The mdframed package [1]

auto-split frame environment

Marco Daniel, Elke Schubert 1.9 2013/04/27

The standard methods for framing text (`\fbox` or `\fcolorbox`) require you to handle page breaks by hand, meaning that you have to split the `\fbox` into two. The present package defines the environment `mdframed` which automatically deals with pagebreaks in framed text.

By defining new environments the user may choose between several individual designs.

Linked files: `mdframed-example-default.pdf` `mdframed-example-tikz.pdf` `mdframed-example-pstricks.pdf` `mdframed-example-texsx.pdf`

FYI: I create a repository for `mdframed` on github where you can download the current development status.

Contents

1. Motivation

Many users wish to (further) emphasize lemmata, definitions, proofs, etc. The package `mdframed` allows you to create environments with breakable frames. I think an example is the best way to demonstrate its properties.

[1] Extending the package `framed.sty`

Theorem 1.1 (Pythagorean theorem) *In any right triangle, the area of the square whose side is the hypotenuse is equal to the sum of the areas of the squares whose sides are the two legs.*

$$a^2 + b^2 = c^2$$

The frame was defined with the following settings.

```
\newmdtheoremenv[%
    outerlinewidth   = 2                ,%
    leftmargin       = 40               ,%
    rightmargin      = 40               ,%
    backgroundcolor  = yellow           ,%
    outerlinecolor   = blue             ,%
    innertopmargin   = \topskip         ,%
    splittopskip     = \topskip         ,%
    ntheorem         = true             ,%
 ]{theorem}{Theorem}[section]
\begin{theorem}[Pythagorean theorem]
...
\end{theorem}
```

2. Syntax

Required packages by `mdframed`

The package itself loads the packages

`kvoptions` `xparse` `etoolbox` `color`.

Depending on the option `framemthod` `mdframed` will load

`xcolor` `tikz` `pstricks`.

Load the package as usual:

```
\usepackage[<GLOBAL OPTIONS>]{mdframed}
```

Only the option `framemethod` should be loaded by the optional argument of `\usepackage`. All other options should be loaded with `\mdfsetup` or related environments. The package should be loaded after `amsthm` if you need the package.

Provided environment

The package defines only one environment with the following syntax:

```
\begin{mdframed}[<LOCAL OPTIONS>]
    <CONTENT>
\end{mdframed}
```

To create own environments with mdframed see section 4.

Autodetecting floats

mdframed detects whether the environment is used inside float or minipage environments. If you use mdframed in such an environment mdframed will use the option nobreak automatically.

Twoside-mode

If you are using mdframed inside twoside-mode you can set the option innermargin and outermargin (see section 5.2.1). The length will be ignored if you use the option usetwoside=false.

3. The frames

Normally you can say mdframed draws only some lines. To allow page breaks the following designs are supported. If you load the package with framemethod=default you can only draw a single line. Inside the gray boxes of the images below the text will be printed.

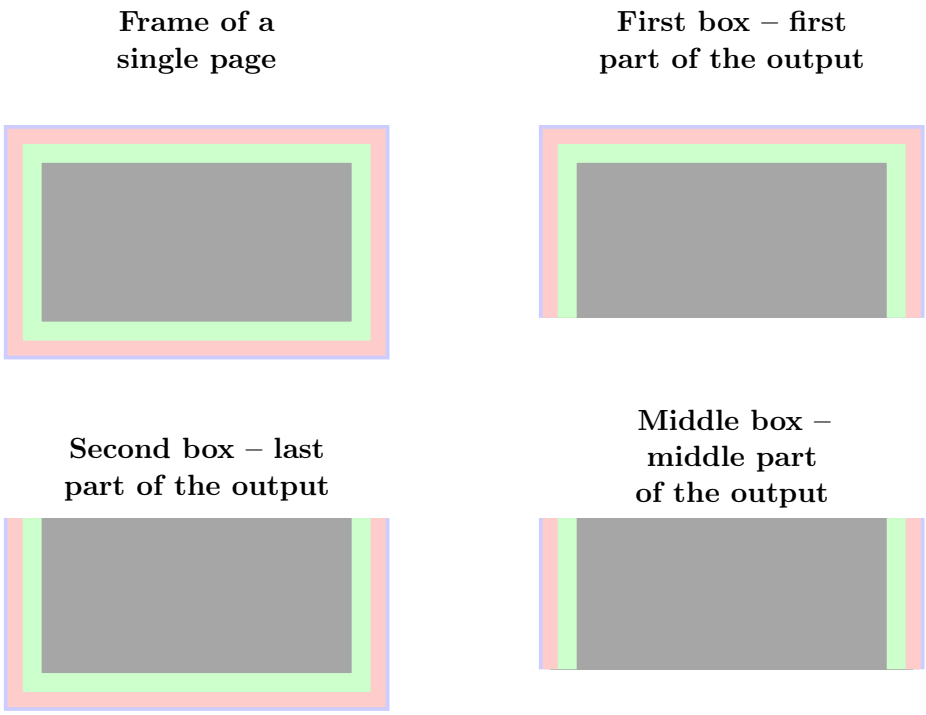

Figure 1: The basic frames

4. Commands

The following commands should countenance your by the handling with mdframed.

`\newmdenv[options]{env-name}`

The command allows the definition of a new environment which is surrounded by mdframed. The command has the following syntax:

\newmdenv[<MDFRAMED OPTIONS>]{Name of the environment}

In this way you can say:

```
\newmdenv[linecolor=red,frametitle=Infobox]{infobox}

\begin{infobox}[backgroundcolor=yellow]
  Some Infos\ldots
\end{infobox}
```

`\renewmdenv[options]{env-name}`

By using this command you can redefine environments which are created by `\newmdenv`.

