The Asian Journal of TEX, Volume 2, No. 1, April 2008
Article revision 2008/3/15

Extending MlBibTeX to Asian Languages: Some Directions

Jean-Michel Hufflen

LIFC (EA CNRS 4157) — University of Franche-Comté
16, route de Gray, 25030 Besançon CEDEX, France
hufflen@lifc.univ-fcomte.fr

KEYWORDS LaTeX, BibTeX, MlBibTeX, multilingual bibliographies, bst, XML, XSLT, nbst, Asian languages.

ABSTRACT MlBibTeX is a reimplementation of BibTeX with particular focus on multilingual features. The current version deals with most of European languages and here we point out the problems we have to face in order to extend this program to Asian languages. We show that MlBibTeX's expressive power allows us to envisage this extension and discuss the open ways, with some examples using the Korean language.

0 Introduction

It is well-known that the 'References' section of a printed document can be done manually, but such an approach leads to texts difficult to maintain and reuse, because they are tightly bound to *bibliography styles*. If we consider bibliographies of English documents, a publisher or anthology editor might like authors' last names to be typeset using small capitals, whereas another publisher would require the use of standard Roman letters for these last names. Likewise, first names may be abbreviated or put *in extenso*, w.r.t. the bibliography style used. We see that combining these choices quickly leads to a combinatorial explosion. In addition, this 'manual' approach is error-prone: if a bibliography is *unsorted*, that is, if the order of items is the order of first citations of these items throughout the document, some change within the document's body can cause the whole of the bibliography to be reorganized.

In fact, many LaTeX users build 'References' sections by means of the BibTeX bibliography processor: this program is given *citation keys*, searches bibliography database (.bib) files for resources associated with these keys, and arranges them according to a bibliography style, the result being a source file (.bbl file) suitable for LaTeX.

Now, let us come to the ability of processing documents written in languages other than English, 'LaTeX's native language'. Much progress has been accomplished in LaTeX, as we can see by comparing the first and second editions of the *LaTeX Companion*: cf. [3, Ch. 9] and [15, Ch. 9]. In particular, LaTeX is now able to deal with some non-Latin alphabets: Russian [1], Greek [21], Hebrew [15, § 9.4.3], Arabic and Farsi [11], Hindi [24], ... Moreover, some tools suitable for the languages of the Far East have

come out: Hangul TEX and the koTEX package [2], the CJK package [14], pTEX, a TEX engine suitable for Japanese [16], ... On the contrary, BIBTEX has been kept stable for a very long period of time, as mentioned in [15, § 13.1]. Workarounds allow users to overcome some limitations of this tool — some tricks usable for non-English texts are given in [22, pp. 229ff] — but often they consist of inserting LATEX commands into the values associated with BIBTEX fields. Here is an example given in [15, § 13.2.2]. Let us consider the following name of a writer:

```
AUTHOR = {Lester del Rey}
```

Since 'del' is uncapitalized, this word is supposed to be the *particle*, that is, the *von* part, w.r.t. BIBTEX's terminology. The two other capitalized words, 'Lester' and 'Rey', put before and after the *von* part, are supposed to be the first and last names. More precisely, given a person name, BIBTEX recognizes four parts: the *first name*, the *particle*, the *last name*, the *lineage* ('*Senior*', '*Junior*', etc.) The rules followed by BIBTEX when it analyses the parts of a name are explained in detail in [9]. In general, the components of particle only use lowercase letters. However, they are sometimes capitalized, in which case the solution is to use a LATEX command. For example:

```
AUTHOR = {Maria {\MakeTextUppercase{d}e La} Cruz}          (1)
```

The first letter of the group '`...{d}e La`' appears to be lowercase for BIBTEX — so this group is supposed to be the particle — although LATEX will typeset the first letter uppercase. Of course, this works provided that the `\MakeTextUppercase` command is defined.[1] This means that such entries can be used only within bibliographies suitable for LATEX and might be usable for deriving bibliographies for other typeset engines built out of TEX — e.g., ConTEXt [5] or pTEX [16] — but such a trick complicates a conversion of .bib files into HTML[2] pages.

