

Erstellen einfacher Register mit TUSTEP

Bader, Winfried

Veröffentlichungsversion / Published Version

Zeitschriftenartikel / journal article

Zur Verfügung gestellt in Kooperation mit / provided in cooperation with:

GESIS - Leibniz-Institut für Sozialwissenschaften

Empfohlene Zitierung / Suggested Citation:

Bader, W. (1990). Erstellen einfacher Register mit TUSTEP. *Historical Social Research*, 15(2), 119-137. <https://doi.org/10.12759/hsr.15.1990.2.119-137>

Nutzungsbedingungen:

Dieser Text wird unter einer CC BY Lizenz (Namensnennung) zur Verfügung gestellt. Nähere Auskünfte zu den CC-Lizenzen finden Sie hier:

<https://creativecommons.org/licenses/by/4.0/deed.de>

Terms of use:

This document is made available under a CC BY Licence (Attribution). For more information see:

<https://creativecommons.org/licenses/by/4.0>

Erstellen einfacher Register mit TUSTEP

*Winfried Bader**

Abstract: The index-programmes of the Tübingen University System for Text-Processing Programmes (TUSTEP) are not only suitable for drawing up of indexes ready for the press when publishing manuscripts. The module-character of TUSTEP also forms them into a tool which can be employed for the analysis of texts (linguistic statistics, lexicography, retrieval of databases). The use of TUSTEP index-programmes is demonstrated by three basic examples: In the first example an index of word forms is made out with references and frequencies. The second example shows the drawing up of a selected subject-matter index belonging to a continuous text. And in the third example a KWIC-index (Key Word In Context) of key words is produced from the entries of a bibliography (format of a database).

»Ein Register ist eine Suchhilfe zur Erschließung von Informationssammlungen (z.B. Monographien, Kartenwerke, Verzeichnisse, Sammlungen, Kataloge) über die vorgegebene Ordnung der Informationssammlung hinaus« (aus dem Entwurf DIN 31630). Doch Registerprogramme werden nicht nur gebraucht, um der Edition von Manuskripten schnell druckfertige Register begeben zu können, sondern sie eignen sich auch zur Analyse von Texten, z. B. um anhand von Wortlisten schnell zur

* Address all Communications to Winfried Bader, Zentrum für Datenverarbeitung (ZDV), Brunnenstraße 27, D-7400 Tübingen.

TUSTEP gibt es als PC-Version unter MS-DOS (IBM-PCs und kompatibel). Auf Großrechnern läuft TUSTEP derzeit unter den Betriebssystemen MVS (IBM), VM/CMS (IBM), VMS (DEC VAX); die Implementierung unter BS2000 (SIEMENS) und UNIX ist geplant.

Für Hochschulen gibt es für TUSTEP Campus-Lizenzen. Nähere Informationen über den Vertrieb von TUSTEP und den Bezug des Handbuchs sind bei Prof. Dr. Ott am Zentrum für Datenverarbeitung der Universität Tübingen zu erhalten. Für die Beratung in Fragen zu TUSTEP stehen die Mitarbeiter der Abteilung »Literarische und Dokumentarische Datenverarbeitung« am ZDV in Tübingen zur Verfügung. Köln 1.

Erstellung eines Sachregisters zu kommen, um über Wortlisten Rechtschreibfehler abzuprüfen, um durch rückläufige Register eine Lemmatisierung vorzubereiten, um absolute und relative Häufigkeiten der Wörter auszugeben zur sprachstatistisch Auswertung und um bestimmte Kategorien strukturierter Daten (Datenbank) aufzulisten, zu zählen und auszuwerten.

TUSTEP bietet zur Registererstellung kein fertiges Programm an, sondern Bausteine, die vom Benutzer in vielfältiger Weise programmiert und zu maßgeschneiderten Problemlösungen zusammengestellt werden können. Der modulare Aufbau von TUSTEP erlaubt eine sonst nicht erreichbare Flexibilität, der nicht nur die Erstellung druckfertiger Register in verschiedenen Formen und Sortierungen berücksichtigt, sondern auch die Ausgabe der Register und Zwischenergebnisse in Dateien, die erneut der Bearbeitung zugeführt werden können, wie es vor allem für den Einsatz der Registerprogramme als Analyseinstrument benötigt wird.

Der folgende Artikel möchte anhand von drei typischen Beispielen in das Arbeiten mit TUSTEP-Registerprogrammen einführen. Zunächst wird der einfache Fall gezeigt, zu einem Text einen vollständigen Wortindex zu erstellen. Ausgehend von diesem Prototyp aller Registerprogramme, zeigt das nächste Beispiel, wie man ein Register zu markierten Daten erstellt und dabei die Registereinträge differenziert behandeln kann. Dabei wird ausführlich besprochen, wie man die Auszeichnung der Quelldaten für das Register vornimmt. Ein letztes Beispiel zeigt die Erstellung eines KWIC-Index. Das Besondere an den Quelldaten dieses Beispiels ist ihre datenbankähnliche Struktur, wobei das Register nur auf eine bestimmte Kategorie zugreift und die Referenz aus dem Text genommen wird. Mit den drei einfachen Beispielen ist bereits ein weites Spektrum der Verwendung von Registern abgedeckt.

Grundbegriffe der Registererstellung

Zuerst müssen einige Grundbegriffe der Registererstellung und ihre Entsprechungen in TUSTEP angesprochen werden.

Ein Register besteht aus Registereinträgen, die sich in einen Registerbegriff (»Stichwort«, »Suchwort«, »Schlagwort«, »Registereingang« nach dem Entwurf zu DIN 31630) und eine oder mehrere Referenzen (»Registerausgang« nach DIN) gliedern. Dabei können sowohl der Registerbegriff als auch die Referenzen aus mehreren Teilen bestehen. Die Anordnung der Registereinträge erfolgt (meist) nach einer alphabetischen Ordnung der Registerbegriffe. Um von den Quelldaten zu diesem Endprodukt zu gelangen, müssen die Registerprogramme verschiedene Aufgaben wahrnehmen, die sich in TUSTEP auf mehrere Kommandoaufrufe verteilen.

In einem ersten Schritt werden aus dem Text die einzelnen Registerbegriffe isoliert. Jedem Registerbegriff wird dabei eine Referenz zugeordnet. Das Zusammenfassen mehrfach vorkommender gleicher Begriffe zu einem einzigen Registereintrag erfolgt erst nach der Sortierung. Zu jedem Registerbegriff wird ein Sortierschlüssel erstellt, der beim anschließenden Sortieren für die richtige Reihenfolge der Begriffe verantwortlich ist. Diese Aufgaben leistet das Kommando REGISTER-VORBEREITE (* RVORBEREITE). Das Erstellen der Sortierschlüssel erfolgt wie beim Kommando Sortier-Vorbereite. Die dafür erforderlichen Parameter und die inhaltliche Problematik des Sortierens wurden von mir in HSR Nr. 51, Vol. 14 (1989) Heft 3, S. 139-160 dargestellt. Darauf sei verwiesen.

