BibTEX

Making MlBibTEX Fit for a Particular Language. Example of the Polish Language

Jean-Michel HUFFLEN

Abstract

The MlBibTEX project aims to provide a multilingual bibliography program. In this article, we show how to make MlBibTEX's Version 1.3 fit for a particular language. In particular, we explain how bibliographical keywords such that 'and', 'chapter', ... should be defined in this particular language. We also show how to refine bibliography styles. For the BachoTEX conference, we chose the Polish language, nevertheless reading this paper should be suitable for any people who would like to adapt MlBibTEX.

Keywords Bibliographies, multilingual features, BibTEX, MlBibTEX, bibliography styles, nbst.

Streszczenie

Projekt MlBibTEX ma na celu stworzenie wielojezykowego programu bibliograficznego. W tym artykule, pokazujemy jak dostosować MlBibTEX (wersja 1.3) dla dowolnego jezyka. Przede wszystkim, przedstawiamy jak należy definiować bibliograficzne słowa kluczowe takie jak „i“, „rozdział“, ... Pokazujemy także jak udoskonalać style bibliograficzne. Na potrzeby konferencji BachoTEX, wybraliśmy jezyk polski, jakkolwiek zapoznanie sie z ta praca jest wskazane dla każdego, kto chciałby zaadaptować MlBibTEX dla dowolnego jezyka europejskiego.

Słowa kluczowe Bibliografie, wielojezykowość, BibTEX, MlBibTEX, style bibliograficzne, nbst.

0. Introduction

This article is the written and updated version of a talk given at the BachoTEX conference on Friday 2nd May 2003. Since October 2000, we are developing a new version of BibTEX [33], the bibliography program associated with LATEX [30]. This new version is MlBibTEX, for 'MultiLingual BibTEX', because it includes multilingual features.

We assume that readers are quite familiar with LATEX and BibTEX, but in order to make precise our terminology, let us recall that BibTEX uses *bibliography files* containing *entries* and generates a file containing *bibliographical references*, that is, items cited throughout a printed work. MlBibTEX enlarges BibTEX in the sense that it can more easily deal with language changes. It can be used to build bibliographical references in English for a book in English, in French for an article in French, in Polish for a manual in Polish, etc. An example of MlBibTEX entry is given in Figure 1. We can notice the use of a `LANGUAGE` field, that gives the entry's language. Let us also remark the use of the notation '`[...] : polish`' to express that some words belonging to the value of the `NOTE` field are in Polish, even if this note is written in English.

```
@INPROCEEDINGS{lem1964,
        AUTHOR = {Stanislas Lem},
        TITLE = {The Mask},
        BOOKTITLE = {Mortal Engines},
        PUBLISHER = {The Seabury Press},
        ADDRESS = {New York},
        NOTE = {Polish title of the anthology:
                ``[Bajki robot\'{o}w] :
                  polish''},
        YEAR = 1964,
        LANGUAGE = english}
```

Figure 1: Example of a MlBibTEX entry.

If this entry is cited within a printed work, the reference built by MlBibTEX will result in:

> [1] Stanislas Lem. 'The Mask'. In *Mortal Engines*. The Seabury Press, New York, 1964. Polish title of the anthology: 'Bajki robotów'.

after processing by LATEX.

Like BibTEX, our MlBibTEX program uses *bibliography styles*, that rule the layout of bibliographical references. Here and in the bibliography of the present article, we use a 'plain' style, that is, references are labelled with numbers, and the first names of authors are not abbreviated. Other choices are possible: the survey given in [31, § 13.5.1] shows how diverse these bibliography styles are. In this article, we do not go thoroughly into typographic rules used for bibliographies. In fact, they depend on cultural background, as we mention that in [20] shortly and in [24] in a more synthetic way. Readers interested in more 'official' references can find such rules in [7, § 15.54-15.76] for English, [9, § 94] for French, [16, § 12.4.1] for German, [8] for Polish. Here we assume that such rules—including the translation of month names, keywords such that 'and', 'chapter', 'edition', ...—are known and we focus on the way to specify such knowledge within our framework.

The first version of MlBibTeX has been technically described in [18], and we explained our choices about our syntax for multilingual features in [19]. When we designed this first version, we believed that we were able to use the bst language [32], for sake of compatibility with previous versions of BibTeX. So we tried to add as few statements as possible to this language, and in fact, only one statement[1] was needed. But, as we explain in [24], going on according to this way would result in complicated bibliography styles, tedious to develop and maintain. Version 1.2, sketched in [21], was a working version[2] that allowed us to experiment new syntax for authors' and editors' names. Let us consider the following name:

```
{Lyon Sprague de Camp}
```

BibTeX correctly parses it, in the sense that '*de*' is supposed to be a particle (the '*von*' part of a person name, according to BibTeX's terminology [33, § 4]), because it is not capitalised. '*Lyon Sprague*' is the first name, '*Camp*' the last one. If the same rule is applied to this name:

```
{Jean Le Clerc de la Herverie}
```

the result is wrong. The two correct ways[3] to specify this name with BibTeX are:

```
{Jean {Le Clerc de la Herverie}}
{Le Clerc de la Herverie, Jean}
```

In addition to BibTeX's syntax, MlBibTeX allows *keywords* for the four parts of a person name: `first`, `von`, `last`, `junior`:

```
AUTHOR = {first => Jean,
          last => Le Clerc de la Herverie}
```

If the author or editor is not a person, but an organisation name, the org keyword can be used:

```
EDITOR = {org => Bacho{\TeX},
          sortingkey => bachotex}
```

If the `sortingkey` keyword is given a value, this value will be used when bibliographical references are sorted. Here, this organisation name is alphabeticised as '*bachotex*', however it is printed. Since Version 1.2, multilingual switches with a default language are allowed within names:

```
{[Stanisław Lem] * polish
 [Stanislas Lem] * english}
```

so the name of this science-fiction writer will be processed using the correct Polish spelling within a document in Polish, and the other form, used in many other countries, will be considered for documents in other languages.

