tcolorbox

Manual for version 3.71 (2015/07/23)

Thomas F. Sturm

Cover code

```
% \usepackage{incgraph}
\begin{inctext}
\begin{tikzpicture}
\definecolorseries{boxcol}{rgb}{last}{blue}{red}
\resetcolorseries[28]{boxcol}
\coordinate (A) at (0,0); \coordinate (B) at (21,29.7);
\path[use as bounding box] (A) rectangle coordinate (C) (B);
\node[transform shape,xslant=0.7,rotate=-10,xshift=0cm] at (C) {%
  \begin{tcbraster}[raster columns=4,title=tcolorbox \version,
    fonttitle=\small\bfseries,raster width=50cm]
  \foreach \b in {1,...,28} {\begin{tcolorbox}[enhanced,
    watermark text=\thetcbrasternum,
    colframe=boxcol!30!white,
    colback=boxcol!25!white!30!white,
    colbacktitle=boxcol!!+!50!black!30!white,
    colupper=black!30!white]\lipsum[2]\end{tcolorbox}}
  \end{tcbraster}%
};
\node at (C) {%
  \begin{tcbitemize}[title=tcolorbox \version,fonttitle=\small\bfseries,
    enhanced jigsaw,opacityback=0.5,opacitybacktitle=0.75,
    halign=center,valign=center,arc=5mm,
    raster width=16cm,raster column skip=8mm,raster halign=center,
    raster force size=false,
    raster row 1/.style={height=6cm},
    raster row 2/.style={width=6cm,height=4cm},
    raster column 1/.style={flushright title,
      frame style={left color=yellow!50!black,right color=green!50!black},
      title style={left color=yellow!50!blue,right color=blue!50!green!50!black},
      interior style={left color=yellow!70,right color=green!70},
      underlay={\draw[line width=6mm,line cap=round,black!60]
        ([shift={(0.4,-0.15)}]frame.north east)
        --([shift={(0.4,0.15)}]frame.south east); }},
    raster column 2/.style={
      frame style={left color=green!50!black,right color=yellow!50!black},
      title style={left color=blue!50!green!50!black,right color=yellow!50!blue},
      interior style={left color=green!70,right color=yellow!70}}]
  \tcbitem[fontupper=\Huge\bfseries,sharp corners=east,
    underlay={\draw[line width=6mm,line cap=round,black!60]
      ([shift={(0.4,0.30)}]frame.north east)-- coordinate(A) +(0,0.2);
      \draw[line width=1mm,line cap=round,black!60](A) -- +(30:1.5cm);
      \draw[line width=1mm,line cap=round,black!60](A) -- +(150:1.5cm);}]
    tcolorbox
  \tcbitem[fontupper=\large\bfseries,sharp corners=west]
    Manual for\\ version\\ \version\\(\datum)
  \tcbitem[sharp corners=northeast]
  \tcbitem[sharp corners=northwest] Thomas F.~Sturm
  \end{tcbitemize}%
};
\end{tikzpicture}
\end{inctext}
```

The tcolorbox package

Manual for version 3.71 (2015/07/23)

Thomas F. Sturm[1]

Abstract

tcolorbox provides an environment for colored and framed text boxes with a heading line. Optionally, such a box can be split in an upper and a lower part. The package tcolorbox can be used for the setting of LaTeX examples where one part of the box displays the source code and the other part shows the output. Another common use case is the setting of theorems. The package supports saving and reuse of source code and text parts.

Contents

1 Introduction **7**
- 1.1 Installation . . . 7
- 1.2 Loading the Package . . . 7
- 1.3 Libraries . . . 8

2 Quick Reference **10**

3 Macros for Box Creation **11**

4 Option Keys **17**
- 4.1 Title . . . 17
- 4.2 Subtitle . . . 20
- 4.3 Upper Part . . . 21
- 4.4 Lower Part . . . 23
- 4.5 Colors and Fonts . . . 26
- 4.6 Text Alignment . . . 29
- 4.7 Geometry . . . 33
  - 4.7.1 Width . . . 33
  - 4.7.2 Rules . . . 34
  - 4.7.3 Arcs . . . 35
  - 4.7.4 Spacing . . . 37
  - 4.7.5 Size Shortcuts . . . 42
- 4.8 Corners . . . 46
- 4.9 Transparency . . . 49
- 4.10 Height Control . . . 51
- 4.11 Box Content Additions . . . 58
- 4.12 Overlays . . . 64
- 4.13 Floating Objects . . . 69
- 4.14 Side by Side . . . 71
- 4.15 Embedding into the Surroundings . . . 76
- 4.16 Bounding Box . . . 80

[1]Prof. Dr. Dr. Thomas F. Sturm, Institut für Mathematik und Informatik, Universität der Bundeswehr München, D-85577 Neubiberg, Germany; email: thomas.sturm@unibw.de

4.17 Layered Boxes and Every Box Settings . . . 86
4.18 Capture Mode . . . 89
4.19 Text Characteristics . . . 90
4.20 Files . . . 91
4.21 \tcbox Specials . . . 91
4.22 Counters, Labels, and References . . . 93
4.23 Externalization . . . 95
4.24 Miscellaneous . . . 95

5 Initialization Option Keys 97
5.1 Numbered Boxes . . . 97
5.2 Lists of tcolorboxes . . . 104

6 Saving and Loading of Verbatim Texts 105

7 Recording 106
7.1 Makros . . . 106
7.2 Options . . . 106
7.3 Example: Exercises . . . 107
7.4 Example: Solutions . . . 110

8 Technical Overview and Customization 112
8.1 Skins and Drawing Engines . . . 112
8.2 Code Option Keys . . . 116
8.3 Subskins . . . 119
8.4 Drawing Scheme . . . 120

9 Library LIB skins 124
9.1 Style Option Keys . . . 124
9.2 Boxed Title Option Keys . . . 131
9.3 Watermark Option Keys . . . 139
9.4 Clip Environments . . . 146
9.5 Border Line Option Keys . . . 151
9.6 Shadow Option Keys . . . 156
9.6.1 Common Shadows and Halos . . . 156
9.6.2 Lifted Shadows . . . 161
9.6.3 Generic Shadows . . . 162
9.6.4 TikZ Shadows . . . 165
9.7 TikZ Picture Option Keys . . . 166
9.8 Underlay Option Keys . . . 169
9.9 Finish Option Keys . . . 171
9.10 Jigsaw Skin Variants . . . 173
9.11 Draft Mode . . . 175
9.12 Skin Family 'standard' . . . 176
9.13 Skin Family 'enhanced' . . . 178
9.14 Skin Family 'bicolor' . . . 191
9.15 Skin Family 'beamer' . . . 196
9.16 Skin Family 'widget' . . . 201
9.17 Skin Family 'empty' . . . 205
9.18 Skin 'spartan' . . . 214
9.19 Skin 'draft' . . . 215
9.20 Skin Family 'freelance' . . . 217

10 Inclusion of Boxed Image Files **218**
10.1 Macros . . . 218
10.2 Option Keys . . . 221

11 TikZ Image and Picture Fill Extensions; Auxiliary Macros **222**
11.1 Fill Plain . . . 222
11.2 Fill Stretch . . . 223
11.3 Fill Overzoom . . . 224
11.4 Fill Zoom . . . 225
11.5 Fill Shrink . . . 226
11.6 Fill Tile . . . 227
11.7 Filling Options . . . 228
11.8 Straightening of the Arcs . . . 229
11.9 Extracting Node Dimensions . . . 230

12 Library LIB raster **231**
12.1 Concept of Rasters . . . 231
12.2 Macros of the Library . . . 233
12.3 Option Keys of the Library . . . 235
12.4 Adding Styles for Specific Boxes . . . 240

13 Libraries LIB listings, LIB listingsutf8, and LIB minted **242**
13.1 Loading the Libraries . . . 242
13.1.1 Loading LIB listings . . . 242
13.1.2 Loading LIB listingsutf8 . . . 242
13.1.3 Loading LIB minted . . . 243
13.2 Common Macros of the Libraries . . . 243
13.3 Option Keys of the LIB listings Library . . . 249
13.4 Option Keys of the LIB listingsutf8 Library . . . 251
13.5 Option Keys of the LIB minted Library . . . 252
13.6 Common Option Keys of all Libraries . . . 254
13.7 Option Keys for Processing and Full Document Examples . . . 262
13.8 Creation of LaTeX Tutorials . . . 268
13.9 Creation of LaTeX Exercises . . . 275
13.10 List of Exercises . . . 278
13.11 Solutions for the given LaTeX Exercises . . . 279

14 Library LIB theorems **281**
14.1 Macros of the Library . . . 281
14.2 Option Keys of the Library . . . 285
14.3 Examples for Definitions and Theorems . . . 298
14.4 Using other theorem environments with `tcolorbox` . . . 303

15 Library LIB breakable **304**
15.1 Technical Overview . . . 304
15.2 Limitations and Known Bugs . . . 305
15.3 Main Option Keys . . . 306
15.4 Option Keys for the Break Appearance . . . 310
15.5 Extra Options for Partial Boxes . . . 312
15.6 Breakable boxes and the `multicol` package . . . 314
15.7 Break Sequence for the Skins . . . 317

15.8 Break by Hand (Faked Break) . . . 325

16 Library magazine **326**
16.1 Creation and Resetting of Box Arrays . . . 326
16.2 Storing Content . . . 327
16.3 Retrieving Content . . . 329
16.4 Box Dimensions . . . 332

17 Library fitting **334**
17.1 Macros of the Library . . . 334
17.2 Option Keys of the Library . . . 336

18 Library hooks **345**
18.1 Concept of Hooks . . . 345
18.2 Box Content Additions . . . 346
18.3 Embedding into the Surroundings . . . 347
18.4 Overlays . . . 348
18.5 Watermarks . . . 350
18.6 Underlays . . . 352
18.7 Finishes . . . 353
18.8 Skin Code . . . 353
18.9 Extras . . . 355

19 Library xparse **356**
19.1 Option Keys . . . 356
19.2 Producing `tcolorbox` Environments and Commands . . . 358
19.3 Producing `tcbox` Commands . . . 361
19.4 Producing `tcblisting` Environments . . . 364
19.5 Producing `tcbinputlisting` Commands . . . 366
19.6 Producing `tboxfit` Commands . . . 367

20 Library external **369**
20.1 Preparation of a Document for Externalization . . . 370
20.2 Marking Externalization Snippets . . . 371
20.3 Customization . . . 376
20.4 Troubleshooting and FAQ . . . 379

21 Library documentation **380**
21.1 Macros of the Library . . . 380
21.2 Option Keys of the Library . . . 390
21.3 Predefined Colors of the Library . . . 396

A Picture Credits **397**

References **398**

Index **400**

1 Introduction

The package originates from the first edition of my book «*LaTeX – Einführung in das Textsatzsystem*» [19] in about 2006. For the LaTeX examples and tutorials given there, I wanted to have accentuated and colored boxes to display source code and compiled text in combination. Since, in my opinion, this type of boxes is also quite useful to highlight definitions and theorems, I applied them for my lecture notes in mathematics [16–18] as well. With this package, you are invited to apply these boxes for similar projects.

Starting with version 2.00, for all internal calculations ε-TeX [2] expressions are used in replacement of the package `calc` [21]. The breaking news for version 2.00 is the support for breakable boxes. This new feature allows new applications of the package without affecting the core package too much if you do not need boxes to break automatically. With version 2.20, the often requested 'side by side' mode for listings has been added. With version 3.00, boxed titles are introduced together with improved customization options for overlays, underlays, finishes, and own code extensions.

Since the first public release in 2011, I received a lot of feedback from all over the world. I want to thank all who wrote me for supporting this package by sending bug reports and ideas for new or better features.

1.1 Installation

Typically, `tcolorbox` will be installed as part of a major LaTeX distribution and there is nothing special to do for a user.

If you intend to make a local installation *by hand*, see the `README` file of the `tcolorbox` package for some hints. The short story is: you have to install not only `tcolorbox.sty`, but also all `*.code.tex` files in the local `texmf` tree.

1.2 Loading the Package

The base package `tcolorbox` loads the packages `pgf` [20], `verbatim` [15], `etoolbox` [7], and `environ` [13]. `tcolorbox` itself is loaded in the usual manner in the preamble:

```
\usepackage{tcolorbox}
```

The package takes option keys in the key-value syntax. Alternatively, you may use these keys later in the preamble with `\tcbuselibrary` → P. 8 (see there). For example, the key to typeset listings is:

```
\usepackage[listings]{tcolorbox}
```

1.3 Libraries

The base package `tcolorbox` is extendable by program libraries. This is done by using option keys while loading the package or inside the preamble by applying the following macro with the same set of keys.

`\tcbuselibrary{`⟨*key list*⟩`}`

Loads the libraries given by the ⟨*key list*⟩.

```
\tcbuselibrary{listings,theorems}
```

The following keys are used inside `\tcbuselibrary` respectively `\usepackage` without the key tree path `/tcb/library/`.

`/tcb/library/skins` (LIB skins)

Loads the package `tikz` [20] and provides additional styles (skins) for the appearance of the colored boxes; see Section 9 from page 124.

`/tcb/library/raster` (LIB raster)

Provides additional macros and options for typesetting multiple boxes arranged in a kind of raster; see Section 12 from page 231.

`/tcb/library/listings` (LIB listings)

Loads the package `listings` [6] and provides additional macros for typesetting listings which are described in Section 13 from page 242.

`/tcb/library/listingsutf8` (LIB listingsutf8)

Loads the packages `listings` [6] and `listingsutf8` [10] for UTF-8 support. This is a variant of the library LIB listings and is described in Section 13 from page 242.

`/tcb/library/minted` (LIB minted)

Loads the package `minted` [14] to typeset listings with the `Pygments` [12] tool, also see Section 13 on page 242.

`/tcb/library/theorems` (LIB theorems)

Provides additional macros for typesetting theorems which are described in Section 14 from page 281.

`/tcb/library/breakable` (LIB breakable)

Provides support for automatic box breaking from one page to another; see Section 15 on page 304.

`/tcb/library/magazine` (LIB magazine)

Provides support for storing broken box parts to be used later or in interchanged order, Section 16 on page 326.

`/tcb/library/fitting` (LIB fitting)

Provides support for font size adaption of the box content to the box dimensions; see Section 17 from page 334.

`/tcb/library/hooks` (LIB hooks)

Extends several option keys to 'hookable' keys; see Section 18 from page 345.

`/tcb/library/xparse` (LIB xparse)

Provides document command production with `xparse` for `tcolorbox`; see Section 19 from page 356.

/tcb/library/external (LIB external)

Provides externalization support for stand-alone document snippets, see Section 20 on page 369.

/tcb/library/documentation (LIB documentation)

Provides additional macros for typesetting LaTeX documentations which are described in Section 21 from page 380.

/tcb/library/many (style, no value)

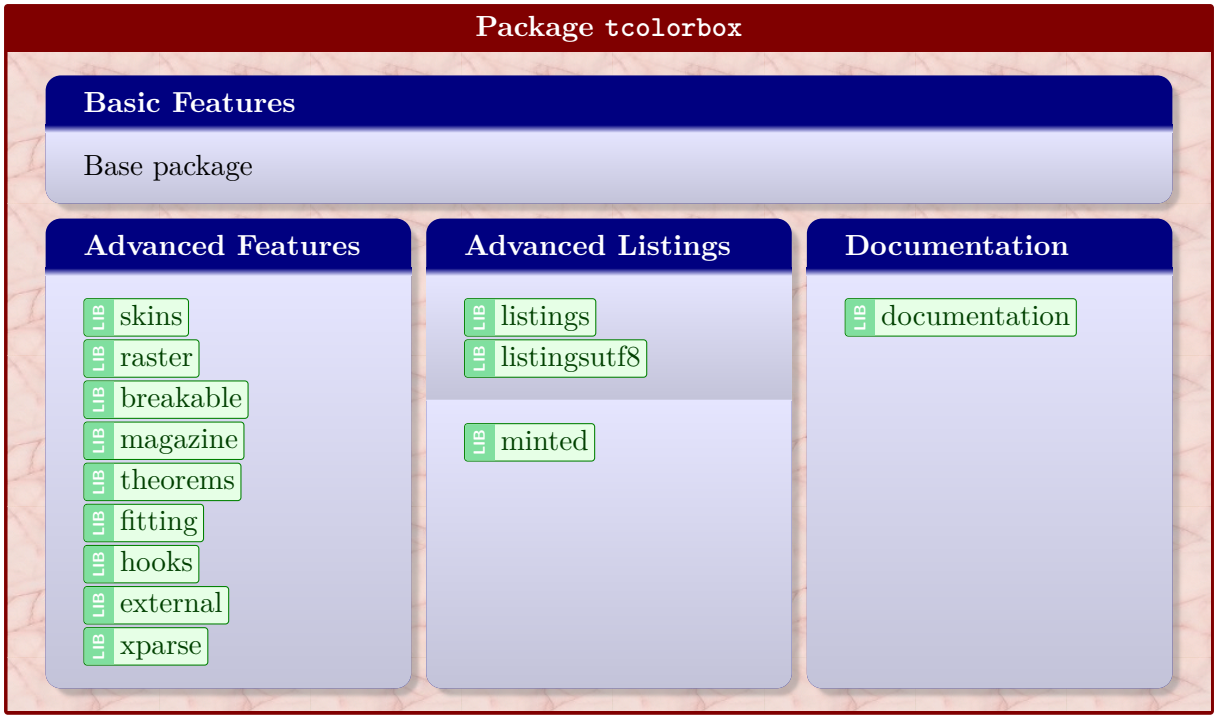
Loads the libraries LIB skins, LIB breakable, LIB raster, LIB hooks, LIB theorems, LIB fitting, and LIB xparse. Use this shortcut, if you want to use all features of `tcolorbox` with exception of typesetting listings and using the specialized LIB documentation library.

/tcb/library/most (style, no value)

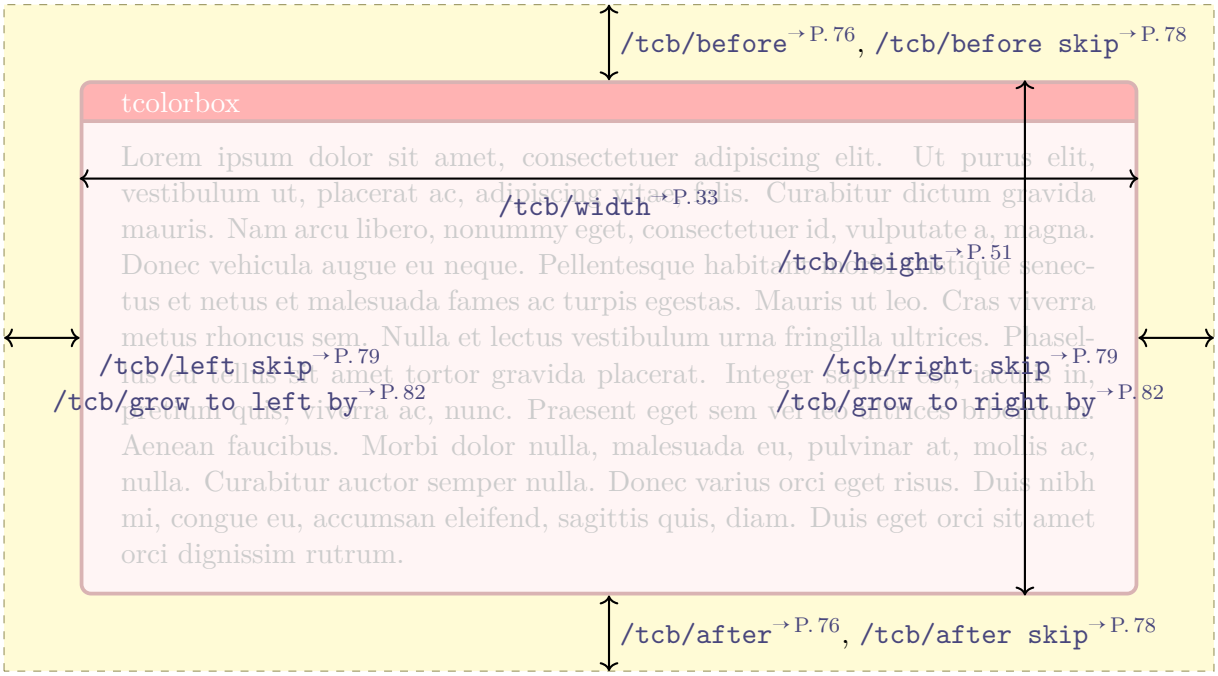
Loads all libraries except LIB minted and LIB documentation. Use this shortcut, if you want to use all features of `tcolorbox` with exception of using the `minted` package and using the specialized LIB documentation library.

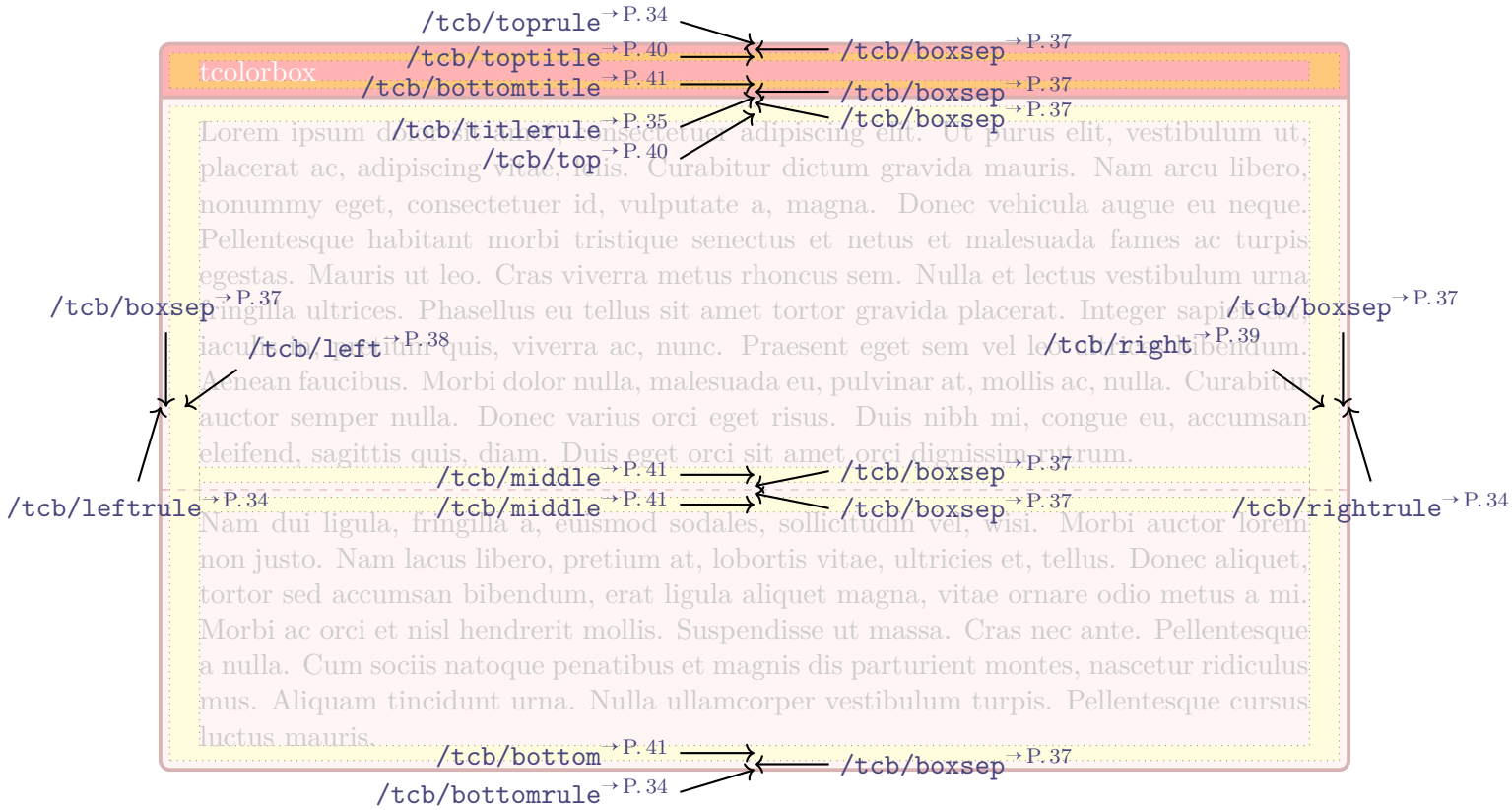
/tcb/library/all (style, no value)

Loads all libraries. Use this shortcut only, if you indend to use the LIB documentation library.



2 Quick Reference





3 Macros for Box Creation

\begin{tcolorbox}[⟨*options*⟩]
⟨*environment content*⟩
\end{tcolorbox}

This is the main environment to create an accentuated colored text box with rounded corners and, optionally, two parts. The appearance of this box is controlled by numerous options. In the most simple case the source code

```
\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

creates the following compiled text box:

This is a **tcolorbox**.

The text content of the box can be divided in an upper and a lower part by the command `\tcblower`. Visually, both parts are separated by a line. For example:

```
\begin{tcolorbox}
This is another \textbf{tcolorbox}.
\tcblower
Here, you see the lower part of the box.
\end{tcolorbox}
```

This code gives the following box:

This is another **tcolorbox**.

Here, you see the lower part of the box.

The ⟨*options*⟩ control the appearance and several functions of the boxes, see Section 4 on page 17 for the complete list. A quick example is given here:

```
\begin{tcolorbox}[colback=red!5!white,colframe=red!75!black,title=My nice heading]
This is another \textbf{tcolorbox}.
\tcblower
Here, you see the lower part of the box.
\end{tcolorbox}
```

My nice heading

This is another **tcolorbox**.

Here, you see the lower part of the box.

\tcblower

Used inside `tcolorbox` to separate the upper box part from the optional lower box part. The upper and the lower part are treated as separate functional units. If you only want to draw a line, see `\tcbline` → P. 181.

\tcbset{⟨*options*⟩}

Sets options for every following tcolorbox → P. 11 inside the current TEX group. By default, this does not apply to nested boxes, see Section 4.17 on page 86.

For example, the colors of the boxes may be defined for the whole document by this:

```
\tcbset{colback=red!5!white,colframe=red!75!black}
```

\tcbsetforeverylayer{⟨*options*⟩}

Sets options for every following tcolorbox → P. 11 inside the current TEX group. In contrast to \tcbset, this does also apply to nested boxes, see Section 4.17 on page 86. Technically, the ⟨*options*⟩ are appended to the default values for every tcolorbox which are applied by /tcb/reset → P. 95.

You should not use this macro, if you are not completely sure that you want to have the ⟨*options*⟩ also for boxes in boxes (in boxes in boxes . . .).

```
\tcbset{colback=green!10!white}
\tcbsetforeverylayer{colframe=red!75!black}

\begin{tcolorbox}[title=All options for this box]
  This is a tcolorbox.\par\medskip
  \begin{tcolorbox}[title=Nested box]
    Note that this nested box has a red frame but no green background.
  \end{tcolorbox}
\end{tcolorbox}
\bigskip

\begin{tcolorbox}[reset]
  Options given with |\tcbsetforeverylayer| survive a |reset|.
\end{tcolorbox}
```

All options for this box

This is a tcolorbox.

Nested box

Note that this nested box has a red frame but no green background.

Options given with \tcbsetforeverylayer survive a reset.

`\tcbox[⟨options⟩]{⟨box content⟩}`

Creates a colored box which is fitted to the width of the given ⟨*box content*⟩. In principle, most ⟨*options*⟩ for a `tcolorbox` → P. 11 can be used for `\tcbox` with some restrictions. A `\tcbox` cannot have a lower part and cannot be broken.

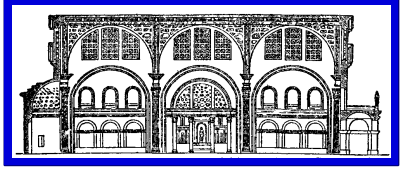
```
\tcbset{colframe=blue!50!black,colback=white,colupper=red!50!black,
        fonttitle=\bfseries,nobeforeafter,center title}

Text \tcbox[tcbox raise base]{Hello World}\hfill
%
\tcbox[left=0mm,right=0mm,top=0mm,bottom=0mm,boxsep=0mm,
  toptitle=0.5mm,bottomtitle=0.5mm,title=My table]{%
  \arrayrulecolor{blue!50!black}\renewcommand{\arraystretch}{1.2}%
  \begin{tabular}{r|c|l}
  One   & Two    & Three \\\hline\hline
  Men   & Mice   & Lions \\\hline
  Upper & Middle & Lower
  \end{tabular}}\hfill
%
\tcbox[colback=blue!85!black,
  left=0mm,right=0mm,top=0mm,bottom=0mm,boxsep=1mm,arc=0mm,boxrule=0.5pt,
  title=My picture]{%
  \includegraphics[width=5cm]{Basilica_5.png}}
```

Text Hello World

My table

One	Two	Three
Men	Mice	Lions
Upper	Middle	Lower

My picture



```
% \usepackage{tikz}
\tcbset{colframe=blue!50!black,colback=white,colupper=red!50!black,
        fonttitle=\bfseries,center title}

% Fixed width box
\begin{tcolorbox}Hello\\World!\end{tcolorbox}

% Fitted width box (like hbox or makebox)
\tcbox{Hello\\World!}

% Fitted width box (using a \tikzname\  node)
\tcbox[tikznode]{Hello\\World!}
```

Hello
World!

HelloWorld!

Hello
World!

See Section 19.2 on page 358 and Section 19.3 on page 361 for more elaborate methods to create new environments and commands.

\newtcolorbox[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Creates a new environment ⟨*name*⟩ based on `tcolorbox`→ P. 11. Basically, `\newtcolorbox` operates like `\newenvironment`. This means, the new environment ⟨*name*⟩ optionally takes ⟨*number*⟩ arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying `tcolorbox`. Note that `/tcb/savedelimiter`→ P. 25 is set to the given ⟨*name*⟩ automatically. The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

```
\newtcolorbox{mybox}{colback=red!5!white,
  colframe=red!75!black}

\begin{mybox}
This is my own box.
\end{mybox}
```

This is my own box.

```
\newtcolorbox{mybox}[1]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#1}

\begin{mybox}{Hello there}
This is my own box with a mandatory title.
\end{mybox}
```

Hello there

This is my own box with a mandatory title.

```
\newtcolorbox{mybox}[2][]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  colbacktitle=red!85!black,enhanced,
attach boxed title to top center={yshift=-2mm},
  title=#2,#1}

\begin{mybox}[colback=yellow]{Hello there}
This is my own box with a mandatory title
and options.
\end{mybox}
```

Hello there

This is my own box with a mandatory title and options.

Definition in the preamble:

```
\newtcolorbox[auto counter,number within=section]{pabox}[2][]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Examp.~\thetcbcounter: #2,#1}
```

```
\begin{pabox}[colback=yellow]{Hello there}
This is my own box with a mandatory
numbered title and options.
\end{pabox}
```

Examp. 3.1: Hello there

This is my own box with a mandatory numbered title and options.

\renewtcolorbox[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like `\newtcolorbox`, but based on `\renewenvironment` instead of `\newenvironment`. An existing environment is redefined.

`\newtcbox[⟨init options⟩]{\⟨name⟩}[⟨number⟩][⟨default⟩]{⟨options⟩}`

Creates a new macro \⟨*name*⟩ based on `\tcbox` → P. 13. Basically, `\newtcbox` operates like `\newcommand`. The new macro \⟨*name*⟩ optionally takes ⟨*number*⟩+1 arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying `tcbox`. The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

```
\newtcbox{\mybox}{colback=red!5!white,
  colframe=red!75!black}

\mybox{This is my own box.}
```

This is my own box.

```
\newtcbox{\mybox}[1]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#1}

\mybox{Hello there}{This is my own box.}
```

Hello there

This is my own box.

```
\newtcbox{\mybox}[2][]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#2,#1}

\mybox[colback=yellow]{Hello there}%
  {This is my own box.}
```

Hello there

This is my own box.

Definition in the preamble:

```
% counter from previous example
\newtcbox[use counter from=pabox]{\pbbox}[2][]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=(\thetcbcounter) #2,#1}
```

```
\pbbox[colback=yellow]{Hello there}%
  {This is my own box.}
```

(3.2) Hello there

This is my own box.

```
\newtcbox{\mybox}[1][red]{on line,
  arc=0pt,outer arc=0pt,colback=#1!10!white,colframe=#1!50!black,
  boxsep=0pt,left=1pt,right=1pt,top=2pt,bottom=2pt,
  boxrule=0pt,bottomrule=1pt,toprule=1pt}
\newtcbox{\xmybox}[1][red]{on line,
  arc=7pt,colback=#1!10!white,colframe=#1!50!black,
  before upper={\rule[-3pt]{0pt}{10pt}},boxrule=1pt,
  boxsep=0pt,left=6pt,right=6pt,top=2pt,bottom=2pt}

The \mybox[green]{quick} brown \mybox{fox} \mybox[blue]{jumps} over the
\mybox[green]{lazy} \mybox{dog}.\par
The \xmybox[green]{quick} brown \xmybox{fox} \xmybox[blue]{jumps} over the
\xmybox[green]{lazy} \xmybox{dog}.
```

The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog.

`\renewtcbox[⟨init options⟩]{\⟨name⟩}[⟨number⟩][⟨default⟩]{⟨options⟩}`

Operates like `\newtcbox`, but based on `\renewcommand` instead of `\newcommand`. An existing macro is redefined.

N 2014-10-20 **\tcolorboxenvironment**{⟨*name*⟩}{⟨*options*⟩}

An existing environment ⟨*name*⟩ is redefined to be boxed inside a `tcolorbox` with the given ⟨*options*⟩.

```
% tcbuselibrary{skins}
\newenvironment{myitemize}{%
  \begin{itemize}}{\end{itemize}}

\tcolorboxenvironment{myitemize}{blanker,
  before skip=6pt,after skip=6pt,
  borderline west={3mm}{0pt}{red}}

Some text.
\begin{myitemize}
\item Alpha
\item Beta
\item Gamma
\end{myitemize}
More text.
```

Some text.

- Alpha
- Beta
- Gamma

More text.

See further examples in Section 14.4 on page 303.

4 Option Keys

For the ⟨*options*⟩ in `tcolorbox` → P. 11 respectively `\tcbset` → P. 12 the following `pgf` keys can be applied. The key tree path `/tcb/` is not to be used inside these macros. It is easy to add your own style keys using the syntax for `pgf` keys, see [19, 20] or the examples starting from page 268.

4.1 Title

`/tcb/title=`⟨*text*⟩ (no default, initially empty)

Creates a heading line with ⟨*text*⟩ as content.

```
\begin{tcolorbox}[title=My heading line]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My heading line

This is a **tcolorbox**.

`/tcb/notitle` (no value, initially set)

Removes the title line if set before.

`/tcb/adjusted title=`⟨*text*⟩ (style, no default, initially unset)

Creates a heading line with ⟨*text*⟩ as content. The minimal height of this line is adjusted to fit the text given by `/tcb/adjust text`. This option makes sense for single line headings if boxes are set side by side with equal height. Note that it is very easy to trick this adjustment.

```
\tcbset{colback=White,arc=0mm,width=(\linewidth-4pt)/4,
equal height group=AT,before=,after=\hfill,fonttitle=\bfseries}

The following titles are not adjusted:\\
\foreach \n in {xxx,ggg,AAA,\"Agypten}
{\begin{tcolorbox}[title=\n,colframe=red!75!black]
  Some content.\end{tcolorbox}}
Now, we try again with adjusted titles:\\
\foreach \n in {xxx,ggg,AAA,\"Agypten}
{\begin{tcolorbox}[adjusted title=\n,colframe=blue!75!black]
  Some content.\end{tcolorbox}}
```

The following titles are not adjusted:

xxx	ggg	AAA	Ägypten
Some content.	Some content.	Some content.	Some content.

Now, we try again with adjusted titles:

xxx	ggg	AAA	Ägypten
Some content.	Some content.	Some content.	Some content.

`/tcb/adjust text=`⟨*text*⟩ (no default, initially `Äpgjy`)

This sets the reference text for `/tcb/adjusted title`. If your texts never exceed 'Äpgjy' in depth and height you don't need to care about this option.

N 2014-11-24 **/tcb/squeezed title=**⟨*text*⟩ (style, no default, initially unset)

Creates a single heading line with ⟨*text*⟩ as content. If the ⟨*text*⟩ is longer than the available space, the text is squeezed to fit into the available space.

```
% \tcbuselibrary{raster}
\begin{tcbitemize}[raster columns=3,raster equal height,
  colframe=red!75!black,colback=red!5!white,fonttitle=\bfseries]
\tcbitem[squeezed title={Short title}]
  First box
\tcbitem[squeezed title={This is a very very long title}]
  Second box
\tcbitem[squeezed title={This title is clearly to long for this application}]
  Third box
\end{tcbitemize}
```

Short title
First box

This is a very very long title
Second box

This title is clearly to long for this application
Third box

N 2014-11-24 **/tcb/squeezed title*=**⟨*text*⟩ (style, no default, initially unset)

This is a combination of /tcb/adjusted title → P. 17 and /tcb/squeezed title.

```
% \tcbuselibrary{raster}
\begin{tcbitemize}[raster columns=3,raster equal height,
  colframe=red!75!black,colback=red!5!white,fonttitle=\bfseries]
\tcbitem[squeezed title*={Short title}]
  First box
\tcbitem[squeezed title*={This is a very very long title}]
  Second box
\tcbitem[squeezed title*={This title is clearly to long for this application}]
  Third box
\end{tcbitemize}
```

Short title
First box

This is a very very long title
Second box

This title is clearly to long for this application
Third box

/tcb/detach title (no value)

Detaches the title from its normal position. The text of the title is stored into **\tcbtitletext** and the formatted title is available by **\tcbtitle**. The main application is to move the title from its usual place to another one.

```
\newtcolorbox{mybox}[2][]{colbacktitle=red!10!white,
  colback=blue!10!white,coltitle=red!70!black,
  title={#2},fonttitle=\bfseries,#1}

\begin{mybox}{My title}
  This is a \textbf{tcolorbox}.
\end{mybox}
\begin{mybox}[detach title,before upper={\tcbtitle\quad}]{My title}
  This is a \textbf{tcolorbox}.
\end{mybox}
\begin{mybox}[detach title,after upper={\par\hfill\tcbtitle}]{My title}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

My title

This is a **tcolorbox**.

My title This is a **tcolorbox**.

This is a **tcolorbox**.

My title

/tcb/attach title (no value)

Attaches the title to its normal position. This option is used to reverse **/tcb/detach title**.

U 2015-07-08 **/tcb/attach title to upper=**⟨*text*⟩ (style, default empty, initially unset)

Attaches the title to the begin of the upper part of the box content. The optional ⟨*text*⟩ is set between the formatted title and the box content.

```
\newtcolorbox{mybox}[2][]{colbacktitle=red!10!white,
  colback=blue!10!white,coltitle=red!70!black,
  title={#2},fonttitle=\bfseries,#1}

\begin{mybox}[attach title to upper={\ ---\ }]{My title}
  This is a \textbf{tcolorbox}.
\end{mybox}
\begin{mybox}[attach title to upper,after title={:\ }]{My title}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

My title — This is a **tcolorbox**.

My title: This is a **tcolorbox**.

More title options are documented in Section 4.11 on page 58 and Section 9.2 on page 131.

4.2 Subtitle

Inside the box content, one or more subtitles can be added. In general, a subtitle is a further tcolorbox→P. 11 which inherits some color and geometry options from the enclosing box. It may be customized just like any other tcolorbox→P. 11.

N 2014-10-10 **\tcbsubtitle**[⟨*options*⟩]{⟨*text*⟩}

Used inside a tcolorbox→P. 11 to add a subtitle box with the given ⟨*text*⟩. This is an independent tcolorbox→P. 11 which is formatted by several inherited properties of the enclosing box, by further settings from /tcb/subtitle style, and by the given ⟨*options*⟩.

```
\begin{tcolorbox}[title=My title,
    colback=red!5!white,
    colframe=red!75!black,
    fonttitle=\bfseries]
  This is a \textbf{tcolorbox}.
\tcbsubtitle[before skip=\baselineskip]%
  {My subtitle}
  Further text.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

My subtitle

Further text.

```
\begin{tcolorbox}[title=My title,
    colback=red!5!white,
    colframe=red!75!black,
    colbacktitle=yellow!50!red,
    coltitle=red!25!black,
    fonttitle=\bfseries]
  This is a \textbf{tcolorbox}.
\tcbsubtitle[before skip=\baselineskip]%
  {My subtitle}
  Further text.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

My subtitle

Further text.

N 2014-10-10 **/tcb/subtitle style**=⟨*options*⟩ (no default, initially empty)

Adds tcolorbox ⟨*options*⟩ to the settings for \tcbsubtitle.

```
\begin{tcolorbox}[title=My title,
  colback=red!5!white,
  colframe=red!75!black,
  colbacktitle=yellow!50!red,
  coltitle=red!25!black,
  fonttitle=\bfseries,
  subtitle style={boxrule=0.4pt,
    colback=yellow!50!red!25!white} ]
  This is a \textbf{tcolorbox}.
\tcbsubtitle{My subtitle}
  Further text.
\tcbsubtitle{Second subtitle}
  Further text.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

My subtitle

Further text.

Second subtitle

Further text.

4.3 Upper Part

The text content of a `tcolorbox`→ P. 11 may be parted into a mandatory *upper part* and an optional *lower part*. These parts are separated by `\tcblower`→ P. 11. If there is no `\tcblower`→ P. 11 present, there is no *lower part* and the *upper part* forms the complete text content.

N 2015-01-06 **`/tcb/upperbox`=⟨*mode*⟩** (no default, initially `visible`)

Controls the treatment of the upper part of the box. If there is no lower part, this is the complete text content. Feasible values for ⟨*mode*⟩ are:

- `visible`: usual type setting of the upper part,
- `invisible`: empty space instead of the upper part contents.

```
\begin{tcolorbox}[upperbox=invisible,colback=white]
This is a \textbf{tcolorbox} (but invisible).
\end{tcolorbox}

\bigskip

\begin{tcolorbox}[upperbox=invisible,colback=white]
This is a \textbf{tcolorbox} (but invisible).
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is the lower part.

N 2015-01-06 **`/tcb/visible`** (style, no value)

Shortcut for setting `/tcb/upperbox` and `/tcb/lowerbox`→ P. 23 to be `visible`.

N 2015-01-06 **`/tcb/invisible`** (style, no value)

Shortcut for setting `/tcb/upperbox` and `/tcb/lowerbox`→ P. 23 to be `invisible`.

```
\begin{tcolorbox}[invisible]
This is a \textbf{tcolorbox} (but invisible).
\end{tcolorbox}
```

N 2015-05-04 **/tcb/saveto=**⟨*file name*⟩ (no default, initially empty)

Saves the content of the box into a file for an optional later usage. This is the counterpart of /tcb/savelowerto → P. 23, but is saves not only the upper part but the whole content. If a lower part is present, it is also saved including \tcblower → P. 11.

This option cannot be combined with /tcb/savelowerto → P. 23.

```
\begin{tcolorbox}[invisible,saveto=\jobname_mysave1.tex,colback=white]
This is a \textbf{tcolorbox} which seems to be empty.
The content is saved for later usage.
\end{tcolorbox}

Now, we load the saved text:\\
\input{\jobname_mysave1.tex}
```

Now, we load the saved text:
This is a **tcolorbox** which seems to be empty. The content is saved for later usage.

```
\begin{tcolorbox}[saveto=\jobname_mysave2.tex]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}

Now, we load the saved text:
\begin{tcolorbox}[colframe=red,colback=red!10,
  coltitle=black,colbacktitle=red!20,sidebyside,
  title=Here we see the saved content including the lower part]
\input{\jobname_mysave2.tex}
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

Now, we load the saved text:

Here we see the saved content including the lower part

This is a **tcolorbox**. | This is the lower part.

4.4 Lower Part

/tcb/lowerbox=⟨*mode*⟩ (no default, initially `visible`)

Controls the treatment of the lower part of the box. Feasible values for ⟨*mode*⟩ are:

- `visible`: usual type setting of the lower part,
- `invisible`: empty space instead of the lower part contents,
- `ignored`: the lower part is not used (here).

The last two values are usually applied in connection with `savelowerto`.

```
\begin{tcolorbox}[lowerbox=invisible,colback=white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part (but invisible).
\end{tcolorbox}

\begin{tcolorbox}[lowerbox=ignored,colback=white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part (but ignored).
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

U 2014-11-28 **/tcb/savelowerto=**⟨*file name*⟩ (no default, initially empty)

Saves the content of the lower part into a file for an optional later usage.

```
\begin{tcolorbox}[lowerbox=invisible,savelowerto=\jobname_bspsave.tex,colback=white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part which may be quite complex:
$\displaystyle f(x)=\frac{1+x^2}{1-x^2}$.
\end{tcolorbox}

Now, we load the saved text:\\
\input{\jobname_bspsave.tex}
```

This is a **tcolorbox**.

Now, we load the saved text:

This is the lower part which may be quite complex: $f(x)=\frac{1+x^2}{1-x^2}$.

`/tcb/lower separated=true|false` (default `true`, initially `true`)

If set to `true`, the lower part is visually separated from the upper part. It depends on the chosen skin how the visualization of the separation is done.

```
% \tcbuselibrary{skins,raster}
\begin{tcbraster}[colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries,fontlower=\itshape]
%
\begin{tcolorbox}[title=Lower separated]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
%
\begin{tcolorbox}[title=Lower not separated,lower separated=false]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
%
\begin{tcolorbox}[sidebyside,title=Lower separated]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
%
\begin{tcolorbox}[sidebyside,title=Lower not separated,lower separated=false]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
%
\begin{tcolorbox}[beamer,title=Lower separated]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
%
\begin{tcolorbox}[beamer,title=Lower not separated,lower separated=false]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
%
\end{tcbraster}
```

Lower separated

This is the upper part.

This is the lower part.

Lower not separated

This is the upper part.

This is the lower part.

Lower separated

This is the upper part.

This is the lower part.

Lower not separated

This is the upper part.

This is the lower part.

Lower separated

This is the upper part.

This is the lower part.

Lower not separated

This is the upper part.

This is the lower part.

/tcb/savedelimiter=⟨*name*⟩ (no default, initially `tcolorbox`)

Used in connection with new environment definitions which extend `tcolorbox` and use or allow the option `savelowerto`. To catch the end of the new box environment ⟨*name*⟩ has to be the name of this environment. Additionally, the environment definition has to use `\tcolorbox` instead of `\begin{tcolorbox}` and `\endtcolorbox` instead of `\end{tcolorbox}`.

```
\newenvironment{mybox}[1]{%
  \tcolorbox[savedelimiter=mybox,
             savelowerto=\jobname_bspsave2.tex,lowerbox=ignored,
             colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
             title=#1]}%
  {\endtcolorbox}

\begin{mybox}{My Example}
Upper part.
\tcblower
Saved lower part!
\end{mybox}

Now, the saved part is used:
\begin{tcolorbox}[colback=green!5]
\input{\jobname_bspsave2.tex}
\end{tcolorbox}
```

My Example

Upper part.

Now, the saved part is used:

Saved lower part!

The `savedelimiter` is used implicitely with `\newtcolorbox` → P. 14 which allows a more convenient usage:

```
\newtcolorbox{mybox}[1]{%
             savelowerto=\jobname_bspsave2.tex,lowerbox=ignored,
             colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
             title=#1}%

\begin{mybox}{My Example}
Upper part.
\tcblower
Saved lower part!
\end{mybox}

Now, the saved part is used:
\begin{tcolorbox}[colback=green!5]
\input{\jobname_bspsave2.tex}
\end{tcolorbox}
```

My Example

Upper part.

Now, the saved part is used:

Saved lower part!

4.5 Colors and Fonts

/tcb/colframe=⟨*color*⟩ (no default, initially `black!75!white`)

Sets the frame ⟨*color*⟩ of the box.

```
\begin{tcolorbox}[colframe=red!50!white]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/colback=⟨*color*⟩ (no default, initially `black!5!white`)

Sets the background ⟨*color*⟩ of the box.

```
\begin{tcolorbox}[colback=red!50!white]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/title filled=true|false (default `true`, initially `false`)

Switches the drawing of the title background according to the given value. This option is set to `true` automatically by `/tcb/colbacktitle`, `/tcb/opacitybacktitle` → P. 49, and `/tcb/title style` → P. 127, and `/tcb/title code` → P. 118.

```
\begin{tcolorbox}[title=My title,title filled]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[title=My title,
  title filled=false]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

My title

This is a **tcolorbox**.

/tcb/colbacktitle=⟨*color*⟩ (no default, initially `black!50!white`)

Sets the background ⟨*color*⟩ of the title area of the box.

```
\begin{tcolorbox}[colbacktitle=red!50!white,
  title=My title,coltitle=black,
  fonttitle=\bfseries]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/colupper=⟨*color*⟩ (no default, initially `black`)

Sets the text ⟨*color*⟩ of the upper part.

```
\begin{tcolorbox}[colupper=red!75!black]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/collower=⟨*color*⟩ (no default, initially `black`)

Sets the text ⟨*color*⟩ of the lower part.

```
\begin{tcolorbox}[collower=red!75!black]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/coltext=⟨*color*⟩ (style, no default, initially `black`)

Sets the text ⟨*color*⟩ of the box. This is an abbreviation for setting `colupper` and `collower` to the same value.

```
\begin{tcolorbox}[coltext=red!75!black]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/coltitle=⟨*color*⟩ (no default, initially `white`)

Sets the title text ⟨*color*⟩ of the box.

```
\begin{tcolorbox}[coltitle=red!75!black,
  colbacktitle=black!10!white,title=Test]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Test

This is a **tcolorbox**.

/tcb/fontupper=⟨*text*⟩ (no default, initially empty)

Sets ⟨*text*⟩ before the content of the upper part (e. g. font settings).

```
\begin{tcolorbox}[fontupper=Hello!~\sffamily]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Hello! This is a **tcolorbox**.

/tcb/fontlower=⟨*text*⟩ (no default, initially empty)

Sets ⟨*text*⟩ before the content of the lower part (e. g. font settings).

```
\begin{tcolorbox}[fontlower=\sffamily\bfseries]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/fonttitle=⟨*text*⟩ (no default, initially empty)

Sets ⟨*text*⟩ before the content of the title text (e. g. font settings).

```
\begin{tcolorbox}[fonttitle=\sffamily\bfseries\large,title=Hello]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Hello

This is a **tcolorbox**.

More color options are provided by using skins documented in Section 9 from page 124.

4.6 Text Alignment

N 2015-05-07 **/tcb/halign=**⟨*alignment*⟩ (no default, initially `justify`)

If there is no lower part, `halign` determines the horizontal ⟨*alignment*⟩ of the text content. Otherwise, `halign` determines the horizontal ⟨*alignment*⟩ of the upper part of the box only. The feasible values for ⟨*alignment*⟩ are more or less identical to the corresponding `/tikz/align` settings, even if the implementation differs.

- `justify`: usual left and right justified type setting.
- `left`: left border justification in analogy to plain TEX.
- `flush left`: left border justification with `\raggedright` of LATEX.
- `right`: right border justification in analogy to plain TEX.
- `flush right`: right border justification with `\raggedleft` of LATEX.
- `center`: centering in analogy to plain TEX.
- `flush center`: centering with `\centering` of LATEX.

The differences between the flush and non-flush version are explained in detail in the TikZ manual [20]. The short story is that the non-flush versions will often look more balanced but with more hyphenations.

```
\tcbset{colback=red!5!white,colframe=red!75!black,size=small,
  fonttitle=\bfseries,width=3.5cm,box align=top,
  nobeforeafter}

\begin{tcolorbox}[adjusted title=flush center,halign=flush center]
This is a demonstration text for showing how line breaking works.
\end{tcolorbox}
\begin{tcolorbox}[adjusted title=flush left,halign=flush left]
This is a demonstration text for showing how line breaking works.
\end{tcolorbox}
\begin{tcolorbox}[adjusted title=flush right,halign=flush right]
This is a demonstration text for showing how line breaking works.
\end{tcolorbox}

\begin{tcolorbox}[adjusted title=center,halign=center]
This is a demonstration text for showing how line breaking works.
\end{tcolorbox}
\begin{tcolorbox}[adjusted title=left,halign=left]
This is a demonstration text for showing how line breaking works.
\end{tcolorbox}
\begin{tcolorbox}[adjusted title=right,halign upper=right]
This is a demonstration text for showing how line breaking works.
\end{tcolorbox}
```

flush center
This is a demonstration text for showing how line breaking works.

flush left
This is a demonstration text for showing how line breaking works.

flush right
This is a demonstration text for showing how line breaking works.

center
This is a demonstration text for showing how line breaking works.

left
This is a demonstration text for showing how line breaking works.

right
This is a demonstration text for showing how line breaking works.

N 2015-05-07 **/tcb/halign upper=**⟨*alignment*⟩ (no default, initially `justify`)

Alias for `/tcb/halign`.

N 2015-05-07 **/tcb/halign lower=**⟨*alignment*⟩ (no default, initially **justify**)

halign lower determines the horizontal ⟨*alignment*⟩ of the lower part of the box. The feasible values for ⟨*alignment*⟩ are the same as for **/tcb/halign** → P. 29.

```
\begin{tcbraster}[raster columns=3,fonttitle=\bfseries,
  colback=red!5!white,colframe=red!75!black]

  \begin{tcolorbox}[adjusted title=flush center,halign lower=flush center]
    Upper part. \tcblower Lower part.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=flush left,halign lower=flush left]
    Upper part. \tcblower Lower part.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=flush right,halign lower=flush right]
    Upper part. \tcblower Lower part.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=center,halign lower=center]
    Upper part. \tcblower Lower part.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=left,halign lower=left]
    Upper part. \tcblower Lower part.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=right,halign lower=right]
    Upper part. \tcblower Lower part.
  \end{tcolorbox}

\end{tcbraster}
```

flush center
Upper part.
Lower part.

flush left
Upper part.
Lower part.

flush right
Upper part.
Lower part.

center
Upper part.
Lower part.

left
Upper part.
Lower part.

right
Upper part.
Lower part.

N 2015-05-07 **/tcb/halign title=⟨*alignment*⟩** (no default, initially `justify`)

`halign lower` determines the horizontal ⟨*alignment*⟩ of the title of the box. The feasible values for ⟨*alignment*⟩ are the same as for `/tcb/halign` → P. 29.

```
\begin{tcbraster}[raster columns=3,fonttitle=\bfseries,
  colback=red!5!white,colframe=red!75!black]

  \begin{tcolorbox}[adjusted title=flush center,halign title=flush center]
    This is a \textbf{tcolorbox}.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=flush left,halign title=flush left]
    This is a \textbf{tcolorbox}.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=flush right,halign title=flush right]
    This is a \textbf{tcolorbox}.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=center,halign title=center]
    This is a \textbf{tcolorbox}.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=left,halign title=left]
    This is a \textbf{tcolorbox}.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=right,halign title=right]
    This is a \textbf{tcolorbox}.
  \end{tcolorbox}

\end{tcbraster}
```

flush center
This is a **tcolorbox**.

flush left
This is a **tcolorbox**.

flush right
This is a **tcolorbox**.

center
This is a **tcolorbox**.

left
This is a **tcolorbox**.

right
This is a **tcolorbox**.

U 2015-05-07 **/tcb/flushleft upper** (style, no value)

Shortcut for setting `/tcb/halign` → P. 29 to `flush left`.

U 2015-05-07 **/tcb/center upper** (style, no value)

Shortcut for setting `/tcb/halign` → P. 29 to `flush center`.

U 2015-05-07 **/tcb/flushright upper** (style, no value)

Shortcut for setting `/tcb/halign` → P. 29 to `flush right`.

U 2015-05-07 **/tcb/flushleft lower** (style, no value)

Shortcut for setting `/tcb/halign lower` → P. 30 to `flush left`.

U 2015-05-07 **/tcb/center lower** (style, no value)

Shortcut for setting `/tcb/halign lower` → P. 30 to `flush center`.

U 2015-05-07 **/tcb/flushright lower** (style, no value)

Shortcut for setting `/tcb/halign lower` → P. 30 to `flush right`.

U 2015-05-07 **/tcb/flushleft title** (style, no value)

Shortcut for setting `/tcb/halign title` to `flush left`.

U 2015-05-07 **/tcb/center title** (style, no value)

Shortcut for setting `/tcb/halign title` to `flush center`.

U 2015-05-07 **/tcb/flushright title** (style, no value)

Shortcut for setting `/tcb/halign title` to `flush right`.

The vertical alignment settings are only relevant for boxes which are larger than their natural height, see Section 4.10 on page 51.

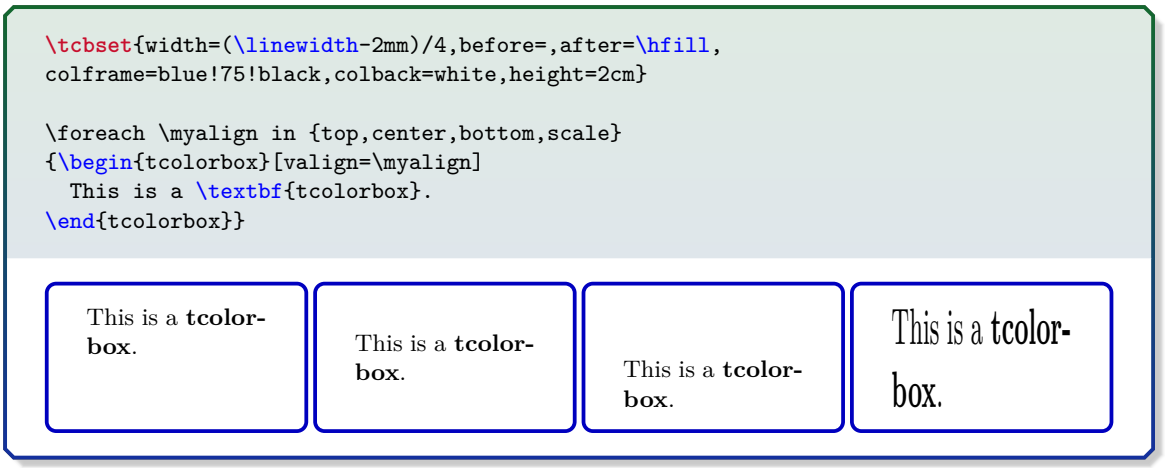
U 2015-07-16 **/tcb/valign=⟨*alignment*⟩** (no default, initially `top`)

If the height of a `tcolorbox` is not the natural height, `valign` determines the vertical ⟨*alignment*⟩ of the upper part. Feasible values are

- `top`: Anchor text at top.
- `center`: Anchor text at center.
- `bottom`: Anchor text at bottom.
- `scale`: Scale text vertically to fit into the available space. This is brutal and may not look very good. Consider Section 17 on page 334 alternatively.
- `scale*`: Like `scale`, but scaling is bounded by `/tcb/valign scale limit`.

For a box with natural height, these settings are meaningless.

```
\tcbset{width=(\linewidth-2mm)/4,before=,after=\hfill,
colframe=blue!75!black,colback=white,height=2cm}

\foreach \myalign in {top,center,bottom,scale}
{\begin{tcolorbox}[valign=\myalign]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}}
```



N 2015-05-07 **/tcb/valign upper=⟨*alignment*⟩** (no default, initially `top`)

Alias for `/tcb/valign`.

/tcb/valign lower=⟨*alignment*⟩ (no default, initially `top`)

This key has the same meaning for the lower part as `valign` for the upper part, i. e., it determines the vertical ⟨*alignment*⟩ of the lower part with feasible values `top`, `center`, `bottom`, `scale`, and `scale*`.

N 2015-07-16 **/tcb/valign scale limit=⟨*real number*⟩** (no default, initially `1.1`)

Sets an upper scale limit for the `scale*` setting in `/tcb/valign` and `/tcb/valign lower`. Note that this value is not reset by `/tcb/reset` → P. 95. So, changes also apply to embedded boxes.

Also see `/tcb/sidebyside align` → P. 71 for alignment settings when upper part and lower part are set side-by-side.

4.7 Geometry

4.7.1 Width

/tcb/width=⟨*length*⟩ (no default, initially `\linewidth`)

Sets the total width of the colored box to ⟨*length*⟩. See also **/tcb/height** → P. 51.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=\linewidth/2]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-31 **/tcb/text width=**⟨*length*⟩ (style, no default)

Sets the text width of the upper part to ⟨*length*⟩. See also **/tcb/text height** → P. 52.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[text width=4cm]
This is a \textbf{tcolorbox} where the text has a width of 4cm.
\end{tcolorbox}
```

This is a **tcolorbox** where the text has a width of 4cm.

