# *Groff* PDF features

*Deri James*

deri@chuzzlewit.myzen.co.uk

July 2026

# Table of Contents

Introduction . . . 1
Meta Data . . . 1
  Document Info . . . 1
    pdf.tmac . . . 1
    mom . . . 2
    ms . . . 2
  Annotation Notes . . . 2
    pdf.tmac . . . 2
Bookmarks . . . 3
  pdf.tmac . . . 3
  mom . . . 3
  ms . . . 3
  man . . . 4
Named Destination . . . 4
  pdf.tmac . . . 4
  mom . . . 5
  ms . . . 5
Internal Hotspot Links . . . 5
  pdf.tmac . . . 5
  mom . . . 6
  ms . . . 6
  man . . . 6
External Hotspot Links . . . 7
  pdf.tmac . . . 7
  mom . . . 7
  man . . . 7
Table of Contents . . . 7
  pdf.tmac . . . 8
  mom . . . 8
  ms . . . 8
Slideshow Presentations . . . 8
  pdf.tmac . . . 9
  mom . . . 9
  ms . . . 9
Boxes and PDF paper colour . . . 9
  pdf.tmac . . . 9
  ms . . . 10
  mom . . . 10
For macro developers . . . 10
  Stop/Start hotlink . . . 10
  Forward References . . . 10
  Marking Hotlinks . . . 11

## 1. Introduction

This is an attempt to draw together information gleaned from **pdf.tmac**, **gropdf(1)**, and the various full service macros which manage pdf features.

Previously groff contained **pdfmark.tmac** and **spdf.tmac** (to integrate pdf features into the **ms** macros). Since these have now been dropped from groff and are now supported by Keith Marshall at his *groff−pdfmark* site.

PDF features can either be controlled at a low level, using the macros defined in **pdf.tmac** or rely on these macros being integrated into a full-service macro package. Currently this is the state of play with regard to full-service:-

| | | | Link | | |
|---|---|---|---|---|---|
| **Macro** | **Metadata** | **Bookmark** | **Internal** | **External** | **Named Destination** |
| **mom** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **man** | | ✓ | ✓[1] | ✓ | |
| **ms** | ✓ | ✓ | ✓ | | ✓ |
| **me** | | | | | |
| **mm** | | | | | |

The macros outlined below cannot be mix'n'matched between full service macro sets, but the information given for **pdf.tmac** can be used in any document, particularly if a full macro set is missing a particular facility.

## 2. Meta Data

### 2.1. Document Info

A PDF document can contain meta data such as "Created Date", "Author", "Title", etc..

#### 2.1.1. pdf.tmac

**pdf.tmac**

To enter pdf meta data use:

**.pdfinfo /<label>** "*text*"

Where **/<label>** can be:-

**/Title**, **/Author**, **/Subject**, **/Keywords**

And ***text*** can be multi-line if you use:-

***[PDFLB]** where you want the line-break.

These entries are stored in the pdf and can be seen if you run the command:-

**pdfinfo** *filename*

Or are probably visible if you look at "Document Properties" in your favourite pdf viewer.

[1] In a man page collection the MR macro can be used to link to other entries in the collection, otherwise it is an external link.

### 2.1.2. mom

**mom**

In **mom** you can use:-

**.DOCTITLE** ***text***

or more usually

**.PDF_TITLE** ***text***

The text given to:-

**.AUTHOR** ***text***

Will be added to the pdf meta data.

### 2.1.3. ms

**ms**

**.TL** and **.AU** Provide the Title and Author for the pdf.

## 2.2. Annotation Notes



These are the clickable/hoverable icons which appear in the document which can contain pop-up comments on the text. Only the **pdf.tmac** macros provide access to this facility.

