bib2gls: symbols

Nicola L.C. Talbot
dickimaw-books.com/contact

Dickimaw Books

The 42nd Annual Conference of the TeX Users Group
August 5–8, 2021

Overview

- bib2gls is a command line tool incorporated into the document build.
- Designed to work with glossaries-extra and the record package option:

```
\usepackage[record]{glossaries-extra}
```

- Entries are defined in one or more BIB files; bib2gls:
  - selects entries according to records found in the AUX file (similar to BibTeX);
  - hierarchically sorts entries and collates locations lists (similar to makeindex).

Overview

- `bib2gls` is a command line tool incorporated into the document build.
- Designed to work with `glossaries-extra` and the `record` package option:

```
\usepackage[record]{glossaries-extra}
```

- Entries are defined in one or more BIB files; `bib2gls`:
  - selects entries according to records found in the AUX file (similar to BibTeX);
  - hierarchically sorts entries and collates locations lists (similar to MakeIndex).
- Development discussed in *TUGboat* 40:1.

Overview

- `bib2gls` is a command line tool incorporated into the document build.
- Designed to work with `glossaries-extra` and the `record` package option:

```
\usepackage[record]{glossaries-extra}
```

- Entries are defined in one or more BIB files; `bib2gls`:
  - selects entries according to records found in the AUX file (similar to BibTeX);
  - hierarchically sorts entries and collates locations lists (similar to MakeIndex).
- Development discussed in *TUGboat* 40:1.
- Selection, cross-references and locations discussed in *TUGboat* 41:3.

MakeIndex

```
\documentclass{article}
\usepackage[style=treegroup]{glossaries}
\makeglossaries
\loadglsentries{constants}
\begin{document}
\gls{pi}, \gls{e}, \gls{gelfondcons} and
\gls{root2}.
\printglossary[nonumberlist]
\end{document}
```

MakeIndex

```
\documentclass{article}
\usepackage[style=treegroup]{glossaries}
\makeglossaries
\loadglsentries{constants}
\begin{document}
\gls{pi}, \gls{e}, \gls{gelfondcons} and
\gls{root2}.
\printglossary[nonumberlist]
\end{document}
```

MakeIndex

```
\documentclass{article}
\usepackage[style=treegroup]{glossaries}
\makeglossaries
\loadglsentries{constants}
\begin{document}
\gls{pi}, \gls{e}, \gls{gelfondcons} and
\gls{root2}.
\printglossary[nonumberlist]
\end{document}
```

constants.tex

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

\newglossaryentry{e}{name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

\newglossaryentry{root2}{
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

\newglossaryentry{gelfondcons}{
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.tex

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

\newglossaryentry{e}{name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

\newglossaryentry{root2}{
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

\newglossaryentry{gelfondcons}{
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.tex

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

\newglossaryentry{e}{name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

\newglossaryentry{root2}{
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

\newglossaryentry{gelfondcons}{
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.tex

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

\newglossaryentry{e}{name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

\newglossaryentry{root2}{
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

\newglossaryentry{gelfondcons}{
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.tex

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

\newglossaryentry{e}{name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

\newglossaryentry{root2}{
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

\newglossaryentry{gelfondcons}{
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.tex

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

\newglossaryentry{e}{name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

\newglossaryentry{root2}{
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

\newglossaryentry{gelfondcons}{
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

Document Build

- Using `makeglossaries` Perl script:

```
pdflatex myDoc.tex
makeglossaries myDoc
pdflatex myDoc.tex
```

Document Build

- Using `makeglossaries` Perl script:

```
pdflatex myDoc.tex
makeglossaries myDoc
pdflatex myDoc.tex
```

- Using `makeglossaries-lite` Lua script:

```
pdflatex myDoc.tex
makeglossaries-lite myDoc
pdflatex myDoc.tex
```

Document Build

- Using `makeglossaries` Perl script:

```
pdflatex myDoc.tex
makeglossaries myDoc
pdflatex myDoc.tex
```

- Using `makeglossaries-lite` Lua script:

```
pdflatex myDoc.tex
makeglossaries-lite myDoc
pdflatex myDoc.tex
```

- Or use MakeIndex explicitly:

```
pdflatex myDoc.tex
makeindex -s myDoc.ist -o myDoc.gls myDoc.glo
pdflatex myDoc.tex
```

MakeIndex Sorting

- No `sort` key was used so `name` provides sort value.

MakeIndex Sorting

- No `sort` key was used so `name` provides sort value.
- MakeIndex doesn't recognise LaTeX commands.