N 2014-11-07 **/tcb/add to width=**⟨*length*⟩ (style, no default)

Adds ⟨*length*⟩ to the current total width of the colored box.

```
\tcbset{width=4cm,colback=red!5!white,
  colframe=red!75!black}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\end{tcolorbox}

\begin{tcolorbox}[add to width=1cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

See Section 4.10 on page 51 for setting fixed height values.

4.7.2 Rules

/tcb/toprule=⟨*length*⟩ (no default, initially `0.5mm`)

Sets the line width of the top rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[toprule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/bottomrule=⟨*length*⟩ (no default, initially `0.5mm`)

Sets the line width of the bottom rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[bottomrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/leftrule=⟨*length*⟩ (no default, initially `0.5mm`)

Sets the line width of the left rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[leftrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/rightrule=⟨*length*⟩ (no default, initially `0.5mm`)

Sets the line width of the right rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[rightrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/titlerule=⟨*length*⟩ (no default, initially `0.5mm`)

Sets the line width of the rule below the title to ⟨*length*⟩.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,
  colbacktitle=red!90!black}

\begin{tcolorbox}[titlerule=3mm,title=This is the title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is the title

This is a **tcolorbox**.

/tcb/boxrule=⟨*length*⟩ (style, no default, initially `0.5mm`)

Sets all rules of the frame to ⟨*length*⟩, i. e. /tcb/toprule→ P. 34, /tcb/bottomrule→ P. 34, /tcb/leftrule→ P. 34, /tcb/rightrule→ P. 34, and /tcb/titlerule.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[boxrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

More options for drawing a /tcb/borderline→ P. 151 are provided by using skins documented in Section 9 from page 124.

4.7.3 Arcs

/tcb/arc=⟨*length*⟩ (no default, initially `1mm`)

Sets the inner radius of the four frame arcs to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[arc=0mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[arc=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

N 2015-05-05 **/tcb/circular arc** (style, no value)

Sets /tcb/arc → P. 35 to match the half of the inner width of the colored box. If width and height of the box are identical, this gives a circle.

If the height of the box is smaller than the width, the result will look quite ugly.

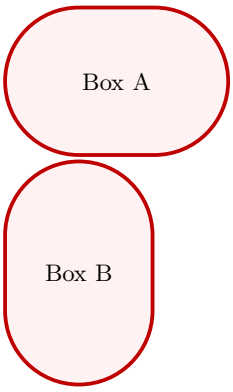
```
\begin{tcolorbox}[width=3cm,
  colback=red!5!white,
  colframe=red!75!black,
  halign=center,valign=center,
  square,circular arc]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

N 2015-05-05 **/tcb/bean arc** (style, no value)

Sets /tcb/arc → P. 35 to match the smaller value of the half of the inner width and of the inner height of the colored box.

This only works for a fixed /tcb/height → P. 51. Also, /tcb/bean arc must be used *after* width and height are set by option keys.

```
\tcbset{size=fbox,boxrule=0.5mm,
  colback=red!5!white,
  colframe=red!75!black,
  halign=center,valign=center}

\begin{tcolorbox}[width=3cm,height=2cm,
  bean arc]
Box A
\end{tcolorbox}

\begin{tcolorbox}[width=2cm,height=3cm,
  bean arc]
Box B
\end{tcolorbox}
```



N 2015-05-05 **/tcb/octogon arc** (style, no value)

Sets /tcb/arc → P. 35 to match $\frac{1}{2+\sqrt{2}}$ of the inner width of the colored box. If width and height of the box are identical, the interior is a regular octogon.

```
\begin{tcolorbox}[enhanced,
  size=minimal,auto outer arc,
  width=2.1cm,octogon arc,
  colback=red,colframe=white,colupper=white,
  fontupper=\fontsize{7mm}{7mm}\selectfont\bfseries\sffamily,
  halign=center,valign=center,
  square,arc is angular,
  borderline={0.2mm}{-1mm}{red}  ]
STOP
\end{tcolorbox}
```

N 2015-05-05 **/tcb/arc is angular** (no value, initially unset)

Using this options applies a patch which straightens the corners arcs of the boxes. The little arcs are replaced by little straight lines.

This patch is considered as an experimental feature. It changes some of the original TikZ code. This change may break with future updates of TikZ.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  arc=3mm}

\begin{tcolorbox}[arc is angular]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[arc is curved]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

N 2015-05-05 **/tcb/arc is curved** (no value, initially set)

This option resets the patch from /tcb/arc is angular. The original TikZ code is activated.

/tcb/outer arc=⟨*length*⟩ (no default, initially unset)

Sets the outer radius of the four frame arcs to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[arc=4mm,outer arc=1mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/auto outer arc (no value, initially set)

Sets the outer radius of the four frame arcs automatically in dependency of the inner radius given by /tcb/arc → P. 35.

4.7.4 Spacing

/tcb/boxsep=⟨*length*⟩ (no default, initially 1mm)

Sets a common padding of ⟨*length*⟩ between the text content and the frame of the box. This value is added to the key values of left, right, top, bottom, and middle at the appropriate places.

```
\tcbset{colback=red!5!white,colframe=red!75!black,width=(\linewidth-4mm)/2,
  before=,after=\hfill}

\begin{tcolorbox}[boxsep=5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[boxsep=5mm,draft]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

frame: w=195.33255pt, h=48.97505pt

upper: w=111.2724pt, h=6.296pt

interior: w=192.4873pt, h=46.1298pt

/tcb/left=⟨*length*⟩ (style, no default, initially 4mm)

Sets the left space between all text parts and frame (additional to `boxsep`). This is an abbreviation for setting `lefttitle`, `leftupper`, and `leftlower` to the same value.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[left=0mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/lefttitle=⟨*length*⟩ (no default, initially 4mm)

Sets the left space between title text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[lefttitle=3cm,title=My Title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My Title

This is a **tcolorbox**.

/tcb/leftupper=⟨*length*⟩ (no default, initially 4mm)

Sets the left space between upper text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[leftupper=3cm,title=My Title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My Title

This is a **tcolorbox**.

/tcb/leftlower=⟨*length*⟩ (no default, initially 4mm)

Sets the left space between lower text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[leftlower=3cm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/right=⟨*length*⟩ (style, no default, initially 4mm)

Sets the right space between all text parts and frame (additional to boxsep). This is an abbreviation for setting righttitle, rightupper, and rightlower to the same value.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,right=2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/righttitle=⟨*length*⟩ (no default, initially 4mm)

Sets the right space between title text and frame (additional to boxsep).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,righttitle=2cm,title=My very long title text]
This is a \textbf{tcolorbox} with standard upper box dimensions.
\end{tcolorbox}
```

My very long title text

This is a **tcolorbox** with standard upper box dimensions.

/tcb/rightupper=⟨*length*⟩ (no default, initially 4mm)

Sets the right space between upper text and frame (additional to boxsep).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,rightupper=2cm,title=My very long title text]
This is a \textbf{tcolorbox} with compressed upper box dimensions.
\end{tcolorbox}
```

My very long title text

This is a **tcolorbox** with compressed upper box dimensions.

/tcb/rightlower=⟨*length*⟩ (no default, initially `4mm`)

Sets the right space between lower text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,rightlower=2cm]
This is a \textbf{tcolorbox} with standard upper box dimensions.
\tcblower
This is the lower part with large space at right.
\end{tcolorbox}
```

This is a **tcolorbox** with standard upper box dimensions.

This is the lower part with large space at right.

/tcb/top=⟨*length*⟩ (no default, initially `2mm`)

Sets the top space between text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[top=0mm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/toptitle=⟨*length*⟩ (no default, initially `0mm`)

Sets the top space between title and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[toptitle=3mm,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/bottom=⟨*length*⟩ (no default, initially 2mm)

Sets the bottom space between text and frame (additional to boxsep).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[bottom=0mm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/bottomtitle=⟨*length*⟩ (no default, initially 0mm)

Sets the bottom space between title and frame (additional to boxsep).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[bottomtitle=3mm,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/middle=⟨*length*⟩ (no default, initially 2mm)

Sets the space between upper and lower text to the separation line (additional to boxsep).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[middle=0mm,boxsep=0mm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.
This is the lower part.

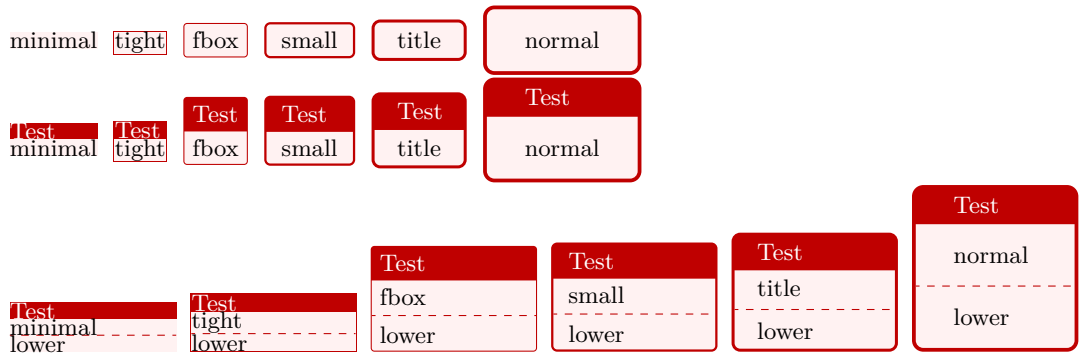
4.7.5 Size Shortcuts

/tcb/size=⟨*name*⟩ (no default, initially `normal`)

Sets all geometry keys with exception of `/tcb/width` → P. 33 to predefined length values. For ⟨*name*⟩, the following values are feasible:

- `normal`: normal sized boxes e.g. of width `\linewidth`.
- `title`: title line sized boxes.
- `small`: small boxes e.g. for keyword highlighting.
- `fbox`: identical to the standard `\fbox`.
- `tight`: no padding space at all.
- `minimal`: no padding space, no box rules.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\foreach \s in {minimal,tight,fbox,small,title,normal} {
  \tcbox[size=\s,on line]{\s} }

\foreach \s in {minimal,tight,fbox,small,title,normal} {
  \tcbox[size=\s,on line,title=Test]{\s} }

\foreach \s in {minimal,tight,fbox,small,title,normal} {
  \begin{tcolorbox}[size=\s,on line,title=Test,width=2.2cm]
    \s \tcblower lower\end{tcolorbox} }
```



Predefined values						
	`normal`	`title`	`small`	`fbox`	`tight`	`minimal`
`boxrule`	0.5mm	0.4mm	0.3mm	0.4pt	0.4pt	0.0pt
`boxsep`	1.0mm	1.0mm	1.0mm	3.0pt	0.0pt	0.0pt
`left`	4.0mm	2.0mm	1.0mm	0.0pt	0.0pt	0.0pt
`right`	4.0mm	2.0mm	1.0mm	0.0pt	0.0pt	0.0pt
`top`	2.0mm	0.25mm	0.0mm	0.0pt	0.0pt	0.0pt
`bottom`	2.0mm	0.25mm	0.0mm	0.0pt	0.0pt	0.0pt
`toptitle`	0.0mm	0.0mm	0.0mm	0.0pt	0.0pt	0.0pt
`bottomtitle`	0.0mm	0.0mm	0.0mm	0.0pt	0.0pt	0.0pt
`middle`	2.0mm	0.75mm	0.5mm	1.0pt	0.2pt	0.0pt
`arc`	1.0mm	0.75mm	0.5mm	1.0pt	0.0pt	0.0pt
`outer arc`	`auto`	`auto`	`auto`	`auto`	0.0pt	0.0pt

/tcb/oversize=⟨*length*⟩ (style, default `0pt`)

Sets the text width of the upper part to the current line width plus an optional ⟨*length*⟩. This is achieved by changing the keys `/tcb/width` →P. 33 `/tcb/enlarge left by` →P. 81, and `/tcb/enlarge right by` →P. 81 appropriately. The resulting box is overlapping into the left and right margin of the page. Note that this style option has to be given *after* all other geometry keys!

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\textit{Normal text for comparison:}\\
\lipsum[2]

\begin{tcolorbox}[oversize,title=Oversized box]
\lipsum[2]
\end{tcolorbox}

\begin{tcolorbox}[title=Normal box]
\lipsum[2]
\end{tcolorbox}
```

Normal text for comparison:
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.

Oversized box

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.

Normal box

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.

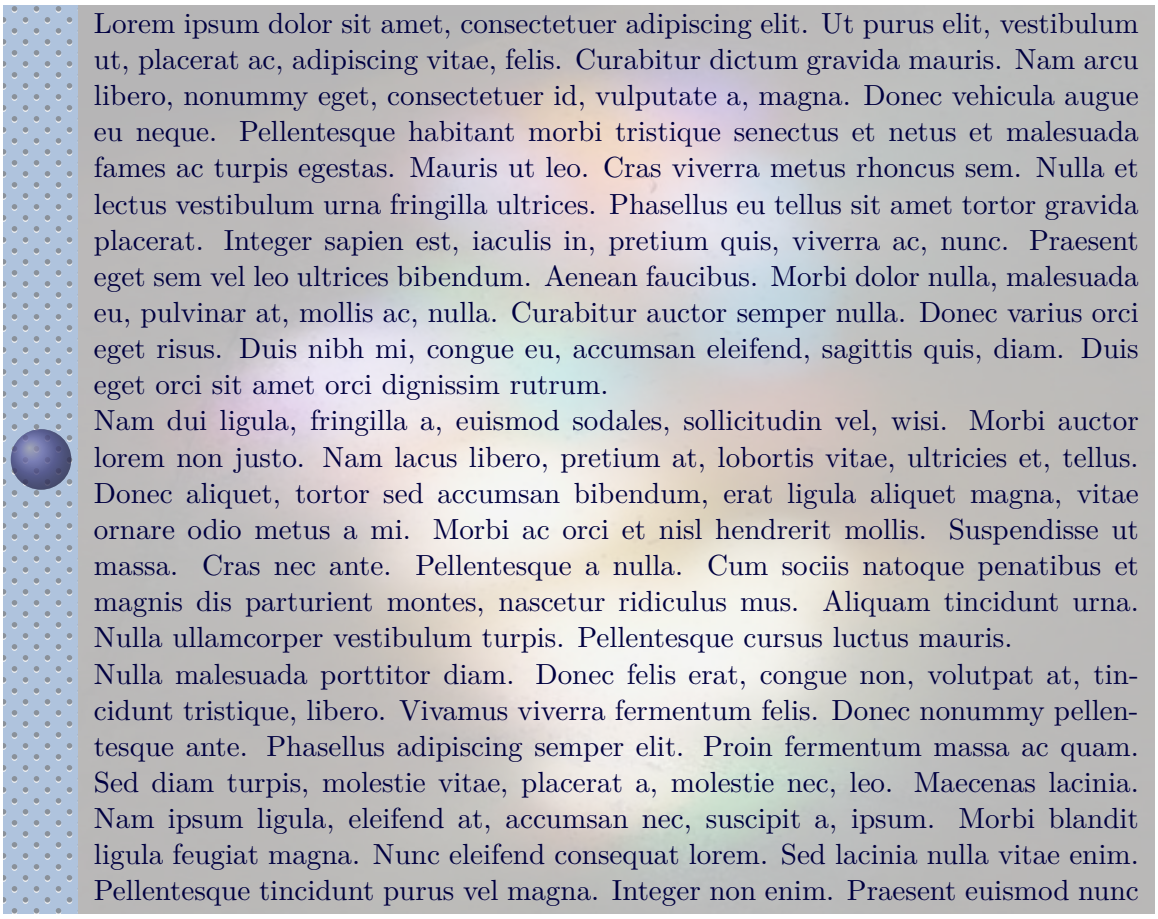
/tcb/toggle left and right=⟨*toggle preset*⟩ (default `evenpage`, initially `none`)

According to the ⟨*toggle preset*⟩, the left and the right settings of the `tcolorbox` are switched or not. Feasible values are:

- `none`: no switching.
- `forced`: the values of the left and right rules and spaces are switched.
- `evenpage`: if the page is an even page, the values of the left and right rules and spaces are switched. It is recommended that you use this setting in conjunction with `/tcb/check odd page` → P. 94.

```
% \usepackage{changepage} for 'check odd page'
% \usepackage{lipsum}
% \usetikzlibrary{patterns}
% \tcbuselibrary{skins,breakable}
\begin{tcolorbox}[skin=enhancedmiddle,breakable,
  check odd page,toggle left and right,
  boxrule=0mm,top=0mm,bottom=0mm,left=1mm,right=1mm,
  rightrule=1cm,colupper=blue!25!black,
  interior style={fill overzoom image=lichtspiel.jpg,fill image opacity=0.25},
  frame style={pattern=crosshatch dots light steel blue},
  overlay={%
    \ifoddpage\coordinate (X) at ([xshift=-5mm]frame.east);
         \else\coordinate (X) at ([xshift=5mm]frame.west);\fi
    \fill[shading=ball,ball color=blue!50!white,opacity=0.5] (X) circle (4mm);}]
\lipsum[1-6]
\end{tcolorbox}
```

This example switches a `1cm` thick rule from the left to the right side depending on the page number. Thereby, the rule is always on the outer side of the double-sided paper. Additionally, a ball is drawn on the outer side with help of an overlay.


Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc

eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

4.8 Corners

The four corners of any tcolorbox can be set individually as /tcb/sharp corners or as /tcb/rounded corners→ P. 47. These settings are also reflected in the behavior of /tcb/borderline→ P. 151 and /tcb/shadow→ P. 162 as one would expect.

By default, all four corners are *rounded*. So, only the /tcb/sharp corners option will be necessary for most use cases. The /tcb/rounded corners→ P. 47 option can be used to revert a /tcb/sharp corners setting.

/tcb/sharp corners=⟨*position*⟩ (default all, initially unset)

The ⟨*position*⟩ denotes one or more of the four box corners to be set as *sharp* corners. The not assigned corners will retain their mode. Feasible values for ⟨*position*⟩ are:

- northwest
- northeast
- southwest
- southeast
- north
- south
- east
- west
- downhill
- uphill
- all

```
\begin{tcolorbox}[colback=red!5!white,
  colframe=red!75!black,
  sharp corners=northwest ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

```
\begin{tcolorbox}[colback=red!5!white,
  colframe=red!75!black,
  sharp corners ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/rounded corners=⟨*position*⟩ (default `all`, initially `all`)

The `/tcb/rounded corners` can be used to revert a `/tcb/sharp corners` → P. 46 setting. The ⟨*position*⟩ denotes one or more of the four box corners to be set as *rounded* corners. The not assigned corners will retain their mode. Feasible values for ⟨*position*⟩ are[2]:

- `northwest`
- `northeast`
- `southwest`
- `southeast`
- `north`
- `south`
- `east`
- `west`
- `downhill`
- `uphill`
- `all`

```
\begin{tcolorbox}[colback=red!5!white,
  colframe=red!75!black,sharp corners,
  rounded corners=northwest ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/sharpish corners (style, no value)

Shortcut for setting `/tcb/arc` → P. 35 and `/tcb/outer arc` → P. 37 to `0pt`. With this setting, rounded corners will appear als quasi-sharp, but e. g. the shadow will be somewhat rounder than the shadow of really sharp corners.

Corners are still of type *rounded* with this option, but appear *sharp*. To switch back to rounded corners, one has to adapt `/tcb/arc` → P. 35 and `/tcb/outer arc` → P. 37.

```
\begin{tcolorbox}[colback=red!5!white,
  colframe=red!75!black,
  sharpish corners ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

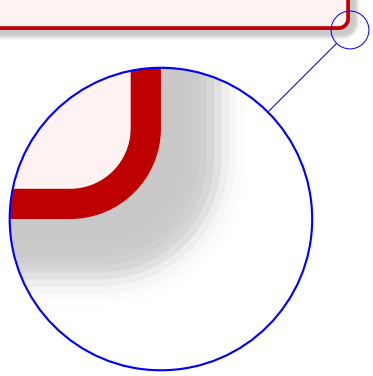
[2] The graphical examples assume that the boxes where set to have sharp corners before.

The following examples will show the differences between `/tcb/rounded corners` → P. 47, `/tcb/sharpish corners` → P. 47, and `/tcb/sharp corners` → P. 46. The later two give the same core box, but `/tcb/borderline` → P. 151 and `/tcb/shadow` → P. 162 settings are slightly different. The following examples use `/tcb/drop fuzzy shadow` → P. 156.

My title

This is a **tcolorbox**.

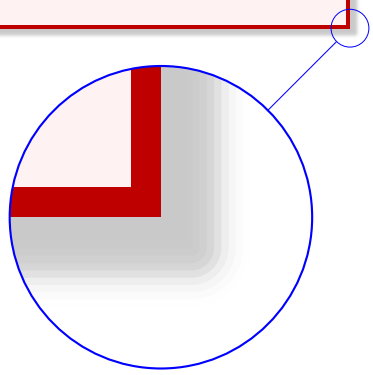
rounded corners



My title

This is a **tcolorbox**.

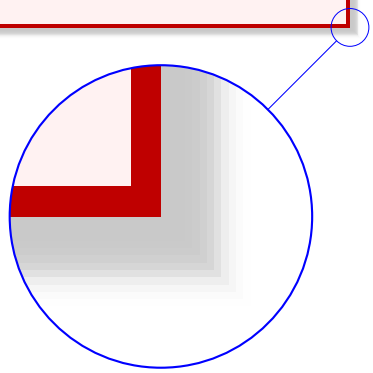
sharpish corners



My title

This is a **tcolorbox**.

sharp corners



4.9 Transparency

Transparency effects are likely to be used in conjunction with *jigsaw* skin variants, see Section 9.10 on page 173.

/tcb/opacityframe=⟨*fraction*⟩ (no default, initially `1.0`)

Sets the frame opacity of the box to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[opacityframe=0.25,
  colframe=red]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/opacityback=⟨*fraction*⟩ (no default, initially `1.0`)

Sets the background opacity of the box to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[standard jigsaw,colframe=red,
  opacityframe=0.5, opacityback=0.5]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/opacitybacktitle=⟨*fraction*⟩ (no default, initially `1.0`)

Sets the title background opacity of the box to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[standard jigsaw,colframe=red,
  opacityframe=0.5, opacitybacktitle=0.5,
  title filled, title=This is a title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a title

This is a **tcolorbox**.

/tcb/opacityfill=⟨*fraction*⟩ (style, no default, initially `1.0`)

Sets the fill opacity for frame, interior and optionally the title background to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[standard jigsaw,colframe=red,
  opacityfill=0.7, title=This is a title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a title

This is a **tcolorbox**.

/tcb/opacityupper=⟨*fraction*⟩ (no default, initially 1.0)

Sets the text opacity of the upper box part to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[enhanced,opacityupper=0.5,
  interior style={pattern=checkerboard light ⤸
      ⤹gray}]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/opacitylower=⟨*fraction*⟩ (no default, initially 1.0)

Sets the text opacity of the lower box part to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[enhanced,opacitylower=0.5,
  interior style={pattern=checkerboard light ⤸
      ⤹gray}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/opacitytext=⟨*fraction*⟩ (no default, initially 1.0)

Sets the text opacity of the upper and the lower box part to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[enhanced,opacitytext=0.5,
  interior style={pattern=checkerboard light ⤸
      ⤹gray}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/opacitytitle=⟨*fraction*⟩ (no default, initially 1.0)

Sets the text opacity of the box title to the given ⟨*fraction*⟩.

```
\begin{tcolorbox}[enhanced,opacitytitle=0.7,
  coltitle=black,
  fonttitle=\bfseries,title=This is a title,
  title style={pattern=checkerboard light ⤸
      ⤹gray}]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a title

This is a **tcolorbox**.

```
\begin{tcolorbox}[enhanced jigsaw,fonttitle=\bfseries,title=This is a title,
  opacityframe=0.5,opacityback=0.25,opacitybacktitle=0.25,opacitytext=0.8,
  colback=red!5!white,colframe=red!75!black,colbacktitle=yellow!20!red]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a title

This is a **tcolorbox**.

4.10 Height Control

In a typical usage scenario, the height of a `tcolorbox` is computed automatically to fit the content. Nevertheless, the height can be set to a fixed value or to fit commonly for several boxes, e. g. if boxes are set side by side.

The height control keys are only applicable to unbreakable boxes. If a box is set to be `/tcb/breakable` → P. 306, the height is always computed according to the *natural height*.

`/tcb/natural height` (no value, initially set)

Sets the total height of the colored box to its natural height depending on the box content.

`/tcb/height`=⟨*length*⟩ (no default)

Sets the total height of the colored box to ⟨*length*⟩ independent of the box content. ⟨*length*⟩ is the minimum height of the box, if `/tcb/height plus` is larger than zero.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white}

\begin{tcolorbox}[height=1cm,valign=center]
  This box has a height of 1cm.
\end{tcolorbox}
\begin{tcolorbox}[height=2cm,valign=center]
  This box has a height of 2cm.
\end{tcolorbox}
\begin{tcolorbox}[height=3cm,split=0.5,valign=center,valign lower=center]
  This box has a height of 3cm.
  \tcblower
  Lower part.
\end{tcolorbox}
```

This box has a height of 1cm.

This box has a height of 2cm.

This box has a height of 3cm.

Lower part.

`/tcb/height plus`=⟨*length*⟩ (no default, initially `0pt`)

The box may extend a given fixed `/tcb/height` up to the given ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,width=3cm,nobeforeafter}

\begin{tcolorbox}[height=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[height=1cm,height plus=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[height=1cm,height plus=1cm]
This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

This is a tcolorbox.

This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.

/tcb/height from=⟨*min*⟩ to ⟨*max*⟩ (style, no default)

Sets the box height to a dimension between ⟨*min*⟩ and ⟨*max*⟩.

```
% \usepackage{lipsum}
\newtcolorbox{mybox}{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,
  bottom=1mm,right=1mm,boxsep=0mm,width=4.5cm,nobeforeafter,
  height from=2cm to 8cm}

\begin{mybox}
This is a tcolorbox.
\end{mybox}
\begin{mybox}
This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.
\end{mybox}
\begin{mybox}
\lipsum[2]
\end{mybox}
```

This is a tcolorbox.

This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.

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.

N 2014-10-31 **/tcb/text height=⟨*length*⟩** (style, no default)

Sets the text height to ⟨*length*⟩. This is the length from the top of the upper part to the bottom of the optional lower part. See also **/tcb/text width** → P. 33.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[text height=2cm]
This is a \textbf{tcolorbox} where the text area has a height of 2cm.
\end{tcolorbox}
```

This is a **tcolorbox** where the text area has a height of 2cm.

N 2014-11-07 **/tcb/add to height=⟨*length*⟩** (style, no default)

Adds ⟨*length*⟩ to the current height of the colored box. /tcb/height → P. 51 has to be set before this key is used!

```
\tcbset{height=1cm,
  valign=center,width=(\linewidth-2mm)/2,
  before=,after=\hfill,colframe=blue!75!black,colback=white}

\begin{tcolorbox}
  This box has a height of 1cm.
\end{tcolorbox}
\begin{tcolorbox}[add to height=1cm]
  This box has a height of 2cm.
\end{tcolorbox}
```

This box has a height of 1cm.

This box has a height of 2cm.

N 2014-09-22 U 2014-12-15 **/tcb/height fill=true|false|maximum** (default `true`, initially `false`)

If set to `true`, the height of the `tcolorbox` is set to the rest of the available vertical space of the current page. If set to `maximum`, the page is compressed as much as possible. Note that the `tcolorbox` is always set as its own paragraph using this option. Also see /tcb/text fill → P. 60.

Note that the library breakable has to be loaded to use this key!

This height control key is only applicable to unbreakable boxes but it uses code from the library breakable. The counterpart for breakable boxes is /tcb/height fixed for → P. 311.

```
% \usepackage{lipsum}
% \tcbuselibrary{breakable}
\begin{tcolorbox}[height fill,
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Box which fills the rest of the page]
\lipsum[1]
\end{tcolorbox}
```

Box which fills the rest of the page

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

N 2015-05-05 **/tcb/square** (style, no value)

Sets /tcb/height → P. 51 to match the width of the colored box.

```
\begin{tcolorbox}[width=3cm,
  colback=red!5!white,
  colframe=red!75!black,
  halign=center,valign=center,
  square]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/space=⟨*fraction*⟩ (no default, initially 0)

If the height of a `tcolorbox` is not the natural height, the space difference between the forced and the natural size is distributed between the upper and the lower part of the box. This space could also be negative. ⟨*fraction*⟩ with a value between 0 and 1 is the amount of space which is added to the upper part, the rest is added to the lower part. If there is no lower part, then all of the space is added to the upper part always.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white,height=3cm}

\foreach \f in {0.2,0.4,0.7}
{\begin{tcolorbox}[space=\f]
    This is the upper part.
    \tcblower
    This is the lower part.
\end{tcolorbox}}
```

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.

/tcb/space to upper (style)

This is an abbreviation for `space=1`, i. e. all extra space is added to the upper part.

/tcb/space to lower (style, initially set)

This is an abbreviation for `space=0`, i. e. all extra space is added to the lower part (if there is any).

/tcb/space to both (style)

This is an abbreviation for `space=0.5`, i. e. the extra space equally distributed between the upper and the lower part.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white,height=3cm}

\foreach \myspace in {space to upper,space to both,space to lower}
{\begin{tcolorbox}[\myspace]
    This is the upper part.
    \tcblower
    This is the lower part.
\end{tcolorbox}}
```

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.

/tcb/split=⟨*fraction*⟩ (no default)

If the height of a `tcolorbox` is not the natural height, the ⟨*fraction*⟩ with a value between 0 and 1 determines the positioning of the segmentation between the upper and the lower part. Here, 0 stands for top and 1 for bottom. Note that the box is split regardless of the actual dimensions of the text parts!

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,height=3cm,
colback=white,colframe=blue!75!black,valign=center,valign lower=center}

\foreach \f in {0.1,0.5,0.8}
{\begin{tcolorbox}[split=\f]
This is the upper part.
\tcblower
This is the lower part with a lot of text in several lines.
\end{tcolorbox}}
```

This is the upper part.

This is the lower part with a lot of text in several lines.

This is the upper part.

This is the lower part with a lot of text in several lines.

This is the upper part.

This is the lower part with a lot of text in several lines.

U 2014-11-07 **/tcb/equal height group=⟨*id*⟩** (no default)

Boxes which are members of an **equal height group** will all get the same height, i. e. the maximum of all their natural heights. The ⟨*id*⟩ serves to distinguish between different height groups. Note that you have to compile twice to see changes and that height groups are global definitions.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,arc=0mm,
colframe=blue!75!black,colback=white,fonttitle=\bfseries}

\begin{tcolorbox}[equal height group=A,adjusted title={One}]
  My smallest box.
\end{tcolorbox}%
\begin{tcolorbox}[equal height group=A,adjusted title={Two}]
  This box is also small.
  \tcblower
  But with a lower part.
\end{tcolorbox}%
\begin{tcolorbox}[equal height group=A,adjusted title={Three}]
  This box contains a lot of text just to fill the space
  with word flowing and flowing and flowing until the box
  is filled with all of it.
\end{tcolorbox}\linebreak
%
\tcbset{width=(\linewidth-1mm)/2,before=,after=\hfill,arc=0mm,
colframe=red!75!black,colback=white}
%
\begin{tcolorbox}[equal height group=B]
  Now, we use another equal height group.
\end{tcolorbox}%
\begin{tcolorbox}[equal height group=B]
  \begin{equation*}
    \int\limits_{0}^{1} x^2 = \frac13.
  \end{equation*}
\end{tcolorbox}
```

One

My smallest box.

Two

This box is also small.

But with a lower part.

Three

This box contains a lot of text just to fill the space with word flowing and flowing and flowing until the box is filled with all of it.

Now, we use another equal height group.

$$\int_0^1 x^2 = \frac{1}{3}.$$

See Section 12 on page 231 for more equal height options.

/tcb/minimum for equal height group=⟨*id*⟩:⟨*length*⟩ (no default, initially unset)

Plants a ⟨*length*⟩ into the equal height group with the given ⟨*id*⟩. This ensures that the height will not drop below ⟨*length*⟩. Note that you cannot reduce a computed height value by using this key with a small value. The difference to applying /tcb/height → P. 51 directly is that the boxes are never too small for their content.

```
\tcbset{colframe=blue!75!black,colback=white,arc=0mm,
  before=,after=\hfill,fonttitle=\bfseries,left=2mm,right=2mm,
  width=3.5cm,
  equal height group=C,
  minimum for equal height group=C:3.5cm}

\begin{tcolorbox}
  My first box. All boxes will get 3.5cm times 3.5cm
  if the content height is not too large.
\end{tcolorbox}%
\begin{tcolorbox}
  My second box.
  \tcblower
  This is the lower part.
\end{tcolorbox}%
\begin{tcblisting}{}
\textbf{Mixed}
with a listing.
\end{tcblisting}
\begin{tcolorbox}[title={Fourth box}]
  My final box.
\end{tcolorbox}%
```

My first box. All boxes will get 3.5cm times 3.5cm if the content height is not too large.

My second box.

This is the lower part.

```
\textbf{Mixed}
with a listing.
```

Mixed with a listing.

Fourth box

My final box.

4.11 Box Content Additions

The following options introduce some arbitrary ⟨*code*⟩ to the content of a `tcolorbox`. These additions can be given at the beginning or at the ending of the title, the upper part, or the lower part.

/tcb/before title=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the color and font settings and *before* the content of the title.

```
\tcbset{before title={\textcolor{yellow}{\large Important:}~},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Important: **My title**

This is a **tcolorbox**.

/tcb/after title=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the content of the title.

```
\tcbset{after title={\hfill\colorbox{Navy}{approved}},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title **approved**

This is a **tcolorbox**.

/tcb/before upper=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the color and font settings and *before* the content of the upper part.

```
\tcbset{before upper={\textit{The story:}\par},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

The story:
This is a **tcolorbox**.

/tcb/after upper=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the content of the upper part.

```
\tcbset{after upper={\par\hfill\textit{Read more next week}},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

Read more next week

/tcb/before lower=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the color and font settings and *before* the content of the lower part.

```
\tcbset{before lower=\textit{Behold:~},colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

Behold: This is the lower part.

/tcb/after lower=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the content of the lower part.

```
\tcbset{after lower=\ \textit{This is the end.},
  colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part. *This is the end.*

If /tcb/text fill is used, one cannot have a lower part and the box is unbreakable.

N 2015-07-15 **/tcb/text fill** (style, no value)

This style sets /tcb/before upper→P. 58 and /tcb/after upper→P. 59 to embedd the upper part with a minipage. If a fixed height was applied e.g. by /tcb/height→P. 51 or /tcb/height fill→P. 53, this minipage gets a matching height. This allows to use vertical glue macros like \vfill to act like expected. If the box has no fixed height, setting /tcb/text fill has no other effect as making the box unbreakable.

```
\begin{tcolorbox}[colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  height=8cm,text fill,
  title=My filled box]
This is a \textbf{tcolorbox}.
\par\vfill
\begin{center}
  My middle text.
\end{center}
\par\vfill
This is the end of my box.
\end{tcolorbox}
```

My filled box

This is a **tcolorbox**.

My middle text.

This is the end of my box.

If /tcb/tabularx or /tcb/tabularx* are used, one cannot have a lower part.

/tcb/tabularx=⟨*preamble*⟩ (style)

This style sets /tcb/before upper → P. 58 and /tcb/after upper → P. 59 and several geometry keys to support a tabularx with the given ⟨*preamble*⟩. The packages tabularx [4], array, and colortbl have to be loaded separately.

```
% \usepackage{array,tabularx}
% \usepackage{colortbl} - or - \usepackage[table]{xcolor}
\newcolumntype{Y}{>{\raggedleft\arraybackslash}X}% see tabularx
\tcbset{enhanced,fonttitle=\bfseries\large,fontupper=\normalsize\sffamily,
  colback=yellow!10!white,colframe=red!50!black,colbacktitle=Salmon!30!white,
  coltitle=black,center title}

\begin{tcolorbox}[tabularx={X||Y|Y|Y|Y||Y},title=My table]
Group & One      & Two      & Three     & Four      & Sum\\\hline\hline
Red   & 1000.00 & 2000.00 &  3000.00 &  4000.00 & 10000.00\\\hline
Green & 2000.00 & 3000.00 &  4000.00 &  5000.00 & 14000.00\\\hline
Blue  & 3000.00 & 4000.00 &  5000.00 &  6000.00 & 18000.00\\\hline\hline
Sum   & 6000.00 & 9000.00 & 12000.00 & 15000.00 & 42000.00
\end{tcolorbox}
```

My table

Group	One	Two	Three	Four	Sum
Red	1000.00	2000.00	3000.00	4000.00	10000.00
Green	2000.00	3000.00	4000.00	5000.00	14000.00
Blue	3000.00	4000.00	5000.00	6000.00	18000.00
Sum	6000.00	9000.00	12000.00	15000.00	42000.00

/tcb/tabularx*={⟨*code*⟩}{⟨*preamble*⟩} (style)

This is a variant of /tcb/tabularx which adds some ⟨*code*⟩ before the table starts.

```
% \usepackage{array,tabularx}
% \usepackage{colortbl} - or - \usepackage[table]{xcolor}
\tcbset{enhanced,fonttitle=\bfseries\large,fontupper=\normalsize\sffamily,
  colback=yellow!10!white,colframe=red!50!black,colbacktitle=Salmon!30!white,
  coltitle=black,center title}

\begin{tcolorbox}[tabularx*={\arrayrulewidth0.5mm}{X|X|X},title=My table]
One      & Two      & Three \\\hline\hline
1000.00 & 2000.00 &  3000.00\\\hline
2000.00 & 3000.00 &  4000.00
\end{tcolorbox}
```

My table

One	Two	Three
1000.00	2000.00	3000.00
2000.00	3000.00	4000.00

/tcb/tikz upper=⟨*options*⟩ (style)

This style adds a centered `tikzpicture` environment to the start and end of the upper part. The ⟨*options*⟩ may be given as TikZ picture options.

```
% \usepackage{tikz}

\begin{tcolorbox}[tikz upper,fonttitle=\bfseries,colback=white,colframe=black,
                  title=\tikzname\ drawing]
  \path[fill=yellow,draw=yellow!75!red] (0,0) circle (1cm);
  \fill[red] (45:5mm) circle (1mm);
  \fill[red] (135:5mm) circle (1mm);
  \draw[line width=1mm,red] (215:5mm) arc (215:325:5mm);
\end{tcolorbox}
```

TikZ drawing

/tcb/tikz lower=⟨*options*⟩ (style)

This style adds a centered `tikzpicture` environment to the start and end of the lower part. The ⟨*options*⟩ may be given as TikZ picture options.

```
% \usepackage{tikz}
% \tcbuselibrary{skins,listings}
\tcbset{tikz lower,listing side text,fonttitle=\bfseries,
  bicolor,colback=LightBlue!50!white,colbacklower=white,colframe=black,
  righthand width=3cm}

\begin{tcblisting}{title=\tikzname\  drawing}
\path[fill=yellow,draw=yellow!75!red]
    (0,0) circle (1cm);
\fill[red] (45:5mm) circle (1mm);
\fill[red] (135:5mm) circle (1mm);
\draw[line width=1mm,red]
    (215:5mm) arc (215:325:5mm);
\end{tcblisting}
```

TikZ drawing

```
\path[fill=yellow,draw=yellow!75!red]
    (0,0) circle (1cm);
\fill[red] (45:5mm) circle (1mm);
\fill[red] (135:5mm) circle (1mm);
\draw[line width=1mm,red]
    (215:5mm) arc (215:325:5mm);
```

/tcb/tikznode upper=⟨*options*⟩ (style)

This style places the upper part content into a centered TikZ node. The ⟨*options*⟩ may be given as TikZ node options. This style is especially useful for boxes with multiline texts which are fitted to the text width.

```
% \usepackage{tikz}
\newtcbox{\headline}[1][]{enhanced,before=\begin{center},after=\end{center},
  ignore nobreak,fontupper=\Large\bfseries,
  colframe=red!50!black,colback=red!10!white,
  drop fuzzy shadow=yellow,tikznode upper,#1}

\headline{Important\\Headline}
```

Important Headline

/tcb/tikznode lower=⟨*options*⟩ (style)

This style places the lower part content into a centered TikZ node. The ⟨*options*⟩ may be given as TikZ node options.

```
% \usepackage{tikz}
\begin{tcolorbox}[bicolor,colback=LightBlue!50!white,colbacklower=white,
  colframe=black,tikznode lower={inner sep=2pt,draw=red,fill=yellow}]
Upper part.
\tcblower
Lower part.
\end{tcolorbox}
```

Upper part.

Lower part.

/tcb/tikznode=⟨*options*⟩ (style)

Shortcut for setting `/tcb/tikznode upper` and `/tcb/tikznode lower` the same time.

/tcb/varwidth upper=⟨*length*⟩ (style, default `/tcb/width`→ P. 33)

This style places the upper part content into a `varwidth` environment. This style needs the `varwidth` package [1] to be loaded manually. The resulting box has a maximal width of ⟨*length*⟩. This option is only senseful for a `\tcbox`→ P. 13.

```
% \usepackage{varwidth}
\newtcbox{\varbox}{colframe=red!50!black,
  colback=red!10!white,varwidth upper}

\varbox{Short text.}
\varbox{This box contains is a longer text
  which is broken.}
```

Short text.

This box contains is a longer text which is broken.

4.12 Overlays

With an overlay, arbitrary ⟨*graphical code*⟩ can be added to a `tcolorbox`. This code is executed *after* the frame and interior are drawn and *before* the text content is drawn. Therefore, you can decorate the `tcolorbox` with your own extensions. Common special cases are *watermarks* which are implemented using overlays. See Subsection 9.3 from page 139 if you want to add *watermarks*.

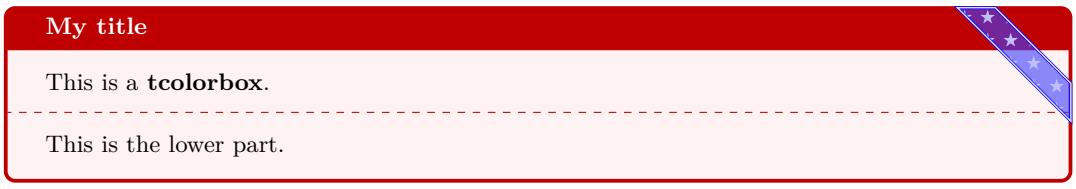
If you use the core package only, the ⟨*graphical code*⟩ has to be `pgf` code and there is not much assistance for positioning. Therefore, the usage of the `/tcb/enhanced` → P. 178 mode from the library skins is recommended which allows `tikz` code and gives access to `/tcb/geometry nodes` → P. 115 for positioning.

`/tcb/overlay=`⟨*graphical code*⟩ (no default, initially unset)

Adds ⟨*graphical code*⟩ to the box drawing process. This ⟨*graphical code*⟩ is drawn *after* the frame and interior and *before* the text content.

```
% \tcbuselibrary{skins} % preamble
\tcbset{frogbox/.style={enhanced,colback=green!10,colframe=green!65!black,
  enlarge top by=5.5mm,
  overlay={\foreach \x in {2cm,3.5cm} {
    \begin{scope}[shift={([xshift=\x]frame.north west)}]
      \path[draw=green!65!black,fill=green!10,line width=1mm] (0,0) arc (0:180:5mm);
      \path[fill=black] (-0.2,0) arc (0:180:1mm);
    \end{scope}}}}}

\begin{tcolorbox}[frogbox,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

```
% \usetikzlibrary{patterns} % preamble
% \tcbuselibrary{skins}     % preamble
\tcbset{ribbonbox/.style={enhanced,colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries,
  overlay={\path[fill=blue!75!white,draw=blue,double=white!85!blue,
    preaction={opacity=0.6,fill=blue!75!white},
    line width=0.1mm,double distance=0.2mm,
    pattern=fivepointed stars,pattern color=white!75!blue]
    ([xshift=-0.2mm,yshift=-1.02cm]frame.north east)
    -- ++(-1,1) -- ++(-0.5,0) -- ++(1.5,-1.5) -- cycle;}}}

\begin{tcolorbox}[ribbonbox,title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/no overlay (style, no default, initially set)

Removes the overlay if set before.

/tcb/overlay broken=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay unbroken=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 but *is not* broken actually or if the box is set to be /tcb/unbreakable→P. 307, then the ⟨*graphical code*⟩ is added to the box drawing process. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay first=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *first* part of the break sequence. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay middle=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *middle* parts (if any) of the break sequence. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay last=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *last* part of the break sequence. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay unbroken and first=⟨*graphical code*⟩ (no default, initially unset)

This is an optimized abbreviation for setting /tcb/overlay unbroken and /tcb/overlay first together. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay middle and last=⟨*graphical code*⟩ (no default, initially unset)

This is an optimized abbreviation for setting /tcb/overlay middle and /tcb/overlay last together. /tcb/overlay→P. 64 overwrites this key.

/tcb/overlay unbroken and last=⟨*graphical code*⟩ (no default, initially unset)

This is an optimized abbreviation for setting /tcb/overlay unbroken and /tcb/overlay last together. /tcb/overlay→P. 64 overwrites this key.

N 2014-09-19 **/tcb/overlay first and middle=**⟨*graphical code*⟩ (no default, initially unset)

This is an optimized abbreviation for setting /tcb/overlay first and /tcb/overlay middle together. /tcb/overlay→P. 64 overwrites this key.

This example demonstrates the application of break sequence specific overlay options. Here, we define an environment `myexample` based on `tcolorbox` where the visible drawing is done totally by overlay keys.
Here, the first application of `myexample` produces an unbroken `tcolorbox`. The frame is drawn by the code given with /tcb/overlay unbroken.
The second application of `myexample` is broken into several parts which are drawn by the codes given with /tcb/overlay first, /tcb/overlay middle, and /tcb/overlay last.

```
% Preamble:
%\usepackage{tikz,lipsum}
%\tcbuselibrary{skins,breakable}
%\newcounter{example}
\colorlet{colexam}{red!75!black}
\newtcolorbox[use counter=example]{myexample}{%
```

```
  empty,title={Example \thetcbcounter},attach boxed title to top left,
  boxed title style={empty,size=minimal,toprule=2pt,top=4pt,
    overlay={\draw[colexam,line width=2pt]
      ([yshift=-1pt]frame.north west)--([yshift=-1pt]frame.north east);}},
  coltitle=colexam,fonttitle=\Large\bfseries,
  before=\par\medskip\noindent,parbox=false,boxsep=0pt,left=0pt,right=3mm,top=4pt,
  breakable,pad at break*=0mm,vfill before first,
  overlay unbroken={\draw[colexam,line width=1pt]
    ([yshift=-1pt]title.north east)--([xshift=-0.5pt,yshift=-1pt]title.north-|frame.east)
    --([xshift=-0.5pt]frame.south east)--(frame.south west); },
  overlay first={\draw[colexam,line width=1pt]
    ([yshift=-1pt]title.north east)--([xshift=-0.5pt,yshift=-1pt]title.north-|frame.east)
    --([xshift=-0.5pt]frame.south east); },
  overlay middle={\draw[colexam,line width=1pt] ([xshift=-0.5pt]frame.north east)
    --([xshift=-0.5pt]frame.south east); },
  overlay last={\draw[colexam,line width=1pt] ([xshift=-0.5pt]frame.north east)
    --([xshift=-0.5pt]frame.south east)--(frame.south west);},%
}

\begin{myexample}
\lipsum[1]
\end{myexample}

\begin{myexample}
\lipsum[2-11]
\end{myexample}

\lipsum[12]% following text
```

Example 1

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Example 2

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent

euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in,

sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Floating box from `floatplacement`

This floating box is placed at the top of a page.

4.13 Floating Objects

`/tcb/floatplacement=`⟨*values*⟩ (no default, initially `htb`)

Sets ⟨*values*⟩ as default values for the usage of `/tcb/float` and `/tcb/float*`. Feasible are the usual parameters for floating objects.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,
   watermark color=red!15!white}

\begin{tcolorbox}[floatplacement=t,float,
                  title=Floating box from |floatplacement|,
                  watermark text={I am floating}]
  This floating box is placed at the top of a page.
\end{tcolorbox}
```

`/tcb/float=`⟨*values*⟩ (default from `floatplacement`)

Turns the box to a floating object where ⟨*values*⟩ are the usual parameters for such floating objects. If they are not used, the placement uses the default values given by `floatplacement`.

```
\begin{tcolorbox}[float, title=Floating box from |float|,
    enhanced,watermark text={I'm also floating}]
  This box floats to a feasible place automatically. You do not have to
  use a numbering for this floating object.
\end{tcolorbox}
```

Floating box from `float`

This box floats to a feasible place automatically. You do not have to use a numbering for this floating object.

`/tcb/float*=`⟨*values*⟩ (default from `floatplacement`)

Identical to `/tcb/float`, but for wide boxes spanning the whole page width of two column documents or in conjunction with the packages `multicol` or `paracol`. Note that you have to set `width=\textwidth` additionally, if the box should span the whole page width in these cases!

```
\begin{tcolorbox}[float*=b, title=Floating box from |float*|,width=\textwidth,
    enhanced,watermark text={I'm also floating}]
  In this single column document, you will see no difference to |float|.
\end{tcolorbox}
```

`/tcb/nofloat` (style, initially set)

Turns the floating behavior off.

N 2014-09-19 `/tcb/every float=`⟨*code*⟩ (no default, initially empty)

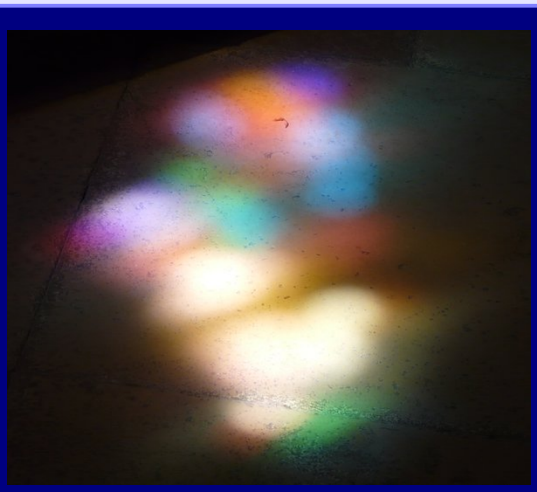
Floating box from `float*`

In this single column document, you will see no difference to `float`.

For floating objects, the `/tcb/before` → P. 76 and `/tcb/after` → P. 76 settings are ignored. Instead, the given ⟨*code*⟩ is inserted before a floating box. If the box is `/tcb/breakable` → P. 306, the given ⟨*code*⟩ is inserted before every part of the break sequence. The most common use case is `every float=\centering`.

```
\tcbox[float=htb,title={Floating box},every float=\centering,
  colback=blue!50!black,colframe=blue!50!white,colbacktitle=blue!10!white,
  coltitle=black,center title]
  {\includegraphics[height=6cm]{lichtspiel.jpg}}
```

Floating box



4.14 Side by Side

Further side by side options for code examples are `/tcb/listing side text` → P. 258, `/tcb/text side listing` → P. 258, `/tcb/listing outside text` → P. 258, and `/tcb/text outside listing` → P. 258.

`/tcb/sidebyside=true|false` (default `true`, initially `false`)

Normally, the upper part and the lower part of the box have their positions as their names suggest. If `sidebyside` is set to `true`, the upper part is drawn *left-handed* and the lower part is drawn *right-handed*. Both parts are drawn together with the geometry settings of the upper part but the space is divided horizontally according to the following options. Colors, fonts, and box content additions are used individually. The resulting box is unbreakable.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside]
This is the upper (\textit{left-handed}) part.
\tcblower
This is the lower (\textit{right-handed}) part.
\end{tcolorbox}
```

My title

This is the upper (*left-handed*) part. | This is the lower (*right-handed*) part.

U 2015-02-06 `/tcb/sidebyside align=`⟨*alignment*⟩ (no default, initially `center`)

Sets the vertical ⟨*alignment*⟩ for the left-handed and right-handed part. Feasible values are `center`, `top`, `bottom`, `center seam`, `top seam`, and `bottom seam`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,nobeforeafter,
  left=2mm,right=2mm,sidebyside,sidebyside gap=6mm,width=(\linewidth-2mm)/3}

\begin{tcolorbox}[adjusted title=center,sidebyside align=center]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=top,sidebyside align=top]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=bottom,sidebyside align=bottom]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}
```

center

This is a text which is too long for one line. | This is a short text.

top

This is a text which is too long for one line. | This is a short text.

bottom

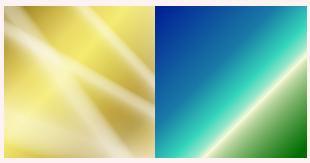
This is a text which is too long for one line. | This is a short text.

center, **top**, and **bottom** are identical to the known corresponding `minipage` options. For example, **top** aligns the top lines of the lefthand and righthand side according to their baselines. While this is the preferred approach for text content, the result for boxed content like tables or images may not be as expected.
For such content, one may use **center seam**, **top seam**, and **bottom seam**. For example, **top seam** aligns the very top seam of the lefthand and righthand side.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  size=small,righthand width=4cm,sidebyside,sidebyside gap=6mm,lower separated=false}

\begin{tcolorbox}[adjusted title=center seam,sidebyside align=center seam]
  This is my description text for the pictures displayed on the righthand side.
  \tcblower
  \includegraphics[width=\linewidth/2]{goldshade}%
  \includegraphics[width=\linewidth/2]{blueshade}
\end{tcolorbox}

\begin{tcolorbox}[adjusted title=top seam,sidebyside align=top seam]
  This is my description text for the pictures displayed on the righthand side.
  \tcblower
  \includegraphics[width=\linewidth/2]{goldshade}%
  \includegraphics[width=\linewidth/2]{blueshade}
\end{tcolorbox}

\begin{tcolorbox}[adjusted title=bottom seam,sidebyside align=bottom seam]
  This is my description text for the pictures displayed on the righthand side.
  \tcblower
  \includegraphics[width=\linewidth/2]{goldshade}%
  \includegraphics[width=\linewidth/2]{blueshade}
\end{tcolorbox}
```

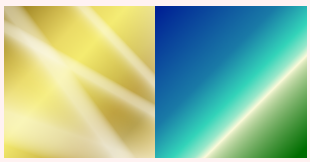
center seam

This is my description text for the pictures displayed on the righthand side.



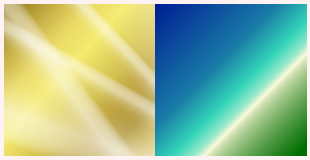
top seam

This is my description text for the pictures displayed on the righthand side.



bottom seam

This is my description text for the pictures displayed on the righthand side.



/tcb/sidebyside gap=⟨*length*⟩ (no default, initially 10mm)

Sets the horizontal distance between the left-handed and right-handed part to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,nobeforeafter,
  sidebyside,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[adjusted title=Wide gap,sidebyside gap=30mm]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=Narrow gap,sidebyside gap=1mm]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}
```

Wide gap

This is a text which is too long for one line.

This is a short text.

Narrow gap

This is a text which is too long for one line.

This is a short text.

/tcb/lefthand width=⟨*length*⟩ (no default, initially unset)

Sets the width of the left-handed part to the given ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,lefthand width=3cm]
This is the upper (\textit{left-handed}) part.
\tcblower
This is the lower (\textit{right-handed}) part.
\end{tcolorbox}
```

My title

This is the upper (*left-handed*) part.

This is the lower (*right-handed*) part.

/tcb/righthand width=⟨*length*⟩ (no default, initially unset)

Sets the width of the right-handed part to the given ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,righthand width=3cm]
This is the upper (\textit{left-handed}) part.
\tcblower
This is the lower (\textit{right-handed}) part.
\end{tcolorbox}
```

My title

This is the upper (*left-handed*) part.

This is the lower (*right-handed*) part.

/tcb/lefthand ratio=⟨*fraction*⟩ (no default, initially 0.5)

Sets the width of the left-handed part to the given ⟨*fraction*⟩ of the available space. ⟨*fraction*⟩ is a value between 0 and 1.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,lefthand ratio=0.25]
This is the upper (\textit{left-handed}) part.
\tcblower
This is the lower (\textit{right-handed}) part.
\end{tcolorbox}
```

My title

This is the upper (*left-handed*) part.

This is the lower (*right-handed*) part.

/tcb/righthand ratio=⟨*fraction*⟩ (no default, initially 0.5)

Sets the width of the right-handed part to the given ⟨*fraction*⟩ of the available space. ⟨*fraction*⟩ is a value between 0 and 1.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,righthand ratio=0.25]
This is the upper (\textit{left-handed}) part.
\tcblower
This is the lower (\textit{right-handed}) part.
\end{tcolorbox}
```

My title

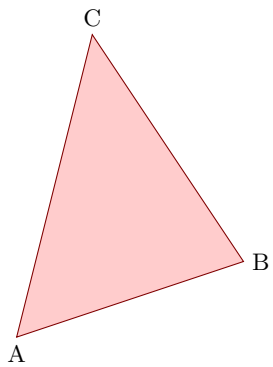
This is the upper (*left-handed*) part.

This is the lower (*right-handed*) part.

If one side of a side-by-side box should be adapted to the width of its content, this width has to be computed beforehand. The following example uses a savebox `\mysavebox` to store the picture to determine its width.

```
% \tcbuselibrary{skins,xparse}
% \usepackage{lipsum}
% \newsavebox\mysavebox  % preamble
\DeclareTotalTColorBox{\mysidebox}{ O{} +m +m }{
  bicolor,colback=white,colbacklower=yellow!10,
  fonttitle=\bfseries,center title,
  sidebyside,
  code={\sbox{\mysavebox}{#2}},
  lefthand width=\wd\mysavebox,
  drop lifted shadow,
  #1
}
{\usebox{\mysavebox}\tcblower#3}

\mysidebox[title=The Triangle]{%
  \begin{tikzpicture}
    \path[fill=red!20,draw=red!50!black]
      (0,0) node[below]{A} -- (3,1) node[right]{B}
      -- (1,4) node[above]{C} -- cycle;
  \end{tikzpicture}%
}{%
  \lipsum[1]
}
```

The Triangle



Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

4.15 Embedding into the Surroundings

Typically, but not necessarily, a `tcolorbox` is put inside a separate paragraph and has some vertical space before and after it. This behavior is controlled by the keys `before` and `after`.

/tcb/before=⟨*code*⟩ (no default, initially see /tcb/autoparskip)

Sets the ⟨*code*⟩ which is executed before the colored box. It is not used for floating boxes. Also, it is not used, if the box follows a heading immediately and /tcb/ignore nobreak → P. 79 is set to `false`.

/tcb/after=⟨*code*⟩ (no default, initially see /tcb/autoparskip)

Sets the ⟨*code*⟩ which is executed after the colored box. It is not used for floating boxes.

/tcb/parskip (style, no value)

Sets the keys `before` and `after` to values which are recommended, if the package `parskip` *is* used and there is no better idea for `before` and `after`.

```
\tcbset{parskip/.style={before={\par\pagebreak[0]\parindent=0pt},
                        after={\par}}}
```

U 2015-03-16 **/tcb/noparskip** (style, no value)

Sets the keys `before` and `after` to values which are recommended, if the package `parskip` is *not* used and there is no better idea for `before` and `after`.

```
\tcbset{noparskip/.style={before={\par\smallskip\pagebreak[0]\parindent=0pt},
                          after={\par\smallskip}}}
```

/tcb/autoparskip (style, no value, initially set)

Tries to detect the usage of the package `parskip` and sets the keys `before` and `after` accordingly. Actually, the following is done:

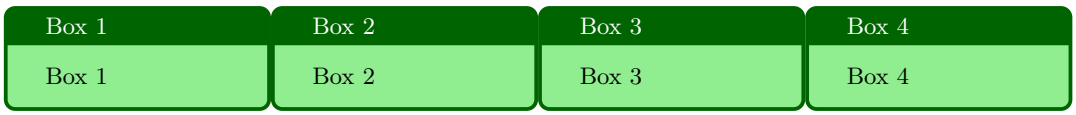
- If the length of `\parskip` is greater than `0pt` at the beginning of the document, /tcb/parskip is executed. Here, the usage of package `parskip` is *assumed*.
- Otherwise, if the length of `\parskip` is not greater than `0pt` at the beginning of the document, /tcb/noparskip is executed. Here, the absence of package `parskip` is *assumed*.

`autoparskip` is the default for the package `tcolorbox`, if `before` or `after` are not changed otherwise.

/tcb/nobeforeafter (style, no value)

Abbreviation for clearing the keys `before` and `after`. The colored box is not put into a paragraph and there is no space before or after the box.

```
\tcbset{myone/.style={colback=LightGreen,colframe=DarkGreen,
  equal height group=nobefaf,width=\linewidth/4,nobeforeafter}}
\begin{tcolorbox}[myone,title=Box 1]Box 1\end{tcolorbox}%
\begin{tcolorbox}[myone,title=Box 2]Box 2\end{tcolorbox}%
\begin{tcolorbox}[myone,title=Box 3]Box 3\end{tcolorbox}%
\begin{tcolorbox}[myone,title=Box 4]Box 4\end{tcolorbox}
```



/tcb/forces nobeforeafter (style, no value)

Forces the setting of /tcb/nobeforeafter even if /tcb/before and /tcb/after are set to other values later. Do not use this option globally unless you *really* know what you do. Note that embedded boxes do not inherit this forced clearance.

/tcb/baseline=⟨*length*⟩ (no default, initially `0pt`)

Used to set the `\pgfsetbaseline` value of the resulting `tcolorbox`.

```
\tcbset{colframe=red!50!white,width=4cm,nobeforeafter}
Some text\dotfill
\begin{tcolorbox}[baseline=3mm]
One line.
\end{tcolorbox}
\begin{tcolorbox}[baseline=3mm]
First line.\\Second line.
\end{tcolorbox}
```

Some text . . . . . One line. First line. Second line.

