

Schriftkundliche Methoden

Objektyp: **Group**

Zeitschrift: **Geschichte und Informatik = Histoire et informatique**

Band (Jahr): **5-6 (1994-1995)**

PDF erstellt am: **12.07.2024**

Nutzungsbedingungen

Die ETH-Bibliothek ist Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Inhalten der Zeitschriften. Die Rechte liegen in der Regel bei den Herausgebern.

Die auf der Plattform e-periodica veröffentlichten Dokumente stehen für nicht-kommerzielle Zwecke in Lehre und Forschung sowie für die private Nutzung frei zur Verfügung. Einzelne Dateien oder Ausdrucke aus diesem Angebot können zusammen mit diesen Nutzungsbedingungen und den korrekten Herkunftsbezeichnungen weitergegeben werden.

Das Veröffentlichen von Bildern in Print- und Online-Publikationen ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Die systematische Speicherung von Teilen des elektronischen Angebots auf anderen Servern bedarf ebenfalls des schriftlichen Einverständnisses der Rechteinhaber.

Haftungsausschluss

Alle Angaben erfolgen ohne Gewähr für Vollständigkeit oder Richtigkeit. Es wird keine Haftung übernommen für Schäden durch die Verwendung von Informationen aus diesem Online-Angebot oder durch das Fehlen von Informationen. Dies gilt auch für Inhalte Dritter, die über dieses Angebot zugänglich sind.

Möglichkeiten der Transkription mittelhochdeutscher und frühneuhochdeutscher Texte auf dem Computer

Thomas Fährdrich

Im Zusammenhang mit Vorbereitungsarbeiten zur Untersuchung eines größeren Korpus von spätmittelalterlichen Texten stellte sich das Problem, wie diese auf einem Computersystem möglichst textnahe dargestellt werden können. Da die Texte nach der Transkription sprachlich untersucht werden sollten, war die Zielvorgabe, möglichst nahe am handschriftlichen Text zu bleiben, d.h. Diphthonge, Umlaute, spezielle Schreibungen einzelner Zeichen, Häkchen, Akzente u.ä. möglichst so wiederzugeben wie in der Handschrift. Die Transkription sollte also möglichst wenig Interpretationen des Transkribenten enthalten. Die Untersuchung des Sprachsystems sollte danach in einem zweiten Arbeitsschritt geleistet werden.

Ziel war, dass die Spezialzeichen auf dem Bildschirm des Computers dargestellt und auf einem gängigen Computerdrucker ausgegeben werden können. Zur Verfügung standen ein Macintosh-Computer und ein postscriptfähiger Laserdrucker.

Es zeigte sich bald, dass man nicht auf eine bestehende Lösung zurückgreifen konnte. Es ergaben sich schliesslich zwei Möglichkeiten. Eine erfordert das Textverarbeitungsprogramm «Microsoft® Word 4» oder «Word 5» für Macintosh, die andere die Anschaffung des Programms «Fontographer®», mit dessen Hilfe sich jedes gewünschte Zeichen entwerfen lässt.

Meine eigene Beschäftigung mit der Wiedergabe von Spezialschriftzeichen auf dem Computer fand in den Jahren 1989 und 1990 statt. Unterdessen sind die erwähnten Programme weiterentwickelt worden. Gemäss Herstellerangaben erlaubt die neueste Version von «Fontographer®» auch die Herstellung von Zeichensätzen im «TrueType™»-Format. Diese Schriftart erfordert einen weniger aufwendigen Drucker und ermöglicht es trotzdem, jedes selbstentworfene Zeichen in jeder Grösse ohne Qualitätsverlust darzustellen.

Beide der nachfolgend erwähnten Möglichkeiten sollen auch auf Computern mit dem «MS-DOS»-Betriebssystem und der Benutzeroberfläche «Windows» realisierbar sein.

Die Herstellung eigener Zeichensätze mit dem Programm «Fontographer»

Mit Hilfe des Programms «Fontographer» lassen sich komplette, selbstgestaltete Zeichensätze herstellen, die auf dem Computer wie die Schriftarten «Times», «Helvetica» oder andere angewählt werden können.