MakeIndex Sorting

- No `sort` key was used so `name` provides sort value.
- MakeIndex doesn't recognise LaTeX commands.
- All sort values are treated as literal strings.

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

- The first twelve characters are the same.

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

- The first twelve characters are the same.
- Order is determined by the 13th character onwards.

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { **e** \ s p \ p i }
4. \ e n s u r e m a t h { **e** }

- The first twelve characters are the same.
- Order is determined by the 13th character onwards.
- \ comes before e.

MakeIndex Order

1. \ e n s u r e m a t h { \ **p** i }
2. \ e n s u r e m a t h { \ **s** u r d 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

- The first twelve characters are the same.
- Order is determined by the 13th character onwards.
- `\` comes before `e`.
- `p` comes before `s`.

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

- The first twelve characters are the same.
- Order is determined by the 13th character onwards.
- \ comes before e.
- p comes before s.
- \ comes before }.

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

- The first twelve characters are the same.
- Order is determined by the 13th character onwards.
- \ comes before e.
- p comes before s.
- \ comes before }.
- Order: π $\sqrt{2}$ e^{π} e.

MakeIndex Order

1. \ e n s u r e m a t h { \ p i }
2. \ e n s u r e m a t h { \ s u r d _ 2 }
3. \ e n s u r e m a t h { e \ s p \ p i }
4. \ e n s u r e m a t h { e }

- The first twelve characters are the same.
- Order is determined by the 13th character onwards.
- \ comes before e.
- p comes before s.
- \ comes before }.
- Order: π $\sqrt{2}$ e^{π} e.
- The first character of all the sort values is a symbol so they are all placed in the ``symbols'' letter group.

π, e, e^{π} and $\sqrt{2}$.

Glossary

Symbols

π (3.14159) ratio of circumference of a circle to its diameter.
$\sqrt{2}$ (1.41421) Pythagoras' constant.
e^{π} (23.1406926) Gelfond's constant.
e (2.71828) Euler's number.

Xindy

- Requires `xindy` package option:

```
\usepackage[style=treegroup,xindy]{glossaries}
```

Xindy

- Requires `xindy` package option:

```
\usepackage[style=treegroup,xindy]{glossaries}
```

- Document build:

```
pdflatex myDoc.tex
makeglossaries myDoc
pdflatex myDoc.tex
```

Xindy

- Requires `xindy` package option:

```
\usepackage[style=treegroup,xindy]{glossaries}
```

- Document build:

```
pdflatex myDoc.tex
makeglossaries myDoc
pdflatex myDoc.tex
```

- Or use Xindy explicitly:

```
pdflatex myDoc.tex
xindy -L english -C utf8 -I xindy -M myDoc -o myDoc.gls
pdflatex myDoc.tex
```

Xindy Order

1. `\ensuremath{\pi}`
2. `\ensuremath{\surd 2}`
3. `\ensuremath{e}`
4. `\ensuremath{e\sp\pi}`

Xindy Order

1. `\ensuremath{\pi}` ➜ empty
2. `\ensuremath{\surd 2}` ➜ `2`
3. `\ensuremath{e}` ➜ `e`
4. `\ensuremath{e\sp\pi}` ➜ `e`

- **Commands and braces are stripped.**

Xindy Order

1. `\ensuremath{\pi}` ➜ empty ✗
2. `\ensuremath{\surd 2}` ➜ `2`
3. `\ensuremath{e}` ➜ `e`
4. `\ensuremath{e\sp\pi}` ➜ `e`

- Commands and braces are stripped.
- **Empty values are forbidden.**

Xindy Order

1. `\ensuremath{\pi}` ➜ empty ✘
2. `\ensuremath{\surd 2}` ➜ `2`
3. `\ensuremath{e}` ➜ `e`
4. `\ensuremath{e\sp\pi}` ➜ `e` Duplicate!

- Commands and braces are stripped.
- Empty values are forbidden.
- **Duplicates are merged.**

Xindy Order

1. Xindy fails
2. `\ensuremath{\surd 2}` ➜ 2
3. `\ensuremath{e}` ➜ e
4. Duplicate lost

- Commands and braces are stripped.
- Empty values are forbidden.
- Duplicates are merged.
- This example doesn't work with Xindy.

The sort Key

- The sort key should be used with MakeIndex and Xindy when the name value contains commands.

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159},sort={pi}}
```

The sort Key

- The sort key should be used with MakeIndex and Xindy when the name value contains commands.

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159},sort={pi}}
```

- The sort field should *NOT* be used with bib2gls.