`\surroundwithmdframed[options]{environment}`

Sometimes you have predefined environments. This commands allows you to surround an predefined `environment` with `mdframed` without changing the original name. To set a `mdframed` around the environment `verbatim` you can simple say.

```
\surroundwithmdframed[linewidth=2pt]{verbatim}
```

`\mdflength{options}`

If you want to work with length defined by `mdframed` (for example `innerleftmargin`) you can use the command `\mdflength`.

```
The distance is \hspace{\mdflength{innerleftmargin}}
\the\mdflength{innerleftmargin}
```

The result will be:

The distance is 10.0pt

`\mdfsetup{options}`

To set the options you can use the optional argument of `\usepackage` or you can use the command `\mdfsetup` which is not limited to the preamble. Inside a group the settings work only local.

At this point I want to recommend the usage of the command `\mdfsetup` instead of setting package option via the optional argument of `\usepackage`. So you are avoiding breaking of non robust commands.[2] The sole exception is the option `framemethod` which must given as an optional argument of `\usepackage`.

`\mdfdefinestyle{options}`

`\mdfdefinestyle` allows the user to define different styles which can be used as an option of `mdframed` via `style`. The option `style` is explained in section 5.2.3.

Here a small example:

```
\mdfdefinestyle{mystyle}{leftmargin=1cm,linecolor=blue}
\begin{mdframed}[style=mystyle]
  foo
\end{mdframed}
```

`\mdfapptodefinestyle{style name}{options}`

[2] Thanks to Heiko Oberdiek and Philipp Stephani kvoptions-Declaration von Optionen schlägt fehl

This commands add options to a defined style.[3]

5. Options

The package provides various options to manipulate frames. In the following section all options are listed. Some internal macros which can be manipulated are not shown in this documentation. The listed options are divided in global and local options. The global options can not be used inside `\mdfsetup`.

5.1. Global Options

The following options are only global options.

`xcolor` default=`none`

By setting this key, the package `xcolor` will be loaded with the given value(s). Without any value `mdframed` loads the package `color` without any options. If the package `xcolor` is already loaded the given option will be ignored. I recommend to load `xcolor` before `mdframed`.

`framemethod` default=`default`

With this key you can change the way frames are drawn. You can decide whether the frame is drawn with

1. LaTeX-commands `\hrule`, `\vrule`, `\rule`,
2. `TikZ` (the package `TikZ` will be loaded) or
3. `PSTricks` (the package `pstricks` will be loaded).

The option `framemethod` requires a string. Allowed combinations are listed in the following table.

Table 1: Allowed keys for `framemethod`

Method	Allowed keys
LaTeX-commands	`default, tex, latex, none, 0`
TikZ	`tikz, pgf, 1`
PSTricks	`pstricks, ps, postscript, 2`

FYI
It is independently whether the `method` is written with no, one or more capital letter.

Note
The manipulation of the frames depends on the option `framemethod`. For further information see below.

5.2. Global and Local Options

The options listed below can be set globally or locally and they are not limited to the preamble. I tried to define self explained names.

[3] Thanks to Martin Scharrer and Enrico Gregorio:
http://tex.stackexchange.com/questions/34684/argument-of-setkeys

`default`

`style=defaultoptions`

This is sets the default options defined by `mdframed`.

Here a small example:

```
\mdfsetup{%
   middlelinecolor=red,
   middlelinewidth=2pt,
   backgroundcolor=red!10,
   roundcorner=10pt}
\begin{mdframed}
 Text\par Text
\end{mdframed}

\begin{mdframed}[default]
 Text\par Text
\end{mdframed}
```

Text
Text

Text
Text

5.2.1. Options with lengths

In figure (2) you can see the adjustable lengths (compare also figure (1)) which will be described below. All lengths accept two kinds of input. The first one is a length (e.g. 2pt) and the second one is a number (e.g. 2) which will be multiplied by `1 defaultunit`. The figure shows three different colored frames.

`defaultunit` default=`pt`

see the sentence above.

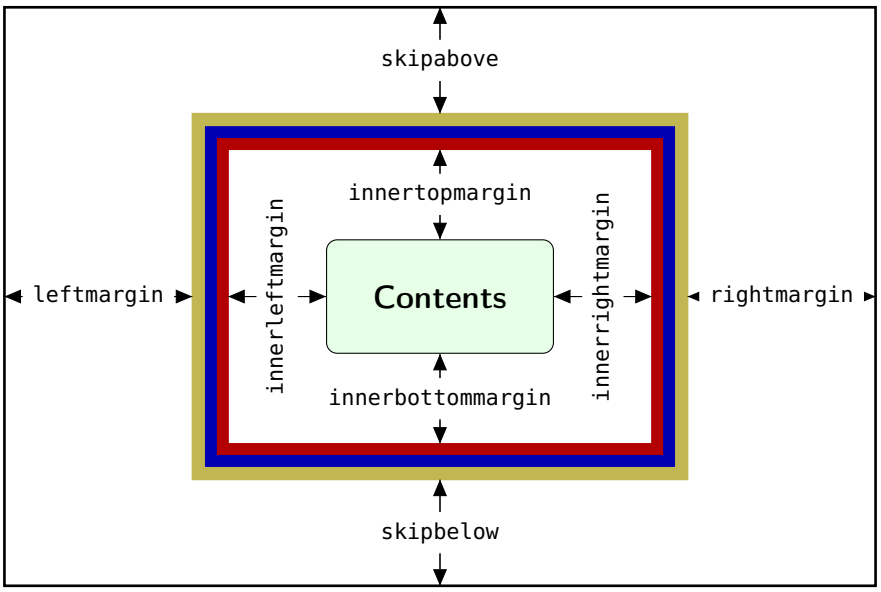

Figure 2: adjustable lengths of `mdframed`

`skipabove` default=`0pt`

Sets an additional skip above the frame.

`skipbelow` default=`0pt`

Sets an additional skip below the frame.

`margin` default=

This option is not longer supported. Use `leftmargin` and `rightmargin` instead.

`leftmargin` default=`0pt`

Sets the length of the left margin of the environment. This option has an effect only in singleside-mode or, in twoside-mode, if the option `usetwoside=false` has been given. See also options `outermargin` and `innermargin`.

`rightmargin` default=`0pt`

Sets the length of the right margin of the environment. This option has an effect only in singleside-mode or, in twoside-mode, if the option `usetwoside=false` has been given. See also options `outermargin` and `innermargin`.

