

# Die $\text{\LaTeX}$ -Dokumenten-Stil-Option $\text{\BibA}^{\text{L}}_{\text{S}}$

TIMO BAUMANN

Version 1.3 vom 28.Mar.1998

---

## Zusammenfassung

In den meisten geisteswissenschaftlichen Disziplinen ist es üblich, die Überprüfbarkeit von Aussagen mittels Angaben in Fußnoten sicherzustellen. Wird ein bestimmtes Schriftwerk zum ersten Mal im Text erwähnt, ist es Standard, in einer Fußnote darüber eine vollständige bibliographische Angabe zu machen, also Vorname und Name des Autors, sowie Titel, Erscheinungsort und Erscheinungsjahr des Werkes anzugeben. Bei nochmaligem Verweis auf ein schon in dieser Weise erwähntes Werk wird nur noch der Autorenname und ein signifikanter Teil aus dem Titel genannt. Zuletzt sind noch alphabetisch geordnete Listen der verwendeten Schriftwerke abzdrukken. Solche Listen stimmen im wesentlichen mit den Angaben in den Fußnoten überein, wie sie schon bei den erwähnten „Erstzitat“ zu machen waren. Ein ständiger Abgleich dieser Listen mit den in Fußnoten gemachten Angaben ist aber äußerst mühsam — von dem häufigen Fehler ganz zu schweigen, daß Erstzitate und bibliographische Listen unbeabsichtigt Unterschiede aufweisen. Die  $\text{\LaTeX}$ -Befehle  $\text{\index}$  und  $\text{\glossary}$  lösen das Problem nur schlecht, besonders, da sie in Fußnoten Schwierigkeiten mit längeren Einträgen machen.

$\text{\LaTeX}$ -Anwender kennen die Lösung für ein vergleichbares Problem bezüglich des Inhaltsverzeichnisses. Hier wird einem Gliederungsbehl — etwa  $\text{\section}$  — als Argument Text für eine Überschrift übergeben, die sowohl am Ort des Befehls, als auch im Inhaltsverzeichnis ausgedruckt wird. Das Problem hier unterscheidet sich von dem des Inhaltsverzeichnisses nur in der Notwendigkeit der alphabetischen Reihenfolge. Wird die Dokumenten-Stil-Option  $\text{\bibarts.sty}$  in das  $\text{\LaTeX}$ -File eingeladen, steht der Befehl  $\text{\makebar}$  für den Vorspann des  $\text{\LaTeX}$ -Files zur Verfügung. Dieser Befehl legt eine Bibliographie-Datei mit dem Suffix  $\text{\.bar}$  an. Befehle namens  $\text{\vli}$  und  $\text{\vqu}$  drucken den Text, der ihnen in Argumenten übergeben wird, am Ort des Befehls aus, schreiben diesen Text aber auch in die Bibliographie-Datei. Diese Datei kann danach mit dem für dieses Sortier-Problem geschriebenen Programm  $\text{\gbibsort}$  bearbeitet werden. Die in einer Datei mit Suffix  $\text{\.phy}$  sortiert abgelegten Angaben können mit Befehlen ausgedruckt werden, die in  $\text{\bibarts.sty}$  bereitgestellt werden. Um diese Grundidee herum fügt sich noch einiges passendes Beiwerk.

---

## Inhaltsverzeichnis

<b>1</b>	<b>Änderungen gegenüber Vers. 1 und Vers. 1.1</b>	<b>2</b>
<b>2</b>	<b>Änderungen gegenüber Vers. 1.2 und Vers. 1.2a</b>	<b>3</b>
<b>3</b>	<b>Vorstellung der Optionen</b>	<b>4</b>
3.1	Die bibliographischen Listen . . . . .	4
3.1.1	Das Bibliographie-File . . . . .	4
3.1.2	Die Literaturangaben . . . . .	6
3.1.3	Die Quellenangaben . . . . .	8
3.1.4	Das Kurzzitate-Verzeichnis . . . . .	9
3.2	Die Verweise . . . . .	11
<b>4</b>	<b>Beispiele</b>	<b>13</b>
<b>5</b>	<b>Grenzen der Mächtigkeit von <math>\text{\BibA}^{\text{F}}\text{\S}</math></b>	<b>17</b>

---

## 1 Änderungen gegenüber Vers. 1 und Vers. 1.1

Für diesen Abschnitt werden sich (abgesehen vom zweiten Punkt) nur diejenigen interessieren, die  $\text{\BibA}^{\text{F}}\text{\S}$  schon in einer älteren Version kennen. Diese neue Version ist teilweise mit den älteren inkompatibel. Die neue Anwendungsweise kann aus dem übernächsten Abschnitt entnommen werden.

- Den Befehlen `\vli`, `\vqu`, `\addtovli`, `\addtovqu`, `\printonlyvli` und `\printonlyvqu` müssen ab jetzt drei Argumente übergeben werden, statt bisher einem.
- Bisher gab es zwei Style-Files namens `bibarts.sty` und `gbibarts.sty`; das zweitgenannte mußte verwendet werden, wenn auch `german.sty` verwendet wurde. Dies ist nichtmehr der Fall; es gibt jetzt nur noch ein Style-File mit dem Namen `bibarts.sty`. Es erkennt jetzt selbständig, ob vor `bibarts` das Paket `german` eingeladen wurde.<sup>1</sup> Nach wie vor muß `german` vor `bibarts` eingeladen werden.<sup>2</sup>
- Der Befehl `\makereportbar` ist nicht mehr notwendig und auch nicht mehr definiert. Der Befehl `\makebar` kann jetzt selbständig die Unterscheidungen vornehmen, die vorher der Anwender machen mußte, indem er entweder `\makebar` oder `\makereportbar` schrieb.

---

<sup>1</sup>Und zwar anhand der Überprüfung, ob `\newumlaut` ein bekannter Befehl ist.

<sup>2</sup>D.h. bei  $\text{\LaTeX}2\text{\epsilon}$ : `\usepackage{german} \usepackage{bibarts}`, bzw. unter  $\text{\LaTeX} 2.09$ : `\documentstyle{...german,bibarts}`.

- Der Anwender soll ab jetzt der Literaturliste, dem Quellenverzeichnis und dem Stichwortverzeichnis selber eine Überschrift geben. Die früher verwendeten `\section`-Befehle stehen in `bibarts.sty` noch hinter Auskommentierungen, die der Anwender ggf. entfernen kann.<sup>3</sup>
- Die Datei `bibarts.sty` enthält die Definitionen von `\@footnotetext` aus L<sup>A</sup>T<sub>E</sub>X 2.09 und L<sup>A</sup>T<sub>E</sub>X 2e, die hier einen Zusatz erhalten haben, der bei der Übersetzung abprüfbar macht, ob man sich gerade innerhalb oder außerhalb einer Fußnote aufhält. B<sub>I</sub>A<sub>T</sub>S entscheidet sich selbständig für die richtige L<sup>A</sup>T<sub>E</sub>X-Version.<sup>4</sup>
- `\makebar` kann auskommentiert werden, wenn gerade keine Notwendigkeit besteht, über aktuelle bibliographische Listen zu verfügen.
- Folgendes betrifft nur die Dokumentengrößen `report` und `book`: B<sub>I</sub>A<sub>T</sub>S übernimmt jetzt, wenn die Gliederungsbefehle `\part` und `\chapter` verwendet werden, in sein Stichwortverzeichnis den Gliederungsnamen, der tatsächlich am Ort des `\bib`-Befehls aktuell war. Denn `\partname` und `\chaptername` können sich im Laufe des Textes ändern, z.B. von `>Teil<` in `>Anlagen<` und von `>Kapitel<` in `>Anhang<`.
- In den Index werden Teil-, Kapitel-, Seiten- und Fußnotenzähler jetzt so übernommen, wie sie im Text stehen, also z.B. `roman` oder `Alph`.<sup>5</sup>