The sort Key

- The sort key should be used with MakeIndex and Xindy when the name value contains commands.

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159},sort={pi}}
```

- The sort field should *NOT* be used with bib2gls.
  1. Use entry types and fallbacks, or

The sort Key

- The sort key should be used with MakeIndex and Xindy when the name value contains commands.

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159},sort={pi}}
```

- The sort field should *NOT* be used with bib2gls.
  1. Use entry types and fallbacks, or
  2. use field aliasing or the sort-field resource option.

The sort Key

- The sort key should be used with MakeIndex and Xindy when the name value contains commands.

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159},sort={pi}}
```

- The sort field should *NOT* be used with bib2gls.
  1. Use entry types and fallbacks, or
  2. use field aliasing or the sort-field resource option.
- The TEX parser library incorporated into bib2gls can be used to interpret common commands.

The sort Key

- The sort key should be used with MakeIndex and Xindy when the name value contains commands.

```
\newglossaryentry{pi}{name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159},sort={pi}}
```

- The sort field should *NOT* be used with bib2gls.
  1. Use entry types and fallbacks, or
  2. use field aliasing or the sort-field resource option.
- The TEX parser library incorporated into bib2gls can be used to interpret common commands.
- Provides a flexible system that allows a BIB file to be shared across multiple documents that have different sorting requirements.

Document (bib2gls)

```
\documentclass{article}
\usepackage[stylemods,style=treegroup,record]
  {glossaries-extra}
\GlsXtrLoadResources[
 src=constants,% data in constants.bib
 save-locations=false% no locations required
]
\begin{document}
\gls{pi}, \gls{e}, \gls{gelfondcons} and
\gls{root2}.
\printunsrtglossary
\end{document}
```

Document (bib2gls)

```
\documentclass{article}
\usepackage[stylemods,style=treegroup,record]
  {glossaries-extra}
\GlsXtrLoadResources[
 src=constants,% data in constants.bib
 save-locations=false% no locations required
]
\begin{document}
\gls{pi}, \gls{e}, \gls{gelfondcons} and
\gls{root2}.
\printunsrtglossary
\end{document}
```

constants.bib

\newglossaryentry ➜ @entry

```
% Encoding: UTF-8
@entry{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@entry{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@entry{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@entry{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.bib

\newglossaryentry ➜ @entry

```
% Encoding: UTF-8
@entry{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@entry{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@entry{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@entry{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.bib

\newglossaryentry ➜ @entry

```
% Encoding: UTF-8
@entry{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@entry{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@entry{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@entry{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.bib

\newglossaryentry ➜ @entry

```
% Encoding: UTF-8
@entry{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@entry{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@entry{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@entry{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.bib

\newglossaryentry ➜ @entry

```
% Encoding: UTF-8
@entry{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@entry{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@entry{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@entry{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

constants.bib

\newglossaryentry ➜ @entry

```
% Encoding: UTF-8
@entry{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@entry{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@entry{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@entry{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

Document Build

- Default:

```
pdflatex myDoc.tex
bib2gls myDoc
pdflatex myDoc.tex
```

- Verbose:

```
pdflatex myDoc.tex
bib2gls --verbose myDoc
pdflatex myDoc.tex
```

- Letter groups required:

```
pdflatex myDoc.tex
bib2gls --group myDoc
pdflatex myDoc.tex
```

Fallbacks

- Fallback fields are used when a value is required from a field that **hasn't been set**.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.
- The designated fallback field varies according to entry type.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.
- The designated fallback field varies according to entry type.
- For example, `bib2gls` tries to look up the value of the `sort` field. *If it isn't set,* it will then try the `sort` fallback field.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.
- The designated fallback field varies according to entry type.
- For example, `bib2gls` tries to look up the value of the `sort` field. *If it isn't set,* it will then try the `sort` fallback field.

  `@entry`, `@index` the `sort` fallback is the `name` field.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.
- The designated fallback field varies according to entry type.
- For example, `bib2gls` tries to look up the value of the `sort` field. *If it isn't set,* it will then try the `sort` fallback field.

  `@entry`, `@index` the `sort` fallback is the `name` field.
  `@abbreviation`, `@acronym` the `sort` fallback is the `short` field.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.
- The designated fallback field varies according to entry type.
- For example, `bib2gls` tries to look up the value of the `sort` field. *If it isn't set,* it will then try the `sort` fallback field.

  `@entry`, `@index` the `sort` fallback is the `name` field.
  `@abbreviation`, `@acronym` the `sort` fallback is the `short` field.
  `@symbol`, `@number` the `sort` fallback is the *label*.

Fallbacks

- Fallback fields are used when a value is required from a field that hasn't been set.
- The `bib2gls` fallback field isn't necessarily the same as the `\newglossaryentry` or `\newabbreviation` default value.
- The designated fallback field varies according to entry type.
- For example, `bib2gls` tries to look up the value of the `sort` field. *If it isn't set,* it will then try the `sort` fallback field.

  `@entry`, `@index` the `sort` fallback is the `name` field.
  `@abbreviation`, `@acronym` the `sort` fallback is the `short` field.
  `@symbol`, `@number` the `sort` fallback is the *label*.
- If the `sort` field is set then there's no need to look up the fallback field.

constants.bib `@entry` sort fallbacks

- Sort value obtained from `name` field:

```
          pi \ensuremath{\pi}
           e \ensuremath{e}
       root2 \ensuremath{\surd 2}
gelfondcons \ensuremath{e\sp\pi}
```

- `bib2gls` will use the TEX parser library if the sort value contains `\ ~ $ {` or `}`
- Unrecognised commands are ignored.

constants.bib `@entry` sort fallbacks

- Sort value obtained from `name` field:

```
          pi \ensuremath{\pi} ➜ π
           e \ensuremath{e} ➜ e
       root2 \ensuremath{\surd 2} ➜ √2
gelfondcons \ensuremath{e\sp\pi}
```

- `bib2gls` will use the TEX parser library if the sort value contains `\ ~ $ {` or `}`
- Unrecognised commands are ignored.
- Symbol commands are converted to the closest Unicode equivalent.

constants.bib `@entry` sort fallbacks

- Sort value obtained from `name` field:

```
          pi \ensuremath{\pi} ➜ π
           e \ensuremath{e} ➜ e
       root2 \ensuremath{\surd 2} ➜ √2
  gelfondcons \ensuremath{e\sp\pi} ➜ eπ
```

- `bib2gls` will use the TEX parser library if the sort value contains `\ ~ $ {` or `}`
- Unrecognised commands are ignored.
- Symbol commands are converted to the closest Unicode equivalent.
- Fonts and similar markup are discarded.

constants.bib `@entry` sort fallbacks

- Sort value obtained from `name` field:

```
          pi \ensuremath{\pi} ➜ π
           e \ensuremath{e} ➜ e
       root2 \ensuremath{\surd 2} ➜ √2
gelfondcons \ensuremath{e\sp\pi} ➜ eπ
```

- `bib2gls` will use the TEX parser library if the sort value contains `\ ~ $ {` or `}`
- Unrecognised commands are ignored.
- Symbol commands are converted to the closest Unicode equivalent.
- Fonts and similar markup are discarded.
- Sort method determines ordering. Non-locale letter and letter-number methods are generally best for symbols.

```
\GlsXtrLoadResources[sort=letter-nocase, ...]
```

constants.bib `@entry` sort fallbacks

- Sort value obtained from `name` field:

```
          pi \ensuremath{\pi} ➜ π (0x1D70B)
           e \ensuremath{e} ➜ e (0x65)
       root2 \ensuremath{\surd 2} ➜ √2 (0x221A 0x32)
gelfondcons \ensuremath{e\sp\pi} ➜ eπ (0x65 0x1D70B)
```

- `bib2gls` will use the TEX parser library if the sort value contains `\ ~ $ {` or `}`
- Unrecognised commands are ignored.
- Symbol commands are converted to the closest Unicode equivalent.
- Fonts and similar markup are discarded.
- Sort method determines ordering. Non-locale letter and letter-number methods are generally best for symbols.

```
\GlsXtrLoadResources[sort=letter-nocase, ...]
```

- Order by Unicode values: e e^{π} $\sqrt{2}$ π.

π, e, e^{π} and $\sqrt{2}$.

Glossary

e (2.71828) Euler's number
e^{π} (23.1406926) Gelfond's constant
$\sqrt{2}$ (1.41421) Pythagoras' constant
π (3.14159) ratio of circumference of a circle to its diameter

π, e, e^{π} and $\sqrt{2}$.

Glossary

e (2.71828) Euler's number
e^{π} (23.1406926) Gelfond's constant
$\sqrt{2}$ (1.41421) Pythagoras' constant
π (3.14159) ratio of circumference of a circle to its diameter

- `treegroup` style used but no letter group heading.

π, e, e^{π} and $\sqrt{2}$.

Glossary

e (2.71828) Euler's number
e^{π} (23.1406926) Gelfond's constant
$\sqrt{2}$ (1.41421) Pythagoras' constant
π (3.14159) ratio of circumference of a circle to its diameter

- `treegroup` style used but no letter group heading.
- `bib2gls` was invoked with default `--no-group` switch.

bib2gls --group