`innerleftmargin` default=`10pt`

Sets the length of the inner left margin of the environment.

`innerrightmargin` default=`10pt`

Sets the length of the inner right margin of the environment.

`innertopmargin` default=`.4\baselineskip`

Sets the length of the inner top margin of the environment.

`innerbottommargin` default=`.4\baselineskip`

Sets the length of the inner bottom margin of the environment.

Before the next options are introduced here an example where the described length are used.

```
\mdfdefinestyle{mdfexample1}{leftmargin=1cm,rightmargin=2cm,%
  innerleftmargin=1cm,innerrightmargin=1cm,roundcorner=10pt}
\begin{mdframed}[style=mdfexample1]
  In any right triangle, the area of the square whose side is the hypotenuse
  is equal to the sum of the areas of the squares whose sides are the two
  legs.
\end{mdframed}
```

In any right triangle, the area of the square whose side is the hypotenuse is equal to the sum of the areas of the squares whose sides are the two legs.

The following lengths are not shown in figure (2). Of course you have some more length which can be manipulate.

`userdefinedwidth` default=`\linewidth`

Sets the width of the whole `mdframed` environment. The width represent the width including the line width and the inner margins. The outer margins will be ignored.

`outermargin` default=`0pt`

Sets the length of the outer margin. This option is only available in `twoside`-mode.

`innermargin` default=`0pt`

Sets the length of the inner margin. This option is only available in `twoside`-mode.

`splittopskip` default=`0pt`

Sets the length of the skip above the split part of the environment.

`splitbottomskip` default=`0pt`

Sets the length of the skip below the split part of the environment.

`linewidth` default=`0.4pt`

Sets the width of the line around the environment.
This works only with `framemethod=default`.

`roundcorner` default=`0pt`

Sets the size of the radius of the corners of the frames.
This works only with `framemethod=TikZ or PSTricks`.

`innerlinewidth` default=`0pt`

Sets the width of the inner line around the environment.
This works only with `framemethod=TikZ or PSTricks`.

`outerlinewidth` default=`0pt`

Sets the width of the outer line around the environment.
This works only with `framemethod=TikZ or PSTricks`.

`middlelinewidth` default=`0.4pt`

Sets the width of the middle line around the environment.
This works only with `framemethod=TikZ or PSTricks`.

5.2.2. Colored Options

Now we want to bring some color on your frames.

`linecolor` default=`black`

Sets the color of the line around the environment.

`backgroundcolor` default=`white`

Sets the color of the background of the environment.

`fontcolor` default=`black`

Sets the color of the contents of the environment.

`innerlinecolor` default=`linecolor`

Sets the color of the inner line around the environment.
This works only with `framemethod=TikZ or PSTricks`.

`middlelinecolor` default=`linecolor`

Sets the color of the middle line around the environment.
This works only with `framemethod=TikZ or PSTricks`.

`outerlinecolor` default=`linecolor`

Sets the color of the outer line around the environment.
This works only with `framemethod=TikZ or PSTricks`.

Ok after we have some length and some color we can improve our example.

```
\mdfapptodefinestyle{example1}{backgroundcolor=brown!20,%
    linecolor=red!40!black,linewidth=4pt}
\begin{mdframed}[style=mdfexample1]
  In any right triangle, the area of the square whose side is the hypotenuse
  is equal to the sum of the areas of the squares whose sides are the two
  legs.
\end{mdframed}
```

In any right triangle, the area of the square whose side is the hypotenuse is equal to the sum of the areas of the squares whose sides are the two legs.

5.2.3. General options

`everyline` default=`false`

Allows to draw a bottom and a top line at splitted frames.

`font` default=`{}`

Sets the font of the environment.

`ntheorem` default=`false`

Before setting this boolean key, you have to load the package `ntheorem`. With this option you set the values `\theorempreskipamount` and `\theorempostskipamount` to 0 pt.

`nobreak` default=`false`

Sometimes it is useful to prevent a frame from splitting. The `nobreak` option is used for this purpose. If you activate this option you can enable it by setting `nobreak=false`.

`usetwoside` default=`true`

If you set the twoside option you can work with `outermargin`. This option disable this and you work with `leftmargin` and `rightmargin`.

`needspace` default=`0pt`

Sometimes it is useful to set a minimum height before a frame should be splitted. For such cases you can use `needspace`. The option requires a length which sets the minimum height before a frame will be splitted.

`style`

If you define a special style with `\mdfdefinestyle` you can use the key `style` to load the style. `mdframed` has no predefined styles yet.

`settings` default=`none`

This option allows the user to commit some macros. An example is shown in the example files.

`align` default=`left`

Sometimes it is useful to align the environment itself. For this you have the option `align` which can be set to the following strings:

- `left`,
- `right` and
- `center`.

The alignments `left` or `right` depend on the given lengths `leftmargin` and `rightmargin`. Later I will present an example to demonstrate my bad English explanation.

`ignorelastdescenders` default=`false`

Try to ignore the last descenders of the environment `mdframed`. The complete idea was inspired by Tobias Weh and the solution was provided by Stefan Lemke. See How to make mdframed ignore descenders in last line

`draft` default=`false`

Activate the draft mode for the package. This option is useful for `framemethod=tikz` which clips the contents related to the surrounded frame. The options is motivated by the question mdframed+tikz to display overfullrule at tex.stackexchange.

`shadow` default=`false`

Draw a shadow. The shadow doesn't influence the bounding box so the shadow can be drawn in the margin without any overfull box. Note if you are using the TikZ you must load the library. `mdframed` doesn't do the job to avoid double loading of a library.

`shadowsize` default=`8 pt`

Specify the size of the shadow.

`shadowcolor` default=`black!50`

Specify the color of the shadow.

5.2.4. PSTricks options

`pstrickssetting` default=`{}`

With this key you can pass several options to `\psset`. For example if you want all lines dashed you will have to set `pstrickssetting={linestyle=dashed}`. It is very important to put the options of `pstrickssetting` in brackets.
This works only with `framemethod=PSTricks`.

