

The mdframed package¹

auto-split frame environment

Marco Daniel², Elke Schubert v0.9e September 11, 2011

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 page-breaks in framed text.

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

Contents

1. Motivation	1	6. Errors, Warnings and Messages	10
2. Syntax	2	7. Known Problems	11
3. Commands	2	8. ToDo	11
4. Options	3	9. Previous versions	11
4.1. Global Options	4	10. Acknowledgements	12
4.2. Global and Local Options	4	A. More information	13
4.3. Hidden Lines	6	A.1. How does <code>mdframed</code> work?	13
4.4. Frametitle	6	A.2. The Frametitle commands .	13
5. Examples	7	A.3. Modify the frame	15
5.1. Working in twocolumn mode	9	A.4. <code>mdframed</code> in combination with <code>multicol</code> . . .	16
5.2. Framed listings	10	A.5. Revision history	17

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.

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*

¹Extending the package `framed.sty`.

²With thanks to Heiko Oberdiek, Rolf Niepraschk, Martin Scharrer and Herbert Voss. Sorry for bad English.

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=0pt,%
  splittopskip=\topskip,skipbelow=\baselineskip,%
  skipabove=\baselineskip,ntheorem]{theorem}%
  {Theorem}[section]
\begin{theorem}[Pythagorean theorem]
...
\end{theorem}
```

2. Syntax

The package itself loads the packages kvoptions, etoolbox, calc and color. By setting the correct options mdframed will load xcolor, tikz or pstricks.

Load the package as usual:

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

The package defines only one environment with the following syntax:

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

With version 0.8 you have two commands to create environments with mdframed.

autodetecting floats With version 0.9 I added a detection of float or minipage environments. If you use mdframed in such an environment mdframed will use the option **nobreak** automatically.

twoside-mode With version 0.9d I added the detection of the twoside. When you are using mdframed inside twoside mode you have to set the option **innermargin** and **outermargin** (see section 4.2.1).

3. Commands

The following commands should countenance your by the handling with mdframed

\newmdenv The command has the following syntax:

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

In this way you can simply use:

```

\newmdenv[ linecolor=red , frametitle=Infobox ]{ infobox }
...
\begin{infobox}[ backgroundcolor=yellow ]
foo   foo   foo   foo   foo   foo
\end{infobox}

```

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

\newmdtheoremenv Since the package is often used to highlight theorem environments, I have created a command³ 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`!

\mdfsetup To set the options globally you can use the optional argument of `\usepackage` or you can use the command `\mdfsetup` which is not limited to the preamble.

\mdfdefinestyle `\mdfdefinestyle` allow the user to define different styles and use as an option of `mdframed` via **style**. The option **style** is explained in section 4.4.1. Here a small example:

```

\mdfdefinestyle{mystyle}{ leftmargin=0pt ,%
                           linecolor=blue }
...
\begin{mdframed}[ style=mystyle ]
foo
\end{mdframed}

```

4. Options

The package allows the following global and local options:

³Thanks to Martin Scharer and Enrico Gregorio:
[Own command to create new environment](#)

4.1. Global Options

The following options are only global options.

- xcolor** By setting this key, the package `xcolor` will be loaded with the given value(s). default=none
Without any value `mdframed` loads the package `color` without any options.
- framemethod** With this key you can change the way frames are drawn. You can decide default=default
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 combination are listed in the following table.

Table 1: Allowed keys for **framemethod**

Method	Allowed keys
\LaTeX -commands	default, tex, latex, none, 0
<code>TikZ</code>	tikz, 1
<code>PSTricks</code>	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.

4.2. Global and Local Options

The options listed below can be set globally or locally.

4.2.1. Options with lengths

In figure (1) you can see the adjustable lengths 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 **defaultunit**.

- defaultunit** see the sentence above. default=pt

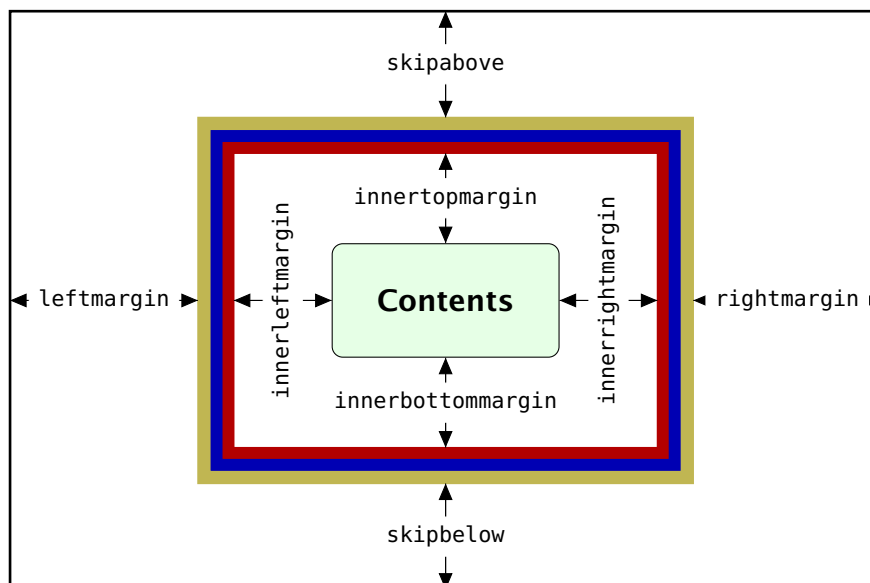


Figure 1: adjustable lengths of mdframed

skipabove Sets an additional skip above the frame. default=0pt

skipbelow Sets an additional skip below the frame. default=0pt

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

leftmargin Sets the length of the left margin of the environment. default=0pt

rightmargin Sets the length of the right margin of the environment. default=0pt

innerleftmargin Sets the length of the inner left margin of the environment. default=10pt

innerrightmargin Sets the length of the inner right margin of the environment. default=10pt

innertopmargin Sets the length of the inner top margin of the environment. default=.4\baselineskip

innerbottommargin Sets the length of the inner bottom margin of the environment. default=.4\baselineskip

The following lengths are not shown in figure (1).

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

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

splittopskip Sets the length of the skip above the split part of the environment. default=0pt

splitbottomskip Sets the length of the skip below the split part of the environment. default=0pt

linewidth	Sets the width of the line around the environment.	default=0.4pt
roundcorner	Sets the size of the radius of the corners of the frames. This works only with <code>framemethod=TikZ</code> and <code>framemethod=PSTricks</code>.	default=0pt
innerlinewidth	Sets the width of the inner line around the environment. This works only with <code>framemethod=TikZ</code>.	default=0pt
outerlinewidth	Sets the width of the outer line around the environment. This works only with <code>framemethod=TikZ</code>.	default=0pt
middlelinewidth	Sets the width of the middle line around the environment. This works only with <code>framemethod=TikZ</code>.	default=linewidth

4.2.2. Colored Options

linecolor	Sets the color of the line around the environment.	default=black
innerlinecolor	Sets the color of the inner line around the environment. This works only with <code>framemethod=TikZ</code>.	default=linecolor
middlelinecolor	Sets the color of the middle line around the environment. This works only with <code>framemethod=TikZ</code>.	default=linecolor
outerlinecolor	Sets the color of the outer line around the environment. This works only with <code>framemethod=TikZ</code>.	default=linecolor
backgroundcolor	Sets the color of the background of the environment.	default=white
fontcolor	Sets the color of the contents of the environment.	default=black

4.3. Hidden Lines

topline	Draws a line at the top.	default=true
bottomline	Draws a line at the bottom.	default=true
leftline	Draws a line on the left.	default=true
rightline	Draws a line on the right.	default=true

4.4. Frametitle

frametitle	The environment get a title. To set a title use <code>frametitle={The Title of the frame}</code> as an option of the environment.	default=none
frametitleformat	Sets the format of the frame title.	default= <code>\bfseries</code> <code>\large</code>

4.4.1. General options

ntheorem	Before setting this boolean key, you have to load the package <code>ntheorem</code> . With this option you set the values <code>\theorempreskipamount</code> and <code>\theorempostskipamount</code> to 0pt.	default=false
nobreak	Sometimes it is useful to prevent a frame from splitting. The nobreak option is used for this purpose.	default=false
needspace	Sometimes it is useful to set a minimum height before a frame should be splitted. For such cases you can use needspace .	default=0pt
style	If you define a special style with <code>\mdfdefinestyle</code> you can use the key style to load the style. <code>mdframed</code> has no predefined styles.	
pstrickssetting	With this key you can pass several options to <code>\psset</code> . 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=PS Tricks.	default=none
tikzsetting	With this key you can pass several options to <code>\tikzset</code> . 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.	default=none

5. Examples

Now I present some examples. Of course I know that some of them are only gimmicks. To create the following examples I am using **framemethod=TikZ**.