For bibliography styles, MlBibTeX's Version 2.3 [25] uses a language close to XLST[4]. This language is described in [22, 26]. Here we show how to take advantage of this new feature to specify the instantiation for a particular language. Of course, we have chosen the Polish language for the BachoTeX conference[5]. Readers are assumed to be quite familiar with LaTeX 2ε and in particular with the babel package [4, 5]. We also use some basic notions related to XML[6], especially about XPath and XSLT: they can easily be found in [36, 15, 38, 39], the official documents being [42, 41, 43].

First, we explain how MlBibTeX works and how it looks for files. Then we show how LaTeX's commands that appear within the files generated by MlBibTeX should be defined. In the third section, we show how bibliography styles are organised and why the new language nbst for styles makes easier the design of features related to a particular language. We end with a new entry using the multilingual features of MlBibTeX as much as possible.

1. How MlBibTeX works

From the point of view of an end-user, MlBibTeX behaves exactly like BibTeX in the sense that after processing a source file (.tex file) by LaTeX, users just type:

```
mlbibtex source-file
```

[1] The `REFERENCEDEPENDENT` statement, in order to rule multilinguism [18].

[2] ... and, in fact, a prototype written using the Scheme programming language (cf. [27]).

[3] ... and a workaround given in [31, p. 767], which seems to us to be some *hack*.

[4] **e**X**t**ensible **S**tylesheet **L**anguage **T**ransformations.

[5] In addition, we intentionally chose a language we do not speak, so we would like to show that such an instantiation can be done by any experienced user of LaTeX, with advice from a linguist.

[6] **e**X**t**ensible **M**arkup **L**anguage.

month names	ordinal numbers	other keywords	quotations
\bbljan	\bblfirsto	\bblalii	\bblquotedtitle
\bblfeb	\bblsecondo	\bbland	\endbblquotedtitle
\bblmar	\bblthirdo	\bblbc	
\bblapr	\bblfourtho	\bblchap	
\bblmay	\bblfiftho	\bbled	
\bbljun	\bblst	\bbledby	
\bbljul	\bblnd	\bbledn	
\bblaug	\bblrd	\bbleds	
\bblsep	\bblth	\bbletal	
\bbloct		\bblin	
\bblnov		\bblmasterthesis	
\bbldec		\bblno	
		\bblof	
		\bblp	
		\bblphdthesis	
		\bblpp	
		\bbltechrep	
		\bblvol	
		\bblwith	

Table 1: Additional LaTeX commands used in bibliographies generated by MlBibTeX.

MlBibTeX uses the \bibcite commands stored in the auxiliary file *source-file*.aux, searches the bibliography files for the entries cited throughout the source text, and formats the 'References' section onto the *source-file*.bbl file. The bibliography files to be searched and the bibliography style to be used are respectively specified when LaTeX is applied to the *source-file*.tex file, by means of the two commands \bibliography and \bibliographystyle, as users get used to do with BibTeX.

We wanted the bibliography data bases written according to the 'new' syntax and the 'new' bibliography styles to be usable only by MlBibTeX and not by 'old' BibTeX—unless end-users wish that explicitly—so MlBibTeX extends the search rules of BibTeX[7]. When it is looking for a bibliography data basis (.bib file), it uses first the value of the MLBIBINPUTS environment variable[8]. In case of fail, it uses the value of the two environment variables BIBINPUTS and TEXBIB, in turn. Rules for looking for bibliography style files are analogous, as we see in Section 3.1. More generally, the use of such environment variables by means of the programs of the kpathsea library is precisely described in [1] and [12, § 13.8].

References generated by MlBibTeX may use additional LaTeX commands. The default definitions of these commands—that is, definitions in English—are given in the babelbst.sty file. Definitions for other languages are stored in files:

bbl-*language-name*.sty

When your system is configured, the kpsewhich command, provided by the kpathsea library, should be able to find all these files.

MlBibTeX only deals with the languages specified by the user. Let us consider Entry lem1964 given in Figure 1 and let us assume that it is cited within a document. If this document does not use any multilingual package, the note will appear as follows within the file of references generated by MlBibTeX:

```
Polish title of the anthology:
\begin{bblquotedtitle}Bajki robot\'{o}w
\end{bblquotedtitle}.
```

in which case a message warns users that some foreign words may be incorrectly processed. The note will appear the same if the babel package is loaded without the polish option. If this option [4, § 47] has been selected, the note will appear as:

[7] Readers interested in more details about the *TeX Directory Structure*, used to organise the components of TeX-related software, can refer to [40]. Search rules of 'old' BibTeX are described in [2, § 7.1].