## 2 Änderungen gegenüber Vers. 1.2 und Vers. 1.2a

- Ein `Index-Style-File` in der Art von `gbibarts.ist` ist nun notwendige Voraussetzung zum Erzeugen eines B<sub>I</sub>A<sub>T</sub>S-Index.<sup>6</sup> Als Gegenleistung sind die bisherigen Probleme mit dem Index gelöst, die auftraten, wenn `\bib`-Befehle außerhalb von Fußnoten standen.
- Dies betrifft ebenfalls den Index. Die Eigenschaft des Befehls `\bib` hat jetzt der Befehl `\addtobib`, der bisher nicht existierte. `\addtobib` schreibt nur in den Index, `\bib` schreibt ebenfalls in den Index, druckt jetzt aber auch zusätzlich das ihm übergebene Argument einfach aus. Diese Änderung wurde vorgenommen, um einheitliche Verhaltensweisen von (1) `\vli`, `\vqu` und `\bib`, sowie auch (2) `\addtovli`, `\addtovqu` und `\addtobib` zu erreichen.
- Dies betrifft Literatur- und Quellen-Zugänge. `gbibsort` kann jetzt mit der Option `-k` gestartet werden. Falls von *einem* Autor mehrere Werke

---

<sup>3</sup>Ich habe darauf verzichtet, anhand einer Abprüfung darüber, ob die Zähler `part` oder `chapter` oder keiner von beiden verwendet wurde, zu entscheiden, welche Größe der dazugehörige Gliederungsbefehl haben müßte. Solche Definitionen vorwegzunehmen ist nämlich für die Anwender kein Service, die z.B. unter `book` eben keine Überschrift in Größe `chapter` haben wollen und sich dann überlegen müssen, wie sie die Voreinstellung loswerden sollen.

<sup>4</sup>Und zwar anhand der Überprüfung, ob `\LaTeX2e` ein bekannter Befehl ist.

<sup>5</sup>Aber nicht `Roman`; solche Einträge werden von `MAKEINDEX` zurückgewiesen!

<sup>6</sup>Für DOS: `gbib2e.bat` bzw. `gbib209.bat` suchen zunächst im aktuellen Verzeichnis und dann in `C:\EMTEX\IDXSTYLE` nach einer Datei `gbibarts.ist`.

zitiert werden, wird der entsprechende Vor- und Nachname auf der jeweiligen Liste nur einmal ausgedruckt. Alle weiteren Nennungen werden durch  $\sim$  ersetzt. Alternativ kann — das muß aber durchgehend geschehen — unmittelbar nach dem Vornamen `\female` oder `\male` stehen, um statt  $\sim$  geschlechts-spezifisch `Dies.` oder `Ders.` ausgedruckt zu bekommen. Der Vorname ist das erste Argument, das `\vli`, `\vqu`, `\addtovli` oder `\addtovqu` übergeben werden soll.

- Dies betrifft nur Literatur- und Quellen-Zugänge. `BiArTs` verfügt jetzt über ein Instrument, um die Sortier-Reihenfolge der bibliographischen Listen beeinflussen zu können. In einem der je drei Argumente der Befehle `\vli`, `\vqu`, `\addtovli` und `\addtovqu` kann jetzt der Befehl `\sort{RELEVANT}` verwendet werden. `gbibsort` sortiert den Eintrag so, als ob nur `RELEVANT` zwischen vorausgehendem und nachfolgendem Text im Argument steht; `RELEVANT` wird aber nirgends ausgedruckt. Zu beachten ist, daß erstes und zweites Argument (Vor- und Nachname) ausgetauscht werden, bevor der Eintrag in das Bibliographie-File geht. Innerhalb des Arguments des Befehls `\sort{ARG}` darf keine Konstruktion in geschweiften Klammern stehen. `\sort` ist für die Argumente von `\bib` und `\addtobib` ungeeignet.<sup>7</sup>

## 3 Vorstellung der Optionen

### 3.1 Die bibliographischen Listen

#### 3.1.1 Das Bibliographie-File

Im folgenden werden bibliographische Angaben in Vorname, Name und restliche Angaben getrennt. Rest ist dabei Titel, Erscheinungsort und Erscheinungsjahr. Während die bibliographische Angabe in einer Fußnote die Reihenfolge `>Vorname, Name, Rest<` aufweisen soll, wird in der bibliographischen Liste die Reihenfolge `>Name, Vorname, Rest<` erwartet. `VORNAME`, `NAME` und `REST` stehen ab jetzt symbolhaft für die drei Argumente, die nachfolgend beschriebenen Befehlen übergeben werden müssen.

Das Erzeugen der nachfolgend beschriebenen Listen setzt voraus, daß `LATEX` während der Übersetzung eine Datei angefertigt, in der die bibliographischen Daten gesammelt werden. `BiArTs` stellt Befehle bereit, die das Anfertigen einer solchen Datei ermöglichen. Diese Datei hat den Namen der Datei, die mit `LATEX` bearbeitet wird und das Suffix `.bar`. Das zu `bibarts.tex` gehörende Literaturdaten-File heißt also `bibarts.bar`. Sollen ein Literaturverzeichnis oder zusätzlich auch ein Quellenverzeichnis zum Ausdruck bereitgestellt werden, muß die Datei `NAME.bar` sortiert werden auf eine Datei `NAME.phy`. Die erzeugte Datei muß also das Suffix `.phy` aufweisen. Im Falle von `bibarts.tex` muß der Inhalt von `bibarts.bar` nach Zeilen sortiert unter dem Namen `bibarts.phy`

<sup>7</sup>Die Sortier-Reihenfolge im Index ist aber `MAKEINDEX`-spezifisch mit einem `@` in der Art `\addtobib{RELEVANT@TEXT}` zu erreichen. Das Zeichen `@` wird im Argument von `\bib` ausgedruckt; deswegen sollte `\bib` zerlegt werden in seine beiden Bestandteile `\printonlybib{TEXT}` und `\addtobib{RELEVANT@TEXT}`. `gbibsort` hat kein Maskierungszeichen; `{RELEVANT}@` — wozu `\sort` expandiert — wird „verschluckt“, `@` ansonsten ausgedruckt.

abgespeichert werden. Das Sortierprogramm `gbibsort` löst das Sortierproblem. (Für DOS ist die ausführbare Datei `gbibsort.exe` im Paket `B1bArts` mit enthalten. Anwender anderer Betriebssysteme müssen sich die Quell-Datei `gbibsort.c` in eine ausführbare Datei übersetzen.<sup>8)</sup> Der Befehl zum Sortieren lautet allgemein `gbibsort QUELLE ZIEL`, also für `bibarts.tex`:

```
gbibsort bibarts.bar bibarts.phy
```

`B1bArts` stellt Befehle bereit, die die erwünschten Daten aus dieser alphabetisch sortierten Datei an einer bestimmten Textstelle ausdrucken. Sie werden unten beschrieben.

- Für alle Listen gemeinsam gilt aber folgendes: Ein Befehl im Vorspann<sup>9</sup> muß die Datei `NAME.bar` anlegen. Dieser Befehl heißt `\makebar`.

