NAME

refer - preprocess bibliographic references for groff

SYNOPSIS

refer [**-benvCPRS**] [**-a***n*] [**-c** *fields*] [**-f***n*] [**-i** *fields*] [**-k** *field*] [**-l***m*,*n*] [**-p***filename*] [**-s** *fields*] [**-t***n*] [**-B** *field.macro*] [filename. . . *]*

DESCRIPTION

This file documents the GNU version of **refer**, which is part of the groff document formatting system. **refer** copies the contents of *filename* . . . to the standard output, except that lines between **.[** and **.]** are interpreted as citations, and lines between **.R1** and **.R2** are interpreted as commands about how citations are to be processed.

Each citation specifies a reference. The citation can specify a reference that is contained in a bibliographic database by giving a set of keywords that only that reference contains. Alternatively it can specify a reference by supplying a database record in the citation. A combination of these alternatives is also possible.

For each citation, **refer** can produce a mark in the text. This mark consists of some label which can be separated from the text and from other labels in various ways. For each reference it also outputs **groff** commands that can be used by a macro package to produce a formatted reference for each citation. The output of **refer** must therefore be processed using a suitable macro package. The **-ms** and **-me** macros are both suitable. The commands to format a citation's reference can be output immediately after the citation, or the references may be accumulated, and the commands output at some later point. If the references are accumulated, then multiple citations of the same reference will produce a single formatted reference.

The interpretation of lines between **.R1** and **.R2** as commands is a new feature of GNU **refer**. Documents making use of this feature can still be processed by Unix refer just by adding the lines

```
.de R1
.ig R2
..
```

to the beginning of the document. This will cause **troff** to ignore everything between **.R1** and **.R2**. The effect of some commands can also be achieved by options. These options are supported mainly for compatibility with Unix refer. It is usually more convenient to use commands.

refer generates **.lf** lines so that filenames and line numbers in messages produced by commands that read **refer** output will be correct; it also interprets lines beginning with **.lf** so that filenames and line numbers in the messages and **.lf** lines that it produces will be accurate even if the input has been preprocessed by a command such as soelim(1).

OPTIONS

It is possible to have whitespace between a command line option and its parameter.

Most options are equivalent to commands (for a description of these commands see the **Commands** subsection):

-b	**no-label-in-text; no-label-in-reference**
-e	**accumulate**
-n	**no-default-database**
-C	**compatible**
-P	**move-punctuation**
-S	**label "(A.n|Q) ’, ’ (D.y|D)"; bracket-label " (") "; "**
-a*n*	**reverse A***n*
-c *fields*	**capitalize** *fields*
-f*n*	**label %***n*

-i *fields* **search-ignore** *fields*

-k **label L~%a**

-k *field* **label** *field*~**%a**

-l **label A.nD.y%a**

-l*m* **label A.n+***m***D.y%a**

-l,*n* **label A.nD.y-***n***%a**

-l*m***,***n* **label A.n+***m***D.y-***n***%a**

-p *filename*
database *filename*

-s*spec* **sort** *spec*

-t*n* **search-truncate** *n*

These options are equivalent to the following commands with the addition that the filenames specified on the command line are processed as if they were arguments to the **bibliography** command instead of in the normal way:

-B **annotate X AP; no-label-in-reference**

-B *field***.***macro*
annotate *field macro***; no-label-in-reference**

The following options have no equivalent commands:

-v Print the version number.

-R Don't recognize lines beginning with **.R1**/**.R2**.

USAGE

Bibliographic databases

The bibliographic database is a text file consisting of records separated by one or more blank lines. Within each record fields start with a **%** at the beginning of a line. Each field has a one character name that immediately follows the **%**. It is best to use only upper and lower case letters for the names of fields. The name of the field should be followed by exactly one space, and then by the contents of the field. Empty fields are ignored. The conventional meaning of each field is as follows:

%A The name of an author. If the name contains a title such as**Jr .** at the end, it should be separated from the last name by a comma. There can be multiple occurrences of the **%A** field. The order is significant. It is a good idea always to supply an **%A** field or a **%Q** field.