Das Kommando SORTIERE sortiert die Registerbegriffe nach ihren Sortierschlüsseln. Die Registerbegriffe erhalten dadurch die gewünschte Reihenfolge. Gleichlautende Registerbegriffe kommen hintereinander zu stehen.

Im fertigen Register will man jeden Registerbegriff - auch wenn er mehrmals im Text belegt ist - nur einmal stehen haben, dafür aber mit all seinen Referenzen. Die Zusammenfassung gleicher Registerbegriffe zu einem Registereintrag mit mehreren Referenzen ist die spezifische Aufgabe des Kommandos REGISTER-AUFBEREITE (# RAUFBEREITE). Mit diesem Kommando läßt sich das Register auch typographisch gestalten: bei allen Bestandteilen des Registereintrags lassen sich Zeichenfolgen ergänzen; absolute und relative Häufigkeiten können errechnet und mit ausgegeben werden. Bei mehrteiligen Begriffen können gleichlautende Teile weggelassen oder - z. B. durch Sonderzeichen - ersetzt werden.

Das Ergebnis von REGISTER-AUFBEREITE kann mit dem Kommando DRUCKE ohne eine weitere Nachbearbeitung direkt ausgedruckt oder für die weitere Bearbeitung auf einer Datei zur Verfügung gestellt werden.

Diese drei Kommandoaufrufe benötigt man zur Erstellung eines Registers. P. Stahl hat in seinem Artikel in HSR Nr. 52, Vol. 14 (1989) Heft 4, S. 159-162 gezeigt, wie sich mit Hilfe von Makros die Erzeugung der drei Kommandoaufrufe in eine vom Benutzer definierte dialogische Kommandosprache fassen lassen. Auf diese Möglichkeit von TUSTEP sei hingewiesen. Die folgenden Beispiele zeigen die Programme, die Stahls Dialogabfragen zugrundeliegen.

1. Beispiel: Erstellung eines Wortindexes

Als erstes Beispiel wird aus einem fortlaufenden Text ein Wortindex mit Referenzen (Seiten- und Zeilennummer der TUSTEP-Datei) erstellt. Genauer gesagt soll es sich um ein vollständiges Wortformenregister handeln, d. h. es werden *alle* Wörter des Textes *unverändert* in das Register aufgenommen. Das Programm ist der Prototyp aller TUSTEP-Registerprogramme, und braucht für andere Anwendungen nur abgeändert oder verfeinert werden. An die Quelldatei werden keine besonderen Anforderungen gestellt. Man kann jede TUSTEP-Datei damit bearbeiten. Das Beispiel hat jedoch vorwiegend Texte im Blick. Als Referenz dienen die Satznummern (Seiten- und Zeilennummer). Für die Registerprogramme ist das die sogenannte Standardreferenz, mit der sich besonders leicht arbeiten läßt.

Das Programm zur Erzeugung des Wortindex ist in Abbildung 1 wiedergegeben. Das Programm - es wird in TUSTEP mit dem Editor einfach in eine Datei geschrieben und dann mit dem Kommando TUE unter Angabe des Dateinamens zur Ausführung gebracht - beginnt mit einer Kommentarzeile (*=). Dann folgt der Kommandoaufruf von REGISTERVORBEREITE (*rv). Hinter dem Kommandonamen folgen in derselben Zeile die Spezifikationen des Kommandos: zunächst die Angabe der Quelldatei, die im Beispiel den Namen **text** hat, dann die Zieldatei, in die das Ergebnis von REGISTERVORBEREITE geschrieben werden soll (TUSTEP-Programme arbeiten immer kopierend, d. h. die Quelldaten bleiben unverändert erhalten, das Ergebnis eines Programms findet man in einer anderen Datei); als Name der Zieldatei wird **-std-** (Standard) angegeben, d. h. die Datei wird von TUSTEP zur Verfügung gestellt. Mit **modus = +** wird verlangt, die Registerinträge mit Referenzen zu versehen, **loeschen = +** bewirkt das Löschen eventuell schon vorhandener Daten in der Zieldatei, und **parameter = *** bedeutet, daß die Parameterangaben der Kommandozeile folgen.

Mit den Parametern macht man die Angaben, auf welche Weise das Programm die Quelldaten zu bearbeiten hat. Parameter **str** gibt an, daß die Silbentrennung der Quelldaten aufgehoben werden soll, d. h. getrennte Wörter werden vollständig zu der Zeile gezählt, in der sie beginnen. In Zeile 4 und 5 folgt das Kernstück von REGISTERVORBEREITE. Der Text der Quelldatei wird in einzelne Registerbegriffe zerlegt. Dies erfolgt durch Angabe eines Trennzeichens (Parameter **tr**), das hier eines der Zeichen aus der in der darüberliegenden Zeile definierten Zeichengruppe 1 ist: Komma, Blank, Punkt, die weiteren Satzzeichen und die Klammern sind Trennzeichen für die Registerbegriffe. Die Daten der Quelldatei werden am Trennzeichen in Registerbegriffe zerlegt, wobei die Trennzeichen selbst entfallen und der dazwischenliegende Text jeweils einen Registerbegriff bildet. Auf diese Weise erhält man alle Wörter (zwischen Blanks) je

für sich als Registerbegriffe, wobei die Satzzeichen nicht Bestandteil der Registerbegriffe sind. Würde man nur das Blank als Worttrenner angeben, so blieben die Satzzeichen erhalten; sie bleiben beim Wort stehen. Dies ergibt neue Registerbegriffe, die sich vom gleichen Wort ohne Satzzeichen unterscheiden und im Register getrennt behandelt werden.

Die weiteren Parameter **xsl** und **ssl** dienen dem Aufbauen des Sortierschlüssels für die Sortierung der Registerbegriffe. Sie entsprechen in vollem Umfang den betreffenden Parametern von SORTJER-VORBEREITE (vgl. HSR Nr. 51, 150-152). Der Einfachheit halber ist hier nur ein Sortierschlüssel verwendet, was für die meisten Quelldaten bereits genügt.