<code>\makebar</code>	Bereitet die Datei vor, die die bibliographischen Daten aufnehmen soll.
-----------------------	---

Das Anfertigen dieser Datei kann möglicherweise die Übersetzung eines Textes mit `LATEX` unerwünscht verlangsamen. Deswegen kann der Befehl auch auskommentiert (ein `%` davorgeschrieben) werden, wenn eine Übersetzung nicht mit dem Ziel erfolgt, die bibliographischen Daten zu aktualisieren, sondern anderer Änderungen wegen erfolgt. Dann bleibt alles übrige gleich, nur wird nicht in das Literaturdaten-File geschrieben. Die bibliographischen Listen sind lediglich nicht auf dem neuesten Stand, wenn sie ausgedruckt werden. (Für die letzten beiden Übersetzungen muß der Befehl `\makebar` natürlich wirksam sein, also ohne vorangestelltes Kommentarzeichen (`%`) dastehen!)

Zuletzt noch ein Hinweis auf die Möglichkeiten von `gbibsort`: Es versteht in der Kommando-Zeile die Option `-K`, was `author killing` andeuten soll. Falls ein Autor mehrfach im Literaturverzeichnis (oder Quellenverzeichnis) erscheint, bewirkt `-K`, daß Name und Vorname des Autors nur das erste Mal genannt und danach durch `~` ersetzt werden. Falls das `author killing` benutzt werden soll, gilt: Die Befehlszeile für die Bearbeitung eines bibliographischen Files zu einer Datei `text.tex` lautet explizit

```
gbibsort -k text.bar text.phy
```

- Das Ersetzungszeichen `~` wird durch `\killname` erzeugt.

<code>\renewcommand{\killname}{---}</code>	Ändert das Ersetzungszeichen in <code>---</code> .
--	--

Oft möchte man aber geschlechtsspezifische Ersetzungen haben. Üblich sind `Dies.` und `Ders.`. Unten werden gleich die Befehle `\vli`, `\vqu`, `\addtovli` und `\addtovqu` erklärt werden. Sie sollen drei Argumente übergeben bekommen (`\vli{ARG1}{ARG2}{ARG3}`). Das erste soll der Vorname eines Autors sein.

<sup>8</sup>`gbibsort.c` ist in ANSI-C geschrieben und läßt sich erfahrungsgemäß mit dem GNU-C-Compiler auch auf HPUX und PC-UNIX übersetzen.

<sup>9</sup>Mit dem Begriff Vorspann bezeichne ich in einem `LATEX`-File den Bereich nach dem Befehl `\documentclass` bzw. `\documentstyle` und vor dem Befehl `\begin{document}`.

- Unmittelbar nach dem Vornamen im Voll-Zitat (vor `}`) kann `\female` oder `\male` stehen, wenn statt `~ Dies.` oder `Ders.` ausgedruckt werden soll.

<code>\vli{VORNAME\female}{NAME}{REST}</code>	Mehrfachnennungen von Autoren werden
<code>\vli{VORNAME\male}{NAME}{REST}</code>	durch <code>&gt;Dies.&lt;</code> und <code>&gt;Ders.&lt;</code> ersetzt.

- `Dies.` und `Ders.` werden durch `\femalename` und `\malename` definiert:

<code>\renewcommand{\malename}{Derselbe}</code>	Ändert <code>&gt;Ders.&lt;</code> in <code>&gt;Derselbe&lt;</code> .
---	--

Zuletzt können noch Fälle auftreten, die eine Änderung der Sortier-Reihenfolge der Literaturliste oder des Quellenverzeichnisses erstrebenswert erscheinen lassen. Dies wird mittels eines Eintrags von `\sort` in eines der drei Argumente von `\vli`, `\vqu`, `\addtovli` oder `\addtovqu` erreicht. `\sort{ARG}` wird wie `ARG` sortiert, aber nichts ausgedruckt.

- Ein für `gbibsort` sortier-relevantes, aber nicht auszudruckendes Wort ist das Argument von `\sort`:

<code>\sort{RELEVANT}</code>	Greift in die Reihenfolge im Literatur- oder Quellen-Verzeichnis ein. Es wird wie <code>RELEVANT</code> sortiert, <code>RELEVANT</code> wird aber nicht ausgedruckt.
------------------------------	--

Unter den beiden nächsten Überschriften werden das Literaturverzeichnis und das Quellenverzeichnis vorgestellt, wie sie mit `BibATS` erzeugt werden können.

### 3.1.2 Die Literaturangaben

Literaturangaben der nachfolgenden Art sind in allen geisteswissenschaftlichen Disziplinen und auch weit darüber hinaus üblich.

- Der Befehl `\vli` druckt die drei ihm übergebenen Argumente an dem Ort aus, an dem `\vli` steht, und übergibt diese Argumente auch dem Literaturdaten-File. `\vli` steht für volle Literaturangabe.

<code>\vli{VORNAME}{NAME}{REST}</code>	<code>VORNAME, <u>NAME</u>: REST</code> wird ausgedruckt.
	<code><u>NAME</u>, VORNAME: REST</code> wird im Literaturverzeichnis ausgedruckt.

Der Befehl `\vli` setzt sich aus den beiden Teilbefehlen `\printonlyvli` und `\addtovli` zusammen, die auch statt `\vli` einzeln verwendet werden können, falls Unterschiede zwischen den Angaben in Fußnoten und dem Quellenverzeichnis erwünscht sind. Ein möglicher Unterschied ist, daß der Titel einer Reihe, in der ein Buchtitel erschienen ist, im Literaturverzeichnis angegeben werden soll, nicht aber in der Fußnote. Dann ist `REST` in beiden Fällen unterschiedlich.

- Der Befehl `\printonlyvli` druckt die drei ihm übergebenen Argumente wie `\vli` an dem Ort aus, an dem der Befehl `\printonlyvli` steht. Es erfolgt aber keine Wiedergabe im Literaturverzeichnis.

<code>\printonlyvli{VORNAME}{NAME}{REST}</code>	VORNAME, <u>NAME</u> : REST wird ausgedruckt. Es erfolgt kein Eintrag ins Literaturverzeichnis.
---	--

- Auf den Befehl `\addtovli` hin erfolgt lediglich ein Eintrag ins Literaturverzeichnis.

<code>\addtovli{VORNAME}{NAME}{REST}</code>	<u>NAME</u> , VORNAME: REST wird dem Literaturdaten-File übergeben. Es erfolgt kein Ausdruck am Ort des Befehls.
---	---

- Jetzt muß das Literaturverzeichnis noch ausgedruckt werden. Die schon beschriebene Prozedur muß zuvor noch durchlaufen werden, um aus der Datei `FILE.bar` eine alphabetisch sortierte Datei `FILE.phy` zu erzeugen. Dann kann mittels des Befehls `\printvli` das Literaturverzeichnis ausgedruckt werden. Unmittelbar vor `\printvli` kann der Anwender, wenn er möchte, einen Gliederungsbefehl seiner Wahl setzen. Möglich ist:

<code>\section{Literaturverzeichnis}</code> <code>\printvli</code>	Druckt die Literaturliste unter der Überschrift <code>&gt;Literaturverzeichnis&lt;</code> aus.
---	--

- Die Befehle `\vli` und `\addtovli` notieren in das Bibliographie-File auch Informationen über den Ort, an dem sie stehen: Die Seitenzahl, weiter auch, ob sie in einer Fußnote stehen — und wenn ja: die Nummer der Fußnote. So gibt es noch einen zweiten Befehl, um das Literaturverzeichnis ausdrucken zu können. Der Befehl heißt `\printnumvli` und unterscheidet sich von `\printvli` dadurch, daß hinter jedem Zugang die Seitenzahl steht, auf der der Zugang stattfand. Falls der Zugang in einer Fußnote erfolgte, wird die Nummer der Fußnote als Exponent zur Seitenzahl ausgedruckt. Die Gliederungsüberschrift zu setzen ist wieder Sache des Anwenders — Beispiel: eine `\subsection`.

