

Programming Bibliographies

Jean-Michel Hufflen

TUG

8 August 2021

Contents

An Original View

Bibliographies

Sorting

Generating labels

Other views

News from
MIBIB_TE_X

Conclusion

An Original View

Bibliographies

Sorting

Generating labels

Other views

News from MIBIB_TE_X

Conclusion

An Original View

Basic Idea

Programming
Bibliographies

Jean-Michel
Hufflen

Contents

An Original View

Bibliographies

Sorting

Generating labels

Other views

News from
MIBIB_TE_X

Conclusion

An Original View

Basic Idea \Leftarrow a text processor runs on a *computer*. That is, it is a *program*, but this program's end-users are not directly related to its programming, to the successive statements of this program. Example: interactive word processors such as Microsoft Word. They are customisable, but by means of a graphical interface (most often by interactive menus).

What about L^AT_EX?

Commands allow end-users to customise its behaviour. These commands are produced by a *programming language*.

What about L^AT_EX?

Commands allow end-users to customise its behaviour. These commands are produced by a *programming language*.
Can a text processed by L^AT_EX take advantage of programming features?

Contents

An Original View

Bibliographies

Sorting

Generating labels

Other views

News from
MIB_BT_EX

Conclusion

What about L^AT_EX?

Commands allow end-users to customise its behaviour. These commands are produced by a *programming language*.

Can a text processed by L^AT_EX take advantage of programming features?

OK if these features are (quite) easily programmable with T_EX's language.

What about L^AT_EX?

Commands allow end-users to customise its behaviour. These commands are produced by a *programming language*.

Can a text processed by L^AT_EX take advantage of programming features?

OK if these features are (quite) easily programmable with T_EX's language.

Counter-example: sort an array with L^AT_EX before formatting it using L^AT_EX. (Try to program a sort procedure using T_EX's commands.)

Now

LuaT_EX!

Programming
Bibliographies

Jean-Michel
Hufflen

Contents

An Original View

Bibliographies

Sorting

Generating labels

Other views

News from
MiBibT_EX

Conclusion

LuaTeX!

Example: perform numeric calculations before formatting data.

Building 'References' sections

- (i) *Searching* bibliography databases.
- (ii) *Sorting* extracted resources.
- (iii) *Format* each reference.

Building 'References' sections

- (i) *Searching* bibliography databases.
- (ii) *Sorting* extracted resources.
- (iii) *Format* each reference.
- (i), (ii), (iii) were often done by BIBT_EX.

(i) *Searching* bibliography databases.

(ii) *Sorting* extracted resources.

(iii) *Format* each reference.

(i), (ii), (iii) were often done by BIB_TE_X.

Nowadays, the biblatex package is quite often used, so (iii) uses commands belonging to this package and 'actual' formatting is deferred to L^AT_EX's next pass. (i) & (ii) can be delegated to BIB_TE_X, but biber is preferred.

Still used. . . at least by some conference submission tools. . .

Still used. . . at least by some conference submission tools. . .
The language for bibliography styles is old-fashioned, more
suitable for small changes than programming a new style.
However, this language expresses *algorithms*.

Still used. . . at least by some conference submission tools. . .

The language for bibliography styles is old-fashioned, more suitable for small changes than programming a new style.

However, this language expresses *algorithms*.

BIB_TE_X cannot deal with *numeric* sorts!

Sorting

In $\text{BIB}_{\text{T}}\text{E}_{\text{X}}$: very limited, as mentioned above.

In `BIBTEX`: very limited, as mentioned above.

Using `biblatex`: many specific fields allows special orders to be expressed, e.g.:

`sortname` `sorttitle` `sortyear`

many schemes are predefined, specified by means of *mnemonics*, but:

as far as I know, month names are not considered,

[Contents](#)

[An Original View](#)

[Bibliographies](#)

[Sorting](#)

[Generating labels](#)

[Other views](#)

[News from
MIB_TE_X](#)

[Conclusion](#)

In `BIBTEX`: very limited, as mentioned above.

Using `biblatex`: many specific fields allows special orders to be expressed, e.g.:

`sortname` `sorttitle` `sortyear`

many schemes are predefined, specified by means of *mnemonics*, but:

as far as I know, month names are not considered, difficult cases are supposed to be addressed by the `\DeclareSortingTemplate` command, but practically, many complicated situations are solved by the last key `sortkey`.