Die Zieldatei von REGISTER-VORBEREITE wird nach dem Sortierschlüssel sortiert. Der Sortierschlüssel steht hinter der Referenz (14 Zeichen) und zwei weiteren für spezielle Anwendungen benötigten Zeichen ab Position 17 in der Zieldatei von REGISTER-VORBEREITE. Beim Kommando SORTIERE müssen zur Spezifikation Sortierfeld die Positionen angegeben werden (17 + 20), auf denen sich der Sortierschlüssel befindet. Nach dem Sortieren stehen die Registerbegriffe mit ihren Referenzen in einer alphabetischen Reihenfolge in der Datei; gleiche Registerbegriffe stehen hintereinander.

Das folgende Kommando REGISTER-AUFBEREITE dient dazu, gleiche Registerbegriffe zu einem Registereintrag zusammenzufassen und die Registerausgabe typographisch zu gestalten.

Der Sortierschlüssel hat in REGISTER-AUFBEREITE keine Bedeutung mehr. Die obligate, mit der Angabe in REGISTER-VORBEREITE identische Angabe von Parameter **ssl** bewirkt, daß der Sortierschlüssel getilgt wird.

Der Parameter **dr** steuert das Druckformat, wie beim Kommando FORMATIERE: hier ist ein dreispaltiger Ausdruck mit einer Spaltenbreite von 23 Zeichen angegeben. Der Parameter **ger** gibt die Geräart des Druckers an, da das Ergebnis (Protokolldatei) von REGISTER-AUFBEREITE direkt ausgedruckt werden soll.

Die folgenden Parameter dienen zur typographischen Gestaltung eines Registereintrags. Es werden Angaben zur Behandlung der Referenzen gemacht: Parameter **rlf** besagt, daß die Referenz aus zwei Teilen (mit 6 und 3 Zeichen) besteht (Seiten- und Zeilennummer), Parameter **nrt** besagt, daß nach dem ersten Referenzteil ein Punkt ergänzt werden soll, und Parameter **rtw** besagt, daß der erste Teil der Referenz auch dann ausgegeben werden soll, wenn er mit dem ersten Teil der zuletzt genannten Referenz übereinstimmt.

Mit dem Druckkommando (***dr**) wird das Ergebnis gleich auf einen Drucker vom Typ **fx** geschickt. Einige Zeilen eines solchen Ergebnisses sind in Abbildung 2 wiedergegeben.

Zur typographischen Gestaltung der Registereinträge ist in dem Beispiel *ein* Vorschlag gemacht. Vielfältige Variationen sind möglich. Es können Zeichenfolgen (d. h. auch typographische Steuerzeichen für das Ausdrucken) an folgenden Stellen des Registereintrags ergänzt werden: Vor und hinter dem Registerbegriff, vor der ersten und nach der letzten Referenz, sowie zwischen den einzelnen Referenzen. Dasselbe gilt für jede einzelne Referenz, wenn sie aus mehreren Teilen besteht. Bei mehrgliedrigen Registerbegriffen können zwischen den einzelnen Teilen des Registerbegriffes Zeichenfolgen ergänzt werden, und mit jedem einzelnen Teil eine neue Zeile begonnen werden. Es können Ersatzzeichenfolgen bestimmt werden für Textteile, die wegen Identität mit den vorhergehenden wegfallen. Außerdem kann vor dem Registerbegriff eine laufende Nummer verlangt werden, nach dem Registerbegriff die absolute und relative Häufigkeit, jeweils umschlossen von Zeichenfolgen, die man ebenfalls eigens angeben kann. Das Einfügen von Leerzeilen und Zwischenüberschriften, sowie die Positionierung jedes Textteiles, des Beginns der Referenzen und das Einrücken von Folgezeilen ist möglich.

Einige dieser Möglichkeiten werden in den folgenden Beispielen noch gezeigt.

Wortindex mit absoluten und relativen Häufigkeiten

Eine leichte Abwandlung des ersten Beispiels ist die Erstellung eines Wortindexes mit absoluten und relativen Häufigkeiten. Eine solche Liste benötigt man, um einen Überblick über den Wortbestand eines Textes zu bekommen, z. B. als Hinweis auf wichtige Wörter für ein Sachregister. Für sprachstatistische Zwecke kann man die Häufigkeitsliste mit weiteren Programmen auswerten, z. B. die Liste nach Häufigkeiten sortieren oder mit Registerprogrammen Häufigkeitsklassen bilden oder Berechnungen durchführen.

Das Programm für den Wortindex mit Häufigkeiten ist in Abbildung 3 wiedergegeben. Neben der Ausgabe der Häufigkeiten soll an diesem Programm gezeigt werden, wie man im Registerprogramm die Ausgabe von Referenzen unterdrücken kann (natürlich sind die Häufigkeitsangaben auch möglich, wenn Referenzen ausgegeben werden) und wie man die Ausgabe von REGISTER-ALTBEREITE in eine Datei lenkt, so daß man das Ergebnis anschließend mit weiteren Programmen bearbeiten kann.

Zur Erstellung einer Wortformenliste mit Häufigkeiten braucht das in Abbildung 1 vorgestellte Programm nur leicht abgeändert zu werden. Das Ergebnis wird in eine Datei namens **liste** geschrieben, die man vorher einrichten bzw. anmelden muß.

Bei REGISTER-VORBEREITE wird **modus** = - angegeben, damit unterbleibt die Ausgabe der Referenzen. Durch den Wegfall der Referenzen

ändert sich in SORTIERE die Position des Sortierfeldes: der Sortierschlüssel beginnt nun gleich auf der ersten Position (1 +20). Auch die anderen Angaben zur Referenz entfallen in REGISTER-VORBEREITE und in REGISTER-AUFBEREITE.

Neu hinzu kommen in REGISTER-AUFBEREITE die Parameter, die die Ausgabe der Häufigkeiten und ihre Gestaltung verlangen. Parameter **ah** verlangt die Ausgabe der absoluten Häufigkeit. Parameter **vah** und **nah** geben die Zeichenfolgen an, die vor und nach der absoluten Häufigkeit ergänzt werden sollen. Parameter **rh** verlangt die Ausgabe der relativen Häufigkeit mit einer Dezimalstelle (Parameter **rhd**). Die Parameter **vrh** und **nrh** dienen zur Gestaltung, was vor und hinter der relativen Häufigkeit stehen soll.

Das Ergebnis wird nicht zum Drucken vorbereitet, sondern in die Datei namens **liste** geschrieben.

Der Anfang der Wortformenliste mit Häufigkeiten (Datei **liste**) ist in Abbildung 4 wiedergegeben. Die Datei läßt sich mit TUSTEP-Programmen weiterbearbeiten.