<code>\subsection{Literaturverzeichnis}</code> <code>\printnumvli</code>	Druckt Literaturliste aus, derart, daß auf den Ort zurückverwiesen wird, an dem das Erstzitat steht.
---	--

Natürlich kann `\printnumvli` auch während der Entstehung eines Textes verwendet werden, um bei Korrekturen, die an der Literaturliste ansetzen, schneller in den Text zurückzufinden. Vor der letzten Übersetzung mit  $\text{\LaTeX}$  kann dieser Befehl dann durch `\printvli` ersetzt werden, falls die Rückverweise nicht erwünscht sind.

### 3.1.3 Die Quellenangaben

In der Geschichtswissenschaft ist es darüberhinaus üblich, zwischen Literatur und Quellen zu trennen. Die bibliographischen Angaben unterscheiden sich zwar nicht qualitativ voneinander, jedoch sollen zwei bibliographische Listen angefertigt werden: Ein Quellenverzeichnis und ein Literaturverzeichnis. (Quellen sind Schriften aus der Zeit, über die geschrieben wird, während mit Literatur später verfaßte Titel bezeichnet werden, die die behandelte Epoche anhand der Quellen beschreiben.) Auch hierfür hält  $\LaTeX$  eine Option offen.

- Der Befehl  $\backslash\text{vqu}$  druckt die drei ihm übergebenen Argumente sowohl an dem Ort aus, an dem  $\backslash\text{vqu}$  steht, als auch in das Quellen-Verzeichnis.  $\backslash\text{vqu}$  steht für volle Quellenangabe.

$\backslash\text{vqu}\{\text{VORNAME}\}\{\text{NAME}\}\{\text{REST}\}$	VORNAME, <u>NAME</u> : REST wird ausgedruckt. <u>NAME</u> , VORNAME: REST wird im Quellenverzeichnis ausgedruckt.
--	--

Der Befehl  $\backslash\text{vqu}$  setzt sich aus den beiden Teilbefehlen  $\backslash\text{printonlyvqu}$  und  $\backslash\text{addtovqu}$  zusammen, die auch statt  $\backslash\text{vqu}$  einzeln verwendet werden können, falls Unterschiede zwischen den Angaben in Fußnoten und dem Quellenverzeichnis erwünscht sind. Ein möglicher Unterschied ist, daß der Titel einer Reihe, in der die Quellenedition erschienen ist, im Quellenverzeichnis angegeben werden soll, nicht aber in der Fußnote. Dann ist REST in beiden Fällen unterschiedlich.

- Der Befehl  $\backslash\text{printonlyvqu}$  druckt die drei ihm übergebenen Argumente an dem Ort aus, an dem der Befehl  $\backslash\text{printonlyvqu}$  steht. Es erfolgt aber keine Wiedergabe im Quellenverzeichnis.

$\backslash\text{printonlyvqu}\{\text{VORNAME}\}\{\text{NAME}\}\{\text{REST}\}$	VORNAME, <u>NAME</u> : REST wird ausgedruckt. Es erfolgt kein Eintrag ins Quellenverzeichnis.
---	--

- Auf den Befehl  $\backslash\text{addtovqu}$  hin erfolgt lediglich ein Eintrag in das Quellenverzeichnis.

$\backslash\text{addtovqu}\{\text{VORNAME}\}\{\text{NAME}\}\{\text{REST}\}$	<u>NAME</u> , VORNAME: REST wird im Quellenverzeichnis ausgedruckt. Es erfolgt kein Ausdruck am Ort des Befehls.
---	---

- Jetzt muß das Quellenverzeichnis noch tatsächlich ausgedruckt werden. Die schon beschriebene Prozedur muß durchlaufen werden, um aus der Datei `FILE.bar` eine alphabetisch sortierte Datei `FILE.phy` zu erzeugen. Die Daten für die Literaturliste und das Quellenverzeichnis entstammen

der selben Datei. Aus ihr drucken die Befehle `\printvli`, `\printnumvli`, `\printvqu` und `\printnumvqu` selektiv spezifische Daten aus. Mit dem Befehl `\printvqu` kann das Quellenverzeichnis ausgedruckt werden. Unmittelbar vor den Befehl `\printvqu` kann der Anwender, wenn er möchte, einen Gliederungsbefehl seiner Wahl setzen. Möglich ist:

<code>\section{Quellenverzeichnis}</code> <code>\printvqu</code>	Druckt das Quellenverzeichnis unter der Überschrift <code>&gt;Quellenverzeichnis&lt;</code> aus.
---	--

- Die Befehle `\vqu` und `\addtovqu` notieren in das Bibliographie-File auch Informationen über den Ort, an dem sie stehen: Die Seitenzahl, dann: ob sie in einer Fußnote stehen — und wenn ja: die Nummer der Fußnote. So gibt es noch einen zweiten Befehl, um das Quellenverzeichnis ausdrucken zu können. Der Befehl heißt `\printnumvqu` und unterscheidet sich von `\printvqu` dadurch, daß hinter jedem Eintrag die Seitenzahl steht, auf der der Zugang stattfand. Falls der Zugang in einer Fußnote erfolgte, wird die Nummer der Fußnote als Exponent zur Seitenzahl ausgedruckt. Als Gliederungsüberschrift wurde eine `\subsection` gewählt.

<code>\subsection{Quellenverzeichnis}</code> <code>\printnumvqu</code>	Druckt Quellenverzeichnis aus, derart, daß auf den Ort zurückverwiesen wird, an dem das Erstzitat steht.
---	--

### 3.1.4 Das Kurzzitate-Verzeichnis

- Ähnlich den Befehlen `\index` oder `\glossary` stellt  $\LaTeX$  den Befehl `\bib` zur Verfügung. Als Argument muß diesem Befehl ein Stichwort übergeben werden. `STICHWORT` kann speziell ein Kurz-Zitat (Marx, Kapital) sein; das Schriftwerk, auf das sich `STICHWORT` bezieht, soll vorher schon vollständig zitiert worden sein. Ein Kurz-Zitat besteht aus dem Namen des Autors und (durch Komma getrennt) einem oder wenigen signifikanten Worten aus dem Titel, die eine eindeutige Feststellung der vollständigen bibliographischen Daten anhand des Literatur- oder Quellen-Verzeichnisses möglich machen. Gegenüber dem  $\LaTeX$ -Befehl `\index` wird das Argument von `\bib` erweitert indexiert: Hier verweisen die Stichworte nämlich nicht nur auf die Seite, sondern auch auf die Nummer der Fußnote zurück, falls der Befehl in einer Fußnote steht. Diese Informationen gehen ebenfalls in das `FILE.bar`. Auch wenn die zuvor vorgestellten Befehle zum Erzeugen einer Literaturliste oder eines Quellenverzeichnisses nicht benutzt werden, so muß das `FILE.bar` dennoch durch den schon beschriebenen Befehl `\makebar` bereitgestellt werden.