```
pdflatex myDoc.tex
bib2gls --group myDoc
pdflatex myDoc.tex
```

bib2gls --group

```
pdflatex myDoc.tex
bib2gls --group myDoc
pdflatex myDoc.tex
```

- Fails with error:

```
! Package inputenc Error: Unicode character π (U+1D70B)
(inputenc)                not set up for use with LaTeX.
```

bib2gls --group

```
pdflatex myDoc.tex
bib2gls --group myDoc
pdflatex myDoc.tex
```

- Fails with error:

  ```
  ! Package inputenc Error: Unicode character π (U+1D70B)
  (inputenc)                not set up for use with LaTeX.
  ```

- bib2gls recognises π as a letter and uses π as the group title.

bib2gls --group

Switch to XeLaTeX or LuaLaTeX. Add `fontspec`:

```
\usepackage{fontspec}
\setmainfont{DejaVu Math TeX Gyre}
```

Document build:

```
xelatex myDoc.tex
bib2gls --group myDoc
xelatex myDoc.tex
```

π, e, e^{π} and $\sqrt{2}$.

Glossary

E

e (2.71828) Euler's number
e^{π} (23.1406926) Gelfond's constant

Symbols

$\sqrt{2}$ (1.41421) Pythagoras' constant

π

π (3.14159) ratio of circumference of a circle to its diameter

bib2gls --group

- Force all entries into a specific group

```
\GlsXtrLoadResources[
 group=glssymbols,
 sort=letter-nocase,
 src=constants,% data in constants.bib
 save-locations=false% no locations required
]
```

bib2gls --group

- Force all entries into a specific group

```
\GlsXtrLoadResources[
 group=glssymbols,
 sort=letter-nocase,
 src=constants,% data in constants.bib
 save-locations=false% no locations required
]
```

- Value of `group` option is a *label*

bib2gls --group

- Force all entries into a specific group

```
\GlsXtrLoadResources[
 group=glssymbols,
 sort=letter-nocase,
 src=constants,% data in constants.bib
 save-locations=false% no locations required
]
```

- Value of `group` option is a *label*
- Switch back to pdfLaTeX

π, e, e^{π} and $\sqrt{2}$.

Glossary

Symbols

e (2.71828) Euler's number
e^{π} (23.1406926) Gelfond's constant
$\sqrt{2}$ (1.41421) Pythagoras' constant
π (3.14159) ratio of circumference of a circle to its diameter

constants.bib

@entry ➜ @number

```
% Encoding: UTF-8
@number{pi,name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}

@number{e,name={\ensuremath{e}},
description={Euler's number},symbol={2.71828}}

@number{root2,
name={\ensuremath{\surd 2}},symbol={1.41421},
description={Pythagoras' constant}}

@number{gelfondcons,
name={\ensuremath{e\sp\pi}},symbol={23.1406926},
description={Gelfond's constant}}
```

@number and @symbol

- @number and @symbol entries fallback on the *label* if the sort field is missing.
  1. pi
  2. e
  3. root2
  4. gelfondscons

`@number` and `@symbol`

- `@number` and `@symbol` entries fallback on the *label* if the `sort` field is missing.
  1. `pi`
  2. `e`
  3. `root2`
  4. `gelfondscons`
- Labels shouldn't contain UTF-8 characters with pdfLaTeX, but may with XeLaTeX or LuaLaTeX.

@number and @symbol

- @number and @symbol entries fallback on the *label* if the sort field is missing.
  1. pi
  2. e
  3. root2
  4. gelfondscons
- Labels shouldn't contain UTF-8 characters with pdfLaTeX, but may with XeLaTeX or LuaLaTeX.
- Choose locale alphabetical or non-locale letter sort methods as appropriate. For example:

```
\GlsXtrLoadResources[
 sort=en,% English alphabet
 src=constants,% data in constants.bib
 save-locations=false% no locations required
]
```

π, e, e^{π} and $\sqrt{2}$.

Glossary

E

e (2.71828) Euler's number

G

e^{π} (23.1406926) Gelfond's constant

P

π (3.14159) ratio of circumference of a circle to its diameter

R

$\sqrt{2}$ (1.41421) Pythagoras' constant

Change Sort Value Field

- Select a different field to obtain the sort value. For example:

```
\GlsXtrLoadResources[sort-field=symbol, ...]
```

Change Sort Value Field

- Select a different field to obtain the sort value. For example:

```
\GlsXtrLoadResources[sort-field=symbol, ...]
```

- If the given field isn't set, the fallback for *that* field (not the `sort` field) is used.

Change Sort Value Field

- Select a different field to obtain the sort value. For example:

```
\GlsXtrLoadResources[sort-field=symbol, ...]
```

- If the given field isn't set, the fallback for *that* field (not the `sort` field) is used.
- No fallback for `symbol` field!

Change Sort Value Field

- Select a different field to obtain the sort value. For example:

```
\GlsXtrLoadResources[sort-field=symbol, ...]
```

- If the given field isn't set, the fallback for *that* field (not the `sort` field) is used.
- No fallback for `symbol` field!
- To order numerically use a number sort. For example, for double-precision (decimal) numbers:

```
\GlsXtrLoadResources[sort=double, ...]
```

Change Sort Value Field

- Select a different field to obtain the sort value. For example:

```
\GlsXtrLoadResources[sort-field=symbol, ...]
```

- If the given field isn't set, the fallback for *that* field (not the `sort` field) is used.
- No fallback for `symbol` field!
- To order numerically use a number sort. For example, for double-precision (decimal) numbers:

```
\GlsXtrLoadResources[sort=double, ...]
```

- Numeric sort methods always assign entries to the ``Numbers'' (`glsnumbers`) group (if `--group` switch is used).

π, e, e^{π} and $\sqrt{2}$.

Glossary

Numbers

$\sqrt{2}$ (1.41421) Pythagoras' constant
e (2.71828) Euler's number
π (3.14159) ratio of circumference of a circle to its diameter
e^{π} (23.1406926) Gelfond's constant

π, e, e^{π} and $\sqrt{2}$.

Glossary

Numbers

$\sqrt{2}$ (1.41421) Pythagoras' constant
e (2.71828) Euler's number
π (3.14159) ratio of circumference of a circle to its diameter
e^{π} (23.1406926) Gelfond's constant

- Group title can be changed.

π, e, e^{π} and $\sqrt{2}$.

Glossary

Numbers

$\sqrt{2}$ (1.41421) Pythagoras' constant
e (2.71828) Euler's number
π (3.14159) ratio of circumference of a circle to its diameter
e^{π} (23.1406926) Gelfond's constant

- Group title can be changed.
- Either redefine \⟨*label*⟩groupname:

```
\renewcommand{\glsnumbersgroupname}{Constants}
```

π, e, e^{π} and $\sqrt{2}$.

Glossary

Numbers

$\sqrt{2}$ (1.41421) Pythagoras' constant
e (2.71828) Euler's number
π (3.14159) ratio of circumference of a circle to its diameter
e^{π} (23.1406926) Gelfond's constant

- Group title can be changed.
- Either redefine \⟨*label*⟩groupname:

```
\renewcommand{\glsnumbersgroupname}{Constants}
```

- Or use \glsxtrsetgrouptitle:

```
\glsxtrsetgrouptitle{glsnumbers}{Constants}
```

pictographs.bib

```
% Encoding: UTF-8
@symbol{heartsuit,name={\ensuremath{\heartsuit}},
 description={heart}}
@symbol{spadesuit,name={\ensuremath{\spadesuit}},
 description={spade}}
@symbol{diamondsuit,name={\ensuremath{\diamondsuit}},
 description={diamond}}
@symbol{clubsuit,name={\ensuremath{\clubsuit}},
 description={club}}
@symbol{email,name={\Email},description={email}}
@symbol{envelope,name={\Letter},description={letter}}
@symbol{phone,name={\Mobilefone},
description={mobile phone}}
@symbol{landline,name={\Telefon},description={telephone}}
```

Multiple BIB Files

- Multiple BIB files can be specified. For example:

```
\GlsXtrLoadResources[src={constants,pictographs}, ...]
```

Multiple BIB Files

- Multiple BIB files can be specified. For example:

```
\GlsXtrLoadResources[src={constants,pictographs}, ...]
```

- The same sort method is applied to all entries in the same resource set.

Document (constants.bib and pictographs.bib)