2. Beispiel: Erstellung eines Sachregisters

Eine andere Problemlage ist die Erstellung eines Registers (z. B. ein Sach- oder Autorenregister) zu einem Buch. Es sollen kein vollständiger Wortindex erstellt werden, sondern ausgewählte Begriffe sollen mit bestimmten Referenzen im Register erscheinen. Aus dem riesigen Angebot des Wortindex will man also nur einen kleinen ausgewählten Teil wirklich verwenden.

Falls man in den Quelldaten noch keinerlei Vorbereitungen hinsichtlich des Registers getroffen hat, kann man - falls der Umbruch des Buches schon endgültig feststeht - im wörtlichen Sinn diesen Weg beschreiten: man erstellt einen Wortindex, gibt ihn in eine Datei aus und streicht im Editor all die Begriffe und Referenzen, die man nicht haben will. Dies ist ein schneller Weg, zu einem einfachen Sachregister zu kommen. Normalerweise wird man aber den im folgenden beschriebenen Weg einschlagen. Für ein ausgewähltes Register ist es sinnvoll, in den Quelldaten die Wörter auszuzeichnen, die man im Register haben will. Am einfachsten ist es, bei der Erfassung des Textes die für das Register (eventuell) wichtigen Wörter zu markieren; in dieser Phase beschäftigt man sich ohnehin intensiv mit den einzelnen Stellen, und kennt daher am besten die für das Register relevanten Begriffe.

Das Markieren von Registerbegriffen im Text

Bei der Wahl der Zeichen, die man für die Markierung der Registerbegriffe im Text verwendet, ist zu beachten, daß die Markierungen den Umbruch nicht stören und beim Ausdrucken nicht erscheinen dürfen, intern aber erhalten bleiben müssen, so daß das Registerprogramm nach erfolgtem Umbruch auf sie zugreifen kann. Drei Möglichkeiten der Markierung bieten sich an:

a) Die Markierung ist identisch mit einer typographischen Auszeichnung. Werden z. B. in einem Text alle Autorennamen (und nur die) mit Kapitälchenschrift gesetzt, so ist die typographische Auszeichnung der Kapitälchen (bei FORMATIERE *k+ und *k -) gleichzeitig Anfangs- und Endekennung der Registerbegriffe fürs Autorenregister. Im Registerprogramm wird *k+ und *k - als Begrenzung der Registerbegriffe angegeben. So erscheinen im Register nur die Autoren, denn ausschließlich sie sind auf diese Weise im Text gekennzeichnet. Die Auszeichnung der Kapitälchen stört den Umbruch nicht, denn wie alle Steueranweisungen bleibt sie beim Ausdrucken unsichtbar. Intern bleibt sie aber erhalten.

b) Die Markierungen werden als Kommentar eingetragen.

In den meisten Fällen stimmen typographische Auszeichnungen mit der Zuordnung der ausgezeichneten Begriffe zu einem Register nicht überein. Z. B. kommen in ein Sachregister Begriffe, die im Text selbst keine besondere Funktion haben und daher auch nicht besonders typographisch hervorgehoben sind. In diesem Fall muß man die Begriffe fürs Register durch andere Markierungen auszeichnen, die keine typographischen Steueranweisungen sind und beim Ausdrucken selbst auch nicht erscheinen. Man verwendet zur Markierung beliebige Zeichen oder Zeichenfolgen, die man aber durch Steueranweisungen explizit als Kommentar kennzeichnet, d. h. man gibt die Steueranweisung, beim Ausdrucken die folgenden Zeichen zu ignorieren, sie aber intern beizubehalten. Bei FORMATIERE ist dies die Steueranweisung @I. Für ein Sachregister wählt man z. B. die Zeichenfolge fs zur Kennzeichnung des Anfangs eines Registerbegriffs und] für das Ende. Die Zeichen werden mit der Steueranweisung @I auf Kommentar gesetzt. Damit hat ein im Text gekennzeichnete Begriff in der Datei folgendes Aussehen: @I [s Sachbegriff©I] oder auch @I [s zusammengesetzter Sachbegriff @I]. Ein Registerbegriff kann - im Unterschied zum einfachen Wortindex - auch aus mehreren Wörtern bestehen.

c) Der Registerbegriff wird als Kommentar eingetragen.

Die bisherigen Vorschläge zur Markierung waren davon ausgegangen, daß das markierte Wort selbst im Druck erscheint. Das ist aber nicht immer der Fall. Z. B. stehen viele Wörter in grammatisch veränderter Form im Text, im Register will man sie aber in ihrer Grundform zitieren; oder man

will ins Register Schlagwörter aufnehmen, die im Text selbst nicht vorkommen, die aber inhaltlich einen bestimmten Abschnitt charakterisieren; auch Verweise von einem Begriff auf einen anderen müssen in die Quelldatei geschrieben werden, haben aber mit dem auszudruckenden Text nichts zu tun. In diesen Fällen schreibt man den gesamten Registerbegriff als Kommentar in den Text. Die Steueranweisungen für Kommentaranfang und Commentarende dienen dann als Kennungen fürs Registerprogramm. Im Text kann das z. B. folgendermaßen aussehen:

zu druckender Text @1+ Schlagwort des Sachregisters @1- weher im Text @1 + Text, siehe auch Ausdruck @T und weitere Texte @1 + Text @1-.

In diesen kurzen Text wurde fürs Register ein Schlagwort eingefügt, ein Verweis gemacht und ein Wort auf seine Grundform zurückgeführt. Sinnvoll ist es, den Verweis noch zusätzlich zu kennzeichnen, falls er im Register ohne Referenz erscheinen soll, z. B. @1+ **'Verweis, siehe auch Hinweis** @1-.

Die Arbeit des Auszeichnens kann man sich erleichtern, indem man die Kennzeichnungen im Text in einer abgekürzten - aber dennoch eindeutigen - Form vornimmt, und anschließend - z. B. im FORMATIERE-Programm selbst mit dem Parameter x - in die ausführliche Form austauscht.

Quelldatei für das Registerprogramm?

Nachdem im Text die Registerbegriffe markiert sind, erhebt sich die Frage nach der Quelldatei fürs Registerprogramm. Man kann nicht einfach die Datei nehmen, in der die Daten erfaßt wurden. Denn als Referenz soll die Seitenzahl des umbrochenen Textes dienen, die man aber erst weiß, nachdem das Kommando FORAMTIERE oder das Satzprogramm den Text bearbeitet hat.

