abusing_tex

Using TEX

To do what it was never really intended to do

With `custom-bib` as an example

Patrick W. Daly

Abusing TEX

To do what it was never really intended to do

With `custom-bib` as an example

Patrick W. Daly

Abusing TEX

Making TEX do what it was never really intended to do

With `custom-bib` as an example

Patrick W. Daly

TEX is a typesetting program

- The main intention of the TEX program is

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*
  - ➠ DEK, opening sentence of the *TEXbook*

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*
  - ➠ DEK, opening sentence of the *TEXbook*
- Notwithstanding, I have also heard that

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*
  - ➠ DEK, opening sentence of the *TEXbook*
- Notwithstanding, I have also heard that
  - *"TEX is programming language. . .*

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*
  - ➠ DEK, opening sentence of the *TEXbook*
- Notwithstanding, I have also heard that
  - *"TEX is programming language. . .*
  - *and the best programming language ever!"*

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*
  - ➠ DEK, opening sentence of the *TEXbook*
- Notwithstanding, I have also heard that
  - *"TEX is programming language. . .*
  - *and the best programming language ever!"*
  - ➠ email (sender unknown) to Helmut Kopka.

TEX is a typesetting program

- The main intention of the TEX program is
  - *"the creation of beautiful books. . .*
  - *especially those containing a lot of mathematics"*
  - ➠ DEK, opening sentence of the *TEXbook*
- Notwithstanding, I have also heard that
  - *"TEX is programming language. . .*
  - *and the best programming language ever!"*
  - ➠ email (sender unknown) to Helmut Kopka.
- So this is where the abuse begins.

And as a programming language?

- Yes, one can program with TEX, in that it can

And as a programming language?

- Yes, one can program with TEX, in that it can
  - open files, read text, do (integer) math...

And as a programming language?

- Yes, one can program with TEX, in that it can
  - open files, read text, do (integer) math...
  - manipulate text (or strings), write to files...

And as a programming language?

- Yes, one can program with TEX, in that it can
  - open files, read text, do (integer) math. . .
  - manipulate text (or strings), write to files. . .
  - and never write a single byte of DVI output.

And as a programming language?

- Yes, one can program with TEX, in that it can
  - open files, read text, do (integer) math. . .
  - manipulate text (or strings), write to files. . .
  - and never write a single byte of DVI output.
- Existing examples of such ‘programs’ are:

And as a programming language?

- Yes, one can program with TEX, in that it can
  - open files, read text, do (integer) math. . .
  - manipulate text (or strings), write to files. . .
  - and never write a single byte of DVI output.
- Existing examples of such ‘programs’ are:
  - `fontinst` by Alan Jeffrey and Rowland McDonnell, used to generate the necessary font metric and installation files for PostScript fonts;

And as a programming language?

- Yes, one can program with TEX, in that it can
  - open files, read text, do (integer) math. . .
  - manipulate text (or strings), write to files. . .
  - and never write a single byte of DVI output.
- Existing examples of such ‘programs’ are:
  - `fontinst` by Alan Jeffrey and Rowland McDonnell, used to generate the necessary font metric and installation files for PostScript fonts;
  - `docstrip` by Frank Mittelbach, Johannes Braams, Denys Duchier, Marcin Woliński, Mark Wooding for extracting code from documented sources.

Pros and cons

- ✔ The major advantage of programming with TEX is *portability*; no other program is needed.

Pros and cons

- ✔ The major advantage of programming with TEX is *portability*; no other program is needed.
- ✘ The major disadvantage is *slowness*; also difficulty in programming.

Pros and cons

- ✔ The major advantage of programming with TEX is *portability*; no other program is needed.
- ✘ The major disadvantage is *slowness*; also difficulty in programming.
- ➧ For programs applying directly to TEX and LATEX, such as font and package installations, portability outweighs speed.

And now `custom-bib`

- My `custom-bib` bundle (aka `makebst`, `merlin`) is a double abuse:

And now `custom-bib`

- My `custom-bib` bundle (aka `makebst`, `merlin`) is a double abuse:
  - it is an application of `docstrip` (this is `merlin`)

And now `custom-bib`

- My `custom-bib` bundle (aka `makebst`, `merlin`) is a double abuse:
  - it is an application of `docstrip` (this is `merlin`)
  - it has its own TEX program to manage the `docstrip` options (this is `makebst`)

And now `custom-bib`

- My `custom-bib` bundle (aka `makebst`, `merlin`) is a double abuse:
  - it is an application of `docstrip` (this is `merlin`)
  - it has its own TEX program to manage the `docstrip` options (this is `makebst`)
- The latter is the more interesting and complicated part.