Given these considerations, we have designed and implemented MlBIBTEX — for '**M**ulti**L**ingual BIBTEX' — which aims to be a 'better BIBTEX', especially about multilingual features. Due to its conception, we think that MlBIBTEX should be able to be successfully used for deriving bibliographies in Asian languages: we explain that in Section 1. Then Section 2 points out the problems we have to face in order to extend this program to these languages and discusses the ways we plan to solve them. Finally, our conclusion sketches a workplan.

1 MlBIBTEX's features

A complete description of MlBIBTEX features is given in [7]. This section does not replace it, we only aim to show that most features used within bibliographies written in the Korean language can be easily implemented in MlBIBTEX.

Figure 1 gives an example of a bibliographical entry[3] using MlBIBTEX's syntax. First, we remark that a nicer syntax using keywords may be used for person names, so the example given in (1) could be specified in MlBIBTEX by:

1. This command is provided by the textcase package [15, § 3.1.5].
2. **H**yper**T**ext **M**arkup **L**anguage, the language of Web pages.
3. Precise terminology is used within MlBIBTEX: **entries** are specified in .bib files, and MlBIBTEX builds **references** (in .bbl files when they are to be processed by LATEX).

```
@BOOK{honaker1989a,
        AUTHOR = {first => Michel, last => Honaker},
        TITLE = {[Bronx Ceremonial] : english},
        SERIES = {Le [Commander] : english},
        NUMBER = 1,
        PUBLISHER = {Fleuve Noir},
        NOTE = {[No English translation] ! english
                [Keine deutsche Übersetzung] ! german},
        YEAR = 1989,
        MONTH = dec,
        LANGUAGE = french}
```

FIGURE 1. Bibliographical entry using MlBibTeX's syntax.

```
AUTHOR = {first => Maria, von => De La, last => Cruz}
```

MlBibTeX allows the specification of co-authors, like BibTeX. *Collaborators* can be given after the `with` keyword, as shown by the co-authors and collaborators of the *LaTeX Companion* [15]:

```
AUTHOR = {Frank Mittelbach and Michel Goossens with
          Johannes Braams with David Carlisle with
          first => Chris A., last => Rowley with Christine Detig with
          Joachim Schrod}
```

Second, *annotations* related to natural languages can be used. Let us consider the entry given in Figure 1, the `LANGUAGE` field — which defaults to English — expresses that this book is written in French, so the information given within this entry is in French, except as otherwise specified. Text surrounded by square brackets followed by '**:**' means that a foreign language is used. In our example, the book is in French, but its title uses English words. Our specification is not equivalent to:

```
TITLE = {\foreignlanguage{english}{Bronx Ceremonial}}
```

because the latter is usable only if the `\foreignlanguage` command has been defined in the source text of the document. If the babel package has been loaded [15, § 9.2], the english option must be selected, otherwise an error occurs. On the contrary, the former is not related to particular multilingual packages. More precisely, MlBibTeX detects the languages used throughout a document [8] and puts a `\foreignlanguage` command for this title only if the babel package is loaded with the english option. Otherwise, only a warning message is emitted, but these words may be incorrectly hyphenated. Such a specification of a language change may concern the whole value associated with a field, like in the `TITLE` field of our example, or only a substring, like in the `SERIES` field.

Square brackets followed by the '**!**' character, as in the `NOTE` field, are used for conditional texts. If we use this entry within a bibliography for a document written in German and if this bibliography uses information written in German as far as possible, the corresponding reference will include the text given in German — notice the month name in German, too — :

```
<book id="honaker1989a" language="french">
  <author>
    <name>
      <personname><first>Michel</first><last>Honaker</last></personname>
    </name>
  </author>
  <title>
    <foreigngroup language="english">Bronx Ceremonial</foregingroup>
  </title>
  <publisher>Fleuve Noir</publisher>
  <year>1989</year>
  <month><dec/></month>
  <number>1</number>
  <series>Le <foreigngroup language="english>Commander</foreigngroup></series>
  <note>
    <group language="english">No English translation</group>
    <group language="german">Keine deutsche Übersetzung</group>
  </note>
</book>
```

FIGURE 2. XML form used by MlBIBTEX for the entry given in Figure 1.