<code>\bib{STICHWORT}</code>	Stellt einen Eintrag für das Stichwort-Verzeichnis zur Verfügung. Der Befehl druckt <code>STICHWORT</code> aus.
------------------------------	---

- Falls STICHWORT nur im Index erscheinen soll, steht neu der Befehl `\addtobib` zur Verfügung.

<code>\addtobib{STICHWORT}</code>	Stellt einen Eintrag für das Stichwort-Verzeichnis zur Verfügung. Der Befehl druckt nichts aus an dem Ort, an dem er steht.
-----------------------------------	---

Das mittels des Befehls `\bib` gefütterte `FILE.bar` muß mit dem Programm MAKEINDEX bearbeitet werden. Zugänge der Befehle, die zum Erzeugen der Literatur- und Quellen-Verzeichnisse ebenfalls vorhanden sein mögen, wird der Stichwortprozessor MAKEINDEX als fehlerhaft einstufen und nicht verwenden. Das ist so erwünscht. Um eine Datei `FILE.bar` zu bearbeiten, muß unter DOS

```
makeindx FILE.bar
```

befohlen werden. Unter UNIX lautet der Aufruf

```
makeindex FILE.bar
```

Für deutsche Texte muß die Option `-g` übergeben werden, um die Sortierreihenfolge auf `german` umzuschalten. Das genügt aber noch nicht: In Grundeinstellung ist " das Maskierungszeichen von MAKEINDEX. Maskierungszeichen dienen dazu, `!`, `|` und `@` via `"!`, `"|` und `"@` ausdrucken lassen zu können. Unmaskiert stellen sie Befehle dar, die das Verhalten von MAKEINDEX steuern. " wird aber für die deutschen Umlaute gebraucht, wie mit `german.sty` definiert. Um das Maskierungszeichen zu ändern, muß ein Index-Style-File nach der Option `-s` in der Befehlszeile stehen. In diesem Index-Style-File kann eine Zeile

```
quote '='
```

das Maskierungszeichen in `=` umbenennen. Erst dann wird die Option `-g` akzeptiert. Dazu wurde zu `BiATS` die Datei `gbibarts.ist` beigefügt. Der restliche Inhalt dieser Datei ist sprachunabhängig für `BiATS` notwendig. (In `\bib` ist `|` übrigens `i m m e r` schon besetzt!<sup>10</sup>) — Unter DOS lautet die Befehlszeile des bei der Bearbeitung von `bibarts.tex` erzeugten Files `gbibarts.bar` also:

```
makeindx -s gbibarts.ist -g bibarts.bar
```

- MAKEINDEX fertigt eine Index-Datei `FILE.ind` an. Der Befehl `\printind` druckt diese Datei zweispaltig unter der Überschrift `Index` aus. Der Ausdruck unterscheidet sich vom normalen L<sup>A</sup>T<sub>E</sub>X-Index: An die Seitenzahlen ist als Exponent die Nummer der Fußnote hinzugefügt, falls der Zugang in einer Fußnote erfolgte.

<code>\printind</code>	Druckt das mit <code>\bib</code> gefütterte Stichwortverzeichnis.
------------------------	---

<sup>10</sup>Ebenso steht `!`, das in `\index` zweimal gebraucht werden darf, in `\bib` nur einmal zur Verfügung, wenn `chapter`-Überschriften, und garnicht, wenn darüber hinaus `part`-Überschriften benutzt werden. In `bibarts.tex` sind also zwei `!` erlaubt. (Bedeutung von `!` in Anm. 11).

- Falls jedes Zitat in das Kurzzitate-Verzeichnis aufgenommen werden soll, ist es sinnvoll, neben `\bib` einen Befehl zu besitzen, der (1) innerhalb von `\vli` oder `\vqu` im voraus ankündigt, wie der Titel kurzzitiert wird, (2) gleich einen Eintrag ins Kurzzitate-Verzeichnis macht und (3) hinter dem Eintrag auf der Literatur- oder Quellen-Liste in eckigen Klammern ausdrückt, wie kurz-zitiert wurde.

<code>\kurz{STICHWORT}</code>	Stellt einen Eintrag für das Stichwort-Verzeichnis zur Verfügung. Der Befehl druckt (im folgenden STICHWORT) aus. Falls <code>\kurz</code> am Ende des dritten Arguments von <code>\vli</code> , <code>\vqu</code> , <code>\addtovli</code> oder <code>\addtovqu</code> steht, wird [STICHWORT] am Ende des Eintrags auf der bibliographischen Liste stehen.
-------------------------------	--

Ergänzen möchte ich noch, daß der Befehl `\subitem` in den `FILES.ind` stehen wird, die von `MAKEINDEX` erzeugt werden, nachdem die Gliederungs-Überschrift `\chapter` zum ersten Mal verwendet wurde und darüber hinaus auch `\subsubitem`, falls `\part` verwendet wurde. Sie entsprechen Unter-Stichworten. Mit Befehlen der Art `\renewcommand{\subitem}{\hspace{.3em}}` kann erzwungen werden, daß Text nach dem Index-Haupteintrag der Reihe nach heruntergeschrieben wird. Zur Sicherheit sollte auf den `MAKEINDEX`-Befehl `!`, der Stichwort und Unterstichwort trennt, im Argument von `\bib` verzichtet werden.<sup>11</sup>

### 3.2 Die Verweise

Was in diesem Unterabschnitt zu Sprache kommt, hat nichts mit dem Bibliographie-File zu tun. „Verweis“ wird hier eine zweite Zitier-Konvention genannt, die manche Verlage für Aufsätze verwenden. Hier werden ebenso wie zuvor anlässlich des Erstzitats vollständige bibliographische Angaben gemacht. Wird auf einen Titel ein zweites Mal eingegangen, wird dieser ebenfalls kurz-zitiert, aber dahinter kommt ein Verweis auf die vollständige erste Angabe. Zu solchen Aufsätzen gehören keine bibliographischen Listen.

- Nach jedem Erstzitat kann `\bibmark{MARKE}` dazu verwendet werden, eine Marke zu erzeugen. Im Unterschied zu dem `LATEX`-Befehl `\label` wird aber ausgedruckt, wie künftig kurzzitiert wird. `MARKE` und Kurzzitat sind identisch.

<code>\bibmark{MARKE}</code>	Setzt die Referenz <code>MARKE</code> und druckt aus: im folgenden <code>MARKE</code>
------------------------------	---

<sup>11</sup>Zur Erzeugung von `\subitems` verwendet `BhAts` Ausrufezeichen, die für `MAKEINDEX` Befehlscharakter haben. Wenn z.B. `\index{Maier!Arbeit}` und `\index{Maier!Recht}` geschrieben wird, gibt es die beiden Unter-Einträge `Arbeit` und `Recht` unter dem Haupteintrag `Maier`. Maximal zwei `!` können verwendet werden. Damit scheidet die beschriebene Untergliederung mit `\bib` aus, wenn `\part`-Überschriften verwendet werden. (Siehe auch Anm. 10.)

- Mit dem Befehl `\bibref{MARKE}` kann man sich dann Seitenzahl und Fußnotennummer in einer übliche Konstruktion zurückgeben lassen.