The problem: BibTeX styles

- The formatting of bibliographies is done with BibTeX, using `.bst` files to define the format.

The problem: BibTeX styles

- The formatting of bibliographies is done with BibTeX, using `.bst` files to define the format.
- There is no uniform standard in English for such a format.

The problem: BibTeX styles

- The formatting of bibliographies is done with BibTeX, using `.bst` files to define the format.
- There is no uniform standard in English for such a format.
- Every publisher, journal has its own requirements, with fiddly little differences, each requiring a separate `.bst` file.

The problem: BibTeX styles

- The formatting of bibliographies is done with BibTeX, using `.bst` files to define the format.
- There is no uniform standard in English for such a format.
- Every publisher, journal has its own requirements, with fiddly little differences, each requiring a separate `.bst` file.
- Nelson Beebe has provided some 50 style files for various journals and applications, but this does not exhaust all the possibilities.

The problem: BibTeX styles

- The formatting of bibliographies is done with BibTeX, using `.bst` files to define the format.
- There is no uniform standard in English for such a format.
- Every publisher, journal has its own requirements, with fiddly little differences, each requiring a separate `.bst` file.
- Nelson Beebe has provided some 50 style files for various journals and applications, but this does not exhaust all the possibilities.
- What is the regular user to do when demands something that is not available? Creating or hacking the existing `.bst` files is a daunting experience.

Solution: Master Bibliography Style File

- I invented the `.mbs` files to be `.bst` files with alternative coding, selectable with `docstrip` options.

Solution: Master Bibliography Style File

- I invented the `.mbs` files to be `.bst` files with alternative coding, selectable with `docstrip` options.
- For example the `emphasize` function in the standard `plain.bst` contains the simple code:

Solution: Master Bibliography Style File

```
FUNCTION {emphasize}
{ duplicate$ empty$
    { pop$ "" }
    { "{\em " swap$ * "}" * }
  if$
}
```

Solution: Master Bibliography Style File

- I invented the `.mbs` files to be `.bst` files with alternative coding, selectable with `docstrip` options.
- For example the `emphasize` function in the standard `plain.bst` contains the simple code:
- while in `merlin.mbs`, the equivalent code is:

```
FUNCTION {emphasize}
%<*!em-x>
{ duplicate$ empty$
    { pop$ "" }
%<em-ul>    { "\uline{" swap$ * "}" * }
%<*!em-ul>
%<*!nfss>
%<!em-it>    { "{\em " swap$ * "\/}" * }
%<em-it>    { "{\it " swap$ * "\/}" * }
%</!nfss>
%<*nfss>
%<!em-it>    { "\emph{" swap$ * "}" * }
%<em-it>    { "\textit{" swap$ * "}" * }
%</nfss>
%</!em-ul>
    if$
}
%</!em-x>
%<em-x>{ skip$ }
```

Solution: Master Bibliography Style File

- I invented the `.mbs` files to be `.bst` files with alternative coding, selectable with `docstrip` options.
- For example the `emphasize` function in the standard `plain.bst` contains the simple code:
- while in `merlin.mbs`, the equivalent code is:
- By selecting none of the options `em-it`, `em-x`, `em-ul`, `nfss`, one obtains the original code by default.

Not really new

Alternative coding for `.bst` files is not a new idea.

Not really new

Alternative coding for `.bst` files is not a new idea.

- Oren Patashnik uses a file `btxbst.doc` with 6 options to generate the four standard BibTeX `.bst` files.

Not really new

Alternative coding for `.bst` files is not a new idea.

- Oren Patashnik uses a file `btxbst.doc` with 6 options to generate the four standard BibTeX `.bst` files.
- Charles Karney has a `physics.bst` with 30 options to generate `.bsts` for various journals

Not really new

Alternative coding for `.bst` files is not a new idea.

- Oren Patashnik uses a file `btxbst.doc` with 6 options to generate the four standard BibTeX `.bst` files.
- Charles Karney has a `physics.bst` with 30 options to generate `.bsts` for various journals
- Both of these are meant to be processed with the C preprocessor program (not TeX!) to produce the selected output.

