MlBibTEX 1.4: the New Version

Jean-Michel HUFFLEN
DISC — University of Franche-Comté

Trento, 17th October 2015

Contents

Bibliographies presently

BibTeX is ageing. . .

Bibliographies presently

BibTeX is ageing. . .

biber usable with the biblatex package,

Bibliographies presently

BibTeX is ageing. . .

biber usable with the biblatex package,

interesting extensions, e.g., the `DATE` field:

```
DATE = {2015-10-17/2015-10-18}
```

Beyond BibTeX

What happens if an end-user of biblatex has to revert to BibTeX?

Beyond BibTEX

What happens if an end-user of biblatex has to revert to BIBTEX?

What about ConTEXt?

Beyond BibTEX

What happens if an end-user of biblatex has to revert to BIBTEX?

What about ConTEXt?

After BIBTEX $\Longleftarrow$ incompatible extensions.

MlBibTEX

Reimplementation of BibTEX using the Scheme programming language,

MlBibTEX

Reimplementation of BibTEX using the Scheme programming language,

with particular focus on multilingual features, provides some syntactical extensions,

MlBibTeX

Reimplementation of BibTeX using the Scheme programming language,

with particular focus on multilingual features, provides some syntactical extensions,

based on an XML-like format for bibliographical items,

MlBibTEX

Reimplementation of BibTEX using the Scheme programming language,

with particular focus on multilingual features, provides some syntactical extensions,

based on an XML-like format for bibliographical items,

includes support for ConTEXt and biblatex,

MlBibTEX

Reimplementation of BibTEX using the Scheme programming language,

with particular focus on multilingual features, provides some syntactical extensions,

based on an XML-like format for bibliographical items,

includes support for ConTEXt and biblatex,

and other applications.

MlBibTEX

Reimplementation of BIBTEX using the Scheme programming language,

with particular focus on multilingual features, provides some syntactical extensions,

based on an XML-like format for bibliographical items,

includes support for ConTEXt and biblatex,

and other applications.

All the programs may take advantage of an extension (e.g., `DATE`).

Feedback

Not used widely, but users are satisfied, as far as I know.

Feedback

Not used widely, but users are satisfied, as far as I know.

Less permissive than BibTeX. For example, a `YEAR` field *must* be an integer,

Feedback

Not used widely, but users are satisfied, as far as I know.

Less permissive than BibTeX. For example, a `YEAR` field *must* be an integer, possibly negative,

Feedback

Not used widely, but users are satisfied, as far as I know.

Less permissive than BIBTEX. For example, a `YEAR` field *must* be an integer, possibly negative, possibly *inexact* (cf. GUIT 2014).

Why a new version?

Even if MlBibTeX 1.3 is based on a canonical representation of all accented letters, it only deals with Latin 1 encoding:

Why a new version?

Even if MlBibTeX 1.3 is based on a canonical representation of all accented letters, it only deals with Latin 1 encoding:

Łódż $\Longrightarrow$ `{\L}ó\.{z}`

Why a new version?

Even if MlBibTeX 1.3 is based on a canonical representation of all accented letters, it only deals with Latin 1 encoding:

Łódż $\Longrightarrow$ `{\L}ó\.{z}`

whereas modern TeX engines—e.g., ConTeXt—are based on UTF-8.

Why a new version?

Even if MlBibTeX 1.3 is based on a canonical representation of all accented letters, it only deals with Latin 1 encoding:

$$\text{Łódż} \Longrightarrow \texttt{\{\textbackslash L\}ó\textbackslash .\{z\}}$$

whereas modern TeX engines—e.g., ConTeXt—are based on UTF-8.

Now, Scheme's new standard is Unicode-compliant, so MlBibTeX can be, too.

Why a new version?

Even if MlBibTeX 1.3 is based on a canonical representation of all accented letters, it only deals with Latin 1 encoding:

Łódż $\Longrightarrow$ `{\L}ó\.{z}`

whereas modern TeX engines—e.g., ConTeXt—are based on UTF-8.

Now, Scheme's new standard is Unicode-compliant, so MlBibTeX can be, too.

The interface with Scheme should be more customisable.

MlBibTeX 1.4 and encodings

First release $\Longleftarrow$ byte-based encodings will be processed: Latin-1, Latin-2, UTF-8. . .