`pstricksappsetting` default=`{}`

`mdframed` works with defined styles for the different elements. By using `\addtopsstyle` in combination with this option you can expand the definition. The predefined styles are

- `mdfbackgroundstyle`
- `mdfframetitlebackgroundstyle`
- `mdfouterlinestyle`
- `mdfinnerlinestyle`
- `mdfmiddlelinestyle`

Before you change one please have a look at the file `md-frame-2.mdf` to see the settings.
This works only with `framemethod=PSTricks`.

5.2.5. TikZ options

`tikzsetting` default=`{}`

With this key you can pass several options to `\tikzset`. Some examples are listed in the next section. It is very important to put the options of `tikzsetting` in brackets.
This works only with `framemethod=TikZ`.

`apptotikzsetting` default={}

With this key you can add several options to `tikzsetting`. This key based on the idea of manipulation of predefined keys of `mdframed`. The package `mdframed` defines via `\tikzset` the following keys to draw frames.

- `\tikzset{mdfbox/.style}`
- `\tikzset{mdfcorners/.style}`
- `\tikzset{mdfbackground/.style}`
- `\tikzset{mdfinnerline/.style}`
- `\tikzset{mdfouterline/.style}`
- `\tikzset{mdfmiddleline/.style}`
- `\tikzset{mdfframetitlerule/.style}`
- `\tikzset{mdfframetitlebackground/.style}`
- `\tikzset{mdfshadow/.style}`

Before you change one please have a look at the file `md-frame-1.mdf` to see the settings.
This works only with `framemethod=TikZ`.

5.3. Hooks and Bools

The following options and bool flags can be used by the any user. Of course some needs more experience than other.

`singleextra` default={}

With this key you can put extra material to the drawing environment of `mdframed` only for a non splitted frame.
This works only with `framemethod=TikZ and PSTricks`.

`firstextra` default={}

With this key you can put extra material to the drawing environment of `mdframed` only for the first part of the splitted frame.
This works only with `framemethod=TikZ and PSTricks`.

`middleextra` default={}

With this key you can put extra material to the drawing environment of `mdframed` only for the middle part of the splitted frame.
This works only with `framemethod=TikZ and PSTricks`.

`secondextra` default={}

With this key you can put extra material to the drawing environment of `mdframed` only for the second part of the splitted frame.
This works only with `framemethod=TikZ and PSTricks`.

After you can add material to any part of the frame you can also detect which part of the frame is drawn. Therefor the bool flags are defined which can be test by `\ifbool{boolflag}`. All flags are set to true before the output of the content occurs.

`mdfsingleframe` default=`false`

This bool is only true for a non splitting frame.

`mdffirstframe` default=`false`

This bool is only for the first part of the frame true.

`mdfmiddleframe` default=`false`

see above.

`mdflastframe` default=`false`

see above.

For the advanced users there are also some other hooks which can be used to manipulate the output:

`beforesingleframe` default=`{}`

Every given code to this option is executed before a single frame is is printed.

`aftersingleframe` default=`{}`

Every given code to this option is executed after a single frame is is printed.

`beforebreak` default=`{}`

The value of this option is only executed at breakable frames. Related to the introduction the first and middle frame can be manipulated.

`afterbreak` default=`{}`

The value of this option is only executed at breakable frames. Related to the introduction the first and middle frame can be manipulated.

`beforelastframe` default=`{}`

The option is executed only for the last frame of a splitted frame.

`afterlastframe` default=`{}`

The option is executed only for the last frame of a splitted frame.

5.4. complexe example – Matlab Terminal

The following example was inspiered by a question on TeX.SX.

Command Window

```
% >> help sin
%  sin    Sine of argument in radians.
%     sin(X) is the sine of the elements of X.
%
%     See also asin, sind.
%
%     Overloaded methods:
%        sdpvar/sin
%        codistributed/sin
%        gpuArray/sin
%
%     Reference page in Help browser
%        doc sin
%
% >>
%
```

The code for this result is:

```
\definecolor{DarkBlue}{rgb}{.11,.23,.60}
\mdfdefinestyle{commandline}%
{leftmargin=5pt, rightmargin=10pt,innerleftmargin=15pt,
```

```
  middlelinecolor=DarkBlue,
  middlelinewidth=2pt,
  frametitlerule=false,
  backgroundcolor=black!10!white,
  frametitle={Command Window},
  frametitlefont={\normalfont\sffamily\color{white}\hspace{-1em}},
  frametitlebackgroundcolor=DarkBlue,
  singleextra={\draw[black!20,line width=12pt]
          ($(O)+(7pt,1pt)$) --
          ($(O|-P)+(7pt,-\mdframetitleboxtotalheight)-(0,1pt)$);
          \node[inner sep=0pt,color=black]at ($(O)+(7pt,9pt)$)%
          {$\scriptstyle f\!x$}; },
  nobreak,
}

\lstnewenvironment{script} {%
   \lstset{language=Matlab,basicstyle=\tiny\ttfamily,breaklines=true,%
                aboveskip=0pt,belowskip=0pt}}{}
\surroundwithmdframed[style=commandline]{script}
\begin{script}
>> help sin
  sin Sine of argument in radians.
       sin(X) is the sine of the elements of X.

       See also asin, sind.

       Overloaded methods:
            sdpvar/sin
            codistributed/sin
            gpuArray/sin

       Reference page in Help browser
            doc sin

>>
\end{script}
```

5.5. Hidden Lines

`topline` default=`true`

Draws a line at the top.

`bottomline` default=`true`

Draws a line at the bottom.

`leftline` default=`true`

Draws a line on the left.

`rightline` default=`true`

Draws a line on the right.

`hidealllines` default=`false`

With this option you can decide whether all lines should be drawn or not.

5.6. Frametitle

In this section all relevant options of the frame title will be presented. They are not divided in their properties.

`frametitle` default=`none`

The environment gets a title. To set a title use `frametitle={The Title of the frame}` as an option of the environment.