N 2014-10-10 **/tcb/box align=**⟨*alignment*⟩ (style, no default, initially `bottom`)

Used to set the `/tcb/baseline` value of the resulting `tcolorbox`. Feasible values for ⟨*alignment*⟩ are:

- `bottom`: alignment with the box bottom,
- `top`: alignment with the box top,
- `center`: alignment with the box center,
- `base`: alignment with the box content base. This option is not applicable for a `tcolorbox`→P. 11 but for a `\tcbox`→P. 13 only. It is an alias for `/tcb/tcbox raise base`→P. 91.

```
\tcbset{colframe=red!50!white,width=4cm,nobeforeafter}
Some text\dotfill
\begin{tcolorbox}[box align=bottom]
One line.
\end{tcolorbox}
\begin{tcolorbox}[box align=bottom]
First line.\\Second line.
\end{tcolorbox}
```

Some text . . . . . One line. First line. Second line.

```
\tcbset{colframe=red!50!white,width=4cm,nobeforeafter}
Some text\dotfill
\begin{tcolorbox}[box align=top]
One line.
\end{tcolorbox}
\begin{tcolorbox}[box align=top]
First line.\\Second line.
\end{tcolorbox}
```

Some text . . . . . One line. First line. Second line.

```
\tcbset{colframe=red!50!white,width=4cm,nobeforeafter}
Some text\dotfill
\begin{tcolorbox}[box align=center]
One line.
\end{tcolorbox}
\begin{tcolorbox}[box align=center]
First line.\\Second line.
\end{tcolorbox}
```

Some text . . . . . One line. First line. Second line.

```
\tcbset{colframe=red!50!white,nobeforeafter}
Some text\dotfill
\tcbox[nobeforeafter,box align=base]{One line}
\tcbox[nobeforeafter,box align=base,size=fbox]{Another line}
```

Some text . . . . . One line Another line

N 2014-10-10
U 2015-03-16

/tcb/before skip=⟨*glue*⟩ (style, no default)

Inserts some vertical space of the given ⟨*glue*⟩ before the colored box. This style sets /tcb/before→P. 76.

```
Some text.
\begin{tcolorbox}[before skip=1cm,
    colframe=red!50!white]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Some text.

This is a **tcolorbox**.

N 2014-10-10
U 2015-03-16

/tcb/after skip=⟨*glue*⟩ (style, no default)

Inserts some vertical space of the given ⟨*glue*⟩ after the colored box. This style sets /tcb/after→P. 76.

```
\begin{tcolorbox}[after skip=1cm,
    colframe=red!50!white]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
Some text.
```

This is a **tcolorbox**.

Some text.

N 2014-10-10

/tcb/beforeafter skip=⟨*glue*⟩ (style, no default)

Inserts some vertical space of the given ⟨*glue*⟩ before *and* after the colored box. This style sets /tcb/before→P. 76 and /tcb/after→P. 76.

```
\tcbset{beforeafter skip=0pt,
  colframe=red!50!white}
\begin{tcolorbox}
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}
  Second box.
\end{tcolorbox}
```

This is a **tcolorbox**.

Second box.

N 2014-11-07 **/tcb/left skip=⟨*length*⟩** (style, no default, initially 0mm)

Inserts some horizontal space of the given ⟨*length*⟩ before the colored box. This style sets /tcb/grow to left by → P. 82 with the negated ⟨*length*⟩, i.e. the bounding box and box width are changed.

```
\noindent\rule{\linewidth}{2pt}

\begin{tcolorbox}[left skip=1cm,
    colframe=red!50!white]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-11-07 **/tcb/right skip=⟨*length*⟩** (style, no default, initially 0mm)

Inserts some horizontal space of the given ⟨*length*⟩ after the colored box. This style sets /tcb/grow to right by → P. 82 with the negated ⟨*length*⟩, i.e. the bounding box and box width are changed.

```
\noindent\rule{\linewidth}{2pt}

\begin{tcolorbox}[right skip=1cm,
    colframe=red!50!white]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-10 **/tcb/leftright skip=⟨*length*⟩** (style, no default)

Inserts some horizontal space of the given ⟨*length*⟩ before *and* after the colored box. This style changes the bounding box and the box width.

```
\noindent\rule{\linewidth}{2pt}

\begin{tcolorbox}[leftright skip=1cm,
    colframe=red!50!white]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-12-11 **/tcb/ignore nobreak=true|false** (default true, initially false)

After a heading, LaTeX tries to avoid a break by setting a nobreak boolean value. Starting from version 3.33, the /tcb/before → P. 76 respectively /tcb/before skip → P. 78 settings are not used after a heading if /tcb/ignore nobreak is set to **false**. For an unbreakable box, /tcb/before nobreak is used instead. Further, a /tcb/breakable → P. 306 box will also try to avoid a break between a heading and a directly following first part of a break sequence. Set /tcb/ignore nobreak to **true**, if nobreak should be ignored as prior to version 3.33. Also, such a setting may be used locally to enforce the /tcb/before → P. 76 setting.

N 2014-12-16 **/tcb/before nobreak=⟨*code*⟩** (no default, initially \noindent)

Sets the ⟨*code*⟩ which is executed before the colored box if it is unbreakable, if /tcb/ignore nobreak is not set, and if the box follows a heading.

4.16 Bounding Box

Normally, every `tcolorbox` has a bounding box which fits exactly to the dimensions of the outer frame. Therefore, LaTeX reserves exactly the space needed for the box. This behavior can be changed by enlarging (or shrinking) the bounding box. If the bounding box is enlarged, the `tcolorbox` will get some clearance around it. If the bounding box is shrunk, i. e. enlarged with negative values, the `tcolorbox` will overlap to other parts of the page. For example, the `tcolorbox` could be stretched into the page margin.

The following examples use `/tcb/show bounding box` → P. 153 to display the actual bounding box. For this, the library LIB skins has to be included and `/tcb/enhanced` → P. 178 has to be set.

`/tcb/enlarge top initially by=`⟨*length*⟩ (no default, initially `0mm`)

Enlarges the bounding box distance to the top of the box by ⟨*length*⟩. If the box is *breakable*, only the first box of the break sequence gets enlarged. `/tcb/enlarge top by` → P. 81 overwrites this key.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge top initially by=-5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge top initially by=5mm,enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

`/tcb/enlarge bottom finally by=`⟨*length*⟩ (no default, initially `0mm`)

Enlarges the bounding box distance to the bottom of the box by ⟨*length*⟩. If the box is *breakable*, only the last box of the break sequence gets enlarged. `/tcb/enlarge bottom by` → P. 81 overwrites this key.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge bottom finally by=5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge bottom finally by=-5mm,enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/enlarge top at break by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to the top of the box by ⟨*length*⟩, *if* the box is /tcb/breakable → P. 306. In this case, it is applied to *middle* and *last* parts in a break sequence. /tcb/enlarge top by overwrites this key.

/tcb/enlarge bottom at break by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to the bottom of the box by ⟨*length*⟩, *if* the box is /tcb/breakable → P. 306. In this case, it is applied to *first* and *middle* parts in a break sequence. /tcb/enlarge bottom by overwrites this key.

/tcb/enlarge top by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to the top of the box by ⟨*length*⟩. /tcb/enlarge top initially by → P. 80 and /tcb/enlarge top at break by are set to ⟨*length*⟩.

/tcb/enlarge bottom by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to the bottom of the box by ⟨*length*⟩. /tcb/enlarge bottom finally by → P. 80 and /tcb/enlarge bottom at break by are set to ⟨*length*⟩.

/tcb/enlarge left by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to the left side of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge left by=2cm,width=5cm,enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge left by=-2cm,width=\linewidth+2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/enlarge right by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to the right side of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge right by=-2cm,width=\linewidth+2cm,
  enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge right by=2cm,width=\linewidth-2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/enlarge by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the bounding box distance to all sides of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white,width=5cm,nobeforeafter}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge by=5mm,enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/grow to left by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the current box width by ⟨*length*⟩ and enlarges (shrinks) the bounding box distance to the left side of the box by −⟨*length*⟩. Also see /tcb/left skip → P. 79.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[width=5cm,grow to left by=2cm,enhanced,show bounding box]
This is a \textbf{tcolorbox} with a width of 7cm.
\end{tcolorbox}
```

This is a **tcolorbox** with a width of 7cm.

/tcb/grow to right by=⟨*length*⟩ (no default, initially 0mm)

Enlarges the current box width by ⟨*length*⟩ and enlarges (shrinks) the bounding box distance to the right side of the box by −⟨*length*⟩. Also see /tcb/right skip → P. 79.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[grow to right by=2cm,enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}

\bigskip

\begin{tcolorbox}[grow to right by=2cm,grow to left by=1cm,
  enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

Floating box from `toggle enlargement`

This page is an odd page. Therefore, the left and right enlargements are not toggled (with some luck; otherwise use `forced`). This box stretches to the right margin on odd pages and to the left margin on even pages. The current document is one-sided – this feature makes sense for two-sided documents only.

`/tcb/toggle enlargement=`⟨*toggle preset*⟩ (default `evenpage`, initially `none`)

According to the ⟨*toggle preset*⟩, the left and the right enlargements of the bounding box are switched or not. Feasible values are:

- `none`: no switching.
- `forced`: the values of the left and right enlargement are switched.
- `evenpage`: if the page is an even page, the values of the left and right enlargement are switched. It is recommended that you use this setting in conjunction with `/tcb/check odd page` → P. 94.

```
% \usepackage{changepage} for 'check odd page'
\tcbset{colframe=blue!75!black,colback=white,
  grow to left by=20mm,grow to right by=-5mm}

\begin{tcolorbox}[toggle enlargement=none,enhanced,show bounding box]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[toggle enlargement=forced]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[toggle enlargement=evenpage,check odd page]
  This page is an \ifthenelse{\isodd{\thepage}}{odd}{even} page.
  Therefore, the left and right enlargements
  \ifthenelse{\isodd{\thepage}}{are not}{are} toggled.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

This page is an odd page. Therefore, the left and right enlargements are not toggled.

```
% \usepackage{changepage} for 'check odd page'
\begin{tcolorbox}[colframe=red!60!black,colback=red!15!white,check odd page,
  fonttitle=\bfseries,title=Floating box from \texttt{toggle enlargement},
  width=\textwidth,grow to right by=2cm,toggle enlargement=evenpage,float=t]
  This page is an \ifthenelse{\isodd{\thepage}}{odd}{even} page.
  Therefore, the left and right enlargements
  \ifthenelse{\isodd{\thepage}}{are not}{are} toggled (with some luck; otherwise
  use |forced|). This box stretches to the right margin on odd pages and to the left
  margin on even pages. The current document is one-sided -- this feature makes
  sense for two-sided documents only.
\end{tcolorbox}
```

The following keys should not be used with breakable boxes or boxes with a lower part.

/tcb/shrink tight (style, no value, initially unset)

The total colored box is shrunk to the dimensions of the upper part. There should be no lower part and no title. This style sets the /tcb/boxsep → P. 37 to 0pt and other geometry keys to fitting values. This option is likely to be used with the following extrusion keys.

```
\tcbset{colframe=blue!75!black,colback=white,arc=0mm,boxrule=0.4pt,
        nobeforeafter,tcbox raise base,shrink tight}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\end{tcolorbox}

Lorem \tcbox{ipsum} dolor sit amet, consectetuer adipiscing elit.
```

This is a **tcolorbox**.
Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

U 2014-09-19 **/tcb/extrude left by=**⟨*length*⟩ (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the left side. The inner width and the bounding box is kept unchanged and the operation is additive!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude left by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

U 2014-09-19 **/tcb/extrude right by=**⟨*length*⟩ (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the right side. The inner width and the bounding box is kept unchanged and the operation is additive!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude right by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

/tcb/extrude top by=⟨*length*⟩ (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the top side. The inner width and the bounding box is kept unchanged and the operation is additive!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude top by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

/tcb/extrude bottom by=⟨*length*⟩ (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the bottom side. The inner width and the bounding box is kept unchanged and the operation is additive!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude bottom by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

/tcb/extrude by=⟨*length*⟩ (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to all sides. The inner width and the bounding box is kept unchanged and the operation is additive!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis. \tcbox{Curabitur} dictum
gravida mauris. \tcbox[colframe=Green,interior style={opacity=0.0}]{Nam}
arcu libero, nonummy eget, consectetuer id, \tcbox{vulputate} a, magna. Donec
vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus
et malesuada fames ac turpis egestas. \tcbox{Mauris ut leo.}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

4.17 Layered Boxes and Every Box Settings

A `tcolorbox` may contain another `tcolorbox` and so on. The package takes track of the nesting level using a counter `tcblayer`. Counter values may be used for doing some fancy things, but you should never change the counter value yourself.

The package takes special care for the first four layers or nesting levels, called managed layers. Here, footnote texts are administrated to find their intended place and specific layer dependent options may be set by changing `/tcb/every box on layer n` →P. 87. If needed, the number of managed layers can be increased by setting `\tcbsetmanagedlayer` →P. 87 to a higher value than 4.

The following styles have a considerable influence on how layered boxes are processed. Note especially that nested boxes are getting a `/tcb/reset` →P. 95 by default. You can change this, but be prepared for suprises if you do.

If the defaults are *not changed*, a `tcolorbox` gets its options in the following order. Following options overwrite preceding options.

1. On package load, all options are set to default values.
2. Every `\tcbset` →P. 12 command adds or changes options for the following boxes inside the current TEX group.
3. While entering a `tcolorbox`, a `/tcb/every box on layer n` →P. 87 or `/tcb/every box on higher layers` →P. 87 option list is applied. With default settings this means:
   - For layer 1 (lowest layer), the `/tcb/every box` option list is applied. Not overwritten options given by a preceding `\tcbset` →P. 12 survive.
   - For layer 2 and above (nested boxes), a `/tcb/reset` →P. 95 followed by `/tcb/every box` option list is applied. Every resettable options given by a preceding `\tcbset` →P. 12 and by the sourrounding box(es) are reset.
4. The ⟨*options*⟩ given to the `tcolorbox` are applied. Or, if the box was generated by `\newtcolorbox` →P. 14 or friends, the ⟨*options*⟩ given there are applied.
5. If the box was generated by `\newtcolorbox` →P. 14 or friends, some automated options are applied.

`/tcb/every box` (style)

By default, this style is empty.

```
% default setting:
\tcbset{every box/.style={}}
```

It may be changed by redefining this style.

```
% setting all boxes to be enhanced:
\tcbset{every box/.style={enhanced}}
```

The alternative for setting something for every box (on every layer) is `\tcbsetforeverylayer` →P. 12:

```
% setting all boxes to be enhanced:
\tcbsetforeverylayer{enhanced}
```

/tcb/every box on layer n (style)

Here, n has to be replaced by a number ranging from 1 to the highest managed layer number (4 by default).

```
% default settings:
\tcbset{
  every box on layer 1/.style={every box},
  every box on layer 2/.style={reset,every box},
  every box on layer 3/.style={reset,every box},
  every box on layer 4/.style={reset,every box},
  }
```

/tcb/every box on higher layers (style)

Higher layers are layers above the highest managed layer number (4 by default).

```
\tcbset{every box on higher layers/.style={reset,every box}}
```

\tcbsetmanagedlayer{⟨*number*⟩}

Replaces the highest managed layer number by ⟨*number*⟩ where 4 is the default. This macro can only be used inside the preamble. Using a ⟨*number*⟩ lower than 4 typically makes no sense, but is not forbidden.

```
% \usepackage{lipsum}
% \tcbuselibrary{skins,breakable}
\tcbset{colframe=red!75!black,fonttitle=\bfseries,
  colback=red!5!white,
  every box/.style={enhanced,watermark text=\thetcblayer,
    before=\par\smallskip,after=\par\smallskip},
  every box on layer 2/.style={reset,every box,colback=yellow!10!white,
    drop fuzzy shadow}}
\begin{tcolorbox}[enhanced jigsaw,breakable,title=Layer 1 Box]
Here comes a footnote\footnote{Footnote from layer 1 box}.
\lipsum[2]
  \begin{tcolorbox}[title=Layer 2 Box]
  abc\footnote{The footnote of abc}
  \end{tcolorbox}
  \begin{tcolorbox}[title=Another Box,ams equation]
    \tcbhighmath{\sum\limits_{n=1}^{\infty} \frac{1}{n}} = \infty.
  \end{tcolorbox}
Some text\footnote{Footnote from some text}.
  \begin{tcolorbox}[title=Yet Another Box]
    \tcboxfit[height=2cm]{\lipsum[1]}
    My text.
    \begin{tcolorbox}
      Another lipsum text\footnote{A lipsum text}. \lipsum[3]
      \begin{tcolorbox}[title=Layer 4,colframe=blue,colback=white]
        Layer 4\footnote{Layer 4 footnote}
      \end{tcolorbox}
      The End\footnote{Last footnote}.
    \end{tcolorbox}
  \end{tcolorbox}
\end{tcolorbox}
```

Layer 1 Box

Here comes a footnote[a]. 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.

Layer 2 Box

abc[a]

[a] The footnote of abc

Another Box

$$\boxed{\sum_{n=1}^{\infty} \frac{1}{n}} = \infty. \tag{1}$$

Some text[b].

Yet Another Box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My text.

Another lipsum text[a]. Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Layer 4

Layer 4[a]

[a] Layer 4 footnote

The End[b].

[a] A lipsum text
[b] Last footnote

[a] Footnote from layer 1 box
[b] Footnote from some text

4.18 Capture Mode

/tcb/capture=⟨*mode*⟩ (no default, initially `minipage`)

The capture ⟨*mode*⟩ defines how the box content is processed.

Feasible values for ⟨*mode*⟩ are:

- **minipage**:
  This is the default ⟨*mode*⟩ for `tcolorbox`→P. 11. The content may have an upper and a lower part. Optionally, the box can be `/tcb/breakable`→P. 306. The box content is put into a minipage or into something similar to a minipage.
- **hbox**:
  This is the default ⟨*mode*⟩ for `\tcbox`→P. 13. The content cannot have a lower part and cannot be broken. The colored box is sized according to the dimensions of the content. A shortcut to set this mode is `/tcb/hbox`.
- **fitbox**: (needs the LIB fitting library)
  This is the default ⟨*mode*⟩ for `\tcboxfit`→P. 334. The content cannot have a lower part and cannot be broken. The content is sized according to the dimensions of the colored box. A shortcut to set this mode is `/tcb/fit`→P. 336.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[capture=minipage]
This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[capture=hbox]
This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[capture=fitbox,height=9mm]% needs the 'fitting' library
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

This is a tcolorbox.

This is a tcolorbox.

/tcb/hbox (style, no default)

Shortcut for `capture=hbox`.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[hbox]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/minipage (style, no default)

Shortcut for `capture=minipage`.

4.19 Text Characteristics

`/tcb/parbox=true|false` (default `true`, initially `true`)

The text inside a `tcolorbox` is formatted using a LaTeX `minipage` if the box is unbreakable. If breakable, the box tries a mimicry of a `minipage`. In a `minipage` or `parbox`, paragraphs are formatted slightly different as the main text. If the key value is set to `false`, the normal main text behavior is restored. In some situations, this has some unwanted side effects. It is recommended that you use this experimental setting only where you really want to have this feature.

```
% \usepackage{lipsum}  % preamble
\tcbset{width=(\linewidth-2mm)/2,nobeforeafter,arc=1mm,
  colframe=blue!75!black,colback=white,fonttitle=\bfseries,fontupper=\small,
  left=2mm,right=2mm,top=1mm,bottom=1mm,equal height group=parbox}

\begin{tcolorbox}[parbox,adjusted title={parbox=true (normal)}]
  \lipsum[1-2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[parbox=false,adjusted title={parbox=false}]
  \lipsum[1-2]
\end{tcolorbox}%
```

parbox=true (normal)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.
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.

parbox=false

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

/tcb/hyphenationfix=true|false (default `true`, initially `false`)

Long words at the beginning of paragraphs in very narrow boxes will not be hyphenated using `pdflatex`. This problem is circumvented by applying the `hyphenationfix` option.

```
\tcbset{colframe=blue!75!black,
  fontupper=\normalsize,
  colback=blue!5!white,width=4cm}

\begin{tcolorbox}
Rechnungsadjunktentochter.\par
Statthaltereikonzipist.
\end{tcolorbox}

\begin{tcolorbox}[hyphenationfix]
Rechnungsadjunktentochter.\par
Statthaltereikonzipist.
\end{tcolorbox}
```

Rechnungsadjunktentochter.
Statthaltereikonzipist.

Rechnungsad-
junktentochter.
Statthal-
tereikonzipist.

`parbox=false` and `hyphenationfix` should not be used together. They are targeting different box types and they do not blend very well.

4.20 Files

/tcb/tempfile=⟨*file name*⟩ (no default, initially `\jobname.tcbtemp`)

Sets ⟨*file name*⟩ as name for the temporary file which is used inside `tcbwritetemp` → P. 105 and `\tcbusetemp` → P. 105 implicitely.

4.21 \tcbox Specials

The following options are applicable for `\tcbox` → P. 13 and `\tcboxmath` → P. 283 only.

/tcb/tcbox raise=⟨*length*⟩ (no default, initially `0pt`)

Raises the `\tcbox` → P. 13 by the given ⟨*length*⟩.

```
\tcbset{colframe=blue!50!black,colback=white,colupper=red!50!black,
        fonttitle=\bfseries,nobeforeafter,center title}

Test\dotfill
\tcbox[tcbox raise base]{Hello World 1}\dotfill
\tcbox{Hello World 2}\dotfill
\tcbox[tcbox raise=5mm]{Hello World 3}
```

Test . . . . . Hello World 1 . . . . . Hello World 2 . . . . . Hello World 3

/tcb/tcbox raise base (style, no value, initially unset)

Raises the `\tcbox` → P. 13 such that the base of its content matches the base of the environmental line; see example above.

/tcb/on line (style, no value, initially unset)

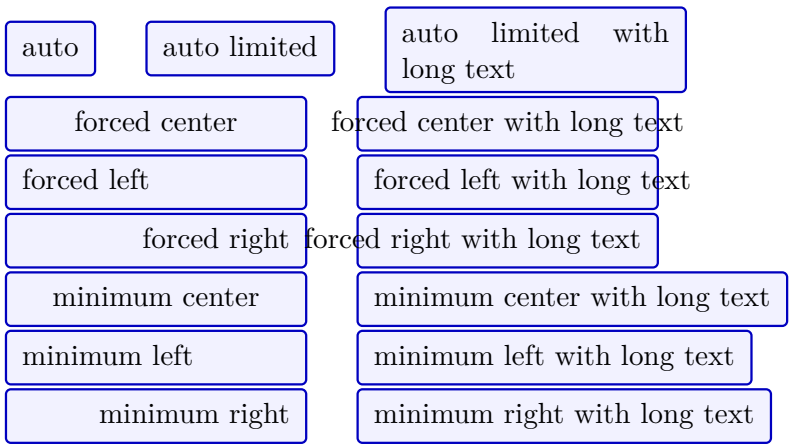
Combines `/tcb/tcbox raise base` with `/tcb/nobeforeafter` → P. 76. The resulting box behaves analogue to `\fbox`.

N 2015-03-23 **/tcb/tcbox width=⟨*mode*⟩** (no default, initially `auto`)

Controls how `\tcbox`→P. 13 respects a `/tcb/width`→P. 33 setting. Feasible values for ⟨*mode*⟩ are:

- **`auto`** (initial setting): ignore `/tcb/width`→P. 33 and set box width according to its content.
- **`auto limited`**: Set box width according to its content, if it is smaller than `/tcb/width`→P. 33. Otherwise, the content is set like in a `tcolorbox`→P. 11 with line breaks.
- **`forced center`**: Set box width according to `/tcb/width`→P. 33. The content is centered and may overlap the box borders.
- **`forced left`**: Set box width according to `/tcb/width`→P. 33. The content is left aligned and may overlap the box borders.
- **`forced right`**: Set box width according to `/tcb/width`→P. 33. The content is right aligned and may overlap the box borders.
- **`minimum center`**: Set box width according to `/tcb/width`→P. 33, if the content fits into. The content is centered and the box width may grow beyond `/tcb/width`→P. 33.
- **`minimum left`**: Set box width according to `/tcb/width`→P. 33, if the content fits into. The content is left aligned and the box width may grow beyond `/tcb/width`→P. 33.
- **`minimum right`**: Set box width according to `/tcb/width`→P. 33, if the content fits into. The content is right aligned and the box width may grow beyond `/tcb/width`→P. 33.

```
\tcbset{size=small,on line,before upper=\strut,
  colframe=blue!75!black,colback=blue!5!white,
  fontupper=\normalsize,width=4cm}

\tcbox[tcbox width=auto]{auto}\qquad
\tcbox[tcbox width=auto limited]{auto limited}\qquad
\tcbox[tcbox width=auto limited]{auto limited with long text}\\
\tcbox[tcbox width=forced center]{forced center}\qquad
\tcbox[tcbox width=forced center]{forced center with long text}\\
\tcbox[tcbox width=forced left]{forced left}\qquad
\tcbox[tcbox width=forced left]{forced left with long text}\\
\tcbox[tcbox width=forced right]{forced right}\qquad
\tcbox[tcbox width=forced right]{forced right with long text}\\
\tcbox[tcbox width=minimum center]{minimum center}\qquad
\tcbox[tcbox width=minimum center]{minimum center with long text}\\
\tcbox[tcbox width=minimum left]{minimum left}\qquad
\tcbox[tcbox width=minimum left]{minimum left with long text}\\
\tcbox[tcbox width=minimum right]{minimum right}\qquad
\tcbox[tcbox width=minimum right]{minimum right with long text}
```



4.22 Counters, Labels, and References

/tcb/phantom=⟨*code*⟩ (no default, initially unset)

The ⟨*code*⟩ is put in a box at the upper left corner of the `tcolorbox`. If the `tcolorbox` is breakable, the ⟨*code*⟩ is executed for the first box of the break sequence only. If there already was some phantom code given, the new ⟨*code*⟩ is appended.

The ⟨*code*⟩ is intended to be used for counter stepping, labelling, and related operations which do not produce visible text.

- The ⟨*code*⟩ is executed before the title and box content, i. e. counter values are ensured to be increased before usage.
- Labels are ensured to reference the correct page number.
- The ⟨*code*⟩ is executed only once even during fitting operations for title and box content.
- In combination with the `hyperref` package, the hyper anchor is set to the upper left corner of the `tcolorbox`, i. e. links inside the pdf document will jump to the box pleasantly.
- Since the ⟨*code*⟩ is executed inside a TEX group, only global operations can survive this group.

Examples for the `phantom` usage are given in Section 13.9 from page 275, e. g. Example 13.1 on page 276.

/tcb/nophantom (no value, initially set)

Removes the phantom code if set before.

/tcb/label=⟨*marker*⟩ (no default, initially unset)

The ⟨*marker*⟩ is set as label text for a reference with the `\ref` macro. Typically, this option is used for numbered boxes, see Subsection 5.1 from page 97, e. g. `/tcb/new/auto counter`→ P. 97.

N 2014-11-28 **/tcb/phantomlabel**=⟨*marker*⟩ (no default, initially unset)

Equivalent to `/tcb/label` for an *unnumbered* box. A `\phantomsection` from the package `hyperref` is used to set a correct hyperlink target. This is not needed for a numbered box.

/tcb/label type=⟨*type*⟩ (no default, initially unset)

This option key can be used only in conjunction with the `cleveref` package [5] which has to be loaded separately. ⟨*type*⟩ has to be a cross-reference type *known* to `cleveref` like `theorem`, `algorithm`, `result`, etc. References made with `cleveref` will use this type. Note that using `label type` will result in compilation errors, if `cleveref` is not loaded. For an example, see Theorem 14.3.5 on page 300.

/tcb/no label type (no value, initially set)

Removes a `/tcb/label type`, if set before.

/tcb/step=⟨*counter*⟩ (no default, initially unset)

Shortcut for `phantom={\refstepcounter{#1}}`. The given ⟨*counter*⟩ is increased and ready for labelling. This option is not needed when using the convenient automated numbering introduced with version 2.40, see Subsection 5.1 from page 97.

/tcb/step and label={⟨*counter*⟩}{⟨*marker*⟩} (no default, initially unset)

Shortcut for using `/tcb/step` and `/tcb/label`. This option is not needed when using the convenient automated numbering introduced with version 2.40, see Subsection 5.1 from page 97.

/tcb/list entry=⟨*text*⟩ (no default, initially unset)

If the «list of tcolorbox(es)» feature described in Subsection 5.2 from page 104 is used, this key describes the ⟨*text*⟩ for an entry into the generated list, e. g.

```
list entry={\protect\numberline{\thetcbcounter}My beautiful Example}
```

See Section 13.9 from page 275 for a complete example.

N 2014-09-19 **/tcb/list text=⟨*text*⟩** (style, no default)

This is a shortcut for setting /tcb/list entry to \protect\numberline{\thetcbcounter}⟨*text*⟩. So, the following settings are identical:

```
list text={My beautiful Example},
list entry={\protect\numberline{\thetcbcounter}My beautiful Example}
```

See Section 13.9 from page 275 for a complete example.

/tcb/add to list={⟨*list*⟩}{⟨*type*⟩} (no default, initially unset)

If the «list of tcolorbox(es)» feature described in Subsection 5.2 from page 104 is used, list entries are generated automatically. With this key, you can enforce an entry to the given ⟨*list*⟩ with the given ⟨*type*⟩. This issues:
\addcontentsline{⟨*list*⟩}{⟨*type*⟩}{⟨*entry text*⟩}

/tcb/check odd page=true|false (default true, initially false)

If set to true, the even/odd page testing from the package changepage is applied. The /tcb/toggle enlargement^{→P. 83} check and the /tcb/toggle left and right^{→P. 44} check will use the \ifoddpage macro from this package or another package like ifoddpage. This options is independent from /tcb/phantom^{→P. 93}. Note that you have to include the package changepage by hand[3]:

```
\usepackage{changepage}
\strictpagecheck
```

The macro \ifoddpage can be used inside overlay or watermark code to test if the box is on an odd page. This will work also for boxes in a break sequence. Note that you cannot use the test inside the normal box content.

```
% \usepackage{changepage}
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enhanced,check odd page,
  watermark text={\ifoddpage Odd\else Even\fi\ page!}]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Even page!

[3] If changepage is not included, a compilation error will arise.

4.23 Externalization

See Section 20 on page 369 for the LIB external library of `tcolorbox`.

If the *externalization* library of the `tikz` package is used and `/tcb/graphical environment` → P. 112 is set to `tikzpicture`, a `tcolorbox` could trigger the externalization process which will arise a compilation error.

To avoid this, there are two possible strategies:

- Ensure, that `\tikzexternaldisable` is set before a `tcolorbox` is used. If you typically use the pattern `\tikzexternalenable` *some picture* `\tikzexternaldisable`, there is nothing to care about.
- If *externalization* is enabled globally, use `/tcb/shield externalize` to shield any `tcolorbox`. The preamble code could look like this:

```
\usetikzlibrary{external}
\tikzexternalize
\tcbset{shield externalize}
```

`/tcb/shield externalize=true|false` (default `true`, initially `false`)

If set to `true`, the drawing part of the `tcolorbox` is not being externalized which is a good thing at the current state of art. Nevertheless, if the `tcolorbox` contains a `tikzpicture`, this picture is still externalized. Pictures drawn with help of `/tcb/tikz upper` → P. 62 or alike are *not* externalized.

If a `tcolorbox` is used inside a node of an encircling `tikzpicture` which is externalized, do *not* use `\tikzexternaldisable` in front of the `tcolorbox`. `/tcb/shield externalize` is deactivated automatically inside a `tikzpicture`.

`/tcb/external=`⟨*file name*⟩ (no default, initially unset)

Convenience option which calls `\tikzsetnextfilename{`⟨*file name*⟩`}`. Typically, it may be used inside the option list of a `tcolorbox` to set the externalization ⟨*file name*⟩ for the first `tikzpicture` which is discovered *inside* the box content. The package `tikz` [20] or the library LIB skins has to be loaded to use this option. Additionally, `\usetikzlibrary{external}` has to be used.

`/tcb/remake=true|false` (default `true`, initially `false`)

Convenience option which calls `/tikz/external/remake next`. Typically, it may be used inside the option list of a `tcolorbox` to force the remake of the first `tikzpicture` which is discovered *inside* the box content. The package `tikz` [20] or the library LIB skins has to be loaded to use this option. Additionally, `\usetikzlibrary{external}` has to be used.

4.24 Miscellaneous

`/tcb/reset` (no value, initially set)

Sets (nearly) all `tcolorbox` settings (including loaded libraries) back to their default values *plus* any settings given by `\tcbsetforeverylayer` → P. 12. `/tcb/savedelimiter` → P. 25 and `/tcb/capture` → P. 89 keep their values. Also, all raster values (see Section 12 on page 231) are not resetted.

This option is useful for boxes in boxes where the inner box should not inherit the settings of the outer box. Note that for boxes inside boxes the `reset` is done automatically, if the standard settings of the package are used (v2.40 and above), see Section 4.17 from page 86.

N 2015-01-09 **/tcb/only=<⟨*overlay specification*⟩>{⟨*options*⟩}** (style, no default, initially unset)

Sets the given `tcolorbox` ⟨*options*⟩ in dependency of a `beamer` ⟨*overlay specification*⟩. Note that this needs the `beamer` class. The ⟨*options*⟩ will only be used on the specified `beamer` frames.

```
\documentclass{beamer}
\usepackage[many]{tcolorbox}
\begin{document}

\begin{frame}
  \begin{tcolorbox}[title=My title,fonttitle=\bfseries,
      enhanced,colframe=red!50!black,colback=red!10,colbacktitle=red,
      sidebyside,righthand width=3cm,
      lowerbox=invisible,lower separated=false,
      drop lifted shadow,
      only=<1>{colbacktitle=yellow,coltitle=red!50!black,colframe=red},
      only=<3>{colback=yellow!50,watermark text={Attention!}},
      only=<3->{lowerbox=visible}  ]
    This is a test.
    \begin{itemize}[<+->]
      \item One
      \item Two
      \item \alert<3>{Three}
      \item Four
    \end{itemize}
  \tcblower
    \begin{equation*}
    \int\limits_{1}^{x} \frac{1}{t}~dt = \ln(x).
    \end{equation*}
  \end{tcolorbox}
\end{frame}

\end{document}
```

/tcb/code=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is executed immediately. This option is useful to place some arbitrary code into an option list.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  code={Useless at this spot but functional.},
  fonttitle=\bfseries}

\begin{tcolorbox}[code={\newcommand{\mycommand}{\textit{working}}},
  title=My \mycommand\ title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Useless at this spot but functional.

My *working* title

This is a **tcolorbox**.

5 Initialization Option Keys

The *initialization* options are only applicable for the generation of new environments and commands based on `tcolorbox` and friends. Particularly, they can be used for

- `\newtcolorbox` → P. 14,
- `\newtcbox` → P. 15,
- `\newtcblisting` → P. 246,
- `\newtcbinputlisting` → P. 248,
- `\newtcbtheorem` → P. 281, and
- `\newtcboxfit` → P. 335.

Typically, these options may generate counters and alike. It is **strongly** recommended that you use initialization options inside the preamble only. Otherwise, you may get trouble when using LaTeX's `\include` features.

5.1 Numbered Boxes

Counters assigned using the initialization options are administrated automatically. Especially, they are increased for each new box. Independent from the real counter name, the counter value can be referenced by `\thetcbcounter`, e. g. inside the title of the box. The real counter name is stored inside `\tcbcounter`.

`/tcb/new/auto counter` (no value, initially unset)

Creates a new counter automatically. With `/tcb/new/number format` → P. 99 and `/tcb/new/number within` → P. 99, the appearance and behavior of the counter can be changed. The counter value is referenced by `\thetcbcounter`.

Definition in the preamble:

```
\newtcolorbox[auto counter,number within=section]{pabox}[2][]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Examp.~\thetcbcounter: #2,#1}
```

```
\begin{pabox}[label={myautocounter}]{Title with number}
This box is automatically numbered with \ref{myautocounter} on page
\pageref{myautocounter}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{pabox}
```

Examp. 5.1: Title with number

This box is automatically numbered with 5.1 on page 97. Inside the box, the 5.1 can also be referenced by `\thetcbcounter`. The real counter name is `tcb@cnt@pabox`.

/tcb/new/use counter from=⟨*tcolorbox*⟩ (no default, initially unset)

Here, a counter from another ⟨*tcolorbox*⟩ is reused. Note that the settings for /tcb/new/number format→P. 99 and /tcb/new/number within→P. 99 are inherited and cannot be changed. The counter value is referenced by **\thetcbcounter**.

```
\newtcolorbox[use counter from=pabox]{mybox}[2][]{%
colback=blue!5!white,colframe=blue!75!black,fonttitle=\bfseries,
title=Some Box \thetcbcounter: #2,#1}

\begin{mybox}[label={myusecounterfrom}]{Title with continued number}
This box is automatically numbered with \ref{myusecounterfrom} on page
\pageref{myusecounterfrom}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{mybox}
```

Some Box 5.2: Title with continued number

This box is automatically numbered with 5.2 on page 98. Inside the box, the 5.2 can also be referenced by `\thetcbcounter`. The real counter name is `tcb@cnt@pabox`.

/tcb/new/use counter=⟨*counter*⟩ (no default, initially unset)

Here, an ordinary existing LaTeX ⟨*counter*⟩ is used for numbering. With /tcb/new/number format→P. 99 and /tcb/new/number within→P. 99, the appearance and behavior of the counter can be changed. The counter value is referenced by **\thetcbcounter**.

```
% \newcounter{myexample}%  preamble
\newtcolorbox[use counter=myexample,number format=\Alph]{mybox}[2][]{%
colback=green!5!white,colframe=green!55!black,fonttitle=\bfseries,
title=Some Box \thetcbcounter: #2,#1}

\begin{mybox}[label={myusecounter}]{Title with \LaTeX\ number}
This box is automatically numbered with \ref{myusecounter} on page
\pageref{myusecounter}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{mybox}
```

Some Box A: Title with LaTeX number

This box is automatically numbered with A on page 98. Inside the box, the A can also be referenced by `\thetcbcounter`. The real counter name is `myexample`.

N 2014-09-19 **/tcb/new/use counter*=⟨*counter*⟩** (no default, initially unset)

An existing LaTeX ⟨*counter*⟩ is used for numbering. In contrast to /tcb/new/use counter, the options /tcb/new/number format→P. 99 and /tcb/new/number within→P. 99 are ignored. Use this for counters which are already configured outside the `tcolorbox` package, e. g. the standard `figure` counter.

/tcb/new/no counter (no value, initially set)

The created boxes are not numbered. This is the default. The option may be used to overrule a previous option.

/tcb/new/number within=⟨*counter*⟩ (no default, initially unset)

The automatic counter is set to zero, if ⟨*counter*⟩ is increased. Additionally, during output, the value of ⟨*counter*⟩ is prepended to the value of the automatic counter.

To prepend the automatic counter with the chapter number and to reset it with every new chapter, use:

```
number within=chapter
```

See /tcb/new/use counter → P. 98 for a complete example.

/tcb/new/number format=⟨*format macro*⟩ (no default, initially \arabic)

Declares the format of the automatic counter. The ⟨*format macro*⟩ can be any valid LaTeX number formatting macro like \arabic, \roman, etc.

To display the counter value in large roman numbers, use:

```
number format=\Roman
```

See /tcb/new/auto counter → P. 97 for a complete example.

/tcb/new/number freestyle=⟨*code*⟩ (no default, initially unset)

Allows advanced control over the complete number format. This option overrules the format given by /tcb/new/number within and /tcb/new/number format. Nevertheless, you can combine it with /tcb/new/number within to get the desired reset property.

The ⟨*code*⟩ is some formatting code which should contain \tcbcounter to reference the automated counter. Since this ⟨*code*⟩ is expanded, you have to secure each macro with \noexpand with *exception* of \tcbcounter.

Definition in the preamble:

```
\newtcolorbox[auto counter,number within=section,
  number freestyle={(Q/\noexpand\thesection/\noexpand\Alph{\tcbcounter})},
  ]{phbox}[2][]{%
  colback=yellow!15!white,colframe=blue!75!black,fonttitle=\bfseries,
  title=Question~\thetcbcounter: #2,#1}
```

```
\begin{phbox}[label={myfreestyle}]{Title with freestyle number}
This box is automatically numbered with \ref{myfreestyle} on page
\pageref{myfreestyle}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{phbox}
```

Question (Q/5/A): Title with freestyle number

This box is automatically numbered with (Q/5/A) on page 99. Inside the box, the (Q/5/A) can also be referenced by \thetcbcounter. The real counter name is tcb@cnt@phbox.

The following options /tcb/new/crefname and /tcb/new/Crefname need to be set inside the preamble.

U 2014-12-01 **/tcb/new/crefname**={⟨*singular*⟩}{⟨*plural*⟩} (no default, initially unset)

This option key can be used only in conjunction with the `cleveref` package [5] which has to be loaded separately. It creates a cross-reference type for the new `tcolorbox`'es, where the lowercase ⟨*singular*⟩ and ⟨*plural*⟩ forms of the cross-reference are given. This type is the environment or macro name and /tcb/label type→ P. 93 is set automatically. See /tcb/label type→ P. 93 and [5] for more information.

U 2014-12-01 **/tcb/new/Crefname**={⟨*singular*⟩}{⟨*plural*⟩} (no default, initially unset)

This option key can be used only in conjunction with the `cleveref` package [5] which has to be loaded separately. It creates a cross-reference type for the new `tcolorbox`'es, where the uppercase ⟨*singular*⟩ and ⟨*plural*⟩ forms of the cross-reference are given. This type is the environment or macro name and /tcb/label type→ P. 93 is set automatically. See /tcb/label type→ P. 93 and [5] for more information.

Definition in the preamble:

```
% \usepackage{cleveref}
\newtcolorbox[auto counter,number within=section,
  crefname={bluebox}{blueboxes}]%
  {mybluebox}[2][]{colback=blue!5!white,colframe=blue!75!black,fonttitle=\bfseries,
     title=Bluebox \thetcbcounter: #2,#1}
```

```
% \usepackage{varioref}
% \usepackage{cleveref}
\begin{mybluebox}[label={myreference}]{My title}
This is an example.
\end{mybluebox}

\Cref{myreference}, \cref{myreference}.\\
\Cpageref{myreference}, \cpageref{myreference}.\\
\nameCref{myreference}, \namecref{myreference}.\\
\labelcref{myreference}, \labelcpageref{myreference}.\\
With \texttt{varioref}:\\
\Vref{myreference}, \vref{myreference}.\\
\Vref*{myreference}, \vref*{myreference}.
```

Bluebox 5.1: My title

This is an example.

Bluebox 5.1, bluebox 5.1.
Page 100, page 100.
Bluebox, bluebox.
5.1, 100.
With `varioref`:
Bluebox 5.1, bluebox 5.1.
Bluebox 5.1, bluebox 5.1.

N 2014-09-19 **/tcb/new/blend into=⟨*name*⟩** (style, no default, initially unset)

Used to comfortably blend into an existing schema of naming and numbering for some selected cases. For example, a `tcolorbox` can be used to display and entitle an image pretending to be a standard `figure` environment. Here, `/tcb/title` → P. 17 is used instead of the standard `\caption` and `/tcb/list text` → P. 94 can be used instead of the optional parameter of the standard `\caption`.

Feasible values for ⟨*name*⟩ are:

- `figures`: blend into the standard `figure` environment.
- `tables`: blend into the standard `table` environment.
- `listings`: blend into the standard `lstlistings` environment of the package `listings` [6].

Note that `blend into=listings` can only be used in the document content or, preferably, inside a `\AtBeginDocument` clause! Using it without `\AtBeginDocument` inside the preamble does not work since the `listings` packages initializes its counter also inside `\AtBeginDocument`.

```
\begin{figure}[htb]
  \centering\includegraphics[height=4cm]{lichtspiel.jpg}
  \caption{A standard figure}
\end{figure}

\newtcolorbox[blend into=figures]{myfigure}[2][]{float=htb,capture=hbox,
  title={#2},every float=\centering,#1}

\begin{myfigure}{A tcolorbox figure}
  \includegraphics[height=4cm]{lichtspiel.jpg}
\end{myfigure}
```

Figure 1: A standard figure

Figure 2: A tcolorbox figure

N 2015-03-13 **/tcb/blend before title=**⟨*value*⟩ (no default, initially **colon**)

This option formats the title output of **/tcb/new/blend into** → P. 101. Note that this is a common **tcolorbox** option which should be set globally or in the normal option part of **\newtcolorbox** → P. 14.

Feasible values for ⟨*value*⟩ are:

- **colon**: use name/number plus colon.
- **dash**: use name/number plus dash.
- **colon hang**: use name/number plus colon with hanging indent.
- **dash hang**: use name/number plus dash with hanging indent.

```
\newtcolorbox[blend into=figures]{myfigure}[2][]{float=htb,capture=hbox,
  blend before title=dash hang,title={#2},every float=\centering,#1}

\begin{myfigure}{A tcolorbox figure with quite a long title}
  \includegraphics[height=5cm]{lichtspiel.jpg}
\end{myfigure}
```

Figure 3 – A tcolorbox figure with quite a long title

N 2015-03-13 **/tcb/blend before title code=⟨*code*⟩** (no default)

This option formats the title output of /tcb/new/blend into → P. 101. The ⟨*code*⟩ takes one parameter, the name/number. Use this, if /tcb/blend before title → P. 102 is not flexible enough.

```
\newtcolorbox[blend into=figures]{myfigure}[2][]{float=htb,capture=hbox,
  blend before title code={\fbox{##1}\ },title={#2},every float=\centering,#1}

\begin{myfigure}{A tcolorbox figure}
  \includegraphics[height=6cm]{lichtspiel.jpg}
\end{myfigure}
```

Figure 4 A tcolorbox figure

5.2 Lists of tcolorboxes

For figures and tables, LaTeX provides the `\listoffigures` and `\listoftables` commands to create lists of these numbered entities. Also, a `tcolorbox` can be part of such a kind of list.

1. Assign a list ⟨*name*⟩ by the *initialization* option `/tcb/new/list inside`.
2. Optionally, a new ⟨*type*⟩ for list entries may be assigned by the *initialization* option `/tcb/new/list type`.
3. List entries a generated automatically within each new `tcolorbox` using the above initialization.
   - If `/tcb/list entry` → P. 94 is set, the entry is generated with it.
   - Otherwise, if `/tcb/title` → P. 17 is set, the entry is generated with it.
   - Otherwise, the entry is generated with the current number and the environment name.
4. The generated list is displayed by `\tcblistof`.

`/tcb/new/list inside=`⟨*name*⟩ (no default, initially unset)

Assigns a list or contents file to the generated `tcolorbox`es. Entries to this list are saved to a file which gets the ⟨*name*⟩ as file name extension. The list is referenced by this name in `\tcblistof`. For example:

```
list inside=exam
```

See Section 13.9 from page 275 for a complete example.

`/tcb/new/list type=`⟨*type*⟩ (no default, initially `tcolorbox`)

Optionally, some ⟨*type*⟩ can be assigned to the list entries. For a new ⟨*type*⟩, a macro `\l@`⟨*type*⟩ has to exist which controls the format of the list entry. The default type is defined by

```
\newcommand*\l@tcolorbox{\@dottedtocline{1}{1.5em}{2.3em}}
```

This is identical to the `\l@section` setting of LaTeX. `\l@tcolorbox` can be redefined or a new ⟨*type*⟩ can be assigned.

`\tcblistof[`⟨*macro*⟩`]{`⟨*name*⟩`}{`⟨*title text*⟩`}`

Displays the generated list of `tcolorbox`es with the given ⟨*name*⟩. The heading is generated by ⟨*macro*⟩`{`⟨*title text*⟩`}` where `\section` is the default setting for ⟨*macro*⟩.
To display the list inside a subsection, use for example:

```
\tcblistof[\subsection]{exam}{List of Exercises}
```

The result of the example is found as Subsection 13.10 on page 278.

The core of the list is generated by `\@starttoc{`⟨*name*⟩`}` which can be wrapped into an own macro.

6 Saving and Loading of Verbatim Texts

The following macros are slightly modified versions of the original macros from the known packages `moreverb` and `verbatim`. They are used implicitly inside of a `tcolorbox` environment, but they can be used outside also.

`\begin{tcbverbatimwrite}{`⟨*file name*⟩`}`
⟨*environment content*⟩
`\end{tcbverbatimwrite}`

Saves the ⟨*environment content*⟩ to a file named by ⟨*file name*⟩. TEX macros inside the environment are not expanded.

```
\begin{tcbverbatimwrite}{\jobname_verbexp.tex}
  This text is saved \textit{as is}.
\end{tcbverbatimwrite}

Now, we are using the file:\par
\input{\jobname_verbexp.tex}
```

Now, we are using the file:
This text is saved *as is*.

This environment may be used inside an own environment. Note, that inside the environment definition `\tcbverbatimwrite` has to be used instead of `\begin{tcbverbatimwrite}` and `\endtcbverbatimwrite` instead of `\end{tcbverbatimwrite}`.

```
\newenvironment{myverbatim}{%
  \begingroup\tcbverbatimwrite{\jobname_myverb.tex}}%
  {\endtcbverbatimwrite\endgroup}

\begin{myverbatim}
  This is the text which is saved by my own environment.
\end{myverbatim}

Now, we are using the file:\par
\input{\jobname_myverb.tex}
```

Now, we are using the file:
This is the text which is saved by my own environment.

`\begin{tcbwritetemp}`
⟨*environment content*⟩
`\end{tcbwritetemp}`

Has the same function as `tcbverbatimwrite`, but uses the key value of `tempfile` for the file name.

```
\begin{tcbwritetemp}
  This text is saved \textit{as is}.
\end{tcbwritetemp}

Now, we are using the file:\par
\tcbusetemp
```

Now, we are using the file:
This text is saved *as is*.

`\tcbusetemp`

Loads the current temporary file which was saved by `tcbwritetemp`.

7 Recording

The package provides some macros and options to take *records* during compilation. This is done by LaTeX file operations to save some data to a file for later usage. The main application scenario is depicted in Section 7.3 on the next page where information about example solutions is recorded and read again in Section 7.4 on page 110.

7.1 Makros

N 2014-11-28 **\tcbstartrecording**[⟨*file name*⟩]

Opens a file denoted by ⟨*file name*⟩ for writing the records. The default file name is `\jobname.records`. See Section 7.3 on the next page for an example application.

N 2014-11-28 **\tcbrecord**{⟨*content*⟩}

Records any ⟨*content*⟩ to the record file. `\tcbrecord` is implemented as `\immediate\write`. `\tcbstartrecording` has to be called before; otherwise, `\tcbrecord` is silently ignored.

```
\tcbrecord{\string\solution{\thetcbcounter}{solutions/exercise-\thetcbcounter.tex}}
```

N 2014-11-28 **\tcbstoprecording**

Closes the current record file which was opened by `\tcbstartrecording` before.

N 2014-11-28 **\tcbinputrecords**[⟨*file name*⟩]

Opens a file denoted by ⟨*file name*⟩ for reading the records via `\input`. The default file name is the name of the last used record file for saving. `\tcbstoprecording` has to be called before.

7.2 Options

N 2014-11-28 **/tcb/record**=⟨*content*⟩ (style, no default)

Records any ⟨*content*⟩ to the record file, see `\tcbrecord`. This key can be used several times to write several lines.

```
record={\string\solution{\thetcbcounter}{solutions/exercise-\thetcbcounter.tex}}
```

N 2014-11-28 **/tcb/no recording**

Disables `\tcbrecord` and `/tcb/record` inside the current group.

7.3 Example: Exercises

The following application example creates exercises and their corresponding solutions. Each pair is generated inside a single `tcolorbox` where the solution is given below `\tcblower` → P. 11. For every example, the solution part is saved by `/tcb/savelowerto` → P. 23 to a file. The saving is recorded using `/tcb/record` → P. 106. To enlighten the possibilities, the second exercise has no solution. Finally, the solutions are input in Section 7.4 on page 110.

Definition in the preamble:

```
% \tcbuselibrary{skins,xparse}

\NewTColorBox[auto counter,number within=section]{exercise}{+O{}}{%
  enhanced,colframe=green!20!black,colback=yellow!10!white,coltitle=green!40!black,
  fonttitle=\bfseries,
  underlay={\begin{tcbclipinterior}
    \shade[inner color=green!80!yellow,outer color=yellow!10!white]
      (interior.north west) circle (2cm);
    \draw[help lines,step=5mm,yellow!80!black,shift={(interior.north west)}]
      (interior.south west) grid (interior.north east);
    \end{tcbclipinterior}},
  title={Exercise~\thetcbcounter:},
  label={exercise@\thetcbcounter},
  attach title to upper=\quad,
  after upper={\par\hfill\textcolor{green!40!black}%
    {\itshape Solution on page~\pageref{solution@\thetcbcounter}}},
  lowerbox=ignored,
  savelowerto=solutions/exercise-\thetcbcounter.tex,
  record={\string\solution{\thetcbcounter}{solutions/exercise-\thetcbcounter.tex}},
  #1
}

\NewTotalTColorBox{\solution}{mm}{%
  enhanced,colframe=red!20!black,colback=yellow!10!white,coltitle=red!40!black,
  fonttitle=\bfseries,
  underlay={\begin{tcbclipinterior}
    \shade[inner color=red!50!yellow,outer color=yellow!10!white]
      (interior.north west) circle (2cm);
    \draw[help lines,step=5mm,yellow!80!black,shift={(interior.north west)}]
      (interior.south west) grid (interior.north east);
    \end{tcbclipinterior}},
  title={Solution of Exercise~\ref{exercise@#1} on page~\pageref{exercise@#1}:},
  phantomlabel={solution@#1},
  attach title to upper=\par,
}{\input{#2}}

\tcbset{no solution/.style={no recording,after upper=}}
```

```
\tcbstartrecording

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=\sin((\sin x)^2)
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( \sin((\sin x)^2) \right)'
    =\cos((\sin x)^2) 2\sin x \cos x.
  \end{align*}
```

```
\end{exercise}

\begin{exercise}[no solution]
  It holds:
  \begin{equation*}
    \frac{d}{dx}\left(\ln|x|\right) = \frac{1}{x}.
  \end{equation*}
\end{exercise}

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=(\sin(\sin x))^2
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( (\sin(\sin x))^2 \right)'
    =2\sin(\sin x)\cos(\sin x)\cos x.
  \end{align*}
\end{exercise}

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=\sqrt{x^3-6x^2+2x}
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( \sqrt{x^3-6x^2+2x} \right)'
    = \frac{3x^2-12x+2}{2\sqrt{x^3-6x^2+2x}}.
  \end{align*}
\end{exercise}

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=\left(\frac{2+3x}{1-2x}\right)^3
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( \left(\frac{2+3x}{1-2x}\right)^3 \right)'
    = 3 \left(\frac{2+3x}{1-2x}\right)^2 \frac{(1-2x)3-(2+3x)(-2)}{(1-2x)^2}
    = \frac{21(2+3x)^2}{(1-2x)^4}.
  \end{align*}
\end{exercise}

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=\frac{\cos x}{(\tan 2x)^2}
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( \frac{\cos x}{(\tan 2x)^2} \right)'
    = \left( \frac{\cos x (\cos 2x)^2}{(\sin 2x)^2} \right)'\\
    &= \frac{(\sin 2x)^2 [(-\sin x)(\cos 2x)^2+(\cos x)4\cos 2x (-\sin 2x)]
       - \cos x (\cos 2x)^2 4\sin 2x \cos 2x}{(\sin 2x)^4}\\
    &= -\frac{\cos(2x) [\sin x \sin 2x \cos 2x+ 4\cos x(\sin 2x)^2
       + 4 \cos x (\cos 2x)^2]}{(\sin 2x)^3}\\
```

```
    &= -\frac{\cos(2x) [\sin x \sin 2x \cos 2x+ 4\cos x]}{(\sin 2x)^3}.
  \end{align*}
\end{exercise}

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=\cos((2x^2+3)^3)
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( \cos((2x^2+3)^3) \right)'
    =-\sin((2x^2+3)^3) 3(2x^2+3)^2 2\cdot 2x\\
    &=-12x(2x^2+3)^2\sin((2x^2+3)^3).
  \end{align*}
\end{exercise}

\begin{exercise}
  Compute the derivative of the following function:
  \begin{equation*}
    f(x)=(x^2+1)\sqrt{x^4+1}
  \end{equation*}
\tcblower
  The derivative is:
  \begin{align*}
    f'(x) &= \left( (x^2+1)\sqrt{x^4+1} \right)'
    = 2x\sqrt{x^4+1} + \frac{2x^3(x^2+1)}{\sqrt{x^4+1}}.
  \end{align*}
\end{exercise}

\tcbstoprecording
```

Exercise 7.1: Compute the derivative of the following function:

$$f(x) = \sin((\sin x)^2)$$

Solution on page 110

Exercise 7.2: It holds:

$$\frac{d}{dx}(\ln \mathtt{x}) = \frac{1}{x}.$$

Exercise 7.3: Compute the derivative of the following function:

$$f(x) = (\sin(\sin x))^2$$

Solution on page 110

Exercise 7.4: Compute the derivative of the following function:

$$f(x) = \sqrt{x^3 - 6x^2 + 2x}$$

Solution on page 110

Exercise 7.5: Compute the derivative of the following function:

$$f(x) = \left(\frac{2+3x}{1-2x}\right)^3$$

Solution on page 111

Exercise 7.6: Compute the derivative of the following function:

$$f(x) = \frac{\cos x}{(\tan 2x)^2}$$

Solution on page 111

Exercise 7.7: Compute the derivative of the following function:

$$f(x) = \cos((2x^2+3)^3)$$

Solution on page 111

Exercise 7.8: Compute the derivative of the following function:

$$f(x) = (x^2+1)\sqrt{x^4+1}$$

Solution on page 111

7.4 Example: Solutions

This concludes the example given in Section 7.3 on page 107. Now, the saved and recorded solutions are included.

```
\tcbinputrecords
```

Solution of Exercise 7.1 on page 109:
The derivative is:

$$f'(x) = \Big(\sin((\sin x)^2)\Big)' = \cos((\sin x)^2)2\sin x\cos x.$$

Solution of Exercise 7.3 on page 109:
The derivative is:

$$f'(x) = \Big((\sin(\sin x))^2\Big)' = 2\sin(\sin x)\cos(\sin x)\cos x.$$

Solution of Exercise 7.4 on page 109:
The derivative is:

$$f'(x) = \Big(\sqrt{x^3-6x^2+2x}\Big)' = \frac{3x^2-12x+2}{2\sqrt{x^3-6x^2+2x}}.$$

Solution of Exercise 7.5 on page 110:
The derivative is:

$$f'(x) = \left(\left(\frac{2+3x}{1-2x}\right)^3\right)' = 3\left(\frac{2+3x}{1-2x}\right)^2 \frac{(1-2x)3-(2+3x)(-2)}{(1-2x)^2} = \frac{21(2+3x)^2}{(1-2x)^4}.$$

Solution of Exercise 7.6 on page 110:
The derivative is:

$$\begin{aligned} f'(x) &= \left(\frac{\cos x}{(\tan 2x)^2}\right)' = \left(\frac{\cos x(\cos 2x)^2}{(\sin 2x)^2}\right)' \\ &= \frac{(\sin 2x)^2[(-\sin x)(\cos 2x)^2 + (\cos x)4\cos 2x(-\sin 2x)] - \cos x(\cos 2x)^2 4\sin 2x\cos 2x}{(\sin 2x)^4} \\ &= -\frac{\cos(2x)[\sin x\sin 2x\cos 2x + 4\cos x(\sin 2x)^2 + 4\cos x(\cos 2x)^2]}{(\sin 2x)^3} \\ &= -\frac{\cos(2x)[\sin x\sin 2x\cos 2x + 4\cos x]}{(\sin 2x)^3}. \end{aligned}$$

Solution of Exercise 7.7 on page 110:
The derivative is:

$$\begin{aligned} f'(x) &= \left(\cos((2x^2+3)^3)\right)' = -\sin((2x^2+3)^3)3(2x^2+3)^2 2\cdot 2x \\ &= -12x(2x^2+3)^2\sin((2x^2+3)^3). \end{aligned}$$

Solution of Exercise 7.8 on page 110:
The derivative is:

$$f'(x) = \left((x^2+1)\sqrt{x^4+1}\right)' = 2x\sqrt{x^4+1} + \frac{2x^3(x^2+1)}{\sqrt{x^4+1}}.$$

8 Technical Overview and Customization

This section provides a technical overview of the skin concept of `tcolorbox`. For most applications of `tcolorbox`, one will not need to know the bells and whistles described herein. You may proceed to Section 9 on page 124 where the customization options for most users are documented.

The following explanations also cover options and settings from the LIB skins library, see Section 9 on page 124.

8.1 Skins and Drawing Engines

From a technical point of view, a *skin* is a style definition for the appearance of a `tcolorbox`. The core package provides some additional option keys for skins but only two skins called `standard`→P. 176 and `standard jigsaw`→P. 177. The LIB skins library adds several more skins. To change to a skin, only one option from the core package has to be set.

/tcb/skin=⟨*name*⟩ (style, no default, initially `standard`)
Sets the current skin to ⟨*name*⟩. This is a style definition which sets all the following keys, i. e. for many use cases there is nothing more to do.