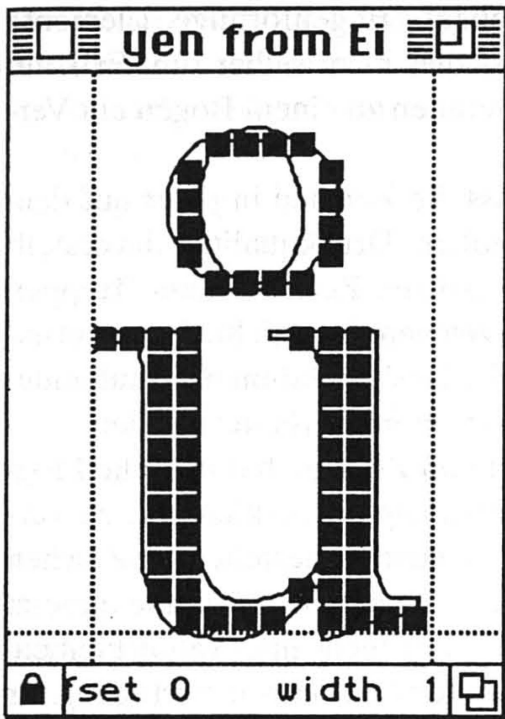


Abb. 1: Bitmap-Zeichen

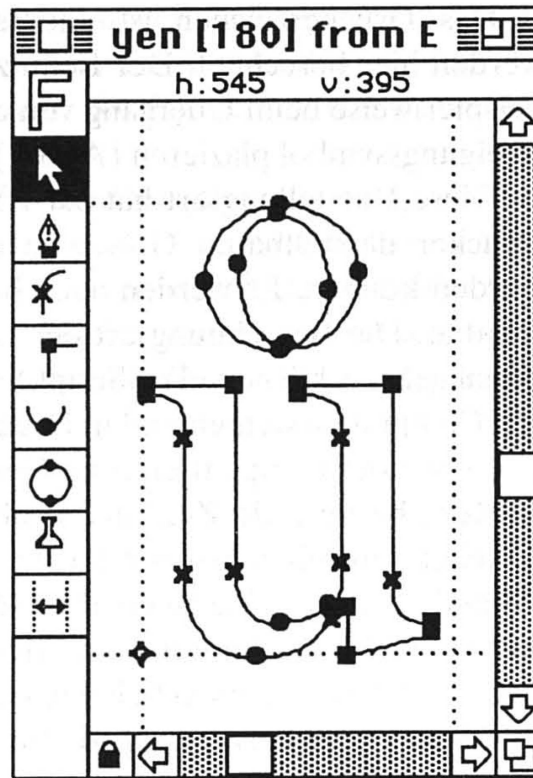


Abb. 2: PostScript-Zeichen

Im folgenden erkläre ich die Möglichkeiten dieses Programmes anhand des selbstentworfenen Zeichensatzes. Er basiert auf der Schriftart «Times». Der Macintosh-Zeichensatz enthält 255 Zeichen. Die ersten 31 Zeichen sind Steuerzeichen, die nicht für die Darstellung von Schriftzeichen zur Verfügung stehen. Auf alle anderen Plätze kann im Grunde genommen jedes beliebige Zeichen gesetzt werden. Dies kann man feststellen, wenn man beispielsweise die Tastaturbelegung der Schriftarten «Times» und «Symbol» miteinander vergleicht.

In der Praxis ist es aber ratsam, nur die höheren Nummern des Zeichensatzes mit Sonderzeichen zu belegen, um die gebräuchlichen, normalen Schriftzeichen im gleichen Zeichensatz auf den gewohnten Tasten zur Verfügung zu haben. Sonst müsste man beim Gebrauch der Schrift häufig von einem Zeichensatz in den anderen wechseln.

Jedes Zeichen besteht aus zwei Elementen, einem sogenannten «Bitmap-Zeichen» für die Darstellung auf dem Bildschirm und einem «PostScript-Zeichen», das die Daten für den Drucker enthält. Das «Bitmap-Zeichen» ist aus Punkten zusammengesetzt. Da die Anzahl Punkte, die für ein einzelnes Zeichen zur Verfügung stehen, beschränkt ist, werden bei dieser Darstellungsart bogenförmige Elemente eines Schriftzeichens als Treppen sichtbar (Abb. 1).