%B For an article that is part of a book, the title of the book.

%C The place (city) of publication.

%D The date of publication. The year should be specified in full. If the month is specified, the name rather than the number of the month should be used, but only the first three letters are required. It is a good idea always to supply a **%D** field; if the date is unknown, a value such as **in press** or **unknown** can be used.

%E For an article that is part of a book, the name of an editor of the book. Where the work has editors and no authors, the names of the editors should be given as **%A** fields and **, (ed)** or **, (eds)** should be appended to the last author.

%G US Government ordering number.

%I The publisher (issuer).

%J For an article in a journal, the name of the journal.

%K Keywords to be used for searching.

%L Label.

%N Journal issue number.

%O Other information. This is usually printed at the end of the reference.

%P Page number. A range of pages can be specified as*m***-***n*.

%Q The name of the author, if the author is not a person. This will only be used if there are no **%A** fields. There can only be one**%Q** field.

%R Technical report number.

%S Series name.

%T Title. For an article in a book or journal, this should be the title of the article.

%V Volume number of the journal or book.

%X Annotation.

For all fields except **%A** and **%E**, if there is more than one occurrence of a particular field in a record, only the last such field will be used.

If accent strings are used, they should follow the character to be accented. This means that the **AM** macro must be used with the **-ms** macros. Accent strings should not be quoted: use one \ rather than two.

Citations

The format of a citation is

.[*opening-text*
flags keywords
fields
.]*closing-text*

The *opening-text*, *closing-text* and *flags* components are optional. Only one of the *keywords* and *fields* components need be specified.

The *keywords* component says to search the bibliographic databases for a reference that contains all the words in *keywords*. It is an error if more than one reference if found.

The *fields* components specifies additional fields to replace or supplement those specified in the reference. When references are being accumulated and the *keywords* component is non-empty, then additional fields should be specified only on the first occasion that a particular reference is cited, and will apply to all citations of that reference.

The *opening-text* and *closing-text* component specifies strings to be used to bracket the label instead of the strings specified in the **bracket-label** command. If either of these components is non-empty, the strings specified in the **bracket-label** command will not be used; this behaviour can be altered using the **[** and **]** flags. Note that leading and trailing spaces are significant for these components.

The *flags* component is a list of non-alphanumeric characters each of which modifies the treatment of this particular citation. Unix refer will treat these flags as part of the keywords and so will ignore them since they are non-alphanumeric. The following flags are currently recognized:

This says to use the label specified by the **short-label** command, instead of that specified by the **label** command. If no short label has been specified, the normal label will be used. Typically the short label is used with author-date labels and consists of only the date and possibly a disambiguating letter; the **#** is supposed to be suggestive of a numeric type of label.

[Precede *opening-text* with the first string specified in the **bracket-label** command.

] Follow *closing-text* with the second string specified in the **bracket-label** command.

One advantages of using the **[** and **]** flags rather than including the brackets in *opening-text* and *closing-text* is that you can change the style of bracket used in the document just by changing the **bracket-label** command. Another advantage is that sorting and merging of citations will not necessarily be inhibited if the flags are used.

If a label is to be inserted into the text, it will be attached to the line preceding the **.[** line. If there is no such line, then an extra line will be inserted before the **.[** line and a warning will be given.

There is no special notation for making a citation to multiple references. Just use a sequence of citations, one for each reference. Don't put anything between the citations. The labels for all the citations will be attached to the line preceding the first citation. The labels may also be sorted or merged. See the description of the **<>** label expression, and of the **sort-adjacent-labels** and **abbreviate-label-ranges** command. A label will not be merged if its citation has a non-empty *opening-text* or *closing-text*. However, the labels for a citation using the **]** flag and without any *closing-text* immediately followed by a citation using the **[** flag and without any *opening-text* may be sorted and merged even though the first citation's *opening-text* or the second citation's *closing-text* is non-empty. (If you wish to prevent this just make the first citation's *closing-text* **\&**.)