MlBibTEX 1.4 and encodings

First release $\Longleftarrow$ byte-based encodings will be processed: Latin-1, Latin-2, UTF-8. . . but not UTF-16.

MlBibTeX 1.4 and encodings

First release $\Longleftarrow$ byte-based encodings will be processed: Latin-1, Latin-2, UTF-8... but not UTF-16.

You can make precise the encoding at the beginning of a .bib file:

```
%encoding = latin-1
```

MlBibTeX 1.4 and encodings

First release $\Longleftarrow$ byte-based encodings will be processed: Latin-1, Latin-2, UTF-8. . . but not UTF-16.

You can make precise the encoding at the beginning of a .bib file:

```
%encoding = latin-1
```

(Another directive, `%prefix`, allows name clashes to be avoided.)

Bibliography database files

.bib files, but also XML files.

Bibliography database files

.bib files, but also XML files.

JSON and Refer formats $\Longleftarrow$ planned.

Rules for names

Accented letters will be allowed only in values associated with fields.

Rules for names

Accented letters will be allowed only in values associated with fields.

Field names (`AUTHOR`, . . .) and entry types (`@ARTICLE`, . . .) $\Longrightarrow$ the same rule holds.

Initialisation file

```
((encodings-pv 'set-default-4-bib-files)
 'utf-8)
```

(or accept the predefined default $\Longrightarrow$ `latin-1`).

Initialisation file

```
((encodings-pv 'set-default-4-bib-files)
 'utf-8)
```

(or accept the predefined default $\Longrightarrow$ `latin-1`).

`mlbibtex` $\longleftarrow$ ~/.mlbibtex

Initialisation file

```
((encodings-pv 'set-default-4-bib-files)
 'utf-8)
```

(or accept the predefined default $\Longrightarrow$ `latin-1`).

`mlbibtex` $\longleftarrow$ ~/.mlbibtex

`mlbibcontext` $\longleftarrow$ ~/.mlbibcontext

...

Revision of MlBibTeX's commands

`-encoding` for the programs `mlbibtex` and `mlbibtex2xml`.

Revision of MlBibTeX's commands

`-encoding` for the programs `mlbibtex` and `mlbibtex2xml`.

New argument for the program `mlbiblatex`.

Revision of MlBibTeX's commands

`-encoding` for the programs `mlbibtex` and `mlbibtex2xml`.

New argument for the program `mlbiblatex`.

No change for the program `mlbibcontext`, but the output defaults to the UTF-8 encoding.

Revision of MlBibTeX's commands

`-encoding` for the programs `mlbibtex` and `mlbibtex2xml`.

New argument for the program `mlbiblatex`.

No change for the program `mlbibcontext`, but the output defaults to the UTF-8 encoding.

The other options are still recognised, e.g., the `-inexact` option (cf. GUIT 2014).

What is to be done

Order relations.

What is to be done

Order relations.

Output routine, e.g.:

$$\begin{aligned} \text{Łódż} &\Longrightarrow \texttt{Łódż} && (\text{UTF-8}) \\ &\Longrightarrow \texttt{\{\textbackslash L\}ó\textbackslash .\{z\}} && (\text{Latin-1}) \end{aligned}$$

What is to be done

Order relations.

Output routine, e.g.:

$$
\begin{aligned}
\text{Łódż} &\Longrightarrow \texttt{Łódż} && (\text{UTF-8})\\
&\Longrightarrow \texttt{\{\textbackslash L\}ó\textbackslash.\{z\}} && (\text{Latin-1})
\end{aligned}
$$

Parser.

Now. . . today

‘Semantic’ functions—including order relations—at the end of debugging.

Now. . . today

‘Semantic’ functions—including order relations—at the end of debugging.

Output routine $\Longleftarrow$ in test.

Now. . . today

'Semantic' functions—including order relations—at the end of debugging.

Output routine $\Longleftarrow$ in test.

What is missing $\Longrightarrow$ parser and interface.

Now. . . today

'Semantic' functions—including order relations—at the end of debugging.

Output routine $\Longleftarrow$ in test.

What is missing $\Longrightarrow$ parser and interface.

Installation procedure $\Longrightarrow$ in refurbishment.

Conclusion

I have already reworked and extended MlBibTeX,. . . and succeeded. I am confident.

See you soon for 1.4's first demonstration!