---

[1] Michel Honaker. *Bronx Ceremonial*. Nr. 1 *in* Le Commander. Fleuve Noir, Dezember 1989. Keine deutsche Übersetzung.

Successive texts marked up by '`[...] ! ...`' are replaced by an empty string if no language matches. On the contrary, a sequence of '`[...] * ...`' texts cannot yield empty information. For example:[4]

```
AUTHOR = {[James C. Alexander] * english [제임스 시 알렉산더] * korean}
```

would put the author's name in Korean within a bibliography for a document written in Korean, and put it in English otherwise. Let us remark that this is not equivalent to using the `KRAUTHOR` field in the halpha bibliography style included in the koTEX distribution [13] because `AUTHOR` and `KRAUTHOR` may be used both within a reference, but we can get such a behaviour by means of accurate bibliography styles.

When MlBIBTEX parses a .bib file, the result can be viewed as an XML[5] tree. More precisely, this result is conformant to SXML[6] conventions [12], SXML being a representation of XML texts in Scheme,[7] the implementation language of MlBIBTEX. As an example, the entry of Figure 1 can be viewed as the XML text given in Figure 2.

Other projects use converters from the bib format to an XML-like format: [4, 17, 25]. In addition, MlBIBTEX provides a compatibility mode for bibliography styles written

4. Let us assume that we can include Korean characters using a right encoding in .bib files as well as bibliography style files. We go thoroughly into this point in Section 2.
5. e**X**tensible **M**arkup **L**anguage. We assume that readers are familiar with the main outlines of this metalanguage. A good introductory book to it is [18].
6. **S**cheme implementation of XML.
7. A good introductory book to Scheme is [20].

```
<nbst:template match="volume">
  <nbst:text>Vol. </nbst:text>
  <nbst:value-of select="."/>
</nbst:template>

<nbst:template match="volume" language="korean">
  <nbst:value-of select="."/>
  <nbst:text>권</nbst:text>
</nbst:template>
```

FIGURE 3. Language-dependent redefinition of a template in nbst.

using the bst language [15, § 13.6], that is, 'old' bibliography styles of BIBTEX are still usable with MlBIBTEX [8].

If you would like to take as much advantage as possible of the new multilingual features of MlBIBTEX, use nbst:[8] this is a language close to XSLT[9] [23], the language of transformations used for XML texts, but it also provides a kind of inheritance about languages. For example, let us look at Figure 3. The first block could be used to put a number after the 'Vol.' abbreviation, as did in English. By default, this template can be applied to process the volume information of a book, but it can be redefined for the Korean language, as shown by the second template, usable when a reference to a Korean work is formatted. In other words, the mark for a volume number precedes the number itself by default, except when this is redefined, an example being given by the Korean language.

2 Dealing with Asian languages

MlBIBTEX's present version is able to deal with most of European languages. In fact, it has been experienced mostly about languages using the Latin alphabet. It should be probably possible to write bibliographical references using the Greek or Cyrillic alphabet, but we have got only a little feedback until now concerning these non-Latin alphabets. Let us now examine what we have to do in order to extend MlBIBTEX to Asian languages.

- As part of the experience resulting from the development of MlBIBTEX, we think that it is difficult to add syntactic sugar to the conventions used for .bib files. In the future, the best method will be probably the direct use of XML files. Such XML-like syntax is probably more suitable for entries expressed using Asian languages, especially if these languages do not use the Latin alphabet. In this framework, a precise taxonomy of bibliographical entries could be specified by schemas.
- When BIBTEX processes a person name, the parts it recognizes — *first*, *von*, *last*, *junior* [15, § 13.2.2] — clearly originates from American names. As we explained

8. **N**ew **B**ibliography **ST**yles.
9. e**X**tensible **S**tylesheet **L**anguage **T**ransformations.

in [9], BIBTEX's conventions may apply to extra-European names, but often by means of workarounds. On the contrary, our XML-like syntax should be able to express other decompositions for names.[10] A good example is given by Indian names, where a person name may be preceded by the father's name and birthplace. This will allow nicer expressive power, provided that any bibliography style is able to process any person name. There is probably a lot of work about this subject, but we are interested in doing it.

