Towards a New Bibliography Format

Jean-Michel Hufflen

Abstract

MlBibTeX, our reimplementation of BibTeX, uses an enriched format for bibliography database files. Due to some features of this format, it is not backward-compatible with the conventions recognised by 'old' BibTeX. We explain why and make precise the requirements for a modern bibliography format and the specification of our new format.

Keywords MlBibTeX, bibliography processor, Unicode, character encodings, bibliography format.

Sommario

MlBibTeX, la nostra reimplementazione di BibTeX, usa un formato ampliato per il database bibliografico. A causa di alcune particolarità di questo formato, non è retrocompatibile con le convenzioni del 'vecchio' BibTeX. Spieghiamo perché e mostriamo con precisione i requisiti per un formato bibliografico moderno e le specifiche del nostro nuovo formato.

Parole chiave MlBibTeX, elaboratore bibliografico, Unicode, codifche dei caratteri, formato bibliografico.

Preliminary note The bibliography of the present article has been processed with MlBibTeX, as a showcase for this program.

Introduction

As mentioned in Mittelbach and Goossens (2004, § 13.1), the LaTeX typesetting system has deeply evolved since its first versions whereas BibTeX, the bibliography processor usually associated with LaTeX, has remained stable for many years. Let us recall that this program reads an auxiliary (.aux) file built when LaTeX typesets a source text (.tex file). The citation keys—used by the `\cite` command throughout .tex files—are extracted from this auxiliary file, and BibTeX looks in bibliography database (.bib) files for these keys. The auxiliary file also gives the pathnames of the .bib files to be searched and the information needed for the layout of a 'References' section. With BibTeX, such layouts are put into action by means of *bibliography styles*.

BibTeX is ageing, and we observe some possible replacement for several years. In particular, the biber program (Kime and Charette, 2014), generating references suitable for the biblatex package (Lehman et al., 2014). Let us recall that such references are marked up by means of LaTeX commands; accurate redefinitions of these commands allow bibliography styles to be put into action. Another example of a possible replacement is our reimplementation, MlBibTeX[1] (Hufflen, 2015). Since there is a *huge* number of .bib files—especially on the Web—BibTeX's successors still use this format for bibliography database files.

In Hufflen (2015) we introduced a new version of MlBibTeX, presently in beta-test. We recalled some syntactic extensions for .bib files. But these extensions induce some backward-compatibility problems: if end-users of MlBibTeX have written .bib files with this enriched syntax, what happens if they have to revert to 'old' BibTeX? Such a *scenario* is possible: nowadays, the whole process of publishing research papers in conference proceedings is often controlled by Web programs[2]. On such sites, all of the successive steps are controlled: electronic submission, acceptance or rejection, and depositing the final version if the paper has been accepted. Concerning this last step, often BibTeX is the only usable bibliography processor when articles are typeset with LaTeX.

This is clearly a drawback of using MlBibTeX[3], but in our personal opinion this drawback is the price to pay to get more expressive power within .bib files. We think that this format should be redefined and extended in next years, since there are requirements—especially modern ones—difficultly implementable within 'original' .bib format. In fact, this article aims to show that most of our syntactic extensions are debatable solutions out of .bib format expressive power. A comparable work about abbreviating authors' and editors' first names within bibliographies has already been presented in Hufflen (2016). The present work aims to give a more general view. In the first section, we recall this .bib format's origin and show how some LaTeX commands may be used as workarounds in quite simple cases. That is suitable for a 'basic' use of BibTeX, but makes other applications more complicated: for example, formatting the contents of .bib files and displaying the result on the Web by means of HTML[4] pages. Section 2 shows other examples where putting LaTeX commands does not apply or is unsatisfactory. As an alternative, the technique used by the biblatex package

1. **M**ulti**L**ingual BibTeX.
2. The most famous site for Computer Science conferences is indisputably `http://www.easychair.org`.
3. Some analogous drawbacks exist for biblatex users, as shown in § 3.
4. **H**yper**T**ext **M**arkup **L**anguage.

```
@BOOK{de-camp1991,
        AUTHOR = {Lyon Sprague de Camp and
                  Catherine Crook de Camp},
        TITLE = {The Incorporated Knight},
        PUBLISHER = {Baen Books},
        YEAR = 1991,
        MONTH = feb}
```

FIGURE 1: Example of a bibliographical entry.

to increase .bib files' expressive power is described in § 3. Then our choices are discussed in § 4. Reading this article requires some good knowledge of BIBTEX; advanced technical details can be found in Mittelbach and Goossens (2004, Ch. 13).