<code>\bibref{MARKE}</code>	Druckt in Fußnote aus: MARKE (wie Anm. XX, S. YY), ansonsten: MARKE (siehe S. YY).
-----------------------------	---

MARKE kann dabei ohne weiteres ein Kurz-Zitat in der Art `Marx, Kapital` sein. Auch `B"uLow` funktioniert als Marke, denn Umlaute, ß, Akut, Zirkumflex und Gravis sind in einer Struktur namens `\@markprotect` geschützt. Bei hier nicht geschützten Befehlen müssen auszudruckender `TEXT` und Schlüsselbegriff `MARKE` voneinander getrennt werden.

- Der Befehl `\xbibmark{TEXT}{MARKE}` setzt die `MARKE` und druckt den `TEXT` aus.

<code>\xbibmark{TEXT}{MARKE}</code>	Setzt die Referenz <code>MARKE</code> und druckt aus: im folgenden <code>TEXT</code> .
-------------------------------------	---

- Mit dem Befehl `\xbibref{TEXT}{MARKE}` kann man sich dann Seitenzahl und Fußnotennummer von `MARKE` zurückgeben lassen.

<code>\xbibref{TEXT}{MARKE}</code>	Druckt in Fußnote aus: <code>TEXT</code> (wie Anm. XX, S. YY), ansonsten: <code>TEXT</code> (siehe S. YY).
------------------------------------	---

Alle vier zuletzt angesprochenen Befehle `\bibmark`-`\bibref` und `\xbibmark`-`\xbibref` sollen paarweise entweder beide in Fußnoten oder außerhalb von Fußnoten stehen. Für die Verweis-Schlüsselbegriffe `MARKE` aller vier Befehle ist das Makro `\@markprotect` zuständig. Die `\x...`-Befehle sind eigentlich nicht-mehr unbedingt notwendig, denn mit `\@markprotect` kann eine unliebsame Expansion von Befehls-Makros durch eine an dieser Stelle erwünschte Expansion ausgetauscht werden. Hauptsächlich aus Gründen der Kompatibilität mit älteren `ByATS`-Versionen sind die `\x...`-Befehle erhalten geblieben.

---

Auf der nächsten Seite werden Beispiele zu den bibliographischen Listen folgen. Hierzu ein Hinweis für DOS: `gbibsort.exe` enthält einen `go32`-Extender. Es ist oft notwendig, für diesen ein Verzeichnis `NAME` anzulegen, das dessen *paging-files* aufnehmen kann (damit wird der Arbeitsspeicher virtuell vergrößert). Auf dieses Verzeichnis ist dann mit `set go32tmp=NAME` eine Umgebungsvariable zu setzen. Dieser Befehl sollte sinnvollerweise in der Datei `autoexec.bat` stehen.

---

## 4 Beispiele

Natürlich kann in die formatierte Ausgabe der Literaturangaben eingegriffen werden. Die Befehle `\stressing` und `\punctuation` definieren die Betonung des Autoren-Nachnamens und das Satzzeichen, das zwischen dem Autorennamen und dem REST steht. Dies betrifft einerseits die Befehle `\vqu`, `\addtovqu` und `\printonlyvqu`, andererseits `\vli`, `\addtovli` und `\printonlyvli`. Voreinstellung für `\stressing` ist `underline`.

- `\stressing` kann im Vorspann und vor jedem Listen-Ausdruckbefehl stehen. Sein Argument kann z.B. `sc`, `large`, `underline` oder `fbox` enthalten.

```
\stressing{bf}
```

Ändert die Betonung des Autorennamens von unterstreichen auf fett (bold face).

- `\punctuation` sollte nur im Vorspann verwendet werden:

```
\renewcommand{\punctuation}{,}
```

Ändert den Doppelpunkt zwischen VORNAME bzw. NAME und REST in Komma.

Jetzt soll der Weg eines `\vli`-Befehls verfolgt werden:

- Hier steht (und man vergleiche dies mit der Fußnote unter besonderer Beachtung der serifenlosen Schrift):

```
... Text.\footnote{ ... mit Bezug auf  
\vli{Helmut}{Kopka}{\LaTeX} -- Eine Einführung,  
Bonn 1993},  
S. 187 belegen. }
```

... Text.<sup>12</sup>

<sup>12</sup>Ich möchte meine Aussagen mit Bezug auf Helmut Kopka: L<sup>A</sup>T<sub>E</sub>X – Eine Einführung, Bonn 1993, S. 187 belegen.

13

- Der `\vli`-Eintrag über Kopka kommt ins Literaturverzeichnis:

```
\section*{Literaturverzeichnis} \printvli
```

### Literaturverzeichnis

Kopka, Helmut: L<sup>A</sup>T<sub>E</sub>X – Eine Einführung, Bonn 1993  
Schwarz, Norbert: Einführung in T<sub>E</sub>X, Bonn 1988 [Schwarz]

- Im Literaturverzeichnis kann auch auf die Seite zurückverwiesen werden, auf der der Literatur-Zugang mit `\vli` stattfand:

```
\section*{Literaturverzeichnis} \printnumvli
```

## Literaturverzeichnis

Kopka, Helmut:  $\LaTeX$  – Eine Einführung, Bonn 1993  $\gg$  13<sup>12</sup>

Schwarz, Norbert: Einführung in  $\TeX$ , Bonn 1988 [Schwarz]  $\gg$  14

Wie angegeben, läßt sich bei Kopkas Buch sowohl Seite, als auch Fußnotennummer nachvollziehen.

- Die Angabe zu Norbert Schwarz steht nicht in einer Fußnote. Hier steht:

```
\vli{Norbert}{Schwarz}{Einführung in
\protect\TeX, Bonn 1988\kurz{Schwarz}}
```

Norbert Schwarz: Einführung in  $\TeX$ , Bonn 1988 (im folgenden: Schwarz)

- Die Schriftart, in der hier das Erstzitat „Kopka“ oder „Schwarz“ ausgedruckt ist, kann einheitlich von *sans serif* auf *italics* umgestellt werden.

```
\renewcommand{\schrift}{\it}
```

Ändert die Schrift, in der Erstzitate ausgedruckt werden, in *italics*.

- Das Zeichen  $\gg$  in den `\printnum...`-Listen ist durch den Befehl `\verw` definiert; es kann beispielsweise so geändert werden:

```
\renewcommand{\verw}{; Erstzitat:~} Ändert das Zeichen, das vor den
Angaben des Zugangsorts steht.
```

- Die Schriftart, in der Literaturverzeichnis und Quellenverzeichnis ausgedruckt werden, heißt `\barschrift`. Änderungsvorschlag:

```
\renewcommand{\barschrift}{\rm}
```

Ändert die Schrift der bibliographischen Verzeichnisse in roman.

- Die Schriftart, in der das Stichwortverzeichnis ausgedruckt wird, heißt `\indschrift`. Änderungsvorschlag:

```
\renewcommand{\indschrift}{\sf}
```

Ändert die Schrift, in der der Index ausgedruckt wird in sans serif.

Jetzt ein Punkt für Fortgeschrittene: Ein noch nicht erwähnter Befehl heißt `\Kapitell`. Falls er genutzt wird, soll er im Vorspann direkt nach `\makebar` stehen. Er bewirkt, daß Groß-Buchstaben A bis Z in großer Schrift in das Quellenverzeichnis, das Literaturverzeichnis und das Kurzzitateverzeichnis (Index, Register) ausgedruckt werden. Diese Kapitelle stehen leicht abgesetzt am Anfang jeder Gruppe von Autoren mit einem bestimmten Anfangsbuchstaben. `\Kapitell` ist im Vorspann von `bibarts.tex` modifiziert worden, denn es sollte dazu unter jeden Buchstaben mindestens ein Zugang mit dem gleichen Anfangsbuchstaben fallen. Das ist im Register dieses Textes nicht der Fall. Auf Kapitelle in den Beispielen für Literatur- und Quellenverzeichnis wurde verzichtet. Der Vorspann von `bibarts.tex` kann als Beispiel für eigene Texte dienen. Notwendig sind hier die Befehle `\makeatletter` und `\makeatother` vor und nach Befehlen, die `@` enthalten.