```
\mdfsetup{outerlinewidth=5pt,innerlinewidth=0pt,
           outerlinecolor=red,roundcorner=5pt}
```

Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text

Now I will add the next options

```
\mdfsetup{topline=false,leftline=false,}
```

Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text Some Text

Gimmick:

```
\mdfsetup{splitbottomskip=0.8cm, splittopskip=0cm,
  innerrightmargin=2cm, innertopmargin=1cm,%
  innerlinewidth=2pt, outerlinewidth=2pt,
  middlelinewidth=10pt, backgroundcolor=red,
  linecolor=blue, middlelinecolor=gray,
  tikzsetting={draw=yellow, line width=3pt,%
    dashed,%
    dash pattern= on 10pt off 3pt},
  rightline=false, bottomline=false}
```

Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text
 Some Text Some Text Some Text Some Text Some Text

5.1. Working in twocolumn mode

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac,

nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Some Text Some Text Some Text
Some Text Some Text Some Text

5.2. Framed listings

The example below is inspired by the following post on StackExchange [Background overflows when using rounded corners for listings \(package: 'listings'\)](#)

Here the solution which can be decorate as usual.

```
\BeforeBeginEnvironment{\lstlisting}{\%
\begin{mdframed}[<modification>]\%
\vspace{-0.7em}}
\AfterEndEnvironment{\lstlisting}{\%
\vspace{-0.5em}\%
\end{mdframed}}
```

6. Errors, Warnings and Messages

The package `mdframed` provides different errors, warnings and messages in the log-file. Some \LaTeX -editors like \TeX Maker(X) have a special tab for errors and warnings but not for messages. So you should look in the log-File itself.

The followings 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 mdframed
```

With version 0.9d `mdframed` changed the meaning of the option `style`. The option is used to load a defined style by `\mdfdefinestyle`. Instead uses `framemethod` (see section 4.1).

```
Unknown framemethod .... mdframed
```

The input string for the option `framemethod` is unknown. See section 4.1.

```
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 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.

7. 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](mailto:marco.daniel@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. So far it isn't possible to combine the environment `\begin{multicols}` of the package `multicol` with `mdframed`.
3. So far it isn't possible to use footnotes within a frame.

8. ToDo

1. see „Known Problems“.
2. Create new styles.
3. Improve page breaks.

9. Previous versions

If you have trouble with the new version of `mdframed` you can load `mdframedpre` instead. In this way you load the version v0.6 (see [Revision history](#)).

10. Acknowledgements

Dick Nickalls; Dr. Dietrich Grau; Piazza Luca Jobst Hoffmann.

Thanks for proofreading

Alan Munn and Nahid Shajari

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 `\@tempboxa`. After the calculation of the width and the height of the `\@tempboxa` (done by `mdframed.sty`) the box will be set sequentially (done by `md-frame-X.mdf`). The following figure demonstrates this.

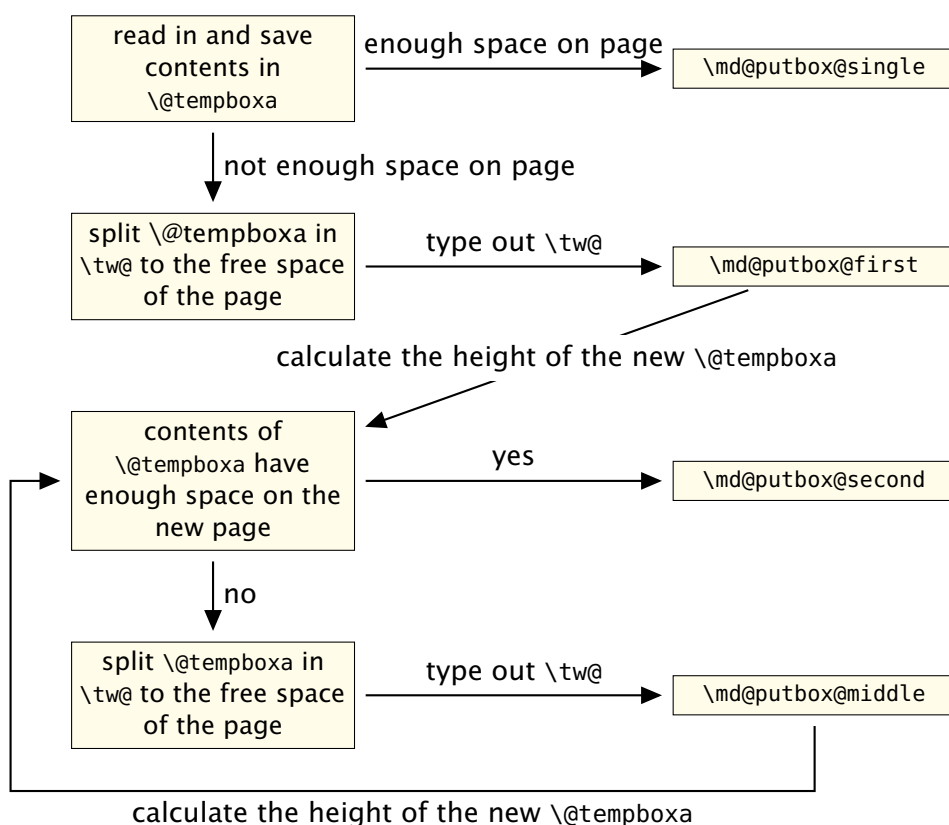


Figure 2: Setting the contents of mdframed

The width of the contents is the result of the settings of **leftmargin**, **rightmargin**, **linewidth**, **innerleftmargin** and **innerrightmargin** (see figure (1)).

A.2. The Framecommands

The package `mdframed` knows four kinds of „Framecommand“. These commands tell \LaTeX how to set the contents of `mdframed`.

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

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

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

\md@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\md@putbox@single{\box\@tempboxa}
\makeatother
```