1 The .bib format

An example of a bibliographical entry usable by BIBTEX is shown in Fig. 1. In reality, this program (Patashnik, 1988) was initially designed to work with Scribe (Reid, 1984). This historical point explains some features of BIBTEX: Scribe has influenced LATEX—in particular, the notion of *document style* originates from Scribe—and was one of the first *markup languages*. However its syntax is close to LATEX's but is not identical. In Scribe, the '@' character is used at the beginning of a command name, such a command argument may be surrounded by braces, parentheses or double quote characters; '\' is not a special character[5], as in (LA)TEX. We see the origin of the '@' characters used within entry types such as `@ARTICLE` or `@BOOK`, the association of field names with corresponding values being surrounded by braces—mostly used—or parentheses. Even though BIBTEX's syntax is close to LATEX's and braces can be used to group some consecutive characters, accent commands are not recognised.

As LATEX has succeeded whilst Scribe has just had some historical value, the former has become the only word processor targeted by BIBTEX: for many years, bibliography styles built 'References' sections for LATEX, not for Scribe. So writers get used to put LATEX commands inside values associated with fields, for example, to specify typographical effects:

```
TITLE = {\emph{Babylon Babies}}
```

(this work's title uses italicised characters for foreign words, the book being written in French), or reach accented letters:

```
AUTHOR = {Herbert Vo{\ss}}
```

(the `\ss` command gives the German letter 'ß').

5. Analogous syntax is still used within Texinfo (Chassell and Stallman, 2008), the markup language used to document the products of the GNU (GNU's Not UNIX) projects.

These two examples are suitable for LATEX, but not for ConTEXt (Hagen, 2001), another format built out of TEX: the `\emph` command is unknown and the `\ss` command causes a switch towards a sans-serif font[6]. The second problem should disappear with the use of Unicode-compliant .bib files; however the 'ß' letter is quite frequent in German, and this `\ss` command would be still used in .bib files already developed. To correctly process these cases in ConTEXt the parser of .bib files should include a kind of mini-TEX parser for such commands. The same point holds if you are interested in building texts according to other formats, e.g., HTML pages.

It is well-known that the `AUTHOR` and `EDITORS` fields are structured: co-authors or co-editors are separated by the '`and`' keyword, as shown in Fig. 1, names are structured into four parts: *First*, *von* (a particle), *Last* and *Junior*[7]. As suggested by our notation, the *First*, *Last* and *Junior* parts begin with an uppercase letter whereas the *von* part begins with an uppercase letter. The complete rules for parsing names in BIBTEX are quite complicated[8], but the first author's name given in Fig. 1 is easily processed as follows:

```
first => Lyon Sprague, von => de,
last => Camp
```

(by using MlBIBTEX's alternative notation). If the particle begins with a lowercase letter, a workaround is needed as shown in Mittelbach and Goossens (2004, p. 766–768):

```
Maria {\MakeUppercase{d}e La} Cruz
```

Because of the 'd' letter, BIBTEX considers that the second token begins with a lowercase letter—the command name being irrelevant—and the dummy `\MakeUppercase` command is only used to put an uppercase letter when the bibliography is typeset. Here also, we need a mini-TEX parser to handle such cases for formats other than LATEX.