Not really new

Alternative coding for `.bst` files is not a new idea.

- Oren Patashnik uses a file `btxbst.doc` with 6 options to generate the four standard BibTeX `.bst` files.
- Charles Karney has a `physics.bst` with 30 options to generate `.bsts` for various journals
- Both of these are meant to be processed with the C preprocessor program (not TeX!) to produce the selected output.
- I have converted both to `.mbs` for `docstrip` processing, simply as an exercise.

Additional features

`Merlin.mbs` can offer some more magical tricks:

Additional features

`Merlin.mbs` can offer some more magical tricks:

- Support for other languages: all explicit words are stored in functions, which can be redefined as needed.

Additional features

`Merlin.mbs` can offer some more magical tricks:

- Support for other languages: all explicit words are stored in fu

```
From merlin.mbs:
FUNCTION {bbl.editor}
%<!ed>{ "editor" }
%<ed>{ "ed." }
```

Additional features

`Merlin.mbs` can offer some more magical tricks:

- Support for other languages: all explicit words are stored in functions, which can be redefined as needed.
- The language redefinitions are contained in files like `finnish.mbs`

Additional features

`Merlin.mbs` can offer some more magical tricks:

- Support for other languages: all explicit words are stored in functions, which can be redefined as needed.
- T
  f

```
From finnish.mbs:
FUNCTION {bbl.editor}
%<!ed>{ "toimittaja" }
%<ed>{ "toim." }
```

Additional features

`Merlin.mbs` can offer some more magical tricks:

- Support for other languages: all explicit words are stored in functions, which can be redefined as needed.
- The language redefinitions are contained in files like `finnish.mbs`
- The `docstrip` job makes two passes through `merlin`, inserting the language file in between.

Additional features

`Merlin.mbs` can offer some more magical tricks:

- Support for other languages: all explicit words are stored in functions, which can be redefined as needed.
- The language redefinitions are contained in files like `finnish.mbs`
- The `docstrip` job makes two passes through `merlin`, inserting the language file in between.
- A choice of predefined abbreviations for journal names may also be inserted between the 2 passes.

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.
- Today there are over 200!

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.
- Today there are over 200!
- Even with 50, it was unthinkable to try to run `docstrip` in the interactive mode;

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.
- Today there are over 200!
- Even with 50, it was unthinkable to try to run `docstrip` in the interactive mode;
- and even preparing a batch file for `docstrip` manually would be prohibitive.

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.
- Today there are over 200!
- I
  t
- a
  w

```
\def\batchfile{demo.ins}
\input docstrip
\generate{\file{demo.sty}
      {\from{demo.dtx}{package}}
   \file{demo.drv}
      {\from{demo.dtx}{driver}}
   }
```

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.
- Today there are over 200!
- Even with 50, it was unthinkable to try to run `docstrip` in the interactive mode;
- and even preparing a batch file for `docstrip` manually would be prohibitive.
- Hence it was necessary to find some other way to let the user interface easily to this mammoth set of choices.

Managing the options

- When I first wrote the predecessor to `merlin.mbs`, there were about 50 options available.
- Today there are over 200!
- Even with 50, it was unthinkable to try to run `docstrip` in the interactive mode;
- and even preparing a batch file for `docstrip` manually would be prohibitive.
- Hence it was necessary to find some other way to let the user interface easily to this mammoth set of choices.
- *`Makebst` to the rescue!* And TEX is abused once more.

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive session to allow the user to generate a `docstrip` batch job by answering a seemly endless list of questions.

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive session to allow the user to generate a `docstrip` batch job by answering a seemly endless list of questions.

- First comes a welcoming message, an offer to print help, and the chance to enter the name of the `.mbs` file.

The makebst.tex (TEX) program

```
**********************************
This is Make Bibliography Style *
**********************************
It makes up a docstrip batch job to produce
a customized .bst file for running with BibTeX
Do you want a description of the usage?  (NO)

\yn=

Enter the name of the MASTER file
(default=merlin.mbs)

\mfile=
```

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive session to allow the user to generate a `docstrip` batch job by answering a seemly endless list of questions.

- First comes a welcoming message, an offer to print help, and the chance to enter the name of the `.mbs` file.
- Next, one enters the name of the output `.bst` file, and can enter a line of explanatory text.

