

**NAME**

groff\_filenames – filename extensions for roff and groff

**DESCRIPTION**

Since the evolution of *roff* in the 1970s, a whole bunch of filename extensions for *roff* files were used.

The *roff* extensions refer to preprocessors or macro packages. These extensions are fixed in all *Unix*-like operating systems.

Later on, **groff** added some more extensions. This *man page* is about these filename extensions.

**COMPRESSION OF ROFF FILES**

Each *roff* file can be optionally *compressed*. That means that the *total filename* ends with a *compressor name*. So the whole filename has the structure `<name>.<extension>[.<compression>]`.

Best-known are the *compressor extensions* **.Z**, **.gz**, and **.bzip2**. Relatively new is **.xz**.

From now on, we will ignore the *compressions* and only comment the structure `<name>.<extension>`.

**MAN PAGES**

The *Unix manual pages* are widely called *man pages*. The *man page* style is the best known part of the *roff* language.

The extensions for *man* should be better documented. So this is documented here.

Files written in the *man* language use the following extension: `*.<section>[<group>]`.

**Man page Sections**

The traditional *man page* `<section>` is a digit from **1** to **8**.

`<name>.1`

`<name>.2`

`<name>.3`

`<name>.4`

`<name>.5`

`<name>.6`

`<name>.7`

`<name>.8`

Classic *man page sections*.

In older commercial *Unix* systems, the 3 characters **l**, **n**, and **o** were also used as *section* names. This is today *deprecated*, but there are still documents in this format.

`<name>.l`

`<name>.n`

`<name>.o`

*Deprecated* *man page sections*, which stood for “local”, “new”, and “old”, respectively.

**Man page Group Extensions**

The `<group>` extension in `.<section>[<group>]` is optional, but it can be any string of word characters. Usually programmers use a group name that is already used, e.g. **x** for X Window System documents or **tcl** to refer to the *Tcl* programming language.

Examples:

**groff.1** is the *man page* for **groff** in *section 1* without a *group*

**xargs.1posix.gz**

is the *man page* for the program **xargs** in *section 1* and *group* **posix**; moreover it is *compressed* with **gz** (**gzip**).

**config.5ssl**

*OpenSSL CONF* library configuration files from *section 5* with *group* **ssl**.

**dpkg-reconfigure.8cdebconf**

man page for the program **dpkg-reconfigure** in *section 8* and *group cdebconf*.

**Source of man pages**

There are 2 *roff* languages for writing man pages: **man** and **mdoc**.

The names of these 2 styles are taken as *extensions* for the source code files of man pages in the *groff* package.

**<name>.man**

*traditional* Unix-like man page format within groff source files.

**<name>.n**

A temporary man page file produced from a *name.man* man page by a run of **make** within the **groff** source package.

**<name>.mdoc**

Man page format in BSD.

**<name>.1b**

Man page format in *heirloom roff*.

**<name>.mandoc**

Files using this extension recognize both man page formats in **groff** and other processors.

**TRADITIONAL TROFF EXTENSIONS****Files Using Macro Packages**

The *classical roff* languages were interpreted by the traditional **troff** and **nroff** programs.

There were several *roff* languages, each represented by a *macro-package*. Each of these provided a suitable file name *extension*:

**<name>.me**

*roff* file using the **me** macro package.

**<name>.mm**

*roff* file using the **mm** macro package

**<name>.ms**

*roff* file using the **ms** macro package

All of these classical *roff* languages and their extensions are still very active in **groff**.

**Source Code for Macro Packages (TMAC Files)**

In traditional *roff* the source code for the macro packages was stored in *TMAC* files. Their file names have the form:

**tmac.<package>**,

*<package>* is the name of the macro package without the leading **m** character, which is reintegrated by the option **-m**.

For example, **tmac.an** is the source for the **man** macro package.

In the **groff** source, more suitable file names were integrated, see later on.

**Preprocessors**

Moreover, the following *preprocessors* were used as filename extension:

**<name>.chem**

for the integration of chemical formulas

**<name>.eqn**

for the mathematical use of equations

**<name>.pic**

graphical tool

<name>.tbl  
for tables with *tbl*

<name>.ref  
for files using the **prefer** *preprocessor*

### Classical Roff Files

<name>.t  
<name>.tr  
for files using the *roff* language of any kind

## NEW GROFF EXTENSIONS

*GNU roff* **groff** is the actual *roff* standard, both for classical *roff* and new extensions. So even the used new extensions in the source code should be regarded as actual standard. The following extensions are used instead of classical **.t** or **.tr**:

<name>.groff  
<name>.roff  
general ending for files using the *groff* language

### Source Code for Macro Packages (TMAC Files)

As the classical form **tmac**.<package\_without\_m>, of the *TMAC* file names is quite strange, *groff* added the following structures:

<package\_without\_m>.tmac  
**m**<package>.tmac  
**groff\_m**<package>.tmac

### Files Using new Macro Packages

*Groff* uses the following new macro packages:

<name>.mmse  
file with swedish **mm** *macros* for **groff**

<name>.mom  
files written in the *groff* macro package **mom**

<name>.www  
files written in **HTML**-like *groff* macros.

### Preprocessors and Postprocessors

<name>.hdtbl  
Heidelberger tables, an alternative to the preprocessor *tbl*. See **groff\_hdtbl**(7).

<name>.grap  
files written for the graphical **grap** processor.

<name>.grn  
for including **gremlin**(1), pictures, see **grn**(1).

<name>.pdfroff  
transform this file with **pdfroff** of the *groff* system

## AUTHORS

This document was written by Bernd Warken (groff-bernd.warken-72@web.de).

## SEE ALSO

History and future  
**roff**(7), **man-pages**(7), **groff\_diff**(7), **groff**(7)

*Compression*  
**uncompress**(1posix), **gzip2**(1), **bzip2**(1), **xz**(1)

A man page of the naming form *name*(*n*) can be read in text mode by  
**man** *n* *name*

or in graphical mode (PDF) by

**groffer** *n name*

*Gunnar Ritter's Heirloom roff project* (<https://github.com/n-t-roff/heirloom-doctools>). You can get this package with the shell command:

```
$ git clone https://github.com/n-t-roff/heirloom-doctools
```