`frametitlefont` default=`\normalfont\bfseries`

Sets the format of the `frametitle`.

`frametitlealignment` default=`\raggedleft`

Align the `frametitle`. This option must be set via `\mdfsetup`.

`frametitlerule` default=`false`

Set this key to `true` to get a line between the frame title and the text.

`frametitlerulewidth` default=`.2pt`

Sets the width of the line between the text and the title of `mdframed`.

`frametitleaboveskip` default=`5pt`

Sets the skip of the frame title to the margin above of `mdframed`.

`frametitlebelowskip` default=`5pt`

Sets the skip of the frame title to the rule of the frame title.

`frametitlebackgroundcolor` default=`backroundcolor`

Sets the color of the background of the frametitle

`repeatframetitle` default=`false`

Repeat the frame title on every frame.

The following picture demonstrates the behaviour of the lengths if the option `frametitle` is used.

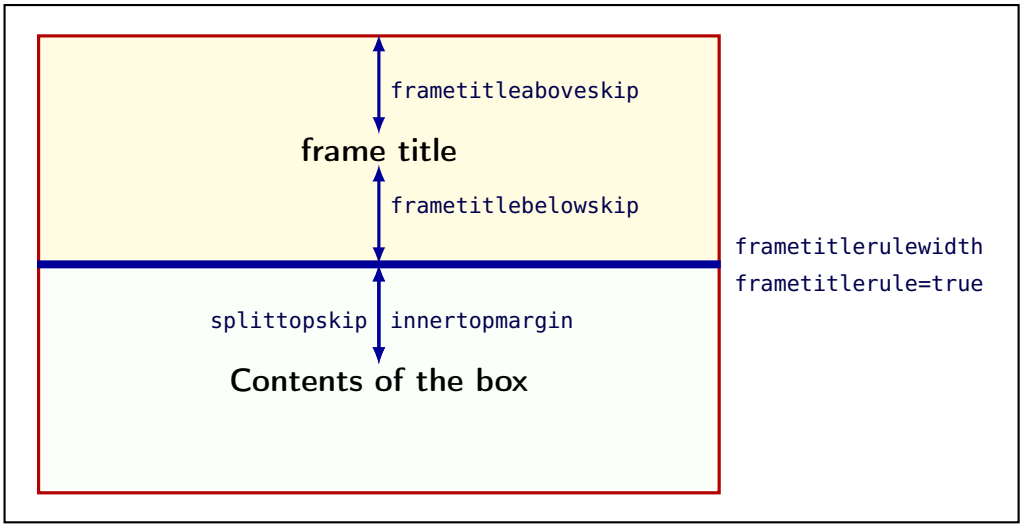

Figure 3: Behavior of the lengths if `frametitle` is used

FYI and Note

The splitting of the frame title is really a fiddly issue. If you want to use the option `repeatframetitle` a splitting is more than wrong. On the other hand if you use the option `repeatframetitle` the user must prepare the contents well.

5.7. Title commands inside the environment

To provide titles inside the environment `mdframed` you can one of the two following commands. The relavant options are listed below.

`\mdfsubtitle`

Set a title inside `mdframed` of the internal level 1.

`\mdfsubsubtitle`

Set a title inside `mdframed` of the internal level 2.

Both commands have the same syntax. They accept one optional and one mandatory argument. The optional argument sets the option of `mdframed` whereby everything will be local. The second argument of subtitle also allows paragraph breaking.

> **\mdfsubtitle**[<options>]{the subtitle}

5.7.1. Options related to the title of level 1

`subtitleaboveline` default=`false`

Decide to draw a line above the subtile.

`subtitlebelowline` default=`false`

Decide to draw a line below the subtitle.

`subtitlefont` default=`\normalfont\bfseries`

Sets the font for subtitles.

`subtitlebackgroundcolor` default=`white`

Sets the background color of the subtitle between the above and below line.

`subtitleabovelinecolor` default=`black`

Sets the line color of the line above.

`subtitlebelowlinecolor` default=`black`

Sets the line color of the line below.

`subtitleabovelinewidth` default=`0.8pt`

Sets the line width of the line above.

`subtitlebelowlinewidth` default=`0.6pt`

Sets the line width of the line below.

`subtitleaboveskip` default=`\baselineskip`

Sets the skip before the subtitle line above will be drawn.

`subtitlebelowskip` default=`1.2\baselineskip`

Sets the skip after the subtitle line below is drawn.

`subtitleinneraboveskip` default=`0.5\baselineskip`

Sets the skip after the line above and the subtitle itself.

`subtitleinnerbelowskip` default=`0.5\baselineskip`

Sets the skip after the subtitle and the line below.

Here an example to demonstrate the behaviour:

Some Text ...

Notes

Some Text ...

```
\newmdenv[%
 roundcorner=5pt,
 subtitlebelowline=true,subtitleaboveline=true,
 subtitlebackgroundcolor=yellow,
 backgroundcolor=blue!20!white
 ]{subtitleenv}
\begin{subtitleenv}
Some Text \ldots
\mdfsubtitle{Notes}
Some Text \ldots
\end{subtitleenv}
```

5.8. Theorems

In this section is described which commands can help you to define theorem environments with `mdframed`.

`\newmdtheoremenv`

Since the package is often used to highlight theorem environments, the package providesa command to simplify this process. The command has the following syntax:

```
\newmdtheoremenv[<mdframed-options>]{<envname>}%
                [<numberedlike>]{<caption>}[<within>]
```

The last four arguments are equivalent to the command `\newtheorem`. Only the first optional argument is able to pass `mdframed`-options. A simple example is:

```
\theoremstyle{<some style>}
\newmdtheoremenv[linecolor=blue]{lemma}{Lemma}[section]
...
\begin{lemma}[Some title]
   foo foo foo foo foo foo
\end{lemma}
```

So far there is no `\renewmdtheoremenv`!

`\mdtheorem`

This is a special kind of `\newtheorem`. The command has the following syntax.

```
\mdtheorem[<mdframed-options>]{<envname>}%
                [<numberedlike>]{<caption>}[<within>]
```

As you can see the arguments are equal to `\newtheorem` but the command ignores every `\theoremstyle`. This is based on the following behavior.