```
\documentclass{article}
\usepackage{marvosym}
\usepackage[stylemods,style=treegroup,record]
{glossaries-extra}
\GlsXtrLoadResources[
sort=letter-nocase,
src={constants,pictographs},% bib files
selection=all% select all entries
]
\begin{document}
\printunsrtglossary
\end{document}
```

@symbol and @number

- The sort field hasn't been set.

@symbol and @number

- The sort field hasn't been set.
- The fallback for the missing sort field is obtained from the label for @symbol and @number.

@symbol and @number

- The sort field hasn't been set.
- The fallback for the missing sort field is obtained from the label for @symbol and @number.
- Sort order: clubsuit, diamondsuit, e, email, envelope, gelfondcons, heartsuit, landline, phone, pi, root2, spadesuit.

Glossary

♣ club
♢ diamond
e (2.71828) Euler's number
✉ email
✉ letter
e^{π} (23.1406926) Gelfond's constant
♡ heart
☎ telephone
mobile phone
π (3.14159) ratio of circumference of a circle to its diameter
$\sqrt{2}$ (1.41421) Pythagoras' constant
♠ spade

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.
- For `@symbol` and `@number` use `symbol-sort-fallback`. For example:

```
\GlsXtrLoadResources[symbol-sort-fallback=name, ...]
```

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.
- For `@symbol` and `@number` use `symbol-sort-fallback`. For example:

```
\GlsXtrLoadResources[symbol-sort-fallback=name, ...]
```

- TEX parser library is used.

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.
- For `@symbol` and `@number` use `symbol-sort-fallback`. For example:

```
\GlsXtrLoadResources[symbol-sort-fallback=name, ...]
```

- TEX parser library is used.
- Current version doesn't recognise `marvosym` commands.

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.
- For `@symbol` and `@number` use `symbol-sort-fallback`. For example:

```
\GlsXtrLoadResources[symbol-sort-fallback=name, ...]
```

- TEX parser library is used.
- Current version doesn't recognise `marvosym` commands.
- Unknown commands are ignored.

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.
- For `@symbol` and `@number` use `symbol-sort-fallback`. For example:

```
\GlsXtrLoadResources[symbol-sort-fallback=name, ...]
```

- TEX parser library is used.
- Current version doesn't recognise `marvosym` commands.
- Unknown commands are ignored.
- `bib2gls` doesn't mind empty sort values. (Fallbacks are only used if the required field *isn't set* not if the value ends up empty after pre-processing.)

Changing the Sort Fallback

- The default fallback for the `sort` field can be changed.
- For `@symbol` and `@number` use `symbol-sort-fallback`. For example:

```
\GlsXtrLoadResources[symbol-sort-fallback=name, ...]
```

- TEX parser library is used.
- Current version doesn't recognise `marvosym` commands.
- Unknown commands are ignored.
- `bib2gls` doesn't mind empty sort values. (Fallbacks are only used if the required field *isn't set* not if the value ends up empty after pre-processing.)
- Duplicate sort values ordered relative to each other according to `identical-sort-action` option. Default is non-locale case-sensitive letter ordering of the entry label.

symbol-sort-fallback=name

```
          pi \ensuremath{\pi}
           e \ensuremath{e}
       root2 \ensuremath{\surd 2}
 gelfondcons \ensuremath{e\sp\pi}
   heartsuit \ensuremath{\heartsuit}
   spadesuit \ensuremath{\spadesuit}
 diamondsuit \ensuremath{\diamondsuit}
    clubsuit \ensuremath{\clubsuit}
       email \Email
    envelope \Letter
       phone \Mobilefone
    landline \Telefon
```

symbol-sort-fallback=name

pi \ensuremath{\pi} ➜ π

e \ensuremath{e} ➜ e

root2 \ensuremath{\surd 2} ➜ $\sqrt{2}$

gelfondcons \ensuremath{e\sp\pi} ➜ e^{π}

heartsuit \ensuremath{\heartsuit} ➜ ♡

spadesuit \ensuremath{\spadesuit} ➜ ♠

diamondsuit \ensuremath{\diamondsuit} ➜ ♢

clubsuit \ensuremath{\clubsuit} ➜ ♣

email \Email ➜ *empty*

envelope \Letter ➜ *empty*

phone \Mobilefone ➜ *empty*

landline \Telefon ➜ *empty*

Duplicate Sort Values

- The `letter-nocase` sort method puts empty sort values at the start.

Duplicate Sort Values

- The `letter-nocase` sort method puts empty sort values at the start.
- email, envelope, phone and landline will all go at the start of the list.

Duplicate Sort Values

- The `letter-nocase` sort method puts empty sort values at the start.
- email, envelope, phone and landline will all go at the start of the list.
- The relative order is determined by the label: email, envelope, landline, phone.

Glossary

🖃 email
🖂 letter
☎ telephone
📱 mobile phone
e (2.71828) Euler's number
e^{π} (23.1406926) Gelfond's constant
$\sqrt{2}$ (1.41421) Pythagoras' constant
♠ spade
♡ heart
♢ diamond
♣ club
π (3.14159) ratio of circumference of a circle to its diameter

Different Fallbacks for Numbers and Symbols

- What if I want to sort pictographs by the description and constants by their name field?

Different Fallbacks for Numbers and Symbols

- What if I want to sort pictographs by the description and constants by their name field?
  - Use `symbol-sort-fallback` for `@symbol`

Different Fallbacks for Numbers and Symbols

- What if I want to sort pictographs by the description and constants by their name field?
  - Use `symbol-sort-fallback` for `@symbol`
  - Use `custom-sort-fallbacks` for `@number`