The `makebst.tex` (TEX) program

Ru
ses
an

```
Name of the final OUTPUT .bst file?
(default extension=bst)

\ofile=mystyle

Give a comment line to include in the style
file.
Something like for which journals it is
applicable.

\ans=For journals of MY house
```

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive session to allow the user to generate a `docstrip` batch job by answering a seemly endless list of questions.

- First comes a welcoming message, an offer to print help, and the chance to enter the name of the `.mbs` file.
- Next, one enters the name of the output `.bst` file, and can enter a line of explanatory text.
- The user then selects a language file. If this is `merlin` itself, the language will be English.

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive sessi
answ

- F
  t
- N
  e
- T
  the language will be English.

```
EXTERNAL FILES:

Name of language definition file
(default=merlin.mbs)

\cfile=german

Name of language file:
\cfile=german.mbs.
```

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive session to allow the user to generate a `docstrip` batch job by answering a seemly endless list of questions.

- First comes a welcoming message, an offer to print help, and the chance to enter the name of the `.mbs` file.
- Next, one enters the name of the output `.bst` file, and can enter a line of explanatory text.
- The user then selects a language file. If this is `merlin` itself, the language will be English.
- Then one has the chance to add files with prestored journal abbreviations.

The `makebst.tex` (TEX) program

```
Include file(s) for extra journal names?  (NO)

\yn=y

File to include (default=physjour, geojour,
photjour.mbs)

\jfile=

Name of included files:  \jfile=physjour,
geojour, photjour,.mbs.
```

- Then one has the chance to add files with prestored journal abbreviations.

The `makebst.tex` (TEX) program

Running `makebst.tex` with TEX or LATEX produces an interactive session to allow the user to generate a `docstrip` batch job by answering a seemly endless list of questions.

- First comes a welcoming message, an offer to print help, and the chance to enter the name of the `.mbs` file.
- Next, one enters the name of the output `.bst` file, and can enter a line of explanatory text.
- The user then selects a language file. If this is `merlin` itself, the language will be English.
- Then one has the chance to add files with prestored journal abbreviations.

interrogation continues. . .

Now the real questioning begins.

interrogation continues. . .

Now the real questioning begins.

- It is to be numerical or author-year citations?

```
STYLE OF CITATIONS:
(*) Numerical as in standard LaTeX
(a) Author-year with some non-standard interface
(b) Alpha style, Jon90 or JWB90 for single or
multiple authors
(o) Alpha style, Jon90 even for multiple authors
(f) Alpha style, Jones90 (full name of first author)
(c) Cite key (special for listing contents of bib
file)
Select:

\ans=a
   You have selected:  Author-year
```

interrogation continues. . .

Now the real questioning begins.

- It is to be numerical or author-year citations?
- If author-year, which support system?

```
AUTHOR--YEAR SUPPORT SYSTEM (if author-year
citations)
(*) Natbib for use with natbib v5.3 or later
(o) Older Natbib without full authors citations
(l) Apalike for use with apalike.sty
(h) Harvard system with harvard.sty
(a) Astronomy system with astron.sty
(c) Chicago system with chicago.sty
(n) Named system with named.sty
(d) Author-date system with authordate1-4.sty
Select:

\ans=
   You have selected:  Natbib
```

interrogation continues. . .

Now the real questioning begins.

- It is to be numerical or author-year citations?
- If author-year, which support system?
- And onwards and onwards. Like order of authors,

```
ORDERING OF REFERENCES (if author-year citations)
(*) Alphabetical by all authors
(l) By label (Jones before Jones and James before
Jones et al)
(k) By label and cite key instead of label and title,
as above
(d) Year ordered and then by authors (for publication
lists)
(r) Reverse year ordered and then by authors (most
recent first)
(c) Citation order (unsorted, only meaningful for
numericals)
Select:

\ans=k
   You have selected:  By label and cite key
```

interrogation continues. . .

Now the real questioning begins.

- It is to be numerical or author-year citations?
- If author-year, which support system?
- And onwards and onwards. Like order of authors,
- and style of giving their names.

```
AUTHOR NAMES:
(*) Full, surname last (John Frederick Smith)
(f) Full, surname first (Smith, John Frederick)
(i) Initials + surname (J. F. Smith)
(r) Surname + initials (Smith, J. F.)
(s) Surname + dotless initials (Smith J F)
(x) Surname + pure initials (Smith JF)
(y) Surname + spaceless initials (Smith J.F.)
(a) Only first name reversed, initials (AGU style:
Smith, J. F., H. K. Jones)
(b) First name reversed, with full names (Smith, John
Fred, Harry Kab Jones)
Select:

\ans=r
   You have selected:  Surname + initials
```

interrogation continues. . .