[8] Concerning the way to assign a value to such a variable, more details are given in [28].

```
\@ifpackageloaded{babel}{%
 \addto{\extraspolish}{%
  %%  ------------------------------  Months:
  \def\bbljan{stycze\'{n}}%
  \def\bblfeb{luty}%
  \def\bblmar{marzec}%
  \def\bblapr{kweicie\'{n}}%
  \def\bblmay{maj}%
  \def\bbljun{czerwiec}%
  \def\bbljul{lipiec}%
  \def\bblaug{sierpie\'{n}}%
  \def\bblsep{wrzesie\'{n}}%
  \def\bbloct{pa\'{z}dziernik}%
  \def\bblnov{listopad}%
  \def\bbldec{grudzie\'{n}}%
  %%  ------------------------------  Ordinal numbers:
  \def\bblth{.}%
  %%  ------------------------------  Other keywords:
  \def\bblalii{\emph{alii}}%     (others)
  \def\bbland{i}%
  \def\bblbc{a.Chr.}%            (‘before Christ’, negative years are allowed)
  \def\bblchap{rozdzia{\l}}%
  \def\bbled{redaktor}%          (editor)
  \def\bbledby{zredagowane przez}%   (edited by)
  \def\bbledn{wydanie}%          (edition)
  \def\bbleds{redaktorzy}%       (editors)
  \def\bbletal{\emph{et al.}}&   (and others)
  \def\bblin{w}%
  \def\bblmasterthesis{praca magisterska}%
  \def\bblno{nr}%
  \def\bblp{strona}%             (page)
  \def\bblphdthesis{praca doktorska}%
  \def\bblpp{strony}%            (pages)
  \def\bbltechrep{raport techniczny}%
  \def\bblvol{tom}%
  \def\bblwith{i}%               (now like ‘and’ because of declensions)
  %%  ------------------------------  Quoting some titles:
  \def\bblquotedtitle{%
   \stepcounter{@bbl@quote@depth}%
   \ifnum\value{@bbl@quote@depth}=1\quotedblbase\else\guillemotright\fi%
   \nobreak\ignorespaces}%
  \def\endbblquotedtitle{%
   \ifdim\lastskip>\z@\unskip\fi%
   \ifnum\value{@bbl@quote@depth}=1\textquotedblleft\else\guillemotleft\fi%
   \addtocounter{@bbl@quote@depth}{-1}}%
 }%
}{%%  ----------  If the babel package has not been loaded:
 \@ifpackageloaded{polski}{%%  -----  Another way to write in Polish:
  ...
 }{%%  ----------  If the polski package has not been loaded, either:
 ...
 }%
}
```

Figure 2: Defining additional commands and environments in Polish.

```
\def\save@sf@q#1{\leavevmode\begingroup\edef\@SF{\spacefactor \the\spacefactor}#1\@SF\endgroup}%
\def\quotedblbase{\save@sf@q{\set@low@box{\textquotedblright\/}\box\z@\kern-.04em\allowhyphens}}

\def\ly{\usefont{U}{lasy}{m}{n}}
\def\guillemotsleft{\leavevmode%
 \hbox{\ly\char'050\kern-0.2em\char'050\kern+0.2em}\nobreak\ignorespaces}%
\def\guillemotsright{\nobreak\leavevmode\hbox{\ly\kern+0.2em\char'051\kern-0.2em\char'051}}%
```

Figure 3: A possible way to get Polish opening quotation marks and French *guillemets*.

```
Polish title of the anthology:
\begin{bblquotedtitle}
\foreignlanguage{polish}{Bajki
robot\'{o}w}
\end{bblquotedtitle}.
```

Of course, if the `\foreignlanguage` command [5, § 1 & 12.2] has not been put, it can result in incorrect hyphenation of Polish words, but this behaviour is due to the static character of the babel package: adding options dynamically is impossible[9]. However, this example shows that MlBibTeX deals as far as possible with the languages specified for a document, but cannot add a language just because some foreign words are used within an entry. This example also shows the use of our additional LaTeX environment, `bblquotedtitle`, for quotations. Of course, the default version of this environment—that is, its version in American English—surrounds a text with ‘ ‘‘ ’ and ‘ ’’ ’.

2. Requirements about LaTeX commands

In the following, we assume that you have installed a recent version of LaTeX 2$_\varepsilon$ [30], including a working and recent version of the multilingual babel package [5]. Of course, examples assume that the polish option of this package is available. MlBibTeX's Version 1.3—more precisely, the version written in Scheme [28]—has been finalised using LaTeX's implementation included in TeX Live 8, issued in September 2003 [10], available on CD-ROM[11], DVD[12]-ROM, or at the Web sites of most LUGs[13] of TeX users[14]. More generally, we think that it should work with recent versions, in particular those included in recent distributions of the Linux operating system[15].

To adapt MlBibTeX for your language, you have to provide definitions in your language for the commands given in Table 1: the commands for month names, ordinal numbers, and other keywords[16]. This table also includes two additional commands, `\bblquotedtitle` and `\endbblquotedtitle`, used to open and close the `bblquotedtitle` environment (see above about it). As an example, a possible definition in Polish is given in Figure 2.

Defining these commands would be the only thing to do for a basic adaptation that only uses substitutions, as in Version 1.1 [18]. But we will see how to adapt bibliography styles in § 3.3. First, it can be noticed that some commands of Table 1 can be left undefined, if you are sure that your bibliography styles will never use them. As an example, Figure 2 intentionally does not provide definitions for the underlined commands of Table 1. In order for the definitions of this figure to work, let us notice that the following hypotheses have been assumed.

- These definitions should be stored into a .sty file, because we use the ‘`@`’ character, reserved for LaTeX's internal commands. If it is not the case within a personal adaptation you are doing, surround all these definitions as follows:

  `\makeatletter ... \makeatother`

[9] This point should be improved in future versions [28, 29]. Another possible solution is the use of Ω [34] instead of LaTeX, but we have not yet gone thoroughly into this direction.

[10] A description, in Polish, of the use of TeX Live, based on the 7th version, can be found in [3].

[11] **C**ompact **D**isc **R**ead-**O**nly **M**emory.

[12] **D**igital **V**ersatile **D**isc.

[13] **L**ocal **U**ser **G**roups.

[14] `http://www.tug.org`, or `http://www.gust.org.pl` for the Polish-speaking group.

[15] We tested that with Version 9.0 of SuSE Linux' distribution.

[16] For historical reasons, most of these commands originate from the bibliography styles built by means of the makebst program [10], which is one of the first attempts to interface bibliographical references with the babel package.

cf. [31, § 1.3.3] for a description of these two commands.

- The `\addto` command is provided by the babel package [5, § 6] and used to add definitions of LaTeX's commands specific to a language. Such definitions should be grouped into the body of the `\extras`*`language-name`* command (e.g. `\extraspolish`). The `\addto` command allows the body of a command to be extended, so basic `\extras`*`language-name`* commands are defined within the configuration files of the options of the babel package and can be extended as we do or as any end-user can do.