If you build a 'References' section where first names are to be abbreviated, let us recall that in most cases, only the first letter is retained, but some abbreviations use several letters, e.g., 'Clive', abbreviated into 'Cl.' In addition, some authors drop out their middle name, so 'Clive Eric Cussler' should be abbreviated into 'Cl. Cussler'. You can specify such *modus operandi* in BIBTEX:

```
{\relax Cl}{ive Eric} Cussler
```

6. In the first case, you have to use the '`{\em ...}`' construct. It exists in LATEX, but it is preferable to use the `\emph` command. Concerning the 'ß' letter, the ConTEXt command to get it is `\SS`, the namesake command in LATEX causes the case of 'ß' to be raised and the result is 'SS'.
7. This part is also known as *Lineage*.
8. You can find them in Hufflen (2006).

but from our point of view, it is actually a *dirty* trick. In such a case, additional syntax to specify a non-standard abbreviation is missing in the .bib format. The solution of MlBibTEX is more readable:

```
AUTHOR = {Clive Eric Cussler,
          abbr => Cl.}
```

An alternative solution could be:

```
Cussler,, Clive Eric, Cl.
```

—an empty *Junior* part being specified between the first two commas—but such notation using three commas is incorrectly processed by BibTEX and causes biber to crash.

2 More features

2.1 Difficultly with LaTeX commands

If we go on with fields for person names, it is impossible to specify *additional collaborators* if you use only standard fields[9]. Either collaborator names are dropped out, or viewed as co-authors. You can add specific fields, but they will not be recognised by standard bibliography styles. The syntax put into action by MlBibTEX just uses another connector, ignored by standard bibliography styles:

```
AUTHOR = {Robert Silverberg with
          Karen Huber}
```

(another example is given by Mittelbach and Goossens (2004)).

Sometimes there should be no space between a particle and the *Last* part:

Guy d'Antin

As far as we know, there is no satisfactory solution for specifying this French name in BibTEX, in order for it to be typeset nicely.

Let us go back to abbreviating first names, some authors retain their middle name when their first name is abbreviated, e.g.:

Henry Rider Haggard ⟶ H. Rider Haggard

Some books about LaTeX and BibTEX—e.g., Mittelbach and Goossens (2004, Ch. 13)—propose the use of square brackets:

```
H[enry] Rider Haggard
```

but only a few styles are able to handle them. Here also, MlBibTEX's additional syntax is unusable with 'old' BibTEX, but seems to us to be clearer:

```
Henry Rider Haggard, abbr => H. Rider
```

9. Such specification is allowed within the bibliographies managed by the DocBook system (Walsh, 2010).

```
<personname>
  <first>Guy</first>
  <von space-after-f="no">d'</von>
  <last>Antin</last>
</personname>
```

Figure 2: MlBibTEX's internal format.

2.2 Considerations about sorting

In Hufflen (2014), we mentioned that BibTEX was only able to perform *lexicographic* sorts, the YEAR field is just concatenated to other information. An analogous problem complicates the use of organism names as authors or editors. The following specification, using only a *Last* part:

```
AUTHOR = {{\GuIT}}
```

aims to put the GuIT logo of the Italian TEX Users Group as the author of a collective work. Unfortunately, the commands at the beginning of a BibTEX token are pruned when such a token is used as a sort key, as done about the \MakeUppercase command mentioned in § 1. As a consequence, the sort key derived from this specification is empty.

3 The solutions of biblatex

The biblatex package did not extend the look of existing fields, but added new ones. This is successful for sort operations since some information can be redefined at the sort step. For example, the fields SORTNAME and SORTYEAR take precedence over the fields AUTHOR and YEAR. In particular, the example given in § 2.2 can be extended as follows:

```
AUTHOR = {{\GuIT}},
SORTNAME = {GuIT}
```

whereas MlBibTEX's solution is:

```
AUTHOR = {org => \GuIT,
          sortingkey => GuIT}
```

The other problems mentioned above are not handled by the biblatex package. In addition, let us remark that biblatex end-users also know some backward-compatibility problems if they are to revert to 'old' BibTEX. A simple example is given by the DATE field, which extends the specification of dates—you can specify not only a simple date, but also a *range* of dates—: this field takes precedence over the predefined fields YEAR and MONTH. Obviously, these end-users prefer to use this DATE field, which increases the expressive power of .bib files, but its drawback is that it is not recognised by standard bibliography styles.