FORMATIERE gibt den Text zum einen in die Protokoll-Datei aus. Diese Datei kann mit dem Kommando DRUCKE ausgedruckt werden. Als Quelle fürs Register ist sie jedoch ungeeignet, da sie nur noch Druckersteuerzeichen enthält und die Steueranweisungen, damit auch die Markierungen der Registerbegriffe, nicht mehr erkennbar sind. FORMATIERE gibt die formatierten Daten gleichzeitig noch auf eine Zieldatei aus. In dieser Datei bleiben alle Steueranweisungen, somit auch die Markierungen der Registerbegriffe, erhalten. Gegenüber der Quelldatei ändert sich in der Zieldatei aber die Satzeinteilung. Die Zeileneinteilung (TUSTEP-Sätze) dieser Datei entspricht der Seiten- und Zeileneinteilung des auf dem Papier gedruckten Textes. Die TUSTEP-Satznummer gibt dabei Seite und Zeile des Ausdrucks wieder. Der Text mit der Satznummer 5.27 in der Zieldatei steht auf dem Papier auf der Seite 5 in der 27. Zeile. D. h. die Satznummern der Zieldatei sind die Referenzen, die fürs Register benötigt

werden. Die Zieldatei des Kommandos **FORMATIERE** ist die Datei, die als Quelle fürs Registerprogramm genommen wird.

Werden in der Quelldatei noch Veränderungen vorgenommen, 50 entspricht die Zieldatei nach einem erneuten Lauf von **FORMATIERE** wiederum dem neuen Umbruch und kann erneut als Quelle für ein Register dienen, das die neuen, aktuellen Referenzen angibt.

Hat man eine Quelldatei namens **text**, in der der Text erfaßt wurde, so kann man den Text mit dem Kommando

***fo, text, umbdat, fx, +, pr ***» -std-

formatieren. Die Standard-Protokoll-Datei läßt sich mit dem Kommando ***dr, -std-, fx** ausdrucken. In der (vorher eingerichteten) Zieldatei namens **umbdat** stehen die Daten in der Zeileneinteilung des Ausdrucks, aber mit den Steueranweisungen der Quelldatei **text**. Die Datei **umbdat** dient als Eingabe fürs Registerprogramm.

Wird die Quelldatei mit dem TUSTEP-Satz-Programm für die Lichtsaterstellung weiterverarbeitet, so ist das Vorgehen dasselbe. Mit den Steueranweisungen **&x** und **&x{** gibt es die Möglichkeit, Kommentare in den Text zu schreiben. Das Kommando **SATZ** erzeugt ebenfalls eine Zieldatei, bei der die TUSTEP-Satznummern dem Umbruch auf Papier entsprechen, so daß sie als Quelle für das Registerprogramm dienen kann.

Register ausgezeichneter Begriffe

Mit der Datei **umbdat** als Quelle kann zum Text ein Register gemacht werden. Neben der Abgrenzung der Registerbegriffe aufgrund von Auszeichnungen kommen gegenüber dem Wortindex fürs folgende Programm noch weitere Anforderungen hinzu: 1) Registerbegriffe, die aus mehreren Wörtern bestehen, sollen auch in umgedrehter Form ins Register. Der Begriff *zusammengesetztes Schlagwort* - um bei obigem Beispiel zu bleiben - soll zusätzlich als *Schlagwort, zusammengesetztes* im Register erscheinen. 2) Die (mit einem Stern gekennzeichneten) Verweise sollen ohne Referenz ins Register aufgenommen werden. 3) Bei der Referenz soll die Angabe der Zeile unterbleiben und nur die Seitennummer ausgegeben werden. 4) Beim Vorkommen eines Begriffs auf zwei aufeinanderfolgenden Seiten ist die erste Referenz mit / zu versehen und die zweite zu unterdrücken. Kommt ein Begriff auf mehr als zwei unmittelbar aufeinanderfolgenden Zeilen vor, so ist die Referenz als Bereich anzugeben, z. B. 15-19.

Das Programm, das diesen Bedingungen genügt, ist in Abbildung 5 wiedergegeben.

Der Hauptunterschied zu den bisherigen Programmen besteht in der Isolierung der Registerbegriffe. Mit Parameter **ea** und **ee** werden Zeichenfol-

gen angegeben, die in den Eingabedaten den Anfang bzw. das Ende eines Registerbegriffs markieren. Der Text zwischen den angegebenen Zeichen bildet den Registerbegriff. Geht man von obigen Beispielen aus, so beginnt ein Registerbegriff entweder bei @l +, wenn es sich um einen zusätzlich eingetragenen Begriff handelt, oder bei @l[s, wenn der Registerbegriff auch gedruckt wird. Das Ende des Registerbegriffs ist entweder bei @I- oder @l] je nach dem, was das Programm nach einer gefundenen Anfangskennung als nächstes im Text findet.

Das Umdrehen der Registerbegriffe (vgl. oben Anforderung 1) erfolgt durch die Angabe von Parameter `umd` (Angaben zum Umdrehen). Durch die Zahl 2 wird verlangt, daß die Registereinträge sowohl in umgedrehter als auch in nicht-umgedrehter Form in das Register sollen. Das Umdrehen erfolgt gemäß der Voreinstellung (eine andere Angabe ist möglich) am letzten Blank des Registereintrags, zwischen den umgedrehten Teilen wird standardmäßig ein Komma ergänzt (auch hier ist eine andere Angabe möglich).

Mit dem Parameter `orf` ist das Zeichen angegeben, mit dem Registerbegriffe beginnen, die ohne Referenz ausgegeben werden sollen (vgl. oben Anforderung 2). Das Zeichen selbst wird, nachdem es interpretiert ist, entfernt. Im obigen Beispiel war der Stern am Beginn des Verweises als Kennzeichen für diesen Zweck eingetragen worden; der Stern wird daher in Parameter `orf` angegeben.

Mit Parameter `irl` wird die Ausgabe der Standardreferenz auf 6 Zeichen beschränkt, das ist der Teil, der die Seitennummer enthält (vgl. oben Anforderung 3).

Sortiert wird anschließend nach dem Feld, das auf Position 9 beginnt (davor stehen 6 Zeichen für die Referenz, das 7. und 8. Zeichen sind Typ und Steuerbyte).

In REGISTER-AUFBEREITE ist es wichtig, dieselben Längenangaben für Sortierschlüssel und Referenz wie in REGISTER-VORBEREITE zu machen (Parameter `ssl` und `irl`). Parameter `ger` zur Angabe des Drucker-typs ist ebenfalls obligat.