The command `\mdtheorem` creates two environments based on the given first mandatory argument. The first environment is named like the given argument and creates a numbered theorem. The second environment is named like the first mandatory argument with a star. This environment has the same formating but isn't numbered.

The syntax of the new defined environments is equal to the normal theorem environments.

```
\begin{environment}[optional title]
  ...
\end{environment}
```

What happened? The caption of the command will be set as the frame title. In this way all options of the frame title are available. Furthermore `mdframed` provides additional options explained below.

`theoremseparator` default=`{:}`

Sets the separator of the caption and the title of the theorem. The `theoremseparator` will be printed only if an theorem title is given.

`theoremtitlefont` default=`{}`

Via the option `frametitlefont` you can manipulate the font of the frame title. The option `theoremtitlefont` allows to set a different font to the title of the theorem.

`theoremspace` default=`\space`

Sets the space after `theoremseparator`.

Examples can be found in the attached files.

5.9. Footnotes

Inside the environment you can use the command `\footnote` as usual. `mdframed` uses the syntax of environment `minipage` with the same counter.

Every footnote text will be collected inside a box and will be displayed at the end of the environment `mdframed`.

`footnotedistance` default= `\bigskipamount`

The length is the distance between the end of the environment `mdframed` and the displaying of the `\footnoterule`.

`footnoteinside` default=`true`

The position of the footnotes can be changed with the option `footnoteinside`. The footnotes will be displayed at the end of the environment but you can decide whether the output is inside `mdframed` or after.

Note

The output of the footnotes with the option `footnoteinside=false` are not in a splitted frame. I think it isn't useful because the first line of a new page shouldn't be a footnote.

6. Examples

I outsource the examples in four files to limit the documentation. The files are

mdframed-example-default
Demonstration of examples created with `framemethod=default`.

mdframed-example-tikz
Demonstration of examples created with `framemethod=TikZ`.

mdframed-example-pstricks
Demonstration of examples created with `framemethod=pstricks`.

mdframed-example-texsx
Demonstration of examples like interaction with `listings`

The examples are often not equivalent but normally they can be adapted to another method. So I really recommend to have a look to all example files.

The Korean TEXGroup created a very nice presentation. I want to show the link because it's really a great work: kts 2012 mdframed.

7. Errors, Warnings and Messages

The package `mdframed` provides different errors, warnings and messages in the `log`-file. Some LATEX-editors like TEXMaker or TEXStudio have a special tab for errors and warnings but not for messages. So you should look in the `log-File` itself.

The following errors and warnings are generated by `mdframed`.

> The package ... does not exist but
> needed by **mdframed**

To avoid this problem you should install the required packages which are listed in section 2.

> package option **style** is depreciated
> use **framemethod** instead **style**

With version 0.9d `mdframed` changed the meaning of the option `style`. The option is used to load a defined style by `\mdfdefinestyle`. Instead use `framemethod` (see section 5.1).

> Unknown **framemethod** .... **mdframed**

The input string for the option `framemethod` is unkown. See section 5.1.

> You have not loaded **ntheorem** yet

To use the option `ntheorem` you have to load the package `ntheorem`.

> You have only a width of 3cm

The package `mdframed` calculates the width of the contents based on the given options. If the width of the contents is smaller than 3 cm you will get this warnings. You should change the settings to get a greater width.

> You got a bad break
> you have to change it manually
> by changing the **text**, the space
> or something else

Sometimes you have enough vertical space for the rules and the space between the rules and the contents but not for the contents itself. In this situation you will get this warning because the contents of this box is empty. You have the possibility to change the settings or include a `\clearpage` in front of the environment `mdframed`. So far I have no idea how to avoid such things.

> You got a bad break
> because the split box is empty
> You have to change the page **settings**
> like enlargethispage or something else
> You got a bad break

See the explanation above.

> You got a bad break
> because the last split box is empty
> You have to change the **settings**

The same reason as above but only in the last box.

> Option ... is already consumed
> and has no effect on input line ...

If you set a global option inside the document body you will get this warning.

8. Known Problems

In this section I will collect known problems. In case you encounter any further problems, please drop me an email, marco.daniel at mada-nada.de.

Do you have any ideas / wishes on further extensions to this package? Please let me know!

1. So far the environment isn't compatible with the package `gmverb`.
2. If you load the package `picins` the frame will no be splitted. That based on a problem of the package 'picins' which defines `\@captype` global. To work with the package picins you can use the following hack.

```
\usepackage{picins}
\makeatletter
\let\@captype\@undefined
\def\newcaption{%
  \begingroup%
    \def\@captype{figure}%
    \refstepcounter\@captype\@dblarg{\@newcaption\@captype}%
  \endgroup%
```

```
}
\makeatother
```

3. `mdframed` can't handle the option `allowframebreaks` of the class `beamer`.
4. A nested `mdframed` environment can't be splitted.

9. ToDo

It is important to update the documentation

1. see "Known Problems".
2. So far it isn't possible to combine the environment `\begin{multicols}` of the package `multicol` with `mdframed` with the whole option list.
3. Create new styles.
4. Improve page breaks.
5. Improve footnotes.
6. Improve documentation and examples.
7. Create styles for `frametitle`.
8. Create an inline version of `mdframed` that's works like `\fbox`
9. Add `\ht\strutbox` to file `md-frame-1.mdf`

10. Acknowledgements

Dick Nickalls; Dietrich Grau; Piazza Luca; Jobst Hoffmann; Martin Scharrer; Enrico Gregorio; Heiko Oberdiek; Philipp Stephani.

Thanks for proofreading
Alan Munn and Nahid Shajari
I hope I forgot nobody.

A. More information

In the following section I want to present how to create your own frame.

A.1. How does mdframed work?

With the environment \begin{mdframed} ... \end{mdframed} the whole contents will be saved in a \savebox called \mdf@splitbox@one. After the calculation of the width and the height of the \mdf@splitbox@one (done by mdframed.sty) the box will be set sequently (done by md-frame-X.mdf). The following figure demonstrates this.

