

fref, reference list formatter for *troff*

John Ankarström

Introduction

fref (“format reference”) is a *troff* preprocessor that formats reference lists embedded in the *troff* source code according to the Harvard system. It supports multiple languages and is macro package-independent.

fref has been developed to address concerns unsatisfactorily met by previous solutions. It differs from the much beloved *refer* in the following ways:

1. *fref* is a reference *formatter*, not a reference *resolver* like *refer*. It formats references specified in the *troff* source, not in an external database.
2. As a result of the first point, *fref* requires no macro package support, unlike *refer*, which relies on macro packages to do the formatting.
3. As a further result of the first point, *fref* is a much simpler program than *refer* and is thus easier for the user to extend according to his own preferences.

Because *fref* does not resolve references, it is limited in its functionality. It is not a good fit for reference styles that require elaborate inline citations. For this reason, *fref* is primarily designed for reference styles with simple inline citations, such as the Harvard system, that the user can type himself without much trouble. Indeed, for parenthetical citations, it is likely quicker to manually type the citation than to instruct the computer to generate it. In such cases, I hope you will find *fref* to be a very fitting solution.

If you do use the Harvard system, *fref* offers many benefits over *refer*. While the format of *refer* references depend on the macro package one uses, the output of *fref* is completely independent of any macro package. Furthermore, *fref* supports a richer variety of reference fields, such as translator and hypertext reference.

Operation

The *fref* program takes *troff* source code on standard input and emits the same on standard out-

put. Lines beginning with ‘%’ (percent) are processed specially. Each such line denotes a field in a reference. For example:

```
.SH
References
.XP
%au Baudouin de Courtenay, J.
%da 1972
%ti The Difference between Phonetics and ...
%bo A Baudouin de Courtenay Anthology
%ed T. A. Sebeok et al.
%tr Edward Stankiewicz
%ci Bloomington
%pu Indiana University Press
.XP
... another reference ...
```

fref collects all percent lines until the next non-percent line, at which point a formatted reference is generated from the information specified in the fields. In the example above, the second *.XP* line will trigger *fref* to print a reference containing all the fields gathered thus far.

If, for some reason, a specified field is left out by *fref*, a warning is emitted on standard error. This might happen if an *%ed* (editor) field has been specified without a corresponding *%bo* (book) or *%jo* (journal) field.

If an unknown field is specified, *fref* dies.

The hard-coded reference style includes some language-specific words. The language (English by default) is controlled with the *-l* flag:

```
$ <example.t fref -lsv # Swedish
$ <example.t fref -lru # Russian
```

Reference fields

fref supports a large number of fields. All have two-letter names, much like *troff* requests:

%au *Author*. This is the only field of which more than one instance is allowed. Note that the author name is output as is.

%ad *Access date*. This field is output after a hypertext reference.

`%bo` *Book*.
`%ci` *City*.
`%da` *Date (year)*.
`%ed` *Editor*.
`%hr` *Hypertext reference*.
`%jo` *Journal*.
`%la` *Label*. This field is output at the beginning of the reference, followed by an equals sign.
`%lg` *Reference-specific language*. This field temporarily overrides the `-l` flag and the default language of English for a single reference.
`%no` *TODO: Issue number*.
`%pp` *Pages*.
`%pu` *Publisher*.
`%se` *TODO: Series*.
`%ti` *Title*.
`%tr` *Translator*.
`%vo` *Volume*.
`%xx` *Extra information*. The text is output at the end of the reference, but before any hypertext reference and access date.

Source code

Users of *fref* are encouraged to modify the source code according to their own requirements. *fref* is written in C using *lex* and contains circa 330 lines of code.