Die drei weiteren Parameter dienen zum Zusammenfassen der Referenzen, falls ein Registerbegriff auf zwei oder mehr aufeinanderfolgenden Seiten belegt ist (vgl. oben Anforderung 4). Mit dem Parameter `rff` wird verlangt, daß das Zusammenfassen aufeinanderfolgender Referenzen vorzusehen ist. Mit dem Parameter `fl` wird die Zeichenfolge angegeben, die bei genau zwei aufeinanderfolgenden Referenzen hinter der ersten Referenz unter Wegfall der zweiten Referenz geschrieben werden soll. Aus 14 15 wird 14f. Mit dem Parameter `zrp` ist angegeben, welche Zeichenfolge bei mehr als zwei aufeinanderfolgenden Referenzen zwischen der ersten und letzten unter Wegfall der mittleren ergänzt werden soll: aus 7 8 9 10 wird 7-10.

Das Ergebnis, das in der Standard-Protokoll-Datei steht, kann mit dem Kommando***dr**, **-std-**, **fx** ausgedruckt werden.

Verschiedene Register zu einem Text

Will man zu einem Text verschiedene Register erstellen, z. B. Sach-, Orts- und Personenregister, so muß man die Auszeichnung der Begriffe für die verschiedenen Register im Text differenzieren. Für den einen Fall wurde eine solche Differenzierung bereits oben angedeutet: @1[s ist der Anfang für einen Begriff des Sachregisters, entsprechend könnte dann @1[o und @1[p die Markierung für Orts- und Personenregister sein. Auch für die andere Art der Markierung läßt sich die Anfangskennung erweitern, statt @I+ schreibt man z. B. @1+ s: bzw. @1+ o: oder @1+ p: fürs Sach-, Orts- und Personenregister. Dann werden einfach mit drei getrennten Programmläufen je für sich die Register gemacht, wobei man für jedes Register die Angaben des Parameters **ea** im Programm entsprechend ändert. Bei der Erstellung des Ortsregisters schreibt man z. B.

```
ea /@i [o/@i+ o:/
```

und wählt damit nur die Begriffe aus, deren Anfangskennung das **o** enthalten. Die Endekennung (Parameter **ee**) aller Register bleibt gleich.

Über die Unterscheidung verschiedener Typen von Registerbegriffen läßt sich dasselbe Ergebnis auch in *einem* Programmlauf erreichen. Auch gibt es noch die Möglichkeit, die verschiedenen Register in ein einziges Alphabet zu sortieren, und die Zugehörigkeit der Registereinträge zur jeweiligen Kategorie nur typographisch zu unterscheiden. Die notwendigen Erklärungen dazu findet man im TUSTEP-Handbuch.

3. Beispiel: Erstellen eines KWIC-Index

Neben der Erstellung eines KWIC-Index wird im dritten Beispiel auf das Problem eingegangen, wie man bei strukturierten Daten (Bibliographie, Datenbank) ein Register zu den Wörtern in einer Kategorie erstellt und als Referenz den Inhalt einer anderen Kategorie in dieser Datenbank (laufende Nummer, Datum, Ort, etc.) ausgibt.

Quelldatei für das folgende Beispiel ist die in meinem Artikel in HSR 51 verwendete Bibliographie namens **bib**. Durch die Kennungen *&a*, *&t*, *&o*, ... sind die verschiedenen Kategorien jedes Eintrags eindeutig bezeichnet. Allgemein gesprochen handelt es sich dabei um datenbankähnlich strukturierte Daten, wobei ein Datensatz (= Bibliographieeintrag) mehrere Felder (= Kategorien) enthält. Zu dieser Bibliographie soll ein KWIC-Index der Titelstichwörter hergestellt werden. Als Referenz wird der Name des Autors angegeben. Allgemein gesprochen wird ein Feld des Datensatzes untersucht, wobei als Referenz der Inhalt eines anderen Feldes dient.

Ein KWICTndex (keyword in context) ist ein Register, bei dem zu den einzelnen Registerbegriffen nicht nur die Referenzen aufgezählt werden, sondern es werden die einzelnen Registerbegriffe jeweils vollständig mit ihrem Kontext aufgeführt. Die Registerbegriffe stehen als »Schlüsselwörter« in der Mitte der Zeile. Man kann sie durch das Ergänzen von Zeichenfolgen eigens hervorheben. Zu Beginn jeder Zeile steht die Referenz, unter der man das Schlüsselwort mit diesem Kontext in der Datei findet. Wechselt das Schlüsselwort, so wird es zunächst in einer eigenen Zeile genannt.

Praktische Anwendung findet ein solcher KWIC-Index zu einer Bibliographie bei der Erstellung von Katalogen zur Literatursuche. Die Suche in einem solchen Katalog erfolgt nach den alphabetisch geordneten Schlüsselwörtern. Der zu jedem Schlüsselwort als Kontext aufgeführte Titel bietet die weitere Information, unter welcher Hinsicht der entsprechende Begriff im Buch bzw. Aufsatz behandelt wird.

Bei fortlaufenden Texten ist ein KWIC-Index für die lexikographische Arbeit wichtig. Er ist bereits eine einfache (unlemmatisierte) Konkordanz. Der Kontext kann dem Wissenschaftler zur Unterscheidung von Homographen dienen. Er liefert auch Information, wo welches Wort in gleicher grammatikalischer Konstruktion oder gleicher Bedeutung verwendet wurde.

Das Programm zur Erstellung des KWIC-Index, das bereits viele Sonderwünsche berücksichtigt - man kommt auch zu einem vernünftigen KWIC-Index mit weniger Parametern -, ist in Abbildung 6 wiedergegeben. Auch die Erstellung eines KWIC-Indexes geht von dem in Abbildung 1 wiedergegebenen Registerprogramm aus.

Bei REGISTER-VORBEREITE wird zur Spezifikation **modus** der Wert KWIC angegeben. Außerdem ist eine Angabe zur Spezifikation daten notwendig. Der für die Sortierung notwendige Sortierschlüssel und die eigentlichen Daten werden bei der Erstellung eines KWIC-Indexes aus Platzgründen intern getrennt behandelt und erst bei REGISTER-AUFBEREITE wieder zusammengefügt. Die Angabe **-STD-** bedeutet, daß die für das Zwischenspeichern der Daten notwendige Datei von TUSTEP selbst eingerichtet wird.

In REGISTER-VORBEREITE werden mit Parameter **aa** die TUSTEP-Sätze, die zu jeweils einem Bibliographieeintrag gehören, zu einer Texteinheit zusammengefaßt (angegeben ist die Zeichenfolge, die eindeutig den Beginn eines Bibliographieeintrags bezeichnet)

Mit den Parametern **ea** und **ee** wird der Teil der Texteinheit abgegrenzt, der den (maximalen) Kontext für die Schlüsselwörter bildet, der im ausgedruckten Ergebnis nur durch das Ausgabedatenformat beschränkt wird. Im vorliegenden Beispiel dient als Kontext der Titel, der nach **&i** beginnt und vor dem nächsten **&** endet.