I am using the command **\leftline** to start the „Framcommands“ at the left.

```
\makeatletter
\def\md@putbox@single{\leftline{\box\@tempboxa}}
\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 **\md@putboxsingle** as follows

```
\makeatletter
\def\md@putbox@single{%
  \leftline{%
    \hspace*{\mdf@leftmargin@length}%
    \rule[-\dp\@tempboxa]{\mdf@linewidth}%
      {\ht\@tempboxa+\dp\@tempboxa}%
    \box\@tempboxa
  }%
}
\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 **style** to load the file by setting **style=X**.

A.3. Modify the frame

The example below is inspired by the following post on StackExchange [Theorem decorations that stay with theorem environment](#)

First I will show the the result.

Lemma A.1 *Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.*

You can see how the lines on the left and the right are interrupted. Now I will show the code.

I define a command which draws white lines on the left and the right side in the same way as in the style file. For this example I must include the file md-frame-1.mdf because this document used the global style 1.

With the new length `\interruptlength` you can manipulate the amount of overlap.

```
\newlength{\interruptlength}
\setlength{\interruptlength}{1.5 ex}
\makeatletter
\newlength{\interruptlength}
\setlength{\interruptlength}{2.5 ex}
\newrobustcmd\overlaplines{%
  \input{md-frame-0.mdf}%
  \global\defcounter{mdf@globalstyle@cnt}{0}
  \appto\md@frame@leftline@single{%
    \llap{\color{white}%
      \rule[-\dp\@tempboxa
        -\mdf@innerbottommargin@length%
        -\mdf@middlelinewidth@length
        +\interruptlength]{%
          {\mdf@middlelinewidth@length}%
          {\mdf@boundingboxheight%
            +2\mdf@middlelinewidth@length
            -2\interruptlength}%
        }%
    }%
  }%
  \appto\md@frame@rightline@single{%
    \rlap{\color{white}%
      \hspace*{\mdf@boundingboxwidth}%
      \hspace*{\mdf@innerrightmargin@length}%
    }%
  }%
```

```

\rule[-\dp\@tempboxa
      -\mdf@innerbottommargin@length%
      -\mdf@middlelinewidth@length
      +\interruptlength]{%
      {\mdf@middlelinewidth@length}
      {\mdf@boundingboxheight%
       +2\mdf@middlelinewidth@length%
       -2\interruptlength}{%
      }%
    }%
  }
\overlapiines
\newmdtheoremenv[style=0,ntheorem,linewidth=10pt]{\lem}{\section}
\makeatother

```

A.4. *mdframed* in combination with *multicol*

How I wrote in „Known Problems“ you can't combine *multicol* with *mdframed*. In a simple way without any breaks you can use:

```

\begin{multicols}{2}
\lipsum[1]
\begin{mdframed}
Text\par Text\par Text
\end{mdframed}
\lipsum[2]
\end{multicols}

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis,

diam. Duis eget orci sit amet orci dignissim rutrum.

Text
Text
Text

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

A.5. Revision history

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 `twocolumn`-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

Version 0.6 submitted 18 Dec 2010

- added **`style=3`** with `pstricks`
- added option **`pstrickssetting`**
- added option **`splitbottomskip`**
- added option **`splittopskip`**
- added options for hidden lines
- changed the calculation
- fixes bugs

Version 0.4a submitted 14 May 2010

- fixes bug in `fontcolor`

Version 0.4 submitted 13 May 2010

- Elke Schubert creates style file for `tikz`
- fixes some bugs – calculation of the page dimen (thanks Dick Nickalls)
- using `tikz` for the frame with different styles

Version 0.3b submitted 1 May 2010

- fixes some bugs – thanks to Dr. Dietrich Grau
- added new options: **`ntheorem`**

Version 0.3a submitted 23 Apr 2010

- added new options: **`leftmargin`** and **`rightmargin`**
- fixes some bugs

Version 0.3 submitted 16 Apr 2010

- first upload to [CTAN](#)