  It can be viewed that we process such definitions only if the babel package is loaded. In fact, within the standard configuration, all the commands of Table 1 default to English keywords, as abovementioned. If you would like to generate 'References' section only with Polish keywords, without any use of the babel package, define them at the top level. If you use an *ad hoc* package to write in Polish[17], proceed analogously, that is, test whether or not the *ad hoc* package has been loaded. As shown in Figure 2, that can be done by means of the `\@ifpackageloaded` command (cf. [31, § A.4.8]).

- In Figure 2, the `@bbl@quoted@depth` counter is supposed to be defined: that is done within the default configuration of MlBibTeX, that is, in the file babelbst.sty (cf. Section 1). This counter allows quotations enclosed within a quotation to be surrounded by 'second-level' delimiters. In Polish:

  ```
  \begin{bblquotedtitle}
  III Symfonia \begin{bblquotedtitle}
   Symfonia pie\'{s}ni \.{z}a{\l}osnych
   \end{bblquotedtitle}, op.~36
  \end{bblquotedtitle}
  ```

  would produce:

  > „III Symfonia »Symfonia pieśni żałosnych«, op. 36“

```
<inproceedings id="lem1964" language="english">
  <author>
    <name>
      <personname>
        <first>Stanislas</first>
        <last>Lem</last>
      </personname>
    </name>
  </author>
  <title>The Mask</title>
  <booktitle>Mortal Engines</booktitle>
  <publisher>The Seabury Press</publisher>
  <year>1964</year>
  <address>New York</address>
  <note>
    Polish title of the anthology:
    <emph emf="no" quotedbf="yes">
      <foreigngroup language="polish">
        Bajki robotów
      </foreigngroup>
    </emph>
  </note>
</inproceedings>
```

Figure 4: An MlBibTeX entry viewed as an XML tree.

- The `\textquotedblleft` command is known by the present implementation of LaTeX and is for the opening quotation marks in American English (‘\’), that is, the closing quotation marks in Polish. If the fontenc package has been loaded with the T1 encoding[18]—that is, if the Cork encoding is used, with a range of 256 characters including most of accented letters from European languages [13]—the `\quotedblbase` command is defined as the opening quotation marks in Polish (‘ „ ’). If this command is unknown, you can define it as shown in Figure 3: that is an alternative definition of this command which is used when the babel package has been loaded with LaTeX's default encoding[19] [5, § 12.10].

- As the quotation of Henryk Mikołaj Górecki's third symphony shows it, the 'second-level'

[17] As examples, the french [14], german and ngerman [35] packages can be directly used to write in French or German, without being options of the babel package. It seems that Polish-speaking people can use the polski package [11, § F7] this way, but we confess that we experienced it only a little.

[18] LaTeX defaults to the OT1 encoding. It seems that the OT4 encoding is sometimes used for documents in Polish, but here also we confess that we did not experience it.

[19] The commands of Figure 3 can be found in the babel.def file. Locate it by means of the kpsewhich command.

```
<nbst:template name="format.date">
  <nbst:choose>
    <nbst:when test="year">
      <nbst:choose>
        <nbst:when test="month">
          <nbst:apply-templates
            select="month"/>
          <nbst:text> </nbst:text>
          <nbst:apply-templates select="year"/>
        </nbst:when>
        <nbst:otherwise>
          <nbst:apply-templates select="year"/>
        </nbst:otherwise>
      </nbst:choose>
    </nbst:when>
    <nbst:otherwise>
      <nbst:if test="month">
        <nbst:warning>
          There's a month but no year in
          <nbst:value-of select="@id"/>
        </nbst:warning>
        <nbst:apply-templates select="month"/>
      </nbst:if>
    </nbst:otherwise>
  </nbst:choose>
</nbst:template>
```

Figure 5: Putting a date with nbst.

```
<nbst:template match="jan">
  \bbljan{}
</nbst:template>

<nbst:template match="feb">
  \bblfeb{}
</nbst:template>

...
```

Figure 6: Putting months names down with nbst.

quotation marks in Polish are the French quotation marks (French *guillemets*), but reversed: the French opening (resp. closing) quotation mark closes (resp. opens) a second-level Polish quotation. These characters are implemented by the `\guillemotleft` and `\guillemotright` commands, defined if the T1 encoding is used. Figure 3 also gives alternative definitions of these two commands, used when the babel package is loaded with the default encoding.

3. Dealing with bibliography styles

3.1. Looking for bibliography style files

Let us assume that the bibliography style chosen for a printed document is S, that is, the command `\bibliographystyle{`S`}` has been included in the source text processed with LaTeX. First MlBibTeX tries to find a 'new' bibliography style—written using the nbst language—so it looks for the file S.nbst using the kpathsea library and the `MLBSTINPUTS` environment variable.

- If the file S.nbst is found, MlBibTeX searches the preamble of the source file (.tex file) in order to know whether or not the babel package is used. If yes, MlBibTeX gets its options[20] L_0, ..., L_n, so it also looks for the files:
  - -L_0.nbst, ..., -L_n.nbst, these bibliography style files containing *general* definitions for the languages L_0, ..., L_n,
  - S-L_0.nbst, ..., S-L_n.nbst, these files being used to store redefinitions concerning the S bibliography style used within the languages L_0, ..., L_n.

  Such files can also be used when some *ad hoc* packages, recognised by MlBibTeX, are used[21].
- Otherwise, MlBibTeX looks for the file S.bst using the value of the `BSTINPUTS` environment variable, and behaves like 'old' BibTeX.

Global definitions, suitable for all the bibliography styles, are stored in the global-mlbibtex.nbst file, that MlBibTeX can locate. In addition, a bibliography style written using the nbst language can include definitions put in another file. This is done by the element `<nbst:include href="F"/>`. If the file name F has no suffix, MlBibTeX looks for first the F.nbst file, second the F.bst file, as described above. If a suffix has been specified, MlBibTeX looks for the specified file only. As we have just suggested it, a bibliography style written using the

[20] In future versions, this point could be improved, and MlBibTeX should be able to get this information by looking into the auxiliary file.