Innerhalb des Titels werden die Schlüsselwörter durch Trennzeichen mit Parameter **swt** abgegrenzt, analog der oben beschriebenen Abgrenzung der Registerbegriffe mit Parameter **tr**. So wird jedes Wort eines Titels, einmal zum Schlüsselwort, der gesamte Titel wird dabei als Kontext so oft wiederholt, wie er Schlüsselwörter enthält. (Damit ist ein Problem verbunden: in dieser Phase beträgt das für den KWIC-Index zu verarbeitende Datenmaterial ein vielfaches des Ausgangstextes. Für das Zusammenfügen der Sortierschlüssel mit den eigentlichen Daten wird nach der Sortierung sehr viel (Plattenzugriff-)Zeit benötigt. Bei der Erstellung eines KWIC-Index mit einer großen Datenmenge ist dies zu bedenken, und es sind eventuelle Vorkehrungen zu treffen)

Die Sortierung erfolgt nach den Sortierschlüsseln, die in erster Linie (1. Sortierschlüssel) aus dem Schlüsselwort, in zweiter Linie (4. Sortierschlüssel) aus dem Kontext gebildet werden, so daß der KWIC nach den Schlüsselwörtern sortiert ist, und bei mehreren Einträgen mit gleichem Schlüsselwort die alphabetische Reihenfolge des rechten Kontextes maßgebend ist. Sind zwei Schlüsselwörter und ihr rechter Kontext identisch, so entscheidet die Reihenfolge der Referenz, die normalerweise mit der Reihenfolge in der Quelldatei übereinstimmt und daher nicht gesondert berücksichtigt zu werden braucht. Im vorliegenden Fall, in dem es sich bei den Referenzen um Text handelt, ist zur Sortierung nach den Referenzen ein weiterer Sortierschlüssel (7. Sortierschlüssel) notwendig.

Das Austauschen der Umlaute für die Sortierschlüssel ist wieder vereinfacht wiedergegeben. Für die genauere Sortierung vgl. HSR 51, S. 139-160 und das TUSTEP-Handbuch.

Als Referenz wird nicht die Standardreferenz (Satznummer) genommen, sondern ein Teil des Textes, im vorliegenden Fall der Autorenname. Man gibt dazu mit Parameter **rfa** und **rfe** Zeichenfolgen an, die den Anfang bzw. das Ende der Referenz im Text kennzeichnen. In den Beispieldaten sind die Autoren durch die Zeichenfolgen **&a** und **&** eindeutig abgegrenzt. Für die Referenz muß ihre maximale Länge in Parameter **rfl** und **irf** angegeben werden. Diese Länge ist bei der Bestimmung der Position des Sortierfeldes zu berücksichtigen.

Sortiert wird nach dem Sortierschlüssel, der sich aus den einzelnen Sortierschlüsseln für Schlüsselwort, Kontext und Referenz zusammensetzt. Anzugeben ist bei SORTIERE das gesamte Sortierfeld, d. h. die Summe aller Angaben in Parameter **ssl**. Der Sortierschlüssel beginnt auf Position 23 (20 Positionen für die Referenz plus 2 Position für Typ und Steuerbyte stehen davor).

REGISTER-AUFBEREITE bereitet den KWIC-Index zum Druck vor. Eine Ausgabe in die ZIEL-Datei ohne die Drucksteuerzeichen ist nicht vorgesehen. Die Schlüsselwörter werden entsprechend der Angabe in Parameter **swp** positioniert (sinnvoll ist etwa die Zeilenmitte), und mit

ihrem Kontext, der maximal eine Druckzeile beträgt, ausgegeben. Vor und hinter dem Schlüsselwort lassen sich mit den Parametern **ysw** und **nsw** Zeichenfolgen ergänzen, hier die Steueranweisung zum Untersfrächen und vor dem Schlüsselwort noch zwei zusätzliche Blanks. Der Druckertyp und das Druckformat werden über Parameter **ger** und **dr** angegeben.

Die Referenz steht am Beginn jeder Zeile. Mit den Parameter **nrf** wird hinter der Referenz eine Zeichenfolge ergänzt, hier ein Doppelpunkt. Parameter **rtp** besagt, nach welcher Position der Kontext frühestens beginnen soll. Mit Parameter **vk** wird eine Zeichenfolge für den Beginn des Kontextes angegeben.

Der Parameter **dre** bestimmt, daß vor jedem neuen Schlüsselwort eine Leerzeile eingefügt und die isoliert stehende Nennung des Schlüsselworts auf die Position 42 positioniert wird, um ein optisch ansprechendes Bild zu erhalten.

Zu den Parametern **lrl**, **rfl** und **ssl** sind dieselben Angaben wie in REGISTER-VORBEREITE zu machen, damit das Programm mit den Feldlängen der verschiedenen Teile zurecht kommt.

Das Ergebnis von REGISTER-AUFBEREITE kann mit dem Kommando ***dr, -std-, fx** auf den Drucker ausgegeben werden. Mit den drei Beispielen wurden typische Probleme der Registerarbeit und ihre Lösung mit Hilfe von TUSTEP-Programmen gezeigt. Durch Kombination verschiedener TUSTEP-Bausteine lassen sich auch weiterführende Probleme der Registererstellung und des Einsatzes von Registerprogrammen bei der Analyse von Datenbanken lösen. Jeder Anwender wird dabei zu seinen eigenen Strategien der Lösung seiner speziellen Probleme kommen, die sich aber nur schwer in einem einführenden Artikel - wie diesem hier - behandeln lassen.