Now the real questioning begins.

- It is to be numerical or author-year citations?
- If author-year, which support system?
- And onwards and onwards. Like order of authors,
- and style of giving their names.
- Finally after about 70 such questions, the end is reached.

interrogation continues. . .

Now the real questioning begins.

```
Finished!!
Batch job written to file ‘mystyle.dbj’
Shall I now run this batch job?  (NO)

\yn=n
```

- and style of giving their names.
- Finally after about 70 such questions, the end is reached.

interrogation continues. . .

Now the real questioning begins.

- It is to be numerical or author-year citations?
- If author-year, which support system?
- And onwards and onwards. Like order of authors,
- and style of giving their names.
- Finally after about 70 such questions, the end is reached.

The `docstrip` batch file

Shall we take a look inside the batch file?

The `docstrip` batch file

Shall we take a look inside the batch file?

- The batch file is also a protocol of the `makebst` session, and can be edited at will.

The `docstrip` batch file

Shall we take a look inside the batch file?

- The batch file is also a protocol of the `makebst` session, and can be edited at will.
- It starts by defining the pre- and post-ambles and setting up the files to be read and output. This is stored in `\MBopts`.

```
\input docstrip
\preamble
-------------------------------------
** For journals of MY house ***
\endpreamble
\postamble
End of customized bst file
\endpostamble

\def\MBopts{\from{merlin.mbs}{%
   head,\MBopta}
\from{german.mbs}{\MBopta}
\from{physjour.mbs}{\MBopta}
\from{geojour.mbs}{\MBopta}
\from{photjour.mbs}{\MBopta}
\from{merlin.mbs}{tail,\MBopta}}
```

The `docstrip` batch file

Shall we take a look inside the batch file?

- The batch file is also a protocol of the `makebst` session, and can be edited at will.
- It starts by defining the pre- and post-ambles and setting up the files to be read and output. This is stored in `\MBopts`.
- The macro `\MBopta` contains all the selected options. It is listed with all possible options present, the unused ones commented out.

```
\def\MBopta{%
exlang%:  External language file
%STYLE OF CITATIONS:
% %:  (def) Numerical
,ay%:  Author-year
% ,alph%:  Alpha style, Jon90 or JWB90
% ,alph,alf-1%:  Alpha style, Jon90
% ,alph,alf-f%:  Alpha style, Jones90
% ,cite%:  Cite key
%AUTHOR--YEAR SUPPORT SYSTEM
,nat%:  Natbib
% %:  (def) Older Natbib
% ,alk%:  Apalike
% ,har%:  Harvard
% ,ast%:  Astronomy
% ,cay%:  Chicago
% ,nmd%:  Named
% ,cn%:  Author-date
.  .  .  .
```

The docstrip batch file

Sh

```
.   .   .   .
%EMPHASIS: (affects all so-called italics)
%:  (def) Use emphasis
% ,em-it%:  Use true italics
% ,em-x%:  No italics
% ,em-ul%:  Underlining
%NEW FONT SELECTION SCHEME:
%:  (def) No NFSS
% ,nfss%:  NFSS
}
```

commented out.

The `docstrip` batch file

Shall we take a look inside the batch file?

- The batch file is also a protocol of the `makebst` session, and can be edited at will.
- It starts by defining the pre- and post-ambles and setting up the files to be read and output. This is stored in `\MBopts`.
- The macro `\MBopta` contains all the selected options. It is listed with all possible options present, the unused ones commented out.
- Finally, the generation command is given.

The `docstrip` batch file

Shall we take a look inside the batch file?

- The batch file is also a protocol of the `makebst` session, and

```
\generate{\file{mystyle.bst}{\MBopts}}
\endbatchfile
```

- The macro `\MBopta` contains all the selected options. It is listed with all possible options present, the unused ones commented out.
- Finally, the generation command is given.

The `docstrip` batch file

Shall we take a look inside the batch file?

- The batch file is also a protocol of the `makebst` session, and can be edited at will.
- It starts by defining the pre- and post-ambles and setting up the files to be read and output. This is stored in `\MBopts`.
- The macro `\MBopta` contains all the selected options. It is listed with all possible options present, the unused ones commented out.
- Finally, the generation command is given.

