

This is an example document

For those who care

by

Oliver Atkinson

Easily creating good-looking PDFs from vim, or similar terminal text editors.
Using the power of GNU roff (groff) and MOM.

Table of Contents

Part 1: Why groff?	1
Part 2: Now we are styling	2
2.1. Basic macros	2
2.2. Headings	2
2.3. Another top-level heading	2
2.3.1. Sub category 1	2
2.3.1.1. Very specific heading	2
2.3.2. Sub category 2	3
2.3.2.1. Specific heading	3
2.4. Font styles	3
2.4.1. Bolid, Italic, etc	3
2.4.2. Extended styling	3
2.4.3. Breaks	3
2.4.4. Colors	5
2.5. Special characters	5
2.6. Footnotes	5
Part 3: Other useful links	6

Part 1

Why groff?

A profound quote

The idea of groff, is good. However, the execution is mediocre. This document, by reading both the source and the final product, are made in hopes that you will be able to successfully create a document using the mom macros. It will (some what obtusely¹) use a wide range of options - such that you can learn as much as possible. Here are the hurdles we are trying to overcome:

1. What macro set do you want?
2. Do you need to use a preprocessor? (For things such as tables)
3. Where do you find documentation for your macroset? MS's is in a mandoc but MOM's is online.

Answers:

1. We will use the mom macro set.
2. For most usage, no, but if we need tables then that is something that can be discussed.
3. The documentation for mom is online (linked throughout this document) and right here!

To start off, I suppose that you should know how to "build" this document. Open a terminal on nearly any UNIX system and type the following in the same directory as this source file. (Replacing "this-file" with the actual file's name.)

```
groff -mom -T pdf this-file.mom > out.pdf
```

¹As seen right here.

Part 2

Now we are styling

First thing's first. We need our basic document outline. *Technically* you can start up a mom document very quickly. But I'm not a big fan of the defaults, so I've created this document to adjust those. Regardless, however you like the superfluous aspects of the document, the content will all be set out the same.

2.1. Basic macros

First I will recommend taking this source file, finding the comment "EDIT NOTE" and deleting form there down. This will give you defaults that look like mine, and setup some fancy extra stuff (mostly documented in the comments.)

You will first start out with (most likely) a `.NEWCHAPTER <Title> .` This will get the document started, tracking chapter numbers for you. Therefor, if you end up re-arranging your chapters, or adding another one in the center, you don't have to change the chapter numbers in the document. Nice!

After this new chapter you will create a paragraph with

```
.PP  
<text>
```

This will automatically handle tab-indentation of your paragraphs. Not indenting the very first of a chapter, but indenting all subsiquent paragraphs.

2.2. Headings

Use the "heading" macro, we create headings. `.HEADING <heading level, 1 being biggest> <title>`

2.3. Another top-level heading

2.3.1. Sub category 1

Here we have a sub category. We use the following code: `.HEADING 2 "Sub category 1"`

2.3.1.1. Very specific heading

```
.HEADING 3 "Very specific heading"
```

2.3.2. Sub category 2

As you may have noticed, the chapter number is prefixed onto the heading numbers.

2.3.2.1. Specific heading

After some more headings, we are now back in a level 3 heading.

2.4. Font styles

2.4.1. Bold, Italic, etc

You will most likely need to use **bold**, *italic*, or ***bold-italic***.

```
\*[BD]bold\*[PREV],    \*[IT]italic\*[PREV],    or    \*[BDI]bold-  
italic\*[PREV]
```

2.4.2. Extended styling

There are also separate ways of doing **bold**, *italic*, or ***bold-italic***. You can also **extend** a word, but I'm not too sure as to the use-case for this. These latter ways are considered "hacky" and only to be used if the first option isn't working. (Because the font doesn't support them.) [These are documented here](#). As you probably noticed, the first character of this paragraph covers 2 lines. This is called a "dropcap", and is used like this:

```
.PP  
.DROPCAP T 2  
here are also separate...
```

If desired, you can [really](#) make them fancy. More you can also make a single underscore, ^{a^{superscript}}, or **SMALL CAPS**.

If you notice in that last paragraph, when we said "SMALL CAPS", the period was a space away from the word. This is because by default new lines prepend a space, so that when we have sentences in our source code, each being on a different line, or just splitting a long sentence like this one across multiple lines, there aren't two words jammed together. But, like in the instance we just saw, this isn't *always* desirable. For those cases we use `\c` to prevent the space from appearing. In the sourcecode for this document the prior sentence looks like this:

```
If you notice in that last paragraph, when we said "\c  
.SMALLCAPS  
Small caps\c  
.SMALLCAPS OFF  
, the period was a space away from the word.
```

2.4.3. Breaks

There are 2 simple breaks you need to be aware of `.br` and `.bp`. These are for break line and break page, respectively.

Here I have manually inserted a page break, using the `.bp` macro. You can think of it as “break page”.

Now that we are on a new page — we will see that a heading has appeared! I have set the defaults here to contain a page number and the current chapter’s title. Then on the opposite side it will have the book’s title and the page number.

You can see the heading changed on the opposite page.

2.4.4. Colors

I have gone ahead and initialized three colors, **red**, **green**, and **blue**. The colors have been initialized by running:

```
.NEWCOLOR RED #FF0000
.NEWCOLOR GREEN #00FF00
.NEWCOLOR BLUE #0000FF
```

and then by writing the sentence like this:

```
\m[red]red\m[black],          \m[green]green\m[black],          and
\m[blue]blue\m[black].
```

Note that the color “black” is predefined.

2.5. Special characters

We can also insert special characters such as ♣♠♥♦ and a great deal more ([see here.](#)) This is done by entering their escape code. The above was done by `\[CL]\[SP]\[HE]\[DI]`

2.6. Footnotes

Footnotes are perhaps one of the most¹ useful things.

Something you are writing\c

```
.FOOTNOTE
```

then need to make a note on

```
.FOOTNOTE END
```

then continue your sentence.

¹probably

Part 3

Other useful links

[Short document on using MOM](#)

[Official documentation](#)