Commands

Commands are contained between lines starting with **.R1** and **.R2**. Recognition of these lines can be prevented by the **-R** option. When a **.R1** line is recognized any accumulated references are flushed out. Neither **.R1** nor **.R2** lines, nor anything between them is output.

Commands are separated by newlines or **;**s. **#** introduces a comment that extends to the end of the line (but does not conceal the newline). Each command is broken up into words. Words are separated by spaces or tabs. A word that begins with **"** extends to the next **"** that is not followed by another **"**. If there is no such **"** the word extends to the end of the line. Pairs of **"** in a word beginning with **"** collapse to a single **"**. Neither **#** nor **;** are recognized inside **"**s. A line can be continued by ending it with\; this w orks everywhere except after a **#**.

Each command *name* that is marked with * has an associated negative command **no-***name* that undoes the effect of *name*. For example, the **no-sort** command specifies that references should not be sorted. The negative commands take no arguments.

In the following description each argument must be a single word; *field* is used for a single upper or lower case letter naming a field; *fields* is used for a sequence of such letters; *m* and *n* are used for a non-ne gative numbers; *string* is used for an arbitrary string; *filename* is used for the name of a file.

abbreviate* fields string1 string2 string3 string4

Abbreviate the first names of *fields*. An initial letter will be separated from another initial letter by *string1*, from the last name by *string2*, and from anything else (such as a **von** or **de**) by *string3*. These default to a period followed by a space. In a hyphenated first name, the initial of the first part of the name will be separated from the hyphen by *string4*; this defaults to a period. No attempt is made to handle any ambiguities that might result from abbreviation. Names are abbreviated before sorting and before label construction.

abbreviate-label-ranges* string

Three or more adjacent labels that refer to consecutive references will be abbreviated to a label consisting of the first label, followed by *string* followed by the last label. This is mainly useful with numeric labels. If *string* is omitted it defaults to **-**.

accumulate*

Accumulate references instead of writing out each reference as it is encountered. Accumulated references will be written out whenever a reference of the form

```
.[
$LIST$
.]
```

is encountered, after all input files have been processed, and whenever **.R1** line is recognized.

annotate* field string

field is an annotation; print it at the end of the reference as a paragraph preceded by the line

.*string*

If *string* is omitted it will default to **AP**; if *field* is also omitted it will default to **X**. Only one field can be an annotation.

articles string . . .
string . . . are definite or indefinite articles, and should be ignored at the beginning of **T** fields when sorting. Initially, **the**, **a** and **an** are recognized as articles.

bibliography filename . . .
Write out all the references contained in the bibliographic databases *filename* . . . This command should come last in a **.R1**/**.R2** block.

bracket-label string1 string2 string3
In the text, bracket each label with *string1* and *string2*. An occurrence of *string2* immediately followed by *string1* will be turned into *string3*. The default behaviour is

bracket-label *([. *(.] ", "

capitalize fields
Convert *fields* to caps and small caps.

compatible*
Recognize **.R1** and **.R2** even when followed by a character other than space or newline.

database filename . . .
Search the bibliographic databases *filename* . . . For each *filename* if an index *filename***.i** created by indxbib(1) exists, then it will be searched instead; each index can cover multiple databases.

date-as-label* string
string is a label expression that specifies a string with which to replace the **D** field after constructing the label. See the **Label expressions** subsection for a description of label expressions. This command is useful if you do not want explicit labels in the reference list, but instead want to handle any necessary disambiguation by qualifying the date in some way. The label used in the text would typically be some combination of the author and date. In most cases you should also use the **no-label-in-reference** command. For example,

date-as-label D.+yD.y%a*D.-y

would attach a disambiguating letter to the year part of the **D** field in the reference.

default-database*
The default database should be searched. This is the default behaviour, so the negative version of this command is more useful. **refer** determines whether the default database should be searched on the first occasion that it needs to do a search. Thus a**no-default-database** command must be given before then, in order to be effective.

discard* fields
When the reference is read, *fields* should be discarded; no string definitions for *fields* will be output. Initially, *fields* are **XYZ**.