```
\tcbset{colback=Salmon!50!white,colframe=FireBrick!75!black,
  width=(\linewidth-8mm)/2,before=,after=\hfill,equal height group=ske}

\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}
\begin{tcolorbox}[skin=beamer,beamer,adjusted title=My title]
  This is my content.
\end{tcolorbox}
```

My title
This is my content.

My title
This is my content.

/tcb/skin first=⟨*name*⟩ (style, no default, initially `standard`)
If the box is set to be `/tcb/breakable`→P. 306 and *is* broken actually, then the skin for the *first* part of the break sequence is set to ⟨*name*⟩, see Subsection 15.7 on page 317. Typically, this key is set by a `/tcb/skin`.

/tcb/skin middle=⟨*name*⟩ (style, no default, initially `standard`)
If the box is set to be `/tcb/breakable`→P. 306 and *is* broken actually, then the skin for the *middle* parts (if any) of the break sequence is set to ⟨*name*⟩, see Subsection 15.7 on page 317. Typically, this key is set by a `/tcb/skin`.

/tcb/skin last=⟨*name*⟩ (style, no default, initially `standard`)
If the box is set to be `/tcb/breakable`→P. 306 and *is* broken actually, then the skin for the *last* part of the break sequence is set to ⟨*name*⟩, see Subsection 15.7 on page 317. Typically, this key is set by a `/tcb/skin`.

/tcb/graphical environment=⟨*name*⟩ (no default, initially `pgfpicture`)
Sets the graphical environment for the `tcolorbox` to ⟨*name*⟩. Feasible values are `pgfpicture` and `tikzpicture` or environments which inherit from one of these two. This key is set by a `/tcb/skin` and may seldom be used directly.

The skin of a `tcolorbox` is drawn by up to four *engines*. Afterwards, the text content is drawn which is not part of a skin. The four steps are:

1. The *frame* of the box, drawn by `/tcb/frame engine`.
2. The *interior* of the box. The interior of a box with title is drawn differently from a box without title. `/tcb/interior titled engine` or `/tcb/interior engine`→P. 114 is used to draw the interior.
3. The *segmentation* (line) of the box, if there is a lower part; drawn by `/tcb/segmentation engine`→P. 114.
4. The *title area* of the box, if there is a title and `/tcb/title filled`→P. 26 is set to `true`; drawn by `/tcb/title engine`→P. 114.

`/tcb/frame engine`=⟨*name*⟩ (no default, initially `standard`)

Sets the *frame* drawing engine for a box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`→P. 112. Feasible values for ⟨*name*⟩ are:

- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathjigsaw`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathfirst`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathfirstjigsaw`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathmiddle`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathmiddlejigsaw`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathlast`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `pathlastjigsaw`: a `tikz` path which is controlled by `/tcb/frame style`→P. 124,
- `freelance`: deprecated.
- `spartan`: a quite spartan code.
- `empty`: draw nothing.

`/tcb/interior titled engine`=⟨*name*⟩ (no default, initially `standard`)

Sets the *interior* drawing engine for a titled box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`→P. 112. Feasible values for ⟨*name*⟩ are:

- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `pathfirst`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `pathmiddle`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `pathlast`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `freelance`: deprecated.
- `spartan`: a quite spartan code.
- `empty`: draw nothing.

`/tcb/interior engine=`⟨*name*⟩ (no default, initially `standard`)

Sets the *interior* drawing engine for an untitled box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`→P. 112. Feasible values for ⟨*name*⟩ are:

- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `pathfirst`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `pathmiddle`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `pathlast`: a `tikz` path which is controlled by `/tcb/interior style`→P. 125,
- `freelance`: deprecated.
- `spartan`: a quite spartan code.
- `empty`: draw nothing.

`/tcb/segmentation engine=`⟨*name*⟩ (no default, initially `standard`)

Sets the *segmentation* (line) drawing engine for a box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`→P. 112. Feasible values for ⟨*name*⟩ are:

- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/segmentation style`→P. 127,
- `freelance`: deprecated.
- `spartan`: a quite spartan code.
- `empty`: draw nothing.

`/tcb/title engine=`⟨*name*⟩ (no default, initially `standard`)

Sets the *title area* drawing engine for a titled box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`→P. 112. Feasible values for ⟨*name*⟩ are:

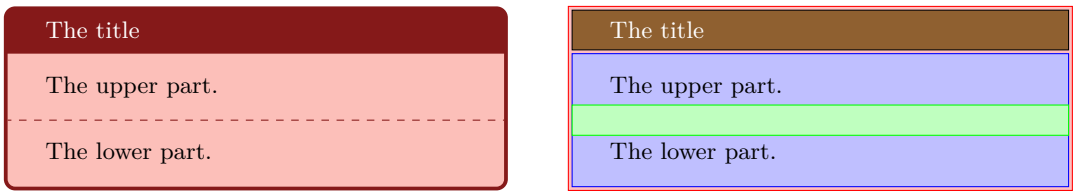
- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/title style`→P. 127,
- `pathfirst`: a `tikz` path which is controlled by `/tcb/title style`→P. 127,
- `pathmiddle`: a `tikz` path which is controlled by `/tcb/title style`→P. 127,
- `pathlast`: a `tikz` path which is controlled by `/tcb/title style`→P. 127,
- `freelance`: deprecated.
- `spartan`: a quite spartan code.
- `empty`: draw nothing.

After an engine is set to an initializing value, the resulting graphical code can be changed using code option keys, see Section 8.2 on page 116.

`/tcb/geometry nodes=true|false` (default `true`, initially `false`)

If set to `true`, up to four `tikz` nodes are defined for a `tcolorbox` which are named `frame`, `interior`, `segmentation`, and `title`. These nodes describe the boundaries of the equally named parts of a `tcolorbox`. They are used by most engines based on Ti*k*Z. Typically, this key is set automatically by a `/tcb/skin` → P. 112.

```
\tcbset{colback=Salmon!50!white,colframe=FireBrick!75!black,
  width=(\linewidth-8mm)/2,before=,after=\hfill,equal height group=geon}

\begin{tcolorbox}[adjusted title=The title]
  The upper part.\tcblower The lower part.
\end{tcolorbox}
\begin{tcolorbox}[enhanced,adjusted title=The title,
  frame code={\path[draw=red,fill=red!25]
      (frame.south west) rectangle (frame.north east);},
  interior titled code={\path[draw=blue,fill=blue!25]
      (interior.south west) rectangle (interior.north east);},
  segmentation code={\path[draw=green,fill=green!25]
      (segmentation.south west) rectangle (segmentation.north east);},
  title code={\path[draw=black,fill=brown!75!black]
      (title.south west) rectangle (title.north east);}]
  The upper part.\tcblower The lower part.
\end{tcolorbox}
```



8.2 Code Option Keys

The following code options are applicable for all skins. The used ⟨*graphical code*⟩ can be any pgf code. For all skins with exception of `standard`→ P. 176 and `standard jigsaw`→ P. 177, the ⟨*graphical code*⟩ can also be any TikZ code.

/tcb/frame code=⟨*graphical code*⟩ (code, default from `standard`)

The given ⟨*graphical code*⟩ is used for drawing the *frame* of the box.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[enhanced,frame code={
  \foreach \n in {north east,north west,south east,south west}
  {\path [fill=red!75!black] (interior.\n) circle (3mm); }; }]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/frame empty (style, no value)

This is a shortcut for setting `/tcb/frame code` to empty. This option removes the drawing of the frame. Alternatively, use `/tcb/frame hidden`→ P. 125.

/tcb/interior titled code=⟨*graphical code*⟩ (code, default from `standard`)

The given ⟨*graphical code*⟩ is used for drawing the *interior* of the box, if the box comes with a title.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,interior titled code={
  \path[draw=red!5!white,line width=5mm,line cap=round]
    ([xshift=3mm,yshift=-3mm]interior.north west)
    --([xshift=-3mm,yshift=3mm]interior.south east)
    ([xshift=3mm,yshift=3mm]interior.south west)
    --([xshift=-3mm,yshift=-3mm]interior.north east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/interior titled empty (style, no value)

This is a shortcut for setting `/tcb/interior titled code` to empty. This option removes the drawing of the untitled interior. Alternatively, use `/tcb/interior hidden`→ P. 126.

/tcb/interior code=⟨*graphical code*⟩ (code, default from **standard**)

The given ⟨*graphical code*⟩ is used for drawing the *interior* of the box, if the box is without a title.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[enhanced,interior code={
  \path[draw=red!5!white,line width=5mm,line cap=round]
    ([xshift=3mm,yshift=-3mm]interior.north west)
    --([xshift=-3mm,yshift=3mm]interior.south east)
    ([xshift=3mm,yshift=3mm]interior.south west)
    --([xshift=-3mm,yshift=-3mm]interior.north east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/interior empty (style, no value)

This is a shortcut for setting **/tcb/interior code** to empty. This option removes the drawing of the interior. Alternatively, use **/tcb/interior hidden** → P. 126.

/tcb/segmentation code=⟨*graphical code*⟩ (code, default from **standard**)

The given ⟨*graphical code*⟩ is used for drawing the *segmentation* area of the box.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,segmentation code={
  \path[top color=red!5!white,bottom color=red!5!white,middle color=blue]
  (segmentation.south west) rectangle (segmentation.north east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/segmentation empty (style, no value)

This is a shortcut for setting **/tcb/segmentation code** to empty. This option removes the drawing of the segmentation line. Alternatively, use **/tcb/segmentation hidden** → P. 127.

/tcb/title code=⟨*graphical code*⟩ (code, default from **standard**)

The given ⟨*graphical code*⟩ is used for drawing the *title* area of the box.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  coltitle=black}

\begin{tcolorbox}[enhanced,title=My title,title code={
  \path[draw=yellow,solid,decorate,line width=2mm,
    decoration={coil,aspect=0,segment length=10.1mm}]
    ([xshift=1mm]title.west) -- ([xshift=-1mm]title.east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/title empty (style, no value)

This is a shortcut for setting **/tcb/title code** to empty. This option removes the drawing of the title area. Alternatively, use **/tcb/title hidden** → P. 128.

8.3 Subskins

A subskin is a new /tcb/skin^{→P. 112} based on an existing skin which is extended or changed.

Never use geometry settings or bounding box options inside a subskin definition! If one skin is replaced by another skin, the overall bounding box should stay constant. Especially, if a skin is used for a breakable box, unpredictable and unpleasant results could arise otherwise. If you want to change the geometry also, use an additional style. See the skin beamer^{→P. 196} and the style /tcb/beamer^{→P. 196} as pattern.

\tcbsubskin{⟨*name*⟩}{⟨*base skin*⟩}{⟨*options*⟩}

Creates a new skin ⟨*name*⟩ which inherits all properties of an existing ⟨*base skin*⟩ plus the given ⟨*options*⟩. The new skin ⟨*name*⟩ can be used as value for the keys /tcb/skin^{→P. 112}, /tcb/skin first^{→P. 112}, /tcb/skin middle^{→P. 112}, and /tcb/skin last^{→P. 112}. As ⟨*base skin*⟩, one can take standard^{→P. 176}, empty^{→P. 205}, enhanced^{→P. 178}, or any skin from the LIB skins library, see Section 9 on page 124.

```
% \tcbuselibrary{skins}
\tcbsubskin{mycross}{empty}{frame code={%
  \draw[red,line width=5pt] (frame.south west)--(frame.north east);
  \draw[red,line width=5pt] (frame.north west)--(frame.south east);},
  skin first=mycross,skin middle=mycross,skin last=mycross }

\begin{tcolorbox}[skin=mycross]
  This is my content.
\end{tcolorbox}
```

This is my content.

/tcb/skin first is subskin of={⟨*base skin*⟩}{⟨*options*⟩} (no default, initially unset)

Creates a new unnamed skin which inherits all properties of an existing ⟨*base skin*⟩ plus the given ⟨*options*⟩. This skin is set as /tcb/skin first^{→P. 112}.
See a detailed example on page 210.

/tcb/skin middle is subskin of={⟨*base skin*⟩}{⟨*options*⟩} (no default, initially unset)

Creates a new unnamed skin which inherits all properties of an existing ⟨*base skin*⟩ plus the given ⟨*options*⟩. This skin is set as /tcb/skin middle^{→P. 112}.
See a detailed example on page 210.

/tcb/skin last is subskin of={⟨*base skin*⟩}{⟨*options*⟩} (no default, initially unset)

Creates a new unnamed skin which inherits all properties of an existing ⟨*base skin*⟩ plus the given ⟨*options*⟩. This skin is set as /tcb/skin last^{→P. 112}.
See a detailed example on page 210.

8.4 Drawing Scheme

Depending on the complexity of a `tcolorbox` definition, the resulting box is drawn in a more or less complex series of steps.

To document and demonstrate these drawing steps, we consider the following box definition:

```
\newtcolorbox{testbox}[1][]{enhanced,title=Test Box,
  boxrule=1mm,titlerule=0.5mm,colframe=blue!50!black,
  interior style={top color=blue!20!green!50!white,bottom color=blue!20!yellow!50!white},
  colbacktitle=blue!50!green!90!white,segmentation style={solid},
  fonttitle=\bfseries,drop fuzzy shadow,borderline={0.3mm}{0.35mm}{yellow!50!white},
  underlay={\path[fill image opacity=0.15,fill image scale=0.9,
    fill stretch picture={\draw[blue,line width=2mm] circle (1);}]
    (interior.south west) rectangle (interior.north east);},
  watermark text={Watermark},watermark color={green!20!white},
  finish={\begin{tcbclipframe}
    \path[bottom color=black,top color=black!50!white,opacity=0.1]
      (frame.south west) -- (frame.south east) -- (frame.north east) -- cycle;
    \path[top color=white,bottom color=black!50!white,opacity=0.1]
      (frame.south west) -- (frame.north east) -- (frame.north west) -- cycle;
    \end{tcbclipframe}},#1}
```

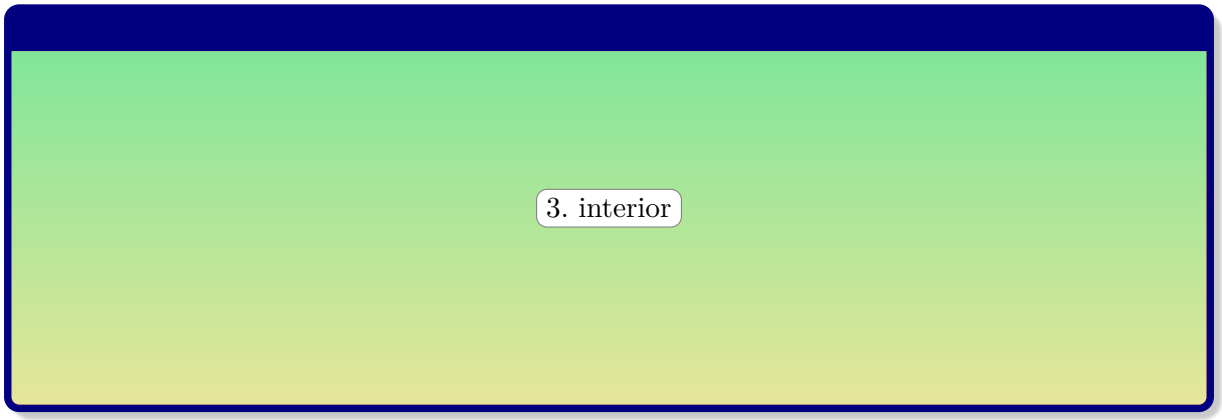
For this definition, we get the maximal number of drawing steps:

1. shadow

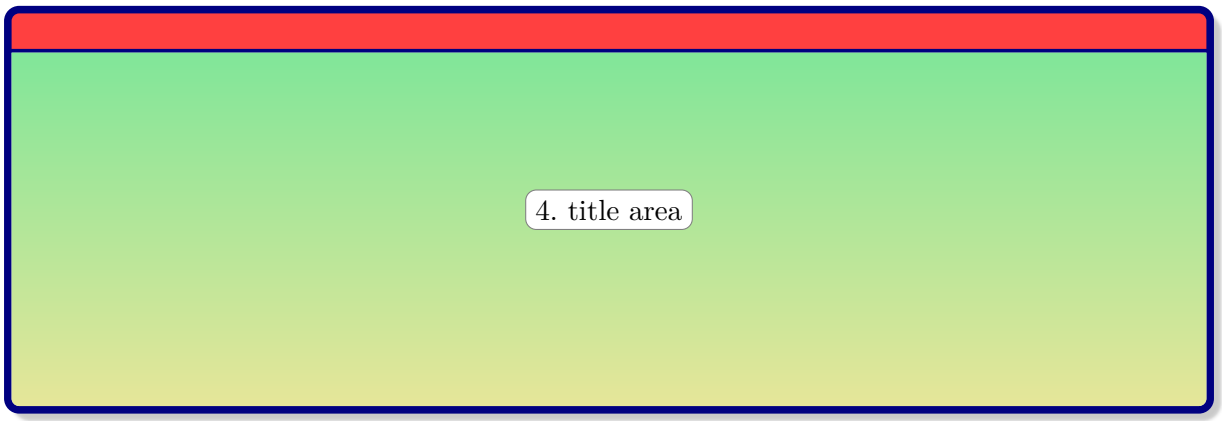
- Section 9.6 on page 156.

2. frame

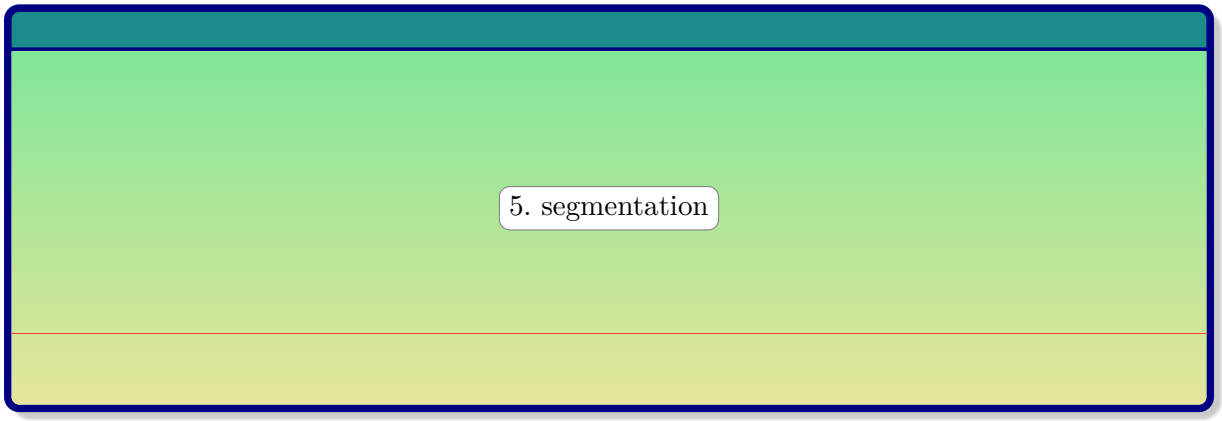
- `/tcb/colframe` → P. 26, `/tcb/opacityframe` → P. 49
- `/tcb/frame code` → P. 116
- `/tcb/frame style` → P. 124



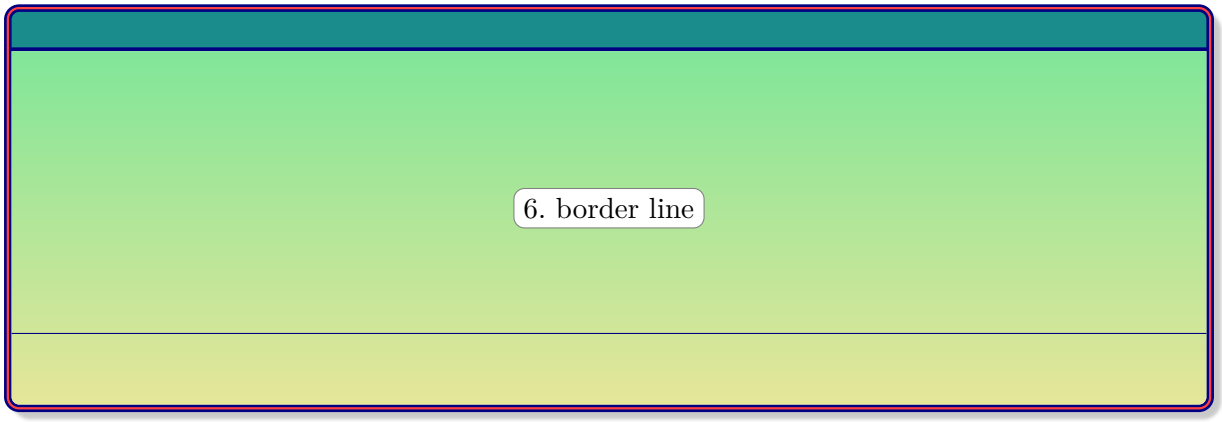
- /tcb/colback → P. 26, /tcb/opacityback → P. 49
- /tcb/interior code → P. 117, /tcb/interior titled code → P. 116
- /tcb/interior style → P. 125



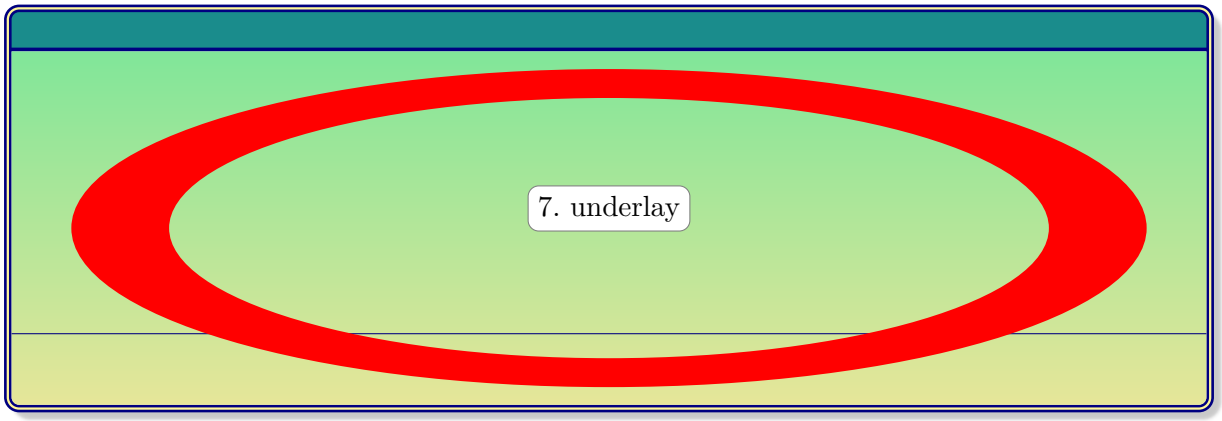
- /tcb/colbacktitle → P. 26, /tcb/opacitybacktitle → P. 49
- /tcb/title code → P. 118
- /tcb/title style → P. 127



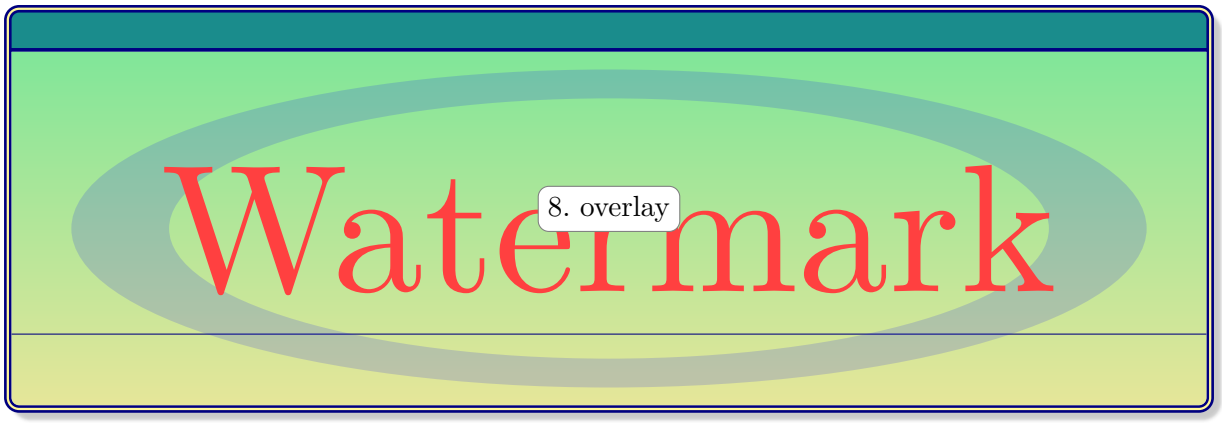
- /tcb/lower separated → P. 24
- /tcb/segmentation code → P. 117
- /tcb/segmentation style → P. 127



- Section 9.5 on page 151



- Section 9.2 on page 131
- Section 9.8 on page 169



- Section 4.12 on page 64
- Section 9.3 on page 139

Test Box

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.

9. text content

Lower part

- `/tcb/colupper`→ P. 27, `/tcb/collower`→ P. 27, `/tcb/coltitle`→ P. 27
- `/tcb/fontupper`→ P. 28, `/tcb/fontlower`→ P. 28, `/tcb/fonttitle`→ P. 28
- `/tcb/opacityupper`→ P. 50, `/tcb/opacitylower`→ P. 50, `/tcb/opacitytitle`→ P. 50

Test Box

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.

10. finish

Lower part

- Section 9.9 on page 171

All together, the box is drawn:

```
% \usepackage{lipsum}
\begin{testbox}
\lipsum[2]
\tcblower
Lower part
\end{testbox}
```

Test Box

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.

Lower part

Watermark

9 Library skins

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{skins}
```

This also loads the package `tikz` [20]. Typically but not necessarily, the following skins use `tikz` instead of `pgf`.

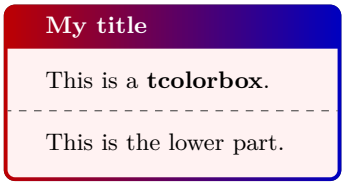
9.1 Style Option Keys

The following style options are applicable for all skins which use engines of type `path`, `pathfirst`, `pathmiddle`, or `pathlast`. Especially, the skin `enhanced` → P. 178 supports *all* of them and `standard` → P. 176 *none*.

/tcb/frame style=⟨`tikz` *keys*⟩ (style, no default)

The ⟨`tikz` *keys*⟩ are used inside the `tikz` path command for drawing the *frame* of the box. This option is available if the `/tcb/frame engine` → P. 113 is set to `path`, `pathfirst`, `pathmiddle`, or `pathlast`. It is *not* available for `standard`.

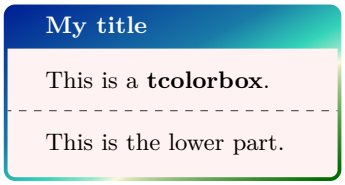
```
\tcbset{colback=red!5!white,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  frame style={left color=red!75!black,
               right color=blue!75!black}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/frame style image=⟨*file name*⟩ (no default, initially unset)

Fills the frame with an external image referenced by ⟨*file name*⟩. For advanced features like blending of a picture with the background, use `/tcb/frame style` together with `/tikz/fill stretch image` → P. 223.

```
\tcbset{colback=red!5!white,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  frame style image=blueshade.png]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/frame style tile={⟨*graphics options*⟩}{⟨*file name*⟩} (no default, initially unset)

Fills the frame with a tile pattern based on an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command. For advanced features like blending of a picture with the background, use /tcb/frame style→P. 124 together with /tikz/fill tile image→P. 227.

```
\tcbset{colback=red!5!white,coltitle=red!30!black,
  opacityback=0.75,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  frame style tile={width=1cm}{pink_marble.png}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/frame hidden (style, no value)

This is a shortcut for `frame style={draw=none,fill=none}`. Depending on the skin, this option switches off the drawing of the frame. Alternatively, use /tcb/frame empty→P. 116.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries,coltitle=black}

\begin{tcolorbox}[enhanced,title=My title,
  frame hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/interior style=⟨tikz *keys*⟩ (style, no default)

The ⟨`tikz` *keys*⟩ are used inside the `tikz` path command for drawing the *interior* of the box. They are used for the titled and for the untitled version as well.

This option is available if the /tcb/interior titled engine→P. 113 or /tcb/interior engine→P. 114 is set to `path`, `pathfirst`, `pathmiddle`, or `pathlast`. It is *not* available for `standard`.

```
\tcbset{colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  interior style={left color=red!20!white,
                  right color=yellow!50!white}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

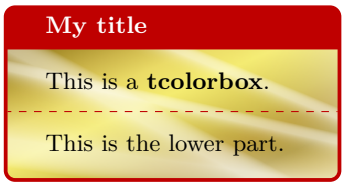
This is a **tcolorbox**.

This is the lower part.

/tcb/interior style image=⟨*file name*⟩ (no default, initially unset)

Fills the interior with an external image referenced by ⟨*file name*⟩. For advanced features like blending of a picture with the background, use /tcb/interior style → P. 125 together with /tikz/fill stretch image → P. 223.

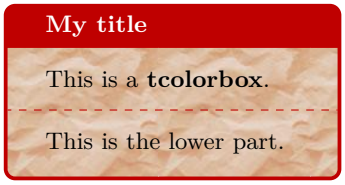
```
\tcbset{colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  interior style image=goldshade.png]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/interior style tile={⟨*graphics options*⟩}{⟨*file name*⟩} (no default, initially unset)

Fills the interior with a tile pattern based on an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying \includegraphics command. For advanced features like blending of a picture with the background, use /tcb/interior style → P. 125 together with /tikz/fill tile image → P. 227.

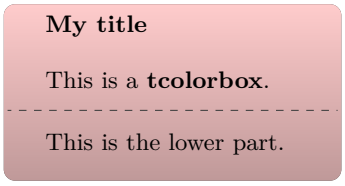
```
\tcbset{colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  interior style tile={width=2cm}{crinklepaper.png}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/interior hidden (style, no value)

This is a shortcut for interior style={draw=none,fill=none}. Depending on the skin, this option switches off the drawing of the interior. Alternatively, use /tcb/interior empty → P. 117 and/or /tcb/interior titled empty → P. 116.

```
\tcbset{frame style={top color=red!20!white,
  bottom color=red!20!white!75!black},
  fonttitle=\bfseries,coltitle=black}

\begin{tcolorbox}[enhanced,title=My title,
  interior hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/segmentation style=⟨tikz *keys*⟩ (style, no default)

The ⟨tikz *keys*⟩ are used inside the tikz path command for drawing the *segmentation* line of the box.

This option is available if the /tcb/segmentation engine →P. 114 is set to path. It is *not* available for standard.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  segmentation style={double=white,draw=blue,
                  double distance=1pt,solid}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/segmentation hidden (style, no value)

This is a shortcut for segmentation style={draw=none,fill=none}. Depending on the skin, this option switches off the drawing of the segmentation line. See also /tcb/lower separated →P. 24 which has the same effect for most skins. Alternatively, use /tcb/segmentation empty →P. 117.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,
  enhanced,segmentation hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

This is the lower part.

/tcb/title style=⟨tikz *keys*⟩ (style, no default)

The ⟨tikz *keys*⟩ are used inside the tikz path command for drawing the *title area* of the box.

This option is available if the /tcb/title engine →P. 114 is set to path, pathfirst, pathmiddle, or pathlast. It is *not* available for standard.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  coltitle=blue!50!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  title style={left color=blue!15!yellow,
               right color=red!85!black}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

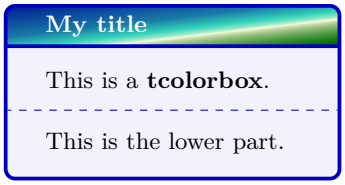
This is a **tcolorbox**.

This is the lower part.

/tcb/title style image=⟨*file name*⟩ (no default, initially unset)

Fills the title area with an external image referenced by ⟨*file name*⟩. For advanced features like blending of a picture with the background, use /tcb/title style→P. 127 together with /tikz/fill stretch image→P. 223.

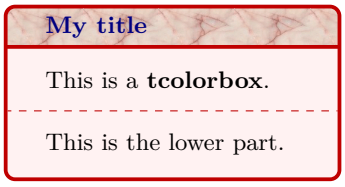
```
\tcbset{colback=blue!5!white,colframe=blue!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  title style image=blueshade.png]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/title style tile={⟨*graphics options*⟩**}{**⟨*file name*⟩**}** (no default, initially unset)

Fills the title area with a tile pattern based on an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command. For advanced features like blending of a picture with the background, use /tcb/title style→P. 127 together with /tikz/fill tile image→P. 227.

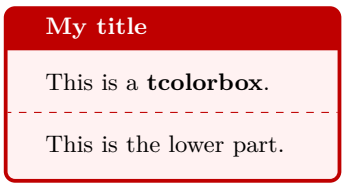
```
\tcbset{colback=red!5!white,colframe=red!75!black,
  coltitle=blue!50!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  title style tile={width=1cm}{pink_marble.png}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tcb/title hidden (style, no value)

This is a shortcut for `title style={draw=none,fill=none}`. Depending on the skin, this option switches off the drawing of the title background. See also /tcb/title filled→P. 26 for a similar effect. Alternatively, use /tcb/title empty→P. 118.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,
  enhanced,title hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



N 2015-01-14 **/tcb/titlerule style=⟨tikz *keys*⟩** (style, no default)

The ⟨tikz *keys*⟩ are used to draw a title rule, i.e. a rule below the optional title. The width of the rule is controlled by /tcb/titlerule → P. 35. It may be set directly to a smaller width to create mixed effects with the standard rule. This option is implemented as an /tcb/underlay → P. 169. Thus, it is not available for standard → P. 176 and standard jigsaw → P. 177, but for all other skins, e.g. enhanced → P. 178. As an underlay, this option can be used multiple times and is removed by /tcb/no underlay → P. 169.

```
\begin{tcolorbox}[enhanced,
  colback=red!5!white,colframe=red!75!black,
  colbacktitle=red!50!yellow,fonttitle=\bfseries,
  title=My title,
  titlerule=1mm,
  titlerule style=yellow  ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

```
\begin{tcolorbox}[enhanced,
  colback=red!5!white,colframe=red!75!black,
  colbacktitle=red!50!yellow,fonttitle=\bfseries,
  title=My title,
  titlerule=1mm,
  titlerule style={yellow,line width=0.5mm}  ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

```
\begin{tcolorbox}[enhanced,
  colback=red!10!white,colframe=red!75!black,
  colbacktitle=red!50!yellow,fonttitle=\bfseries,
  frame hidden,
  title=My title,
  boxrule=0pt,titlerule=1mm,
  titlerule style=red!50!black  ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

```
%\usetikzlibrary{arrows.meta}
\begin{tcolorbox}[empty,
  coltitle=red!75!black,fonttitle=\bfseries,
  borderline horizontal={0.5mm}{0pt}{red!50!white},
  title=My title,
  titlerule style={red,
    arrows = {Hooks[arc=270]-Hooks[arc=270]}} ]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

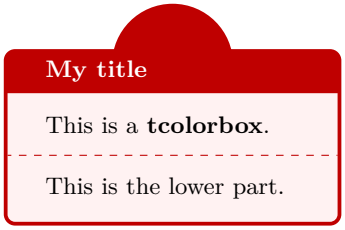
This is a **tcolorbox**.

The combined TikZ style applied to frame, interior, and title background can used by authors in customizing code.

/tikz/tcb fill frame (style, no value)

This is a TikZ style which is finally applied to the *frame* of the box.

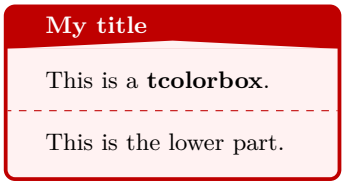
```
% \tcbuselibrary{hooks}
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  frame code app={\path[tcb fill frame]
    ([yshift=-2mm]frame.north)
        circle (8mm); }  }

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tikz/tcb fill interior (style, no value)

This is a TikZ style which is finally applied to the *interior* of the box.

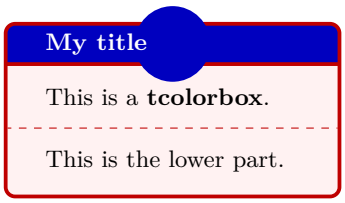
```
% \tcbuselibrary{hooks}
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  interior titled code app={\path[tcb fill interior]
      ([yshift=-0.1pt]interior.north east)
    --([yshift=3pt]interior.north)
    --([yshift=-0.1pt]interior.north west)
    --cycle;}  }

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



/tikz/tcb fill title (style, no value)

This is a TikZ style which is finally applied to the *title area* of the box.

```
% \tcbuselibrary{hooks}
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  colbacktitle=blue!75!black,
  title code app={\path[tcb fill title]
    (title) circle (5mm); }  }

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```



9.2 Boxed Title Option Keys

The following options place the title text into an own `\tcbox`→P. 13. This boxed title can be customized independently from the main box using `/tcb/boxed title style`→P. 133. The placement can be influenced by ⟨*boxtitle options*⟩.

/tcb/attach boxed title to top left=⟨*boxtitle options*⟩ (style, default empty)

The title is boxed with a `\tcbox`→P. 13 and attached to the top left corner of the main box.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top left]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/attach boxed title to top center=⟨*boxtitle options*⟩ (style, default empty)

The title is boxed with a `\tcbox`→P. 13 and attached to the top of the main box.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top center]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/attach boxed title to top right=⟨*boxtitle options*⟩ (style, default empty)

The title is boxed with a `\tcbox`→P. 13 and attached to the top right corner of the main box.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top right]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/attach boxed title to bottom left=⟨*boxtitle options*⟩ (style, default empty)

The title is boxed with a `\tcbox`→P. 13 and attached to the bottom left corner of the main box.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to bottom left]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

My title

/tcb/attach boxed title to bottom center=⟨*boxtitle options*⟩ (style, default empty)

The title is boxed with a `\tcbox`→P. 13 and attached to the bottom of the main box.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to bottom center]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

My title

/tcb/attach boxed title to bottom right=⟨*boxtitle options*⟩ (style, default empty)

The title is boxed with a `\tcbox`→P. 13 and attached to the bottom right corner of the main box.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to bottom right]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

My title

The ⟨*boxtitle options*⟩ of the keys described above are shift values. The dimensions of the boxed title are stored into two macros **\tcboxedtitleheight** and **\tcboxedtitlewidth**. These macros can be used inside the following ⟨*boxtitle options*⟩:

/tcb/boxtitle/xshift=⟨*length*⟩ (no default, initially `0pt`)

The boxed title is shifted by ⟨*length*⟩ in the horizontal direction.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top left={xshift=-2mm},
  boxed title style={size=small,colback=blue}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/boxtitle/yshift=⟨*length*⟩ (no default, initially `0pt`)

The boxed title is shifted by ⟨*length*⟩ in the vertical direction.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top center=
    {yshift=-\tcboxedtitleheight/2},
  boxed title style={size=small,colback=blue}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/boxtitle/yshifttext=⟨*length*⟩ (no default, initially `0pt`)

The text inside the main box by ⟨*length*⟩ to give room for e. g. a sunken title.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=blue}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/boxtitle/yshift*=⟨*length*⟩ (no default, initially `0pt`)

Sets `/tcb/boxtitle/yshift` and `/tcb/boxtitle/yshifttext` the same time. `/tcb/boxtitle/yshifttext` is only set if necessary.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top center={yshift*=-3mm},
  boxed title style={size=small,colback=blue}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

The bounding box of the resulting total `tcolorbox` is adapted automatically to the *vertical* dimensions of the boxed title. Possible horizontal enlargements are *not* automatically computed.

```
\begin{tcolorbox}[enhanced,title=My title,
  attach boxed title to top left=
    {xshift=-2mm,yshift=-2mm},
  boxed title style={size=small,colback=blue},
  show bounding box]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

The boxed title options are implemented as an underlay, see Section 9.8 on page 169. Therefore, a boxed title is not drawn, if a skin does not support underlays like `standard`[→P. 176]. Still, the room for the boxed titles gets reserved in these cases.

A TikZ node `title` is produced by a boxed title which can be used inside `/tcb/frame code`[→P. 116], `/tcb/interior code`[→P. 117], underlays, overlays, and finishes.

A boxed title is almost always the first underlay. The only exceptions are underlays defined by `/tcb/underlay boxed title`[→P. 170] which are drawn before. Additionally, underlays defined by `/tcb/underlay boxed title`[→P. 170] are only drawn, if a boxed title is actually set. They are ignored, if there is no boxed title.

`/tcb/boxed title style`=⟨*options*⟩ (style, initially empty)

By default, a boxed title is dimensioned with `/tcb/size`[→P. 42]=`title` and inherits `/tcb/colframe`[→P. 26] of the main box. Also, the `/tcb/colback`[→P. 26] is inherited from the main `/tcb/colbacktitle`[→P. 26]. Font and color of the title text are set as usual. All other ⟨*options*⟩ are set by the `boxed title style` key. Since a boxed title is set by `\tcbox`[→P. 13], all `tcolorbox` options are applicable here.

```
\begin{tcolorbox}[enhanced,title=My title,
  fonttitle=\bfseries,coltitle=green!25!black,
  attach boxed title to top center=
    {yshift=-2mm,yshifttext=-1mm},
  boxed title style={colframe=green!75!black,
    colback=yellow!50!green}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

```
\begin{tcolorbox}[enhanced,title=My title,
  colframe=red!50!black,colback=red!10!white,
  arc=1mm,colbacktitle=red!10!white,
  fonttitle=\bfseries,coltitle=red!50!black,
  attach boxed title to top left=
    {xshift=3.2mm,yshift=-0.50mm},
  boxed title style={enhanced,
    skin=enhancedfirst jigsaw,
    size=small,arc=1mm,bottom=-1mm,
    interior style={fill=none,
      top color=red!30!white,
      bottom color=red!20!white}}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

```
\begin{tcolorbox}[enhanced,title=My title,
  colframe=blue!50!black,colback=blue!10!white,colbacktitle=blue!5!yellow!10!white,
  fonttitle=\bfseries,coltitle=black,attach boxed title to top center=
    {yshift=-0.25mm-\tcboxedtitleheight/2,yshifttext=2mm-\tcboxedtitleheight/2},
  boxed title style={enhanced,boxrule=0.5mm,
    frame code={ \path[tcb fill frame] ([xshift=-4mm]frame.west)
    -- (frame.north west) -- (frame.north east) -- ([xshift=4mm]frame.east)
    -- (frame.south east) -- (frame.south west) -- cycle; },
    interior code={ \path[tcb fill interior] ([xshift=-2mm]interior.west)
    -- (interior.north west) -- (interior.north east)
    -- ([xshift=2mm]interior.east) -- (interior.south east) -- (interior.south west)
    -- cycle;}  }]
  \lipsum[2]
\end{tcolorbox}
```

My title

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.

```
% \usepackage{varwidth}
\newtcolorbox{mybox}[2][]{enhanced,skin=enhancedlast jigsaw,
  attach boxed title to top left={xshift=-4mm,yshift=-0.5mm},
  fonttitle=\bfseries\sffamily,varwidth boxed title=0.7\linewidth,
  colbacktitle=blue!45!white,colframe=red!50!black,
  interior style={top color=blue!10!white,bottom color=red!10!white},
  boxed title style={empty,arc=0pt,outer arc=0pt,boxrule=0pt},
  underlay boxed title={
    \fill[blue!45!white] (title.north west) -- (title.north east)
      -- +(\tcboxedtitleheight-1mm,-\tcboxedtitleheight+1mm)
      -- ([xshift=4mm,yshift=0.5mm]frame.north east) -- +(0mm,-1mm)
      -- (title.south west) -- cycle;
    \fill[blue!45!white!50!black] ([yshift=-0.5mm]frame.north west)
      -- +(-0.4,0) -- +(0,-0.3) -- cycle;
    \fill[blue!45!white!50!black] ([yshift=-0.5mm]frame.north east)
      -- +(0,-0.3) -- +(0.4,0) -- cycle;  },
  title={#2},#1}

\begin{mybox}{My title}
  \lipsum[2]
\end{mybox}
```

My title

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.

```
% \usepackage{varwidth}
\newtcolorbox{mybox}[2][]{enhanced,
  attach boxed title to top left={xshift=1cm,yshift=-2mm},
  fonttitle=\bfseries,varwidth boxed title=0.7\linewidth,
  colbacktitle=green!45!white,coltitle=green!10!black,colframe=green!50!black,
  interior style={top color=yellow!10!white,bottom color=green!10!white},
  boxed title style={enhanced,boxrule=0.75mm,colframe=white,
    borderline={0.1mm}{0mm}{green!50!black},
    borderline={0.1mm}{0.75mm}{green!50!black},
    interior style={top color=green!10!white,bottom color=green!10!white,
      middle color=green!50!white},
    drop fuzzy shadow},
  title={#2},#1}

\begin{mybox}{My title}
  \lipsum[2]
\end{mybox}
```

My title

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.

```
\newtcolorbox{flipbox}[2][]{
  enhanced,sharp corners=south,
  attach boxed title to bottom center={yshift=0.5mm},
  minipage boxed title*,
  boxed title style={enhanced,size=normal,top=0pt,bottom=0pt,
    interior hidden,sharp corners=north},
  title={#2},#1}

\begin{flipbox}{My title}
\lipsum[2]
\end{flipbox}
```

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.

My title

```
% \usepackage{varwidth}
\newtcolorbox{mybox}[2][]{skin=enhancedlast jigsaw,interior hidden,
  boxsep=0pt,top=0pt,colframe=red,coltitle=red!50!black,
  fonttitle=\bfseries\sffamily,
  attach boxed title to bottom center,
  boxed title style={empty,boxrule=0.5mm},
  varwidth boxed title=0.5\linewidth,
  underlay boxed title={
    \draw[white,line width=0.5mm]
      ([xshift=0.3mm-\tcboxedtitleheight*2,yshift=0.3mm]title.north west)
      --([xshift=-0.3mm+\tcboxedtitleheight*2,yshift=0.3mm]title.north east);
    \path[draw=red,top color=white,bottom color=red!50!white,line width=0.5mm]
    ([xshift=0.25mm-\tcboxedtitleheight*2,yshift=0.25mm]title.north west)
    cos +(\tcboxedtitleheight,-\tcboxedtitleheight/2)
    sin +(\tcboxedtitleheight,-\tcboxedtitleheight/2)
    -- ([xshift=0.25mm,yshift=0.25mm]title.south west)
    -- ([yshift=0.25mm]title.south east)
    cos +(\tcboxedtitleheight,\tcboxedtitleheight/2)
    sin +(\tcboxedtitleheight,\tcboxedtitleheight/2); },
  title={#2},#1}

\begin{mybox}{My title}
  \lipsum[2]
\end{mybox}
```

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.

My title

/tcb/hbox boxed title (no value, initially set)

The title text content is captured with a horizontal box. Especially, there are no linebreak possible.

```
\newtcolorbox{mybox}[1]{hbox boxed title,
  enhanced,attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=red},
  title={#1}}

\begin{mybox}{Short title}
  This is a \textbf{tcolorbox}.
\end{mybox}\bigskip

\begin{mybox}{This title is not really very short}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

Short title
This is a **tcolorbox**.

This title is not really very short
This is a **tcolorbox**.

/tcb/minipage boxed title=⟨*length*⟩ (initially unset)

The title text content is captured with a minipage with a width of ⟨*length*⟩. By default, the resulting boxed title is somewhat smaller than the main box.

```
\newtcolorbox{mybox}[1]{minipage boxed title,
  enhanced,attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=red},
  center title,title={#1}}

\begin{mybox}{Short title}
  This is a \textbf{tcolorbox}.
\end{mybox}\bigskip

\begin{mybox}{This title is not really very short}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

Short title
This is a **tcolorbox**.

This title is not really very short
This is a **tcolorbox**.

/tcb/minipage boxed title*=⟨*length*⟩ (initially unset)

The title text content is captured with a minipage with a width of main box width plus ⟨*length*⟩. By default, the resulting boxed title is somewhat smaller than the main box.

```
\newtcolorbox{mybox}[1]{minipage boxed title*=-2cm,
  enhanced,attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=red},
  center title,title={#1}}

\begin{mybox}{Short title}
  This is a \textbf{tcolorbox}.
\end{mybox}\bigskip

\begin{mybox}{This title is not really very short}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

Short title
This is a **tcolorbox**.

This title is not really very short
This is a **tcolorbox**.

/tcb/tikznode boxed title=⟨*options*⟩ (initially unset)

The title text content is captured with a TikZ node with given TikZ ⟨*options*⟩. The text is centered by default

```
\newtcolorbox{mybox}[1]{tikznode boxed title,
  enhanced,attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=red},
  title={#1}}

\begin{mybox}{Short title}
  This is a \textbf{tcolorbox}.
\end{mybox}\bigskip

\begin{mybox}{This title\\is not really\\very short}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

Short title
This is a **tcolorbox**.

This title
is not really
very short
This is a **tcolorbox**.

/tcb/varwidth boxed title=⟨*length*⟩ (initially unset)

The title text content is captured with a `varwidth` environment with a width of ⟨*length*⟩. This style needs the `varwidth` package [1] to be loaded manually. By default, the resulting boxed title is somewhat smaller than the main box.

```
% \usepackage{varwidth}
\newtcolorbox{mybox}[1]{varwidth boxed title,
  enhanced,attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=red},
  center title,title={#1}}

\begin{mybox}{Short title}
  This is a \textbf{tcolorbox}.
\end{mybox}\bigskip

\begin{mybox}{This title is not really very short}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

Short title
This is a **tcolorbox**.

This title is not really
very short
This is a **tcolorbox**.

/tcb/varwidth boxed title*=⟨*length*⟩ (initially unset)

The title text content is captured with a `varwidth` environment with a width of main box width plus ⟨*length*⟩. This style needs the `varwidth` package [1] to be loaded manually. By default, the resulting boxed title is somewhat smaller than the main box.

```
% \usepackage{varwidth}
\newtcolorbox{mybox}[1]{varwidth boxed title*=-2cm,
  enhanced,attach boxed title to top center=
    {yshift=-3mm,yshifttext=-1mm},
  boxed title style={size=small,colback=red},
  center title,title={#1}}

\begin{mybox}{Short title}
  This is a \textbf{tcolorbox}.
\end{mybox}\bigskip

\begin{mybox}{This title is not really very short}
  This is a \textbf{tcolorbox}.
\end{mybox}
```

Short title
This is a **tcolorbox**.

This title is not
really very short
This is a **tcolorbox**.

9.3 Watermark Option Keys

The following watermark options are applicable for all skins which use `tikzpicture` as `/tcb/graphical environment`→ P. 112. Therefore, the skin `standard`→ P. 176 does not support these watermarks, but all other skins, e. g. `enhanced`→ P. 178.

The watermark options rely on the more general overlay options described in Section 4.12 from page 64. Therefore, *watermarks* and *overlays* cannot be used mixed. But a mixture is possible with the LIB hooks library, see Section 18.

`/tcb/watermark text=`⟨*text*⟩ (no default, initially unset)

Writes some ⟨*text*⟩ in the center of the interior region of a `tcolorbox`. This ⟨*text*⟩ is written *after* the frame and interior are drawn and *before* the text content is drawn. It is zoomed or stretched according the values of `/tcb/watermark zoom`→ P. 142 or `/tcb/watermark stretch`→ P. 144.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark text=My Watermark]
\lipsum[1]
\tcblower
\lipsum[2]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My Watermark

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.

`/tcb/watermark text on=`⟨*part*⟩ `is` ⟨*text*⟩ (no default, initially unset)

This option writes some ⟨*text*⟩ in the center of the interior region of a `tcolorbox` as described for `/tcb/watermark text`. But this is done only for boxes named ⟨*part*⟩ of a break sequence, see `/tcb/breakable`→ P. 306.

Feasible values for ⟨*part*⟩ are:

- `broken`: all broken box parts,
- `unbroken`: unbroken boxes only,
- `first`: first parts of a break sequence,
- `middle`: middle parts of a break sequence,
- `last`: last parts of a break sequence,
- `unbroken and first`: unbroken boxes and first parts of a break sequence,
- `middle and last`: middle and last parts of a break sequence.
- `first and middle`: first and middle parts of a break sequence.

/tcb/watermark graphics=⟨*file name*⟩ (no default, initially unset)

Draws an external picture referenced by ⟨*file name*⟩ in the center of the interior region of a `tcolorbox`. The picture is drawn *after* the frame and interior are drawn and *before* the text content is drawn. It is zoomed or stretched according the values of `/tcb/watermark zoom` → P. 142 or `/tcb/watermark stretch` → P. 144.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark graphics=Basilica_5.png,
  watermark opacity=0.15]
\lipsum[1-2]
\tcblower
This example uses a public domain picture from\\
\url{http://commons.wikimedia.org/wiki/File:Basilica_5.png}
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

This example uses a public domain picture from
http://commons.wikimedia.org/wiki/File:Basilica_5.png

/tcb/watermark graphics on=⟨*part*⟩ `is` ⟨*file name*⟩ (no default, initially unset)

This option draws a picture referenced by ⟨*file name*⟩ in the center of the interior region of a `tcolorbox` as described for `/tcb/watermark graphics`. But this is done only for boxes named ⟨*part*⟩ of a break sequence, see `/tcb/breakable` → P. 306.

Feasible values for ⟨*part*⟩ are:

- **`broken`**: all broken box parts,
- **`unbroken`**: unbroken boxes only,
- **`first`**: first parts of a break sequence,
- **`middle`**: middle parts of a break sequence,
- **`last`**: last parts of a break sequence,
- **`unbroken and first`**: unbroken boxes and first parts of a break sequence,
- **`middle and last`**: middle and last parts of a break sequence.

/tcb/watermark tikz=⟨*graphical code*⟩ (no default, initially unset)

Draws the given `tikz` ⟨*graphical code*⟩ in the center of the interior region of a `tcolorbox`. The code is executed *after* the frame and interior are drawn and *before* the text content is drawn. The result is zoomed or stretched according the values of `/tcb/watermark zoom` → P. 142 or `/tcb/watermark stretch` → P. 144.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  watermark tikz={\draw[line width=2mm] circle (1cm)
    node{\fontfamily{ptm}\fontseries{b}\fontsize{20mm}{20mm}\selectfont ?};}]
\lipsum[1]
\tcblower
\lipsum[2]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

/tcb/watermark tikz on=⟨*part*⟩ `is` ⟨*graphical code*⟩ (no default, initially unset)

This option draws the given `tikz` ⟨*graphical code*⟩ in the center of the interior region of a `tcolorbox` as described for `/tcb/watermark tikz`. But this is done only for boxes named ⟨*part*⟩ of a break sequence, see `/tcb/breakable` → P. 306.

Feasible values for ⟨*part*⟩ are:

- `broken`: all broken box parts,
- `unbroken`: unbroken boxes only,
- `first`: first parts of a break sequence,
- `middle`: middle parts of a break sequence,
- `last`: last parts of a break sequence,
- `unbroken and first`: unbroken boxes and first parts of a break sequence,
- `middle and last`: middle and last parts of a break sequence.

/tcb/no watermark (style, no default, initially set)

Removes the watermark if set before. This is an alias for `/tcb/no overlay` → P. 65.

/tcb/watermark opacity=⟨*fraction*⟩ (no default, initially `1.00`)

Sets the opacity value ∈ [0, 1] for a watermark.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  watermark text=Watermark,nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Opacity 1.00,watermark opacity=1.00]
\lipsum[2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Opacity 0.50,watermark opacity=0.50]
\lipsum[2]
\end{tcolorbox}%
```

Opacity 1.00

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.

Opacity 0.50

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.

/tcb/watermark zoom=⟨*fraction*⟩ (no default, initially `0.75`)

Sets the zoom value for a watermark. The zoom respects the aspect ratio. The value 1.0 means to fill the whole box until the watermark touches the frame.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  watermark text=Watermark,nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Zoom 1.0,watermark zoom=1.0]
\lipsum[2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Zoom 0.5,watermark zoom=0.5]
\lipsum[2]
\end{tcolorbox}%
```

Zoom 1.0

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.

Zoom 0.5

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.

/tcb/watermark shrink=⟨*fraction*⟩ (no default, initially unset)

Identically to /tcb/watermark zoom → P. 142, but the watermark never gets enlarged. Thus, the watermark keeps its original size or is shrunk.

/tcb/watermark overzoom=⟨*fraction*⟩ (no default, initially unset)

Sets the overzoom value for a watermark. The overzoom respects the aspect ratio. The value 1.0 means to fill the whole box until the watermark touches all four sides of the frame.

```
\tcbset{enhanced,colback=white,colframe=blue!50!black,fonttitle=\bfseries,
  watermark opacity=0.5,
  watermark graphics=lichtspiel.jpg,nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Zoom 1.0,watermark zoom=1.0]
\lipsum[1]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Overzoom 1.0,watermark overzoom=1.0]
\lipsum[1]
\end{tcolorbox}%
```

Zoom 1.0

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Overzoom 1.0

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

If a /tcb/watermark overzoom value of 1.0 is used in connection with invisible top and bottom rules which still have a thickness greater than 0pt, the space of these invisible rules may not be covered by the watermark. For example, this situation may occur during the breaking of /tcb/enhanced → P. 178 boxes. To avoid this optical glitch, just set /tcb/pad at break → P. 310 to any desired value.

/tcb/watermark stretch=⟨*fraction*⟩ (no default, initially unset)

Sets the stretch value for a watermark. The stretch value is applied to width and height in relation to the box dimensions. It does not respect the aspect ratio. The value 1.0 means to fill the whole box.

```
\tcbset{enhanced,colback=white,colframe=blue!50!black,fonttitle=\bfseries,
  watermark graphics=lichtspiel.jpg,watermark opacity=0.5,
  nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Stretch 1.00,watermark stretch=1.00]
\lipsum[2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Stretch 0.50,watermark stretch=0.50]
\lipsum[2]
\end{tcolorbox}%
```

Stretch 1.00

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.

Stretch 0.50

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.

/tcb/watermark color=⟨*color*⟩ (no default, initially mixed background and frame color)

Sets the color for the watermark.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark text=My Watermark,
  watermark color=yellow!50!red]
\lipsum[1]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My Watermark

`/tcb/clip watermark=true|false` (default `true`, initially `true`)

Sets the watermark to be clipped to the interior area.

```
\tcbset{enhanced,colback=white,colframe=blue!50!white,fonttitle=\bfseries,
  watermark opacity=0.5,watermark stretch=1.00,arc=3mm,
  watermark graphics=lichtspiel.jpg}

\begin{tcolorbox}[title=Clip (default),clip watermark]
\lipsum[1]
\end{tcolorbox}

\begin{tcolorbox}[title=No clip,clip watermark=false]
\lipsum[1]
\end{tcolorbox}%
```

Clip (default)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

No clip

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

9.4 Clip Environments

The following clip environments are applicable for all skins which use engines of type `path`, `pathfirst`, `pathmiddle`, or `pathlast`. Especially, the skin `enhanced`→ P. 178 supports *all* of them and `standard`→ P. 176 *none*. The typical area of application is inside overlay code, see Section 4.12 from page 64.

```
\begin{tcbclipframe}
 ⟨environment content⟩
\end{tcbclipframe}
```

Defines a `Tikz` scope which clips to the frame area path.

```
\makeatletter
\newtcolorbox{picturebox}[2][]{%
  enhanced,frame hidden,interior hidden,fonttitle=\bfseries,
  overlay={\begin{tcbclipframe}\node at (frame)
    {\includegraphics[width=\tcb@width,height=\tcb@height]{#2}};\end{tcbclipframe}%
    \begin{tcbclipinterior}\fill[white,opacity=0.75]
    (frame.south west) rectangle (frame.north east);\end{tcbclipinterior}},#1}
\makeatother

\begin{picturebox}[title=My Picture Box]{lichtspiel.jpg}
\lipsum[1]
\end{picturebox}
```

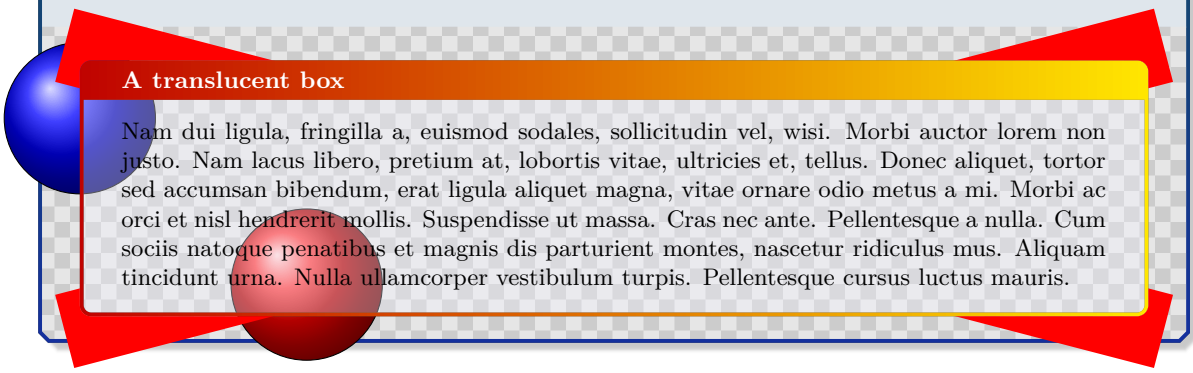
My Picture Box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
\begin{tcbinvclipframe}
 ⟨environment content⟩
\end{tcbinvclipframe}
```

Defines a `Tikz` scope which clips to the *outside* of the frame area path.

```
\tcbset{enhanced jigsaw,fonttitle=\bfseries,opacityback=0.35,colback=blue!5!white,
  frame style={left color=red!75!black,right color=red!10!yellow}}

\begin{tikzpicture}% draw two balls
  \path[use as bounding box] (0,0.8) rectangle +(0.1,0.1);
  \shadedraw [shading=ball] (0,0) circle (1cm);
  \shadedraw [ball color=red] (3,-2.2) circle (1cm);
\end{tikzpicture}

\begin{tcolorbox}[title=A translucent box,
  overlay={\begin{tcbinvclipframe}
    \draw[red,line width=1cm] ([xshift=-2mm,yshift=2mm]frame.north west)
      --([xshift=2mm,yshift=-2mm]frame.south east);
    \draw[red,line width=1cm] ([xshift=-2mm,yshift=-2mm]frame.south west)
      --([xshift=2mm,yshift=2mm]frame.north east);
  \end{tcbinvclipframe}}]
  \lipsum[2]
\end{tcolorbox}
```



```
\begin{tcbclipinterior}
 ⟨environment content⟩
\end{tcbclipinterior}
```

Defines a Tikz scope which clips to the interior area path.

