angekommen, habe ich schon den größten Teil des Funhouse gelesen. Am Ende dieses Pfades rufe ich eine unvermeidbare Fehlermeldung auf: Der Tod - diesmal nicht des Autors, sondern des Lesers. Offensichtlich lehnt hier zumindest der Autor den aktiven Leser ab. Oder ist es der Text (oder das Programm?), der entschieden hat, daß der Leser diese Offenbarungen des Autors nun doch nicht sehen sollte? Der Text von *Uncle Buddy* bietet keine eindeutige Lösung für diese Frage an. Durch die Autorität der Dialogfelder, die normalerweise nicht Teil des Textes sind, scheint der Autor in den Text zurückgeführt zu werden. Durch die Diskrepanz zwischen Aussage und Programmzustand wird der Autorenkommentar jedoch wieder retextualisiert.

Nun ist diese Verschiebung des Autors schon von der Papier-Literatur her bekannt: der Autor ist dabei nie mit der Erzählstimme gleichzusetzen, weder mit dem allwissenden noch mit dem Ich-Erzähler. Es gibt eine Tradition der Zuweisung, in der immer andere Personen als Autor identifiziert werden, sei es, um Gesellschafts- oder Kirchenkritik ohne Gefährdung der eigenen Person üben zu können, sei es wegen sexuell expliziten Inhalts oder als bloßes Verwirrspiel. Gerade die Postmoderne spielt gerne mit dem Auftauchen und Verschwinden eines vermeintlichen Autors im Text und der Verschachtelung der Erzählsituation bis zu dem Punkt, an dem selbst der Erzähler nicht mehr zu bestimmen ist. Der Autor des digitalen Textes geht jedoch einen Schritt weiter, indem er die Rolle des Computers als möglichem Textgenerator und vor allem die Rolle des Lesers als designedem Co-Autor mitreflektieren kann.

Help

Eine weitere Möglichkeit, dem Autor Eingang in den Text zu verschaffen, ist die Hilfefunktion. Gewöhnlich gehört die Hilfe zu den Meta-Funktionen eines Programms, die weder seine Funktionalität noch das Erscheinungsbild des bearbeiteten Dokuments beeinflusst.

Ein digitaler Text mit Leseumgebung kann zum Beispiel so aussehen wie hier in Carolyn Guyers Quibbling:



Die Leseumgebung bietet als Menü- und Werkzeugleisten die notwendige Tools zum Lesen eines Hypertextes an: einen Navigationspfeil, Buttons für die Map-Views, ein Hilfe-Menü 'Links'- und 'History'-Buttons und so fort.

Es scheint also, als würden Storyspace-Hyperfictions die Standard-Ebenen-Schachtelung, wie sie von 'normalen' Programmen wie z.B. einer Textverarbeitung bekannt ist, beibehalten. Diese Ebenen, die den Text einrahmen, könnte man dann lesen wie Gérard Genette das für die Paratexte eines gedruckten Buches beschrieben hat - mit den Ver- und Entortungen des Autors, die dort möglich sind und seit der Erfindung des Buchdrucks auch dort praktiziert werden.

Interessanterweise findet sich die Online-Hilfe hier nicht nur unter den Meta-Buttons, sondern noch einmal auf der Start-Seite. Von dort gelangt man zu einem ausführlicheren Hilfe-Text, der aber die Informationen aus der Menü-Hilfe noch einmal aufnimmt. Vom ersten Hilfe-Screen kommt man nicht nur weiter in den 'eigentlichen' Text oder zu mehr Hilfe, sondern auch zur Bibliographie, zu den Danksagungen und zur Widmung. Als Paratexte eines Drucktextes würden diese Informationen 'horizontal' um den Haupttext geschachtelt: Widmung und Danksagung davor, die Bibliographie am Ende, wobei eine Bibliographie zumindest nach Genette nicht zu den Paratexten gehört, sondern Teil des wissenschaftlichen Textes ist. Für ein Computerprogramm wäre es eher üblich, Äquivalente (zum Beispiel die Programminfo) bei den Metabefehlen ganz links oder ganz rechts in der Menüleiste einzuordnen. Nur auf der Titelseite sieht es so aus, als hätte sich Guyer mehr oder minder an die Print-Konventionen gehalten, aus der Strukturkarte geht aber hervor, daß "help" etwa vier Ebenen tief in den 'Haupt'Text eingeschachtelt ist. Auch in Michael Joyces Hyperfiction-Klassiker *Afternoon* wird die Hilfe aus dem Startscreen erreicht. Joyce geht aber noch einen Schritt weiter und läßt Textone der Hilfe später im Text wieder auftauchen.