Das Druckerzeichen ist anders aufgebaut. Bogenförmige Elemente werden hier berechnet. Der Benutzer muss dies nicht selber tun. Er kann beispielsweise beim Übergang von einer Geraden zu einem Bogen ein Verzweigungssymbol plazieren (Abb. 2).

Diese Darstellungsart hat zur Folge, dass die Zeichen in jeder auf dem Drucker darstellbaren Grösse in einer hohen Druckqualität dargestellt werden können. Es werden auch bei sehr grossen Zeichen keine Treppen sichtbar. Die Berechnungsart der Druckerzeichen entspricht dem international gebräuchlichen «PostScript-Standard». Deshalb können sie auf anderen Computersystemen und in Druckereien weiterverarbeitet werden.

Neben der komplett eigenen Gestaltung von Zeichen hat man die Möglichkeit, bestehende Zeichen zu übernehmen, sie zu verkleinern, zu verschieben, zu drehen. So kann der Diphthong «û» aus dem bestehenden Zeichen «u» und dem verkleinerten und verschobenen Zeichen «o» zusammengesetzt werden (Abb. 2). Eine weitere Möglichkeit darf nicht unerwähnt bleiben. Wie gezeigt, stehen nicht beliebig viele Plätze für Zeichen zur Verfügung. Es ist möglich, Zeichen so zu plazieren, dass sie nachträglich überlagert werden können. So genügt beispielsweise ein Häkchen, das dann unter jedes beliebige Zeichen gesetzt werden kann.

Zum Schluss ist noch zu sagen, dass ein selbstgestalteter Zeichensatz in verschiedenen Programmen wie Textverarbeitung, Datenbank u.ä. verwendet werden kann.

Als Beispiel sind im Anhang die Sonderzeichen des selbstgestalteten Zeichensatzes abgedruckt.

Die Darstellung von Sonderzeichen im Textverarbeitungsprogramm «Microsoft Word»

Neben der komfortableren, dementsprechend aber auch aufwendigeren und teureren Möglichkeit mit dem Programm «Fontographer» hat sich auch eine einfache Variante ergeben, um viele Sonderzeichen darstellen zu können. Man benötigt dazu das Textverarbeitungsprogramm «Microsoft Word» in der Version 4 oder 5. Die Zeichen können auf jedem beliebigen Drucker in der gewohnten Druckqualität wiedergegeben werden.

Im Programm «Word» ist es möglich, mathematische Formeln darzustellen. Dabei kann man Zeichen zurückversetzen, überlagern, hochstellen und tiefstellen. Diese Möglichkeiten können auch für die Darstellung von Diphthongen und Umlauten nutzbar gemacht werden.

Die Formeldarstellung in «Microsoft Word» basiert darauf, dass man mittels eines speziellen Formel-Schriftsatzbefehls Überlagerungen u.ä. definieren kann. Wählt man im Textdokument den Modus «Einblenden», in dem

alle Steuerzeichen auf dem Bildschirm sichtbar sind, erscheint der Formel-Schriftsatzbefehl, wählt man hingegen den Modus «Ausblenden», in dem die Steuerzeichen nicht sichtbar sind, erscheint die Formel so, wie sie ausgedruckt wird.

Ich gebe im folgenden ein Beispiel für die Darstellung des Diphthongs «ü». Die einfachste Möglichkeit besteht darin, dass man die Zeichen «u» und «o» überlagert, d.h. dass sie übereinander geschrieben werden. Das Zeichen «o» wird mit dem Befehl «Format-Zeichen» hochgestellt und verkleinert. Die Formelschreibweise sieht folgendermassen aus: $\small\circ(u,^\circ)$. Die Syntax für die Formel-Schriftsatzbefehle ist im Benutzerhandbuch zu «Microsoft Word» ausführlich beschrieben.

Es hat sich als praktisch erwiesen, Formel-Schriftsatzbefehle als Textbausteine zu speichern. So sind diese bequem verfügbar.

In dieser Darstellungsart können kleinere Probleme auftreten. Wie gezeigt, besteht der Formel-Schriftsatzbefehl aus mehreren Zeichen, die beim Ausblenden der Steuerzeichen nicht alle sichtbar sind. Wenn man den Cursor hinter ein Zeichen setzt, das aus einer Formel besteht, kann es vorkommen, dass der Cursor quasi in die Formel hineingerät und diese auseinanderreißt. In einem solchen Fall empfiehlt es sich, die Steuerzeichen einzublenden, um dann die richtige Ordnung wieder herzustellen.