### 2.2.1. pdf.tmac

**pdf.tmac**

**.pdfnote** [ -T ***title*** ] ***text***

The popup note in the left margin was created with:-

**.pdfnote** -T "Deri James" Just an example

The colour of the icon is set by setting the register **PDFNOTE.COLOUR** (or **PDFNOTE.COLOR**) to three numbers each in the range 0.0 ⇒ 1.0 representing red, green, blue values, so:-

.ds **PDFNOTE.COLOUR** 1.0 1.0 0.0

Produces a yellow icon. The opacity of the icon is controlled by:-

.ds **PDFNOTE.OPACITY** 0.0 ⇒ 1.0

Where the default value is 0.6.

Again you can use ***[PDFLB]** in the ***text***.

## 3. Bookmarks

For PDFs, bookmarks are the entries in the outline panel. Each bookmark has a hierarchical level, and, optionally, can be "named" so that they can be linked to from elsewhere in the document (see: "Named Destination").

Bookmarks in the panel can be open or closed, setting:-

.ds **PDFOUTLINE.FOLDLEVEL** 1

will "close" all bookmarks below the first level. The default is to open all bookmarks.

### 3.1. pdf.tmac

**pdf.tmac**

**pdf.tmac** defines this macro for bookmarks:-

**.pdfbookmark** [**-T** ***name***] ***level descriptive text ...***

Where "level" is the mandatory nesting level for this "bookmark". All the "descriptive text" form the entry in the outline panel. If the **-T** flag is given this becomes a "named destination" which can be linked to from elsewhere in the document (see: "Named Destination").

### 3.2. mom

**mom**

mom uses the **.HEADING** macro to populate the outline panel:-

**.HEADING** ***level*** [**NAMED** ***name***] ***descriptive text ...***

"level", "name" and "descriptive text" have the same meanings as above.

### 3.3. ms

**ms**

There are two separate ways of providing outline bookmarks in **ms**. First there is the traditional heading:-

**.NH** ***level*** [***name***[2]]

This is then followed by input which is used as "descriptive text".

Although there could be several lines of input text, it is the accumulated output line which is used as "descriptive text".

So this input:

```
.NH 1 using
Using
.BI groff
with the
.BI ms
Macro Package
.XS
Using
.BI groff
with the
.BI ms
Macro Package
.XE
```

Would create a named (“using”) level 1 bookmark “Using *groff* with the *ms* Macro Package”, and also create a **TOC** entry. Note if the input text results in multiple output lines it is only the first output line which is used as the bookmark text.

The second method, which combined the “heading” with populating a **TOC**, is the **.XN** command which **must** immediately follow the **.NH** command:-

**.XN** ***descriptive text***

Previously the **.NH** and following heading text had to be followed with **.XS/.XE** commands with the same text to separately populate the **TOC**.

So the equivalent code to the above using **.XN** is:-

```
.NH 1 using
.XN Using \fIgroff\fP with the \fIms\fP Macro Package
```

In both cases the bookmark is named “using”, and clicking on the overview panel or the hotspot link in the **TOC** will jump to the correct part of the document. You can also use the .XR command to jump to the section named “using”:-

.XR using ). (

to enter this hotlink “(see: Using *groff* with the *ms* Macro Package)”.

### 3.4. man

**man**

The

**.TH** ***identifier section***

macro creates (via first call to **.PT**) a named heading with name id of “identifier(section)”. It is normally given a level 1 bookmark, which can be controlled by setting “an*bookmark-base-level” on the command line to increase the level, this can be used in man page collections where the pages are split into separate sections.

The **.SH** and **.SS** macros create unnamed bookmarks at the appropriate level.

## 4. Named Destination

In order to insert hotspot links which jump to other parts of the document, the destination must be “named”. In the section on bookmarks it showed how they can be “named”.

If you want to jump to a place in the document which has not got a bookmark, you can name any particular place in the document, for example, a table or a figure to which you may want to refer. See information below.

### 4.1. pdf.tmac

**pdf.tmac**

This macro will plant a named destination without inserting a bookmark.