Es scheint unproblematisch, in der Hilfe-Funktion die Stimme des Autors zu lesen. Immerhin

ist Hilfe kein klassischer Paratext wie zum Beispiel das Vorwort, bei dem der Leser gewohnt ist, die textuellen Techniken des Haupttextes zu erwarten. Der textexterne Status der technischen Leseanweisungen etabliert zuerst einmal einen Autor-Programmierer in sicherer Entfernung vom Text. Taucht die Hilfe (oder Teile von ihr) aber plötzlich im Text wieder auf, so wird der Autor quasi durch die Hintertür in den Text zurückgeschmuggelt.

Der eingeschachtelte Autor

Offensichtlich hat digitale Literatur - oder zumindest Hyperfiction - die Tendenz, mit Hilfe der auf dem Computer möglichen Struktur-Konzepte traditionelle Auftritte und Situierungen des Autors sowohl zu inszenieren als auch mit ihnen zu brechen. Charakteristischerweise lassen sich dabei die Schachtelungen von Autor-Stimme, Erzählstimme und Text in digitaler Literatur nicht mehr auflösen. Dadurch kann, wie in *Quibbling* oder *Afternoon* die Stellung des Autors im Text gefestigt statt zurückgenommen werden. Damit stehen diese Texte im Gegensatz nicht nur zu aktuellen literarischen Praktiken, sondern auch zum Tenor der Hyperfiction-Theorie. Andere Lösungen, wie z.B. John McDaid's *Funhouse* zeigen aber, daß auch weniger affirmative Reflektionen der Autorrolle auf dem digitalen Medium möglich sind. Der Computer ist durchaus in der Lage, Ansätze der zeitgenössischen Literaturtheorie und -praxis auf intellektuell wie ästhetisch ansprechende Art und Weise zu verhandeln.

Überarbeitete Fassung eines Vortrags auf dem 4. Stipendiatentreffen der Graduiertenförderung des Landes Rheinland-Pfalz an der Johannes Gutenberg-Universität, Mainz, am 6.7.1998.

©Anja Rau
<http://www.uni-mainz.de/~raua000>

The copyright for the Netscape Communicator-Icon used here is probably held by Netscape; the error message presumably belongs to either Microsoft or Hewlett Packard. Should you be the holder of these copyrights, please mail me if you want the picture removed or your ownership acknowledged. Images 3-5 are from *Uncle Buddy's Phantom Funhouse* by John McDaid, published by Eastgate Systems. (c) 1992. Image 6 is from *Quibbling* by Carolyn Guyer, published by Eastgate Systems. (c) 1992.

Barthes, Roland. "The Death of the Author." *Image-Music-Text*. Hrsg., Übers. Stephen Heath. New York: Noonday Press, 1977: 142-148.

Bolter, Jay David. *Writing Space: The Computer, Hypertext and the History of Writing*. Hillsdale: Erlbaum, 1991.

Genette, Gérard. *Paratexte: Das Buch vom Beiwerk des Buches*. Frankfurt a.M.: Campus, 1989.

Guyer, Carolyn. *Quibbling*. Watertown: Eastgate Systems, 1992. (Diskette)

Joyce, James. *A Portrait of the Artist as a Young Man*. New York: Penguin, 1977, 215. Erste Ausgabe 1916.

Joyce, Michael. *Afternoon - A Story*. Watertown: Eastgate Systems, 1997. (Diskette).

McDaid, John. *Uncle Buddy's Phantom Funhouse*. Watertown: Eastgate Systems, 1992. (Disketten).