– A present limitation of MlBIBTEX: it only uses the Latin 1 encoding, even if some tricks allows characters belonging to Eastern-European languages to be handled [10]. This point should be easy to fix because Scheme, MlBIBTEX's implementation language, has just been extended and should be able to deal with Unicode texts now [19].

– Other calendars than Gregorian may be used to date bibliographical references. This point should be easy if we derive bibliographies for LATEX, since some packages provide converters from Gregorian dates to other systems [15, § 9.3.3].

– Lexicographical order relations, used to sort bibliographical items according to authors' names, have to be extended. A first specification in MlBIBTEX of language-dependent order relations has been given in [10]. In addition to this work, we have to be able to specify how to sort names originating from Asian countries, when these names are written using their own characters, e.g., Hangul syllabes for the Korean language. Besides, bibliographies may include references belonging to several writing systems, in which case each subset is sorted apart. It seems that there are several ways to globally organize such bibliographies; we are given the following examples:

  · references in Korean, then in Japanese, then in Chinese, and then in Latin,[11]
  · references in Korean, then in Russian, then in Latin, and then in Japanese.

  This problem is partially addressed by the halpha bibliography style included in the kωTEX distribution [13]. This style is able to distinguish Korean and Latin references by means of the encodings used, so it can apply different rules to format these two kinds of references. But BIBTEX's `SORT` function [15, Table 13.7] is only based on character codes: since Korean character codes are numerically greater than codes for Latin characters, Korean references are always put after Latin references. In order words, the halpha bibliography style provides a partial solution, hard to extend and customize. Language markup for .bib files and expressive power for the bibliography styles provided by MlBIBTEX should allow a better specification of ordering different writing systems.

Last, but not at least, 'original' BIBTEX never parses a source .tex file and only reads auxiliary (.aux) files. That is not true for MlBIBTEX: it has to partially parse the preamble of a .tex file in order to know which languages are used throughout a document [8] and the encodings used. An improved version of the babel package should write this

10. As seen in the introduction, the parts of a person name can be introduced by keywords in MlBIBTEX. Defining other keywords suitable for Asian languages could be possible, but as we explain previously, we do not think that introducing new syntactic sugar into .bib files would be good technique. However, if that is preferred by end-users…

11. Here, 'in Latin' means 'in languages written using the Latin alphabet'.

information in auxiliary files. In order to ease the use of MlBibTeX, the packages dealing with Asian languages should do the same. For example, if we consider the kotex package, there are two main ways to use it: either for a text written in Korean, or for a text written in another language with some fragments in Korean, as we do in the present article in English. If such packages are used, we should be able to determine the main language of a document by just looking into .aux files.[12] This main language of a document will give the main language to be used for the corresponding 'References' section.

3 Conclusion — Workplan

When we launched the MlBibTeX project, we wrote a questionnaire about bibliography layout used throughout European countries [6]. To go on with Asian languages, we have written an extended version of this questionnaire, with more questions about the encodings used, the typeset engines built out of TeX for Asian languages, the organization of the fields of person names, and the order relations used to sort bibliographies. At the time of writing, we have most answers concerning the Korean language. We plan to go on with investigating other Asian languages, in order to emphasize the common points before programming.

To sum up, there is a lot to do, but the objective seems to us to be reachable.

Acknowledgements

First of all, thanks to organizing committee of the first Asian TeX conference, since I was welcome to this event. Many thanks to LEE Ki-Hwang, who kindly answered my questionnaire: I am debtful to him about the fragments in Korean included in this article. Thanks to Werner LEMBERG, too, who proof-read the first version.

References

1. A. S. BERDNIKOV, Olga LAPKO, Mikhail KOLODIN, Andrew JANISHEVSKY, and A. BURYKIN, *Cyrillic Encodings for LaTeX2ε Multi-Language Documents*, TUGboat **19** (1998), no. 4, 403–416.
2. Jin-Hwan CHO, *The Passage of Hangul TeX and koTeX*, The Asian Journal of TeX **1** (2007), no. 2, 113–121.
3. Michel GOOSSENS, Frank MITTELBACH, and Alexander SAMARIN, *The LaTeX Companion*, 1st edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 1994.
4. Vidar Bronken GUNDERSEN and Zeger W. HENDRIKSE, *BibTeX as XML Markup*, January 2007. http://bibtexml.sourceforge.net
5. Hans HAGEN, *ConTeXt, the Manual*, November 2001. http://www.pragma-ade.com/general/manuals/cont-enp.pdf

12. In fact, there is a workaround: running `mlbibtex` ⟨*job-name*⟩ `--language=`⟨*language-name*⟩. However, we do not recommend this feature, which should be used only for debug purpose. No check is performed about ⟨*language-name*⟩.