**.pdfhref M** [**-N** ***name***] [**-E**] [***descriptive text***]

If no ***name*** is specified, the first word of “descriptive text” is used as the “name”, which means that if the **-N** flag is not used then there must be at least one word in the text.

The “descriptive text”, if any, is saved as the “value” of the destination, and can be used when specifying a hotspot link (see: “Internal Hotspot Links”).

[2] If the “level” is “S” (a Berkeley extension) you cannot have a “named” bookmark.

If the **-E** flag is used, the text is output to the document as well.

### 4.2. mom

**mom**

The mom macro:-

**.PDF_TARGET** ***name*** [***descriptive text***]

You can now use ***name*** as the destination of any hotspot link (see: “Internal Hotspot Links”).

### 4.3. ms

**ms**

The *ms* macro

**.XM** ***name*** [***descriptive text***]

Marks a position in the document to which **.XR** can place a hotspot link (see: “Internal Hotspot Links”).

## 5. Internal Hotspot Links

An internal link is to a “named destination” within the document, external are links using a URI.

A hotspot link is a clickable piece of text which will cause the viewer to jump to a Named Destination. The colour of the hotspot is controlled by setting the register **PDFHREF.COLOUR** (or **PDFHREF.COLOR**) to three numbers each in the range 0.0 ⇒ 1.0 representing red, green, blue values, so:-

.ds **PDFHREF.COLOUR** 0.00 0.35 0.60

Uses a turquoise text, which can be referenced as

**\m[*[PDFHREF.TEXT.COLOUR]]**.

Hotspots can have a border drawn around them, which is controlled by setting the register **PDFHREF.BORDER** to an array of 3 numbers which represent – horizontal corner radius, vertical corner radius, and border width. A zero radius gives square (not rounded) corners. The default is:-

.ds **PDFHREF.BORDER** 0 0 0

Which produces no border around the hotspot.

### 5.1. pdf.tmac

**pdf.tmac**

To plant a hotspot link:-

**.pdfhref L** [**-D** ***name***] [**-P** ***prefix-text***] [**-A** ***affixed-text***] [**--**] [***hotlink text***]

The target for the link is given by ***name*** which should be a destination you have named (see: “Named Destination”).

The ***prefix-text*** and ***affixed-text*** are placed around the ***hotlink text***.

If you used a ***descriptive text*** when naming a destination, you can retrieve the value with this code:-

```
.pdf:lookup name
.ie !'\*[pdf:lookup-result]'' \
. ds desc-txt \*[pdf:lookup-value]
.el \
. ds desc-txt Unknown
```

This technique is used by some of the full service macros to provide the “expando” characters “*” and “+”. When they are the final character in “hotlink text” the expando is replaced by ***desc-text***, surrounded by double quotes if the expando character is “+”.

### 5.2. mom

`mom`

mom uses this macro:-

`.PDF_LINK` ***name*** [`PREFIX` ***text***] [`SUFFIX` ***text***] ***hotlink text***

If the ***hotlink text*** is terminated with an expando char it is replaced by the value of the ***descriptive text*** when the target ***name*** was created.

### 5.3. ms

`ms`

An addition to the ms macro set is the `.XR` command, originally found in some versions of Keith Marshall’s `spdf.tmac`, our version now uses expandos

`.XR` ***name*** [ ***post*** [ ***pre*** [ ***hotlink text***] ] ]

If ***hotlink text*** is missing, the contents of `\*[spdf:txt_default]` is used instead. This defaults to “see: +” so, with the use of the terminating expando, the code

`.XR` intro ), (

would add a link (see: “Introduction”), if the first section of this document was:-

.NH 1 intro
.XN Introduction

### 5.4. man

`man`

The

`.MR` ***name section***

macro normally creates an external link (see: “External Hotspot Links”), suitable for linking to another man page, but if the other man page it is calling is part of the same man page collection, it creates an internal link within the document.

## 6. External Hotspot Links

External links are often links to resources on the internet such as website URLs.