```
\begin{tcolorbox}[enhanced,title=My Title,
  overlay={\begin{tcbclipinterior}
    \draw[red,line width=1cm] (interior.north west)--(interior.south east);
    \draw[red,line width=1cm] (interior.south west)--(interior.north east);
  \end{tcbclipinterior}}]
\lipsum[1]
\end{tcolorbox}
```

My Title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
\begin{tcbcliptitle}
 ⟨environment content⟩
\end{tcbcliptitle}
```

Defines a Tikz scope which clips to the title area path.

```
\begin{tcolorbox}[enhanced,title=My Title,colframe=blue,colback=yellow!10!white,
  overlay={\begin{tcbcliptitle}\node at (title)
  {\includegraphics[width=\linewidth]{lichtspiel.jpg}};\end{tcbcliptitle}}]
\lipsum[1]
\end{tcolorbox}
```

My Title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/clip title=true|false (default `true`, initially `false`)

Sets the title to be clipped to the title area.

```
\tcbset{enhanced,width=5cm,colframe=red!50!white,coltitle=black,
  colbacktitle=yellow!50!white}

\begin{tcolorbox}[title=\mbox{This is a title which is unbreakable and far too long}]
This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[title=\mbox{This is a title which is unbreakable and far too long},
  clip title]
This is a tcolorbox.
\end{tcolorbox}
```

This is a title which is unbreakable and far too long

This is a tcolorbox.

This is a title which is unbreak

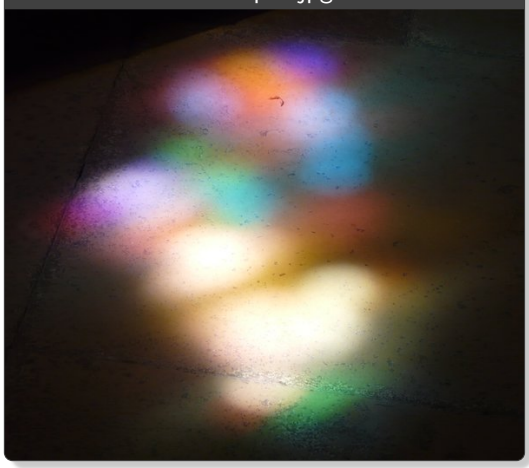
This is a tcolorbox.

/tcb/clip upper=true|false (default `true`, initially `false`)

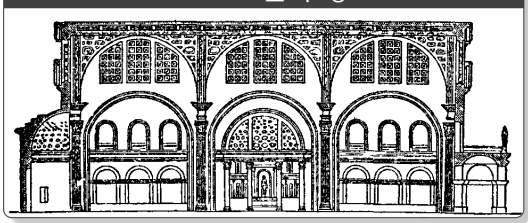
Sets the upper part to be clipped to the interior area.

```
\newcommand{\mygraphics}[2][]{%
  \tcbox[enhanced,boxsep=0pt,top=0pt,bottom=0pt,left=0pt,
    right=0pt,boxrule=0.4pt,drop fuzzy shadow,clip upper,
    colback=black!75!white,toptitle=2pt,bottomtitle=2pt,nobeforeafter,
    center title,fonttitle=\small\sffamily,title=\detokenize{#2}]
  {\includegraphics[width=\the\dimexpr(\linewidth-4mm)/2\relax]{#2}}}

\mygraphics{lichtspiel.jpg}\hfill
\mygraphics{Basilica_5.png}
```

lichtspiel.jpg



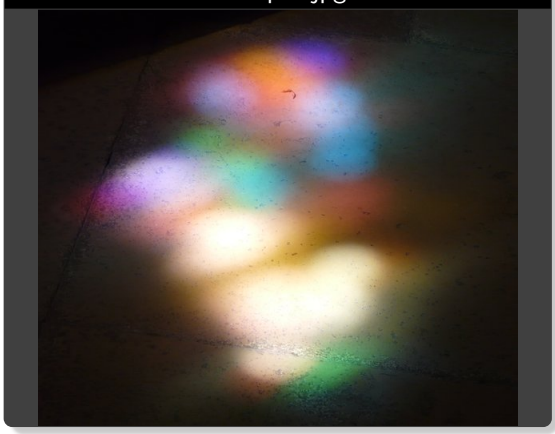
Basilica_5.png



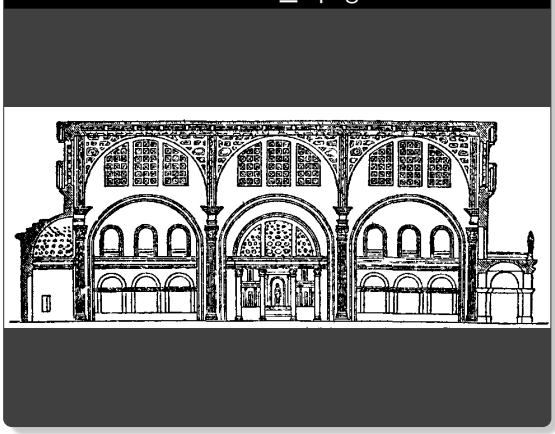
The example for /tcb/clip upper → P. 149 sizes the box according to the dimensions of the picture. To do it the other way around, the watermark options provide an easy solution.

```
\newcommand{\mygraphics}[2][]{%
  \tcbox[enhanced,capture=minipage,boxsep=0pt,top=0pt,bottom=0pt,left=0pt,
    right=0pt,boxrule=0.4pt,drop fuzzy shadow,nobeforeafter,
    colback=black!75!white,toptitle=2pt,bottomtitle=2pt,
    center title,fonttitle=\small\sffamily,title=\detokenize{#2},
    width=(\linewidth-4mm)/2,height=6cm,colbacktitle={black},
    watermark zoom=1.0,watermark graphics={#2}]{}}

\mygraphics{lichtspiel.jpg}\hfill
\mygraphics{Basilica_5.png}
```

lichtspiel.jpg



Basilica_5.png



/tcb/clip lower=true|false (default `true`, initially `false`)

Sets the lower part to be clipped to the interior area.

```
\tcbset{enhanced,width=5cm,colframe=red!50!black,text and listing}

\begin{tcblisting}{}
Donau\-dampf\-schiff\-fahrts\-ka\-pi\-t\"ans\-m\"ut\-zen\-fran\-sen
\end{tcblisting}

\begin{tcblisting}{clip lower}
Donau\-dampf\-schiff\-fahrts\-ka\-pi\-t\"ans\-m\"ut\-zen\-fran\-sen
\end{tcblisting}
```

Donaudampfschifffahrtskapitänsmützenfransen

```
Donau\-dampf\-schiff\-fahrts\-ka\-pi\-t\"ans\-m\"ut\-zen\-fran\-sen
```

Donaudampfschifffahrtskapitänsmützenfransen

```
Donau\-dampf\-schiff\-fa
```

9.5 Border Line Option Keys

The following borderline options are applicable for most skins which use `tikzpicture` as `/tcb/graphical environment`→ P. 112. Therefore, the skin `standard`→ P. 176 does not support these border lines, but most other skins, e. g. `enhanced`→ P. 178.

The borderlines are independent from the normal `tcolorbox` rules. They may be used with or without the `/tcb/segmentation engine`→ P. 114.

The borderlines are stackable, i. e. several different border lines can be used on the same `tcolorbox`. They are drawn *after* the box frame and box interior and *before* overlays or watermarks.

Technically, the normal `tcolorbox` rules result from a TikZ *filling* process. The border lines are created by a TikZ *drawing* process. This can be used to apply different effects.

`/tcb/borderline={`⟨*width*⟩`}{`⟨*offset*⟩`}{`⟨*options*⟩`}` (no default, initially unset)

Adds a new borderline to the stack of border lines. This border line is drawn with the given ⟨*width*⟩ and gets an ⟨*offset*⟩ computed from the frame outline. A positive ⟨*offset*⟩ value moves the borderline inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves it outside without changing the bounding box.

The border line is drawn along a TikZ path with the given TikZ ⟨*options*⟩. Note that the TikZ `line width` option should not be used here.

The border lines adapt to the rounded corners of the `tcolorbox`. An inside borderline will switch to sharp corners if necessary, an outside borderline will always be rounded except for `/tcb/sharp corners`→ P. 46.

```
\begin{tcolorbox}[enhanced,title=Rounded corners,fonttitle=\bfseries,boxsep=5pt,
  arc=8pt,
  borderline={0.5pt}{0pt}{red},
  borderline={0.5pt}{5pt}{blue,dotted},
  borderline={0.5pt}{-5pt}{green} ]
This is a tcolorbox.
\end{tcolorbox}
\bigskip
\begin{tcolorbox}[enhanced,title=Sharp corners,fonttitle=\bfseries,boxsep=5pt,
  arc=8pt,sharp corners=downhill,
  borderline={0.5pt}{0pt}{red},
  borderline={0.5pt}{5pt}{blue,dotted},
  borderline={0.5pt}{-5pt}{green} ]
This is a tcolorbox.
\end{tcolorbox}
```

Rounded corners

This is a tcolorbox.

Sharp corners

This is a tcolorbox.

```latex
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,arc=3mm,boxrule=1.5mm,boxsep=1.5mm,
  colback=yellow!20!white,
  colframe=blue,
  borderline={1mm}{1mm}{white},
  borderline={1mm}{2mm}{red} ]
  \lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```latex
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,arc=3mm,boxrule=1.5mm,
  frame hidden,colback=blue!10!white,
  borderline={1mm}{0mm}{blue,dotted} ]
  \lipsum[2]
\end{tcolorbox}
```

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.

```latex
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,skin=enhancedmiddle,
  frame hidden,interior hidden,top=0mm,bottom=0mm,boxsep=0mm,
  borderline={0.75mm}{0mm}{red},
  borderline={0.75mm}{0.75mm}{red!50!yellow},
  borderline={0.75mm}{1.5mm}{yellow}, ]
  \lipsum[3]
\end{tcolorbox}
```

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

```
% \usepackage{lipsum}
\newtcolorbox{mygreenbox}[2][]{%
  enhanced,width=\linewidth-6pt,
  enlarge top by=3pt,enlarge bottom by=3pt,
  enlarge left by=3pt,enlarge right by=3pt,
  title={#2},frame hidden,boxrule=0pt,top=1mm,bottom=1mm,
  colframe=green!30!black, colbacktitle=green!50!yellow,
  coltitle=black, colback=green!25!white,
  borderline={0.5pt}{-0.5pt}{green!75!blue},
  borderline={1pt}{-3pt}{green!50!blue},#1}

\begin{mygreenbox}{My title}
  \lipsum[4]
\end{mygreenbox}
```

My title

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

/tcb/no borderline (no default, initially set)

Removes all borderlines if set before.

/tcb/show bounding box=⟨*color*⟩ (default `red`, initially unset)

Displays the bounding box borderline of a `tcolorbox`. Its intended use is debugging and fine tuning. It should not be part of a final document. The optional ⟨*color*⟩ is the base color for the bounding box borderline.

```
\tcbset{enhanced,nobeforeafter,width=4cm,fonttitle=\bfseries}

\begin{tcolorbox}[show bounding box,title=Normal]
This is a tcolorbox.
\end{tcolorbox}%
\begin{tcolorbox}[show bounding box=blue,title=Shadow,drop fuzzy shadow]
This is a tcolorbox.
\end{tcolorbox}%
\begin{tcolorbox}[show bounding box=green,title=Enlarged,drop fuzzy shadow,
  enlarge by=2mm]
This is a tcolorbox.
\end{tcolorbox}
```

Normal

This is a tcolorbox.

Shadow

This is a tcolorbox.

Enlarged

This is a tcolorbox.

The following *partial* borderlines act slightly different from the complete borderlines described before. They ignore rounded corner settings, their length is not modified by their ⟨*offset*⟩, they ignore skin settings but adapt to breakable boxes.

N 2014-10-20 **/tcb/borderline north**={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩} (no default, initially unset)

Adds a new borderline with the given ⟨*width*⟩ to the north of the `tcolorbox`. A positive ⟨*offset*⟩ value moves the borderline inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves it outside without changing the bounding box.

```
\begin{tcolorbox}[enhanced,
  borderline north={2pt}{-2pt}{red}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-20 **/tcb/borderline south**={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩} (no default, initially unset)

Adds a new borderline with the given ⟨*width*⟩ to the south of the `tcolorbox`. A positive ⟨*offset*⟩ value moves the borderline inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves it outside without changing the bounding box.

```
\begin{tcolorbox}[enhanced,
  borderline south={2pt}{-2pt}{red}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-20 **/tcb/borderline east**={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩} (no default, initially unset)

Adds a new borderline with the given ⟨*width*⟩ to the east of the `tcolorbox`. A positive ⟨*offset*⟩ value moves the borderline inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves it outside without changing the bounding box.

```
\begin{tcolorbox}[enhanced,
  borderline east={2pt}{-2pt}{red}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-20 **/tcb/borderline west**={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩} (no default, initially unset)

Adds a new borderline with the given ⟨*width*⟩ to the west of the `tcolorbox`. A positive ⟨*offset*⟩ value moves the borderline inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves it outside without changing the bounding box.

```
\begin{tcolorbox}[enhanced,
  borderline west={2pt}{-2pt}{red}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-20 **/tcb/borderline horizontal={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩}** (no default, initially unset)

Adds a new borderline with the given ⟨*width*⟩ to the north and south of the `tcolorbox`. A positive ⟨*offset*⟩ value moves the borderlines inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves them outside without changing the bounding box.

```
\begin{tcolorbox}[blanker,top=3mm,bottom=3mm,
   borderline horizontal={2pt}{0pt}{red}]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

N 2014-10-20 **/tcb/borderline vertical={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩}** (no default, initially unset)

Adds a new borderline with the given ⟨*width*⟩ to the east and west of the `tcolorbox`. A positive ⟨*offset*⟩ value moves the borderlines inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves them outside without changing the bounding box.

```
\begin{tcolorbox}[blanker,left=3mm,right=3mm,
   borderline vertical={2pt}{0pt}{red}]
  This is a \textbf{tcolorbox}.\\
  My second line.
\end{tcolorbox}
```

This is a **tcolorbox**.
My second line.

```
\begin{tcolorbox}[enhanced,colback=yellow!10!white,boxrule=0pt,frame hidden,
  borderline north={1mm}{-2mm}{red},
  borderline south={1mm}{-2mm}{blue},
  borderline west={1mm}{-2mm}{green},
  borderline east={1mm}{-2mm}{yellow}]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

9.6 Shadow Option Keys

The following shadow options are applicable for most skins which use `tikzpicture` as `/tcb/graphical environment`→ P. 112. Therefore, the skin `standard`→ P. 176 does not support these shadows, but most other skins, e. g. `enhanced`→ P. 178.

The shadows are stackable, i. e. several different shadows can be used on the same `tcolorbox`. They are drawn *before* the box frame is drawn.

`/tcb/no shadow` (no default)

Removes all shadows if set before.

9.6.1 Common Shadows and Halos

`/tcb/drop shadow=`⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

`/tcb/drop fuzzy shadow=`⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop fuzzy shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop fuzzy shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

`/tcb/drop midday shadow=`⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop midday shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop midday shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

/tcb/drop fuzzy midday shadow=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop fuzzy midday shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop fuzzy midday shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

/tcb/halo=⟨*size*⟩ with ⟨*color*⟩ (style, default `0.9mm with yellow`)

Adds a new halo shadow with the given ⟨*color*⟩ which overlaps the colorbox an all sides by ⟨*size*⟩.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own halo,halo]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Another halo,
  halo=2mm with green]
This is a tcolorbox.
\end{tcolorbox}
```

My own halo

This is a tcolorbox.

Another halo

This is a tcolorbox.

/tcb/fuzzy halo=⟨*size*⟩ with ⟨*color*⟩ (style, default `0.9mm with yellow`)

Adds a new fuzzy halo shadow with the given ⟨*color*⟩ which overlaps the colorbox an all sides by ⟨*size*⟩ plus `0.48mm`.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own halo,fuzzy halo]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Another halo,
  fuzzy halo=2mm with green]
This is a tcolorbox.
\end{tcolorbox}
```

My own halo

This is a tcolorbox.

Another halo

This is a tcolorbox.

```
\begin{tcolorbox}[blank,enhanced jigsaw,boxsep=2pt,arc=2pt,
  fuzzy halo=2mm with red!50!white,
  fuzzy halo=1mm with white]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

For all following shadows, the optionally given ⟨*color*⟩ for the shadow can be changed equivalent to the preceding examples.

/tcb/drop shadow southeast=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows. This shadow is identical to `/tcb/drop shadow` → P. 156.

```
\begin{tcolorbox}[drop shadow southeast,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow south=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows. This shadow is identical to `/tcb/drop midday shadow` → P. 156.

```
\begin{tcolorbox}[drop shadow south,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow southwest=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop shadow southwest,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow west=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop shadow west,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow northwest=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop shadow northwest,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow north=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop shadow north,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow northeast=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop shadow northeast,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop shadow east=⟨*color*⟩ (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop shadow east,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow southeast=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows. This shadow is identical to `/tcb/drop fuzzy shadow` → P. 156.

```
\begin{tcolorbox}[drop fuzzy shadow southeast,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow south=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows. This shadow is identical to `/tcb/drop fuzzy midday shadow` → P. 157.

```
\begin{tcolorbox}[drop fuzzy shadow south,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow southwest=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop fuzzy shadow southwest,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow west=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop fuzzy shadow west,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow northwest=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop fuzzy shadow northwest,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow north=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop fuzzy shadow north,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow northeast=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop fuzzy shadow northeast,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

/tcb/drop fuzzy shadow east=⟨*color*⟩ (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows.

```
\begin{tcolorbox}[drop fuzzy shadow east,
  enhanced,colback=red!5!white,colframe=red!75!black]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

9.6.2 Lifted Shadows

/tcb/drop lifted shadow=⟨*color*⟩ (style, default `black!50!white`)

Adds a new lifted shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  boxrule=0.4pt,sharp corners,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop lifted shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop lifted shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

/tcb/drop small lifted shadow=⟨*color*⟩ (style, default `black!50!white`)

Adds a new small lifted shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  boxrule=0.4pt,sharp corners,
  colframe=red!75!black,fonttitle=\bfseries}

\tcbox[drop small lifted shadow,size=fbox]
  {This is a tcolorbox.}
\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop small lifted shadow=black]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

/tcb/drop large lifted shadow=⟨*color*⟩ (style, default `black!50!white`)

Adds a new large lifted shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop large lifted shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop large lifted shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

Another shadow

This is a tcolorbox.

9.6.3 Generic Shadows

`/tcb/shadow={⟨xshift⟩}{⟨yshift⟩}{⟨offset⟩}{⟨options⟩}` (no default)

Adds a new shadow to the stack of shadows. This shadow follows the outline of the `tcolorbox` but is shifted by ⟨*xshift*⟩ and ⟨*yshift*⟩. The ⟨*offset*⟩ value is a distance value from the frame outline. A positive ⟨*offset*⟩ value shrinks the shadow and a negative ⟨*offset*⟩ value enlarges the shadow. The shadow is filled along a TikZ path with the given TikZ ⟨*options*⟩.

The shadows adapt to the rounded corners of the `tcolorbox`. An shrinked shadow will switch to sharp corners if necessary, an enlarged shadow may become more rounded depending on several factors. But `/tcb/sharp corners` → P. 46 have sharp shadows.

Shadows are not considered for the bounding box computation by default. Large shadows may be overlaped by the following content. But, the bounding box can be adapted if necessary.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own shadow,
  shadow={2mm}{-1mm}{0mm}{black!50!white}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  shadow={-1mm}{-2mm}{0mm}{fill=blue,
    opacity=0.5}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Double shadow,
  shadow={-1.5mm}{-1.5mm}{0mm}{fill=blue,
    opacity=0.25},
  shadow={1.5mm}{-1.5mm}{0mm}{fill=red,
    opacity=0.25}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Far shadow,
  shadow={5.5mm}{-3.5mm}{2mm}{fill=black,
    opacity=0.25}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Halo shadow,
  shadow={0mm}{0mm}{-1.5mm}%
     {fill=yellow!75!red,opacity=0.5}]
This is a tcolorbox.
\end{tcolorbox}
```

My own shadow
This is a tcolorbox.

Another shadow
This is a tcolorbox.

Double shadow
This is a tcolorbox.

Far shadow
This is a tcolorbox.

Halo shadow
This is a tcolorbox.

/tcb/fuzzy shadow={⟨*xshift*⟩}{⟨*yshift*⟩}{⟨*offset*⟩}{⟨*step*⟩}{⟨*options*⟩} (no default)

Adds a new fuzzy shadow to the stack of shadows. Actually, this option adds several shadows which appear like a shadow with a fuzzy border. This fuzzy shadow follows the outline of the `tcolorbox` but is shifted by ⟨*xshift*⟩ and ⟨*yshift*⟩. The ⟨*offset*⟩ value is a distance value from the frame outline. A positive ⟨*offset*⟩ value shrinks the shadow and a negative ⟨*offset*⟩ value enlarges the shadow. The {⟨*step*⟩} value describes a shrink offset used for the combination of the partial shadows. The shadow is filled along a TikZ path with the given TikZ ⟨*options*⟩ but any `opacity` value will be ignored.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own shadow,
  fuzzy shadow={2mm}{-1mm}{0mm}{0.1mm}%
               {black!50!white}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  fuzzy shadow={-1mm}{-2mm}{0mm}{0.2mm}%
               {fill=blue}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Double shadow,
  fuzzy shadow={-1.5mm}{-1.5mm}{0mm}{0.1mm}%
               {blue},
  fuzzy shadow={1.5mm}{-1.5mm}{0mm}{0.1mm}%
               {red}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Far shadow,
  fuzzy shadow={5.5mm}{-3.5mm}{0mm}{0.3mm}%
               {black}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Glow shadow,
  fuzzy shadow={0mm}{0mm}{-1.5mm}{0.15mm}%
               {yellow!75!red}]
This is a tcolorbox.
\end{tcolorbox}
```

My own shadow

This is a tcolorbox.

Another shadow

This is a tcolorbox.

Double shadow

This is a tcolorbox.

Far shadow

This is a tcolorbox.

Glow shadow

This is a tcolorbox.

```
\newtcolorbox{mybox}[1][]{enhanced,
  fuzzy shadow={1.0mm}{-1.0mm}{0.12mm}{0mm}{blue!50!white},
  fuzzy shadow={-1.0mm}{-1.0mm}{0.12mm}{0mm}{red!50!white},
  fuzzy shadow={-1.0mm}{1.0mm}{0.12mm}{0mm}{green!50!white},
  fuzzy shadow={1.0mm}{1.0mm}{0.12mm}{0mm}{yellow!50!white},#1
}

\begin{mybox}[title=A multi shadow box]
This is a tcolorbox.
\end{mybox}
```

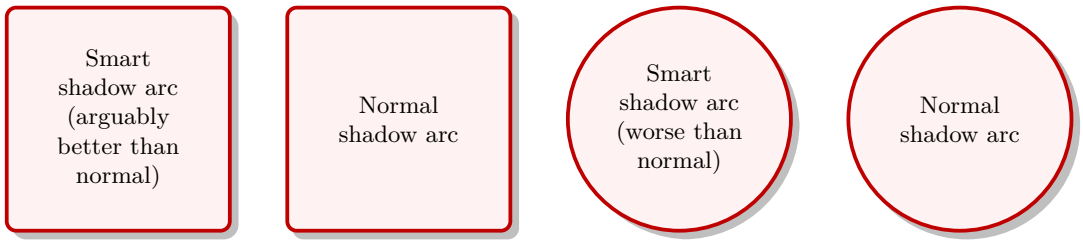
A multi shadow box

This is a tcolorbox.

N 2015-05-05 **/tcb/smart shadow arc=true|false** (default `true`, initially `true`)

If set to `true`, the shadow drawing algorithm tries to do a somewhat smart calculation of the arc for the shadow. The result is pleasing for typical boxes with rounded corners, but gives strange results for circular boxes.

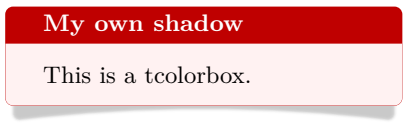
```
\tcbset{enhanced,nobeforeafter,colback=red!5!white,
  colframe=red!75!black,width=3cm,square,halign=center,valign=center
  }

\begin{tcolorbox}[drop shadow]
Smart shadow arc (arguably better than normal)
\end{tcolorbox}
\hfill
\begin{tcolorbox}[smart shadow arc=false, drop shadow]
Normal shadow arc
\end{tcolorbox}
\hfill
\begin{tcolorbox}[circular arc, drop shadow]
Smart shadow arc (worse than normal)
\end{tcolorbox}
\hfill
\begin{tcolorbox}[circular arc, smart shadow arc=false, drop shadow]
Normal shadow arc
\end{tcolorbox}
```



/tcb/lifted shadow={⟨*xshift*⟩}{⟨*yshift*⟩}{⟨*bend*⟩}{⟨*step*⟩}{⟨*options*⟩} (no default)

Adds a new lifted shadow to the stack of shadows. Actually, this option adds several shadows which appear like a shadow with a fuzzy border. This lifted shadow follows the outline of the `tcolorbox` but is shifted by ⟨*xshift*⟩ and ⟨*yshift*⟩ on the lower left corner and by −⟨*xshift*⟩ and ⟨*yshift*⟩ on the lower right corner. Additionally, there is a ⟨*bend*⟩ in the middle. The `{`⟨*step*⟩`}` value describes a shrink offset used for the combination of the partial shadows. The shadow is filled along a Ti*k*Z path with the given Ti*k*Z ⟨*options*⟩ but any `opacity` value will be ignored.

```
\tcbset{enhanced,colback=red!5!white,
  boxrule=0.1pt,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own shadow,
  lifted shadow={1mm}{-2mm}{3mm}{0.1mm}%
              {black!50!white}]
This is a tcolorbox.
\end{tcolorbox}
```



9.6.4 TikZ Shadows

Alternativ to the package shadow options described before, shadows from the «Shadows Library» of TikZ can be used. Such shadows can be added directly to the frame path using `/tcb/frame style` → P. 124.

```
% \usetikzlibrary{shadows}
\begin{tcolorbox}[enhanced,
  colback=red!5!white,colframe=red!75!black,
  frame style={drop shadow} ]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

```
% \usetikzlibrary{shadows}
\begin{tcolorbox}[enhanced,height=3cm,
  colback=red!5!white,colframe=red!75!black,
  halign=center,valign=center,
  frame style={circular drop shadow} ]
  This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

```
% \usetikzlibrary{shadows}
\begin{tcolorbox}[enhanced,width=2.5cm,
  square,circular arc,
  halign=center,valign=center,
  colback=red!5!white,colframe=red!75!black,
  frame style={circular glow={fill=red}} ]
  tcolorbox
\end{tcolorbox}
```

tcolorbox

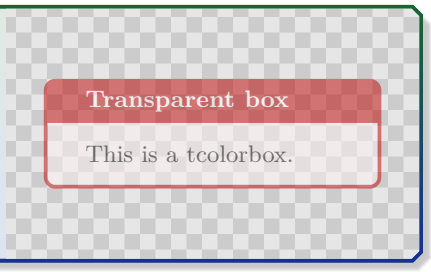
9.7 TikZ Picture Option Keys

The following general options are applicable for skins which use `tikzpicture` as `/tcb/graphical environment`→ P. 112. Therefore, the skin `standard`→ P. 176 does not support these options, but most other skins, e. g. `enhanced`→ P. 178.

`/tcb/tikz`=⟨*tikz option list*⟩ (no default, initially empty)

Adds the given ⟨*tikz option list*⟩ to the main `tikzpicture` environment used to draw the color box, see [20]. If this option is applied a second time, the new ⟨*tikz option list*⟩ is appended to the current option list.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=Transparent box,
  tikz={opacity=0.5,transparency group}]
This is a tcolorbox.
\end{tcolorbox}
```




```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  fontupper=\bfseries\Huge,
  halign title=center,halign=center}

\begin{tcolorbox}[title=Rotated box,
  tikz={rotate=30}]
Sold!
\end{tcolorbox}
```

`/tcb/tikz reset` (initially set)

Removes all options given by `/tcb/tikz`.

`/tcb/at begin tikz`=⟨*tikz code*⟩ (no default, initially empty)

The given ⟨*tikz code*⟩ is executed at the beginning of the `tikzpicture` environment after the TikZ option `execute at begin picture` was applied. If this option is applied a second time, the new ⟨*tikz code*⟩ is appended to the current code.

`/tcb/at begin tikz reset` (initially set)

Removes all code given by `/tcb/at begin tikz`.

`/tcb/at end tikz`=⟨*tikz code*⟩ (no default, initially empty)

The given ⟨*tikz code*⟩ is executed at the ending of the `tikzpicture` environment before the TikZ option `execute at end picture` was applied. If this option is applied a second time, the new ⟨*tikz code*⟩ is appended to the current code.

`/tcb/at end tikz reset` (initially set)

Removes all code given by `/tcb/at end tikz`.

/tcb/rotate=⟨*angle*⟩ (no default, initially unset)

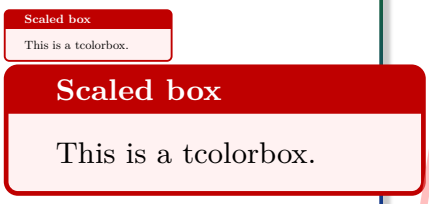
Rotates the `tcolorbox` by the given ⟨*angle*⟩. Note that this is a Ti*k*Z coordinate transformation i.e. not all graphical elements like shadings will really be rotated.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=Rotated box,rotate=30]
This is a tcolorbox.
\end{tcolorbox}
```

/tcb/scale=⟨*fraction*⟩ (no default, initially unset)

Scales the `tcolorbox` by the given ⟨*fraction*⟩. Note that this is a Ti*k*Z coordinate transformation i.e. not all graphical elements like line widths will really be scaled.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=Scaled box,scale=0.5]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[title=Scaled box,scale=1.25]
This is a tcolorbox.
\end{tcolorbox}
```



/tcb/remember (style, initially unset)

Shortcut for `tikz={remember picture}`. This allows one to reference nodes in other Ti*k*Z pictures.

```
\begin{tcolorbox}[enhanced,remember,colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries,title=The four corners of a paper,
  overlay={\draw[red!50!white,line width=1mm,opacity=0.5,shorten >=3mm]
    (frame.north west) edge[->] (current page.north west)
    (frame.north east) edge[->] (current page.north east)
    (frame.south west) edge[->] (current page.south west)
    (frame.south east) edge[->] (current page.south east);}]
This is a tcolorbox.
\end{tcolorbox}
```

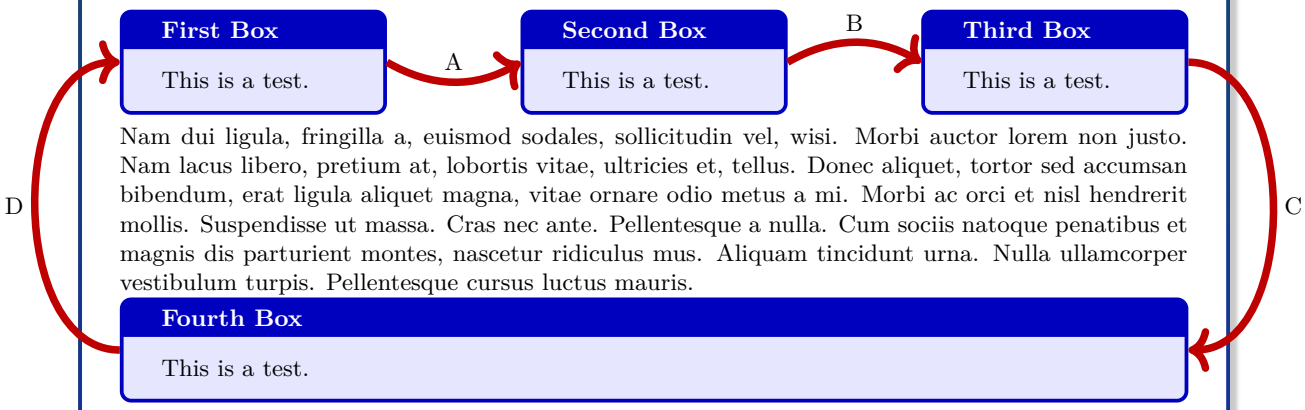
The four corners of a paper

This is a tcolorbox.

/tcb/remember as=⟨*name*⟩ (style, no default, initially unset)

The **frame** node will be remembered by the given ⟨*name*⟩ to be referenced in other TikZ pictures.

```
% \usepackage{lipsum}
\newtcolorbox{mybox}[1][]{enhanced,colframe=blue!75!black,colback=blue!10!white,
  fonttitle=\bfseries,#1}

\begin{mybox}[title=First Box,nobeforeafter,width=\linewidth/4,remember as=one]
This is a test.
\end{mybox}
\hfill
\begin{mybox}[title=Second Box,nobeforeafter,width=\linewidth/4,remember as=two]
This is a test.
\end{mybox}
\hfill
\begin{mybox}[title=Third Box,nobeforeafter,width=\linewidth/4,remember as=three]
This is a test.
\end{mybox}

\lipsum[2]

\begin{mybox}[title=Fourth Box,remember as=four]
This is a test.
\end{mybox}

\begin{tikzpicture}[overlay,remember picture,line width=1mm,draw=red!75!black]
  \draw[->] (one.east) to[bend right] node[above] {A} (two.west);
  \draw[->] (two.east) to[bend left] node[above] {B} (three.west);
  \draw[->] (three.east) to[bend left=90] node[right] {C} (four.east);
  \draw[->] (four.west) to[bend left=90] node[left] {D} (one.west);
\end{tikzpicture}
```



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.

9.8 Underlay Option Keys

Underlays are quite similar to overlays described in Section 4.12 on page 64. Underlays are drawn *after* the frame and interior are drawn and *before* overlays and the text content is drawn; see Section 8.4 on page 120 for the general drawing scheme.

The differences between underlays and overlays are:

- Underlays are not applicable for the skins `standard`[→P. 176] and `standard jigsaw`[→P. 177], whereas overlays are applicable also for these skins. The skin `spartan`[→P. 214] supports underlays but no overlays.

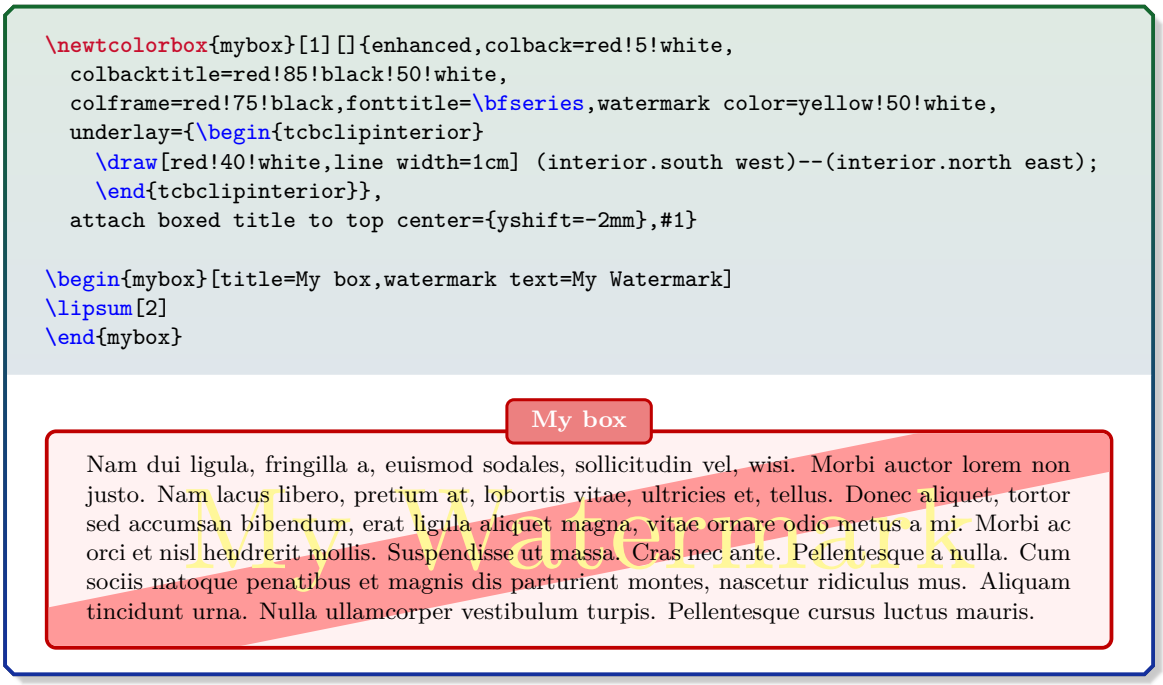
If an underlay is used with the `standard`[→P. 176] skin, it is silently ignored.

- Underlays are stackable, i. e. several different underlays can be used on the same `tcolorbox`. Overlays are not stackable by default (but with some help of the library LIB hooks).
- Boxed titles are implemented with underlays (Section 9.2 on page 131), watermarks are implemented with overlays (Section 9.3 on page 139).

`/tcb/underlay=`⟨*graphical code*⟩ (no default, initially unset)

Adds ⟨*graphical code*⟩ to the box drawing process. This ⟨*graphical code*⟩ is drawn *after* the frame and interior and *before* the text content.

```
\newtcolorbox{mybox}[1][]{enhanced,colback=red!5!white,
  colbacktitle=red!85!black!50!white,
  colframe=red!75!black,fonttitle=\bfseries,watermark color=yellow!50!white,
  underlay={\begin{tcbclipinterior}
    \draw[red!40!white,line width=1cm] (interior.south west)--(interior.north east);
    \end{tcbclipinterior}},
  attach boxed title to top center={yshift=-2mm},#1}

\begin{mybox}[title=My box,watermark text=My Watermark]
\lipsum[2]
\end{mybox}
```



`/tcb/no underlay` (style, no default, initially set)

Removes the underlay if set before.

/tcb/underlay broken=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process. /tcb/underlay→P. 169 overwrites this key.

/tcb/underlay unbroken=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 but *is not* broken actually or if the box is set to be /tcb/unbreakable→P. 307, then the ⟨*graphical code*⟩ is added to the box drawing process. /tcb/underlay→P. 169 overwrites this key.

/tcb/no underlay unbroken (style, no default, initially set)

Removes the unbroken underlay if set before.

/tcb/underlay first=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *first* part of the break sequence. /tcb/underlay→P. 169 overwrites this key.

/tcb/no underlay first (style, no default, initially set)

Removes the first underlay if set before.

/tcb/underlay middle=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *middle* parts (if any) of the break sequence. /tcb/underlay→P. 169 overwrites this key.

/tcb/no underlay middle (style, no default, initially set)

Removes the middle underlay if set before.

/tcb/underlay last=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *last* part of the break sequence. /tcb/underlay→P. 169 overwrites this key.

/tcb/no underlay last (style, no default, initially set)

Removes the last underlay if set before.

/tcb/underlay boxed title=⟨*graphical code*⟩ (no default, initially unset)

If the box has a *boxed title*, see Section 9.2 on page 131, then the ⟨*graphical code*⟩ is added to the box drawing process *before* the boxed title is drawn.

/tcb/no underlay boxed title (style, no default, initially set)

Removes the boxed title underlay if set before.

/tcb/underlay unbroken and first=⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/underlay unbroken and /tcb/underlay first together. /tcb/underlay→P. 169 overwrites this key.

/tcb/underlay middle and last=⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/underlay middle and /tcb/underlay last together. /tcb/underlay→P. 169 overwrites this key.

/tcb/underlay unbroken and last=⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/underlay unbroken and /tcb/underlay last together. /tcb/underlay→P. 169 overwrites this key.

N 2014-09-19 **/tcb/underlay first and middle=**⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/underlay first and /tcb/underlay middle together. /tcb/underlay→P. 169 overwrites this key.

9.9 Finish Option Keys

Finishes are quite similar to underlays described in Section 9.8 on page 169 and overlays described in Section 4.12 on page 64. Finishes are drawn *after* the text content is drawn; see Section 8.4 on page 120 for the general drawing scheme. Therefore, a finish will reduce the readability of the text content.

Finishes are intended for special effects like highlights or glosses or text over text.

- Finishes are only applicable for the skins `enhanced`→P. 178, `empty`→P. 205, `freelance`→P. 217, `bicolor`→P. 191, `beamer`→P. 196, and `widget`→P. 201.

If a finish is used with the `standard`→P. 176 skin, it is silently ignored.

- Finishes are stackable, i. e. several different finishes can be used on the same `tcolorbox`.

`/tcb/finish=`⟨*graphical code*⟩ (no default, initially unset)

Adds ⟨*graphical code*⟩ to the box drawing process. This ⟨*graphical code*⟩ is drawn *after* the text content.

```
\newtcolorbox{mybox}[1][]{enhanced,colback=red!5!white,
  colbacktitle=red!85!black!50!white,colframe=red!75!black,fonttitle=\bfseries,
  finish={\begin{tcbclipframe}
    \path[bottom color=black,top color=black!50!white,opacity=0.1]
      (frame.south west) -- (frame.south east) -- (frame.north east) -- cycle;
    \path[top color=white,bottom color=black!50!white,opacity=0.1]
      (frame.south west) -- (frame.north east) -- (frame.north west) -- cycle;
    \end{tcbclipframe}},#1}

\begin{mybox}[title=My box]
\lipsum[2]
\end{mybox}
```

My box

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.

```
\newtcolorbox{mybox}[1][]{enhanced,colback=red!5!white,
  colbacktitle=red!85!black!50!white,colframe=red!75!black,fonttitle=\bfseries,
  finish={\node[draw,fill=white,fill opacity=0.85,inner sep=5mm,
    rounded corners] at (frame.center) {\Huge\bfseries Finish!};},#1}

\begin{mybox}[title=My box]
\lipsum[2]
\end{mybox}
```

My box

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.

Finish!

/tcb/no finish (style, no default, initially set)

Removes the finish if set before.

/tcb/finish broken=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process. /tcb/finish→P. 171 overwrites this key.

/tcb/finish unbroken=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 but *is not* broken actually or if the box is set to be /tcb/unbreakable→P. 307, then the ⟨*graphical code*⟩ is added to the box drawing process. /tcb/finish→P. 171 overwrites this key.

/tcb/no finish unbroken (style, no default, initially set)

Removes the unbroken finish if set before.

/tcb/finish first=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *first* part of the break sequence. /tcb/finish→P. 171 overwrites this key.

/tcb/no finish first (style, no default, initially set)

Removes the first finish if set before.

/tcb/finish middle=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *middle* parts (if any) of the break sequence. /tcb/finish→P. 171 overwrites this key.

/tcb/no finish middle (style, no default, initially set)

Removes the middle finish if set before.

/tcb/finish last=⟨*graphical code*⟩ (no default, initially unset)

If the box is set to be /tcb/breakable→P. 306 and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *last* part of the break sequence. /tcb/finish→P. 171 overwrites this key.

/tcb/no finish last (style, no default, initially set)

Removes the last finish if set before.

/tcb/finish unbroken and first=⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/finish unbroken and /tcb/finish first together. /tcb/finish→P. 171 overwrites this key.

/tcb/finish middle and last=⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/finish middle and /tcb/finish last together. /tcb/finish→P. 171 overwrites this key.

/tcb/finish unbroken and last=⟨*graphical code*⟩ (no default, initially unset)

This is an abbreviation for setting /tcb/finish unbroken and /tcb/finish last together. /tcb/finish→P. 171 overwrites this key.

N 2014-09-19 **/tcb/finish first and middle=**⟨*graphical code*⟩ (no default, initially unset)

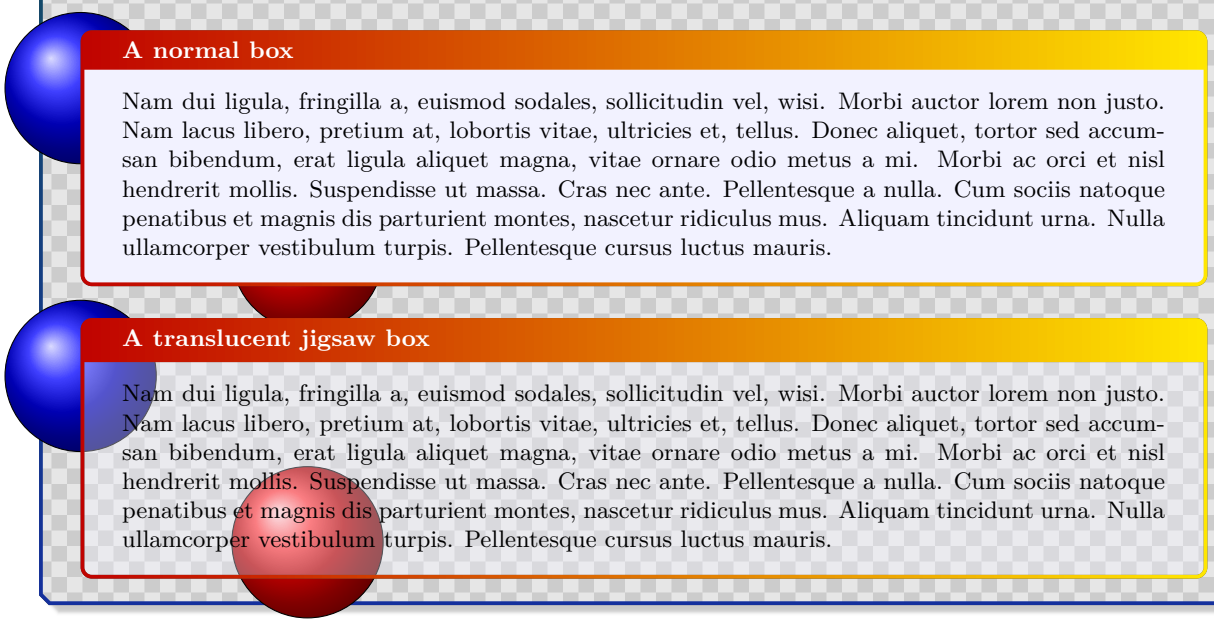
This is an abbreviation for setting /tcb/finish first and /tcb/finish middle together. /tcb/finish→P. 171 overwrites this key.

9.10 Jigsaw Skin Variants

As described in Section 8.1 on page 112, a `tcolorbox` is drawn by up to four *engines*. Typically, the *frame* engine fills the complete box area with color and the other engines fill certain areas with other colors. Finally, only the area which you see as *frame* of the box will display the frame color. For most applications, this is a good approach.

For certain boxes, a more delicate procedure is needed. E.g., if the box should be translucent, an already painted area cannot be made unpainted. Therefore, more elaborate frame engines saw holes into the frame where the interior area and optionally the title area will be painted. The resulting skins are called *jigsaw* skins. For `standard` → P. 176 and `enhanced` → P. 178, there are variants called `standard jigsaw` → P. 177 and `enhanced jigsaw` → P. 185.

```
\newcommand{\ballexample}{\begin{tikzpicture}
  \path[use as bounding box] (0,0.8) rectangle +(0.1,0.1);
  \shadedraw [shading=ball] (0,0) circle (1cm);
  \shadedraw [ball color=red] (3,-2.2) circle (1cm);
\end{tikzpicture}}

\tcbset{enhanced,colback=blue!5!white,
  frame style={left color=red!75!black,right color=red!10!yellow},
  fonttitle=\bfseries }

\ballexample

\begin{tcolorbox}[title=A normal box]
  \lipsum[2]
\end{tcolorbox}

\ballexample

\begin{tcolorbox}[title=A translucent jigsaw box,
  enhanced jigsaw,opacityback=0.35]
  \lipsum[2]
\end{tcolorbox}
```



```
\tcbset{enhanced,colback=red!10!white,coltitle=black,
  frame style={left color=red!75!black,right color=red!10!yellow},
  fonttitle=\bfseries,interior hidden,title hidden}

\begin{tcolorbox}[title=A normal box with hidden interior and title]
  This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[enhanced jigsaw,
  title=A jigsaw box with hidden interior and title]
  This is a tcolorbox.
\end{tcolorbox}
```

A normal box with hidden interior and title

This is a tcolorbox.

A jigsaw box with hidden interior and title

This is a tcolorbox.

```
\newtcolorbox{mybox}{skin=enhancedmiddle jigsaw,leftrule=5mm,rightrule=5mm,
  boxsep=0mm,top=0mm,bottom=0mm,
  frame style={top color=blue,bottom color=red},interior hidden}

\begin{mybox}
  \lipsum[2]
\end{mybox}
```

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.

9.11 Draft Mode

To reduce the compiliation time while drafting a document, the *draft mode* can be applied. Basically, it changes all skins to `spartan`→P. 214 and sets the `/tcb/fit algorithm`→P. 342 to `squeeze`. Especially, when fuzzy shadows are used, the speedup will be considerable high.

It is strongly recommended that the draft mode is *not* used for the final document. Use `spartan`→P. 214 directly, if you want to stay with it. The draft mode implementation may change in future.

Normally, switching to the draft mode should not alter the geometry of your document. Since overlays are deactivated, any code placed there (e.g. counter changes) is not executed anymore! Also, `/tcb/remember as`→P. 168 will not have any effect. You may exclude critical code with `\tcbinterruptdraftmode` / `\tcbcontinuedraftmode` from converting to draft mode.

`\tcbstartdraftmode`

Any following `tcolorbox` code is put into *draft mode*. All skin settings are overruled with `spartan`→P. 214. Overlays, watermarks, shadows, borderlines, and rounded corners are deactivated for all `tcolorbox` layers.

`\tcbstopdraftmode`

The *draft mode* is deactivated for the following code.

`\tcbinterruptdraftmode`

If the compilation is in *draft mode*, the *draft mode* is deactivated until a following `\tcbcontinuedraftmode` is detected.

If the compilation is not in *draft mode*, nothing happens and a following `\tcbcontinuedraftmode` will not start the *draft mode*.

The pair `\tcbinterruptdraftmode` and `\tcbcontinuedraftmode` cannot be used nested.

`\tcbcontinuedraftmode`

Continues the *draft mode* which was suspended by a preceding `\tcbinterruptdraftmode`. Nothing happens, if there was no draft mode before `\tcbinterruptdraftmode`.

Code, which is place between `\tcbinterruptdraftmode` and `\tcbcontinuedraftmode` is shielded from *draft mode*.

`/tcb/draftmode=true|false` (default `true`, initially `false`)

If set to `true`, the *draft mode* is started. If set to `false`, the *draft mode* is stopped.

```
\newtcolorbox{mybeamer}[2][]{beamer,colback=Salmon!50!white,
  colframe=FireBrick!75!black,adjusted title={#2},#1}

\begin{mybeamer}{Beamer box}
This box looks like a box provided by the \texttt{beamer} class.
\end{mybeamer}\par\medskip
\begin{mybeamer}[draftmode]{Beamer box}
This box looks like a box provided by the \texttt{beamer} class.
\end{mybeamer}
```

Beamer box

This box looks like a box provided by the `beamer` class.

Beamer box

This box looks like a box provided by the `beamer` class.

9.12 Skin Family 'standard'

Note that the option keys /tcb/frame style→P. 124, /tcb/interior style→P. 125, /tcb/segmentation style→P. 127, and /tcb/title style→P. 127 are not be applicable to the standard skin. Also, watermarks (see Subsection 9.3) are not usable with the standard skin.

/tcb/skin=standard (skin)

This is the standard skin from the core package. All drawing engines are set to type `standard`. The drawing is based on `pgf` commands and does not need the `tikz` package.

Environment and engines for the skin 'standard'

/tcb/graphical environment→P. 112:	pgfpicture
/tcb/frame engine→P. 113:	standard
/tcb/interior titled engine→P. 113:	standard
/tcb/interior engine→P. 114:	standard
/tcb/segmentation engine→P. 114:	standard
/tcb/title engine→P. 114:	standard

/tcb/standard (style, no value)

This is an abbreviation for setting `skin=standard`.

```
\begin{tcbraster}[standard,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=standard jigsaw (skin)

This is the standard jigsaw skin from the core package. It differs from the skin `standard` →P. 176 by its frame engine, see Section 9.10 on page 173.

Environment and engines for the skin 'standard jigsaw'

/tcb/graphical environment →P. 112:	pgfpicture
/tcb/frame engine →P. 113:	standardjigsaw
/tcb/interior titled engine →P. 113:	standard
/tcb/interior engine →P. 114:	standard
/tcb/segmentation engine →P. 114:	standard
/tcb/title engine →P. 114:	standard

/tcb/standard jigsaw (style, no value)

This is an abbreviation for setting `skin=standard jigsaw`.

```
\begin{tcbraster}[standard jigsaw,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    opacityframe=0.5,opacityback=0.5,opacitybacktitle=0.5,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

9.13 Skin Family 'enhanced'

If you like the standard appearance of a `tcolorbox` but you want to have some 'enhanced' features, the `enhanced` skin is what you are looking for.

/tcb/skin=enhanced (skin)

This skin translates the drawing commands of the core package into `tikz` path commands. Therefore, it allows all `tikz` high level options for these paths and has more flexibility compared to the `standard`→P. 176 skin. You pay for this with some prolonged compilation time. The `tikz` path options can be given with the option keys `/tcb/frame style`→P. 124, `/tcb/interior style`→P. 125, `/tcb/segmentation style`→P. 127, and `/tcb/title style`→P. 127.

Environment and engines for the skin 'enhanced'

`/tcb/graphical environment`→P. 112:	`tikzpicture`
`/tcb/frame engine`→P. 113:	`path`
`/tcb/interior titled engine`→P. 113:	`path`
`/tcb/interior engine`→P. 114:	`path`
`/tcb/segmentation engine`→P. 114:	`path`
`/tcb/title engine`→P. 114:	`path`

/tcb/enhanced (style, no value)

This is an abbreviation for setting `skin=enhanced`.

```
\begin{tcbraster}[enhanced,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

```latex
% \usetikzlibrary{shadings}          % preamble
\tcbset{skin=enhanced,fonttitle=\bfseries,
  frame style={upper left=blue,upper right=red,lower left=yellow,lower right=green},
  interior style={white,opacity=0.5},
  segmentation style={black,solid,opacity=0.2,line width=1pt}}

\begin{tcolorbox}[title=Nice box in rainbow colors]
  With the 'enhanced' skin, it is quite easy to produce fancy looking effects.
  \tcblower
  Note that this is still a \texttt{tcolorbox}.
\end{tcolorbox}
```

Nice box in rainbow colors

With the 'enhanced' skin, it is quite easy to produce fancy looking effects.

Note that this is still a `tcolorbox`.

```latex
% \usetikzlibrary{decorations.pathmorphing} % preamble
\tcbset{skin=enhanced,fonttitle=\bfseries,boxrule=1mm,
  frame style={draw=FireBrick,fill=Salmon},drop fuzzy shadow,
  interior style={draw=FireBrick,top color=Salmon!10,bottom color=Salmon!20},
  segmentation style={draw=FireBrick,solid,decorate,
        decoration={coil,aspect=0,segment length=10.1mm}}}

\begin{tcblisting}{title=A listing box with shadow and some specials}
Of course, skins can be used for listings also.
\begin{equation}
  \int\limits_1^2 \frac{1}{x}~dx = \ln(2).
\end{equation}
\end{tcblisting}
```

A listing box with shadow and some specials

```
Of course, skins can be used for listings also.
\begin{equation}
  \int\limits_1^2 \frac{1}{x}~dx = \ln(2).
\end{equation}
```

Of course, skins can be used for listings also.

$$\int\limits_1^2 \frac{1}{x}\,dx = \ln(2). \tag{2}$$

/tcb/enhanced standard (style, no value)

For unbreakable boxes, this is identical to using /tcb/enhanced→ P. 178. But, for breakable boxes, the *break sequence* is identical to the standard→ P. 176 skin, see Section 15.7 from page 317.

/tcb/blank (style, initially unset)

This style relies on the skin enhanced→ P. 178. All drawing operations are hidden and all margins are set to 0pt. See /tcb/blanker→ P. 206 for switching off the drawing engines.

```
\begin{tcolorbox}[blank,watermark text=A blank box]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

A blank box

\tcbline

Sometimes, a line is only a line. With \tcblower → P. 11 you separate the box content into two functional units. \tcbline draws only a line which looks like the segmentation line between upper and lower part. Furthermore, you can use \tcbline more than just once. \tcbline always uses the path drawing engine. Therefore, the /tcb/segmentation style → P. 127 can be applied.

```
\tcbset{enhanced,colframe=blue!50!black,colback=white}

\begin{tcolorbox}[colupper=red!50!black,collower=green!50!black]
\lipsum[1]
\tcbline
\lipsum[2]
\tcblower
\lipsum[3]
\tcbline
\lipsum[4]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

\tcbline*

Equivalent to \tcbline, but in a breakable box, \tcbline* is removed if at a page/box break. Also, it is removed at the end of a box.

/tcb/skin=enhancedfirst (skin)

This is a flavor of enhanced→P. 178 which is used as a *first* part in a break sequence for enhanced→P. 178. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'enhancedfirst'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathfirst
/tcb/interior titled engine→P. 113:	pathfirst
/tcb/interior engine→P. 114:	pathfirst
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	pathfirst

```
\begin{tcbraster}[skin=enhancedfirst,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

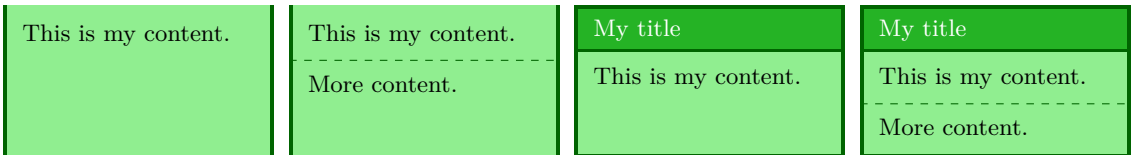
/tcb/skin=enhancedmiddle (skin)

This is a flavor of enhanced→P. 178 which is used as a *middle* part in a break sequence for enhanced→P. 178. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'enhancedmiddle'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathmiddle
/tcb/interior titled engine→P. 113:	pathmiddle
/tcb/interior engine→P. 114:	pathmiddle
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	pathmiddle

```
\begin{tcbraster}[skin=enhancedmiddle,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



/tcb/skin=enhancedlast (skin)

This is a flavor of enhanced→P. 178 which is used as a *last* part in a break sequence for enhanced→P. 178. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'enhancedlast'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathlast
/tcb/interior titled engine→P. 113:	pathlast
/tcb/interior engine→P. 114:	pathlast
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	pathlast

```
\begin{tcbraster}[skin=enhancedlast,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=enhanced jigsaw (skin)

This is the jigsaw variant of skin enhanced→P. 178. It differs by its frame engine, see Section 9.10 on page 173.

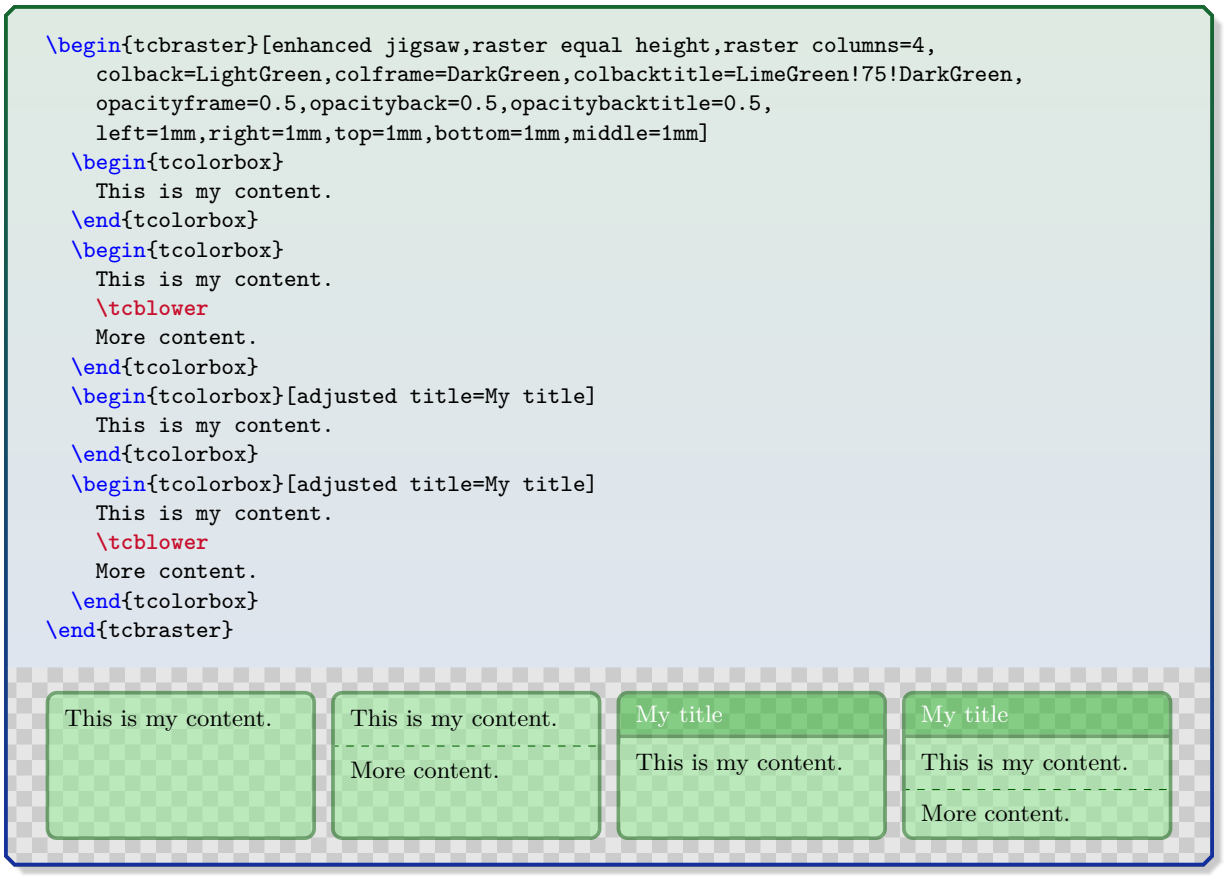
Environment and engines for the skin 'enhanced jigsaw'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathjigsaw
/tcb/interior titled engine→P. 113:	path
/tcb/interior engine→P. 114:	path
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	path

/tcb/enhanced jigsaw (style, no value)

This is an abbreviation for setting skin=enhanced jigsaw.