Number of names

... for authors, editors, etc.

Number of names

... for authors, editors, etc.

In BIB_TE_X \Leftarrow limited since a string is generated as a sort key.

Number of names

... for authors, editors, etc.

In BIB_TE_X \Leftarrow limited since a string is generated as a sort key.

With biblatex \Leftarrow customisable, but not unbounded.

Reference keys

In alpha styles, suffixes are added in case of ambiguity. But how?

In alpha styles, suffixes are added in case of ambiguity. But how?

[Rob 1964a] Kenneth Robeson. *The Man of Bronze*. No. 1 in *Doc Savage Series*. Bantam Books, 1964.

[Rob 1964b] Kenneth Robeson. *The Thousand-Headed Man*. No. 2 in *Doc Savage Series*. Bantam Books, 1964.

[Contents](#)

[An Original View](#)

[Bibliographies](#)

[Sorting](#)

[Generating labels](#)

[Other views](#)

[News from
MiBibT_EX](#)

[Conclusion](#)

In alpha styles, suffixes are added in case of ambiguity. But how?

[Rob 1964a] Kenneth Robeson. *The Man of Bronze*. No. 1 in *Doc Savage Series*. Bantam Books, 1964.

[Rob 1964b] Kenneth Robeson. *The Thousand-Headed Man*. No. 2 in *Doc Savage Series*. Bantam Books, 1964.

or:

[Rob 1965] Kenneth Robeson. *The Polar Treasure*. No. 4 in *Doc Savage Series*. Bantam Books, 1965.

[Rob 1965a] Kenneth Robeson. *Brand of the Werewolf*. No. 5 in *Doc Savage Series*. Bantam Books, 1965.

[Contents](#)

[An Original View](#)

[Bibliographies](#)

[Sorting](#)

[Generating labels](#)

[Other views](#)

[News from
MIBibTeX](#)

[Conclusion](#)

More related to programming

Bibliographic module of ConT_EXt, a significant part is programmed in Lua.

More related to programming

Bibliographic module of Con \TeX t, a significant part is programmed in Lua.

MIBIB \TeX , programmed in Scheme, Version 1.4 deals with *encodings*.

Version 1.3 is working, but the repository has been closed.

Version 1.3 is working, but the repository has been closed. I personally escaped Covid-19, but an additional amount of work has slowed down the final part of Version 1.4 (different installation procedure).

What is (about to be) provided by MIBIB_TE_X

An *open* format for bibliographies, which can be reached by XML tools.

What is (about to be) provided by MIBIB_TE_X

Programming
Bibliographies

Jean-Michel
Hufflen

An *open* format for bibliographies, which can be reached by XML tools.

A better interface with Scheme, for some important procedures, e.g., sort.

Contents

An Original View

Bibliographies

Sorting

Generating labels

Other views

News from
MIBIB_TE_X

Conclusion

What is (about to be) provided by MIBIB_TE_X

An *open* format for bibliographies, which can be reached by XML tools.

A better interface with Scheme, for some important procedures, e.g., sort.

A compatibility for some additional fields of bibl_Tex, even if some standard styles are used.

What is (about to be) provided by MIBIB_TE_X

An *open* format for bibliographies, which can be reached by XML tools.

A better interface with Scheme, for some important procedures, e.g., sort.

A compatibility for some additional fields of biblatex, even if some standard styles are used.

A strong type checking for some important fields.

Executable files provided

`mlbibtex` analogous to `bibtex`;

`mlbiblatex` analogous to `biber`, in the sense that
generated files are suitable for the `biblatex` package.

I am a *programmer*

I prefer to deal with *open formats* and bibliographical references used within L^AT_EX should be such (possibly using a description language based on xml).

I am a *programmer*

I prefer to deal with *open formats* and bibliographical references used within \LaTeX should be such (possibly using a description language based on xml).

MIBIB_TE_X is a wild animal recognising only its master. . .

I am a *programmer*

I prefer to deal with *open formats* and bibliographical references used within L^AT_EX should be such (possibly using a description language based on xml).

MIBIB_TE_X is a wild animal recognising only its master. . .

Less true for the new version, it should recognise any master who will express orders in Scheme. . .

Thanks for your attention

... and be ready for MIBiB_TE_X's new version!