Alle vorausgegangenen Beispiele betrafen Einstellungen im Umfeld des bibliographischen Files und die Einstellungen darüber, wie in das bibliographische File schreibende Befehle ihre Argumente an Ort und Stelle ausdrucken.

Doch jetzt zur zweiten Zitierkonvention, den Rück-Verweisen nach Kurzzitaten auf das vollständige Erstzitat. (Es ist keine Literaturliste oder vergleichbares vorgesehen für das Ende eines Textes, der diesen Teil der `BibA`s-Optionen nutzt.) Es folgt eine `\bibref-\bibmark`-Kombination:

```
... Text.\footnote{In dieser Fußnote sei
  Lecl\‘erc erstzitiert:
  Fran\c{c}ois Lecl\‘erc: Der kalte Sommerregen,
  Musterland 1899
  (\bibmark{Lecl\‘erc, Beruf}).}
Nicht in Fußnote:
  \bibref{Lecl\‘erc, Beruf}
... Text.\footnote{ Wie schon
  \bibref{Lecl\‘erc, Beruf} sagte ... }
```

... Text.<sup>13</sup> Nicht in Fußnote: Leclèrc, Beruf (siehe S. 15) ... Text.<sup>14</sup>

<sup>13</sup>In dieser Fußnote sei Leclèrc erstzitiert: François Leclèrc: Der kalte Sommerregen, Musterland 1899 (im folgenden: Leclèrc, Beruf).

<sup>14</sup>Wie schon Leclèrc, Beruf (wie Anm. 13 auf S. 15) sagte ...

Auf der nachfolgenden Seite soll als letztes Beispiel mit `\printind` der Index ausgedruckt werden. Diese nächste Seite wird vollständig fehlen, falls kein Index existiert. DOS: Der Index wird gleich von `gbib209.bat` oder `gbib2e.bat` mit-erzeugt durch Eingabe von `gbib2e bibarts`, falls die installierte `LATEX`-Version `2e` heißt, oder `gbib209 bibarts`, falls noch `LATEX2.09` verwendet wird.

```
\renewcommand{\indexname}{Register zu \bibarts}
\printind
```

## Register zu BibArts

### A

\addtovli: 2, 6, 7, 16,  
\addtovqu: 2, 8, 9, 16,

### B

\bibmark: 15<sup>13</sup>, 11, 12, 15,  
\bibref: 15<sup>14</sup>, 11, 12, 15,

### F

\female: 3, 6,  
\femalename: 6,

### G

gbibsort  
  @: 4<sup>7</sup>,  
  author killing (-K): 5,  
go32: 12,

### K

\killname: 5,  
\kurz: 11, 14,

### M

\makebar: 1, 2, 3, 5, 9,  
MAKEINDEX  
  !: 10<sup>10</sup>, 11<sup>11</sup>,  
  @: 4<sup>7</sup>,  
  |: 10,  
\makereportbar: 2,  
\male: 3, 6,  
\malename: 6,  
Maskierungszeichen: 4<sup>7</sup>, 10,

### P

\printonlybib: 4<sup>7</sup>,  
\printonlyvli: 2, 7,  
\printonlyvqu: 2, 8,  
\punctuation: 13,

### S

Schwarz: 14,  
\sort: 4<sup>7</sup>, 4, 6,  
\stressing: 13,

### V

\verw: 14,  
\vli: 13<sup>12</sup>, 6, 13, 14,  
\vqu: 1, 2, 8, 9, 16,

### X

\xbibmark: 12,  
\xbibref: 12,

## 5 Grenzen der Mächtigkeit von Bib<sub>A</sub>T<sub>S</sub>

Die Befehle `\vli` und `\vqu`, `\addtovli` und `\addtovqu` sind einer wichtigen Beschränkung unterworfen: Befehle, die in ihren Argumenten auftauchen, können „zerbrechen“, wenn Befehle Makros sind. Nicht der Befehl selbst wird in das `FILE.bar` geschrieben, sondern die expandierten Makros. Solche „zerbrochenen Befehle“ können beim Übersetzen mit L<sup>A</sup>T<sub>E</sub>X oft nicht mehr verstanden, also nicht ausgedruckt werden. T<sub>E</sub>X beschwert sich bei solchen Gelegenheiten auch oftmals darüber, daß seine Kapazität überschritten sei („... Sorry“). Wie bei den Befehlen `\index` oder `\glossary` auch, muß ein solcher Befehl dann durch ein davorgestelltes `\protect` geschützt werden. ODER: In `bibarts.sty` wird dieser Befehl im Makro `\@sIcHerUnG` — so wie dort vorgemacht — mit `\string` geschützt. Hier sind bereits die ganzen Umlaute und viele andere Akzente (wie Akut, Zirkumflex und Gravis) gegen Zerbrechen geschützt. Ich empfehle allerdings, die Konstruktion `\@ifundefined{newumlaut}...` unverändert zu lassen, denn sie erkennt den wirklich bedeutsamen Unterschied, ob `german.sty` verwendet wird, oder nicht. (Das Paket `german` muß — wenn es verwendet wird — vor `bibarts` eingeladen werden!)

Ebenfalls eine Sicherungsstruktur wurde für `\bibmark`-`\bibref`-Kombinationen eingeführt. Diese Befehle nutzen `\label`, `\pageref` und `\ref`, in deren Argumente keine Befehle stehen dürfen. Die Sicherung, die hier z.B. Umlaute möglich macht, heißt `\@markprotect`. Die hier bearbeiteten Befehle sind fast die selben, wie auch im Makro `\@sIcHerUnG`. Allerdings werden die Befehle hier nicht schreibgeschützt, sondern in `|X|`-Konstruktionen übersetzt, wobei `X` ein für jeden Befehl anderer Buchstabe ist und `|` dazu dient, daß Verwechslungsmöglichkeiten mit anderen Schlüsselbegriffen minimiert werden. Da das Makro `\@markprotect` gleichermaßen in den Marken, wie in den Referenzen verwendet wird, kann der Anwender bedenkenlos Befehle in die Marken setzen — *wenn* diese Befehle in `\@markprotect` zu Buchstaben expandiert werden. Da `\underline` und Umlaute derart bearbeitet werden, ist

```
\bibmark{\underline{M"uller}, Hallo}
\bibref{\underline{M"uller}, Hallo}
```

möglich. In `\@markprotect` können — wie dort vorgemacht — weitere häufig verwendete Befehle zur Verwendung als Verweis-Schlüsselworte bereitgestellt werden. Für seltene Konstruktionen steht `\xbibmark`-`\xbibref` zur Verfügung, die MARKE und auszudruckenden TEXT trennen. Verweis-Marken ergeben übrigens `\newlabel`-Einträge im Auxiliary-File mit dem Suffix `.aux`.