4 Discussion and conclusion

BibTEX successors have developed interesting extensions, but often these extensions are mutually incompatible. That is regrettable, but we may think that these experimentation steps should lead to a new format for bibliographical entries in the near future. Maybe the .bib format reached its limits and it is now difficult to extend it. So, the best solution for a new bibliography format could be based on another syntax, e.g., XML[10]. In particular, this format should implement some LaTeX command used within .bib files, in order to ease their translation into other formats, as mentioned in § 1. Obviously, it should provide solutions to problems described in § 2.

When we designed MlBibTEX, we decided for such an internal format, XML-based. We have been able to define our extensions, implement the features of interest for us. Since we would like to get access to these extensions, we defined some concrete syntax, as extensions within .bib files. Most often, but not always. For example, we can specify that there should be no space between a name's *von* and *Last* part in our internal format—as shown in Fig. 2—but we have not proposed yet some concrete syntax for this point within .bib files[11]. If a new format is defined with a comparable expressive power, we think that our functions could be working; we would have just to translate this new format into our internal one. We hope that such a new standard for bibliography database files will be carried out. In the meantime, we show that some syntactic extensions of .bib files—even if they are not backward compatible—can lead to interesting and useful results.

Acknowledgements

I thank Claudio Beccari for his Italian translations of the abstract and keywords.

References

Robert J. Chassell and Richard M. Stallman. *Texinfo. The* GNU *Documentation System. Version 4.13*, September 2008. http://www.gnu.org/software/texinfo.

Hans Hagen. *ConTEXt, the Manual*. http://www.pragma-ade.com/general/manuals/cont-enp.pdf, November 2001.

Jean-Michel Hufflen. Names in BibTEX and MlBibTEX. TUG*boat*, 27(2):243–253, November 2006. TUG 2006 proceedings, Marrakesh, Morocco.

Jean-Michel Hufflen. Dealing with ancient works in bibliographies. *ArsTEXnica*, 18:81–86, October 2014. In Proc. GUIT meeting 2014.

Jean-Michel Hufflen. MlBibTEX 1.4: the new version. *ArsTEXnica*, 20:35–39, October 2015. In Proc. GUIT meeting 2015.

Jean-Michel Hufflen. Abbreviating first names. In *Proc. BachoTEX 2016*, April 2016.

Philip Kime and François Charette. *biber. A Backend Bibliography Processor for biblatex. Version biber 1.9 (biblatex 2.9)*. http://ftp.oleane.net/pub/CTAN/biblio/biber/documentation/biber.pdf, May 2014.

Philipp Lehman, Philip Kime, Audrey Boruvka, and Joseph Wright. *The biblatex Package. Programmable Bibliographies and Citations. Version 2.9a*. http://ctan.mirrorcatalogs.com/macros/latex/contrib/biblatex/doc/biblatex.pdf, June 2014.

Frank Mittelbach and Michel Goossens, with Johannes Braams, David Carlisle, Chris A. Rowley, Christine Detig, and Joachim Schrod. *The LaTeX Companion*. Addison-Wesley Publishing Company, Reading, Massachusetts, 2 edition, August 2004.

Oren Patashnik. *BibTEXing*. Part of the BibTEX distribution, February 1988.

Brian Keith Reid. SCRIBE document production system user manual. Technical report, Unilogic, Ltd., 1984.

Norman Walsh. *DocBook 5. The Definitive Guide*. O'Reilly & Associates, Inc., May 2010. Edited by Richard L. Hamilton.

▷ Jean-Michel Hufflen
FEMTO-ST (UMR CNRS 6174) &
University of Franche-Comté,
16, route de Gray,
25030 BESANÇON CEDEX
FRANCE
jmhuffle at femto-st dot fr

10. e**X**tensible **M**arkup **L**anguage.

11. More precisely, we have not found yet a satisfying solution. Let us go back to Fig. 2, the `spacing-after-f` attribute obviously defaults to `yes`, that is, if a bibliography style orders the insertion of a space character after a *von* part, this space character should not be removed.