```
\GlsXtrLoadResources[
 symbol-sort-fallback=description,
 custom-sort-fallbacks={number=name},
 ...]
```

Different Fallbacks for Numbers and Symbols

- What if I want to sort pictographs by the description and constants by their name field?
  - Use `symbol-sort-fallback` for `@symbol`
  - Use `custom-sort-fallbacks` for `@number`

```
\GlsXtrLoadResources[
 symbol-sort-fallback=description,
 custom-sort-fallbacks={number=name},
 ...]
```

- Description contains words or phrases so the non-locale case-sensitive letter sort is no longer appropriate.

```
\GlsXtrLoadResources[sort=en,% English alphabet
symbol-sort-fallback=description,
custom-sort-fallbacks={number=name},
src={constants,pictographs},% bib files
selection=all% select all entries
]
```

Constants (name) and Pictographs (description)

Glossary

$\sqrt{2}$ (1.41421) Pythagoras' constant
♣ club
♢ diamond
e (2.71828) Euler's number
✉ email
e^{π} (23.1406926) Gelfond's constant
♡ heart
✉ letter
📱 mobile phone
♠ spade
☎ telephone
π (3.14159) ratio of circumference of a circle to its diameter

Split into Groups

- What if I want pictographs (ordered by description) and constants (ordered numerically) in separate groups?

Split into Groups

- What if I want pictographs (ordered by description) and constants (ordered numerically) in separate groups?
- Use multiple `\GlsXtrLoadResources`

```
\GlsXtrLoadResources[sort=en,% English alphabet
symbol-sort-fallback=description,
group={glssymbols},
src={pictographs},% bib file
selection=all% select all entries
]
\GlsXtrLoadResources[sort=double,% numeric
symbol-sort-fallback=symbol,
group={glsnumbers},
src={constants},% bib file
selection=all% select all entries
]
```

Split into Groups

- What if I want pictographs (ordered by description) and constants (ordered numerically) in separate groups?
- Use multiple `\GlsXtrLoadResources`

```
\GlsXtrLoadResources[sort=en,% English alphabet
symbol-sort-fallback=description,
group={glssymbols},
src={pictographs},% bib file
selection=all% select all entries
]
\GlsXtrLoadResources[sort=double,% numeric
symbol-sort-fallback=symbol,
group={glsnumbers},
src={constants},% bib file
selection=all% select all entries
]
```

- Use `--group`

Glossary

Symbols

♣ club
♢ diamond
✉ email
♡ heart
✉ letter
📱 mobile phone
♠ spade
☎ telephone

Numbers

$\sqrt{2}$ (1.41421) Pythagoras' constant
e (2.71828) Euler's number
π (3.14159) ratio of circumference of a circle to its diameter
e^{π} (23.1406926) Gelfond's constant

Alternatives

- Use separate glossaries.

```
\newglossary*{pictographs}{Pictographs}
\newglossary*{constants}{Constants}
\GlsXtrLoadResources[type={pictographs},
 src={pictographs}, sort=en, ...]
\GlsXtrLoadResources[type={constants},
 src={constants}, sort=double, ...]
```

Alternatives

- Use separate glossaries.

```
\newglossary*{pictographs}{Pictographs}
\newglossary*{constants}{Constants}
\GlsXtrLoadResources[type={pictographs},
 src={pictographs}, sort=en, ...]
\GlsXtrLoadResources[type={constants},
 src={constants}, sort=double, ...]
```

- Use hierarchical entries.

```
@indexplural{constant}
@number{pi,parent=constant,
name={\ensuremath{\pi}},
description={ratio of circumference of a circle
to its diameter},symbol={3.14159}}
```

Further Examples

Further examples can be found at `dickimaw-books.com/gallery`