et-al* string m n
Control use of **et al** in the evaluation of **@** expressions in label expressions. If the number of authors needed to make the author sequence unambiguous is *u* and the total number of authors is *t* then the last *t* - *u* authors will be replaced by *string* provided that *t* - *u* is not less than *m* and *t* is not less than *n*. The default behaviour is

et-al " et al" 2 3

include filename
Include *filename* and interpret the contents as commands.

join-authors string1 string2 string3
This says how authors should be joined together. When there are exactly two authors, they will be joined with *string1*. When there are more than two authors, all but the last two will be joined with *string2*, and the last two authors will be joined with *string3*. If*string3* is omitted, it will default to *string1*; if

string2 is also omitted it will also default to *string1*. For example,

join-authors " and " ", " ", and "

will restore the default method for joining authors.

label-in-reference*
When outputting the reference, define the string **[F** to be the reference's label. This is the default behaviour; so the negative version of this command is more useful.

label-in-text*
For each reference output a label in the text. The label will be separated from the surrounding text as described in the **bracket-label** command. This is the default behaviour; so the negative version of this command is more useful.

label string
string is a label expression describing how to label each reference.

separate-label-second-parts string
When merging two-part labels, separate the second part of the second label from the first label with *string*. See the description of the<> label e xpression.

move-punctuation*
In the text, move any punctuation at the end of line past the label. It is usually a good idea to give this command unless you are using superscripted numbers as labels.

reverse* string
Reverse the fields whose names are in *string*. Each field name can be followed by a number which says how many such fields should be reversed. If no number is given for a field, all such fields will be reversed.

search-ignore* fields
While searching for keys in databases for which no index exists, ignore the contents of *fields*. Initially, fields **XYZ** are ignored.

search-truncate* n
Only require the first *n* characters of keys to be given. In effect when searching for a given key words in the database are truncated to the maximum of *n* and the length of the key. Initially*n* is 6.

short-label* string
string is a label expression that specifies an alternative (usually shorter) style of label. This is used when the**#** flag is gi ven in the citation. When using author-date style labels, the identity of the author or authors is sometimes clear from the context, and so it may be desirable to omit the author or authors from the label. The**short-label** command will typically be used to specify a label containing just a date and possibly a disambiguating letter.

sort* string
Sort references according to **string**. References will automatically be accumulated. *string* should be a list of field names, each followed by a number, indicating how many fields with the name should be used for sorting. **+** can be used to indicate that all the fields with the name should be used. Also **.** can be used to indicate the references should be sorted using the (tentative) label. (The **Label expressions** subsection describes the concept of a tentative label.)

sort-adjacent-labels*
Sort labels that are adjacent in the text according to their position in the reference list. This command should usually be given if the **abbreviate-label-ranges** command has been given, or if the label expression contains a **<>** expression. This will have no effect unless references are being accumulated.

Label expressions

Label expressions can be evaluated both normally and tentatively. The result of normal evaluation is used for output. The result of tentative evaluation, called the *tentative label*, is used to gather the information that normal evaluation needs to disambiguate the label. Label expressions specified by the **date-as-label** and **short-label** commands are not evaluated tentatively. Normal and tentative evaluation are the same for all types of expression other than **@**, *****, and **%** expressions. The description below applies to normal evaluation, except where otherwise specified.

field
field n The *n*-th part of *field*. If*n* is omitted, it def aults to 1.

'*string***'** The characters in *string* literally.

@ All the authors joined as specified by the **join-authors** command. The whole of each author's name will be used. However, if the references are sorted by author (that is the sort specification starts with **A+**), then authors last names will be used instead, provided that this does not introduce ambiguity, and also an initial subsequence of the authors may be used instead of all the authors, again provided that this does not introduce ambiguity. The use of only the last name for the*i*-th author of some reference is considered to be ambiguous if there is some other reference, such that the first *i* - 1 authors of the references are the same, the *i*-th authors are not the same, but the *i*-th authors last names are the same. A proper initial subsequence of the sequence of authors for some reference is considered to be ambiguous if there is a reference with some other sequence of authors which also has that subsequence as a proper initial subsequence. When an initial subsequence of authors is used, the remaining authors are replaced by the string specified by the **et-al** command; this command may also specify additional requirements that must be met before an initial subsequence can be used. **@** tentati vely evaluates to a canonical representation of the authors, such that authors that compare equally for sorting purpose will have the same representation.