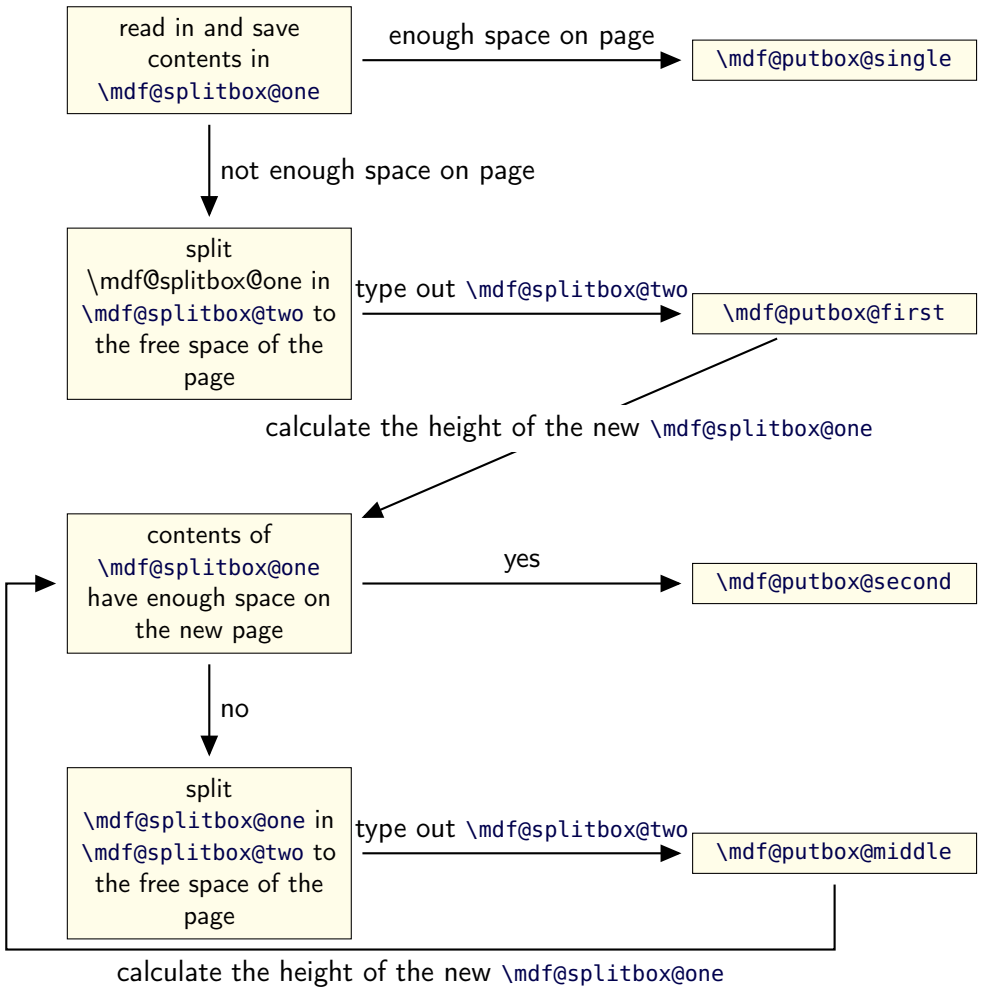

Figure 4: Setting the contents of mdframed

The width of the contents is the result of the settings of leftmargin, rightmargin, linewidth, innerleftmargin and innerrightmargin (see figure (2)).

A.2. The Framecommands

The package mdframed knows four kinds of "Framecommand". These commands tell LaTeX how to set the contents of mdframed.

\mdf@putbox@single This command sets the contents of a single unsplit frame.

`\mdf@putbox@first` This command sets the contents of the first frame of a split frame.

`\mdf@putbox@middle` This command sets the contents of the middle frame of a split frame.

`\mdf@putbox@second` This command sets the contents of the last frame of a split frame.

Using the explained commands we give an example. The command `\box` uses the contents of the savebox and types them out.

First we want to type out the single box without any settings (but with the calculated width).

```
\makeatletter
  \def\mdf@putbox@single{\box\mdf@splitbox@one}
\makeatother
```

I am using the command `\leftline` to start the "Framecommands" at the left.

```
\makeatletter
  \def\mdf@putbox@single{\leftline{\box\mdf@splitbox@one}}
\makeatother
```

Now you have to know how the lengths are named. Every length which can be modified by the options has the following syntax:

```
\mdf@<Name of the Length>@length
```

For example the leftmargin is:

```
\mdf@leftmargin@length
```

To create only a line at the left with the correct `leftmargin` you can set `\mdf@putboxsingle` as follows

```
\makeatletter
\def\mdf@putbox@single{%
                \leftline{%
                        \hspace*{\mdf@leftmargin@length}%
                        \rule[-\dp\mdf@splitbox@one]{\mdf@linewidth}%
                                {\ht\mdf@splitbox@one+\dp\mdf@splitbox@one}%
                        \box\mdf@splitbox@one
                }%
}
\makeatother
```

In this way you can do what you want. If you create your own style you can save the file as `md-frame-X.mdf`. `X` must be an integer. In this way you can use the option `framemethod` to load the file by setting `framemethod=X`.