[21] See Footnote 17, p. 19. For an example in Polish, there can coexist polish-plain.nbst and polski-plain.nbst files as Polish redefinitions for the plain bibliography style. The former (resp. latter) is usable when the babel (resp. polski) package is loaded.

nbst language can use definitions written in the bst language. But in this article, we do not use this feature, described in [22, 23, 26].

3.2. The nbst language

As mentioned in the introduction, the nbst language, for 'new bibliography style language', is close to XSLT. In fact, the main difference between nbst and XSLT is the implementation of multilingual aspects, shown in § 3.3. When MlBibTeX runs, .bib files are processed as if they were XML trees, but without whitespace nodes[22] between fields' specifications. Following this approach, Entry `lem1964` given in Figure 1 is viewed as the tree of Figure 4, except that withespace nodes that an XML parser would produce are ruled out. Presently, we can consider that we use a DTD[23]—more precisely, it is a revised version of the DTD given in [20]—but we plan to rewrite the specification of bibliographical entries using XML Schema [46]. In fact, most of fields are represented by elements, except for some particular information modelled by attributes (for example, the key of the bibliographical entry, given by the `id` attribute).

So bibliography styles can access to the values of MlBibTeX fields by means of *path expressions* close to the XPath language. For example, if we assume that the context node is the node of a bibliographical entry, for example, the `inproceedings` node of the tree given in Figure 4, the month and year can be accessed by means of the path expressions `month` and `year`. So formatting the date of a reference can be done by the `format.date` template given in Figure 5. We can notice that the month name is optional[24], and testing it is done by a path expression used as a boolean value, like in XPath.

Like in XSLT, a template of the nbst language either can be named like the `format.date` template, in which case it is called by:

```
<nbst:call-template name="format.date"/>
```

or is run when the path expression given by the `match` attribute matches the current node. The templates given in Figure 6 show such examples: they are invoked since a `month` element is selected when the template `format.date` (see Figure 5 is processed. By the way, we can see how MlBibTeX's keywords for months are related to the commands of Table 1 for month names.

3.3. Customising bibliography styles

Now we propose a more ambitious example, showing the multilingual features of the nbst language. Let us consider the `EDITION` field for a book. Let us assume that its value is a number or an ordinal written as a word: 'first', 'second', ... Given the rules for deriving an ordinal from a number in English, this `EDITION` field can be formatted as Figure 7 specifies it[25]. The given template processes an element like `<edition>1</edition>` or `<edition>first</edition>`. By the way, we can notice some functions related to our path expressions: `firstcapitalize()`, `is-number()`, `lowercase()`. These functions are analogous to functions related to XPath expressions[26]. Since their effect can be easily guessed, we do not describe them here; readers interested in more details can refer to [26].

Let us come back to formatting the information about the edition of a book, this operation can be refined in Polish because roman numerals are used after the keyword[27]: '*Wydanie I*',

[22] They are text nodes whose contents are only whitespace characters, originating from what has been typed between two tags [38, p. 25–26]. For example, if the XML text of Figure 4 is parsed, there is a whitespace node, containing a new-of-line and four space characters between the opening tags `<author>` and `<name>`. XML parsers are bound by the 'all text counts' constraint, included in the XML specification [43, § 2.10], and cannot ignore such whitespace characters.

[23] **D**ocument **T**ype **D**efinition. A DTD can be viewed as a *template* for XML documents. See [36, Ch. 5] for a more complete explanation.

[24] That is specified in our DTD (cf. [20]). The year node is not optional, but checking its presence is not done when the entry is written by a user, so we have to do that within bibliography styles.

[25] We can notice the use of a parameter `mid.sentence` (there is a namesake in BibTeX)—set to a boolean value that defaults to the *false* value—to test whether or not we are at the beginning of a sentence, that is, whether or not the first word we put down has to be capitalised.

[26] These operations are difficult to perform with the functions provided by the first version of XPath, that is why we added them. Dealing with strings and types should be easier in Version 2.0 of XPath [45], so a future version of MlBibTeX might be fully conformant with path expressions used in XML.

[27] In comparison with what we wrote at the end of the introduction of [28]—'*1. wydanie*', '*2.*

```
<nbst:template match="edition">
  <nbst:param name="mid.sentence" select="false()"/>
  <nbst:choose>
    <nbst:when test="is-number()">
      <nbst:choose>
        <nbst:when test=".=1">\bblfirsto\</nbst:when>
        ...
        <nbst:when test=".=5">\bblfiftho\</nbst:when>
        <nbst:when test=". != 11 and . mod 10=1">
          <nbst:value-of select="."/><nbst:text>\bblst\</nbst:text>
        </nbst:when>
        ...
        <nbst:when test=". !=13 and . mod 10=3">
          <nbst:value-of select="."/><nbst:text>\bblrd\</nbst:text>
        </nbst:when>
        <nbst:otherwise><nbst:value-of select="."/><nbst:text>\bblth\</nbst:text></nbst:otherwise>
      </nbst:choose>
    </nbst:when>
    <nbst:when test="$mid.sentence"><nbst:value-of select="lowercase()"/></nbst:when>
    <nbst:otherwise><nbst:value-of select="firstcapitalize()"/></nbst:otherwise>
  </nbst:choose>
  <nbst:text> \bbledn{}</nbst:text>
</nbst:template>
```

Figure 7: Formatting the information about the edition of a book: default version.

'*Wydanie II*', . . . Associated with the \bblth command given in Figure 2 for the Polish language, the template given in Figure 8 does that[28]. Since it includes a language attribute, its priority is higher than the template matching edition elements given in Figure 7, which has no such attribute.