6. Jean-Michel HUFFLEN, *European Bibliography Styles and MlBibTeX*, TUGboat **24** (2003), no. 3, 489–498. EuroTeX 2003 Proceedings, Brest, France.
7. _____, *MlBibTeX's Version 1.3*, TUGboat **24** (2003), no. 2, 249–262.
8. _____, *BibTeX, MlBibTeX and Bibliography Styles*, Biuletyn GUST **23** (2006), 76–80. BachoTeX 2006 Conference.
9. _____, *Names in BibTeX and MlBibTeX*, TUGboat **27** (2006), no. 2, 243–253. TUG 2006 Proceedings, Marrakesh, Morocco.
10. _____, *Managing Order Relations in MlBibTeX*, Proc. EuroBachoTeX 2007 (Jerzy LUDWICHOWSKI, Tomasz PRZECHLEWSKI, and Stanisław WAWRYKIEWICZ, eds.), April 2007, pp. 59–66.
11. Youssef JABA, *The Arabi System — TeX Writes in Arabic and Farsi*, TUGboat **27** (2006), no. 2, 147–153. TUG 2006 Proceedings, Marrakesh, Morocco.
12. Oleg E. KISELYOV, *XML and Scheme*, September 2005. http://okmij.org/ftp/Scheme/xml.html
13. Dohyun KIM, Kangsoo KIM, and Koanghi UN, *koTeX: Korean TeX*, October 2007. http://project.ktug.or.kr/ko.TeX
14. Werner LEMBERG, *The CJK Package: Multilingual Support beyond babel*, TUGboat **18** (1997), no. 3, 214–224.
15. Frank MITTELBACH and Michel GOOSSENS, with Johannes BRAAMS, David CARLISLE, Chris A. ROWLEY, Christine DETIG, and Joachim SCHROD, *The LaTeX Companion*, 2nd edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 2004.
16. Haruhiko OKUMURA, *Japanese TeX. Past, Present, and Future*, Slides for the Asian TeX Conference 2008, Kongju National University, South Korea, January 2008.
17. Chris PUTNAM, *Bibliography Conversion Utilities*, February 2005. http://www.scripps.edu/~cdputnam/software/bibutils/bibutils.html
18. Erik T. RAY, *Learning XML*, O'Reilly & Associates, Inc. January 2001.
19. Michael SPERBER, William CLINGER, R. Kent DYBVIG, Matthew FLATT, Anton VAN STRAATEN, Richard KELSEY, and Jonathan REES, *Revised$^{5.97}$ Report on the Algorithmic Language Scheme — Standard Libraries*, June 2007. http://www.r6rs.org
20. George SPRINGER and Daniel P. FRIEDMAN, *Scheme and the Art of Programming*, The MIT Press, McGraw-Hill Book Company, 1989.
21. Apostolos SYROPOULOS, *LaTeX. Ενας Πληρης για την Εκμαθηση του Συστηματος Στοιχειοθεσιας LaTeX*, Παρατηρητης, 1998.
22. Apostolos SYROPOULOS, Antonis SOLOMITIS, and Nick SOFRONIOU, *Digital Typography using LaTeX*, Springer-Verlag, New York, 2002.
23. W3C, *XSL Transformations (XSLT), Version 1.0*, W3C Recommendation (edited by James Clark), November 1999. http://www.w3.org/TR/1999/REC-xslt-19991116
24. Zdeněk WAGNER, *Babel Speaks Hindi*, TUGboat **27** (2006), no. 2, 176–180. TUG 2006 Proceedings, Marrakesh, Morocco.
25. Thomas WIDMAN, *Bibulus — a Perl XML Replacement for BibTeX*, TUGboat **24** (2003), no. 3, 468–471. EuroTeX 2003 Proceedings, Brest, France.