A.3. Revision history

Version 1.8X submitted XX Mar 2013

• added `\topskip=0pt` to remove unwanted space after `\newpage` or `\clearpage`

Version 1.8 submitted 09 Mar 2013

• fixed bugs (e.g. related to `\parskip` of KOMA) • allow margin notes with `framemethod=tikz` • fixed some typos

Version 1.6d submitted 21 Sep 2012

• fixed bugs • added option `draft`

Version 1.6b submitted 02 Jun 2012

• added commands `\mdfsubtitle` and `\mdfsubsubtitle` • added options `subtitleaboveline`, `subtitlebelowline`, `subsubtitleaboveline`, `subsubtitlebelowline`, `subtitlefont`, `subsubtitlefont`, `subtitlebackgroundcolor`, `subsubtitlebackgroundcolor`, `subtitleabovelinecolor`, `subtitlebelowlinecolor`, `subsubtitleabovelinecolor`, `subsubtitlebelowlinecolor`, `subtitleabovelinewidth`, `subtitlebelowlinewidth`, `subtitleaboveskip`, `subtitlebelowskip`, `subtitleinneraboveskip`, `subtitleinnerbelowskip`, `subsubtitleabovelinewidth`, `subsubtitlebelowlinewidth`, `subsubtitleaboveskip`, `subsubtitlebelowskip`, `subsubtitleinneraboveskip`, `subsubtitleinnerbelowskip` • improved formating of the file `mdframed.dtx` • fixed bug in combination with `\parskip` – Thanks David Carlisle. • added extra loop to compute the splitting point. • improved splitting algorithm • added new option `ignorelastdescenders` – Thanks Stephan Lehmke. • Improved option `repeatframetitle` • fixed bug: `framemethod=tikz` used wrong computed length by setting `everyline=true` • Tobias Weh inspired the excurs-environment not Tobias Schwan. Sorry, I fixed it. • Improved `\mdtheorem` to handle `\listtheorems` provided by `ntheorem`.

Version 1.5 submitted 10 Mar 2012

• fixed bug (Thanks Nicolas Roy) • expanded documentation (Thanks Martin Wilhelm Leidig) • added options `singleextra`, `firstextra`, `middleextra` and `secondextra` • expanded examples

Version 1.4d submitted 30 Mar 2012

• fixed bug (Thanks Nicolas Roy) • added approach to documentation to work with `picins` • new implementation of option `hidealllines`, now you can set `\mdfsetup{hidealllines=true,leftline=true}` printing only the left line (inspired by Tobias Weh) • added option `everyline` to draw a top and bottom line at splitted frames

Version 1.4 submitted 4 Mar 2012

• fixed bug in combination with `\marginpar` (Thanks Juan Carlos Trujillo Ortega) • fixed bug with option `font` • fixed bug inside `frametitle` (Thanks Yi, Hoze) • removed unnessary groups (Thanks Yi, Hoze) • changed the definition of `listings` to allow copy paste of the examples

Version 1.3a submitted 5 Feb 2012

• fixed bug (Thanks to Dietrich Grau)

Version 1.3 submitted 4 Feb 2012

• fixed documentation (Thanks to Dietrich Grau) • added option `shadow` • improved handling `\parindent` and `\parskip` (Thanks to Enrico Gregorio and Joseph Wright)

Version 1.2 submitted 8 Jan 2012

• fixed documentation (Thanks to Dietrich Grau) • fixed bug in combination with `amsthm` • fixed bug in `\newmdtheoremenv` • defined new styles via `\newpsstyle`
This works only with `framemethod=PSTricks`. • added new commands for interaction with `TikZ` and `PSTricks` • expand frame title option by option `frametitlerule`, `frametitlerulewidth`, `frametitlefont`, `frametitleaboveskip`, `frametitlebelowskip`, `frametitlealignment` • removed limitation of three lines for PSTricks • defined new commands `\surroundwithmdframed`, `\mdflength`, `\mdtheorem` • load `xparse` by default • changed internal names • expanded examples

Version 1.0b submitted 9 Dec 2011

• fixes documentation (Thanks to Dietrich Grau) • fixes bug in `\newmdtheoremenv` • fixes bug with overfull boxes (Thanks to Dietrich Grau) • defined `\newpsstylemdfbackgroundstyle` and `mdflinestyle`
This works only with `framemethod=PSTricks`. • created dtx-file (Thanks to Kevin Godby) • added `\@parboxrestore` to `\mdf@lrbox`

Version 1.0 submitted 13 Nov 2011
• add option `userdefinedwidth` • add option `align` • add option `apptotikzsetting` • create new command `\mdfapptodefinestyle` • changed internal algorithm • removed calc instead using ε-TEX `\dimexpr` • expand documentation • trying to fixe problems with `xcolor` • fixed bug with `framemethod=pstricks` • create file `mdframed-example-default` • create file `mdframed-example-tikz` • create file `mdframed-example-pstricks` • create file `mdframed-example-texsx` (texsx stands for tex stackexchange)

Version 0.9g submitted 08 Oct 2011
• fixed documentation • added small footnote compatibility

Version 0.9f submitted 04 Oct 2011
• fixes bugs (thanks to Lars Madsen) • added option `hidealllines` • fixed documentation

Version 0.9e submitted 11 Sep 2011
• working with `twoside` modus

Version 0.9d submitted 10 Sep 2011
• changed the meaning of the option `style`!!! (inspired by Lars Madsen) • added option `framemethod` (inspired by Lars Madsen) • added options `needspace` (inspired by Lars Madsen) • added new command `\mdfdefinestyle` (inspired by Lars Madsen) • fixes documentation • renamed `md-frame-3.mdf` to `md-frame-2.mdf`

Version 0.9b submitted 7 Sep 2011
• fixes bugs in `\newmdtheoremenv` (Thanks to Enrico Gregorio)

Version 0.9a submitted 5 Sep 2011
• fixes bugs (Thanks to Lars Madson) • expanded documentation (added revision history)

Version 0.9 submitted 4 Sep 2011
• added option `nobreak` • detecting float environments to prevent split calculation • expand documentation (Thanks to Alan Munn)

Version 0.8a
• fixes bugs • fixes documentation

Version 0.8 submitted 22 Aug 2011
• added commands: `\newmdenv`, `\renewmdenv`, `\newmdtheoremenv` • fixes bugs • fixes documentation

Version 0.7a submitted 6 August 2011
• added option `frametitle` • added option `frametitlefont` • allow twolumn-mode • changed the calculation • added option `tikzsetting` • added options for hidden lines for all styles • fixes bugs

Version 0.6a submitted 22 Dec 2010
• fixes bugs • added `\mdfsetup` • expanded documentation

B. Implementation

And finally, here's how it all works...

C. Index

The index only collect package relevant words.

P

R

S

T

U

X