### 6.1. pdf.tmac

**pdf.tmac**

The macro to link to internet resources is:-

`.pdfhref W` [`-D` ***URI***] [`-P` ***prefix-text***] [`-A` ***affixed-text***] [`--`] [***hotlink text***]

If a ***URI*** is not specified the ***hotlink text*** must be a valid ***URI*** itself, and the complete ***URI*** will be visible in the document. If a ***URI*** is given by specifying `-D` ***URI*** then only the ***hotlink text*** will be visible, but clicking it will launch the ***URI***.

The `-P` and `-A` flags operate in the same way as in `.pdfhref L` (see: “Prefix and Affix”).

### 6.2. mom

**mom**

External links are supported by:-

`.PDF_WWW_LINK` ***URI*** [`PREFIX` ***text***] [`SUFFIX` ***text***] ***hotlink text***

If no ***hotlink text*** is given, the ***URI*** is used as the ***hotlink text***. Expandos can be used as the last character of ***hotlink text*** and will insert the ***URI***.

### 6.3. man

**man**

The macro

`.MR` ***name section***

Generates an appropriate external ***URI*** such as:-

man:/groff(1)

except if the link is to another man page in the same collection.

## 7. Table of Contents

Some full service macro sets offer macros for building a Table of Contents, with PDF features these TOCs have clickable entries.

### 7.1. pdf.tmac

**pdf.tmac**

**pdf.tmac** has no specific macros to build Table of Contents, but it does have two macros which can insert a TOC into the appropriate position in the document:-

**.pdfpagename** ***name***

This assigns a name to the current page being rendered which can then be used in the second macro:-

**.pdfswitchtopage** [***when***] ***name***

Where ***name*** is a previously named page, and the optional ***when*** can be “before” or “after”, default is “before”. Alternatively the positions “top” and “bottom” can be used instead of a named page. This macro should be used before a new page is started.

### 7.2. mom

**mom**

The macro

**.AUTO_RELOCATE_TOC**

(must come before **.START**) will position the TOC appropriately in the document.

Pease see the *mom* documentation for details of TOC generation, by default the TOC entries are clickable.

### 7.3. ms

**ms**

The traditional way of populating a TOC was to use the **.XS/.XA/.XE** macros.

Groff 1.23.0 introduced the **.XN** macro which combined producing headings (see: “*ms* Heading Example”) with adding a TOC entry.

In either case the entry will be clickable.

If the document has an **.RP** entry, so a title page is produced, the TOC will be repositioned after the Title Page.

## 8. Slideshow Presentations

The groff pdf driver understands how to create pdfs to be used for slideshow presentations. An alternative is to use the postscript driver and gpresent which is downloadable because the original website is having difficulty downloading files larger than 32kb.

groff documents written to work with **present.tmac** *may* work with -Tpdf with minor alteration (and no need to run **presentps** and **ps2pdf**).

### 8.1. pdf.tmac

`pdf.tmac`

There are just two macros which control the slide presentation:-

`.pdftransition` *feature mode duration dimension motion direction scale bool*

These parameters are explained in the *gropdf*(1) man page, many of them are not required!

There are two "events" which trigger a transition: when a slide is first drawn ( *feature* = `SLIDE`) and when a new element is added to the slide ( *feature* = `BLOCK`). Both can have a different transition *mode*, such as `"Split | Blinds | Box | Wipe | Dissolve | Glitter | R"` all of which are explained (and some more!) in the man page.

Whenever you want the presentation to pause (because you have added a new element to the slide) use the macro:-

`.pdfpause`

and the new element will appear using the current transition set for `BLOCK`.

### 8.2. mom

`mom`

mom has full integration with pdf presentation mode, the documentation is here.

In addition the mom example documents include `slide-demo.mom` and its resultant pdf, so should be included in the examples/mom directory in the groff documentation.

### 8.3. ms

`ms`

An example of using `ms` to create a slideshow is here.