```
\begin{tcbraster}[enhanced jigsaw,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    opacityframe=0.5,opacityback=0.5,opacitybacktitle=0.5,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



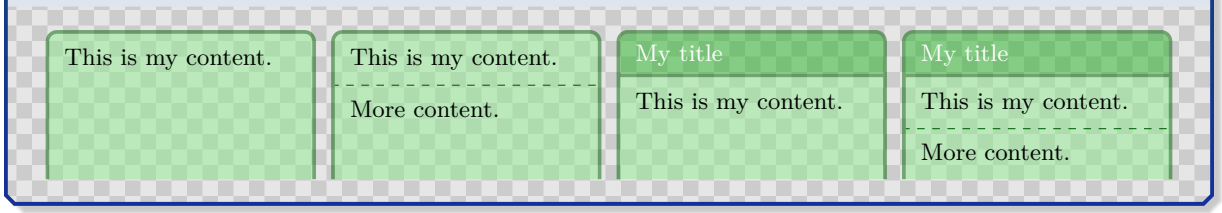
/tcb/skin=enhancedfirst jigsaw (skin)

This is the jigsaw variant of skin enhancedfirst→P. 182. It differs by its frame engine, see Section 9.10 on page 173.

Environment and engines for the skin 'enhancedfirst jigsaw'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathfirstjigsaw
/tcb/interior titled engine→P. 113:	pathfirst
/tcb/interior engine→P. 114:	pathfirst
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	pathfirst

```
\begin{tcbraster}[skin=enhancedfirst jigsaw,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    opacityframe=0.5,opacityback=0.5,opacitybacktitle=0.5,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



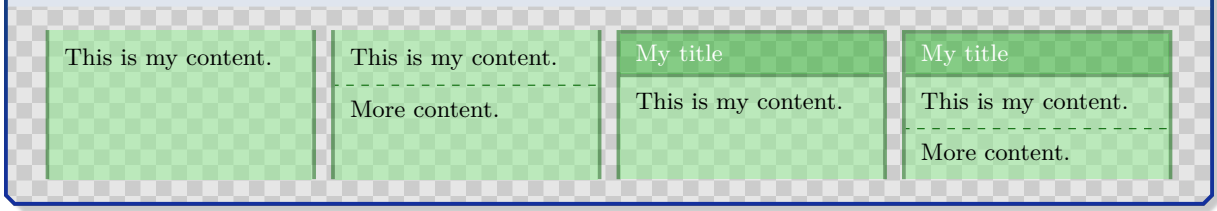
/tcb/skin=enhancedmiddle jigsaw (skin)

This is the jigsaw variant of skin enhancedmiddle→P. 183. It differs by its frame engine, see Section 9.10 on page 173.

Environment and engines for the skin 'enhancedmiddle jigsaw'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathmiddlejigsaw
/tcb/interior titled engine→P. 113:	pathmiddle
/tcb/interior engine→P. 114:	pathmiddle
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	pathmiddle

```
\begin{tcbraster}[skin=enhancedmiddle jigsaw,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    opacityframe=0.5,opacityback=0.5,opacitybacktitle=0.5,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



/tcb/marker (style, no value)

This styles relies on the skin enhancedmiddle jigsaw. It is intended to be used as an optical marker like a highlighter pen.

```
\begin{tcolorbox}[marker]
\lipsum[2]
\end{tcolorbox}
```

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.

This examples demonstrates the creation of several *text marker* environments based on enhancedmiddle → P. 183.

```
\tcbset{textmarker/.style={%
    skin=enhancedmiddle jigsaw,breakable,parbox=false,
    boxrule=0mm,leftrule=5mm,rightrule=5mm,boxsep=0mm,arc=0mm,outer arc=0mm,
    left=3mm,right=3mm,top=1mm,bottom=1mm,toptitle=1mm,bottomtitle=1mm,oversize}}

\newtcolorbox{yellow}{textmarker,colback=yellow!5!white,colframe=yellow}
\newtcolorbox{orange}{textmarker,colback=DarkOrange!5!white,
                      colframe=DarkOrange!75!yellow}
\newtcolorbox{red}{textmarker,colback=red!5!white,colframe=red}
\newtcolorbox{blue}{textmarker,colback=DeepSkyBlue!5!white,colframe=DeepSkyBlue}
\newtcolorbox{green}{textmarker,colback=Chartreuse!5!white,colframe=Chartreuse}
\newtcolorbox{rainbow}{textmarker,interior hidden,
  frame style={top color=blue,bottom color=red,middle color=green}}

\begin{yellow}
  \lipsum[1-3]
\end{yellow}

\begin{orange}
  \lipsum[4]
\end{orange}

\begin{red}
  \lipsum[5]
\end{red}

\begin{green}
  \lipsum[6]
\end{green}

\begin{blue}
  \lipsum[7]
\end{blue}

\begin{rainbow}
  \lipsum[8]
\end{rainbow}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique,

libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

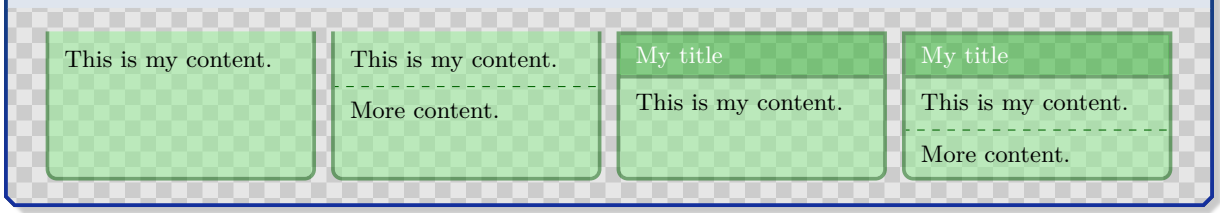
/tcb/skin=enhancedlast jigsaw (skin)

This is the jigsaw variant of skin enhancedlast→P. 184. It differs by its frame engine, see Section 9.10 on page 173.

Environment and engines for the skin 'enhancedlast'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathlastjigsaw
/tcb/interior titled engine→P. 113:	pathlast
/tcb/interior engine→P. 114:	pathlast
/tcb/segmentation engine→P. 114:	path
/tcb/title engine→P. 114:	pathlast

```
\begin{tcbraster}[skin=enhancedlast jigsaw,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    opacityframe=0.5,opacityback=0.5,opacitybacktitle=0.5,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



9.14 Skin Family 'bicolor'

/tcb/skin=bicolor (skin)

This skin is quite similar to the standard[→P. 176] and enhanced[→P. 178] skin. But instead of a segmentation line, the optional lower part of the box is filled with a different color or drawn with a different style.

Environment and engines for the skin 'bicolor'

/tcb/graphical environment[→P. 112]:	tikzpicture
/tcb/frame engine[→P. 113]:	path
/tcb/interior titled engine[→P. 113]:	*special*
/tcb/interior engine[→P. 114]:	*special*
/tcb/segmentation engine[→P. 114]:	*special*
/tcb/title engine[→P. 114]:	path

- The most basic usage of this skin is to set the background color of the lower part by /tcb/colbacklower[→P. 192] and all other options like for the standard[→P. 176] skin.

```
\begin{tcolorbox}[skin=bicolor,title=The title,
    colframe=FireBrick!75!black,colback=Salmon!50!white,colbacklower=Salmon]
  The upper part.
  \tcblower
  The lower part.
\end{tcolorbox}
```

The title

The upper part.

The lower part.

- The more advanced usage of this skin is to apply the /tcb/frame style[→P. 124] and the /tcb/interior style[→P. 125] like for the enhanced[→P. 178] skin. Also, the /tcb/segmentation style[→P. 127] can be used, but it is applied to the whole lower part.

```
\begin{tcolorbox}[skin=bicolor,title=The title,
    frame style={top color=FireBrick,
                 bottom color=FireBrick!15!white,draw=black},
    interior style={left color=Salmon,right color=Salmon!50!white},
    segmentation style={right color=Salmon,left color=Salmon!50!white}]
  The upper part.
  \tcblower
  The lower part.
\end{tcolorbox}
```

The title

The upper part.

The lower part.

/tcb/bicolor (style, no value)

This is an abbreviation for setting skin=bicolor.

```
\begin{tcbraster}[bicolor,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
    colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/colbacklower=⟨*color*⟩ (no default, initially `black!15!white`)

Sets the background ⟨*color*⟩ of the lower part. It depends on the skin, if this value is used.

```
\tcbset{gitexample/.style={listing and comment,comment={#1},
  skin=bicolor,boxrule=1mm,fonttitle=\bfseries,coltitle=black,
  frame style={draw=black,left color=Gold,right color=Goldenrod!50!Gold},
  colback=black,colbacklower=Goldenrod!75!Gold,
  colupper=white,collower=black,
  listing options={language={bash},aboveskip=0pt,belowskip=0pt,nolol,
  basicstyle=\ttfamily\bfseries,extendedchars=true}}}

\begin{tcblisting}{title={Snapshot of the staging area},
  gitexample={The option '-a' automatically stages all tracked and modified
              files before the commit.\par
              This can be combined with the message option '-m'
              as seen in the third line.}}
git commit
git commit -a
git commit -am 'changes to my example'
\end{tcblisting}
```

Snapshot of the staging area

```
git commit
git commit -a
git commit -am 'changes to my example'
```

The option '-a' automatically stages all tracked and modified files before the commit.
This can be combined with the message option '-m' as seen in the third line.

/tcb/skin=bicolorfirst (skin)

This is a flavor of bicolor→P. 191 which is used as a *first* part in a break sequence for bicolor→P. 191. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'bicolorfirst'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathfirst
/tcb/interior titled engine→P. 113:	*special*
/tcb/interior engine→P. 114:	*special*
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	pathfirst

```
\begin{tcbraster}[skin=bicolorfirst,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
    colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

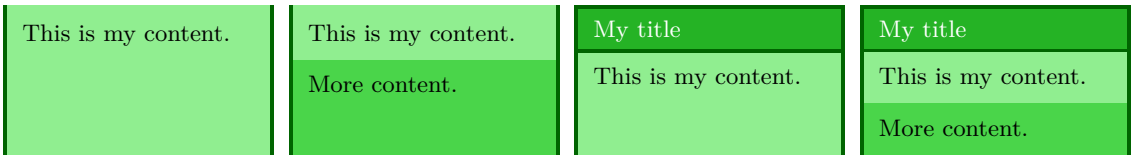
/tcb/skin=bicolormiddle (skin)

This is a flavor of bicolor→ P. 191 which is used as a *middle* part in a break sequence for bicolor→ P. 191. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'bicolormiddle'

/tcb/graphical environment→ P. 112:	tikzpicture
/tcb/frame engine→ P. 113:	pathmiddle
/tcb/interior titled engine→ P. 113:	*special*
/tcb/interior engine→ P. 114:	*special*
/tcb/segmentation engine→ P. 114:	*special*
/tcb/title engine→ P. 114:	pathmiddle

```
\begin{tcbraster}[skin=bicolormiddle,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
    colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



/tcb/skin=bicolorlast (skin)

This is a flavor of bicolor→P. 191 which is used as a *last* part in a break sequence for bicolor→P. 191. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'bicolorlast'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathlast
/tcb/interior titled engine→P. 113:	*special*
/tcb/interior engine→P. 114:	*special*
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	pathlast

```
\begin{tcbraster}[skin=bicolorlast,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
    colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

9.15 Skin Family 'beamer'

/tcb/skin=beamer (skin)

This skin resembles boxes known from the `beamer` class and therefore is called 'beamer'. It uses the normal colors from the core package but shades them a little bit. To use this skin, the `tikz` library `shadings` has to be included in the preamble by:

```
\usetikzlibrary{shadings}
```

The appearance of the skin can be controlled by `/tcb/frame style` → P. 124 and `/tcb/interior style` → P. 125, if needed. Here, the *segmentation* cannot be controlled by a style.

Environment and engines for the skin 'beamer'

`/tcb/graphical environment` → P. 112:	`tikzpicture`
`/tcb/frame engine` → P. 113:	`path`
`/tcb/interior titled engine` → P. 113:	*special*
`/tcb/interior engine` → P. 114:	*special*
`/tcb/segmentation engine` → P. 114:	*special*
`/tcb/title engine` → P. 114:	`path`

/tcb/beamer (style, no value)

This is an abbreviation for setting `skin=beamer`.

It also changes the geometry and some style options.

```
\begin{tcbraster}[beamer,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

```
\begin{tcolorbox}[beamer,colback=Salmon!50!white,colframe=FireBrick!75!black,
  adjusted title=A colored box with the 'beamer' skin]
This box looks like a box provided by the \texttt{beamer} class.
\end{tcolorbox}
```

A colored box with the 'beamer' skin

This box looks like a box provided by the `beamer` class.

```
\begin{tcolorbox}[beamer,colframe=blue,colback=black,
  watermark graphics=lichtspiel.jpg,
  coltext=white,watermark opacity=0.75,watermark stretch=1.0,
  title=Beamer Box with background picture]
\lipsum[1]
\end{tcolorbox}
```

Beamer Box with background picture

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
\newtcolorbox{myblock}[2][]{%
  beamer,breakable,colback=LightBlue,colframe=DarkBlue,#1,title=#2}%

\begin{myblock}{Beamerish \texttt{block}: \texttt{myblock}}
\lipsum[1]
\end{myblock}
```

Beamerish `block`: `myblock`

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/skin=beamerfirst (skin)

This is a flavor of beamer→P. 196 which is used as a *first* part in a break sequence for beamer→P. 196. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'beamerfirst'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathfirst
/tcb/interior titled engine→P. 113:	*special*
/tcb/interior engine→P. 114:	*special*
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	pathfirst

```
\begin{tcbraster}[beamer,skin=beamerfirst,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=beamermiddle (skin)

This is a flavor of beamer→P. 196 which is used as a *middle* part in a break sequence for beamer→P. 196. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'beamermiddle'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathmiddle
/tcb/interior titled engine→P. 113:	*special*
/tcb/interior engine→P. 114:	*special*
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	pathmiddle

```
\begin{tcbraster}[beamer,skin=beamermiddle,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=beamerlast (skin)

This is a flavor of beamer→P. 196 which is used as a *last* part in a break sequence for beamer→P. 196. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'beamerlast'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathlast
/tcb/interior titled engine→P. 113:	*special*
/tcb/interior engine→P. 114:	*special*
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	pathlast

```
\begin{tcbraster}[beamer,skin=beamerlast,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

9.16 Skin Family 'widget'

/tcb/skin=widget (skin)

This skin uses the normal colors from the core package but shades them a little bit. To use this skin, the `tikz` library `shadings` has to be included in the preamble by:

```
\usetikzlibrary{shadings}
```

The appearance of the skin can be controlled by `/tcb/frame style` → P. 124, `/tcb/interior style` → P. 125, and `/tcb/segmentation style` → P. 127, if needed.

Environment and engines for the skin 'widget'

`/tcb/graphical environment` → P. 112:	`tikzpicture`
`/tcb/frame engine` → P. 113:	`path`
`/tcb/interior titled engine` → P. 113:	`path`
`/tcb/interior engine` → P. 114:	`path`
`/tcb/segmentation engine` → P. 114:	*special*
`/tcb/title engine` → P. 114:	*special*

/tcb/widget (style, no value)

This is an abbreviation for setting `skin=widget`.

It also changes the geometry and some style options.

```
\begin{tcbraster}[widget,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

```
\begin{tcolorbox}[widget,colback=Salmon!50!white,colframe=FireBrick!75!black,
  adjusted title=A colored box with the 'widget' skin]
This is my content.
\end{tcolorbox}
```

A colored box with the 'widget' skin

This is my content.

/tcb/skin=widgetfirst (skin)

This is a flavor of widget→P. 201 which is used as a *first* part in a break sequence for widget→P. 201. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'widgetfirst'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathfirst
/tcb/interior titled engine→P. 113:	pathfirst
/tcb/interior engine→P. 114:	pathfirst
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	*special*

```
\begin{tcbraster}[widget,skin=widgetfirst,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=widgetmiddle (skin)

This is a flavor of widget→P. 201 which is used as a *middle* part in a break sequence for widget→P. 201. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'widgetmiddle'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathmiddle
/tcb/interior titled engine→P. 113:	pathmiddle
/tcb/interior engine→P. 114:	pathmiddle
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	*special*

```
\begin{tcbraster}[widget,skin=widgetmiddle,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=widgetlast (skin)

This is a flavor of widget→P. 201 which is used as a *last* part in a break sequence for widget→P. 201. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'widgetlast'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	pathlast
/tcb/interior titled engine→P. 113:	pathlast
/tcb/interior engine→P. 114:	pathlast
/tcb/segmentation engine→P. 114:	*special*
/tcb/title engine→P. 114:	*special*

```
\begin{tcbraster}[widget,skin=widgetlast,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

9.17 Skin Family 'empty'

/tcb/skin=empty (skin)

This skin sets all engines to `empty`, i. e. nothing is drawn at all. Therefore, this skin is a good starting point to create a complete new style by yourself.

Environment and engines for the skin 'empty'

/tcb/graphical environment^{→ P. 112}:	tikzpicture
/tcb/frame engine^{→ P. 113}:	empty
/tcb/interior titled engine^{→ P. 113}:	empty
/tcb/interior engine^{→ P. 114}:	empty
/tcb/segmentation engine^{→ P. 114}:	empty
/tcb/title engine^{→ P. 114}:	empty

/tcb/empty (style, no value)

This is an abbreviation for setting `skin=empty`.

```
\begin{tcbraster}[empty,raster equal height,raster columns=4,
    coltitle=Navy,borderline={2pt}{0pt}{black!10!white},
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/blanker (style, initially unset)

This style relies on the skin `empty` → P. 205. All engines are set to empty and all margins are set to `0pt`. In contrast to `/tcb/blank` → P. 180, the graphical paths are not constructed with exception of the geometry nodes.

```
\begin{tcolorbox}[blanker,watermark text=A blank box]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

A blank box

```
% \tcbuselibrary{fitting}
\newtcboxfit{\mybox}[1]{blanker,width=4cm,height=7cm,top=4pt,
  watermark text=#1}

\begin{tabular}{|c|c|c|}\hline
A & B & C\\\hline
\mybox{A}{\lipsum[1]} & \mybox{B}{\lipsum[2]} & \mybox{C}{\lipsum[3]}\\\hline
\end{tabular}
```

A	B	C
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.	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.	Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

A B C

/tcb/skin=emptyfirst (skin)

This is a flavor of empty→ P. 205 which is used as a *first* part in a break sequence for empty→ P. 205. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'emptyfirst'

/tcb/graphical environment→ P. 112:	tikzpicture
/tcb/frame engine→ P. 113:	empty
/tcb/interior titled engine→ P. 113:	empty
/tcb/interior engine→ P. 114:	empty
/tcb/segmentation engine→ P. 114:	empty
/tcb/title engine→ P. 114:	empty

```
\begin{tcbraster}[empty,skin=emptyfirst,raster equal height,raster columns=4,
    coltitle=Navy,borderline={2pt}{0pt}{black!10!white},
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=emptymiddle (skin)

This is a flavor of empty→P. 205 which is used as a *middle* part in a break sequence for empty→P. 205. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'emptymiddle'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	empty
/tcb/interior titled engine→P. 113:	empty
/tcb/interior engine→P. 114:	empty
/tcb/segmentation engine→P. 114:	empty
/tcb/title engine→P. 114:	empty

```
\begin{tcbraster}[empty,skin=emptymiddle,raster equal height,raster columns=4,
    coltitle=Navy,borderline={2pt}{0pt}{black!10!white},
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

/tcb/skin=emptylast (skin)

This is a flavor of empty→P. 205 which is used as a *last* part in a break sequence for empty→P. 205. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'emptylast'

/tcb/graphical environment→P. 112:	tikzpicture
/tcb/frame engine→P. 113:	empty
/tcb/interior titled engine→P. 113:	empty
/tcb/interior engine→P. 114:	empty
/tcb/segmentation engine→P. 114:	empty
/tcb/title engine→P. 114:	empty

```
\begin{tcbraster}[empty,skin=emptylast,raster equal height,raster columns=4,
    coltitle=Navy,borderline={2pt}{0pt}{black!10!white},
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

This example demonstrates a breakable customized box. Here, we define an environment `freebox`. The first application of `freebox` produces an unbroken `tcolorbox`. The box is drawn by the code given by `/tcb/frame code` →P. 116 and `/tcb/interior code` →P. 117. The second application of `freebox` is broken into several parts which are drawn by the codes given by `/tcb/skin first is subskin of` →P. 119, `/tcb/skin middle is subskin of` →P. 119, and `/tcb/skin last is subskin of` →P. 119.

```
% Preamble:
%\usepackage{tikz,lipsum}
%\tcbuselibrary{skins,breakable}
\tikzset{coltria/.style={fill=red!15!white}}

\newtcolorbox{freebox}[1][]{empty,breakable,leftrule=5mm,left=2mm,
  frame style={fill,top color=red!75!black,bottom color=red!75!black,middle color=red},
  colback=yellow!50!white,
  watermark color=red!50!yellow!75!white,
  watermark text on=unbroken is unbroken box,
  watermark text on=first is first part,
  watermark text on=middle is middle part,
  watermark text on=last is last part,
  % code for unbroken boxes:
  frame code={\path[tcb fill frame] (frame.south west)--(frame.north west)
    --([xshift=-5mm]frame.north east)--([yshift=-5mm]frame.north east)
    --([yshift=5mm]frame.south east)--([xshift=-5mm]frame.south east)--cycle; },
  interior code={\path[tcb fill interior] (interior.south west)--(interior.north west)
    --([xshift=-4.8mm]interior.north east)--([yshift=-4.8mm]interior.north east)
    --([yshift=4.8mm]interior.south east)--([xshift=-4.8mm]interior.south east)
    --cycle; },
  % code for the first part of a break sequence:
  skin first is subskin of={emptyfirst}{%
    frame code={\path[tcb fill frame] (frame.south west)--(frame.north west)
      --([xshift=-5mm]frame.north east)--([yshift=-5mm]frame.north east)
      --(frame.south east)--cycle;
      \path[coltria] ([xshift=2.5mm,yshift=1mm]frame.south west) -- +(120:2mm)
      -- +(60:2mm)-- cycle; },
    interior code={\path[tcb fill interior] (interior.south west|-frame.south)
      --(interior.north west)--([xshift=-4.8mm]interior.north east)
      --([yshift=-4.8mm]interior.north east)--(interior.south east|-frame.south)
      --cycle; },
  },%
  % code for the middle part of a break sequence:
  skin middle is subskin of={emptymiddle}{%
    frame code={\path[tcb fill frame] (frame.south west)--(frame.north west)
      --(frame.north east)--(frame.south east)--cycle;
      \path[coltria] ([xshift=2.5mm,yshift=-1mm]frame.north west) -- +(240:2mm)
        -- +(300:2mm) -- cycle;
      \path[coltria] ([xshift=2.5mm,yshift=1mm]frame.south west) -- +(120:2mm)
        -- +(60:2mm) -- cycle;
      },
    interior code={\path[tcb fill interior] (interior.south west|-frame.south)
      --(interior.north west|-frame.north)--(interior.north east|-frame.north)
      --(interior.south east|-frame.south)--cycle; },
    },
  % code for the last part of a break sequence:
  skin last is subskin of={emptylast}{%
    frame code={\path[tcb fill frame] (frame.south west)--(frame.north west)
      --(frame.north east)--([yshift=5mm]frame.south east)
      --([xshift=-5mm]frame.south east)--cycle;
      \path[coltria] ([xshift=2.5mm,yshift=-1mm]frame.north west) -- +(240:2mm)
      -- +(300:2mm) -- cycle;
      },
    interior code={\path[tcb fill interior] (interior.south west)
```

```
      --(interior.north west|-frame.north)--(interior.north east|-frame.north)
      --([yshift=4.8mm]interior.south east)--([xshift=-4.8mm]interior.south east)
      --cycle; },
    },
  #1}

\begin{freebox}
\lipsum[1]
\end{freebox}

\begin{freebox}
\lipsum[1-12]
\end{freebox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

unbroken box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

first part

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim.

Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

middle part

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus. Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et,

lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

last part

9.18 Skin 'spartan'

/tcb/skin=spartan (skin)

This skin is quite ... spartan. It supports no rounded corners, no overlays, no shadows, no borderlines, and no finishes. The only exception are underlays. One cannot do very fancy things with this skin, but it compiles very fast. Therefore, the `spartan` skin is used for the draft mode, see Section 9.11 on page 175. Nevertheless, it can be used as a normal skin.

Environment and engines for the skin 'spartan'

/tcb/graphical environment →P. 112:	tikzpicture
/tcb/frame engine →P. 113:	spartan
/tcb/interior titled engine →P. 113:	spartan
/tcb/interior engine →P. 114:	spartan
/tcb/segmentation engine →P. 114:	spartan
/tcb/title engine →P. 114:	spartan

/tcb/spartan (style, no value)

This is an abbreviation for setting `skin=spartan`.

```
\begin{tcbraster}[spartan,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```

This is my content.

This is my content.

More content.

My title

This is my content.

My title

This is my content.

More content.

9.19 Skin 'draft'

/tcb/skin=draft (skin)

This skin is intended to be used while drafting new geometric settings for a `tcolorbox`.

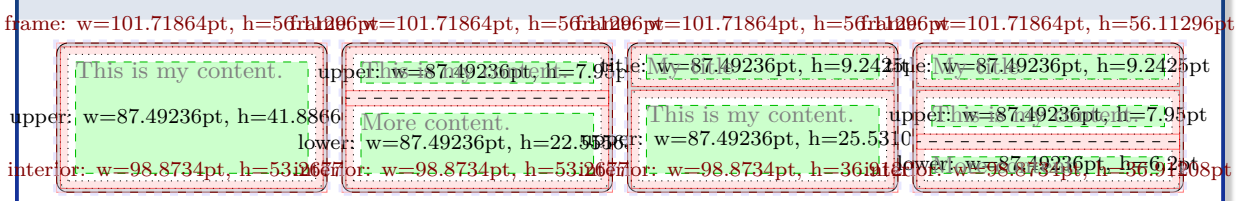
Environment and engines for the skin 'draft'

/tcb/graphical environment→P. 112:	`tikzpicture`
/tcb/frame engine→P. 113:	*special*
/tcb/interior titled engine→P. 113:	*special*
/tcb/interior engine→P. 114:	*special*
/tcb/segmentation engine→P. 114:	`path`
/tcb/title engine→P. 114:	`path`

/tcb/draft (style, no value)

This is an abbreviation for setting `skin=draft`.

```
\begin{tcbraster}[draft,raster equal height,raster columns=4,
    colback=LightGreen,colframe=DarkGreen,colbacktitle=LimeGreen!75!DarkGreen,
    left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm]
  \begin{tcolorbox}
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
  \end{tcolorbox}
  \begin{tcolorbox}[adjusted title=My title]
    This is my content.
    \tcblower
    More content.
  \end{tcolorbox}
\end{tcbraster}
```



```
\vspace*{3mm}
\begin{tcolorbox}[draft,title=A colored box with the 'draft' skin]
\lipsum[1-3]
\tcblower
\lipsum[4-6]
\end{tcolorbox}
```

frame: w=423.94617pt, h=506.20157pt

A colored box with the 'draft' skin title: w=392.64822pt, h=6.2pt

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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 metus odio 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.

upper: w=392.64822pt, h=249.95pt

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

lower: w=392.64822pt, h=205.95pt

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

interior: w=421.10092pt, h=490.04318pt

9.20 Skin Family 'freelance'

This skin family 'freelance' is deprecated with `tcolorbox` 3.00. It is not longer needed, because `/tcb/frame code`[→P. 116], `/tcb/interior code`[→P. 117], `/tcb/interior titled code`[→P. 116], and `/tcb/title code`[→P. 118] can be applied to every skin now. In this sense, everything has become *freelance* now.

For users of `/tcb/freelance`: Old code should continue to work. There may be exceptions for breakable freelance boxes under certain circumstances. For new code, use `/tcb/empty`[→P. 205] or `/tcb/enhanced`[→P. 178] where you would have used `/tcb/freelance` before.

`/tcb/skin=freelance` (skin)

This skin gives full freedom for the appearance of the `tcolorbox`. All drawing engines are set to type `freelance`; they use the `tikz` package and compute the `/tcb/geometry nodes`[→P. 115].

Environment and engines for the skin 'freelance'

`/tcb/graphical environment`[→P. 112]:	`tikzpicture`
`/tcb/frame engine`[→P. 113]:	`freelance`
`/tcb/interior titled engine`[→P. 113]:	`freelance`
`/tcb/interior engine`[→P. 114]:	`freelance`
`/tcb/segmentation engine`[→P. 114]:	`freelance`
`/tcb/title engine`[→P. 114]:	`freelance`

`/tcb/freelance` (style, no value)

This is an abbreviation for setting `skin=freelance`.

`/tcb/skin=freelancefirst` (skin)

This skin equals `freelance` with exception of the break sequence, see Section 15.7 on page 317.

`/tcb/skin=freelancemiddle` (skin)

This skin equals `freelance` with exception of the break sequence, see Section 15.7 on page 317.

`/tcb/skin=freelancelast` (skin)

This skin equals `freelance` with exception of the break sequence, see Section 15.7 on page 317.

`/tcb/extend freelance=`⟨*options*⟩ (no default, initially empty)

The ⟨*options*⟩ are added to the skin definition of `freelance`.

`/tcb/extend freelancefirst=`⟨*options*⟩ (no default, initially empty)

The ⟨*options*⟩ are added to the skin definition of `freelancefirst` which is used as first part of the break sequence of `freelance`. See `/tcb/skin first is subskin of`[→P. 119] for a substitute of this key.

`/tcb/extend freelancemiddle=`⟨*options*⟩ (no default, initially empty)

The ⟨*options*⟩ are added to the skin definition of `freelancemiddle` which is used as middle part of the break sequence of `freelance`. See `/tcb/skin middle is subskin of`[→P. 119] for a substitute of this key.

`/tcb/extend freelancelast=`⟨*options*⟩ (no default, initially empty)

The ⟨*options*⟩ are added to the skin definition of `freelancelast` which is used as last part of the break sequence of `freelance`. See `/tcb/skin last is subskin of`[→P. 119] for a substitute of this key.

10 Inclusion of Boxed Image Files

The LIB skins library adds some commands to conveniently include boxed image files. For the following macros and options, the LIB skins library has to be loaded by a package option or inside the preamble by:

```
\tcbuselibrary{skins}
```

See Section 9 on page 124 for the documentation of all other options of the LIB skins library.

10.1 Macros

N 2014-11-14
U 2014-11-24

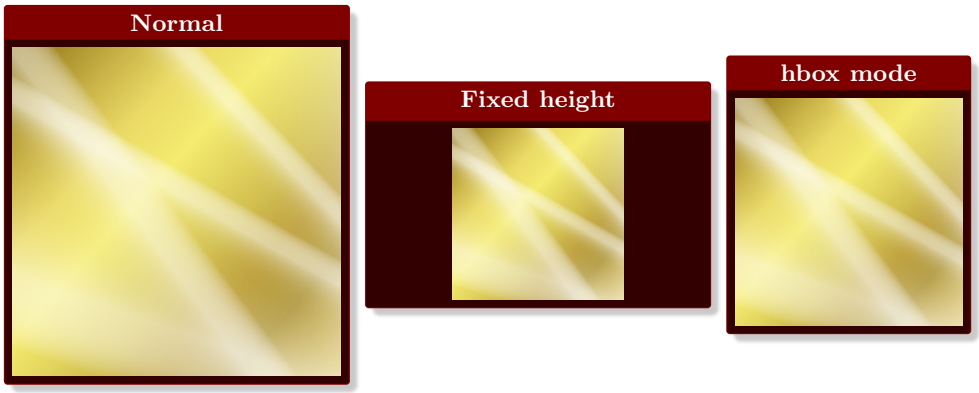
\tcbincludegraphics[⟨*options*⟩]{⟨*file name*⟩}

In principle, this macro includes an image file denoted by ⟨*file name*⟩ using the standard `\includegraphics` and puts it into a `tcolorbox`[→P. 11]. The ⟨*options*⟩ are `tcolorbox` keys to set up the colored box. Use `/tcb/graphics options`[→P. 221] to specify options for the underlying `\includegraphics`. Some `tcolorbox` option keys are automatically set, namely `/tcb/enhanced`[→P. 178] and options to center the image inside the box.

The sizing of the included image is done depending on the following:

- If a `/tcb/width`[→P. 33] is specified, but no fixed `/tcb/height`[→P. 51], the image is sized to fill the inner width of the box. The height of the box adapts to the image.
- If a fixed `/tcb/height`[→P. 51] is specified, the image is sized to fill the fixed inner area of the box.
- If the `/tcb/capture`[→P. 89] mode `/tcb/hbox`[→P. 89] is specified, the image is sized according to given `\includegraphics` options only. The box adapts to the image.

```
% \tcbuselibrary{raster}
\begin{tcbraster}[raster columns=3,raster force size=false,size=fbox,
    colframe=red!50!black,colback=red!20!black,
    fonttitle=\bfseries,center title,drop fuzzy shadow]
  \tcbincludegraphics[title=Normal]{goldshade.png}
  \tcbincludegraphics[title=Fixed height,height=3cm]{goldshade.png}
  \tcbincludegraphics[title=hbox mode,hbox,graphics options={width=3cm}]
    {goldshade.png}
\end{tcbraster}
```



The auxiliary macro **\imagename** may be used inside \tcbincludegraphics → P. 218 to display the name of the file. **\imagename** is already partially detokenized and is allowed to contain special characters like the underscore. Note that an appropriate font is required to display such characters.

```
% \tcbuselibrary{raster}
\begin{tcbraster}[size=fbox,
    colframe=red!50!black,colback=red!20!black,
    fonttitle=\bfseries\ttfamily,center title,drop fuzzy shadow]
  \tcbincludegraphics[title=\imagename]{goldshade.png}
  \tcbincludegraphics[finish={
    \node[fill=white,fill opacity=0.5,text opacity=1]
    at (frame.center) {\bfseries\ttfamily\imagename};}]{blueshade.png}
\end{tcbraster}
```

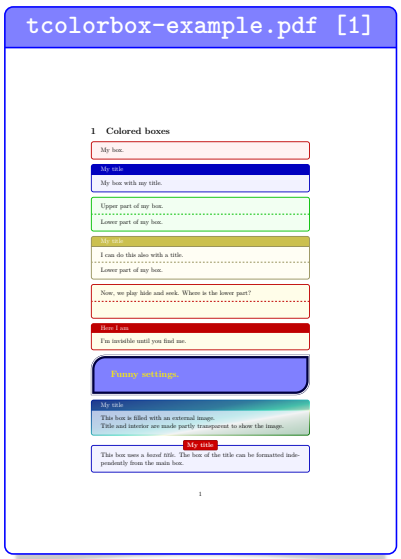
N 2014-11-14 **\tcbincludepdf**[⟨*options*⟩]{⟨*file name*⟩}

This is a generalized version of \tcbincludegraphics → P. 218 which allows to include a complete PDF file denoted by ⟨*file name*⟩. Every page is boxed into an own tcolorbox → P. 11 customized by the given ⟨*options*⟩. It is reasonable to put such a series of boxes inside a tcbraster → P. 233 for alignment.

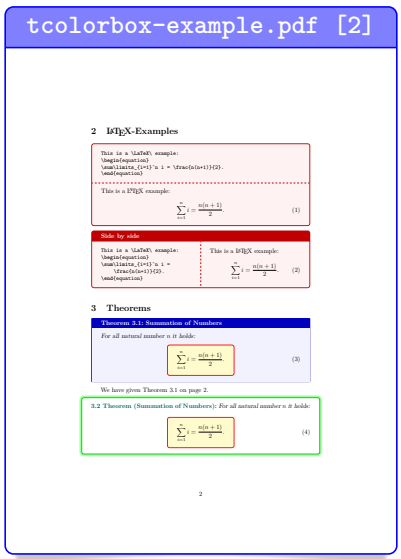
Use /tcb/graphics pages → P. 221 to use a selection of pages instead of using the whole file. The auxiliary macro **\imagepage** may be used inside \tcbincludepdf to display the current page number.

```
% \tcbuselibrary{raster}
\begin{tcbraster}[raster columns=3,colframe=blue,colback=white,
  colbacktitle=blue!50!white,fonttitle=\small\bfseries\ttfamily,
  left=0pt,right=0pt,top=0pt,bottom=0pt,boxsep=0pt,boxrule=0.6pt,
  toptitle=1mm,bottomtitle=1mm,drop lifted shadow,center title,
  graphics pages={1,...,6},title={\imagename\ [\imagepage]}]
\tcbincludepdf{tcolorbox-example.pdf}
\end{tcbraster}
```

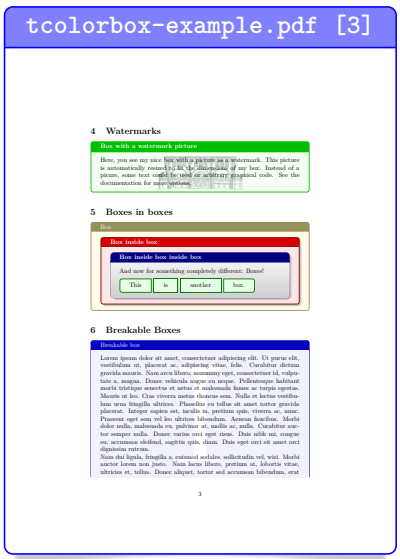
tcolorbox-example.pdf [1]



tcolorbox-example.pdf [2]



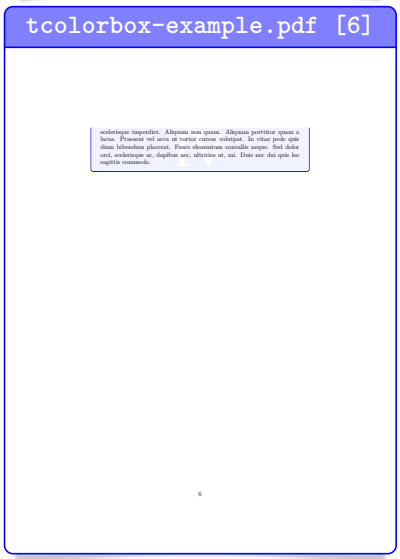
tcolorbox-example.pdf [3]



tcolorbox-example.pdf [4]

tcolorbox-example.pdf [5]

tcolorbox-example.pdf [6]

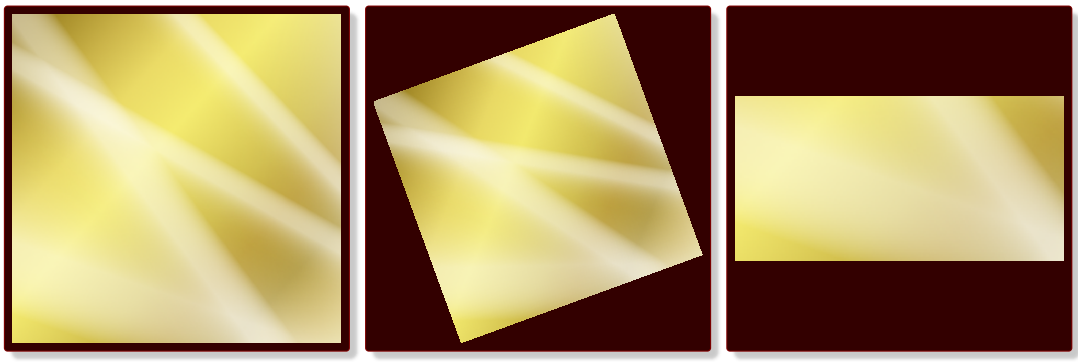


10.2 Option Keys

N 2014-11-14 **/tcb/graphics options=**⟨*options*⟩ (no default, initially empty)

Used for \tcbincludegraphics → P. 218 and \tcbincludepdf → P. 220 to specify \includegraphics ⟨*options*⟩.

```
% \tcbuselibrary{raster}
\begin{tcbraster}[raster columns=3,size=fbox,raster equal height,
    colframe=red!50!black,colback=red!20!black,drop fuzzy shadow]
  \tcbincludegraphics{goldshade.png}
  \tcbincludegraphics[graphics options={angle=20}]{goldshade.png}
  \tcbincludegraphics[graphics options={viewport=0cm 0cm 8cm 4cm,clip}]
    {goldshade.png}
\end{tcbraster}
```



N 2014-11-14 **/tcb/graphics directory=**⟨*directory*⟩ (no default, initially empty)

Used for \tcbincludegraphics → P. 218 and \tcbincludepdf → P. 220 to specify a file system ⟨*directory*⟩ where the image files are located.

```
\tcbset{
  graphics directory={.},
  graphics directory={examples},
  graphics directory={../../pictures},
}
```

The \graphicspath macro from the graphics package is superior to this option. /tcb/graphics directory may be used especially for \tcbincludepdf → P. 220.

N 2014-11-14 **/tcb/graphics pages=**⟨*selection*⟩ (no default, initially 1,...,\pdfpages)

Used for \tcbincludepdf → P. 220 to specify a ⟨*selection*⟩ of pages to be included. The largest page number is accessible by \pdfpages. The ⟨*selection*⟩ has to be given using the \foreach syntax of Ti*k*Z.

```
\tcbset{
  graphics pages={1,3,7},
  graphics pages={1,...,10},
  graphics pages={1,3,...,18},
  graphics pages={100,...,\pdfpages},
}
```

11 TikZ Image and Picture Fill Extensions; Auxiliary Macros

The LIB skins library adds some image and picture fill options to the vast option set of TikZ [20]. These options can be used in any `tikzpicture`. For the following options, the LIB skins library has to be loaded by a package option or inside the preamble by:

```
\tcbuselibrary{skins}
```

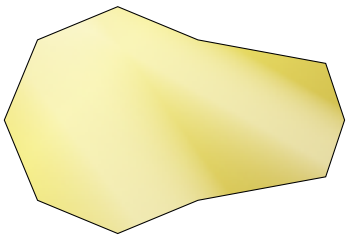
See Section 9 on page 124 for the documentation of all other options of the LIB skins library.

11.1 Fill Plain

/tikz/fill plain image=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The image is put in the center of the path, but it is not resized to fit into the path area.

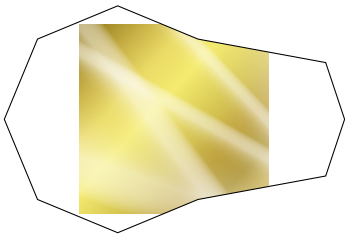
```
\begin{tikzpicture}
\path[draw,fill plain image=goldshade.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill plain image*=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The image is put in the center of the path, but it is not resized to fit into the path area. The ⟨*graphics options*⟩ are given to the underlying `\includegraphics` command.

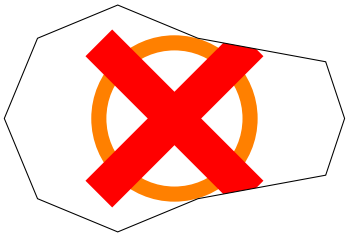
```
\begin{tikzpicture}
\path[draw,fill plain ↩
      ↪image*={width=2.5cm}{goldshade.png}]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill plain picture=⟨*graphical code*⟩ (no default, initially unset)

Fills the current path with the given ⟨*graphical code*⟩. The result is put in the center of the path, but it is not resized to fit into the path area. Note that this is almost identical to the standard `path picture` option.

```
\begin{tikzpicture}
\path[draw,fill plain picture={%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

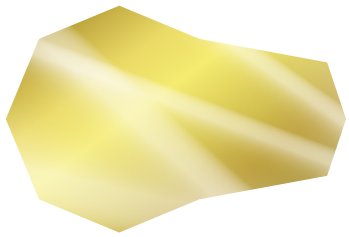


11.2 Fill Stretch

/tikz/fill stretch image=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The image is stretched to fill the path area.

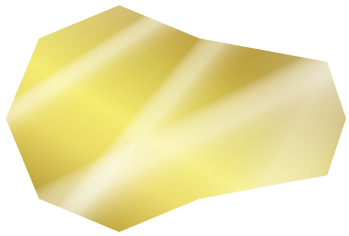
```
\begin{tikzpicture}
\path[fill stretch image=goldshade.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill stretch image*={⟨*graphics options*⟩}{⟨*file name*⟩} (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command. The image is stretched to fill the path area.

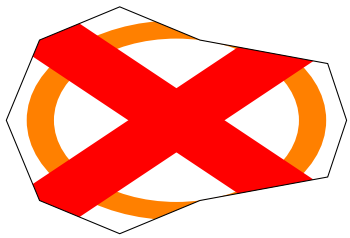
```
\begin{tikzpicture}
\path[fill stretch image*=
  {angle=90,origin=c}{goldshade.png}]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill stretch picture=⟨*graphical code*⟩ (no default, initially unset)

Fills the current path with the given ⟨*graphical code*⟩. The result is stretched to fill the path area.

```
\begin{tikzpicture}
\path[draw,fill stretch picture={%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

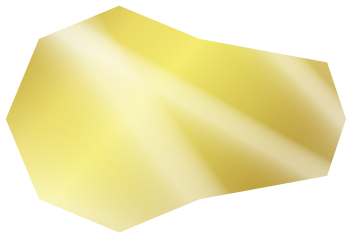


11.3 Fill Overzoom

/tikz/fill overzoom image=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The image is zoomed such that the path area fills the image.

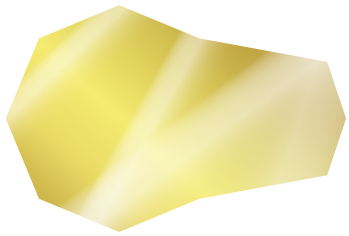
```
\begin{tikzpicture}
\path[fill overzoom image=goldshade.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill overzoom image*={⟨*graphics options*⟩}{⟨*file name*⟩} (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command. The image is zoomed such that the path area fills the image.

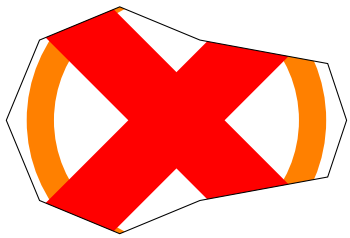
```
\begin{tikzpicture}
\path[fill overzoom image*=
  {angle=90,origin=c}{goldshade.png}]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill overzoom picture=⟨*graphical code*⟩ (no default, initially unset)

Fills the current path with the given ⟨*graphical code*⟩. The result is zoomed such that the path area fills the image.

```
\begin{tikzpicture}
\path[draw,fill overzoom picture={%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

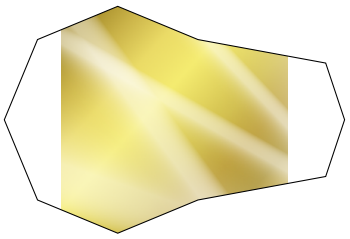


11.4 Fill Zoom

/tikz/fill zoom image=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The image is zoomed such that it fits inside the path area. Typically, some parts of the path area will stay unfilled.

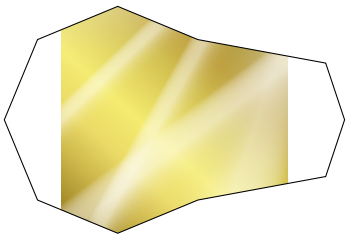
```
\begin{tikzpicture}
\path[draw,fill zoom image=goldshade.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill zoom image*={⟨*graphics options*⟩}{⟨*file name*⟩} (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command. The image is zoomed such that it fits inside the path area. Typically, some parts of the path area will stay unfilled.

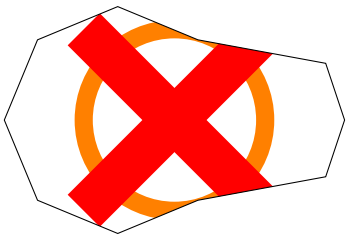
```
\begin{tikzpicture}
\path[draw,fill zoom image*=
  {angle=90,origin=c}{goldshade.png}]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill zoom picture=⟨*graphical code*⟩ (no default, initially unset)

Fills the current path with the given ⟨*graphical code*⟩. The result is zoomed such that it fits inside the path area. Typically, some parts of the path area will stay unfilled.

```
\begin{tikzpicture}
\path[draw,fill zoom picture={%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

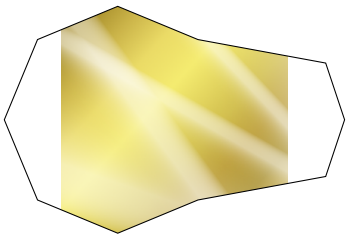


11.5 Fill Shrink

/tikz/fill shrink image=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The image is zoomed such that it fits inside the path area, but it never gets enlarged. Typically, some parts of the path area will stay unfilled.

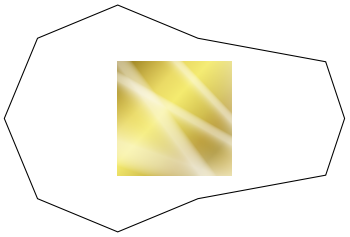
```
\begin{tikzpicture}
\path[draw,fill shrink image=goldshade.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill shrink image*=⟨*file name*⟩ (no default, initially unset)

Fills the current path with an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command. The image is zoomed such that it fits inside the path area, but it never gets enlarged. Typically, some parts of the path area will stay unfilled.

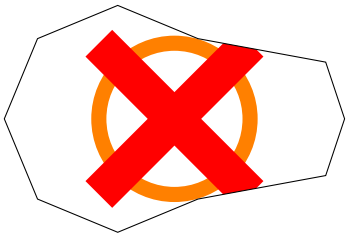
```
\begin{tikzpicture}
\path[draw,fill shrink ↩
      ↪image*={width=1.5cm}{goldshade.png}]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill shrink picture=⟨*graphical code*⟩ (no default, initially unset)

Fills the current path with the given ⟨*graphical code*⟩. The result is zoomed such that it fits inside the path area, but it never gets enlarged. Typically, some parts of the path area will stay unfilled.

```
\begin{tikzpicture}
\path[draw,fill shrink picture={%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



11.6 Fill Tile

/tikz/fill tile image=⟨*file name*⟩ (no default, initially unset)

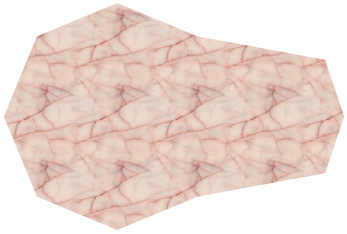
Fills the current path with a tile pattern using an external image referenced by ⟨*file name*⟩.

```
\begin{tikzpicture}
\path[fill tile image=pink_marble.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

/tikz/fill tile image*={⟨*graphics options*⟩}{⟨*file name*⟩} (no default, initially unset)

Fills the current path with a tile pattern using an external image referenced by ⟨*file name*⟩. The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command.

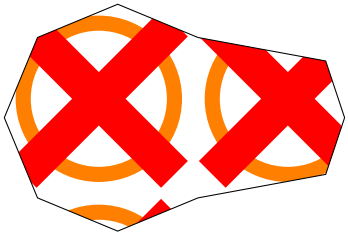
```
\begin{tikzpicture}
\path[fill tile image*={width=1cm}{pink_marble.png}]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill tile picture=⟨*graphical code*⟩ (no default, initially unset)

Fills the current path with a tile pattern using the given ⟨*graphical code*⟩.

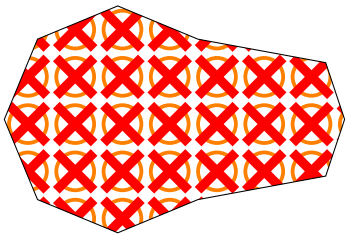
```
\begin{tikzpicture}
\path[draw,fill tile picture={%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```



/tikz/fill tile picture*={⟨*fraction*⟩}{⟨*graphical code*⟩} (no default, initially unset)

Fills the current path with a tile pattern using the given ⟨*graphical code*⟩. The graphic is resized by ⟨*fraction*⟩.

```
\begin{tikzpicture}
\path[draw,fill tile picture*={0.25}{%
  \draw[red!50!yellow,line width=2mm]
    (0,0) circle (1cm);
  \draw[red,line width=5mm] (-1,-1) -- (1,1);
  \draw[red,line width=5mm] (-1,1) -- (1,-1);
  }]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

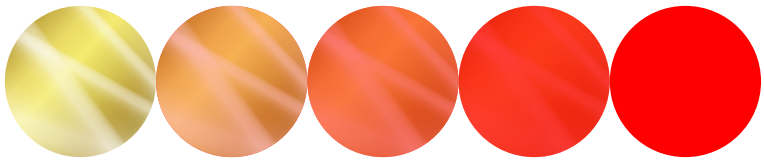


11.7 Filling Options

/tikz/fill image opacity=⟨*fraction*⟩ (no default, initially 1.0)

Sets the fill opacity for the image or picture fill options to the given ⟨*fraction*⟩.

```
\begin{tikzpicture}
\path[fill stretch image=goldshade.png] (0,0) circle (1cm);
\path[fill=red,fill stretch image=goldshade.png,fill image opacity=0.75]
  (2,0) circle (1cm);
\path[fill=red,fill stretch image=goldshade.png,fill image opacity=0.5]
  (4,0) circle (1cm);
\path[fill=red,fill stretch image=goldshade.png,fill image opacity=0.25]
  (6,0) circle (1cm);
\path[fill=red] (8,0) circle (1cm);
\end{tikzpicture}
```



/tikz/fill image scale=⟨*fraction*⟩ (no default, initially 1.0)

Stretches, zooms, overzooms or shrinks the image or picture to the given ⟨*fraction*⟩ of the width and height of the current path.

```
\begin{tikzpicture}
\path[draw,fill zoom image=goldshade.png]
  (0,0) rectangle +(2,2);

\path[draw,fill zoom image=goldshade.png,fill image scale=0.75]
  (3,0) rectangle +(2,2);

\path[draw,fill zoom image=goldshade.png,fill image scale=1.5]
  (6,0) rectangle +(2,2);
\end{tikzpicture}
```

/tikz/fill image options=⟨*graphics options*⟩ (no default, initially empty)

The ⟨*graphics options*⟩ are given to the underlying **\includegraphics** command for the image fill options. This can be just together with /tikz/fill stretch image → P. 223, /tikz/fill overzoom image → P. 224, /tikz/fill zoom image → P. 225, and /tikz/fill tile image → P. 227.

```
\begin{tikzpicture}
\path[fill image options={width=1cm},
  fill tile image=pink_marble.png]
  (2.75,-0.75) -- (3,0) -- (2.75,0.75)
  \foreach \w in {45,90,...,315}
    { -- (\w:1.5cm) } -- cycle;
\end{tikzpicture}
```

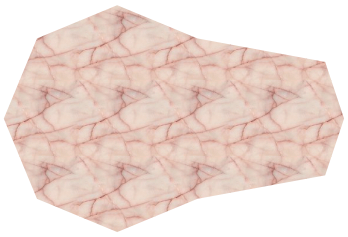
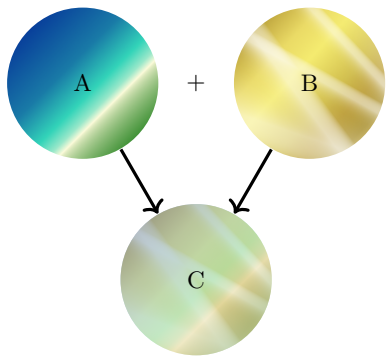


Image blending example

```
\begin{tikzpicture}[every node/.style=
  {circle,minimum width=2cm}]
\node[fill stretch image=blueshade.png]
  (A) at (120:3cm) {A};
\node[fill stretch image=goldshade.png]
  (B) at (60:3cm) {B};
\node[
  preaction={fill stretch image=blueshade.png},
  fill stretch image=goldshade.png,
  fill image opacity=0.5] (C) {C};
\path (A) -- node{$+$} (B);
\draw[->,very thick] (A)--(C);
\draw[->,very thick] (B)--(C);
\end{tikzpicture}
```



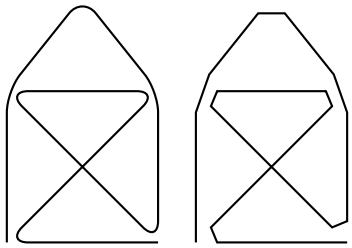
11.8 Straightening of the Arcs

This patch is considered as an experimental feature. It changes some of the original Ti*k*Z code. This change may break with future updates of Ti*k*Z.

N 2014-05-05 **\tcbpatcharcangular**

The Ti*k*Z package provides a nice `rounded corners` option to replace all corners by little arcs. `\tcbpatcharcangular` is a patch which straightens the arcs. To say it more prosaic, the little arcs are replaced by little straight lines.

```
\begin{tikzpicture}
  \draw[thick,rounded corners=8pt]
    (0,0) -- (0,2) -- (1,3.25) -- (2,2) -- (2,0)
     -- (0,2) -- (2,2) -- (0,0) -- (2,0);
  \tcbpatcharcangular
  \draw[thick,rounded corners=8pt,xshift=2.5cm]
    (0,0) -- (0,2) -- (1,3.25) -- (2,2) -- (2,0)
     -- (0,2) -- (2,2) -- (0,0) -- (2,0);
\end{tikzpicture}
```



N 2014-05-05 **\tcbpatcharcround**

This macro reverts `\tcbpatcharcangular`, i.e., the patch from `\tcbpatcharcangular` is replaced by the original code.

11.9 Extracting Node Dimensions

The following auxiliary macros are defined by the LIB skins library. They allow to determine the width and height of an arbitrary TikZ node. To be more specific, they determine the east-to-west and the north-to-south dimensions which may be not the maximal dimensions for a non-rectangular node. Note that the following dimensions are measured exactly including the line width of the border line. If a new rectangle or node with the same dimensions and a border is to be drawn, this border width has to be substracted.

N 2014-09-19 **\tcbsettowidthofnode**{⟨*register*⟩}{⟨*node*⟩}

Sets the east-to-west dimension of the given ⟨*node*⟩ to the TEX ⟨*register*⟩.

N 2014-09-19 **\tcbsetmacrotowidthofnode**{⟨*macro*⟩}{⟨*node*⟩}

Defines ⟨*macro*⟩ as the east-to-west dimension of the given ⟨*node*⟩.

N 2014-09-19 **\tcbsettoheightofnode**{⟨*register*⟩}{⟨*node*⟩}

Sets the north-to-south dimension of the given ⟨*node*⟩ to the TEX ⟨*register*⟩.

N 2014-09-19 **\tcbsetmacrotoheightofnode**{⟨*macro*⟩}{⟨*node*⟩}

Defines ⟨*macro*⟩ as the north-to-south dimension of the given ⟨*node*⟩.