Abbildung 1

```
#= Programm zur Erzeugung eines Wortformenindex
  mit Referenzen
#rv, text, -std-, mo=+, lo=+, par=*
str      1
>1z     , .?;:!"()
tr       |>1|
xs1     |ä|ae|ü|ue|ö|oe|ß|ss|
ssl      20
*eof
#so, -std-, -std-, 17+20, +
#ra, -std-, , mo=+, lo=+, par=*, prot=-std-
ssl      20
dr       3 0  18 2
ger      fx
rfl      6 3
nrt     1 |.|
rtw     1 1
*eof
#dr, -std-, fx
```

Abbildung 2

Auszug aus einem Wortindex

| | | |
|----------------|----------------|---------------------|
| aber 1.5 | gearbeitet 2.4 | schwach 2.5 |
| angelernt 1.4 | griff 3.1 | sie 3.2 3.3 |
| betrogen 1.3 | hast 1.3 1.4 | Siehe 1.2 |
| bin 2.5 | ich 2.5 | Simson 1.1 2.1 3.5 |
| damit 3.2 | ihm 3.3 | sind 3.4 |
| dann 2.5 3.3 | ihn 3.2 | so 2.5 |
| Delila 1.1 3.1 | ihr 2.1 | Stricke 3.1 3.6 |
| denen 2.4 | Jetzt 1.5 | Stricken 2.3 |
| dich 1.6 | kann 1.6 | über 3.4 |
| die 3.6 | man 1.6 2.2 | und 1.4 2.1 3.2 3.6 |
| ... | ... | ... |

Abbildung 3

#= Programm zur Erzeugung einer Häufigkeitsliste
der Wortformen

```
#rv, text, -std-, mo=-, lo=+, par=*
str      1
>lz     , .?;:!"()
tr      |>1|
ssl     20
xs1     |ä|ae|ü|ue|ö|oe|ß|ss|
*eof
#so, -std-, -std-, 1+20, lo=+
#ra, -std-, liste, mo=-, lo=+, *, prot=-
ssl     20
ah      1
vah     |[|
nah     | |
rh      4
rhd     1
vrh     |= |
nrh     |%|
*eof
```

Abbildung 4

Inhalt der Datei namens LISTE

```

1.1  |aber [1 = 1.3%]
1.2  |angelogen [1 = 1.3%]
1.3  |betrogen [1 = 1.3%]
1.4  |bin [1 = 1.3%]
1.5  |damit [1 = 1.3%]
1.6  |dann [2 = 2.7%]
1.7  |Delila [2 = 2.7%]
. . .
1.30 |man [2 = 2.7%]
1.31 |Mensch [1 = 1.3%]
1.32 |mich [3 = 4.0%]
1.33 |mir [1 = 1.3%]
1.34 |mit [2 = 2.7%]
1.35 |neue [1 = 1.3%]
1.36 |neuen [1 = 1.3%]
1.37 |nicht [1 = 1.3%]
1.38 |noch [1 = 1.3%]
1.39 |Philister [1 = 1.3%]
1.40 |sagte [3 = 4.0%]
1.41 |schwach [1 = 1.3%]
1.42 |sie [2 = 2.7%]
1.43 |Siehe [1 = 1.3%]
1.44 |Simson [3 = 4.0%]
. . .

```

Abbildung 5

```

#= Programm für ein Register der im Text
markierten Begriffe
#rv, umbdat, -std-, mo=+, lo=+, par=*
ea      |@i [s|@i+ |
ee      |@i ]|@i-|
umd     2
orf     *
irl     6
xs1     |ä|ae|ö|oe|ü|ue|ß|ss|
ssl     20
*eof
#so, -std-, -std-, 9+20, lo=+
#ra, -std-, mo=+, lo=+, *, prot=-std-
ssl     20
irl     6
ger     fx
rff     1
f1      |f|
zrp    1  |-|
*eof

```

Abbildung 6

```

#= Programm für einen KWIC-Index der Titelstichwörter
#rv,bib, -std-, mo=kwic, lo=+, par=*, dat=-std-
aa      |&a|
        Anfang/Ende Registereintrag = Kontext
ea      |&t|
ee      |&|
        Trennzeichen der Schlüsselwörter
swt     | |,|.!?|;|:|!|(|)|
        Anfang/Ende der Referenz
rfa     |&a |
rfe     |&|
        Austauschen der Umlaute im Sortierschl. für Schlüsselwort
xs1     |ä|ae|ö|oe|ü|ue|ß|ss|
        Austauschen der Umlaute im Sortierschl. für Kontext
xs4     |ä|ae|ö|oe|ü|ue|ß|ss|
        Austauschen der Umlaute im Sortierschl. für Referenz
xs7     |ä|ae|ö|oe|ü|ue|ß|ss|
        Schlüsselw.   Kontext   Referenz
ssl     20 0 0           20 0 0           20 0 0
        Referenzlänge
rfl     20
irl     20
*eof
#so,-std-,-std-,23+60,+
#ra,-std-, mo=kwic, lo=+, pat=*, dat=-std-, prot=-std-
irl     20
rfl     20
ssl     60
ger     xx
dr      1 5 75
        Auszeichnung und Position der Schlüsselwörter
vsw     | #1+|
nsw     |#1-|
swp     40
        Abschluß Referenzteil
nrf     1 |:|
        Position und Einleitung Kontext
rtp     1 22
vk      |-|
        Leerzeilen und Position bei Beginn eines neuen Schl.worts
dre     1 2 2 1 42 4 1
*eof
#dr,-std-,fx
    
```

Abbildung 7

Auszug aus dem KWIC-Index der Titelstichwörter

| | | | |
|--------------------|----|-----------------|--|
| Jaeger, Jakob: | - | Württemberg, | <u>Baden</u> <u>Baden und Hohenzollern</u> |
| Auer, Alfons: | - | Die Geschichte | <u>der</u> <u>der Ethik</u> |
| Denoix, René: | - | Geschichte | <u>der</u> <u>der Stadt Trier</u> |
| Zenger, Zacharias: | - | zur Geschichte | <u>des</u> <u>des saarlands</u> |
| Auer, Alfons: | - | | <u>Die</u> <u>Die Geschichte der Ethik</u> |
| Denoix, René: | -: | Einführung in | <u>die</u> <u>die Methode</u> |
| Denoix, René: | - | Geschichte: | <u>Einführung</u> <u>Einführung in die Methode</u> |
| Auer, Alfons: | - | Geschichte der | <u>Ethik</u> <u>Ethik</u> |
| Auer, Alfons: | - | Die | <u>Geschichte</u> <u>Geschichte der Ethik</u> |
| Denoix, René: | - | | <u>Geschichte</u> <u>Geschichte der Stadt Trier</u> |
| Zenger, Zacharias: | - | ersuchungen zur | <u>Geschichte</u> <u>Geschichte des Saarlands</u> |
| Denoix, René: | - | | <u>Geschichte:</u> <u>Einführung in</u> |
| Zenger, Zacharias: | - | | <u>Hessen</u> <u>Hessen und seine Wirtschaft</u> |
| Jaeger, Jakob: | - | berg, Baden und | <u>Hohenzollern</u> <u>Hohenzollern</u> |
| Denoix, René: | - | hte: Einführung | <u>in</u> <u>in die Methode</u> |
| Jäger, Jakob: | - | Leben und Werk | <u>Karls</u> <u>Karls V.</u> |
| Jäger, Jakob: | - | | <u>Leben</u> <u>Leben und Werk Karls V.</u> |
| . . . | | | |