Zudem können kleine Ungenauigkeiten auftreten, indem das hochgestellte Zeichen nicht genau in der Mitte sitzt. Dies tritt vor allem dann auf, wenn man kursiv schreibt.

Insgesamt ist die Möglichkeit der Formel-Schriftsatzbefehle aber durchaus eine Alternative zur oben beschriebenen Möglichkeit.

Anhang 1:

Hinweise zu den verwendeten Computerprogrammen

- Fontographer, von ALTSYS Corp., Vertrieb für die Schweiz durch SWIP Handels AG, Glattalstrasse 501, 8153 Rümlang
- Word, von Microsoft Corp., überall im Handel erhältlich

Anhang 2:

Verfügbare Zeichen im selbstentworfenen Zeichensatz mit «Fontographer»

Verfügbare Zeichen:

-mittelhochdeutsch / frühneuhochdeutsch:

a:	æ	ā		
i:	ī	î		
o:	œ	ō	ö	õ
u:	û	ū	ü	
v:	ŵ	ŵ		
s:	ʃ	ʒ	ß	
Zahlen:	ʀ (mhd. 4)	ʒ (mhd. 5)	ʌ (mhd. 7)	
Masse:	£ (Pfund)	ð (Pfennig)		
Sonstige:	ϕ (Kapitel)	? (Fragezeichen)		

-althochdeutsch:

Ḅ	ḏ	g	ḡ	ḑ
---	---	---	---	---

-Schweizerdeutsche Dialektschrift nach Dieth:

İ ì
 (alle anderen Zeichen können mit den fliegenden Akzenten geschrieben werden)

-Schweizerdeutscher Sprachatlas (SDS):

˘ (Klammer) ˘ (Klammer)
 (alle anderen Zeichen können mit den fliegenden Akzenten geschrieben werden)

-fliegende Akzente:

Vokale oben:	˘	˙	˚	˛	˜	˝
Punkte oben:	˘	˙	˚	˛	˜	˝
diverse oben:	˘	˙	˚	˛	˜	˝
Striche oben:	˘	˙	˚	˛	˜	˝
diverse unten:	˘	˙	˚	˛	˜	˝

(jeweils ein Akzent für Minuskeln und einer für Majuskeln)

Textbeispiele:

-Jahrzeitbuch St. Michael, Zug:

Hans Wölfflin, genant Howa Henflin, und Elß inn Oetal, fin wirtÿ, unn aller ir kinden und vordern: durch der aller sel willen ist gefetz 3 f kernem, also zeteilen, plebano 1 f, dem frümefter 1/2 ð, dem pründer zu Sant Mcihhael ouch 1/2 f, also daz die egeanten priester uff finen jerlichen tag am abent fingent ein vigily ...

-Phonogramme des Schweizerdeutschen Sprachatlas, Eschenbach (LU):

I verzöw èch dò gäärn öppis über mi Pruef. S Poschthawteramt isch z Eschebach eso en Aart Familietradition ...

Lector – ein Unterrichtsprogramm für historische Schriftkunde

Gerold Ritter

Im Laufe ihrer Forschungen stehen viele HistorikerInnen und interessierte Laien vor dem Problem, sich mit dem Inhalt eines nicht edierten mittelalterlichen Quellentextes befassen zu müssen, diesen aber nicht einmal entziffern zu können. Das Lesen von mittelalterlichen Schriften kann nur durch Übung erlernt werden; selbst Fachleute müssen sich in schwierigere Schriften immer wieder neu einlesen. Durch etwas Training lässt sich aber ein Grundstock an Lesefertigkeit aufbauen, mit dessen Hilfe sich Historikerinnen und Historiker mit einer neuen Schrift wesentlich schneller vertraut machen können.

Konventionell werden solche Leseübungen mit Lehrbüchern oder kopierten Unterlagen durchgeführt. Die Übungen erfolgen am besten zu zweit: Die eine Person liest im Original, die andere kontrolliert in der Transkription. Ein Unterrichtsprogramm soll nun ermöglichen, sich in das Thema Transkription selbständig und so breit und tief wie gewünscht einzuarbeiten.