```
\begin{tikzpicture}
  \node[align=center,draw=red,fill=yellow] (A) {This is my\\example node};
  \tcbsetmacrotowidthofnode\mywidth{A}
  \tcbsetmacrotoheightofnode\myheight{A}
  \path[fill=blue!25!white]           %  rectangle widthout border
    ([xshift=2mm]A.south east)
    rectangle node{Copy} +(\mywidth,\myheight);
  \node[draw=blue,fill=blue!25!white, % standard border width 0.4pt
    minimum width=\mywidth-0.4pt,     % minus width of border
    minimum height=\myheight-0.4pt    % minus height of border
    ]
    at ([xshift=5cm]A) {Copy 2};
\end{tikzpicture}
```

12 Library raster

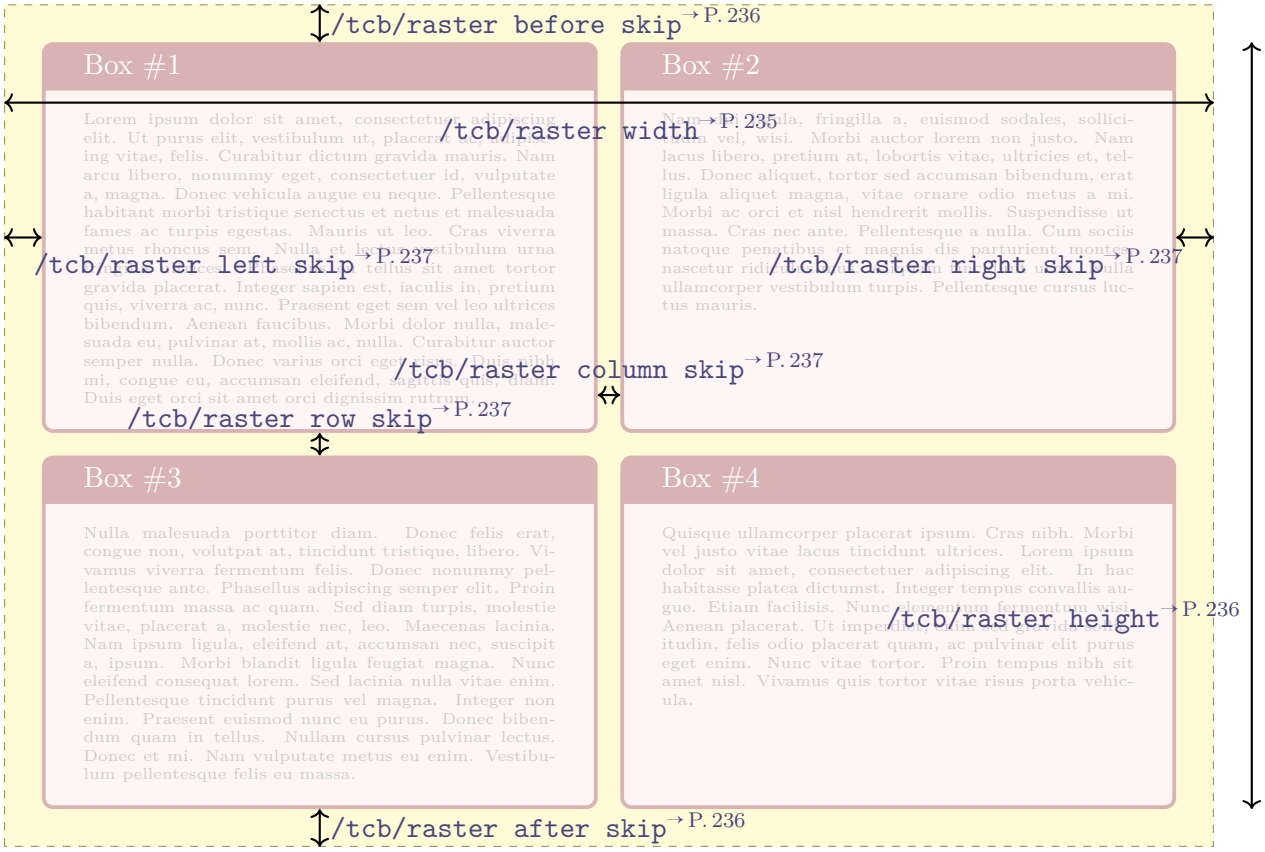
The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{raster}
```

12.1 Concept of Rasters

A *raster* is used to align several colored boxes in a regular way. It can be seen as a far related counterpart to the `matrix` construct of Ti*k*Z, but it differs in many aspects.

In principle, `tcolorbox`es are arranged in rows and columns when put inside a `tcbraster` → P. 233 environment. The boxes are fluently added to the raster like adding text to a paragraph. Especially, line/row breaks are done automatically and one cannot end a line/row ahead of schedule. Further, a *raster* is not restricted to a single page but may break into an arbitrary series of pages.



```
\begin{tcbraster}[raster columns=3,raster rows=3,raster height=\linewidth,
  enhanced,size=small,sharp corners,arc=8mm,colframe=red!50!black,
  colback=yellow!10!white,watermark overzoom=1.0,fit algorithm=hybrid* ]
  \begin{tcolorbox}[rounded corners=northwest,boxrule=0pt,
    watermark graphics=lichtspiel.jpg]\end{tcolorbox}
  \tcboxfit{\lipsum[1]}
  \begin{tcolorbox}[rounded corners=northeast,boxrule=0pt,
    watermark graphics=goldshade.png]\end{tcolorbox}
  \tcboxfit{\lipsum[2]}
  \begin{tcolorbox}[valign=center,halign=center]Nine Boxes.\end{tcolorbox}
  \tcboxfit{\lipsum[3]}
  \begin{tcolorbox}[rounded corners=southwest,boxrule=0pt,
    watermark graphics=goldshade.png]\end{tcolorbox}
  \tcboxfit{\lipsum[4]}
  \begin{tcolorbox}[rounded corners=southeast,boxrule=0pt,
    watermark graphics=lichtspiel.jpg]\end{tcolorbox}
\end{tcbraster}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Nine Boxes.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

12.2 Macros of the Library

N 2014-11-10

```
\begin{tcbraster}[⟨options⟩]
 ⟨environment content⟩
\end{tcbraster}
```

A raster arranges enclosed boxes in a regular way, mainly into rows and columns. The ⟨*options*⟩ are used to control the raster parameters and to set the properties for the enclosed boxes.

- The *raster* is only allowed to contain a series of `tcolorbox` → P. 11 environments or derived constructs. With some small restrictions, boxes created with `\tcboxfit` → P. 334 can also be added. Boxes created with `\tcbox` → P. 13 are not reasonable here, but may be used to a certain degree.
- Do not add anything else between the boxes inside the raster with exception of white-space. Especially, do not use `\\` or `\par` to end a row; row breaks are done automatically.
- The boxes inside a raster are numbered automatically. `\thetcbrasternum` may be used inside a box to access this number.

```
\begin{tcbraster}[raster columns=3, raster equal height,
  size=small,colframe=red!50!black,colback=red!10!white,colbacktitle=red!50!white,
  title={Box \# \thetcbrasternum}]
  \begin{tcolorbox}First box\end{tcolorbox}
  \begin{tcolorbox}Second box\end{tcolorbox}
  \begin{tcolorbox}This is a box\\with a second line\end{tcolorbox}
  \begin{tcolorbox}Another box\end{tcolorbox}
  \begin{tcolorbox}A box again\end{tcolorbox}
\end{tcbraster}
```

Box # 1
First box

Box # 2
Second box

Box # 3
This is a box
with a second line

Box # 4
Another box

Box # 5
A box again

```
\begin{tcbraster}[raster columns=2, raster equal height=rows,
  size=small,colframe=red!50!black,colback=red!10!white,colbacktitle=red!50!white,
  title={Box \# \thetcbrasternum}]
  \begin{tcolorbox}First box\end{tcolorbox}
  \begin{tcolorbox}Second box\end{tcolorbox}
  \begin{tcolorbox}This is a box\\with a second line\end{tcolorbox}
  \begin{tcolorbox}Another box\end{tcolorbox}
  \begin{tcolorbox}A box again\end{tcolorbox}
\end{tcbraster}
```

Box # 1
First box

Box # 2
Second box

Box # 3
This is a box
with a second line

Box # 4
Another box

Box # 5
A box again

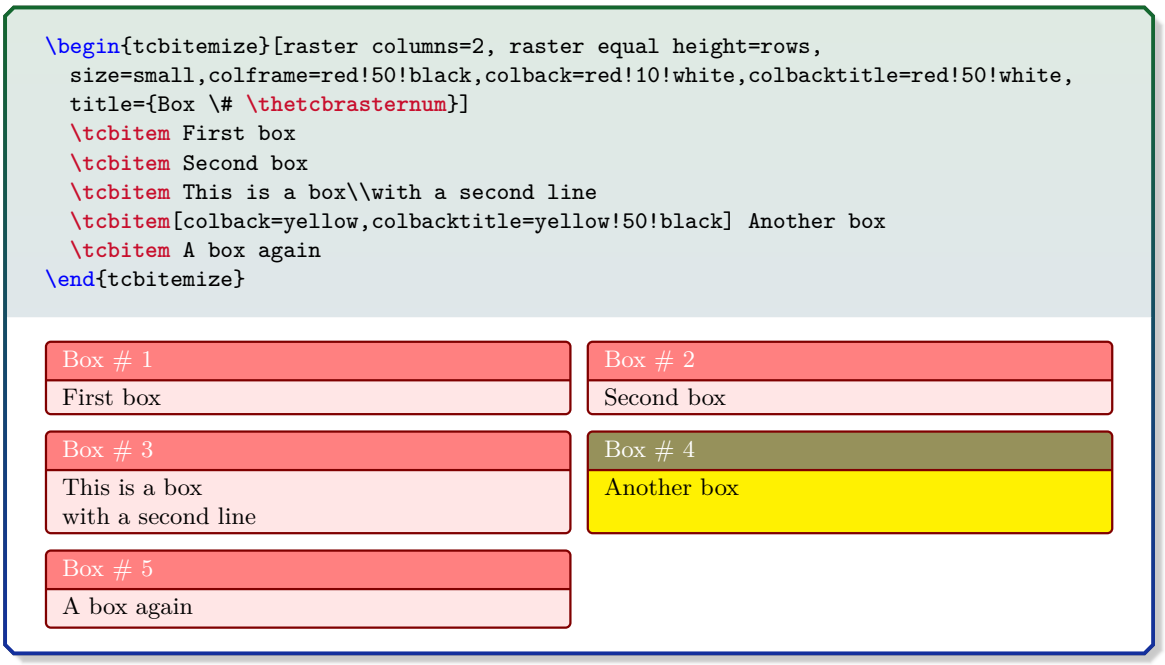
N 2014-11-10 `\begin{tcbitemize}[⟨options⟩]`
⟨*environment content*⟩
`\end{tcbitemize}`

This is a special case of a `tcbraster` → P. 233 with the given ⟨*options*⟩.

- Here, the enclosed boxes are created using `\tcbitem`.
- There has to be at least one `\tcbitem`.
- One cannot use anything else than `\tcbitem` to add something to the *raster*.

This leads to a very compact syntax.

```
\begin{tcbitemize}[raster columns=2, raster equal height=rows,
  size=small,colframe=red!50!black,colback=red!10!white,colbacktitle=red!50!white,
  title={Box \# \thetcbrasternum}]
  \tcbitem First box
  \tcbitem Second box
  \tcbitem This is a box\\with a second line
  \tcbitem[colback=yellow,colbacktitle=yellow!50!black] Another box
  \tcbitem A box again
\end{tcbitemize}
```



`tcbitemize` has more restrictions than `tcbraster` → P. 233. Especially, the `/tcb/capture` → P. 89 mode has to be `minipage`. For example, `/tcb/fit` → P. 336 cannot be used safely. If `/tcb/fit` → P. 336 should be used, turn over to `tcbraster` → P. 233.

N 2014-11-10 `\tcbitem[⟨options⟩]`

Used inside `tcbitemize` to create a new `tcolorbox` → P. 11 with the given ⟨*options*⟩.

12.3 Option Keys of the Library

N 2014-11-10 **/tcb/raster columns=**⟨*number*⟩ (no default, initially 2)

Sets the ⟨*number*⟩ of columns for a *raster*.

```
\begin{tcbitemize}[raster columns=3,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
\begin{tcbitemize}[raster columns=4,
  size=small,colframe=blue!50!black,colback=blue!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One Two Three

Four

One Two Three Four

N 2014-11-10 **/tcb/raster rows=**⟨*number*⟩ (no default, initially 2)

Sets the ⟨*number*⟩ of rows for a *raster*. Note that this is only relevant in connection with setting /tcb/raster height → P. 236 to a value greater than 0pt. Then, it defines the number of rows *per* given height.

N 2014-11-10 **/tcb/raster width=**⟨*length*⟩ (no default, initially \linewidth)

Sets the total raster width to the given ⟨*length*⟩. /tcb/raster left skip → P. 237 and /tcb/raster right skip → P. 237 are part of the total width.

```
\begin{tcbitemize}[raster width=\linewidth/2,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

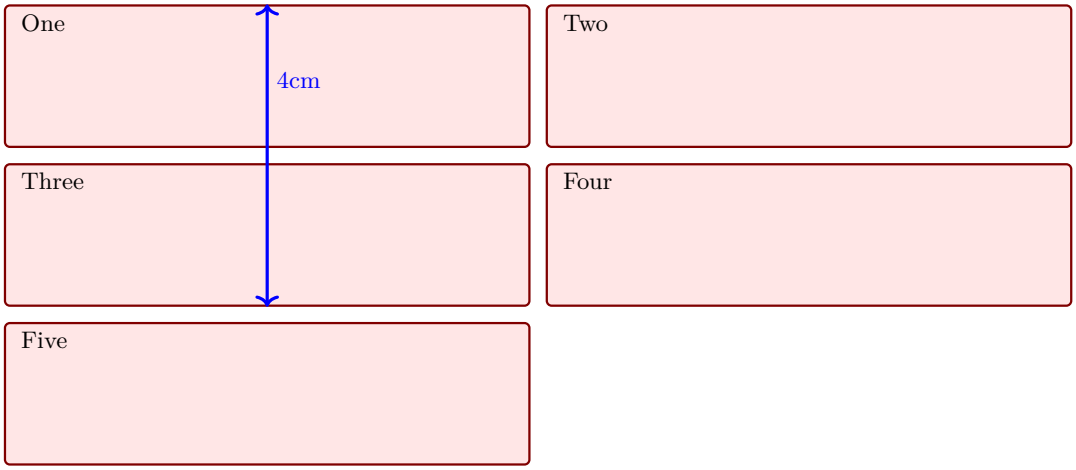
One Two

Three Four

N 2014-11-10 **/tcb/raster height=⟨*length*⟩** (no default, initially `0pt`)

Sets the raster height *per* `/tcb/raster rows` → P. 235 to the given ⟨*length*⟩. This forces an appropriate height for the enclosed boxes. `/tcb/raster before skip` and `/tcb/raster after skip` are not part of this calculation. If the ⟨*length*⟩ is set to `0pt`, this feature is deactivated.

```
\begin{tcbitemize}[raster height=4cm, raster rows=2,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem[enhanced,
    finish={\draw[blue,very thick,<->] (frame.south)
      -- node[right,pos=.75]{4cm} +(0,4); }]
    Three
  \tcbitem Four
  \tcbitem Five
\end{tcbitemize}
```



N 2014-11-10
U 2014-12-16 **/tcb/raster before skip=⟨*glue*⟩** (no default, initially `2mm`)

Space of the given ⟨*glue*⟩ is inserted vertically before the *raster*. This space is discardable.

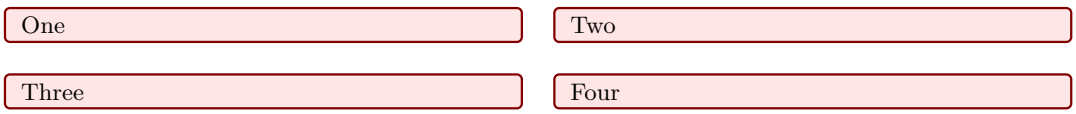
N 2014-11-10
U 2014-12-16 **/tcb/raster after skip=⟨*glue*⟩** (no default, initially `2mm`)

Space of the given ⟨*glue*⟩ is inserted vertically after the *raster*. This space is discardable.

N 2015-01-08 **/tcb/raster equal skip=⟨*length*⟩** (style, no default)

Shortcut to set `/tcb/raster before skip`, `/tcb/raster after skip`, `/tcb/raster column skip` → P. 237, and `/tcb/raster row skip` → P. 237 to the same ⟨*length*⟩ value.

```
\begin{tcbitemize}[raster equal skip=4mm,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```



N 2014-11-10 **/tcb/raster left skip=⟨*length*⟩** (no default, initially 0pt)

Space of the given ⟨*length*⟩ is inserted horizontally left of the *raster*.

```
\begin{tcbitemize}[raster left skip=2cm,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One | Two
Three | Four

N 2014-11-10 **/tcb/raster right skip=⟨*length*⟩** (no default, initially 0pt)

Space of the given ⟨*length*⟩ is inserted horizontally right of the *raster*.

```
\begin{tcbitemize}[raster right skip=2cm,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One | Two
Three | Four

N 2014-11-10 **/tcb/raster column skip=⟨*length*⟩** (no default, initially 2mm)

Space of the given ⟨*length*⟩ is inserted horizontally between the columns.

```
\begin{tcbitemize}[raster column skip=2cm,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One | Two
Three | Four

N 2014-11-10 **/tcb/raster row skip=⟨*length*⟩** (no default, initially 2mm)

Space of the given ⟨*length*⟩ is inserted vertically between the rows.

```
\begin{tcbitemize}[raster row skip=0pt,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One | Two
Three | Four

N 2014-11-10 **/tcb/raster halign=⟨*alignment*⟩** (no default, initially **left**)

Defines the horizontal alignment for the boxes of the rows of a *raster*, if these rows are not completely filled (mainly: the last one).
Feasible values for ⟨*alignment*⟩ are:

- **left**: align to the left side,
- **center**: align to the center,
- **right**: align to the right side.

```
\begin{tcbitemize}[raster halign=center,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
\end{tcbitemize}
```

One Two Three

N 2014-11-10 **/tcb/raster valign=⟨*alignment*⟩** (no default, initially **center**)

Defines the vertical alignment for the boxes of a row, if the boxes do not have equal height. This sets the /tcb/box align →P. 77 option.
Feasible values for ⟨*alignment*⟩ are:

- **top**: align to the top side,
- **center**: align to the center,
- **bottom**: align to the bottom side.

```
\begin{tcbitemize}[raster valign=top, raster columns=3,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem \Huge One
  \tcbitem \Large Two
  \tcbitem Three
\end{tcbitemize}
\begin{tcbitemize}[raster valign=center, raster columns=3,
  size=small,colframe=blue!50!black,colback=blue!10!white]
  \tcbitem \Huge One
  \tcbitem \Large Two
  \tcbitem Three
\end{tcbitemize}
\begin{tcbitemize}[raster valign=bottom, raster columns=3,
  size=small,colframe=green!50!black,colback=green!10!white]
  \tcbitem \Huge One
  \tcbitem \Large Two
  \tcbitem Three
\end{tcbitemize}
```

One Two Three
One Two Three
One Two Three

N 2014-11-10 **/tcb/raster equal height=⟨*type*⟩** (default **all**, initially **none**)

Puts the enclosed boxes into a common /tcb/equal height group → P. 56. The ⟨*id*⟩ of the equal height group is chosen automatically, but it may be set manually by /tcb/raster equal height group.

Feasible values for ⟨*type*⟩ are:

- **none**: no equal height setting,
- **rows**: all boxes in a row are set to equal height,
- **all**: all boxes in the raster are set to equal height.

Note that you have to compile twice to see changes.

```
\begin{tcbitemize}[raster equal height=rows,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem \Huge Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One

Two

Three

Four

```
\begin{tcbitemize}[raster equal height,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem \Huge Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One

Two

Three

Four

N 2014-11-10 **/tcb/raster equal height group=⟨*id*⟩** (no default)

Overwrites the automatically chosen id with the given ⟨*id*⟩. If this is used to share a common height between the *raster* and another raster or box, the /tcb/raster equal height option should be set to **all**.

```
\tcbset{size=small,colframe=red!50!black,colback=red!10!white}
\begin{tcolorbox}[equal height group=raster-manual-id]
  A single box
\end{tcolorbox}
\begin{tcbitemize}[raster equal height,raster equal height group=raster-manual-id]
  \tcbitem One
  \tcbitem \Huge Two
\end{tcbitemize}
```

A single box

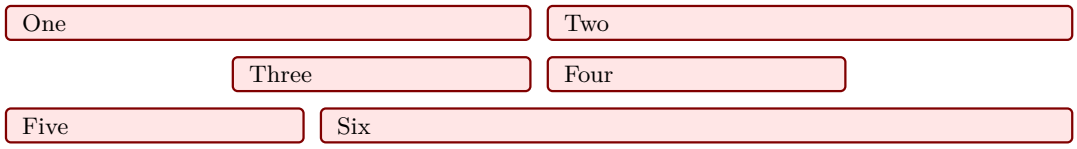
One

Two

N 2014-11-10 **/tcb/raster force size=true|false** (default true, initially true)

Enforces the raster size computations onto the enclosed boxes. If set to **false**, individual settings can be used (for the better or worse).

```
\begin{tcbitemize}[raster force size=false, raster halign=center,
  size=small,colframe=red!50!black,colback=red!10!white]
  \tcbitem One
  \tcbitem Two
  \tcbitem[add to width=-3cm] Three
  \tcbitem[add to width=-3cm] Four
  \tcbitem[add to width=-3cm] Five
  \tcbitem[add to width=3cm] Six
\end{tcbitemize}
```



N 2014-11-10 **/tcb/raster reset** (no value)

Sets all raster settings back to their default values. Note that /tcb/reset → P. 95 does not execute this option. Style settings like /tcb/raster odd column etc. are not touched by /tcb/raster reset.

12.4 Adding Styles for Specific Boxes

The following styles can be defined to address certain boxes inside a *raster*. Note that such style definitions are not removed by /tcb/reset → P. 95 or /tcb/raster reset. The style definitions are used in the order given below.

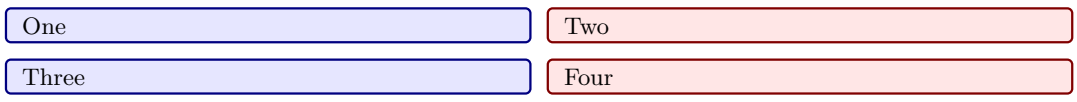
N 2014-11-24 **/tcb/raster every box** (style)

This style is used for every box.

N 2014-11-10 **/tcb/raster odd column** (style)

This style is used for every box in an odd column.

```
\begin{tcbitemize}[size=small,colframe=red!50!black,colback=red!10!white,
  raster odd column/.style={colframe=blue!50!black,colback=blue!10!white}]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```



N 2014-11-10 **/tcb/raster even column** (style)

This style is used for every box in an even column.

N 2014-11-10 **/tcb/raster column n** (style)

This style is used for every box in the n-th column. n has to be replaced by a number.

N 2014-11-10 **/tcb/raster odd row** (style)

This style is used for every box in an odd row.

N 2014-11-10 **/tcb/raster even row** (style)

This style is used for every box in an even row.

N 2014-11-10 **/tcb/raster row m** (style)

This style is used for every box in the m-th row. m has to be replaced by a number.

```
\begin{tcbitemize}[size=small,colframe=red!50!black,colback=red!10!white,
  raster row 2/.style={colframe=blue!50!black,colback=blue!10!white}]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One | Two
Three | Four

N 2014-11-10 **/tcb/raster odd number** (style)

This style is used for every box with an odd number.

N 2014-11-10 **/tcb/raster even number** (style)

This style is used for every box with an even number.

```
\begin{tcbitemize}[size=small,colframe=red!50!black,colback=red!10!white,
  raster columns=3,
  raster even number/.style={colframe=blue!50!black,colback=blue!10!white}]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
  \tcbitem Five
  \tcbitem Six
\end{tcbitemize}
```

One | Two | Three
Four | Five | Six

N 2014-11-10 **/tcb/raster row m column n** (style)

This style is used for the box in the m-th row and n-th column. m and n have to be replaced by numbers.

N 2014-11-10 **/tcb/raster number n** (style)

This style is used for the box with number n. n has to be replaced by a number.

```
\begin{tcbitemize}[size=small,colframe=red!50!black,colback=red!10!white,
  raster number 4/.style={colframe=blue!50!black,colback=blue!10!white}]
  \tcbitem One
  \tcbitem Two
  \tcbitem Three
  \tcbitem Four
\end{tcbitemize}
```

One | Two
Three | Four

13 Libraries LIB listings, LIB listingsutf8, and LIB minted

13.1 Loading the Libraries

In contrast to other `tcolorbox` libraries, the libraries LIB listings, LIB listingsutf8, and LIB minted are concurrent in the sense that they all do the same thing, i. e. displaying listings with or without typesetting the listing in LaTeX parallel. The difference is the underlying LaTeX package which does the core job for displaying a listing. So, typically, you need just *one* of these libraries. If you do not have a clue, which one of them you should use, you should take LIB listingsutf8.

The order in which the libraries are included influences the default settings and the `/tcb/reset` → P. 95 behavior. The settings of a later loaded library overwrite the settings of a previous loaded library. A library is never loaded twice.

13.1.1 Loading LIB listings

This library uses the package `listings` [6] to typeset listings. It is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{listings}
```

This also loads the package `listings` [6].

The `/tcb/listing engine` → P. 254 is set to `listings` by the library. To reactivate this setting, if overwritten by other libraries, use

```
\tcbset{listing engine=listings}
```

13.1.2 Loading LIB listingsutf8

To extend `listings` for UTF-8 encoded sources, you can use the support from the package `listingsutf8` [10] by loading the library variant LIB listingsutf8.

```
\tcbuselibrary{listingsutf8}
\tcbset{listing utf8=latin1}% optional; 'latin1' is the default.
```

This also loads the library LIB listings and the packages `listings` [6] and `listingsutf8` [10].

The `/tcb/listing engine` → P. 254 is set to `listings` by the library. To reactivate this setting, if overwritten by other libraries, use

```
\tcbset{listing engine=listings}
```

13.1.3 Loading LIB minted

This library uses the package `minted` [14] to typeset listings. It is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{minted}
```

This also loads the package `minted` [14].

The `minted` package uses the external tool `Pygments` [12] to apply syntax highlighting. It has to be installed and set up, before the library can be used, see [14] and [12]. The `tcolorbox` library LIB minted does not work, if the package `minted` [14] does not work.

The `/tcb/listing engine` → P. 254 is set to `minted` by the library. To reactivate this setting, if overwritten by other libraries, use

```
\tcbset{listing engine=minted}
```

13.2 Common Macros of the Libraries

`\begin{tcblisting}{`⟨*options*⟩`}`
⟨*environment content*⟩
`\end{tcblisting}`

Creates a colored box based on a `tcolorbox` → P. 11. Controlled by the given ⟨*options*⟩, the environment content is typeset normally and/or as a listing. Furthermore, the ⟨*options*⟩ control appearance and functions of the `tcolorbox`. By default, the listing is interpreted as a LaTeX listing.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{tcblisting}
```

```
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
```

This is a LaTeX example which displays the text as source code and in compiled form.

```
% \tcbuselibrary{listings} /or/ \tcbuselibrary{listingsutf8}
\begin{tcblisting}{colback=yellow!5,colframe=yellow!50!black,listing only,
    title=This is source code in another language (XML), fonttitle=\bfseries,
  listing options={language=XML,columns=fullflexible,keywordstyle=\color{red}}}
<?xml version="1.0"?>
<project name="Package tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
\end{tcblisting}
```

This is source code in another language (XML)

```
<?xml version="1.0"?>
<project name="Package␣tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
```

```
% \tcbuselibrary{minted}
\begin{tcblisting}{colback=yellow!5,colframe=yellow!50!black,listing only,
    title=This is source code in another language (XML), fonttitle=\bfseries,
  listing engine=minted,minted language=xml}
<?xml version="1.0"?>
<project name="Package tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
\end{tcblisting}
```

This is source code in another language (XML)

```
<?xml version="1.0"?>
<project name="Package tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
```

```
% This box is as wide as needed (listing only !!)
% \tcbuselibrary{skins}
\begin{tcblisting}{colback=green!5!white,colframe=green!50!black,listing only,
  hbox,enhanced,drop fuzzy shadow,before=\begin{center},after=\end{center}}
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
\end{tcblisting}
```

```
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
```

```
\begin{tcboutputlisting}
 ⟨environment content⟩
\end{tcboutputlisting}
```

Saves the environment content to a file which is named by the key value of `listing file`. Later, this file can be loaded by `\tcbinputlisting` or `\tcbuselistingtext` or `\tcbuselistinglisting`.

```
\begin{tcboutputlisting}
This \textbf{text} is written to a standardized file for later usage.
\end{tcboutputlisting}
```

`\tcbinputlisting{⟨options⟩}`

Creates a colored boxed based on a `tcolorbox`. The text content is read from a file named by the key value of `listing file`. Apart from that, the function is equal to that of `tcblisting` → P. 243.

```
\tcbinputlisting{colback=red!5!white,colframe=red!75!black,text only}
\tcbinputlisting{colback=green!5,colframe=green!75!black,listing only}
```

```
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
```

`\tcbuselistingtext`

Loads text from a file named by the key value of `listing file`.

```
\tcbuselistingtext
```

`\tcbuselistinglisting`

Typesets text as listing from a file named by the key value of `listing file`.

```
\tcbuselistinglisting
```

```
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
```

`\tcbusetemplisting`

Typesets text as listing from a temporary file which was written by `tcbwritetemp` → P. 105.

See Section 19.4 on page 364 and Section 19.5 on page 366 for more elaborate methods to create new environments and commands.

If a new sort of `tcblisting` environments should be created with one optional argument only, one is highly recommended to use `\DeclareTCBListing` → P. 364 or `\NewTCBListing` → P. 364 instead of `\newtcblisting` to avoid content scanning problems.

`\newtcblisting[⟨init options⟩]{⟨name⟩}[⟨number⟩][⟨default⟩]{⟨options⟩}`

Creates a new environment ⟨*name*⟩ based on `tcblisting` → P. 243. Basically, `\newtcblisting` operates like `\newenvironment`. This means, the new environment ⟨*name*⟩ optionally takes ⟨*number*⟩ arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying `tcblisting`. Note that `/tcb/savedelimiter` → P. 25 is set to the given ⟨*name*⟩ automatically. The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

```
\newtcblisting{mybox}{%
  colback=red!5!white,
  colframe=red!75!black}

\begin{mybox}
This is my \LaTeX\ box.
\end{mybox}
```

This is my \LaTeX\ box.

This is my LaTeX box.

```
\newtcblisting{mybox}[1]{%
  colback=red!5!white,
  colframe=red!75!black,
  fonttitle=\bfseries,
  title=#1}

\begin{mybox}{Listing Box}
This is my \LaTeX\ box.
\end{mybox}
```

Listing Box

This is my \LaTeX\ box.

This is my LaTeX box.

```
\newtcblisting{mybox}[2][]{%
  colback=red!5!white,
  colframe=red!75!black,
  fonttitle=\bfseries,
  title=#2,#1}

\begin{mybox}[listing only]
  {Listing Box}
This is my \LaTeX\ box.
\end{mybox}
\bigskip

\begin{mybox}[listing side text]
  {Listing Box}
This is my
\LaTeX\ box.
\end{mybox}
```

Listing Box

This is my \LaTeX\ box.

Listing Box

This is my \LaTeX\ box.

This is my LaTeX box.

Definition in the preamble:

```
\newtcblisting[auto counter]{mycbox}[1]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Listing \thetcbcounter: #1}
```

```
\begin{mycbox}{Listing Box}
This is my \LaTeX\ box.
\end{mycbox}
```

Listing 1: Listing Box

This is my \LaTeX\ box.

This is my LaTeX box.

\renewtcblisting[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like \newtcblisting → P. 246, but based on \renewenvironment instead of \newenvironment. An existing environment is redefined.

`\newtcbinputlisting[⟨init options⟩]{\⟨name⟩}[⟨number⟩][⟨default⟩]{⟨options⟩}`

Creates a new macro \⟨*name*⟩ based on `\tcbinputlisting` → P. 245. Basically, `\newtcbinputlisting` operates like `\newcommand`. The new macro \⟨*name*⟩ optionally takes ⟨*number*⟩ arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying `tcbinputlisting`. The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

Listing (2) of `tcolorbox.tcbtemp`

This is the included file content:

```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing engine=minted,minted language=latex,minted style=colorful,
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

Listing (3) of `tcolorbox.tcbtemp`

This is the included file content:

```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing engine=minted,minted language=latex,minted style=colorful,
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

`\renewtcbinputlisting[⟨init options⟩]{\⟨name⟩}[⟨number⟩][⟨default⟩]{⟨options⟩}`

Operates like `\newtcbinputlisting`, but based on `\renewcommand` instead of `\newcommand`. An existing macro is redefined.

13.3 Option Keys of the listings Library

/tcb/listing options=⟨*key list*⟩ (no default, initially `style=tcblatex`)

Sets the options from the package `listings` [6] which are used during typesetting of the listing. For LaTeX listings, there is a predefined `listings` style named `tcblatex` which can be used.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!25,left=6mm,
listing options={style=tcblatex,numbers=left,numberstyle=\tiny\color{red!75!black}}}
This is a \LaTeX\ example which displays the text as source code
and in compiled form. Additionally, we use line numbers here.
\end{tcblisting}
```

```
1 This is a \LaTeX\ example which displays the text as source code
2 and in compiled form. Additionally, we use line numbers here.
```

This is a LaTeX example which displays the text as source code and in compiled form. Additionally, we use line numbers here.

/tcb/no listing options (no value, initially unset)

Abbreviation for `listing options={}`. This removes all options for the `listings` package. This includes the `tcblisting` standard style `tcblatex` and the encoding presets. Use this option, if you want to set the `listings` options outside of `tcblisting`, e. g. globally in the preamble.

```
\begin{tcblisting}{no listing options}
All \textit{listings} options removed.
\end{tcblisting}
```

```
All \textit{listings} options removed.
```

All *listings* options removed.

/tcb/listing style=⟨*style*⟩ (no default, initially `tcblatex`)

Abbreviation for `listing options={style=...}`. This key sets a ⟨*style*⟩ for the `listings` package, see [6]. For LaTeX, there is a predefined style named `tcblatex`.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,
listing style=tcblatex}
Here, we use the predefined style.
\end{tcblisting}
```

```
Here, we use the predefined style.
```

Here, we use the predefined style.

/tcb/listing inputencoding=⟨*encoding*⟩ (no default, initially `\inputencodingname`)

Sets the input encoding value for the predefined listing style `tcblatex` and `tcbdocumentation` from the library LIB documentation. The initial value is derived from the package `inputenc` if used.

/tcb/listing remove caption=true|false (default `true`, initially `true`)

If set to `true`, some part of the caption building code of the `listings` package is silenced to prevent some unwanted interaction with the `hyperref` package resulting in additional vertical space. If set to `false`, the `listings` package code is kept unchanged. Note that listings outside `tcblisting` → P. 243 and `\tcbinputlisting` → P. 245 are always processed normally. Typically, a user is not expected to use this key at all.

/tcb/every listing line=⟨*text*⟩ (no default, initially unset/empty)

Inserts some ⟨*text*⟩ to the begin of every line of a listing. Note that this a hack of the `listings` package code. This may become unusable or superfluous in the future.

```
\newtcblisting{commandshell}{colback=black,colupper=white,colframe=yellow!75!black,
  listing only,listing options={style=tcblatex,language=sh},
  every listing line={\textcolor{red}{\small\ttfamily\bfseries root \$> }}}

\begin{commandshell}
ls -al
cd /usr/lib
\end{commandshell}
```

```
root $> ls -al
root $> cd /usr/lib
```

/tcb/every listing line*=⟨*text*⟩ (no default, initially unset/empty)

Identical to `/tcb/every listing line` plus additional enlargement of `/tcb/rightupper` → P. 39 by the width of ⟨*text*⟩. Therefore, this option has to be used after the geometry settings are done. This option is intended to be used in conjunction with `/tcb/hbox` → P. 89.

```
\newtcblisting{commandshell}{colback=black,colupper=white,colframe=yellow!75!black,
  listing only,listing options={style=tcblatex,language=sh},hbox,
  every listing line*={\textcolor{red}{\small\ttfamily\bfseries root \$> }}}

\begin{commandshell}
ls -al
cd /usr/lib
\end{commandshell}
```

```
root $> ls -al
root $> cd /usr/lib
```

See further options in Section 13.6 on page 254.

For an combined example of using `\lstinline` inside a `tcolorbox`, see `\DeclareTotalTCBox` → P. 362.

13.4 Option Keys of the LIB listingsutf8 Library

The LIB listingsutf8 library is an extension of the LIB listings library, so all options from Section 13.3 on page 249 are applicable.

`/tcb/listing utf8=`⟨*one-byte-encoding*⟩ (style, no default, initially `latin1`)

Abbreviation for using `/tcb/listing inputencoding`→ P. 250 together with UTF-8 support from the package `listingsutf8` [10]. This option is available only for the library variant LIB listingsutf8. The ⟨*one-byte-encoding*⟩ is one of the applicable encodings from [10], e. g. `latin1`.

See further options in Section 13.6 on page 254.

13.5 Option Keys of the minted Library

/tcb/minted language=⟨*programming language*⟩ (no default, initially `latex`)

Sets a ⟨*programming language*⟩ known to `Pygments` [12].

```
\begin{tcblisting}{listing engine=minted,minted style=trac,
  minted language=java,
  colback=red!5!white,colframe=red!75!black,listing only}
public class HelloWorld {
  // A 'Hello World' in Java
  public static void main(String[] args) {
    System.out.println("Hello World!");
  }
}
\end{tcblisting}
```

```
public class HelloWorld {
  // A 'Hello World' in Java
  public static void main(String[] args) {
    System.out.println("Hello World!");
  }
}
```

/tcb/minted options=⟨*key list*⟩ (no default, initially `tabsize=2,fontsize=\small`)

Sets the options from the package `minted` [14] which are used during typesetting of the listing.

```
% \tcbuselibrary{skins}
\newtcblisting{myjava}{listing engine=minted,minted style=colorful,
  minted language=java,minted options={fontsize=\small,linenos,numbersep=3mm},
  colback=blue!5!white,colframe=blue!75!black,listing only,
  left=5mm,enhanced,
  overlay={\begin{tcbclipinterior}\fill[red!20!blue!20!white] (frame.south west)
    rectangle ([xshift=5mm]frame.north west);\end{tcbclipinterior}}}

\begin{myjava}
public class HelloWorld {
  // A 'Hello World' in Java
  public static void main(String[] args) {
    System.out.println("Hello World!");
  }
}
\end{myjava}
```

```
1 public class HelloWorld {
2   // A 'Hello World' in Java
3   public static void main(String[] args) {
4     System.out.println("Hello World!");
5   }
6 }
```

`/tcb/minted style=`⟨*style*⟩ (no default, initially unset)

Sets a ⟨*style*⟩ known to `Pygments` [12]. This is independent from `/tcb/minted options` → P. 252. Note that styles are always applied globally; all following examples will be set in the given ⟨*style*⟩ until a new style is set. Also note that setting `\usemintedstyle{`⟨*style*⟩`}` only once per document is more economic, if all styles in a document are the same. For examples of different styles, see `/tcb/minted language` → P. 252 and `/tcb/minted options` → P. 252.

See further options in Section 13.6 on the following page.

13.6 Common Option Keys of all Libraries

For the ⟨*options*⟩ in `tcblisting` → P. 243 respectively `\tcbinputlisting` → P. 245 the following `pgf` keys can be applied. The key tree path `/tcb/` is not to be used inside these macros.

`/tcb/listing engine=`⟨*engine*⟩ (no default)

Sets the ⟨*engine*⟩ which typesets the listings. Feasible values are

- **`listings`**, if library LIB listings or LIB listingsutf8 is loaded.
- **`minted`**, if library LIB minted is loaded.

`/tcb/listing file=`⟨*file name*⟩ (no default, initially `\jobname.listing`)

Sets the ⟨*file name*⟩ of the file which is used to save listings.

`/tcb/listing and text` (no value, initially set)

Typesets the environment content as listing in the upper part and as compiled text in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing and text}
This is a \LaTeX\ example.
\end{tcblisting}
```

```
This is a \LaTeX\ example.
```

This is a LaTeX example.

`/tcb/text and listing` (no value)

Typesets the environment content as compiled text in the upper part and as listing in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text and listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.

```
This is a \LaTeX\ example.
```

`/tcb/listing only` (no value)

Typesets the environment content as listing.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing only}
This is a \LaTeX\ example.
\end{tcblisting}
```

```
This is a \LaTeX\ example.
```

/tcb/text only (no value)

Typesets the environment content as compiled text.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text only}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.

/tcb/comment=⟨*text*⟩ (no default, initially empty)

Records a comment with ⟨*text*⟩ as content. The comment is displayed e. g. in conjunction with **/tcb/listing and comment** → P. 257 and **/tcb/comment and listing** → P. 257.

```
\begin{tcblisting}{comment={This comment is really only a comment},
  colback=red!5!white,colframe=red!75!black}
This is a \textbf{tcolorbox}.
\end{tcblisting}
```

This is a \textbf{tcolorbox}.

This is a **tcolorbox**.

N 2014-11-17 **/tcb/comment only** (no value)

Typesets the environment content with the comment text.

```
\begin{tcblisting}{comment only,
  comment={This is a comment.},
  colback=red!5!white,colframe=red!75!black}
This is a \textbf{tcolorbox}.
\end{tcblisting}
```

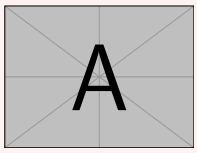
This is a comment.

/tcb/image comment={⟨*options*⟩**}{**⟨*filename*⟩**}** (style, no default, initially unset)

Uses an image denoted by ⟨*filename*⟩ as *comment* for the listing. The image is included by the standard **\includegraphics** macro with given ⟨*options*⟩.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing side comment,
  image comment={width=2.5cm}{example-image-a.pdf},center lower}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.



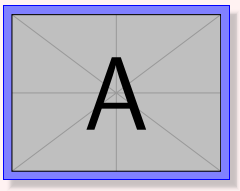
N 2014-11-14 **/tcb/tcbimage comment=⟨*filename*⟩** (style, no default, initially unset)

Uses an image denoted by ⟨*filename*⟩ as *comment* for the listing. The image is included by the `\tcbincludegraphics` → P. 218 macro. The inclusion can be customized by `/tcb/comment style` → P. 257.

The library LIB skins is needed to apply this option.

```
% \tcbuselibrary{skins}
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing side comment,
  righthand width=3cm,lower separated=false,
  tcbimage comment={example-image-a.pdf},
  comment style={size=fbox,colframe=blue,colback=blue!50,sharp corners,
    drop fuzzy shadow}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.



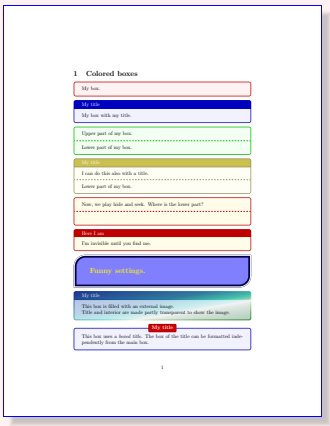
N 2014-11-14 **/tcb/pdf comment=⟨*filename*⟩** (style, default listing file, initially unset)

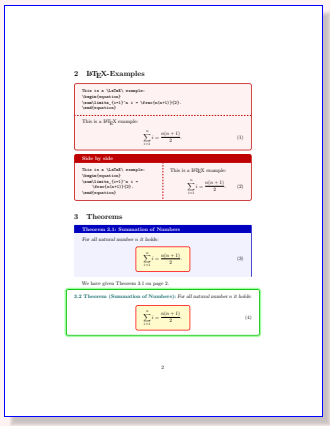
Uses a PDF file denoted by ⟨*filename*⟩ as *comment* for the listing. The image is included by `\tcbincludepdf` → P. 220 inside a `tcbraster` → P. 233. The inclusion can be customized by `/tcb/comment style` → P. 257.

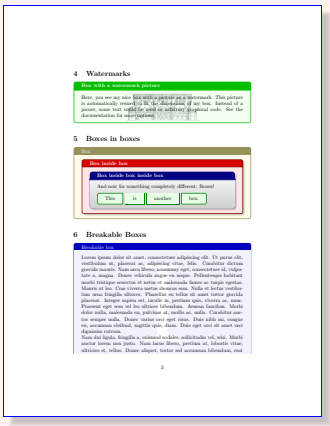
The libraries LIB skins and LIB raster are needed to apply this option.

```
% \tcbuselibrary{skins,raster}
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing and comment,
  righthand width=3cm,lower separated=false,middle=1mm,
  pdf comment={tcolorbox-example.pdf},
  comment style={raster columns=3,graphics pages={1,2,3},
    colframe=blue,drop fuzzy shadow}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.







N 2014-11-14 **/tcb/pdf extension=**⟨*extension*⟩ (no default, initially pdf)

Sets the PDF file name extension for /tcb/pdf comment → P. 256 to ⟨*extension*⟩. Note that ⟨*extension*⟩ always overwrites any actual extension given inside /tcb/pdf comment → P. 256.

N 2014-11-14 **/tcb/comment style=**⟨*options*⟩ (no default, initially empty)

Sets the ⟨*options*⟩ for /tcb/tcbimage comment → P. 256 and /tcb/pdf comment → P. 256. These are tcolorbox options to customize the colored box drawn around the image(s), also image options encapsulated by /tcb/graphics options → P. 221, and tcbraster → P. 233 options for /tcb/pdf comment → P. 256.

/tcb/listing and comment (no value)

Typesets the environment content as listing in the upper part and a given comment in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing and comment,
  comment={This is my comment. It may contain line breaks.\par
    It can even use the environment content
    \flqq\ignorespaces\tcbuselistingtext\unskip\frqq}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

This is my comment. It may contain line breaks.
It can even use the environment content «This is a LaTeX example.»

/tcb/comment and listing (no value)

Typesets a given comment in the upper part and the environment content as listing in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,comment and listing,
comment={This is my comment.}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is my comment.

This is a \LaTeX\ example.

/tcb/listing side text (no value)

Typesets the environment content side by side as listing in the left (upper) part and as compiled text in the right (lower) part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing side text}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

This is a LaTeX example.

/tcb/text side listing (no value)

Typesets the environment content side by side as compiled text in the left (upper) part and as listing in the right (lower) part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text side listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.

This is a \LaTeX\ example.

/tcb/listing outside text (no value)

Typesets the environment content side by side as listing in a `tcolorbox` and as compiled text outside the box in the right part of the page. Nevertheless, the outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The space partitioning is done with the side by side options from Section 4.14, see page 71.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing outside text}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

This is a LaTeX example.

/tcb/text outside listing (no value)

Typesets the environment content side by side as listing in a `tcolorbox` and as compiled text outside the box in the left part of the page. Nevertheless, the outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The space partitioning is done with the side by side options from Section 4.14, see page 71.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text outside listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.

This is a \LaTeX\ example.

/tcb/listing side comment (no value)

Typesets the environment content side by side as listing in the left (upper) part and a given comment in the right (lower) part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing side comment,
  righthand width=1.5cm,image comment={width=1.5cm}{example-image-a.pdf}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

/tcb/comment side listing (no value)

Typesets the environment content side by side with a given comment in the left (upper) part and as listing in the right (lower) part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,comment side listing,
  lefthand width=1.5cm,image comment={width=1.5cm}{example-image-a.pdf}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

/tcb/listing outside comment (no value)

Typesets the environment content side by side as listing in a `tcolorbox` and a given comment outside the box in the right part of the page. Nevertheless, the outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The space partitioning is done with the side by side options from Section 4.14, see page 71.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing outside comment,
  righthand width=1.5cm,image comment={width=1.5cm}{example-image-a.pdf}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

/tcb/comment outside listing (no value)

Typesets the environment content side by side as listing in a `tcolorbox` and a given comment outside the box in the left part of the page. Nevertheless, the outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The space partitioning is done with the side by side options from Section 4.14, see page 71.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,comment outside listing,
  lefthand width=1.5cm,image comment={width=1.5cm}{example-image-a.pdf}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.

/tcb/listing above text (no value)

Typesets the environment content as listing in a `tcolorbox` and as compiled text outside and below the box. The outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The distance between box and text is controlled by `/tcb/middle` → P. 41.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing above text}
This is a \LaTeX\ example.
\end{tcblisting}
```

```
This is a \LaTeX\ example.
```

This is a LaTeX example.

N 2014-11-07 **/tcb/listing above* text** (no value)

Widely equal to `/tcb/listing above text`, but the outside text is not formatted with the lower part options. Also, it is not put into a minipage and it may span several pages. The distance between box and text is controlled by `/tcb/after` → P. 76.

/tcb/text above listing (no value)

Typesets the environment content as listing in a `tcolorbox` and as compiled text outside and above the box. The outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The distance between box and text is controlled by `/tcb/middle` → P. 41.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text above listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.

```
This is a \LaTeX\ example.
```

N 2014-11-07 **/tcb/text above* listing** (no value)

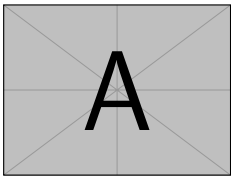
Widely equal to `/tcb/text above listing`, but the outside text is not formatted with the lower part options. Also, it is not put into a minipage and it may span several pages. The distance between box and text is controlled by `/tcb/before` → P. 76.

/tcb/listing above comment (no value)

Typesets the environment content as listing in a `tcolorbox` and a given comment outside and below the box. The outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The distance between box and comment is controlled by `/tcb/middle` → P. 41.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing above comment,
  center lower,image comment={width=3cm}{example-image-a.pdf}}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.



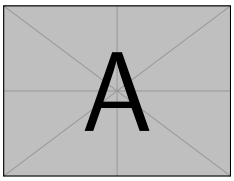
N 2014-11-07 **/tcb/listing above* comment** (no value)

Widely equal to `/tcb/listing above comment`, but the outside comment is not formatted with the lower part options. Also, it is not put into a minipage and it may span several pages. The distance between box and comment is controlled by `/tcb/after` → P. 76.

/tcb/comment above listing (no value)

Typesets the environment content as listing in a `tcolorbox` and a given comment outside and above the box. The outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The distance between box and comment is controlled by `/tcb/middle` → P. 41.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,comment above listing,
  center lower,image comment={width=3cm}{example-image-a.pdf}}
This is a \LaTeX\ example.
\end{tcblisting}
```



This is a \LaTeX\ example.

N 2014-11-07 **/tcb/comment above* listing** (no value)

Widely equal to `/tcb/comment above listing`, but the outside comment is not formatted with the lower part options. Also, it is not put into a minipage and it may span several pages. The distance between box and comment is controlled by `/tcb/before` → P. 76.

13.7 Option Keys for Processing and Full Document Examples

A complete LaTeX document including `\documentclass`, `\begin{document}` and `\end{document}` cannot be processed directly by `tcolorbox`. It always has to be compiled separately. There are two methods supported by the package to process and display such a full document example:

- Prepare and compile the example document independent from your main document. The source file and the resulting PDF file can be included into the main document afterwards. This is the most economic way since the example document can be left untouched after the example is complete.
- The other possibility is to compile the example on the fly while the main document is compiled. This way has some charm, because the example can be edited inside the main document. But be aware that the compilation of the example is issued on every run of the main document. Also, there are fewer degrees of freedom how the example is compiled.

For both methods, the resulting example PDF file can be included as a `/tcb/pdf comment` → P. 256.

The following example shows how to apply the first method. There already is a file `tcolorbox-example.tex` and a PDF file `tcolorbox-example.pdf`. Both of them are input partly by the following:

```
% \tcbuselibrary{breakable,skins,raster}
\tcbinputlisting{
  enhanced jigsaw,breakable,pad at break*=2mm,height fixed for=first and middle,
  lower separated=false,
  leftlower=0pt,rightlower=0pt,middle=0pt,
  colframe=red!50!black,colback=yellow!10!white,
  listing and comment,
  listing file={tcolorbox-example},
  listing options=
    {style=tcblatex,texcsstyle=*\color{red!70!black},firstline=20,lastline=85},
  after upper={\par\bigskip\texttt{\ldots}\par},
  pdf comment,
  comment style={drop lifted shadow,graphics pages={1,...,4}},
}
```

```
\documentclass{article}
\usepackage{tikz,lipsum,lmodern}
\usepackage[most]{tcolorbox}

\begin{document}

%-----------------------------------------------------------
\section{Colored boxes}

\begin{tcolorbox}[colback=red!5!white,colframe=red!75!black]
  My box.
\end{tcolorbox}

\begin{tcolorbox}[colback=blue!5!white,colframe=blue!75!black,title=My title]
  My box with my title.
\end{tcolorbox}

\begin{tcolorbox}[colback=green!5!white,colframe=green!75!black]
  Upper part of my box.
  \tcblower
```

```
  Lower part of my box.
\end{tcolorbox}

\begin{tcolorbox}[colback=yellow!5!white,colframe=yellow!50!black,
  colbacktitle=yellow!75!black,title=My title]
  I can do this also with a title.
  \tcblower
  Lower part of my box.
\end{tcolorbox}

\begin{tcolorbox}[colback=yellow!10!white,colframe=red!75!black,lowerbox=invisible,
  savelowerto=\jobname_ex.tex]
  Now, we play hide and seek. Where is the lower part?
  \tcblower
  I'm invisible until you find me.
\end{tcolorbox}

\begin{tcolorbox}[colback=yellow!10!white,colframe=red!75!black,title=Here I am]
  \input{\jobname_ex.tex}
\end{tcolorbox}


\begin{tcolorbox}[enhanced,sharp corners=uphill,
    colback=blue!50!white,colframe=blue!25!black,coltext=yellow,
    fontupper=\Large\bfseries,arc=6mm,boxrule=2mm,boxsep=5mm,
    borderline={0.3mm}{0.3mm}{white}]
  Funny settings.
\end{tcolorbox}


\begin{tcolorbox}[enhanced,frame style image=blueshade.png,
  opacityback=0.75,opacitybacktitle=0.25,
  colback=blue!5!white,colframe=blue!75!black,
  title=My title]
  This box is filled with an external image.\par
  Title and interior are made partly transparent to show the image.
\end{tcolorbox}


\begin{tcolorbox}[enhanced,attach boxed title to top
    center={yshift=-3mm,yshifttext=-1mm},
  colback=blue!5!white,colframe=blue!75!black,colbacktitle=red!80!black,
  title=My title,fonttitle=\bfseries,
  boxed title style={size=small,colframe=red!50!black} ]
  This box uses a \textit{boxed title}. The box of the title can
  be formatted independently from the main box.
\end{tcolorbox}

...
```

1 Colored boxes

My box.

My title

My box with my title.

Upper part of my box.

Lower part of my box.

My title

I can do this also with a title.

Lower part of my box.

Now, we play hide and seek. Where is the lower part?

Here I am

I'm invisible until you find me.

Funny settings.

My title

This box is filled with an external image.
Title and interior are made partly transparent to show the image.

My title

This box uses a *boxed title*. The box of the title can be formatted independently from the main box.

2 LaTeX-Examples

```
This is a \LaTeX\ example:
\begin{equation}
\sum\limits_{i=1}^n i = \frac{n(n+1)}{2}.
\end{equation}
```

This is a LaTeX example:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}. \tag{1}$$

Side by side

```
This is a \LaTeX\ example:
\begin{equation}
\sum\limits_{i=1}^n i =
    \frac{n(n+1)}{2}.
\end{equation}
```

This is a LaTeX example:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}. \tag{2}$$

3 Theorems

Theorem 3.1: Summation of Numbers

For all natural number n it holds:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}. \tag{3}$$

We have given Theorem 3.1 on page 2.

3.2 Theorem (Summation of Numbers): *For all natural number n it holds:*

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}. \tag{4}$$

4 Watermarks

Box with a watermark picture

Here, you see my nice box, with a picture as a watermark. This picture is automatically resized to fit the dimensions of my box. Instead of a picure, some text could be used or arbitrary graphical code. See the documentation for more options.

5 Boxes in boxes

Box

Box inside box

Box inside box inside box

And now for something completely different: Boxes!

This | is | another | box.

6 Breakable Boxes

Breakable box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.
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.
Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.
Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.
Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.
Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.
Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin

N 2014-11-14 **/tcb/no process** (no default)

Removes all processing commands if set before.

N 2014-11-14 **/tcb/process code=**⟨*code*⟩ (no default, initially empty)

Adds ⟨*code*⟩ which is executed during `\tcbinputlisting` → P. 245 and `tcblisting` → P. 243. At the time of executing the given ⟨*code*⟩, the listing is already written to `/tcb/listing file` → P. 254, but the colored box is not constructed yet. Its intended use is to process the listing somehow before displaying. The processing result can be used inside a `/tcb/comment` → P. 255. Several `/tcb/process code` options can be given which are processed in the given order.

To use the further options, the compiler has to be called with the `-shell-escape` permission to authorize potentially dangerous system calls. Be warned that this is a security risk. Anyway, it's more economic to compile examples independent from the main document and to include them as shown in the previous pages.

N 2014-11-14 **/tcb/run system command=**⟨*system command*⟩ (style, no default, initially unset)

Runs a ⟨*system command*⟩, if the document is compiled with the `-shell-escape` permission. The current listing file can be accessed as `\filename@area\filename@base\filename@ext`.

N 2014-11-14 **/tcb/compilable listing** (style, no default)

Sets `/tcb/listing file` → P. 254 to `\jobname-listing-`⟨*counter*⟩.

The default `/tcb/listing file` → P. 254 setting cannot be used to compile a listing, since the base name equals the `\jobname` and the included PDF files should be unique. Therefore, to use `/tcb/run pdflatex` etc., the `/tcb/listing file` → P. 254 has to be set to a unique value. One may use `/tcb/compilable listing` for this purpose.

N 2014-11-14 **/tcb/run pdflatex=**⟨*arguments*⟩ (style, no default, initially unset)

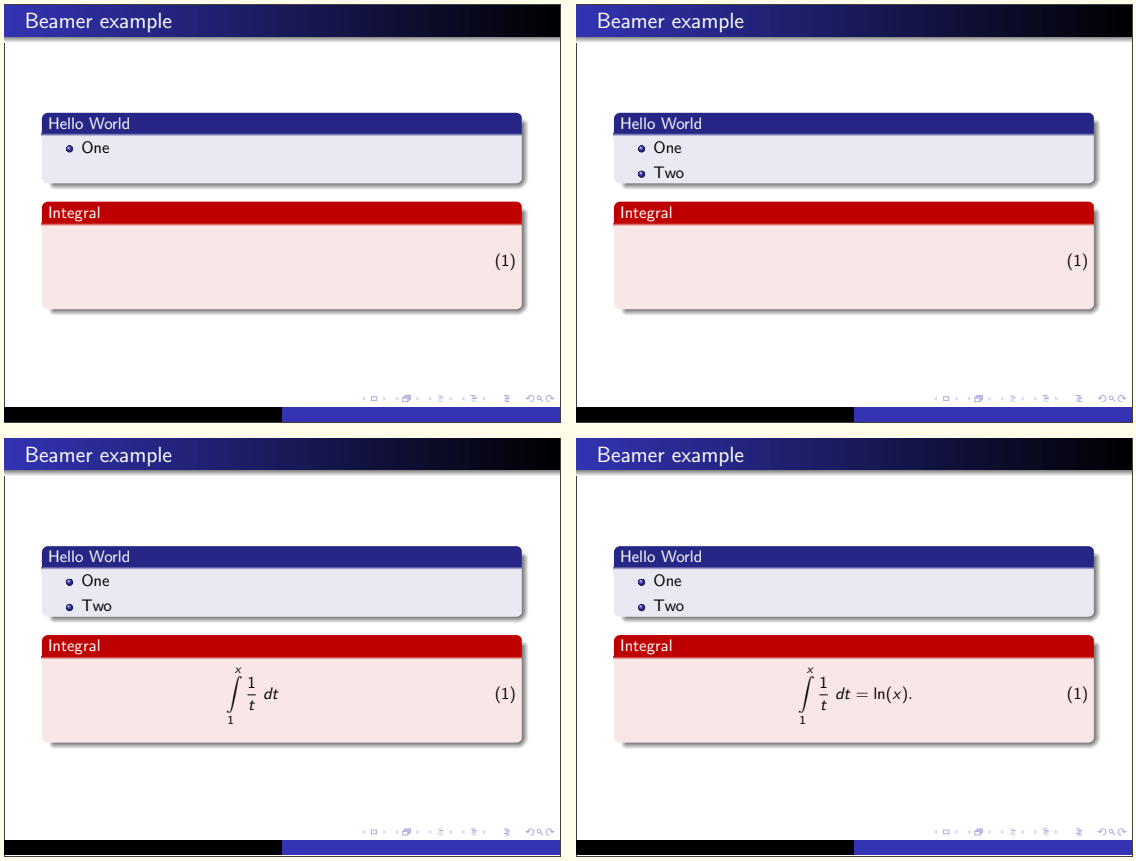
Issues a `pdflatex` compilation of the listing with the given ⟨*arguments*⟩.

- The main document has to be compiled with the `-shell-escape` permission.
- The `/tcb/listing file` → P. 254 has to be unique for the listing.
- If the listing has to be compiled twice, add `run pdflatex` two times to the option list.

```
\begin{tcblisting}{enhanced jigsaw,lower separated=false,
  leftlower=0pt,rightlower=0pt,
  colframe=red!50!black,colback=yellow!10!white,
  listing options={style=tcblatex,texcsstyle=*\color{red!70!black}},
  listing and comment,
  pdf comment,
  compilable listing,
  run pdflatex
}
\documentclass{beamer}
\usetheme{Warsaw}
\begin{document}
\begin{frame}{Beamer example}
  \begin{block}{Hello World}
    \begin{itemize}[<+->]
      \item One
      \item Two
    \end{itemize}
  \end{block}

  \begin{alertblock}{Integral}
    \begin{equation}
      \visible<3->{\int\limits_1^x \frac{1}{t}~dt}
      \visible<4->{ = \ln(x).}
    \end{equation}
```

```
  \end{alertblock}
\end{frame}
\end{document}
\end{tcblisting}
```

```
\documentclass{beamer}
\usetheme{Warsaw}
\begin{document}
\begin{frame}{Beamer example}
  \begin{block}{Hello World}
    \begin{itemize}[<+->]
      \item One
      \item Two
    \end{itemize}
  \end{block}

  \begin{alertblock}{Integral}
    \begin{equation}
      \visible<3->{\int\limits_1^x \frac{1}{t}~dt}
      \visible<4->{ = \ln(x).}
    \end{equation}
  \end{alertblock}
\end{frame}
\end{document}
```



N 2014-11-14 **/tcb/run xelatex=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `xelatex` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run lualatex=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `lualatex` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run makeindex=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `makeindex` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run bibtex=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `bibtex` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run biber=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `biber` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run arara=**⟨*arguments*⟩ (style, no default, initially unset)

Issues an `arara` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run latex=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `latex` compilation of the listing with the given ⟨*arguments*⟩.

N 2014-11-14 **/tcb/run dvips=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `dvips` compilation of the listing with the given ⟨*arguments*⟩.

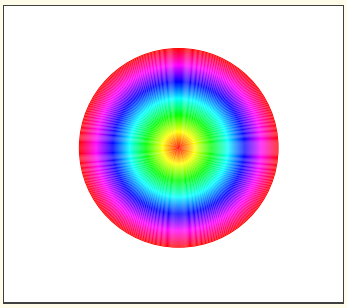
N 2014-11-14 **/tcb/run ps2pdf=**⟨*arguments*⟩ (style, no default, initially unset)

Issues a `ps2pdf` compilation of the listing with the given ⟨*arguments*⟩.