3.4. Using modes

Like in XSLT [42, § 5.7], the templates of nbst can use *modes*. When a mode attribue is used within an nbst:apply-templates tag, only a template with the same mode can be instantiated. A very useful example of this feature is given by month names in Polish, since different cases[29] are used for a date only including the month and year (e.g. '*styczeń*

```
<nbst:template match="edition"
               language="polish">
  <nbst:param name="mid.sentence"/>
  <nbst:text>Wydanie </nbst:text>
  <nbst:choose>
    <nbst:when test="is-number()">
      <nbst:text> \Romannumeral{</nbst:text>
      <nbst:value-of select="."/>
      <nbst:text>}</nbst:text>
    </nbst:when>
    <nbst:when test="$mid.sentence">
      <nbst:value-of select="lowercase()"/>
    </nbst:when>
    <nbst:otherwise>
      <nbst:value-of select="."/>
    </nbst:otherwise>
  </nbst:choose>
</nbst:template>
```

Figure 8: Formatting the information about the edition of a book: refined version in Polish.

wydanie', . . .—this way is more common. Thanks to Paweł D. Mogielnicki, who signalled me that.

[28] . . . provided that the \Romannumeral command is defined. That can be done by:

```
\makeatletter
\def\Romannumeral#1{\@Roman{#1}}
\makeatother
```

in the preamble. The command \@Roman cannot be put directly within a text (cf. [31, § 1.3.3]).

[29] A language uses *cases* when a word's function is directly expressed within this word, most often by a suffix. The Polish language uses such a system. To illustrate it with an example in Latin, 'Peter loves Lydia' is translated into this language by '*Petrus Lydiam amat*', and the word order is irrelevant, '*Lydiam Petrus amat*' has the same meaning.

```
@STRING{lem = "first => [Stanis{\l}aw] * pol [Stanislas] * english, last => Lem"}

@STRING{warsaw = "[Warszawa] * pol [Warsaw] * english [Varsovie] * fr [Warschau] * german"}

@BOOK{lem1968,
        AUTHOR = lem,
        TITLE = {G{\l}os Pana},
        PUBLISHER = {Czytelnik},
        ADDRESS = warsaw,
        EDITION = 1,
        NOTE = {[Title of the English translation: ``His Master's Voice''] ! english
                [Titre de la traduction fran\c{c}aise : ``La voix du ma\^{\i}tre''] ! french
                [Titel den deutschen \"{U}bersetzung: ``Die Stimme des Herrn''] ! german},
        YEAR = 1968}
```

Figure 9: Example of a multilingual entry.

2003' for 'January 2003') and a date with a day number (e.g. '*1 stycznia 2003*' for '1st January 2003'). Month names can be defined for the genitive case this way:

```
<nbst:template match="jan"
               language="polish"
               mode="genitive">
  stycznia
</nbst:template>
```

... and when a date with a day number is to be printed, such templates can be invoked by:

```
<nbst:apply-templates select="month"
                      language="polish"
                      mode="genitive"/>
```

('language="polish"' can be omitted if this tag is put inside a template with the language attribute already set to "polish").

4. A multilingual example

This section aims to give an ambitious example of using multilingual features of MlBibTeX. Let us consider again a book written by Stanislas Lem and published at Warsaw. In the introduction, we already noticed that multilingual switches are allowed within names. In such a case, there must be a default, that is, the result must never be empty, so we use the ∗-notation [18]. We can mix it with the new keywords used for the parts of a person name, as shown in Figure 9. Within this figure, we intentionally put 'pol' down, instead of 'polish'. MlBibTeX recognises non-ambiguous prefixes of the babel package's options or *ad hoc* package names. This convention allows the entry given in Figure 9 to be suitable when the polish option of the babel package has been selected as well as when the polski package is used.

In BibTeX, the @STRING directive is generally used to define abbreviations for journal names[30] [31, Table 13.3]. Such abbreviations are expanded when .bib files are parsed by BibTeX. In MlBibTeX, such @STRING directives are allowed and can be viewed as XML entities [36, p. 45–50], that is, placeholders for contents. The value of such an entity can be an XML subtree; in the same way, MlBibTeX's abbreviations introduced by a @STRING directive can include multilingual switches. So do the abbreviations lem and warsaw in Figure 9. These abbreviations are expanded when the .bib file is parsed by MlBibTeX, so the names 'lem' and 'warsaw' themselves are ignored in bibliography styles.

You may wish to define such abbreviations within a bibliography style file, analogously with the *macros* of the bst language—let us recall that month names and abbreviations known by BibTeX for journal names are specified this way—in which case you can use an nbst:variable element, as shown in Figure 10. As other examples, abbreviations for well-known journal names are specified within the global-mlbibtex.nbst file, grouping global definitions. Let us notice that the corresponding macros in BibTeX are defined in *each* file implementing a bibliography style, whereas MlBibTeX allows us to specify once such parameters

Expressing 'Lydia loves Peter' is '*Lydia Petrum amat*'.

[30] For example, 'acmcs' for 'ACM *Computing Surveys*' or 'acta' for '***ACTA*** *Informatica*'.

```
<nbst:variable name="krakow">
  <nonemptyinformation>
    <!-- This element is for information that must
         be put, possibly in another language if
         there is no translation into the current
         language.
      -->
    <group language="pol">Kraków</group>
    <group language="english">Krakow</group>
    <group language="fr">Cracovie</group>
    <group language="german">Krakau</group>
  </nonemptyinformation>
</nbst:variable>
```

Figure 10: Adding multilingual information into bibliography styles.

in only one file. When an abbreviation name—not surrounded by braces or double-quotes, as in BibTeX—cannot be expanded when the .bib file is parsed, because there is no corresponding @STRING directive, it is modelled by such a symbol element:

```
<address><symbol name="krakow"/></address>
```

and can be processed this way within a bibliography style:

```
<nbst:template match="symbol">
  <nbst:if test="is-defined(@name)">
    <nbst:apply-templates
      select="node-set(${@name})"/>
  </nbst:if>
</nbst:template>
```

Let us notice that this template could not be directly translated into XSLT, because it uses extensions to this language:

- the is-defined function, which checks whether or not a parameter has been defined,
- the node-set function, used to build a set of nodes from a fragment of an XML tree[31].
- an extended use of the '{...}' notation, to evaluate a name dynamically.