Unterrichtsprogramm oder Lehrbuch?

Das wichtigste Kriterium für den Einsatz eines Unterrichtsprogramms ist die Frage, ob die gleiche Aufgabe mit einem Buch nicht gleich gut oder sogar besser zu lösen wäre¹.

Diese Abwägung zwischen Programm und Buch ist besonders bei einem geplanten Einsatz in einem geisteswissenschaftlichen Fach wie Geschichte sehr wichtig. Ein gewisser Goodwill für die Verwendung eines Computers, wie er in den Naturwissenschaften normalerweise zu finden ist, kann hier nicht a priori vorausgesetzt werden. Obwohl sich die elektronische Textverarbeitung auch in den Geisteswissenschaften weitgehend durchgesetzt hat, stehen viele WissenschaftlerInnen dieser Disziplinen einem weitergehenden Computereinsatz kritisch gegenüber. Für diese ablehnende Haltung können neben gewissen eher irrationalen Ängsten durchaus auch rationale Argumente angeführt werden:

- Die grossen Textmengen, die für die wissenschaftliche Arbeit normalerweise zu lesen sind, lassen sich in einem Buch besser und augenfreundlicher darstellen als mit der heutigen Monitortechnik.

¹ Vgl. Nielsen, Hypertext S. 143–162, v.a. S. 152. Persönlich bin ich der Meinung, dass bei gleichwertigen Alternativen das konventionelle Buch vorzuziehen ist. Ein Unterrichtsprogramm muss im Vergleich zu einem Lehrbuch wesentliche Vorteile bieten, damit sich sein Einsatz rechtfertigen lässt.

- Ein Buch ist unabhängig. Es braucht kein teures und (heute noch) meist grosses und schweres «Lesegerät» und kann deshalb nicht nur am Computer, sondern jederzeit und überall gelesen werden.
- Da die nötige Einrichtung zum Lesen einer elektronischen Publikation vielfach gar nicht zur Verfügung steht bzw. die Benutzung an technischen Inkompatibilitäten scheitert, vermag die eigene Arbeit als Buch ein wesentlich grösseres (potentielles) Zielpublikum anzusprechen.
- Auch wenn dabei in den letzten Jahren grosse Fortschritte erzielt wurden, vermag die heutige Computertechnik die «normalen» AnwenderInnen doch noch nicht vollständig von der zugrunde liegenden Technik abzuschirmen, was die BenutzerInnen dazu zwingt, sich mit technischen Einzelheiten aus einem ihnen oft völlig fremden Sachgebiet auseinanderzusetzen.

Das Unterrichtsprogramm *Lector*

Mit den speziellen Anforderungen vor Augen, die ein Unterrichtsprogramm für den Einsatz in einer solchen dem Computer relativ kritisch gegenüberstehenden Umgebung stellt, wurde am Historischen Seminar der Universität Zürich das Unterrichtsprogramm *Lector* entwickelt.

Hardwaremässig wird für den Einsatz von *Lector* ein Apple-Macintosh Computer benötigt. Während an die Rechenleistung keine besonderen Ansprüche gestellt werden, ist eine bestimmte Mindestgrösse des Bildschirms für ein sinnvolles Arbeiten unabdingbar. Deshalb benötigt *Lector* einen 12-Zoll-Monitor (640 × 480 Pixel) bzw. einen Powerbook-Bildschirm (640 × 400 Pixel).

Softwareseitig basiert *Lector* auf HyperCard, einem Hypermedia-Werkzeug für Macintosh-Computer. HyperCard hat sich für eine solche stark graphikbasierte Anwendung sehr bewährt und kann durch die integrierte Scriptingsprache HyperTalk auch die hohen Ansprüche an flexible Ablaufsteuerung und Programmierbarkeit erfüllen².

Kernstück von *Lector* ist das Kapitel «Transkriptionsübungen» (vgl. Abbildung). Das Programm erlaubt es, eingescannte Ausschnitte aus verschiedenartigen Dokumenten interaktiv am Bildschirm zu transkribieren. Dabei kann man das bisher Transkribierte korrigieren und falsche oder ausgelassene Wörter markieren lassen. Diese Fehler können in einem weiteren Transkriptionsdurchgang ausgebessert und erneut überprüft werden.