```
\begin{tcblisting}{enhanced jigsaw,
  title={PSTricks with pdflatex},fonttitle=\bfseries,
  colframe=red!50!black,colback=yellow!10!white,
  listing options={style=tcblatex,texcsstyle=*\color{red!70!black}},
  lower separated=false,middle=0pt,
  listing side comment,righthand width=4.5cm,
  compilable listing,
  run latex,run dvips,run ps2pdf,
  pdf comment,
  comment style={raster columns=1,graphics options={viewport=0in 8in 4in 11.5in,clip}},
}
\documentclass{article}
\usepackage{pstricks,multido}
\begin{document}
\psset{unit=3}%
\multido{\nHue=0.01+0.01}{100}{%
\definecolor{MyColor}{hsb}{\nHue,1,1}%
\pscircle[linewidth=0.01,linecolor=MyColor]{\nHue}}
\end{document}
\end{tcblisting}
```

PSTricks with pdflatex

```
\documentclass{article}
\usepackage{pstricks,multido}
\begin{document}
\psset{unit=3}%
\multido{\nHue=0.01+0.01}{100}{%
\definecolor{MyColor}{hsb}{\nHue,1,1}%
\pscircle[linewidth=0.01,linecolor=MyColor]{\nHue}}
\end{document}
```



13.8 Creation of LaTeX Tutorials

The following source code gives a guideline for the creation of LaTeX tutorials. In the next section, a framework for LaTeX exercises is described. All examples shall be numbered optionally.

Firstly, some additional `tcb` keys are defined for the appearance. For the examples, three environments `texexp`, `texexptitled`, and `texexptitledspec` are defined with automatic numbering.

- `texexp` is used for untitled examles,
- `texexptitled` is used for titled examles,
- `texexptitledspec` is used for titled examles with special treatment.

Definition in the preamble:

```
\tcbset{
  texexp/.style={colframe=red!50!yellow!50!black, colback=red!50!yellow!5!white,
      coltitle=red!50!yellow!3!white,
      fonttitle=\small\sffamily\bfseries, fontupper=\small, fontlower=\small},
  example/.style 2 args={texexp,
      title={Example \thetcbcounter: #1},label={#2}},
}


\newtcblisting{texexp}[1]{texexp,#1}
\newtcblisting[auto counter,number within=section]{texexptitled}[3][]{%
  example={#2}{#3},#1}
\newtcolorbox[use counter from=texexptitled]{texexptitledspec}[3][]{%
  example={#2}{#3},#1}
```

```
\begin{tcblisting}{texexp}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{tcblisting}
```

```
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
```

This is a LaTeX example which displays the text as source code and in compiled form.

```
\begin{texexptitled}{First example with a title line}{firstExample}
Here, we use Example \ref{firstExample} with a title line.
\end{texexptitled}
```

Example 13.1: First example with a title line

```
Here, we use Example \ref{firstExample} with a title line.
```

Here, we use Example 13.1 with a title line.

```
\begin{texexp}{}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{texexp}
```

```
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
```

This is a LaTeX example which displays the text as source code and in compiled form.

```
\begin{texexp}{text and listing}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{texexp}
```

This is a LaTeX example which displays the text as source code and in compiled form.

```
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
```

```
\begin{texexp}{listing only}
This is a \LaTeX\ example which displays the text as source code only.
\end{texexp}
```

```
This is a \LaTeX\ example which displays the text as source code only.
```

```
\begin{texexp}{text only}
This is a \LaTeX\ example which displays the text in compiled form only.
\end{texexp}
```

This is a LaTeX example which displays the text in compiled form only.

```
\begin{texexptitled}{An Example with a Heading}{heading1}
This is a \LaTeX\ example with a numbered heading line
which can be referred to.
\end{texexptitled}
Here, we see Example \ref{heading1}.
```

Example 13.2: An Example with a Heading

```
This is a \LaTeX\ example with a numbered heading line
which can be referred to.
```

This is a LaTeX example with a numbered heading line which can be referred to.

Here, we see Example 13.2.

```
\begin{texexptitled}[listing only]{Another Example with a Heading}{heading2}
The keys can be used in combination. Here, an example with a heading line
and source code only is given.
\end{texexptitled}
Here, we see Example \ref{heading2}.
```

Example 13.3: Another Example with a Heading

The keys can be used in combination. Here, an example with a heading line and source code only is given.

Here, we see Example 13.3.

```
\begin{texexptitled}[float]{A floating Example with a Heading}{heading3}
This is another \LaTeX\ example with numbered heading line.
But now, the box is a floating object.
\end{texexptitled}
```

Example 13.4: A floating Example with a Heading

This is another \LaTeX\ example with numbered heading line.
But now, the box is a floating object.

This is another LaTeX example with numbered heading line. But now, the box is a floating object.

```
The floating box of the last example is seen as Example \ref{heading3}
on page \pageref{heading3}.
```

The floating box of the last example is seen as Example 13.4 on page 270.

```
\begin{texexptitledspec}{Special application}{texexpbox1}
\begin{lstlisting}[style=tcblatex]
Some \LaTeX\ source code.
\end{lstlisting}
\tcblower
For special cases, the environment |texexptitledspec| with style
|example| can be used directly. As one can see, the upper and the lower
part of the box can be used uncoupled also.
\end{texexptitledspec}
```

Example 13.5: Special application

Some \LaTeX\ source code.

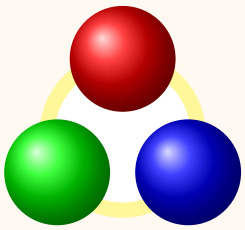
For special cases, the environment `texexptitledspec` with style `example` can be used directly. As one can see, the upper and the lower part of the box can be used uncoupled also.

The following series of examples demonstrate the application of `tcolorbox`→ P. 11 options for diversification.

```
\begin{texexptitled}{How to use options (1):\par The basic example}{options1}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 13.6: How to use options (1): The basic example

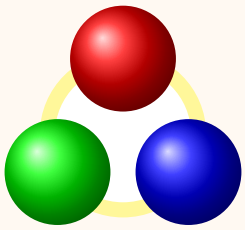
```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
```



```
\begin{texexptitled}[center lower,enhanced,segmentation hidden,middle=0mm]
  {How to use options (2):\par The text output is centered and the
    segmentation line has vanished.}{options2}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 13.7: How to use options (2): The text output is centered and the segmentation line has vanished.

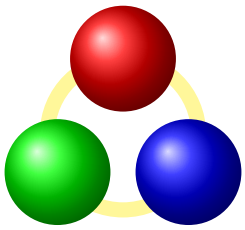
```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
```



```
\begin{texexptitled}[tikz lower,bicolor,colbacklower=white]
  {How to use options (3):\par Here, the |tikzpicture| is totally hidden.
    The |bicolor| skin highlights the output.}{options3}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{texexptitled}
```

Example 13.8: How to use options (3):
Here, the `tikzpicture` is totally hidden. The `bicolor` skin highlights the output.

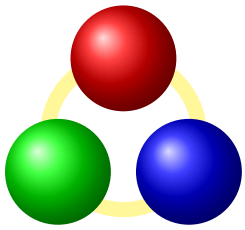
```
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
```



```
\begin{texexptitled}[center lower,listing side text,righthand width=3.5cm,
  bicolor,colbacklower=white]
  {How to use options (4):\par The |bicolor| skin also works with side
   by side mode}{options4}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 13.9: How to use options (4):
The `bicolor` skin also works with side by side mode

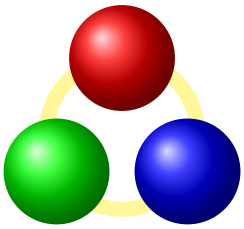
```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```



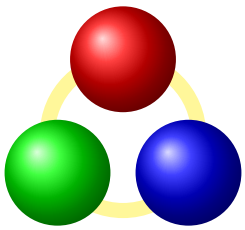
```
\begin{texexptitled}[center lower,listing outside text,righthand width=3.5cm]
  {How to use options (5):\par Putting our picture outside is just
    a matter of one word.}{options5}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 13.10: How to use options (5): Putting our picture outside is just a matter of one word.

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```



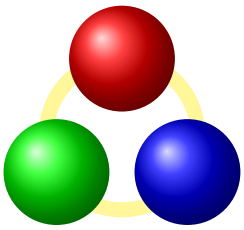
```
\begin{texexptitled}[center lower,text above listing]
  {How to use options (6):\par The picture may also be put above
    the listing box.}{options6}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```



Example 13.11: How to use options (6): The picture may also be put above the listing box.

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```

```
\begin{texexptitled}[beamer,center lower,text outside listing,lefthand width=3.5cm]
  {How to use options (7):\par Our style is easily transformed into
    a beamerish one.}{options7}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```



Example 13.12: How to use options (7):
Our style is easily transformed into a beamerish one.

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```

13.9 Creation of LaTeX Exercises

In the following, a guideline is given for the creation of LaTeX exercises with solutions. These solutions are saved to disk for application at a place of choice. Therefore, all used exercises are logged to a file `\jobname.records` for automatic processing. The solution contents themselves are saved to a subdirectory named `solutions`. Also see Section 7 on page 106.

- Before the first exercise is given, `\tcbstartrecording` → P. 106 has to be called to start recording.
- The solution is given as content of a `tcboutputlisting` → P. 245 environment. Note, that you can use this content also inside the exercise with `\tcbuselistingtext` → P. 245 in compiled form.
- After the last exercise is given (and before using the solutions), `\tcbstoprecording` → P. 106 has to be called to stop recording.
- The solutions are loaded by `\tcbinputrecords` → P. 106.

Inside the exercise text, there may be text parts which are needed as LaTeX source code and as compiled text as well. These parts can be saved by `tcbwritetemp` → P. 105 and used in compiled form by `\tcbusetemp` → P. 105 or as source code by `\tcbusetemplisting` → P. 245.

At first, we generate some a common style for the exercises and the solutions. Further, since exercises and solutions should be numbered, we force to use a label ⟨*marker*⟩. Automatically, the label `exe:`⟨*marker*⟩ is used to mark the exercise and the label `sol:`⟨*marker*⟩ is used to mark the solution.

```
\tcbset{texercisestyle/.style={arc=0.5mm, colframe=blue!25!yellow!90!white,
  colback=blue!25!yellow!5!white, coltitle=blue!25!yellow!40!black,
  fonttitle=\small\sffamily\bfseries, fontupper=\small, fontlower=\small,
  listing options={style=tcblatex,texcsstyle=*\color{red!40!black}},
}}
```

With these preparations, the kernel environment `texercise` for our exercises is created quickly:

Definition in the preamble:

```
\newtcolorbox[auto counter,number within=section,list inside=exam]{texercise}[2][]{%
  texercisestyle,
  listing file={solutions/texercise\thetcbcounter.tex},
  label={exe:#2},
  record={\string\processsol{solutions/texercise\thetcbcounter.tex}{#2}},
  title={Exercise \thetcbcounter\hfill\mdseries Solution on page \pageref{sol:#2}},
  list text={Exercise with solution on page \pageref{sol:#2}},#1}
```

The following examples demonstrate the application.

```
\tcbstartrecording
```

```
\begin{texercise}{tabular_example}
\textit{Create the following table:}\par\smallskip%
\begin{tcboutputlisting}
\begin{tabular}{|p{3cm}|p{3cm}|p{3cm}|p{3cm}|}\hline
\multicolumn{4}{|c|}{\bfseries\itshape Das alte Italien}\\\hline
\multicolumn{2}{|c|}{\bfseries Antike} &
\multicolumn{2}{c|}{\bfseries Mittelalter}\\\hline
\multicolumn{1}{|c|}{\itshape Republik}&
\multicolumn{1}{c|}{\itshape Kaiserreich}&
\multicolumn{1}{c|}{\itshape Franken}&
\multicolumn{1}{c|}{\itshape Teilstaaten}\\\hline
In den Zeiten der r\"{o}mischen Republik standen dem Staat jeweils zwei
Konsuln vor, deren Machtbefugnisse identisch waren. &
Das r\"{o}mische Kaiserreich wurde von einem Alleinherrscher, dem Kaiser,
regiert.
& In der V\"{o}lkerwanderungszeit \"{u}bernahmen die Goten und sp\"{a}ter die
Franken die Vorherrschaft.
& Im sp\"{a}teren Mittelalter regierten F\"{u}rsten einen Fleckenteppich
von Einzelstaaten.\\\hline
\end{tabular}
\end{tcboutputlisting}
\tcbuselistingtext%
\end{texercise}
```

Exercise 13.1 Solution on page 279

Create the following table:

Das alte Italien			
Antike		**Mittelalter**	
Republik	*Kaiserreich*	*Franken*	*Teilstaaten*
In den Zeiten der römischen Republik standen dem Staat jeweils zwei Konsuln vor, deren Machtbefugnisse identisch waren.	Das römische Kaiserreich wurde von einem Alleinherrscher, dem Kaiser, regiert.	In der Völkerwanderungszeit übernahmen die Goten und später die Franken die Vorherrschaft.	Im späteren Mittelalter regierten Fürsten einen Fleckenteppich von Einzelstaaten.

```
\begin{texercise}{macro_oneparam}
\begin{tcboutputlisting}
\newcommand{\headingline}[1]{%
  \begin{center}\Large\bfseries #1\end{center}}
\end{tcboutputlisting}
\tcbuselistingtext%

Create a new macro \verb+\headingline+ which produces the
following output:\par\smallskip
\begin{tcbwritetemp}
\headingline{Very important heading}
\end{tcbwritetemp}
\tcbusetemplisting\tcbusetemp%
\end{texercise}
```

Exercise 13.2 Solution on page 279

Create a new macro `\headingline` which produces the following output:

```
\headingline{Very important heading}
```

Very important heading

```
\begin{texercise}{macro_twoparam}
\begin{tcboutputlisting}
\newcommand{\minitable}[2]{%
  \begin{center}\begin{tabular}{p{10cm}}\hline%
  \multicolumn{1}{c}{\bfseries#1}\\\hline%
  #2\\\hline%
  \end{tabular}\end{center}}
\end{tcboutputlisting}
\tcbuselistingtext%
Create a new macro \verb+\minitable+ which produces the
following output:\par\smallskip
\begin{tcbwritetemp}
\minitable{My heading}{In this tiny tabular, there is only a heading
  and some text below which has a width of ten centimeters.}
\end{tcbwritetemp}
\tcbusetemplisting\par\smallskip\tcbusetemp%
\end{texercise}
```

Exercise 13.3 Solution on page 279

Create a new macro `\minitable` which produces the following output:

```
\minitable{My heading}{In this tiny tabular, there is only a heading
  and some text below which has a width of ten centimeters.}
```

My heading
In this tiny tabular, there is only a heading and some text below which has a width of ten centimeters.

```
\begin{texercise}{macro_threeparam}
\begin{tcboutputlisting}
\newcommand{\synop}[3]{%
  \begin{tabular}{@{}p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}|%
    p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}@{}}\hline
  \multicolumn{2}{c}{\bfseries #1}\\\hline
  \multicolumn{1}{c|}{\itshape English}&
  \multicolumn{1}{c}{\itshape German}\\\hline
  #2 & #3
  \end{tabular}}
\end{tcboutputlisting}
\tcbuselistingtext%
Create a new macro \verb+\synop+ which typesets a synoptic text according
to the following example. Base your macro on a tabular which takes the
total line width.\par\smallskip
\begin{tcbwritetemp}
\synop{Neil Armstrong}%
{That's one small step for a man, one giant leap for mankind.}%
{Das ist ein kleiner Schritt f\"{u}r einen Mann,
   ein riesiger Sprung f\"{u}r die Menschheit.}
\end{tcbwritetemp}
\tcbusetemplisting\par\smallskip\tcbusetemp%
\end{texercise}
```

Exercise 13.4 Solution on page 280

Create a new macro `\synop` which typesets a synoptic text according to the following example. Base your macro on a tabular which takes the total line width.

```
\synop{Neil Armstrong}%
{That's one small step for a man, one giant leap for mankind.}%
{Das ist ein kleiner Schritt f\"{u}r einen Mann,
   ein riesiger Sprung f\"{u}r die Menschheit.}
```

Neil Armstrong	
English	*German*
That's one small step for a man, one giant leap for mankind.	Das ist ein kleiner Schritt für einen Mann, ein riesiger Sprung für die Menschheit.

```
\tcbstoprecording
```

Now, we give a list of all exercises with:

```
\tcblistof[\subsection]{exam}{List of Exercises%
  \label{listofexercises}}
```

13.10 List of Exercises

13.1 Exercise with solution on page 279 276
13.2 Exercise with solution on page 279 277
13.3 Exercise with solution on page 279 277
13.4 Exercise with solution on page 280 278

13.11 Solutions for the given LaTeX Exercises

For all solutions, a macro `\processsol` was written to the file `\jobname.records`. Now, we need a definition for this macro to use the solutions.

```
% \usepackage{hyperref} % for phantomlabel
\newtcbinputlisting{\processsol}[2]{%
  texercisestyle,
  listing only,
  listing file={#1},
  phantomlabel={sol:#2},%
  title={Solution for Exercise \ref{exe:#2} on page \pageref{exe:#2}},
}
```

The loading of all solutions is done by:

```
\tcbinputrecords
```

With this, we get:

Solution for Exercise 13.1 on page 276

```
\begin{tabular}{|p{3cm}|p{3cm}|p{3cm}|p{3cm}|}\hline
\multicolumn{4}{|c|}{\bfseries\itshape Das alte Italien}\\\hline
\multicolumn{2}{|c|}{\bfseries Antike} &
\multicolumn{2}{c|}{\bfseries Mittelalter}\\\hline
\multicolumn{1}{|c|}{\itshape Republik}&
\multicolumn{1}{c|}{\itshape Kaiserreich}&
\multicolumn{1}{c|}{\itshape Franken}&
\multicolumn{1}{c|}{\itshape Teilstaaten}\\\hline
In den Zeiten der r\"{o}mischen Republik standen dem Staat jeweils zwei
Konsuln vor, deren Machtbefugnisse identisch waren. &
Das r\"{o}mische Kaiserreich wurde von einem Alleinherrscher, dem Kaiser,
regiert.
& In der V\"{o}lkerwanderungszeit \"{u}bernahmen die Goten und sp\"{a}ter die
Franken die Vorherrschaft.
& Im sp\"{a}teren Mittelalter regierten F\"{u}rsten einen Fleckenteppich
von Einzelstaaten.\\\hline
\end{tabular}
```

Solution for Exercise 13.2 on page 277

```
\newcommand{\headingline}[1]{%
  \begin{center}\Large\bfseries #1\end{center}}
```

Solution for Exercise 13.3 on page 277

```
\newcommand{\minitable}[2]{%
  \begin{center}\begin{tabular}{p{10cm}}\hline%
  \multicolumn{1}{c}{\bfseries#1}\\\hline%
  #2\\\hline%
  \end{tabular}\end{center}}
```

Solution for Exercise 13.4 on page 278

```
\newcommand{\synop}[3]{%
  \begin{tabular}{@{}p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}|%
    p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}@{}}\hline
  \multicolumn{2}{c}{\bfseries #1}\\\hline
  \multicolumn{1}{c|}{\itshape English}&
  \multicolumn{1}{c}{\itshape German}\\\hline
  #2 & #3
  \end{tabular}}
```

14 Library theorems

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{theorems}
```

This also loads the package `amsmath`.

14.1 Macros of the Library

\newtcbtheorem[⟨*init options*⟩]{⟨*name*⟩}{⟨*display name*⟩}{⟨*options*⟩}{⟨*prefix*⟩}

Creates new environments ⟨*name*⟩ and ⟨*name*⟩* based on `tcolorbox` to frame a (mathematical) theorem. The ⟨*display name*⟩ is used in the title line with a number, e. g. «Theorem 5.1». The ⟨*options*⟩ are given to the underlying `tcolorbox` to control the appearance. The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 on page 97. The new environment ⟨*name*⟩ takes one optional and two mandatory parameters. The optional parameter supplements the options and should be used only in rare cases. The first mandatory parameter is the title text for the theorem and the second mandatory parameter is a ⟨*marker*⟩. The theorem is automatically labeled with ⟨*prefix*⟩:⟨*marker*⟩. The new environment ⟨*name*⟩* takes one optional and one mandatory parameter and represents an unnumbered variant of the environment ⟨*name*⟩. This variant is not labeled and not listed in lists of theorems.

Definition in the preamble:

```
\newtcbtheorem[number within=section]{mytheo}{My Theorem}%
  {colback=green!5,colframe=green!35!black,fonttitle=\bfseries}{th}
```

```
\begin{mytheo}{This is my title}{theoexample}
  This is the text of the theorem. The counter is automatically assigned and,
  in this example, prefixed with the section number. This theorem is numbered with
  \ref{th:theoexample} and is given on page \pageref{th:theoexample}.
\end{mytheo}
```

My Theorem 14.1: This is my title

This is the text of the theorem. The counter is automatically assigned and, in this example, prefixed with the section number. This theorem is numbered with 14.1 and is given on page 281.

```
\begin{mytheo}[label=myownlabel]{This is my title}{}
  The label parameter can be left empty without \LaTeX\ error.
  Or you may use an own label to reference Theorem \ref{myownlabel}.
\end{mytheo}
```

My Theorem 14.2: This is my title

The label parameter can be left empty without LaTeX error. Or you may use an own label to reference Theorem 14.2.

```
\begin{mytheo}{}{}
  The title can also be left empty without problem. Note that the ':'
  vanished magically.
\end{mytheo}
```

My Theorem 14.3

The title can also be left empty without problem. Note that the ':' vanished magically.

```
\begin{mytheo*}{Unnumbered Theorem}
  This theorem is not numbered.
\end{mytheo*}
```

My Theorem: Unnumbered Theorem

This theorem is not numbered.

```
\begin{mytheo*}{}
  This theorem has no number and no title.
\end{mytheo*}
```

My Theorem

This theorem has no number and no title.

\renewtcbtheorem[⟨*init options*⟩]{⟨*name*⟩}{⟨*display name*⟩}{⟨*options*⟩}{⟨*prefix*⟩}

Operates like \newtcbtheorem → P. 281, but based on \renewenvironment instead of \newenvironment. An existing environment is redefined.

\tcbmaketheorem{⟨*name*⟩}{⟨*display name*⟩}{⟨*options*⟩}{⟨*counter*⟩}{⟨*prefix*⟩}

\newtcbtheorem → P. 281 supersedes this macro.

Creates a new environment ⟨*name*⟩ based on tcolorbox to frame a (mathematical) theorem. The ⟨*display name*⟩ is used in the title line with a number, e. g. «Theorem 5.1». The ⟨*options*⟩ are given to the underlying tcolorbox to control the appearance. The ⟨*counter*⟩ is used for automatic numbering. The new environment ⟨*name*⟩ takes one optional and two mandatory parameters. The optional parameter supplements the options and should be used only in rare cases. The first mandatory parameter is the title text for the theorem and the second mandatory parameter is a ⟨*marker*⟩. The theorem is automatically labeled with ⟨*prefix*⟩:⟨*marker*⟩.

deprecated

\tcboxmath[⟨*options*⟩]{⟨*mathematical box content*⟩}

Creates a tcolorbox→P. 11 which is fitted to the width of the given ⟨*mathematical box content*⟩. This box is intended to be applied as part of a larger formula and may be used as replacement for the \boxed macro of amsmath.

```
\begin{equation}
\tcbset{fonttitle=\scriptsize}
\tcboxmath[colback=LightBlue!25!white,colframe=blue]{ a^2 = 16 }
\quad \Rightarrow \quad
\tcboxmath[colback=Salmon!25!white,colframe=red,title=Implication]%
  { a = 4 ~\vee~ a=-4. }
\end{equation}
```

$$\boxed{a^2 = 16} \quad \Rightarrow \quad \boxed{\begin{array}{c}\text{Implication}\\ a = 4 \ \vee \ a = -4.\end{array}} \tag{3}$$

\tcbhighmath[⟨*options*⟩]{⟨*mathematical box content*⟩}

This is a special case of the \tcboxmath macro which uses the style /tcb/highlight math→P. 290. It is intended to provide context sensitive highlighting of formula parts. The color settings via /tcb/highlight math style→P. 290 may be different inside theorems or other colored areas and outside.

```
\tcbset{myformula/.style={colback=yellow!10!white,colframe=red!50!black,
  every box/.style={highlight math style={colback=LightBlue!50!white,colframe=Navy}}
  }}

\begin{align}
  \tcbhighmath{\sum\limits_{n=1}^{\infty} \frac{1}{n}} &= \infty.\\
  \int x^2 ~\text{d}x                                  &= \frac13 x^3 + c.
\end{align}

\begin{tcolorbox}[ams align,myformula]
  \tcbhighmath{\sum\limits_{n=1}^{\infty} \frac{1}{n}} &= \infty.\\
  \int x^2 ~\text{d}x                                  &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\boxed{\sum_{n=1}^{\infty} \frac{1}{n}} = \infty. \tag{4}$$

$$\int x^2 \, \mathrm{d}x = \frac{1}{3}x^3 + c. \tag{5}$$

$$\boxed{\sum_{n=1}^{\infty} \frac{1}{n}} = \infty. \tag{6}$$

$$\int x^2 \, \mathrm{d}x = \frac{1}{3}x^3 + c. \tag{7}$$

\tcbhighmath→ P. 283 can be used in symbiosis with the **empheq** package which allows to specify own boxing commands to mark multiline formulas.

```
% \usepackage{empheq}
\begin{empheq}[box=\tcbhighmath]{align}
a&=\sin(z)\\
E&=mc^2 + \int_a^b x\, dx
\end{empheq}

\tcbset{highlight math style={enhanced,
  colframe=red!60!black,colback=yellow!50!white,arc=4pt,boxrule=1pt,
  drop fuzzy shadow}}

\begin{empheq}[box=\tcbhighmath]{align}
a&=\sin(z)\\
E&=mc^2 + \int_a^b x\, dx
\end{empheq}
```

$$a = \sin(z) \tag{8}$$
$$E = mc^2 + \int_a^b x\, dx \tag{9}$$

$$a = \sin(z) \tag{10}$$
$$E = mc^2 + \int_a^b x\, dx \tag{11}$$

Besides \tcbhighmath→ P. 283, one can easily define an independent new box based on \tcbox→ P. 13 which acts like \tcbhighmath→ P. 283:

```
% \usepackage{empheq}
\newtcbox{\otherbox}[1][]{nobeforeafter,math upper,tcbox raise base,
  enhanced,frame hidden,boxrule=0pt,interior style={top color=green!10!white,
  bottom color=green!10!white,middle color=green!50!yellow},
  fuzzy halo=1pt with green,#1}

\begin{empheq}[box=\otherbox]{align}
a&=\sin(z)\\
E&=mc^2 + \int_a^b x\, dx
\end{empheq}

\begin{equation}
\tcbhighmath{E} = \otherbox{mc^2}
\end{equation}
```

$$a = \sin(z) \tag{12}$$
$$E = mc^2 + \int_a^b x\, dx \tag{13}$$

$$\boxed{E} = \boxed{mc^2} \tag{14}$$

14.2 Option Keys of the Library

/tcb/separator sign=⟨*sign*⟩ (no default, initially :)

The given ⟨*sign*⟩ is used inside the title text of a theorem as separater between display name combined with number and the specific title text. It is omitted, if there is no specific title text.

```
% \usepackage{amssymb}
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   separator sign={\ $\blacktriangleright$}}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.4 ▶ My example

My theorem text.

/tcb/separator sign colon (style, no value, initially set)

Sets /tcb/separator sign to the default colon : sign.

/tcb/separator sign dash (style, no value)

Sets /tcb/separator sign to an en-dash sign.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   separator sign dash}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.5 – My example

My theorem text.

/tcb/separator sign none (style, no value)

Sets /tcb/separator sign to empty.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   separator sign none}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.6 My example

My theorem text.

/tcb/description delimiters={⟨*left*⟩}{⟨*right*⟩} (no default, initially empty)

The given ⟨*left*⟩ and ⟨*right*⟩ delimiter signs are used to frame the descriptive title text of a theorem.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   description delimiters={\flqq}{\frqq}}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.7: «My example»

My theorem text.

/tcb/description delimiters parenthesis (style, no value)

Sets /tcb/description delimiters to (and).

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   description delimiters parenthesis}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.8: (My example)

My theorem text.

/tcb/description delimiters none (style, no value, initially set)

Sets /tcb/description delimiters to the default empty texts.

/tcb/description color=⟨*color*⟩ (default empty, initially empty)

Sets the ⟨*color*⟩ of the descriptive title text deviating from /tcb/coltitle → P. 27. The color is reset to /tcb/coltitle → P. 27, if description color is used without value.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   description color=red!25!yellow}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.9: My example

My theorem text.

/tcb/description font=⟨*text*⟩ (default empty, initially empty)

Sets ⟨*text*⟩ (e. g. font settings) before the descriptive title text deviating from /tcb/fonttitle → P. 28. The ⟨*text*⟩ is removed, if `description font` is used without value.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   description delimiters={\glqq}{\grqq},
   description font=\mdseries\itshape}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.10: *„My example“*

My theorem text.

/tcb/description formatter=⟨*macro*⟩ (default empty, initially empty)

Sets ⟨*macro*⟩ as formatter for the descriptive title text. The ⟨*macro*⟩ has to take one mandatory argument (the description text).

Note that /tcb/description delimiters → P. 286, /tcb/description color → P. 286, and /tcb/description font are ignored, if this option is used.

If `description formatter` is used without value, the formatter is reset to its standard behavior.

```
\newtcbox{\formbox}{enhanced,frame empty,size=minimal,boxsep=2pt,arc=1pt,
  on line,interior style image=goldshade.png}

\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   description formatter=\formbox}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.11: **My example**

My theorem text.

/tcb/terminator sign=⟨*sign*⟩ (no default, initially empty)

The given ⟨*sign*⟩ is used as terminator at the end of the title text of a theorem.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   terminator sign={.}}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.12: My example.

My theorem text.

/tcb/terminator sign colon (style, no value, initially set)

Sets /tcb/terminator sign → P. 287 to the colon : sign.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   separator sign dash,terminator sign colon}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.13 – My example:

My theorem text.

/tcb/terminator sign dash (style, no value)

Sets /tcb/terminator sign → P. 287 to an en-dash sign.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   terminator sign dash}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.14: My example –

My theorem text.

/tcb/terminator sign none (style, no value)

Sets /tcb/terminator sign → P. 287 to the default empty text.

/tcb/theorem name and number (style, no value, initially set)

Prints theorem name followed by theorem number inside the title.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   theorem name and number}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem 14.15: My example

My theorem text.

/tcb/theorem number and name (style, no value)

Prints theorem number followed by theorem name inside the title.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   theorem number and name}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

14.16 Theorem: My example

My theorem text.

/tcb/theorem name (style, no value)

Prints theorem name without number inside the title.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   theorem name,enhanced,watermark text={\thetcbcounter}}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

Theorem: My example

My theorem text. 14.17

/tcb/theorem={⟨*display name*⟩}{⟨*counter*⟩}{⟨*title*⟩}{⟨*marker*⟩} (no default)

This key is internally used by `\tcbmaketheorem` → P. 282, but can be used directly in a `tcolorbox` for a more flexible approach. The ⟨*display name*⟩ is used together with the increased ⟨*counter*⟩ value and the ⟨*title*⟩ for the title line of the box. Additionally, a `\label` with the given ⟨*marker*⟩ is created.

```
% \newcounter{texercise}%  preamble
\begin{tcolorbox}[colback=green!10,colframe=green!50!black,arc=4mm,
                  theorem={Test}{texercise}{Direct usage}{myMarker}]
Here, we see the test \ref{myMarker}.
\end{tcolorbox}
```

Test 1: Direct usage

Here, we see the test 1.

For a common appearance inside the document, the key `theorem` should not be used directly as in the example above, but as part of a new environment created by hand or using `\tcbmaketheorem` → P. 282 or using its successor `\newtcbtheorem` → P. 281.

/tcb/highlight math (style, no value)

Predefined style which is used for \tcbhighmath → P. 283. It can be changed comfortable with /tcb/highlight math style.

/tcb/highlight math style=⟨*style definition*⟩ (style, no default)

Changes the definition for /tcb/highlight math to the given ⟨*style definition*⟩. See \tcbhighmath → P. 283 for another example.

```
% \tcbuselibrary{skins}
\tcbset{highlight math style={enhanced,%<-- needed for the 'remember' options
  colframe=red,colback=red!10!white,boxsep=0pt}}
\begin{align*}
\tcbhighmath[remember as=fx]{f(x)}
     &= \int\limits_{1}^{x} \frac{1}{t^2}~dt
      = \left[ -\frac{1}{t} \right]_{1}^{x}\\
     &= -\frac{1}{x} + \frac{1}{1}\\
     &=
\tcbhighmath[remember,overlay={%
    \draw[blue,very thick,->] (fx.south) to[bend right] ([yshift=2mm]frame.west);}]
  {1-\frac{1}{x}.}
\end{align*}
```

$$\boxed{f(x)} = \int\limits_{1}^{x} \frac{1}{t^2}\,dt = \left[-\frac{1}{t}\right]_{1}^{x}$$
$$= -\frac{1}{x} + \frac{1}{1}$$
$$= \boxed{1 - \frac{1}{x}.}$$

/tcb/math upper (style, no value)

Sets the upper part to mathematical mode with font \displaystyle.

/tcb/math lower (style, no value)

Sets the lower part to mathematical mode with font \displaystyle.

/tcb/math (style, no value)

Sets the upper part *and* lower part to mathematical mode with font \displaystyle.

```
\begin{tcolorbox}[math,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.
\end{tcolorbox}
```

$$\sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.$$

The following styles are only tested to work with the original `amsmath` environments. If e.g. the `equation` environment is redefined as `gather`, then `/tcb/ams equation` should / could not be used. Obviously, you are encouraged to use `/tcb/ams gather` → P. 293 in this case.

U 2014-10-30 **`/tcb/ams equation upper`** (style, no value)

Adds an `amsmath` `equation` environment to the start and end of the upper part.

U 2014-10-30 **`/tcb/ams equation lower`** (style, no value)

Adds an `amsmath` `equation` environment to the start and end of the lower part.

U 2014-10-30 **`/tcb/ams equation`** (style, no value)

Adds an `amsmath` `equation` environment to the start and end of the upper *and* lower part.

```
\begin{tcolorbox}[ams equation,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.
\end{tcolorbox}
```

$$\sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{15}$$

U 2014-10-30 **`/tcb/ams equation* upper`** (style, no value)

Adds an `amsmath` `equation*` environment to the start and end of the upper part.

U 2014-10-30 **`/tcb/ams equation* lower`** (style, no value)

Adds an `amsmath` `equation*` environment to the start and end of the lower part.

U 2014-10-30 **`/tcb/ams equation*`** (style, no value)

Adds an `amsmath` `equation*` environment to the start and end of the upper *and* lower part.

```
\begin{tcolorbox}[ams equation*,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.
\end{tcolorbox}
```

$$\sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.$$

/tcb/ams align upper (style, no value)

Adds an `amsmath` `align` environment to the start and end of the upper part.

/tcb/ams align lower (style, no value)

Adds an `amsmath` `align` environment to the start and end of the lower part.

/tcb/ams align (style, no value)

Adds an `amsmath` `align` environment to the start and end of the upper *and* lower part.

```
\begin{tcolorbox}[ams align,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} &= \infty.\\
  \int x^2 ~\text{d}x                    &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{16}$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c. \tag{17}$$

/tcb/ams align* upper (style, no value)

Adds an `amsmath` `align*` environment to the start and end of the upper part.

/tcb/ams align* lower (style, no value)

Adds an `amsmath` `align*` environment to the start and end of the lower part.

/tcb/ams align* (style, no value)

Adds an `amsmath` `align*` environment to the start and end of the upper *and* lower part.

```
\begin{tcolorbox}[ams align*,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} &= \infty.\\
  \int x^2 ~\text{d}x                    &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c.$$

/tcb/ams gather upper (style, no value)

Adds an `amsmath` `gather` environment to the start and end of the upper part.

/tcb/ams gather lower (style, no value)

Adds an `amsmath` `gather` environment to the start and end of the lower part.

/tcb/ams gather (style, no value)

Adds an `amsmath` `gather` environment to the start and end of the upper *and* lower part.

```
\begin{tcolorbox}[ams gather,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{18}$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c. \tag{19}$$

/tcb/ams gather* upper (style, no value)

Adds an `amsmath` `gather*` environment to the start and end of the upper part.

/tcb/ams gather* lower (style, no value)

Adds an `amsmath` `gather*` environment to the start and end of the lower part.

/tcb/ams gather* (style, no value)

Adds an `amsmath` `gather*` environment to the start and end of the upper *and* lower part.

```
\begin{tcolorbox}[ams gather*,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c.$$

/tcb/ams nodisplayskip upper (style, no value)

Neutralizes the `\abovedisplayskip` of a following `align` or `gather` environment for the upper part. Note that the text content has to start with such a formula.

/tcb/ams nodisplayskip lower (style, no value)

Neutralizes the `\abovedisplayskip` of a following `align` or `gather` environment for the lower part. Note that the text content has to start with such a formula.

/tcb/ams nodisplayskip (style, no value)

Neutralizes the `\abovedisplayskip` of a following `align` or `gather` environment for the upper part *and* lower part. Note that the text content has to start with such a formula.

```
\begin{tcolorbox}[ams nodisplayskip,colback=yellow!10!white,colframe=red!50!black]
  \begin{gather}
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
  \end{gather}
  And now for something completely different.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{20}$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c. \tag{21}$$

And now for something completely different.

New colored mathematical environments are easily created using `\newtcolorbox` → P. 14:

```
\newtcolorbox{mymath}{ams gather*,colback=yellow!10!white,colframe=red!50!black}

\begin{mymath}
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
\end{mymath}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c.$$

All described options like `/tcb/ams gather upper` → P. 293, `/tcb/ams gather lower` → P. 293, `/tcb/ams gather` → P. 293 are (partially) setting (overwritting) the keys `/tcb/before upper` → P. 58, `/tcb/after upper` → P. 59, `/tcb/before lower` → P. 59, `/tcb/after lower` → P. 59.

Therefore, e. g. `\tcbset{ams gather,before upper={\text{Pythagoras:}}}` produces an invalid result. For this case, you are invited to use
`\tcbset{ams gather,before upper app={\text{Pythagoras:}}}`,
see `/tcb/before upper app` → P. 346.

`/tcb/theorem style=`⟨*name*⟩ (no default, initially `standard`)

Applies a predefined style ⟨*name*⟩ to the theorem environment. Some of the feasible ⟨*name*⟩ values resemble style names from the packages `theorem` and `ntheorem` to give convenient access to known patterns.

The styles alter `/tcb/separator sign` → P. 285, `/tcb/description delimiters` → P. 286, `/tcb/terminator sign` → P. 287, and more. Therefore, one should apply such keys *after* a theorem style.

For the following examples, we use:

Definition in the preamble:

```
\newtcbtheorem[use counter from=mytheo]{theorem}{Theorem}{%
  fonttitle=\bfseries\upshape,fontupper=\itshape,
  colframe=green!50!black,colback=green!10!white,
  colbacktitle=green!20!white,coltitle=blue!75!black}{theo}
```

The predefined styles are:

- `standard`: This is the initial value.

```
\begin{theorem}[theorem style=standard]{standard}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

Theorem 14.18: standard

This is my theorem.

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

- `change standard`

```
\begin{theorem}[theorem style=change standard]{change standard}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.19 Theorem: change standard

This is my theorem.

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

- `plain`

```
\begin{theorem}[theorem style=plain]{plain}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

Theorem 14.20 (plain): *This is my theorem.*

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

- **break**

```
\begin{theorem}[theorem style=break]{break}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

Theorem 14.21 (break):
This is my theorem.

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

- **plain apart**

```
\begin{theorem}[theorem style=plain apart]{plain apart}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

Theorem 14.22 (plain apart)

This is my theorem.

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

- **change**

```
\begin{theorem}[theorem style=change]{change}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.23 Theorem (change): *This is my theorem.*

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

- **change break**

```
\begin{theorem}[theorem style=change break]{change break}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.24 Theorem (change break):
This is my theorem.

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

- **change apart**

```
\begin{theorem}[theorem style=change apart]{change apart}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.25 Theorem (change apart)

This is my theorem.

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

- **margin**

```
\begin{theorem}[theorem style=margin,left=10mm]{margin}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
\begin{theorem}[theorem style=margin,left=10mm,oversize]{margin}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.26 Theorem (margin): *This is my theorem.*

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

14.27 Theorem (margin): *This is my theorem.*

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

- **margin break**

```
\begin{theorem}[theorem style=margin break,left=10mm]{margin break}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
\begin{theorem}[theorem style=margin break,left=10mm,oversize]{margin break}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.28 Theorem (margin break):
This is my theorem.

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

14.29 Theorem (margin break):
This is my theorem.

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

- **margin apart**

```
\begin{theorem}[theorem style=margin apart,left=10mm]{margin apart}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
\begin{theorem}[theorem style=margin apart,left=10mm,oversize]{margin apart}{}
This is my theorem. \begin{equation*} a^2 + b^2 = c^2. \end{equation*}
\end{theorem}
```

14.30 Theorem (margin apart)

This is my theorem.

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

14.31 Theorem (margin apart)

This is my theorem.

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

14.3 Examples for Definitions and Theorems

In the following, the application of `\tcbmaketheorem` → P. 282 to highlight mathematical definitions, theorems, or the like is demonstrated.

At first, additional `tcb` keys are created for the appearance of the colored boxes. It is assumed that theorems and corollaries should be identically colored. All following environments are numbered with a common counter, but this can be changed easily. Here, the counter output is supplemented by the subsection number. Further, the `cleveref` package [5] is used for clever references.

Definition in the preamble:

```
% \usepackage{cleveref}
\tcbset{
  defstyle/.style={fonttitle=\bfseries\upshape, fontupper=\slshape,
              arc=0mm, colback=blue!5!white,colframe=blue!75!black},
  theostyle/.style={fonttitle=\bfseries\upshape, fontupper=\slshape,
               colback=red!10!white,colframe=red!75!black},
}
\newtcbtheorem[number within=subsection,crefname={definition}{definitions}]%
  {Definition}{Definition}{defstyle}{def}
\newtcbtheorem[use counter from=Definition,crefname={theorem}{theorems}]%
  {Theorem}{Theorem}{theostyle}{theo}
\newtcbtheorem[use counter from=Definition,crefname={corollary}{corollaries}]%
  {Corollary}{Corollary}{theostyle}{cor}
```

By `\newtcbtheorem` → P. 281, commonly numbered theorem environments are created now. `defstyle` and `theostyle` are used for the appearance.

Now, everything is prepared for the following examples.

```
The following theorem is numbered as \Cref{theo:diffbarstetig} and
referenced with the marker \texttt{theo:diffbarstetig}.\bigskip

\begin{Theorem}{Differenzierbarkeit bedingt Stetigkeit, wobei diese Benennung
  zu Testzwecken ungew\"{o}hnlich lang ist}{diffbarstetig}%
  Eine Funktion $f:I\to\mathbb{R}$ ist in $x_0\in I$ stetig, wenn $f$ in
  $x_0$ differenzierbar ist.
\end{Theorem}
```

The following theorem is numbered as Theorem 14.3.1 and referenced with the marker `theo:diffbarstetig`.

Theorem 14.3.1: Differenzierbarkeit bedingt Stetigkeit, wobei diese Benennung zu Testzwecken ungewöhnlich lang ist

Eine Funktion $f : I \to \mathbb{R}$ ist in $x_0 \in I$ stetig, wenn f in x_0 differenzierbar ist.

```
The following definition is numbered as \Cref{def:diffbarkeit} and
referenced with the marker \texttt{def:diffbarkeit}.\bigskip

\begin{Definition}{Differenzierbarkeit}{diffbarkeit}
  Eine Funktion $f:~I\to\mathbb{R}$ auf einem Intervall $I$ hei\ss{}t in
  $x_0\in I$ differenzierbar oder linear approximierbar,
  wenn der Grenzwert
  \begin{equation*}
  \lim\limits_{x\to x_0}\frac{f(x)-f(x_0)}{x-x_0}=
  \lim\limits_{h\to 0}\frac{f(x_0+h)-f(x_0)}{h}
  \end{equation*}
  existiert. Bei Existenz hei\ss{}t dieser Grenzwert Ableitung
  oder Differentialquotient von $f$ in $x_0$ und man
  schreibt f\"{u}r ihn
  \begin{equation*}
  f'(x_0)\quad\text{oder}\quad\frac{df}{dx}(x_0).
  \end{equation*}
\end{Definition}
```

The following definition is numbered as Definition 14.3.2 and referenced with the marker `def:diffbarkeit`.

Definition 14.3.2: Differenzierbarkeit

Eine Funktion $f : I \to \mathbb{R}$ auf einem Intervall I heißt in $x_0 \in I$ differenzierbar oder linear approximierbar, wenn der Grenzwert

$$\lim_{x\to x_0}\frac{f(x)-f(x_0)}{x-x_0}=\lim_{h\to 0}\frac{f(x_0+h)-f(x_0)}{h}$$

existiert. Bei Existenz heißt dieser Grenzwert Ableitung oder Differentialquotient von f in x_0 und man schreibt für ihn

$$f'(x_0)\quad\text{oder}\quad\frac{df}{dx}(x_0).$$

```
The following corollary is numbered as \Cref{cor:nullstellen} and
referenced with the marker \texttt{cor:nullstellen}.\bigskip

\begin{Corollary}{Nullstellenexistenz}{nullstellen}
  Ist $f:[a,b]\to\mathbb{R}$ stetig und haben $f(a)$ und $f(b)$ entgegengesetzte
  Vorzeichen, also $f(a)f(b)<0$, so besitzt $f$ eine Nullstelle $x_0\in]a,b[$,
  also $f(x_0)=0$.
\end{Corollary}
```

The following corollary is numbered as Corollary 14.3.3 and referenced with the marker `cor:nullstellen`.

Corollary 14.3.3: Nullstellenexistenz

Ist $f : [a, b] \to \mathbb{R}$ stetig und haben $f(a)$ und $f(b)$ entgegengesetzte Vorzeichen, also $f(a)f(b) < 0$, so besitzt f eine Nullstelle $x_0 \in]a, b[$, also $f(x_0) = 0$.

```
\begin{Theorem}[boxrule=2mm,toptitle=-1.5mm,bottomtitle=-1.5mm]{%
    Hinreichende Bedingung f\"{u}r Wendepunkte}{wendehinreichend}%
  $f$ sei eine auf einem Intervall $]a,b[$ dreimal stetig differenzierbare Funktion.
  Ist $f''(x_0)=0$ in $x_0\in]a,b[$ und $f'''(x_0)\ne 0$, so ist
  $(x_0,f(x_0))$ ein Wendepunkt von $f$.
\end{Theorem}
```

Theorem 14.3.4: Hinreichende Bedingung für Wendepunkte

f sei eine auf einem Intervall $]a,b[$ dreimal stetig differenzierbare Funktion. Ist $f''(x_0) = 0$ in $x_0 \in]a,b[$ und $f'''(x_0) \neq 0$, so ist $(x_0, f(x_0))$ ein Wendepunkt von f.

```
% \usepackage{varioref}
% \usepackage{cleveref}
% \tcbuselibrary{skins}
\newtcbtheorem[use counter from=Definition]{YetAnotherTheorem}{Theorem}%
  {theorem style=plain apart,label type=theorem,enhanced,frame hidden,
  boxrule=2mm,titlerule=0mm,toptitle=1mm,bottomtitle=1mm,
  fonttitle=\bfseries\large,fontupper=\normalsize,
  coltitle=green!35!black,colbacktitle=green!15!white,
  colback=green!50!yellow!15!white,borderline={1pt}{0pt}{green!25!blue},
  }{theo}

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{meanvaluetheorem}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}

\medskip
Here, |cleveref| support is used to reference \Cref{theo:meanvaluetheorem}
on \Cpageref{theo:meanvaluetheorem}. This \namecref{theo:meanvaluetheorem}
can also be referenced by |\Vref| resulting in \Vref{theo:meanvaluetheorem}.
```

Theorem 14.3.5 (Mittelwertsatz für n Variable)

Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^{\top}(x - x_0)$$

Here, `cleveref` support is used to reference Theorem 14.3.5 on Page 300. This theorem can also be referenced by `\Vref` resulting in Theorem 14.3.5.

Note that `/tcb/label type` → P. 93 was used in the example above to feed `cleveref` [5] with the needed name information.

```
Here, using |\Vref| resulting in \Vref{theo:meanvaluetheorem} is more interesting\ldots
```

Here, using \Vref resulting in Theorem 14.3.5 on page 300 is more interesting...

```
% \tcbuselibrary{skins}
\newtcbtheorem[use counter from=Definition]{YetAnotherTheorem}{Theorem}%
  {theorem style=change apart,enhanced,arc=0mm,outer arc=0mm,
  boxrule=0mm,toprule=1mm,bottomrule=1mm,left=1mm,right=1mm,
  titlerule=0mm,toptitle=0mm,bottomtitle=1mm,top=0mm,
  colframe=red!50!black,colback=red!5!white,coltitle=red!50!black,
  title style={top color=yellow!50!white,bottom color=red!5!white,
    middle color=yellow!50!white},
  fonttitle=\bfseries\sffamily\normalsize,fontupper=\normalsize\itshape,
  }{theo}

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{mittelwertsatz_n2}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}
```

14.3.6 Theorem (Mittelwertsatz für n Variable)

Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^\top (x - x_0)$$

```
% \usepackage{varwidth}   \tcbuselibrary{skins}
\newtcbtheorem[use counter from=Definition]{YetAnotherTheorem}{Theorem}%
{enhanced,frame empty,interior empty,colframe=ForestGreen!50!white,
  coltitle=ForestGreen!50!black,fonttitle=\bfseries,colbacktitle=ForestGreen!15!white,
  borderline={0.5mm}{0mm}{ForestGreen!15!white},
  borderline={0.5mm}{0mm}{ForestGreen!50!white,dashed},
  attach boxed title to top center={yshift=-2mm},
  boxed title style={boxrule=0.4pt},varwidth boxed title}{theo}

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{mittelwertsatz_n3}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}
```

Theorem 14.3.7: Mittelwertsatz für n Variable

Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^\top (x - x_0)$$

You need more attention for your theorems? Here, you are ...

```
% tcbuselibrary{skins}  % preamble
\begin{Theorem}[enhanced,
    fuzzy halo=3mm with yellow,
    fuzzy halo=2mm with red,
    fuzzy halo=1mm with yellow,
    watermark color=red!35!white,
    watermark text={Overacting\\Fundamental Theorem}]%
  {Fundamental Theorem of Theorems}{fundamental}%
  \lipsum[1-2]
\end{Theorem}
```

Theorem 14.3.8: Fundamental Theorem of Theorems

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Overacting
Fundamental Theorem

Let's try a more conservative approach:

```
% \tcbuselibrary{skins}
\newtcbtheorem[use counter from=Definition]{YetAnotherTheorem}{Theorem}%
  {theorem style=plain,enhanced,colframe=blue!50!black,colback=yellow!20!white,
   coltitle=red!50!black,fonttitle=\upshape\bfseries,fontupper=\itshape,
   drop fuzzy shadow=blue!50!black!50!white,boxrule=0.4pt}{theo}

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{mittelwertsatz_n4}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}
```

Theorem 14.3.9 (Mittelwertsatz für n Variable): *Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt*

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^\top (x - x_0)$$

14.4 Using other theorem environments with `tcolorbox`

Instead of creating theorem environments with the methods described before, environments from other packages can be boxed with a `tcolorbox`.

Environments may be created e.g. by methods from the `theorem` package or the `amsthm` package. `\tcolorboxenvironment` → P. 16 can be used to put a box around these environments.

Definition in the preamble:

```
\usepackage{amsthm}

\theoremstyle{plain}%    from 'amsthm'
\newtheorem{lem}{Lemma}% from 'amsthm'

\tcolorboxenvironment{lem}{
  enhanced jigsaw,colframe=cyan,interior hidden,
  breakable,before skip=10pt,after skip=10pt }

\tcolorboxenvironment{proof}{% 'proof' from 'amsthm'
  blanker,breakable,left=5mm,
  before skip=10pt,after skip=10pt,
  borderline west={1mm}{0pt}{red}}
```

```
\begin{lem}
  \lipsum[2]
\end{lem}

\lipsum[3]

\begin{proof}
  \lipsum*[4]
\end{proof}
```

Lemma 1. *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.*

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Proof. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. □

15 Library breakable

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{breakable}
```

15.1 Technical Overview

The library breakable supports the automatic breaking of a `tcolorbox`. This feature is enabled by `/tcb/breakable` → P. 306 and disabled by `/tcb/unbreakable` → P. 307.

If a `tcolorbox` is set to be `/tcb/breakable` → P. 306, then the following algorithm is executed:

1. The box content is read to a box register similar but not identical to the unbreakable case.
2. If the total box fits into the current page, it is shipped out visibly unbroken and the algorithm stops.

   Unbroken Box

   The box. unbroken

3. Otherwise, it is checked if at least `/tcb/lines before break` → P. 307 of the upper box can be placed on the current page. If not, a page break is inserted and the algorithm goes back to Step 2.
4. Now, the *break sequence* starts. The upper box part or the lower box part is split such that it fits into the current page. The fitting part is named *first part* of the *break sequence* and shipped out.

   Broken Box

   The box. first

5. If the remaining content of the total box fits into the current page, the algorithm continues with Step 7, else with Step 6.
6. The upper box part or the lower box part is split such that it fits into the current page. The fitting part is named *middle part* of the *break sequence* and shipped out. Then, the algorithm goes back to Step 5.

   The box. middle

7. The remaining part is named *last part* of the *break sequence* and shipped out. The algorithm stops.

   The box. last

The algorithm takes care that the optional segmentation line never appears at the end of a box. The optional lower box part is also checked to have at least `/tcb/lines before break` → P. 307.

In principle, all boxes of the *break sequence* share the same geometric parameters. The differences are:

- The given `/tcb/before`→P. 76 and `/tcb/after`→P. 76 values are used only before the *first* and after the *last* part of the *break sequence*.
- A special behavior between the parts of the *break sequence* can be given by `/tcb/toprule at break`→P. 310, `/tcb/bottomrule at break`→P. 310, `/tcb/enlarge top at break by`→P. 81, and `/tcb/enlarge bottom at break by`→P. 81.
- The `/tcb/skin`→P. 112 decides *how* the *first*, *middle*, and *last* part look like. Actually, every part type has its own skin given by the options `/tcb/skin first`→P. 112, `/tcb/skin middle`→P. 112, and `/tcb/skin last`→P. 112. Typically, these options are set automatically by the main skin, see Subsection 15.7 from page 317.

15.2 Limitations and Known Bugs

- The maximal total height of the upper and of the lower part of normal breakable `tcolorbox`es is about 65536pt (ca. 2300cm) apiece. If such a part gets longer, the output will get buggy without warning. For very oversized boxes which are longer than 65536pt, use the **`unlimited`** value for `/tcb/breakable`→P. 306. With the **`unlimited`** setting, the applied algorithm has (virtually) no height limit for boxes, but very likely the compiler memory will have to be increased for boxes longer than 300 pages (depending on compiler settings and box content). But it is recommended to use **`unlimited`** for critical large boxes only.
- You can nest an unbreakable `tcolorbox` inside another `tcolorbox`, even inside a breakable one. But you cannot not nest a breakable box inside a breakable box. The `/tcb/breakable`→P. 306 key for a nested box is ignored automatically[4], i. e. inner boxes are always unbreakable.

  After all, in the unlikely case you really want to have the nested box to be breakable, use `/tcb/enforce breakable`→P. 307 for the nested box[5]. **But, a breakable box inside a breakable box will usually give a mess.**
- If your text content contains some text color changing commands, your color will not survive the break to the next box. But, with the `fontspec` package and `xelatex` or `lualatex`, you can use `\addfontfeatures{Color=mycolor}` to add a font color which survives the break.

N 2014-10-30

- The `perpage` option of the `footmisc` package is deliberately deactivated inside a breakable box since all footnotes are placed at the end of the box (possibly far away from the reference point).

[4] Until `tcolorbox` 3.04, the `/tcb/breakable`→P. 306 key was not ignored for nested boxes.
[5] `/tcb/enforce breakable`→P. 307 acts like `/tcb/breakable`→P. 306 until `tcolorbox` 3.04.

15.3 Main Option Keys

`/tcb/breakable=true|false|unlimited` (default `true`, initially `false`)

Allows the `tcolorbox` to be breakable. If the box is larger than the available space at the current page, the box is automatically broken and continued to the next next page. All sorts of `tcolorbox` can be made breakable. It depends on the skin how the breaking looks like. If you do not know better, use `/tcb/enhanced` → P. 178 for breaking a box. The parts of the *break sequence* are numbered by the counter `tcbbreakpart`.

- `false`: Sets the `tcolorbox` to be unbreakable.
- `true`: Breaks the `tcolorbox` from one page to another. The maximal total height of the upper and of the lower part is about 65536pt (ca. 2300cm or ca. 90 pages) apiece.
- `unlimited`: Experimental code for unlimited total height of breakable boxes. For boxes longer than 300 pages (or even shorter ones) the compiler memory will have to be increased.

```
% \usepackage{lipsum}  % preamble
\tcbset{enhanced jigsaw,colback=red!5!white,colframe=red!75!black,
  watermark color=yellow!25!white,watermark text=\arabic{tcbbreakpart},
  fonttitle=\bfseries}

\begin{tcolorbox}[breakable,title=My breakable box]
\lipsum[1-6]
\end{tcolorbox}
```

My breakable box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea

dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.
Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.
Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

/tcb/unbreakable (no value, initially set)

Sets the `tcolorbox` to be unbreakable.

/tcb/enforce breakable (no value)

A `tcolorbox` inside a `tcolorbox` is automatically set to be unbreakable. Using `/tcb/breakable` → P. 306 on such an inner box has no effect. If one *really* wants the inner box to be breakable, use `/tcb/enforce breakable`. **This will usually give a mess of shattered boxes. You are advised to not use this option.**
Note that `/tcb/enforce breakable` has the functionality that `/tcb/breakable` → P. 306 had until package version 3.04 and exists for backward compatibility.

/tcb/title after break=⟨*text*⟩ (no default, initially empty)

The `/tcb/title` → P. 17 is used only for the *first* part of a *break sequence*. Use `title after break` to create a heading line with ⟨*text*⟩ as content for all following parts.

/tcb/notitle after break (no value, initially set)

Removes the title line or following parts in a *break sequence* if set before.

/tcb/adjusted title after break=⟨*text*⟩ (style, no default, initially unset)

Works like `/tcb/adjusted title` → P. 17 but applied to `/tcb/title after break`.

/tcb/lines before break=⟨*number*⟩ (no default, initially 2)

Assures that the given ⟨*number*⟩ of lines of the upper box part or the lower box part are placed before a break happens.

/tcb/break at=⟨*length*⟩/⟨*length*⟩/.../⟨*length*⟩ (no default, initially `0pt`)

Defines break points at the given ⟨*length*⟩ values. The first ⟨*length*⟩ defines the (maximal) height of the first partial box, the second ⟨*length*⟩ defines the (maximal) height of the second partial box, and so on. The last ⟨*length*⟩ value is applied to all following partial boxes if any. Setting a length to `0pt` means that the naturally available space is used for breaking.

```
% \usepackage{multicol,lipsum}
\begin{multicols}{3}\footnotesize
Breakable boxes inside a |multicols| environment need special attendance.
They are broken by default at |\textheight|.
The |break at| option can be used to insert better break points by hand.
\begin{tcolorbox}[enhanced jigsaw,size=small,vfill before first,
  colframe=red,colback=yellow!10!white,before title=\raggedright,
  title={Broken box inside a |multicols| environment},fonttitle=\bfseries,
  enforce breakable,% use only breakable in the real world!
  pad at break=1mm,break at=3cm/6.3cm ]
\lipsum[1]
\end{tcolorbox}
\refKey{/tcb/height fixed for} may also be considered for |multicols| environments.
\end{multicols}
```

Breakable boxes inside a `multicols` environment need special attendance. They are broken by default at `\textheight`. The `break at` option can be used to insert better break points by hand.

Broken box inside a `multicols` environment

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

`/tcb/height fixed for` → P. 311 may also be considered for `multicols` environments.

/tcb/enlargepage=⟨*length*⟩/⟨*length*⟩/.../⟨*length*⟩ (no default, initially `0pt`)

Inserts a `\enlargethispage{`⟨*length*⟩`}` to the pages of the break sequence, i. e. allows one to enlarge (or shrink) partial boxes. The first ⟨*length*⟩ is applied to the first partial box, the second ⟨*length*⟩ is applied to the second partial box, and so on. The last ⟨*length*⟩ value is applied to all following partial boxes if any. Note that floating boxes will not be enlarged.

```
\begin{tcolorbox}[breakable,enlargepage=0mm/\baselineskip/2\baselineskip/0mm,...
```

The example code enlarged the second partial box by one line, the third partial box by two lines, and all following parts are not enlarged.

If an automated page break occures before the first partial box, the page enlargement is applied to the page before the first partial box *and* again to the page of the first partial box. