

DANTE
Deutschsprachige
Anwendervereinigung T_EX e.V.

Dominik Waßenhoven: *Bibliographien erstellen mit biblatex (Teil 1)*, Die T_EXnische Komödie 2/2008, S. 53–75.

Reproduktion oder Nutzung dieses Beitrags durch konventionelle, elektronische oder beliebige andere Verfahren ist nur im nichtkommerziellen Rahmen gestattet. Verwendungen in größerem Umfang bitte zur Information bei DANTE e.V. melden. Für kommerzielle Nutzung ist die Zustimmung der Autoren einzuholen.

Die T_EXnische Komödie ist die Mitgliedszeitschrift von DANTE, Deutschsprachige Anwendervereinigung T_EX e.V. Einzelne Hefte können von Mitgliedern bei der Geschäftsstelle von DANTE, Deutschsprachige Anwendervereinigung T_EX e.V. erworben werden. Mitglieder erhalten Die T_EXnische Komödie im Rahmen ihrer Mitgliedschaft.

Bibliographien erstellen mit biblatex (Teil 1)

Dominik Waßenhoven

Der Artikel stellt das Paket `biblatex` vor und ist aufgrund seines Umfangs zweigeteilt. Im ersten Teil wird die Benutzung von `biblatex` beschrieben, wobei die Besonderheiten und Vorteile gegenüber »herkömmlichem« `BibTEX` im Vordergrund stehen. Im zweiten Teil, der in der nächsten Ausgabe von „Die T_EXnische Komödie“ erscheinen soll, wird anhand eines konkreten Beispiels aus dem geisteswissenschaftlichen Bereich gezeigt, wie man eigene Zitier- und Bibliografiestile erstellen kann. Da das Paket `biblatex` eine Fülle von Möglichkeiten bietet, kann und will dieser Artikel nicht erschöpfend sein; es sei deshalb explizit auf die sehr ausführliche Paketdokumentation [10] (in englischer Sprache) hingewiesen.

Alle Beispiele in diesem Artikel sind – sofern nicht anders angegeben – der `BibTEX`-Datenbank `examples.bib` entnommen, die zum Paket `biblatex` gehört. Für die Sprachumschaltung und die Verwendung der im Deutschen üblichen Anführungszeichen wurden die Einstellungen

```
\usepackage[ngerman]{babel}  
\usepackage[babel,german=quotes]{csquotes}
```

verwendet. Im Unterordner `examples` der Dokumentation zu `biblatex` finden sich zu jedem Stil weitere Beispiele als `tex`- und `pdf`-Datei.

Danken möchte ich Philipp Lehman, Simon Spiegel und Bernhard Tempel, die meine Fehler korrigiert und viele Anregungen und Verbesserungsvorschläge gemacht haben.

Einleitung

Trotz der vielen Bibliografiestile, die für `BIBTEX` zur Verfügung stehen, gelten Anfragen in Newsgroups und Foren häufig der Darstellung von Literaturverweisen und Bibliografien, die einer bestimmten Vorgabe folgen sollen. Mit `jurabib` [3] und `natbib` [5] stehen zwei Pakete bereit, die so flexibel sind, dass sich viele Anforderungen bereits erfüllen lassen. Außerdem ist es mit Hilfe von `custom-bib` (`makebst`) [4] mit geringem Aufwand möglich, eigene `bst`-Dateien zu erstellen. Häufig sind dennoch kleine Anpassungen erforderlich, für die dann Kenntnisse in der (nicht gerade trivialen) `BIBTEX`-Programmierung nötig sind. Hier setzt `biblatex` [10] von Philipp Lehman an. Das Erscheinungsbild der Bibliografie und der Zitate wird ausschließlich durch `LATEX`-Befehle gesteuert, während `BIBTEX` nur für die Sortierung der Bibliografie sowie für die Erstellung der Label herangezogen wird. Gegenüber `BIBTEX` erweitert `biblatex` aber auch die verfügbaren bibliographischen Kategorien, stellt also mehr Felder und Eintragsstypen zur Verfügung.¹ Das Paket liegt derzeit noch als Beta-Version vor (aktuell ist die Version 0.7b vom 6. 1. 2008), läuft aber bereits sehr stabil und ist bei mir schon seit über einem Jahr im Einsatz.

Die Zitierstile

Ähnlich wie `natbib` unterstützt `biblatex` sowohl die numerische Zitierweise, die bei `LATEX` der Standard ist (und auch in „Die `TEX`nische Komödie“ benutzt wird), als auch das Autor-Jahr-Schema. Darüber hinaus bietet `biblatex` auch Stile für Autor-Titel-Verweise, wie sie mit dem `jurabib`-Paket möglich sind, und für alphanumerische Zitate ähnlich dem Stil `alpha.bst`. Der gewünschte Stil wird als Paketooption an `biblatex` übergeben, also beispielsweise `[style=authoryear]`. Es lassen sich sogar unterschiedliche Zitierstile für die Literaturverweise und die Bibliografie einstellen. Dafür sind die Optionen `citestyle` und `bibstyle` zuständig, die anstelle von `style` benutzt werden können. Damit deckt `biblatex` einen Großteil der Zitierstile ab, für die bisher auf verschiedene Pakete zurückgegriffen werden musste. Einige der unterschiedlichen Zitierstile und ihre Ausgabe sind in Tabelle 1 angeführt.

¹ Mehr dazu auf Seite 16.

Tabelle 1: Einige Zitierstile von biblatex: Zitiert wurden *angenendt*, *companion*, *vazques-de-parga* und *springer*, außerdem wurde *geer* per `nocite` eingebunden. Bei `authoryear` und `authortitle` wurde lediglich *angenendt* zitiert.

Zitierstil	Ausgabe
<code>numeric</code>	[1, 4, 3, 5]
<code>numeric-comp</code>	[1, 3–5]
<code>numeric-verb</code>	[1]; [4]; [3]; [5]
<code>alphabetic</code>	[Ang02; Spr50; GMS94; VLUR93]
<code>alphabetic-verb</code>	[Ang02]; [Spr50]; [GMS94]; [VLUR93]
<code>authoryear</code>	Angenendt 2002
<code>authortitle</code>	Angenendt, „In Honore Salvatoris“

Kurztitelverweise (`authortitle` und `verbose`)

Für das Kurztitelschema, das vornehmlich zur Verwendung in Fußnoten gedacht ist, stellt biblatex nicht nur einen, sondern insgesamt sechs Zitierstile zur Verfügung. Den Kurztitel holt sich biblatex aus dem Eintrag im `shorttitle`-Feld der BibTeX-Datenbank, sofern dieser existiert. Gibt es keinen `shorttitle`-Eintrag, wird der gesamte Titel als Kurztitel genutzt. In der kompakten Variante (`authortitle-comp`) wird der Autor nur einmal ausgegeben, sofern zwei Werke desselben Autors in einem Literaturverweis erscheinen. Der Befehl

```
\cite{aristotle:rhetoric,averroes/bland,aristotle:physics,↔
aristotle:poetics}
```

wird damit als

Aristotle, *Physics*, *Poetics*, *Rhetoric*; Averroes, *Possibility of Conjunction*

ausgegeben. Daneben gibt es zwei Varianten, die den Kurztitel nur ausgeben, sofern ein Autor mit mindestens zwei Werken vertreten ist. Das obige Beispiel wird mit `authortitle-terse` zu

Aristotle, *Rhetoric*; Averroes; Aristotle, *Physics*; Aristotle, *Poetics*

und mit `authortitle-tcomp` zu

Aristotle, *Physics*, *Poetics*, *Rhetoric*; Averroes

Der Kurztitel von Averroes wird also in beiden *terse*-Varianten unterdrückt. Diese beiden Stile sind damit an den MLA-Stil angelehnt und können durchaus auch für Literaturverweise im Text genutzt werden.

Daneben gibt es noch zwei Stile, die direkte Folgezitate durch »ebd.«² (bzw. das Äquivalent der eingestellten Sprache) ersetzen: **authortitle-ibid** und **authortitle-icomp**, wobei die letztgenannte kompakte Variante der Verkürzung von **authortitle-comp** entspricht. Falls das Wiederholungszeit das erste auf einer Seite (bzw. einer Doppelseite) ist, werden jedoch in beiden Stilen Autor und Kurztitel ausgegeben.³

Neben diesen verkürzenden Varianten gibt es auch ausführliche Zitierstile des Kurztitelschemas, die **verbose**-Stile. Der Stil **verbose** liefert beim ersten Zitat eine vollständige Literaturangabe und erst bei Folgezitat die Ausgabe des Autors und des Kurztitels. Das Gleiche ist mit **verbose-ibid** der Fall, der einzige Unterschied besteht darin, dass bei zwei direkt aufeinanderfolgenden Zitaten desselben Werkes »ebd.« ausgegeben wird.

Der Stil **verbose-note** gibt ebenfalls zunächst ein Vollzitat aus. Die Folgezitate enthalten dann aber Verweise zur Fußnote, in der das erste Zitat erfolgt ist, indem »s. Anm. . . .« angefügt wird. Damit lässt sich dieser Stil *ausschließlich* in Fußnoten anwenden. Das gilt genauso für den Stil **verbose-inote**, der direkte Folgezitate durch »ebd.« ersetzt, ansonsten aber identisch ist mit **verbose-note**.

Die »traditionellen« Stile **verbose-trad1** und **verbose-trad2** gehören ebenfalls zu den ausführlichen Stilen und produzieren beim ersten Verweis Vollzitate, für die Folgezitate kommen dann Abkürzungen zum Einsatz. Der Einfachheit halber sei die Funktionalität an einem Beispiel verdeutlicht. Nehmen wir an, die folgenden drei Werke wurden bereits vorher im Text zitiert:

```
\footcite[21]{averroes/bland}
\footcite{aristotle:physics}
\footcite[23]{aristotle:rhetoric}
\footcite[23]{aristotle:rhetoric}
\footcite[21]{averroes/bland}
```

² Abkürzung für »ebenda«.

³ Das hängt von der Option `pagetracker` ab, die die Werte `false` (ausgeschaltet), `page` (einzelne Seite), `spread` (Doppelseite) und `true` (automatische Erkennung von ein- oder zweiseitigem Layout) annehmen kann.

```
\footcite{aristotle:physics}
```

Diese Folgezitate werden mit `verbose-trad1` in den aufeinanderfolgenden Fußnoten ausgegeben als:

Averroes, Possibility of Conjunction, S. 21.
 Aristotle, Physics.
 Ders., Rhetoric, S. 23.
 Ebd., S. 23.
 Averroes, a. a. O.
 Aristotle, Physics.

Mit `verbose-trad2` sieht die Ausgabe folgendermaßen aus:

Averroes, a. a. O., S. 21.
 Aristotle, Physics, a. a. O.
 Ders., Rhetoric, a. a. O., S. 23.
 Ebd., S. 23.
 Averroes, a. a. O., S. 21.
 Aristotle, Physics, a. a. O.

Für eine ausführliche Erläuterung muss an dieser Stelle ein Hinweis auf die Beispiele genügen, die im `examples`-Ordner von `biblatex` enthalten sind und die Funktionsweise dieser Stile hinreichend illustrieren.

Die Zitierbefehle

Wer sich das letzte Beispiel aufmerksam angeschaut hat, dem ist vielleicht aufgefallen, dass »Ebd.« und »Ders.«⁴ jeweils groß geschrieben werden. Dafür sorgt der `\footcite`-Befehl, der neben der Literaturangabe auch eine Fußnote erzeugt. Man kann das Verhalten aber auch mit jedem anderen Zitierbefehl erreichen, denn es gibt sie alle auch mit einem Großbuchstaben (also z. B. `\Cite`), wodurch Großschreibung erzwungen wird. Bei Namen fällt das normalerweise nicht auf, solange sie nicht ein »von« enthalten. Verwendet man einen Zitierbefehl mit Großbuchstaben, wird »von« zu »Von«, »ebd.« zu »Ebd.«, »ders.« zu »Ders.« usw.

Neben `\cite` gibt es die Befehle `\parencite` für Zitate in Klammern (siehe Tabelle 2), `\textcite` für Zitate im Fließtext⁵ (siehe Tabelle 3) und `\footcite`

⁴ Abkürzung für »Derselbe«. Falls es sich um einen weiblichen Autor handelt, muss das Feld `gender = {sf}` gesetzt sein (`sf` steht für »single female«). Weitere Werte für `gender` werden in der `biblatex`-Dokumentation erläutert.

Tabelle 2: Der `parencite`-Befehl: Die Ausgabe wurde mit der Anweisung `\parencite[121]{springer}` erzeugt. Für den Stil `verbose` wird die Ausgabe eines Folgezitates gezeigt.

<i>Zitierstil</i>	<i>Ausgabe</i>
<code>numeric</code>	[1, S. 121]
<code>alphabetic</code>	[Spr50, S. 121]
<code>authoryear</code>	(Springer 1950, S. 121)
<code>authortitle</code>	(Springer, „Mediaeval Pilgrim Routes“, S. 121)
<code>verbose</code>	(Springer, „Mediaeval Pilgrim Routes“, S. 121)

Tabelle 3: Der `textcite`-Befehl: Die Ausgabe wurde mit der Anweisung `\textcite[121]{springer}` erzeugt.

<i>Zitierstil</i>	<i>Ausgabe</i>
<code>numeric</code>	Springer [1, S. 121]
<code>alphabetic</code>	Springer [Spr50, S. 121]
<code>authoryear</code>	Springer (1950, S. 121)
<code>authortitle</code>	Springer („Mediaeval Pilgrim Routes“, S. 121)

für Literaturangaben in Fußnoten. Bei `\footcite` wird dieselbe Ausgabe, die der `\cite`-Befehl erzeugt, in eine Fußnote gepackt, an deren Ende ein Punkt angefügt wird.

Neben diesen Befehlen, die je nach Zitierstil eine unterschiedliche Ausgabe erzeugen, gibt es auch Kommandos, die nur einen Teil der Literaturangabe ausgeben und für Verweise im Text gedacht sind. Die Befehle lauten `\citeauthor` für Autoren oder Herausgeber, `\citetitle` für Titel oder – sofern vorhanden – Kurztitel (`shorttitle`), `\citetitle*` für den Inhalt des gesamten `title`-Feldes, auch wenn ein Kurztitel vorhanden ist, `\citeyear` für das Jahr und `\citeurl` für URLs. Auch `\nocite` für Literatur, die nicht zitiert wird, aber in der Bibliografie erscheinen soll, `\fullcite` für Vollzitate und `\footfullcite` für Vollzitate in Fußnoten werden von `biblatex` bereitgestellt. Alle diese Befehle (außer `\nocite`) können optionale Argumente mit einleitenden Angaben

⁵ Der `\textcite`-Befehl steht allerdings in den `verbose`-Stilen *nicht* zur Verfügung. Stattdessen wird auf den Befehl `\cite` zurückgegriffen.

(*prenote*, also etwa „vgl.“) und abschließenden Angaben (*postnote*, beispielsweise eine Seitenzahl) enthalten. Auch hier zur Veranschaulichung ein kleines Beispiel. Aus dem folgenden Quelltext

```
... schrieb er in \citetitle[121]{springer}.
```

macht biblatex mit dem Stil **authortitle**:

... schrieb er in „Mediaeval Pilgrim Routes“, S. 121.

Außerdem erkennt biblatex automatisch, ob im zweiten optionalen Argument eines `\cite`-Befehls eine Seitenangabe oder sonstiger Text steht und unterdrückt gegebenenfalls den Ausdruck »S.«, wie folgendes Beispiel demonstriert:

```
\cite[Siehe dazu][10]{angenendt}, \cite[vgl.][im Vorwort]{↔
companion}.
```

wird beispielsweise mit dem Stil **alphanumeric** folgendermaßen ausgegeben:

Siehe dazu [Ang02, S. 10], vgl. [GMS94, im Vorwort].

biblatex führt auch Befehle ein, mit denen man mehrere Werke auf einmal zitieren und ihnen dabei eigene optionale Argumente mitgeben kann. Das ist mit einem normalen `\cite`-Befehl nicht möglich, wie folgendes Beispiel verdeutlicht:

```
\cite[Vgl.][433]{angenendt,springer}
```

ergibt mit **authoryear** folgende Ausgabe:

Vgl. Angenendt 2002; Springer 1950, S. 433

Will man jedoch jeweils eigene Seitenangaben machen, kann man den Befehl `\cites` verwenden. Die Syntax ist etwas anders, denn die einzelnen *keys* werden nicht in einer kommaseparierten Liste im obligatorischen Argument angegeben, wie es bei `\cite` der Fall ist, sondern in jeweils eigenen, von geschweiften Klammern umgebenen Argumenten. Diesen Argumenten können jeweils eigene optionale Argumente vorangestellt werden:

```
\cites[433]{angenendt}[vgl.][123]{springer}
```

ergibt dann:

Angenendt 2002, S. 433; vgl. Springer 1950, S. 123

Es ist darüber hinaus auch möglich, *prenotes* und *postnotes* zum `\cites`-Befehl selbst hinzuzufügen, die in der Eingabe von runden Klammern eingeschlossen werden müssen. Das ist vor allem nützlich, wenn man `\parencites` oder `\footcites` benutzt. Auch dazu ein Beispiel:

```
... hier endet ein Satz \parencites(siehe zum Ganzen)(vgl. ←
auch Kap.~xy)[433]{angenendt}[vgl.][123]{springer}.
```

ergibt:

... hier endet ein Satz (siehe zum Ganzen Angenendt 2002, S. 433; vgl. Springer 1950, S. 123, vgl. auch Kap. xy).

Zur Unterstützung einiger Befehle, die aus den Paketen `jurabib` und `natbib` bekannt sind, gibt es die Option `[natbib=true]`. Damit können die von `natbib` eingeführten und von `jurabib` ebenfalls verwendeten Befehle zum Zitieren in Klammern (`\citep`) oder im Text (`\citet`) sowie die alternativen Befehle `\citealp` und `\citealt` weiterhin verwendet werden. Dieser `natbib`-Kompatibilitätsmodus hat den Vorteil, dass bestehende Dokumente ohne großen Aufwand auf die Benutzung von `biblatex` umgestellt werden können.

Mit `biblatex` lässt sich auch die Funktionalität der `natbib`-Option `super` nutzen. Es stellt dafür die Befehle `\supercite` und `\supercites` bereit, die nur in den numerischen Zitierstilen verfügbar sind und die numerischen Verweise als hochgestellte Ziffern ohne Klammern darstellen. Der Befehl `\supercite` ignoriert alle optionalen Argumente.

Schließlich steht noch der Befehl `\autocite` zur Verfügung, der je nach Zitierstil unterschiedliche Ausgaben erzeugt. Das besondere dabei ist, dass Satzzeichen hinter dem Befehl je nach Bedarf verschoben werden und auch ein Leerzeichen vor dem Befehl entfernt wird. Zur Verdeutlichung auch hier ein Beispiel:

```
... wenn es um \LaTeX{} geht \autocite{companion}.
```

Daraus wird mit `numeric`:

... wenn es um \LaTeX geht [1].

Analog dazu sieht die Ausgabe bei `alphabetic` und `authoryear` aus, nur der Verweisstil ändert sich natürlich. Die Ausgabe mit `authortitle` sieht dagegen folgendermaßen aus:

... wenn es um \LaTeX geht.¹

Das Zitat wird dann in der Fußnote ausgegeben. Im zweiten Fall verschiebt biblatex also den Punkt des Satzendes vor das Fußnotenzeichen und löscht das vorangegangene Leerzeichen. Die Zitierstile setzen dazu einen Wert für die Option `autocite`, die das Verhalten steuert. Diese Option kann die Werte `inline` (wie im ersten Beispiel), `footnote` (wie im zweiten Beispiel), `superscript` (nur bei numerischer Zitierweise) und `plain` (wie ein normaler `\cite`-Befehl) annehmen. Will man beispielsweise das Autor-Jahr-Schema benutzen und bei `\autocite`-Befehlen die Literaturverweise in den Fußnoten ausgeben lassen, muss die Option `autocite=footnote` explizit angegeben werden. Ein letztes Beispiel soll die Kombination von `numeric` und `autocite=superscript` verdeutlichen:

```
... antike Klassiker \autocite{aristotle:rhetoric,averroes/←
    bland,aristotle:physics,aristotle:poetics}.
```

Dies führt zu folgender Ausgabe:

```
... antike Klassiker.3,4,1,2
```

Die Ausgabe des Literaturverzeichnisses

Anders als bei `BIBTEX` gewohnt, definiert der Befehl `\bibliography{bibfile}` nur die Datei, der die Informationen entnommen werden, wobei auch hier mehrere bib-Dateien angegeben werden können. Eine Ausgabe der Bibliografie wird hingegen mit dem Befehl `\printbibliography` erzeugt. Mit Hilfe von optionalen Parametern, die `\printbibliography` mitgegeben werden können, ist es so leicht möglich, mehrere Bibliografien in einem Dokument oder ein unterteiltes Literaturverzeichnis ausgeben zu lassen (dazu später mehr). Die einfache Struktur eines `LATEX`-Dokuments, das biblatex benutzt, sieht folgendermaßen aus:

```
\documentclass{...}
\usepackage[...]{biblatex}
\bibliography{bibfile}
\begin{document}
\cite{...}
...
\printbibliography
\end{document}
```

Im Übrigen ist das bekannte Schema `latex – bibtex – latex – latex` zum Erzeugen des fertigen Dokuments um einen `LATEX`-Lauf verkürzt, d. h. es reicht

nach dem BibTeX-Lauf ein weiterer L^AT_EX-Lauf aus, um die Bibliografie und alle Literaturverweise zu erzeugen.⁶

Das Aussehen des Literaturverzeichnisses richtet sich nach dem Bibliografiestil, der für die Label verantwortlich ist. Bei den Stilen **numeric** und **alphabetic** wird das Label, das auch für die Zitate im Text angegeben wird, vor die bibliographischen Angaben geschrieben. Der Stil **authortitle** gibt dieselben Angaben ohne Label aus, während **authoryear** das Erscheinungsjahr in Klammern hinter den Autoren-/Herausgebernamen setzt.

Außerdem lässt sich das Erscheinungsbild der Bibliografie mit Hilfe einiger Optionen beim Laden des Pakets beeinflussen. Hier soll als Beispiel nur die Sortierung angeführt werden – für weitere Optionen sei auf die **biblatex**-Anleitung verwiesen. In der Standardeinstellung sortiert **biblatex** die einzelnen Einträge nach Autornamen, Titel und Jahr (in dieser Reihenfolge). Das entspricht der Paketoption **sorting=nty**. Nutzt man aber beispielsweise den Stil **alphabetic** und möchte zunächst nach Label, dann nach Autornamen, Jahr und Titel sortieren lassen, so geht das mit der Paketoption **sorting=anyt**. Die Option **sorting=none** gibt das Literaturverzeichnis hingegen in der Reihenfolge aus, in der die Einträge zitiert wurden. Darüber hinaus sind weitere Sortieroptionen möglich.

Feinarbeiten wie die Änderung der Interpunktion, das Darstellen der Autornamen in Kapitälchen und dergleichen können nicht über Paketoptionen eingestellt werden. Stattdessen werden die entsprechenden Befehle in der Präambel umdefiniert.⁷ Die Definitionen finden sich allesamt in der Datei **biblatex.def**. Sollen beispielsweise die Titel von Zeitschriftenartikeln nicht von Anführungszeichen umschlossen werden, so ändert man die Definition

```
\DeclareFieldFormat[article]{title}{\mkbibquote{#1}}
```

in folgende Anweisung um:

```
\DeclareFieldFormat[article]{title}{#1}
```

Der Befehl `\mkbibquote` wird in **biblatex.sty** definiert und baut auf `\enquote` aus dem Paket **csquotes** [11] auf, das für die Benutzung von **biblatex**

⁶ Je nach Zitierstil sind jedoch weitere Läufe nötig, um beispielsweise die *ibidem*-Funktionalität zu gewährleisten. **biblatex** gibt aber in jedem Fall eine Warnung aus, falls ein weiterer L^AT_EX-Lauf nötig sein sollte.

⁷ Man kann diese Änderungen sowie die Paketoptionen auch in die Datei **biblatex.cfg** auslagern; sie gelten dann global und müssen nicht in jedem Dokument definiert werden.

dringend empfohlen wird. Die angesprochenen Autorennamen in Kapitälchen erreicht man mit der Anweisung

```
\renewcommand*{\mkbibnamelast}[1]{\textsc{#1}}
```

Eine ausführlichere Darstellung solcher Änderungen erfolgt im zweiten Teil dieses Artikels anhand eines konkreten Beispiels.

Unterteilte Literaturverzeichnisse

Gerade in den Geisteswissenschaften kommt es häufig vor, dass Bibliografien unterteilt werden, beispielsweise in ein Quellenverzeichnis und ein Verzeichnis der Sekundärliteratur. Bisher war man für solche Anforderungen auf zusätzliche Pakete angewiesen, nämlich `bibtopic` [2], `multibib` [8] oder `splitbib` [12]. Mit `biblatex` sind dafür hingegen keine zusätzlichen Pakete nötig, stattdessen kann man dem Befehl `\printbibliography` entsprechende Optionen einfügen.

Will man seine Bibliografie thematisch unterteilen, kann man das Feld `keywords` heranziehen. Braucht man etwa ein Quellen- und ein Literaturverzeichnis, kann man jedem Eintrag, der in das Quellenverzeichnis aufgenommen werden soll, den Ausdruck `Quelle` mitgeben:

```
@book{...,
  ...,
  keywords = {Stichwort1,Stichwort2,Quelle},
  ...
}
```

Diese Stichworte sind nicht als druckbarer Text anzusehen, sondern als »Label«. Es dürfen also nur Zeichen im ASCII-Format benutzt werden, keine \LaTeX -Syntaxelemente wie `\`, `[`, `{}` usw. Außerdem sollte man beachten, dass zwischen Groß- und Kleinschreibung unterschieden wird; »`Quelle`« ist also nicht dasselbe wie »`quelle`«. Hat man seine bibliographischen Daten in dieser Form aufbereitet, braucht man `biblatex` nur noch mitzuteilen, das entsprechende Stichwort zu berücksichtigen:

```
\printbibliography[keyword=Quelle]
\printbibliography[notkeyword=Quelle]
```

Mit Hilfe der Anweisung `notkeyword` lassen sich alle zitierten Literaturangaben ausgeben, die nicht über das Stichwort `Quelle` verfügen. Damit reicht es aus, nur einen Teil der bibliographischen Daten entsprechend zu kennzeichnen.

Natürlich lassen sich auch mehrere Stichworte miteinander kombinieren, falls man ein mehrfach untergliedertes Literaturverzeichnis benötigt.

Wenn man die Bibliografie in dieser Weise teilt und mit den oben genannten `\printbibliography`-Befehlen aufruft, ergibt sich allerdings das Problem, dass beide Verzeichnisse mit der Standardüberschrift »Literatur« versehen werden. Man kann sich jedoch mit dem Befehl `\defbibheading`, der in der Präambel eines Dokumentes eingesetzt wird, eigene Überschriften definieren:

```
\defbibheading{quell}{\section{Quellenverzeichnis}}
\defbibheading{lit}{\section{Literaturverzeichnis}}
```

Das erste Argument von `\defbibheading` ist ein frei zu vergebender Name, mit dem man später auf die Überschrift zurückgreift. Das zweite Argument enthält den Code, der ausgeführt werden soll. Hierbei ist zu beachten, dass nicht nur der gewünschte Name angegeben wird, sondern auch der Gliederungsbefehl. Auf die so definierten Überschriften wird nun mit der Anweisung `heading` im optionalen Argument von `\printbibliography` verwiesen.

```
\chapter{Bibliografie}
\printbibliography[heading=quell,keyword=Quelle]
\printbibliography[heading=lit,notkeyword=Quelle]
```

Damit würde die Bibliografie insgesamt als Kapitel, das Quellen- und Literaturverzeichnis jeweils als Abschnitt ausgegeben und in das Inhaltsverzeichnis aufgenommen. Will man die Aufnahme in das Inhaltsverzeichnis verhindern, müsste man stattdessen die Stern-Varianten der Gliederungsbefehle verwenden, also `\chapter*` und beim Definieren der Überschriften `\section*`.⁸

Doch nun zurück zu den unterteilten Verzeichnissen: Möchte oder kann man die Daten in der `bib`-Datei nicht ändern, gibt es noch eine Alternative, denn `biblatex` bietet die Möglichkeit, die Bibliografie unter Verwendung der `BiBTeX`-Keys in Kategorien zu unterteilen; der Befehl dazu lautet `\DeclareBibliographyCategory`. Auch hierzu ein Beispiel zur Veranschaulichung:

```
...
\DeclareBibliographyCategory{quell}
\DeclareBibliographyCategory{lit}
\addtocategory{quell}{key1,key3,key17}
\addtocategory{lit}{key2,key25,key29}
```

⁸ Zu den vordefinierten Überschriften, die `biblatex` zur Verfügung stellt, siehe S. 16.

```

\defbibheading{quell}{\section{Quellenverzeichnis}}
\defbibheading{lit}{\section{Literaturverzeichnis}}
...
\chapter{Bibliografie}
\bibbycategory
...

```

Man muss also mit Hilfe des Befehls `\addtcategory` für jede Kategorie explizit angeben, welche Werke darin aufgelistet werden sollen. Der Befehl `\bibbycategory` gibt dann für jede Kategorie ein Verzeichnis aus und setzt die Überschrift darüber, die denselben Namen hat wie die Kategorie. Man darf also nicht vergessen, für jede Kategorie eine Überschrift zu definieren.

Alternativ ist es auch möglich, die `\printbibliography`-Befehle dafür zu verwenden. Das hat – zumindest bei nur zwei Kategorien – den Vorteil, dass man nur für eine der beiden Kategorien die *keys* festlegen muss und die andere mit der Angabe `notcategory` im optionalen Argument des `\printbibliography`-Befehls aufrufen kann:

```

...
\chapter{Bibliografie}
\printbibliography[heading=quell,category=quell]
\printbibliography[heading=lit,notcategory=quell]
...

```

Da es für längere Arbeiten trotzdem schnell unübersichtlich wird, wenn man die BibTeX-Keys einzeln angeben muss, kann man sich stattdessen einen eigenen Zitierbefehl erstellen, mit dem man den zitierten Titel automatisch einer Kategorie zuweist. Für das Quellenverzeichnis aus dem Beispiel (Kategorie `quell`) könnte der Befehl folgendermaßen aussehen:

```

\newcommand*{\citequell}[2][{}]{%
  \addtcategory{quell}{#2}\cite[#1]{#2}}

```

Damit ist eine Benutzung analog zur Vorgehensweise von `multibib` möglich.

Nach demselben Prinzip wie die `keywords`-Unterscheidung funktioniert auch eine Unterteilung nach Typen, wenn man beispielsweise Internetquellen separat angeben möchte. Die Anweisung lautet in diesem Fall `type` bzw. `nottype`:

```

...
\defbibheading{online}{\subsection*{Online-Quellen}}
\defbibheading{offline}{\subsection{Literatur}}

```

```
...
\section*{Literaturverzeichnis}
\printbibliography[heading=online,type=online]
\printbibliography[heading=offline,notype=online]
```

Alle zitierten Einträge der bib-Datenbank vom Typ **@online** landen hier also im ersten Verzeichnis, das mit der Überschrift »Online-Quellen« versehen wird. Ebenso könnte man auch nach Zeitschriftenartikeln und Tagungsbänden einerseits sowie nach Büchern und Sammelbänden andererseits unterscheiden:

```
\printbibliography[type=article,incollection,inproceedings]
\printbibliography[type=book,collection,proceedings]
```

Auch mehr als zwei Verzeichnisse sind selbstverständlich möglich.

Mehrere Bibliografien in einem Dokument

Neben unterteilten Literaturverzeichnissen werden auch immer wieder mehrere Bibliografien in einem Dokument benötigt, etwa beim Publizieren eines Tagungsbandes. Die Pakete `chapterbib` [1] oder `bibunits` [7] benötigt man nicht mit `biblatex`; sie sind sogar inkompatibel. Stattdessen benutzt man die Befehle `\begin{refsection}` und `\end{refsection}`, um sein Dokument in verschiedene Bereiche einzuteilen. `biblatex` erzeugt dann für jede *refsection* eine **aux**-Datei und benennt sie »*Dateiname1*-blx.aux«, »*Dateiname2*-blx.aux« usw. Für jede dieser **aux**-Dateien muss man einen `BIBTEX`-Lauf durchführen.

Im einfachsten Fall, wenn am Ende jedes Bereiches das Verzeichnis ausgegeben werden soll, reicht ein simples `\printbibliography` vor dem Befehl `\end{refsection}`. Stimmen die Bereiche mit der Gliederung des Dokumentes überein, gibt es sogar die Möglichkeit, mit der Paketoption `refsection` die *refsections* für jeden Abschnitt automatisch erstellen zu lassen. Diese Option kann die Werte `none`, `part`, `chapter`, `section` und `subsection` annehmen.

Soll die Bibliografie hingegen ganz am Ende erscheinen und nach den einzelnen, im Dokument festgelegten *refsections* unterteilt werden, benutzt man den Befehl `\bibbysection`, der alle *refsections* ausgibt, in denen Zitate vorkommen, und leere *refsections*, in denen also nichts zitiert wurde, überspringt.

```
\documentclass{...}
...
\defbibheading{bibliography}{%
```

```

\subsection*{Literatur zu Abschnitt~\ref{refsection:\↔
therefsection}}}}
...
\begin{document}
\section{...}
\begin{refsection}
...
\end{refsection}
\section{...}
\begin{refsection}
...
\end{refsection}
...
\section*{Bibliografie}
\bibbysection

```

biblatex gibt jeder *refsection* ein Label mit, das den Namen »refsection:« und die fortlaufende Nummer hinter dem Doppelpunkt erhält. Dadurch kann auf die einzelnen *refsections* verwiesen werden, wie im obigen Beispiel bei der Definition der Überschriften. Die einzelnen Abschnitte der Bibliografie würden in diesem Fall also mit der Überschrift »Literatur zu Abschnitt 1« usw. versehen. Man kann auch auf den Namen des jeweiligen Abschnitts verweisen, wenn man das Paket `nameref` [15] einbindet. Die Definition für die Überschrift könnte dann folgendermaßen aussehen:

```

\usepackage{nameref}
\defbibheading{bibliography}{%
  \subsection*{%
    Literatur zum Abschnitt \enquote{%
      \nameref{refsection:\therefsection}}}}

```

Je nach Zitierstil ist für eine unterteilte Gesamtbibliografie die Umgebung `refsegment` den gerade beschriebenen *refsections* vorzuziehen. Der Unterschied besteht darin, dass die Label, die biblatex für jedes zitierte Werk generiert, bei *refsections* lokal sind, während sie mit *refsegments* für das gesamte Dokument gelten. Bei numerischer Zitierweise fangen die Label also bei jeder *refsection* wieder bei »[1]« an. Aber auch bei der Verwendung des Stils `authoryear` kann es sinnvoll sein, *refsegments* einzusetzen, weil nur damit gewährleistet ist, dass im gesamten Dokument nicht zweimal dasselbe Label vergeben wird, sondern z. B. »Meier 2000a« und »Meier 2000b«.

Die *refsections* und *refsegments* können nicht weiter verschachtelt werden, man kann aber innerhalb einer **refsection**-Umgebung mehrere *refsegments* definieren, um beispielsweise für die Literaturverzeichnisse einzelner Artikel in einem Tagungsband jeweils eine thematische Untergliederung zu ermöglichen.

Vordefinierte Überschriften

Normalerweise wird ein Literaturverzeichnis als nicht nummerierter Abschnitt (oder Kapitel) ausgegeben. Will man die Bibliografie in das Inhaltsverzeichnis übernehmen, kann man auf vordefinierte Überschriften für die **heading**-Anweisung im optionalen Argument des `\printbibliography`-Befehls zurückgreifen.⁹ Der Standard nennt sich **bibliography**, das heißt, dass `\"printbibliography[heading= bibliography]` nichts anderes bewirkt als derselbe Befehl ohne optionales Argument. Will man die Literaturangaben eine Gliederungsebene tiefer haben (also **\subsection*** bei Artikel-Klassen, **\section*** bei Buch-Klassen), so gibt es **subbibliography**. Mit **bibintoc** oder **subbibintoc** wird die Bibliografie in das Inhaltsverzeichnis aufgenommen. Soll sie zusätzlich noch nummeriert werden, heißen die Anweisungen **bibnumbered** bzw. **subbibnumbered**.¹⁰

Der Aufbau der BibTEX-Datei

Der Aufbau einer **bib**-Datei, die mit **biblatex** verwendet wird, unterscheidet sich grundsätzlich nicht vom Aufbau herkömmlicher BibTEX-Datenbanken. Im Folgenden soll deshalb nur auf einige Besonderheiten eingegangen werden, während die Grundlagen in [6, 13, 14] nachzulesen sind.

biblatex führt eine ganze Reihe neuer Felder ein, die entweder von den mitgelieferten Stilen bereits verwendet werden oder zur Verfügung stehen, um sie in angepassten Stilen einzusetzen. Das reicht von **redactor** für Bearbeiter eines Werkes über **shorthand** für abgekürzt zitierte Literatur bis hin zu **subtitle** für Untertitel. Alle neuen Felder hier vorzustellen, würde zu weit führen,¹¹ deshalb sollen nur einige Felder erwähnt werden, die gerade für die Geisteswissenschaften interessant sind. Als erstes ist hier **bookauthor** zu

⁹ Wie man sich eigene Überschriften definiert, habe ich bereits auf S. 12 beschrieben.

¹⁰ Die Optionen **bibtotoc** und **bibtotocnumbered** der KOMA-Script-Klassen [9] funktionieren derzeit nicht mit **biblatex**. In der nächsten »Maintenance«-Version (0.7c) werden sie unterstützt.

¹¹ Die **biblatex**-Dokumentation listet aber alle Felder auf und beschreibt sie kurz.

nennen, das für den Fall vorgesehen ist, wenn der Autor eines Aufsatzes und der Buchautor (nicht der Herausgeber) übereinstimmen. Als Beispiel wäre hier an die gesammelten Aufsätze eines Autors zu denken, wobei man nur einen Aufsatz zitieren will, nicht aber das gesamte Buch. Dieser Fall konnte bisher mit `BIBTEX` nicht vernünftig gelöst werden; mit `biblatex` gibt man den Autor (in den Feldern `author` und `bookauthor`), den Titel des Aufsatzes (`title`) und den Buchtitel (`booktitle`) an und erhält eine passende Ausgabe.¹² Daneben gibt es `origlanguage`, `origlocation`, `origtitle` und `origyear`, die für übersetzte oder nachgedruckte Werke benutzt werden können, um die Angaben des Originalwerks in die Bibliografie aufzunehmen. Bis auf `origlanguage` werden diese Felder zwar nicht von den Standardstilen genutzt, sie lassen sich aber in angepassten Stilen verwenden.

`biblatex` unterscheidet dabei verschiedene Arten von Feldern:

- Wortgetreue Felder wie `title`, deren Inhalt ausgegeben wird, wie er ist,
- Bereichsfelder wie `page` oder `year`, bei denen ein einfacher Divis (-) durch einen Halbgeviertstrich (L^AT_EX-Eingabe: --, Ausgabe: -) ersetzt wird,
- Zahlfelder wie `edition` oder `month`, die entweder als Ordinalzahl ausgegeben oder durch ein Wort ersetzt werden können,
- Datumsfelder wie `date` oder `urldate`, die ein bestimmtes Eingabeformat erfordern, nämlich JJJJ-MM-TT, also z. B. 2008-01-25 für den 25. Januar 2008, sowie
- Verbatim-Felder wie `url` oder `doi`, deren Inhalt bestimmte Sonderzeichen enthalten kann, z. B. Unterstriche (`_`) oder die Tilde (`~`).

Hinzu kommen noch Key-Felder, die einen bestimmten, festgelegten Wert enthalten können. Ein Beispiel für ein solches Feld ist `pagination`, in das man die Werte `page`, `column`, `line`, `verse`, `paragraph` und `none` eintragen kann. Dieses Feld bestimmt dann den Paginierungstyp, der beim Zitieren eines Werks benutzt wird. Normalerweise sind das Seiten (`page`), aber beispielweise werden manche Lexika nach Spalten (`column`) gezählt. In Gedichtbänden könnte ein Verweis auf Verse sinnvoll sein, in kritischen Editionen auf Zeilennummern (`line`) oder in juristischen Werken auf Paragraphen. Stellt man `pagination` entsprechend ein, wird die Angabe im optionalen Argument des `\cite`-Befehls durch die sprachabhängige Bezeichnung oder ihr Kürzel (im Deutschen: S., Sp., Z., V., §) eingeleitet. Wählt man `none` aus, wird keine Paginierungsangabe gemacht.

¹² In den mitgelieferten Stilen wird die Nennung des Buchautors unterdrückt. Mit angepassten Stilen ließe er sich aber ausgeben oder durch »Ders.« ersetzen.

Neben den genannten Feldern gibt es noch Listenfelder, die aus mehreren Einheiten bestehen, getrennt jeweils durch ein **and**. In traditionellem $\text{BIB}\text{T}\text{E}\text{X}$ ist diese Art von Feld lediglich bei **author** und **editor** gegeben; das entspricht in **biblatex** den Namenslisten (*name lists*), die in vier Teile zerlegt werden können: Vorname, Namenspräfix (z. B. »von«), Nachname und Namenssuffix (z. B. »senior«). **biblatex** führt aber noch eine weitere Art von Listenfeldern ein, die *literal lists*. Felder dieser Art sind sozusagen eine abgespeckte Version der Namenslisten, die lediglich am »and« getrennt werden. Die Felder **publisher**, **institution** (bzw. **school**), **organization** und **location** (bzw. **address**) sind solche *literal lists*. Man sollte also den Ort einer Publikation folgendermaßen angeben:

```
...
location = {New York and Toronto and Berlin and Tokyo and ↔
  Madrid},
...
```

Das hat den Vorteil, dass man mit den Paketoptionen **maxitems** und **minitems** die Zahl der auszugebenden Orte (Verlage, Institutionen usw.) eingrenzen kann. Die Standardwerte sind »3« für **maxitems** sowie »1« für **minitems**. Das obige Beispiel würde in der Standareinstellung also »New York u. a.« ergeben. **biblatex** gibt das Feld auch aus, wenn die Orte durch Kommata getrennt sind, allerdings ist die **maxitems/minitems**-Funktionalität damit nicht mehr möglich. Für Namenslisten (Autoren/Editoren) heißen die entsprechenden Optionen **maxnames** und **minnames**. Sie können sowohl global (für alle Bibliographien in einem Dokument) als auch separat für jedes einzelne Verzeichnis definiert werden.

Neben neuen Feldern gibt es auch Eintragstypen, die beim Standard- $\text{BIB}\text{T}\text{E}\text{X}$ nicht dabei sind. Dies sind im Einzelnen:

- **@collection** für Bücher mit mehreren eigenständigen Beiträgen, die (normalerweise) keinen Autor, sondern einen oder mehrere Herausgeber haben; Beispiele sind Sammelbände oder Lexika.
- **@online** für Internetquellen; die Typen **@electronic** und **@www** können synonym gebraucht werden. Das Feld **url** ist auch in anderen Eintragstypen verfügbar, das heißt, dass ein Artikel, der auch online erschienen ist, mit **@article** erfasst werden sollte, während **@online** eher für Webseiten, Blogs usw. gedacht ist.
- **@patent** für Patente.

- **@periodical** für Einzelausgaben von Zeitschriften (z. B. Sonderausgaben). Wenn sie einen eigenen Titel haben, kann er im Feld **issuetitle** angegeben werden. Damit wird dieser Eintragstyp anders verwendet als bei **jurabib**, wo **@periodical** für Aufsätze in Zeitschriften gedacht war, die nach Bandzählung zitiert werden, nicht nach Jahreszählung. Das ist bei **biblatex** aber genauso mit **@article**-Einträgen möglich.

Außerdem werden einige Typen anders verwendet als bei traditionellem **BIB-T_EX**. Dazu gehören **@phdthesis** und **@mastersthesis**, die nur als Alias für den allgemeineren Eintragstyp **@thesis** dienen. Dieser Eintragstyp hat wiederum ein Feld **type**, in dem die Art der Arbeit festgelegt werden kann. Vordefiniert sind **phdthesis** für Doktorarbeiten und **mathesis** für Magisterarbeiten. Wenn man nun **@phdthesis** benutzt, wird eigentlich **@thesis** mit **type = {phdthesis}** verwendet. Die als Alias unterstützten Eintragstypen wie **@phdthesis** können jedoch nicht für die Definition von unterteilten Literaturverzeichnissen aufgrund des Eintragstyps verwendet werden, d. h. in `\printbibliography[notttype=phdthesis]` bliebe die angegebene Option ohne Wirkung, während `\printbibliography[notttype=thesis]` auch die Einträge mit **@phdthesis** ausschließt. Auch **@inbook** wird anders behandelt. Es ist kein Kapitel eines Buches, sondern ein selbständiger Beitrag in einem Buch und verhält sich damit zu **@book** wie **@incollection** zu **@collection**. Wenn man überhaupt Buchkapitel in das Literaturverzeichnis aufnehmen möchte, kann man dafür die Felder **chapter** und **pages** heranziehen.

Obwohl **biblatex** für viele Probleme, die beim Umstieg von anderen Paketen wie **natbib** oder **jurabib** auftreten können, bereits Lösungen bereithält, wird man zumindest dann nicht umhinkommen, seine **bib**-Dateien anzupassen, wenn man die Flexibilität von **biblatex** voll ausschöpfen möchte. Das sei noch kurz an einem Beispiel erläutert, nämlich der Handhabung von mehrbändigen Werken. Ein dreibändiges Werk, das man als Ganzes zitieren möchte, gibt man etwa folgendermaßen an:

```
@book{key,
  author = {Max Muster},
  title = {Der Werkttitel},
  volumes = {3},
  location = {Ort},
  year = {2002}
}
```

Das würde ausgegeben als:

Max Muster. *Der Werktitel*. 3 Bde. Ort 2002.

Das Feld `volumes` gibt also die Gesamtzahl der Bände an, der Ausdruck »Bde.« richtet sich nach der Spracheinstellung. Will man nun lediglich den zweiten Band dieses Werkes zitieren, bräuchte man ja zwei `title`-Felder, eins für den Werktitel, das zweite für den Titel des Bandes. In diesem Fall wird der Werktitel mit `maintitle` gesetzt:

```
@book{key,
  author = {Max Muster},
  maintitle = {Der Werktitel},
  title = {Titel des zweiten Bandes}
  volume = {2},
  location = {Ort},
  year = {2002}
}
```

Hier sieht die Ausgabe folgendermaßen aus:

Max Muster. *Der Werktitel*. Bd. 2: *Titel des zweiten Bandes*. Ort 2002.

Das `volumes`-Feld ist in diesem Fall nicht gesetzt, sondern lediglich das `volume` Feld, das die Nummer des Einzelbandes angibt.

Damit lassen sich also für viele Fälle flexible Lösungen finden, wie ich an einem echten Beispiel aus meinen Forschungen noch demonstrieren möchte (das nicht in `examples.bib` vorhanden ist):

```
@BOOK{Christiansen1980a,
  author = {{Saxo Grammaticus}},
  title = {Books X, XI, XII and XIII},
  location = {Oxford},
  year = {1980},
  gender = {sm},
  series = {BAR International Series},
  number = {84},
  mainsubtitle = {Books X--XVI},
  maintitle = {Danorum Regum Heroumque Historia},
  maintitleaddon = {The text of the first edition with ↔
    translation and commentary in three
    volumes},
  translator = {Eric Christiansen},
  volume = {1}
}
```

Dieses Werk wird in der Bibliografie dann so ausgegeben:

Saxo Grammaticus. *Danorum Regum Heroumque Historia. Books X–XVI*. The text of the first edition with translation and commentary in three volumes. Bd. 1: *Books X, XI, XII and XIII*. Übers. von Eric Christiansen. BAR International Series 84. Oxford 1980.

Die Beispiele zeigen aber nicht nur die Flexibilität von biblatex, die eine Anpassung der bibliographischen Daten erfordern, sondern lassen auch erkennen, dass man sich gut überlegen sollte, ob sich der Aufwand, die eigenen bib-Dateien umzustellen, lohnt. Denn hat man seine Daten einmal umgestellt, sind sie unter Umständen nicht mit anderen, »herkömmlichen« Bibliografestilen verwendbar, ohne die Anpassungen wieder rückgängig zu machen.

Sigel

Zum Schluss soll noch eine sehr nützliche Funktion von biblatex vorgestellt werden: Die Handhabung von Sigeln, also von Abkürzungen für häufig zitierte Literatur. Diese Sigel sind in den Geisteswissenschaften recht häufig anzutreffen. Als Beispiel sei hier ein Lexikon angeführt:¹³

```
@COLLECTION{LMA,
  title = {Lexikon des Mittelalters},
  address = {München and Zürich},
  year = {1980-1999},
  volumes = {10},
  gender = {pp},
  shorthand = {LexMA}
}
```

Sobald ein **shorthand** definiert ist, wird es für den Literaturverweis herangezogen. Es wird also bei numerischer Zitierweise keine Zahl ausgegeben, sondern »[LexMA]«. Das Gleiche gilt für die alphabetische Zitierweise. Bei **authoryear** und **authortitle** wird entsprechend »LexMA« ausgegeben. Bei den **verbose**-Stilen hingegen, bei denen das erste Zitat ja ein Vollzitat ist, wird die entsprechende Information angehängt:

Lexikon des Mittelalters. 10 Bde. München und Zürich 1980–1999 (im Folgenden zit. als LexMA)

¹³ Auch dieses Beispiel ist nicht in der Datei `examples.bib` zu finden.

Nutzt man solche Kürzel, sind sie bei Autor-Jahr- und Autor-Titel-Zitierweise nicht mehr einfach im Literaturverzeichnis aufzufinden.¹⁴ Stattdessen kann man sich mit dem Befehl `\printshorthands` an beliebiger Stelle im Dokument eine Liste dieser Kürzel ausgeben lassen. Analog zu den vordefinierten Überschriften (siehe S. 16) kann man dieses Sigelverzeichnis übrigens auch mit den Befehlen `\printshorthands[heading=losintoc]` und `\printshorthands[losnumbered]` in das Inhaltsverzeichnis aufnehmen und (im zweiten Fall) nummerieren.

Zusammenfassung

Das Paket `biblatex`, das sich zwar noch im beta-Status befindet, aber schon zur Erstellung von Aufsätzen und mindestens einem Buch eingesetzt wurde, bietet durch seine neuen Felder und Publikationstypen, seine erweiterten Zitierbefehle und die bereitgestellten Stile die Möglichkeit, Literaturzitate und Bibliografien flexibel zu gestalten. `BIBTEX` wird dabei nur zum Sortieren der Bibliografie und Erstellen der Label herangezogen, so dass alle Anpassungen mit `LATEX`-Befehlen erfolgen können. Wie solche Anpassungen im Einzelnen aussehen können, soll im zweiten Teil dieses Beitrags erläutert werden.

Literatur

- [1] Donald Arseneau: *Chapterbib*; Febr. 2004; CTAN: [macros/latex/contrib/cite](#).
- [2] Pierre Basso und Stefan Ulrich: *Bibtopic*; Sept. 2006; CTAN: [macros/latex/contrib/bibtopic](#).
- [3] Jens Berger: *Das jurabib-Paket*; Jan. 2007; CTAN: [macros/latex/contrib/jurabib](#).
- [4] Patrick W. Daly: *Customizing Bibliographic Style Files*; Apr. 2007; CTAN: [macros/latex/contrib/custom-bib](#).
- [5] Patrick W. Daly: *Natural Sciences Citations and References*; Okt. 2007; CTAN: [macros/latex/contrib/natbib](#).

¹⁴ Bei numerischer und alphabetischer Zitierweise werden in der Bibliografie ja die Label vor die Literaturangabe geschrieben, so dass das Auffinden kein Problem darstellt.

- [6] Jürgen Fenn: *Managing Citations and Your Bibliography with BIBTEX; The PracTEX Journal*; 4; 2006; <http://www.tug.org/pracjourn/2006-4/fenn>.
- [7] Torsten Hansen: *The bibunits Package*; Mai 2004; CTAN: [macros/latex/contrib/bibunits](#).
- [8] Torsten Hansen: *The multibib Package*; Jan. 2004; CTAN: [macros/latex/contrib/multibib](#).
- [9] Markus Kohm, Frank Neukam und Axel Kielhorn: *KOMA-Script. Ein wandelbares L^AT_EX 2_ε-Paket*; Dez. 2007; CTAN: [macros/latex/contrib/koma-script](#).
- [10] Philipp Lehman: *The biblatex package. Programmable bibliographies and citations*; Jan. 2008; CTAN: [macros/latex/exptl/biblatex](#).
- [11] Philipp Lehman: *The csquotes package. Context sensitive quotation facilities*; Jan. 2008; CTAN: [macros/latex/contrib/csquotes](#).
- [12] Nicolas Markey: *Split your bibliography into categories*; Dez. 2005; CTAN: [macros/latex/contrib/splitbib](#).
- [13] Nicolas Markey: *Tame the BeaST*; Okt. 2005; CTAN: [info/bibtex/tamethebeast](#).
- [14] Oren Patashnik: *BibTEXing*; Febr. 1988; CTAN: [biblio/bibtex/contrib/doc/btxdoc.pdf](#).
- [15] Sebastian Rahtz: *Section name references in L^AT_EX*; Mai 2007; CTAN: [macros/latex/contrib/hyperref](#).