5. Conclusion

As readers of this article can see, MlBibTeX provides good compatibility with 'old' BibTeX in the sense that 'old' .bib files can be reused, except that square brackets ('[', ']') now have syntactic meaning. Of course, users can enrich .bib files with MlBibTeX's new features. About bibliography styles, we think that programming and enriching them using the nbst language is easier than in the 'old' bst language. In particular, users who know the XSLT language should learn nbst without difficulty. Concerning multilingual features and an instantiation for a particular language, we use, as abovementioned, implicit rules for looking for files related to a particular language. Such rules do not exist in XSLT, where each importation of definitions must be explicit, but in MlBibTeX, they allow modular management of language-dependent definitions. Users who only know MlBibTeX's global organisation can write or redefine features related to their own languages without changing files devoted to other instantiations.

We think that our approach is suitable at least for European languages [24]. A lot of work will be probably needed for all the languages in the world, and especially for Asian languages, but studying this point belongs to the future. In addition, since no one can claim to know all the languages, it is important that other people than MlBibTeX's programmers can write an instantiation for a particular language.

On another point, it seems that BibTeX has been used to generate 'References' sections only for Scribe [37] and LaTeX, although the bst language is not directly related to these two word processors. In the future, we plan to write MlBibTeX's bibliography styles suitable for ConTeXt, Hans Hagen's format [17], or for Texinfo [6], used for the documentation of GNU[32] software. Besides, the nbst language opens a window to the world of XML and should make easier the writing of XML documents: examples are Web pages in XHTML[33] or XSL-FO[34] documents [44]. In particular, MlBibTeX should be suitable for building bibliographies for documents written according to the XML version of the DocBook [47] DTD, we plan to explore this direction [29].

[31] In fact, some XSLT processors provide it, but it has not been included in the 'official' specification of XSLT [42]. It belongs to the additional functions of the EXSLT ('**E**xtensions to XSLT') project [38, Appendix A].

[32] GNU's **N**ot **U**nix.

[33] e**X**tensible **H**yper**T**ext **M**arkup **L**anguage, that is, the XML version of the HTML language.

[34] e**X**tensible **S**tylesheet **L**anguage—**F**ormatting **O**bjects.

Acknowledgements

I have been able to write this article since Paweł D. Mogielnicki kindly answered all my questions about bibliographies in Polish. He also wrote the Polish translation of the abstract, I am very grateful to him. I also thank Janusz S. Bień, who corrected some mistakes in titles of Polish works.

References

[1] Karl Berry and Olaf Weber: *Kpathsea library for Version 3.3.7.* November 2001. Part of LaTeX' distribution.

[2] Karl Berry and Olaf Weber: *Web2c for Version 7.3.7.* November 2001. Part of LaTeX' distribution.

[3] Janusz S. Bień: „TeXLive w sieci. Koncepcja i wykorzystanie". *Biuletyn Polskiej Grupy Użytkowników Systemu TeX*, tom 19, strony 13–21. maj 2003.

[4] Joannes Braams: *Babel, a Multilingual Package for Use with LaTeX's Standard Document Classes.* Version 3.7. February 2001. CTAN:macros/latex/required/babel/babel.dvi.

[5] Joannes Braams: *Babel, a Multilingual Package for Use with LaTeX's Standard Document Classes.* Version 3.7. May 2002. CTAN:macros/latex/required/babel/babel.dvi.

[6] Robert J. Chassell and Richard M. Stallman: *Texinfo. The* GNU *Documentation Format.* Part of texinfo-4.4 distribution. January 2003.

[7] *The Chicago Manual of Style.* The University of Chicago Press. The 14th edition of a manual of style revised and expanded. 1993.

[8] Robert Chwałowski: *Typografia typowej ksiażki.* Wydawnictwo Helio, Gliwice. 2002.

[9] *Code typographique. Choix de règles à l'usage des professionnels du livre.* Fédération nationale du personnel d'encadrement des industries polygraphiques et de la communication, Paris. 17^e^ édition. 1993.

[10] Patrick W. Daly: *Customizing Bibliographic Style Files.* Version 3.2. February 1999. Part of LaTeX' distribution.

[11] Antoni Diller: *LaTeX wiersz po wierszu.* Wydawnictwo Helion, Gliwice. Polish translation of *LaTeX Line by Line* with an additional annex by Jan Jełowicki. 2001.

[12] Wietse Dol and Erik Frambach: *4TeX for Windows.* NTG. June 1999.

[13] Michael J. Ferguson: "Report on Multilingual Activities". *TUGboat*, Vol. 11, no. 4, p. 514–516. November 1990.

[14] Bernard Gaulle : *Notice d'utilisation du style french multilingue pour LaTeX.* Version pro V5.01. Janvier 2001. CTAN:loria/language/french/pro/french/ALIRE.pdf.

[15] Michel Goossens: XML—*A New Start for the Web.* May 2000. http://webcast.cern.ch/Projects/WebUniversity/AcademicTraining/Goossens%/.

[16] Jurgen Gulbins und Christine Kahrmann: *Mut zur Typographie.* 2. Auflage. Springer-Verlag. 2000.

[17] Hans Hagen: "The Status Quo of the NTS Project". *Cahiers GUTenberg*, Vol. 39–40, p. 205–220. May 2001.

[18] Jean-Michel Hufflen: "MlBibTeX: a New Implementation of BibTeX". In: *EuroTeX 2001*, p. 74–94. Kerkrade, The Netherlands. September 2001.