Merlin means

Merlin means ❋ *MAGIC* ❋

Merlin means ❋ *MAGIC* ❋

The `merlin.mbs` file does more than just act as input to `docstrip`.

Merlin means ❋ *MAGIC* ❋

The `merlin.mbs` file does more than just act as input to `docstrip`.

- It contains all the information for the interactive session, the questions, the choices, translations of the responses into actual options.

```
\beginoptiongroup{STYLE OF CITATIONS:}{}
\optdef{*}{}{Numerical}{as in standard LaTeX}
\optdef{a}{ay}{Author-year}{with some non-standard
interface}
\optdef{b}{alph}{Alpha style, Jon90 or JWB90}{for
single or multiple authors}
\optdef{o}{alph,alf-1}{Alpha style, Jon90}{even for
multiple authors}
\optdef{f}{alph,alf-f}{Alpha style, Jones90}{(full
name of first author)}
\optdef{c}{cite}{Cite key}{(special for listing
contents of bib file)}
\getans
\endoptiongroup
\if\ans a\numericalfalse\else\numericaltrue\fi
\if\ans b\mytempfalse \else\mytemptrue \fi
```

```
\beginoptiongroup{AUTHOR--YEAR SUPPORT SYSTEM
(if author-year citations)}
{\ifnumerical\else*\fi}
\optdef{*}{nat}{Natbib}{for use with natbib v5.3 or
later}
\optdef{o}{}{Older Natbib}{without full authors
citations}
\optdef{l}{alk}{Apalike}{for use with apalike.sty}
\optdef{h}{har}{Harvard}{system with harvard.sty}
\optdef{a}{ast}{Astronomy}{system with astron.sty}
\optdef{c}{cay}{Chicago}{system with chicago.sty}
\optdef{n}{nmd}{Named}{system with named.sty}
\optdef{d}{cn}{Author-date}{system with
authordate1-4.sty}
\getans
```

Merlin means ❋ *MAGIC* ❋

The `merlin.mbs` file does more than just act as input to `docstrip`.

- It contains all the information for the interactive session, the questions, the choices, translations of the responses into actual options.
  - I.e., `makebst.tex` serves as an interface to the information in `merlin.mbs`,

Merlin means ❋ *MAGIC* ❋

The `merlin.mbs` file does more than just act as input to `docstrip`.

- It contains all the information for the interactive session, the questions, the choices, translations of the responses into actual options.
  - I.e., `makebst.tex` serves as an interface to the information in `merlin.mbs`,
  - is not bound to any particular `.mbs` file,

Merlin means ❋ *MAGIC* ❋

The `merlin.mbs` file does more than just act as input to `docstrip`.

- It contains all the information for the interactive session, the questions, the choices, translations of the responses into actual options.
  - I.e., `makebst.tex` serves as an interface to the information in `merlin.mbs`,
  - is not bound to any particular `.mbs` file,
  - need not be changed when `merlin.mbs` is updated.

❋ *MAGIC* ❋...

- `Merlin` must react differently depending on how it is read:

❋ *MAGIC* ❋...

- `Merlin` must react differently depending on how it is read:
  - When processed directly by LaTeX, it acts like the documentation driver file.

❋ *MAGIC* ❋...

- `Merlin` must react differently depending on how it is read:
  - When processed directly by LaTeX, it acts like the documentation driver file.
  - When read by the documentation driver, it must deliver the documentation.

❋ *MAGIC* ❋...

- `Merlin` must react differently depending on how it is read:
  - When processed directly by LaTeX, it acts like the documentation driver file.
  - When read by the documentation driver, it must deliver the documentation.
  - When read by `makebst`, only the menu information must be visible.

❋ *MAGIC* ❋...

- `Merlin` must react differently depending on how it is read:
  - When processed directly by LaTeX, it acts like the documentation driver file.
  - When read by the documentation driver, it must deliver the documentation.
  - When read by `makebst`, only the menu information must be visible.
  - When processed by `docstrip`, only the alternative coding may be handled.

❋ *MAGIC* ❋...

- `Merlin` must react differently depending on how it is read:
  - When processed directly by LaTeX, it acts like the documentation driver file.
  - When read by the documentation driver, it must deliver the documentation.
  - When read by `makebst`, only the menu information must be visible.
  - When processed by `docstrip`, only the alternative coding may be handled.

