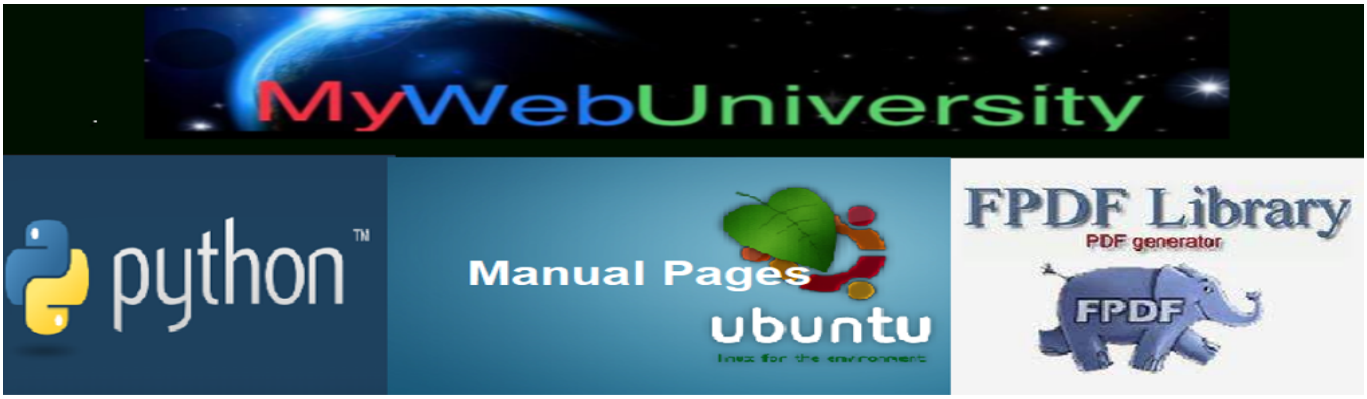

Rocky Enterprise Linux 9.2 Manual Pages on command 'grog.1'

$ man grog.1

GROG(1) General Commands Manual GROG(1)

NAME

grog - guess options for a following groff command

SYNOPSIS

grog [-C] [-T device] [--run] [--warnings] [--ligatures] [groff-option ...] [--] [filespec ...]

grog -h

grog --help

grog -v

grog --version

DESCRIPTION

grog reads the input (file names or standard input) and guesses which of the groff(1) options are needed to perform the input with the groff program. A suitable device is now always written as -Tdevice including the groff default as -T ps.

The corresponding groff command is usually displayed in standard output. With the option --run, the generated line is output into standard error and the generated groff command is run on the standard output. groffer(1) relies on a perfectly running groff(1).

OPTIONS

The option -v or --version prints information on the version number. Also -h or --help prints usage information. Both of these options automatically end the grog program. Other options are thenignored, and no groff command line is generated. The following 3 options are the only grog options,

-C this option means enabling the groff compatibility mode, which is also transfered

to the generated groff command line.

--ligatures

this option forces to include the arguments -P-y -PU within the generated groff command line.

--run with this option, the command line is output at standard error and then run on the computer.

--warnings

with this option, some more warnings are output to standard error.

All other specified short options (words starting with one minus character -) are inter? preted as groff options or option clusters with or without argument. No space is allowed between options and their argument. Except from the -marg options, all options will be passed on, i.e. they are included unchanged in the command for the output without effect? ing the work of grog.

A filespec argument can either be the name of an existing file or a single minus - to mean standard input. If no filespec is specified standard input is read automatically.

DETAILS

grog reads all filespec parameters as a whole. It tries to guess which of the following groff options are required for running the input under groff: -e, -g, -G, -j, -p, -R, -s, -t (preprocessors); and -man, -mdoc, -mdoc-old, -me, -mm, -mom, and -ms (macro packages).

The guessed groff command including those options and the found filespec parameters is put on the standard output.

It is possible to specify arbitrary groff options on the command line. These are passed on the output without change, except for the -marg options.

The groff program has trouble when the wrong -marg option or several of these options are specified. In these cases, grog will print an error message and exit with an error code.

It is better to specify no -marg option. Because such an option is only accepted and passed when grog does not find any of these options or the same option is found.

If several different -marg options are found by grog an error message is produced and the program is terminated with an error code. But the output is written with the wrong op? tions nevertheless.

Remember that it is not necessary to determine a macro package. A roff file can also be written in the groff language without any macro package. grog will produce an output without an -marg option.

As groff also works with pure text files without any roff requests, grog cannot be used to identify a file to be a roff file.

The groffer(1) program heavily depends on a working grog.

EXAMPLES

Calling

```
grog meintro.me
```

results in

```
groff -me meintro.me
```

So grog recognized that the file meintro.me is written with the -me macro package.

On the other hand,

```
grog pic.ms
```

outputs

```
groff -p -t -e -ms pic.ms
```

Besides determining the macro package -ms, grog recognized that the file pic.ms addition? ally needs -pte, the combination of -p for pic, -t for tbl, and -e for eqn.

If both of the former example files are combined by the command

```
grog meintro.me pic.ms
```

an error message is sent to standard error because groff cannot work with two different macro packages:

```
grog: error: there are several macro packages: -me -ms
```

Additionally the corresponding output with the wrong options is printed to standard out? put:

```
groff -pte -me -ms meintro.me pic.ms
```

But the program is terminated with an error code. The call of

```
grog -ksS -Tdvi grnexmpl.g
```

contains several groff options that are just passed on the output without any interface to grog. These are the option cluster -ksS consisting of -k, -s, and -S; and the option -T with argument dvi. The output is

```
groff -k -s -S -Tdvi grnexmpl.g
```

so no additional option was added by grog. As no option -marg was found by grog this file does not use a macro package.

AUTHORS

grog was originally written by James Clark. The current Perl implementation was written

by Bernd Warken ?groff-bernd.warken-72@web.de? with contributions from Ralph Corderoy, and is maintained by Werner Lemberg ?wl@gnu.org?.

SEE ALSO

groff(1), groffer(1)