[19] Jean-Michel Hufflen: "Lessons from a Bibliography Program's Reimplementation". In: LDTA *2002*, Vol. 65.3 of *ENTCS*. Elsevier, Grenoble, France. April 2002.

[20] Jean-Michel Hufflen: "Multilingual Features for Bibliography Programs: from XML to MlBibTeX". In: *EuroTeX 2002*, p. 46–59. Bachotek, Poland. April 2002.

[21] Jean-Michel Hufflen: *Towards MlBibTeX's Versions 1.2 & 1.3.* MaTeX Conference. Budapest, Hungary. November 2002.

[22] Jean-Michel Hufflen: *Die neue Sprache für MlBibTeX.* DANTE 2003, Bremen. April 2003.

[23] Jean-Michel Hufflen: "Mixing Two Bibliography Style Languages". In: LDTA *2003*, Vol. 82.3 of *ENTCS*. Elsevier, Warsaw, Poland. April 2003.

[24] Jean-Michel Hufflen: "European Bibliography Styles and MlBibTeX". *TUGboat*, Vol. 24, no. 3. EuroTeX 2003, Brest, France. June 2003.

[25] Jean-Michel Hufflen: *MlBibTeX—Multilingual BibTeX.* July 2003. http://

`lifc.univ-fcomte.fr/~hufflen/texts/mlbibtex/mlbibtex/`.

[26] Jean-Michel HUFFLEN: "MlBibTeX's Version 1.3". *TUGboat*, Vol. 24, no. 2, p. 249–262. TUG 2003, Outrigger Waikoloa Beach Resort, Hawaii. July 2003.

[27] Jean-Michel HUFFLEN: "Order Relations and European Languages. Extended Abstract". In: WADT *2004*, p. 53–54. Barcelona, Spain. March 2004.

[28] Jean-Michel HUFFLEN: "A Tour around MlBibTeX and Its Implementation(s)". *Biuletyn* GUST, Vol. 20, p. 21–28. In *BachoTeX conference*. April 2004.

[29] Jean-Michel HUFFLEN: "MlBibTeX: beyond LaTeX". In: *Preprints for the 2004 Annual Meetings*, p. 77–84. TUG, Xanthi, Greece. August 2004.

[30] Leslie LAMPORT: *LaTeX. A Document Preparation System. User's Guide and Reference Manual*. Addison-Wesley Publishing Company, Reading, Massachusetts. 1994.

[31] Frank MITTELBACH and Michel GOOSSENS, with Joannes BRAAMS, David CARLISLE, Chris A. ROWLEY, Christine DETIG and Joachim SCHROD: *The LaTeX Companion*. 2nd edition. Addison-Wesley Publishing Company, Reading, Massachusetts. August 2004.

[32] Oren PATASHNIK: *Designing BibTeX Styles*. February 1988. Part of LaTeX' distribution.

[33] Oren PATASHNIK: *BibTeXing*. February 1988. Part of LaTeX' distribution.

[34] John PLAICE and Yannis HARALAMBOUS: *Draft Documentation for the Ω System*. March 1998. `http://www.loria.fr/services/tex/english/moteurs.html`.

[35] Bernd RAICHLE: *Die Makropakete „german" und „ngerman" für LaTeX 2ε, LaTeX 2.09, Plain-TeX and andere darauf Basierende Formate*. Version 2.5. Juli 1998. Im Software LaTeX.

[36] Erik T. RAY: *Learning* XML. O'Reilly & Associates, Inc. January 2001.

[37] Brian Keith REID: SCRIBE *Document Production System User Manual*. Technical Report, Unilogic, Ltd. 1984.

[38] John E. SIMPSON: *XPath and XPointer*. O'Reilly & Associates, Inc. August 2002.

[39] Doug TIDWELL: XSLT. O'Reilly & Associates, Inc. August 2001.

[40] TUG Working Group on a TeX Directory Structure: *A Directory Structure for TeX Files. Version 0.9995*. `CTAN:tex/archive/tds/standard/tds-0.9995/tds.dvi`. January 1998.

[41] W3C: XML *Path Language (XPath). Version 1.0*. W3C Recommendation. Edited by James Clark and Steve DeRose. November 1999. `http://www.w3.org/TR/1999/REC-xpath-19991116`.

[42] W3C: XSL *Transformations (*XSLT*). Version 1.0*. W3C Recommendation. Written by Sharon Adler, Anders Berglund, Jeff Caruso, Stephen Deach, Tony Graham, Paul Grosso, Eduardo Gutentag, Alex Milowski, Scott Parnell, Jeremy Richman and Steve Zilles. November 1999. `http://www.w3.org/TR/1999/REC-xslt-19991116`.

[43] W3C: *Extensible Markup Language (*XML*) 1.0 (Second Edition)*. W3C Recommendation. Edited by Tim Bray, Jean Paoli, C. M. Sperberg-McQueen and Eve Maler. October 2000. `http://www.w3.org/TR/2000/REC-xml-20001006`.

[44] W3C: *Extensible Stylesheet Language (*XSL*). Version 1.0*. W3C Recommendation. Edited by James Clark. October 2001. `http://www.w3.org/TR/2001/REC-xsl-20011015/`.

[45] W3C: XML *Path Language (XPath) 2.0*. W3C Working Draft. Edited by Anders Berglund, Scott Boag, Don Chamberlin, Mary F. Fernandez, Michael Kay, Jonathan Robie and Jérôme Siméon. November 2002. `http://www.w3.org/TR/2002/WD-xpath20-20021115`.

[46] W3C: XML *Schema*. November 2003. `http://www.w3.org/XML/Schema`.

[47] Norman WALSH and Leonard MUELLNER: *DocBook: the Definitive Guide*. O'Reilly & Associates, Inc. October 1999.

◇ Jean-Michel HUFFLEN
hufflen@lifc.univ-fcomte.fr
LIFC, University of Franche-Comté.
16, route de Gray. 25030
BESANÇON CEDEX.
FRANCE.