Zusätzlich stehen verschiedene Hilfsmittel zur Verfügung. Eine Lupe zeigt Ausschnitte des eingescannten Originals in zweifacher Vergrösserung;

² Für weitere Informationen zu HyperCard vgl. z.B. Gilbert, HyperCard.

zur Verfügung. Transkriptionsregeln klären Zweifelsfälle, ein Verzeichnis von üblichen Abkürzungen und deren Auflösung sowie ein Kapitel über Zahlen und Masse helfen bei diesen oft kritischen Stellen einer Transkription, während das Kapitel «Schriftgeschichte», mittels Querverweisen eng verknüpft mit einer Sammlung von Beispielalphabeten, allgemeinere Informationen zur Entstehung der verschiedenen Schrifttypen und -stile gibt. Ein Glossar für verwendete Fachbegriffe und ein Verzeichnis weiterführender Literatur – ebenfalls aus den übrigen Kapiteln mit Links referenziert – runden das Programm *Lector* ab.

Welche Vorteile bietet nun das Unterrichtsprogramm *Lector* im Vergleich zu einem konventionellen Lehrbuch?

- Das Selbststudium wird wesentlich besser unterstützt. Anwenderinnen und Anwender können sich – innerhalb der recht weitreichenden Möglichkeiten von *Lector* – selbständig und gemäss den eigenen Bedürfnissen so breit und so tief wie gewünscht in das Problemfeld einarbeiten, ohne einem bestimmten Lehr- oder Stundenplan folgen zu müssen. Zwar sind auch der in der Einleitung beschriebenen konventionellen Methode des Lesens zu zweit gewisse soziale Vorteile nicht abzuspüren, fördert sie doch die Zusammenarbeit der Studierenden untereinander. Aber es ist eben nicht immer möglich, die Transkriptionsübungen zu zweit durchzuführen. In dieser Situation kann der Computer bis zu einem gewissen Grad die zweite Person ersetzen.
- Die Korrekturfunktion ermöglicht es, jederzeit die bisher erarbeitete Transkription zu korrigieren und die falsch transkribierten oder fehlenden Wörter anzuzeigen. Dies ermöglicht eine interaktive Zusammenarbeit zwischen dem/der BenutzerIn und dem Computer.
- Während die Tips zur Transkription, direkt aus der Übung heraus abrufbar, eine ganz neue Funktionalitätsstufe bilden, die in den entsprechenden Lehrbüchern nicht zu finden ist, bildet auch die selektive Lösungsabfrage einen wesentlichen Fortschritt gegenüber der «Alles-oder-nichts»-Methode des Lehrbuches, zeigt sie doch lediglich das gesuchte Wort und nicht – ungewollt – den ganzen Satz.
- Neben den eigentlichen Übungen stehen ausführliche Zusatz- und Hintergrundinformationen zur Verfügung, die ihrerseits untereinander verbunden sind, auf die aber beispielsweise auch in den Tips mit (automatisierten) Querverweisen hingewiesen wird. Diese hypertextartige Verknüpfung ermöglicht es, für ein konkretes Problem sehr selektiv auf relevante Informationen zuzugreifen.

Erfahrungen

Seit dem Wintersemester 1991/92 wird in den jährlich durchgeführten Seminaren «Erfassung und Auswertung ländlicher Wirtschaftsquellen» von Prof. Dr. Roger Sablonier am Historischen Seminar der Universität Zürich das Programm *Lector* als unterstützendes Hilfsmittel eingesetzt. Zur Einführung werden den TeilnehmerInnen in einer Seminarsitzung jeweils die Möglichkeiten des Programms vorgestellt und erläutert. Für die individuelle Arbeit stehen anschliessend zwei für diese Anwendung reservierte Arbeitsstationen zur Verfügung. Ausserdem können Studierende mit eigenem Macintosh das Programm zum Selbstkostenpreis erwerben. Daneben werden im Seminar weiterhin die kopierten Original- und Transkriptionsreader abgegeben, mit denen zum Teil auch in den Seminarsitzungen gearbeitet wird. Damit werden beide Methoden ungefähr gleichwertig nebeneinander gestellt, und den StudentInnen wird die Wahl überlassen, mit welchem Medium sie ihre Transkriptionsübungen machen wollen.