☛ *How is this accomplished?*

I'm not revealing all my tricks today!

Support from American Physical Society

- Version 4 of `merlin.mbs` has been supported by the APS

Support from American Physical Society

- Version 4 of `merlin.mbs` has been supported by the APS
- with code added by David Carlisle and Arthur Ogawa.

Support from American Physical Society

- Version 4 of `merlin.mbs` has been supported by the APS
- with code added by David Carlisle and Arthur Ogawa.
- The additions are employed by the APS in its RevTeX collection, for their electronic journals.

Language support for `custom-bib`

The following (mostly contributed) language files exist:

Language support for `custom-bib`

The following (mostly contributed) language files exist:

```
catalan.mbs dansk.mbs dutch.mbs
english.mbs esperant.mbs finnish.mbs
french.mbs german.mbs italian.mbs
norsk.mbs polski.mbs portuges.mbs
slovene.mbs spanish.mbs
```

Language support for `custom-bib`

The following (mostly contributed) language files exist:

```
catalan.mbs dansk.mbs dutch.mbs
english.mbs esperant.mbs finnish.mbs
french.mbs german.mbs italian.mbs
norsk.mbs polski.mbs portuges.mbs
slovene.mbs spanish.mbs
```

Others are welcomed. (Note: `english.mbs` serves only as a template for others.)

Prestored journal abbreviations

These are stored in the files:

```
geojour.mbs photjour.mbs physjour.mbs
suppjour.mbs
```

Prestored journal abbreviations

These are stored in the files:

```
geojour.mbs photjour.mbs physjour.mbs
suppjour.mbs
```

They define abbreviations for journal names, in both full and short forms, selectable with one of the `docstrip` options.

Prestored journal abbreviations

```
%<*jabr>
.   .   .   .
MACRO {jgr} {"J. Geophys.  Res."}
.   .   .   .
%</jabr>
%<*!jabr>
.   .   .   .
MACRO {jgr} {"Journal of Geophysical Research"}
.   .   .   .
%</!jabr>
```

Prestored journal abbreviations

These are stored in the files:

```
geojour.mbs photjour.mbs physjour.mbs
suppjour.mbs
```

They define abbreviations for journal names, in both full and short forms, selectable with one of the `docstrip` options.

To get a list of the journals and their abbreviations...

...process (with TEX) the supplied file `shorthnd.ins`.

...process (with TEX) the supplied file `shorthnd.ins`.

- This generates `shorthnd.tex` from the current `.mbs` files, using `docstrip` once more,

...process (with TEX) the supplied file `shorthnd.ins`.

- This generates `shorthnd.tex` from the current `.mbs` files, using `docstrip` once more,
- which when processed with LATEX, gives a listing of the abbreviations with full translations.

`pre`	Physical Review E
`ps`	Physica Scripta
`procrsl`	Proceedings of the Royal Society of London
`rmp`	Reviews of Modern Physics
`rsi`	Review of Scientific Instruments
`science`	Science
`sciam`	Scientific American
`sam`	Studies in Applied Mathematics
`sjpp`	Soviet Journal of Plasma Physics
`spd`	Soviet Physics–Doklady
`sptp`	Soviet Physics–Technical Physics
`spu`	Soviet Physics–Uspeki
`st`	Sky and Telescope

File: `geojour.mbs`, version: 2002/07/10 2.0h (PWD)

`aisr`	Advances in Space Research
`ag`	Annales Geophysicæ
`anigeo`	Annali di Geofisica
`angl`	Annals of Glaciology
`andmet`	Annalen der Meteorologie
`andgeo`	Annales de Geophysique
`andphy`	Annales de Physique
`afmgb`	Archiv für Meteorologie, Geophysik

s files,

...process (with TEX) the supplied file `shorthnd.ins`.

- This generates `shorthnd.tex` from the current `.mbs` files, using `docstrip` once more,
- which when processed with LATEX, gives a listing of the abbreviations with full translations.
- That's the final bit of ❋ *magic*.

Conclusions

Yes, TEX has been abused by all this.

Conclusions

Yes, TEX has been abused by all this.

But it has been for a good cause.

Conclusions

Yes, TEX has been abused by all this.

But it has been for a good cause.

And TEX can take it!

Conclusions

Yes, TEX has been abused by all this.

But it has been for a good cause.

And TEX can take it!