## 9. Boxes and PDF paper colour

PDF viewers usually show contents on a white background, the actual background in the PDF is really transparent (so it appears white), however it is possible to specify a colour for the page background.

In addition it is possible to specify framed coloured boxes on the page to contain the running contents. These boxes will flow onto following pages.

### 9.1. pdf.tmac

`pdf.tmac`

The relevent macros are:-

`.pdfbackground` *cmd left top right bottom weight*
`.pdfbackground off`
`.pdfbackground footnote` *bottom*

which produce a background rectangle on the page. The meanings of the parameters are explained in the *gropdf*(1) man page.

### 9.2. ms

ms

If you include the macro file **-msboxes** in the *groff* command, you will have access to these macros:-

**.BOXSTART SHADED** *colour* **OUTLINED** *colour* **INDENT** *size* **WEIGHT** *size*

begins a box, where the argument after **SHADED** gives the fill colour and that after **OUTLINED** the border colour. Omit the former to get a borderless filled box and the latter for a border with no fill. The specified **WEIGHT** (line thickness) is used if the box is **OUTLINED**.

**INDENT** precedes a value that leaves a gap between the border and the contents inside the box.

Each colour must be a defined groff colour name, and each size a valid *groff* numeric expression. The keyword/value pairs can be specified in any order.

Boxes can be stacked, so you can start a box within another box; usually the later boxes would be smaller than the containing box, but this is not enforced. When using **BOXSTART**, the left position is the current indent minus the **INDENT** in the command, and the right position is the left position (calculated above) plus the current line length and twice the indent.

**BOXSTOP**

takes no parameters. It closes the most recently started box at the current vertical position after adding its **INDENT** spacing.

### 9.3. mom

mom

mom has comprehensive control of boxes see the mom documentation.

## 10. For macro developers

If you want to add PDF features to your own macro packages here is some information which may be helpful.

### 10.1. Stop/Start hotlink

**.pdfmarksuspend/.pdfmarkresume**

If there is a piece of “hotlink text” which appears right at the bottom of the page and continues across to the start of the next page these macros can stop/restart the hotlink. Typically they appear at the start of a page footer macro (to prevent the footer text becoming part of the hotlink), and at the end of a page header macro (to restart the hotlinking from the previous page).

### 10.2. Forward References

Since groff is a single pass interpreter when it finds an Internal Hotspot Links it cannot plant a link if it has not “seen” the Named Destination to which it links, and the link will appear as Unknown resisting all mouse clicks - the same happens if the *destination* of the link is mis-typed and does not exist. The forward link problem can be solved: instead of:-

```
groff -Tpdf ...
```

use:-

```
pdfmom --roff ...
```

The command was originally developed for satisfying forward references in *mom* files (hence the name), but with the *--roff* flag it drops its reliance on *mom* and you can place any macro package on the command line.

### 10.3. Marking Hotlinks

The

`.pdfhref` *command*

general purpose macro, supports these commands:-

- **O** (Outline) This is the command which **.pdfbookmark** calls.
- **M** (Mark) Setup a "Named Destination".
- **L** (Local) Mark an "Internal Hotspot Links".
- **W** (WWW URI) Form an "External Hotspot Links".

These "low-level" calls all expect "descriptive text" or "hotlink text" to be passed as part of the macro call. Sometimes this is not convenient, consider the man macro pair `.MT/.ME` where

```
.MT jh@\:axis\:.se
J\[o ad]rgen H\[a ad]gg
.ME
```

The mail-to *URI* is on the `.MT` line, but the text to use for a *hotlink* is everything up to the following `.ME`.

The `.pdfhref` macro allows an `-S` flag, not documented above, which just turns on *hotlinking* text. It is then up to the macro author to arrange for it to be turned off at the appropriate point. In this example it should be "turned off" in the `.ME` macro.

The code to turn off the "hotlinking" and restore the text colour to what it was before the "hotlinking" started is:-

```
\X'pdf: markend'\m[\*[pdf:curcol]]
```