Bereits beim ersten Einsatz des Programmes (damals noch in einer Testversion mit einer kleineren Übungsauswahl) haben etwa die Hälfte der SeminarteilnehmerInnen ihre Leseübungen mit *Lector* absolviert. Dieser Anteil ist seither noch angestiegen und zeugt von der Akzeptanz dieses Unterrichtsprogrammes bei den Studierenden.

Von allen SeminarteilnehmerInnen, die mit *Lector* arbeiten, wird versucht, mittels eines Fragebogens ein Echo einzuholen. Ein auffälliges Resultat der Auswertung dieser Fragebogen ist der Umstand, dass fast alle BenutzerInnen bereits früher einmal Kontakt mit einem Computer hatten. Offenbar ist dies ein wichtiges Kriterium für die Überwindung der ersten Hemmschwelle (und vielleicht auch schon für das grundsätzliche Interesse).

Alle Befragten geben an, beim Umgang mit dem Programm keine grösseren oder gar keine Schwierigkeiten gehabt zu haben. Die Einführung am Bildschirm und das zur Verfügung stehende Handbuch werden entweder gar nicht benötigt oder dann als nützlich und hilfreich bezeichnet. Der Lernerfolg, verglichen mit der konventionellen Methode, wird mehrheitlich als gleichwertig oder sogar besser beurteilt, wobei bei den negativen Antworten die raschere Ermüdung der Augen als Hauptgrund für den schlechteren Lernerfolg angegeben wird. Alle bisher Befragten bejahten ein Bedürfnis nach einem solchen Programm und attestierten der vorliegenden Fassung, dieses Bedürfnis vollständig oder mindestens teilweise abdecken zu können.

Neben diesem primären Einsatzgebiet wird *Lector* auch immer häufiger in Proseminaren zur Geschichte des Mittelalters, aber auch zur individuellen Vorbereitung auf die Arbeit mit Originalquellen, beispielsweise für eine Seminar- oder Lizentiatsarbeit, mit gutem Erfolg eingesetzt.

Weiterer Ausbau

In der bestehenden Version 1.x ist *Lector* stark auf das hauptsächliche Anwendungsgebiet, die Wirtschaftsquellenseminare, ausgerichtet. Das zeigt sich vor allem in der Auswahl der Transkriptionsübungen, die aus dem Gebiet der Nordostschweiz stammen und den Zeitraum von 1219 bis 1557 abdecken. Daneben sind aber auch die Transkriptionsregeln sowie ein grosser Teil der angebotenen Hintergrundinformationen auf diesen Themenbereich ausgerichtet.

Für die nächsten Versionen von *Lector* ist ein Ausbau in zwei Richtungen vorgesehen. Einerseits soll ermöglicht werden, Übungen wie auch Transkriptionsregeln und Hintergrundinformationen an die jeweiligen Bedürfnisse anzupassen. Dazu ist geplant, in einem modularen Aufbau die Grundfunktionen des Programmes vom eigentlichen Inhalt zu trennen und dann neben dem eigentlichen Programm jeweils einzelne Übungspakete zu bestimmten Themen (beispielsweise zu einer bestimmten Quellengattung, zum Bestand eines Archives, zum Schrifttum einer Region oder zur Erlernung eines bestimmten Schrifttyps) anzubieten. Andererseits soll auch die Bedienungsprache, die bisher starr Deutsch ist, flexibel an die jeweiligen Anforderungen (Englisch, Französisch usw.) angepasst werden können.

Ausserdem werden im Moment Überlegungen angestellt, *Lector* auch auf andere Betriebssysteme – v.a. Microsoft Windows – zu portieren.

Literaturverzeichnis

Gilbert David: *HyperCard: New Ways of Writing, New Ways of Reading*. In: *History & Computing* Vol. 3, No. 3, 1991, S. 186–194.

Nielsen Jakob: *Hypertext and Hypermedia*; Boston 1990.

Ritter Gerold: *Lector – Transkription von mittelalterlichen Quellentexten computergestützt üben* (unveröffentlicht), Liz. Zürich 1992.

Russel Gill: *Hypertext*. In: *History & Computing* Vol. 3, No. 3, 1991, S. 183–185.