%*n*
%a
%A
%i
%I The serial number of the reference formatted according to the character following the **%**. The serial number of a reference is 1 plus the number of earlier references with same tentative label as this reference. These expressions tentatively evaluate to an empty string.

*expr****** If there is another reference with the same tentative label as this reference, then *expr*, otherwise an empty string. It tentatively evaluates to an empty string.

*expr***+***n*
*expr***-***n* The first (**+**) or last (**-**) *n* upper or lower case letters or digits of *expr*. Troff special characters (such as **\('a**) count as a single letter. Accent strings are retained but do not count towards the total.

*expr***.l** *expr* converted to lowercase.

*expr***.u** *expr* converted to uppercase.

*expr***.c** *expr* converted to caps and small caps.

*expr***.r** *expr* reversed so that the last name is first.

*expr***.a** *expr* with first names abbreviated. Note that fields specified in the**ab breviate** command are abbreviated before any labels are evaluated. Thus**.a** is useful only when you w ant a field to be abbreviated in a label but not in a reference.

*expr***.y** The year part of *expr*.

*expr***.+y**
The part of *expr* before the year, or the whole of *expr* if it does not contain a year.

*expr***.-y** The part of *expr* after the year, or an empty string if *expr* does not contain a year.

*expr***.n** The last name part of *expr*.

*expr1***~***expr2*
expr1 except that if the last character of *expr1* is **-** then it will be replaced by *expr2*.

expr1 expr2
The concatenation of *expr1* and *expr2*.

expr1|*expr2*
If *expr1* is non-empty then *expr1* otherwise *expr2*.

*expr1***&***expr2*
If *expr1* is non-empty then *expr2* otherwise an empty string.

*expr1***?***expr2***:***expr3*
If *expr1* is non-empty then *expr2* otherwise *expr3*.

<*expr*> The label is in two parts, which are separated by *expr*. Two adjacent two-part labels which have the same first part will be merged by appending the second part of the second label onto the first label separated by the string specified in the **separate-label-second-parts** command (initially, a comma followed by a space); the resulting label will also be a two-part label with the same first part as before merging, and so additional labels can be merged into it. Note that it is permissible for the first part to be empty; this maybe desirable for expressions used in the **short-label** command.

(*expr***)** The same as *expr*. Used for grouping.

The above expressions are listed in order of precedence (highest first); **&** and **|** have the same precedence.

Macro interface

Each reference starts with a call to the macro **]-**. The string**[F** will be defined to be the label for this reference, unless the **no-label-in-reference** command has been given. There then follows a series of string definitions, one for each field: string **[***X* corresponds to field *X*. The number register **[P** is set to 1 if the **P** field contains a range of pages. The **[T**, **[A** and **[O** number registers are set to 1 according as the **T**, **A** and **O** fields end with one of the characters **.?!**. The**[E** number re gister will be set to 1 if the **[E** string contains more than one name. The reference is followed by a call to the **][** macro. The first argument to this macro gives a number representing the type of the reference. If a reference contains a **J** field, it will be classified as type 1, otherwise if it contains a **B** field, it will type 3, otherwise if it contains a **G** or **R** field it will be type 4, otherwise if contains a **I** field it will be type 2, otherwise it will be type 0. The second argument is a symbolic name for the type: **other**, **journal-article**, **book**, **article-in-book** or **tech-report**. Groups of references that have been accumulated or are produced by the **bibliography** command are preceded by a call to the **]<** macro and followed by a call to the **]>** macro.

FILES

/usr/dict/papers/Ind	Default database.
*file***.i**	Index files.

refer uses temporary files. See the groff(1) man page for details where such files are created.

ENVIRONMENT

REFER If set, overrides the default database.

SEE ALSO

indxbib(1), lookbib(1), lkbib(1)

BUGS

In label expressions, <> expressions are ignored inside **.***char* expressions.

COPYING