Doch zurück zur Bearbeitung des bibliographischen Files. Das Programm `gbibsort` kann so viele Zeilen sortieren, wie in der Größe `MAXLINES` in der Datei `gbibsort.c` zum Zeitpunkt der Übersetzung von `gbibsort.c` in eine ausführbare Datei festgelegt war. Ebenso dürfen Zeilen eines zu sortierenden Files nicht länger als `MAXLEN` (1024) sein. Längere Zeilen sollten gekürzt werden, indem ein Rest des Eintrags in eine Datei (mit dem Namen `DATEI-NAME`) verschoben wird; anstelle des Restes sollte `\input{DATEI-NAME}` eingetragen werden. Der Befehl `\input` muß in den `\vli`-, `\vqu`-, `\addtovli`-, `\addtovqu`-Befehlen durch ein davorgestelltes `\protect` oder durch Definition innerhalb des Makros `\@sIcHerUnG` vor dem „Zerbrechen“ geschützt werden.

Die Kapazität des Befehls `\bib` hat etwas größere Beschränkungen, als die anderen  $\text{\LaTeX}$ -Befehle.

- Einträge im Stichwortverzeichnis werden sicher richtig sortiert, wenn sie von Seiten mit Seitenzahlen zwischen 1 und 999 stammen. Zwei `\bib`-Befehle mit gleichem Argument dagegen werden, wenn auch nur einer von beiden auf einer Seite mit einer Seitenzahl  $\leq 0$  oder  $\geq 1000$  steht, nur durch Zufall richtig sortiert.
- Folgendes gilt für das Stichwortverzeichnis in Dokumentengröße `book` und `part`: Die Sortierhilfen für die Zähler `part` und `chapter` reichen von 1 bis 99. Zwei `\bib`-Einträge mit gleichem Argument aus Teilen oder Kapiteln mit Nummern ab einhundert werden nur durch Zufall richtig sortiert. Tatsächlich erhält man, wenn Zugänge aus mehreren Kapiteln und sowohl aus dem Text als auch aus Fußnoten kommen, Ausgaben in der Art

STICHWORT:  
*Kapitel 1:*  
 $2^5$ ,  $5^8$ , 1, 4,  
*Kapitel 3:*  
 $23^5$ ,  $31^{18}$ , 24, 25,

Zuerst werden kapitelweise alle Zugänge aus Fußnoten ausgedruckt und danach die anderen Zugänge aus dem jeweiligen Kapitel.

- `\bib` kann nur verwendet werden, wenn ein `Index-Style-File` in der Art von `gbibarts.ist` verwendet wird. `gbibarts.ist` muß neben Sprachspezifika wie `quote '='` folgende Zeilen enthalten:

```
delim_0 " "  
delim_1 " "  
delim_2 " "  
delim_n "\\komma "  
delim_t " "
```

Solche `STYLES` werden durch Aufrufe der Art

```
makeind(e)x -s STYLE FILE
```

bei der Bearbeitung der Datei `FILE` berücksichtigt.

Der  $\text{\LaTeX}$ -Index verweist zurück auf Stellen im Text. Falls der Index-Zugang `\bib{STICHWORT}` in den Fußnoten 1, 2 und 3 auf einer Seite 12 steht, erscheint ein Eintrag `STICHWORT 121-3`, wenn der Index mit `\printind` ausgedruckt wird. Zusammenfassungen der Art 1-3 werden nicht erzeugt, falls `\bib` oder `\addtobib` außerhalb von Fußnoten (im normalen Text) verwendet werden. Falls Zugänge von den Seiten 1, 2 und 3 kommen, wird 1, 2, 3 dastehen. Es kann daher oft sinnvoll sein, sich auf dem herkömmlichen Weg via `\index{STICHWORT}` einen separaten Index für Verweise in den Haupt-Text zu machen. Der  $\text{\LaTeX}$ -Index ist insofern eine gute Grundlage für ein Personenregister, das üblicherweise von anderen Registern getrennt wird, die mit den  $\text{\LaTeX}$ -Befehlen `\index` und `\glossary` erzeugt werden können. Die Mehrfachausgabe

einer Seitenzahl im B<sub>i</sub>b<sub>A</sub>T<sub>S</sub>-Kurzziatate-Index wird immerhin unterdrückt, falls von einer Seite mehrere gleiche Index-Zugänge kommen. Dies funktioniert auch dann, falls die Seitennummerierung mittels `\pagenumbering{roman}` auf kleine römische Zahlen umgestellt wird. `\pagenumbering{Roman}` funktioniert nicht.

Eine wirklich elegante Lösung dieser Probleme wird erst eine Erweiterung des Programms MAKEINDEX bringen, das dann nach mehreren Zahlen — neben einem Eintrag des Zählers `page` zumindest noch nach dem des Zählers `footnote` — sortieren können muß.

---

### Message

Die Idee von B<sub>i</sub>b<sub>A</sub>T<sub>S</sub> ist, die bibliographischen Angaben an einer zentralen Stelle *im Text* zu machen, und diese Informationen unter möglichst wenig Hinzutun des Anwenders an allen notwendigen Stellen ausgedruckt zu bekommen. So erhält man um den Preis des Markierens von Erstzitataten eine fertige Literaturliste. Sie erscheint an der Stelle, an der der Ausdruckbefehl plaziert wird. Jede und jeder, die oder der schon einmal tagelang am Erstellen einer Literaturliste saß, wird sofort die Arbeitersparnis einsehen: Das Durchsuchen der Fußnoten nach Erstzitataten entfällt vollständig. Dieses „Definieren an zentraler Stelle“ greift eigentlich eine Idee des Programmierens auf. Dort gilt es als guter Stil, mehrfach verwendete Größen, die jeweils gleich sein sollen, einmal und endgültig an einer Stelle festzulegen. Änderungen des Programmtextes lassen sich so besser beherrschen. Soll diese Idee auf das Schreiben von Texten übertragen werden, stellt sich die Frage, an welcher Stelle solche Definitionen sinnvollerweise zu machen sind.

Sicherlich ist es heute üblich, bei der Literatur-Recherche computer-gestützt zu arbeiten. Es existieren schon mehrere andere Programme oder Formatvorlagen, die die Idee zentralen Definierens in Form von Datenbanken umsetzen. Aus dem Text heraus kann auf einzelne Einträge in dieser Datenbank verwiesen werden, um sich so Titel ausdrucken lassen zu können. Die Frage ist, ob das Instrument zur Datenerfassung auch direkt zum Schreiben eines Textes benutzt werden soll.

Mit B<sub>i</sub>b<sub>A</sub>T<sub>S</sub> wird diese Frage mit *nein* beantwortet. Ich jedenfalls kann mir nicht tausende von *short cuts* oder Schlüsselbegriffen merken, die auf die einzelnen Titel verweisen. Die Literatur-Datenbanken müssen sowieso durchsucht werden. Die Optimierungs-Aufgabe dort stellt sich so: Wie schnell kann der Editor, mit dem der Text geschrieben wird, in einer Datenbank Informationen auffinden und in den Text transferieren? Mit dieser Frage habe ich mich hier nicht befaßt. Das wesentliche Problem im Rahmen geisteswissenschaftlicher Zitierkonventionen — d.h.: der Punkt, wo Optimierung zuerst ansetzen muß — liegt nämlich in der Kombination Erstzitat-Literaturliste. Bibliographische Listen werden (neben ihrem Selbstzweck) mit B<sub>i</sub>b<sub>A</sub>T<sub>S</sub> auch zu Kontrollausdrucken für die Erstzitate. B<sub>i</sub>b<sub>A</sub>T<sub>S</sub> wurde vor diesem Hintergrund veröffentlicht, in der Hoffnung, daß es als nützlich empfunden wird.

Timo.Baumann@hist.unibe.ch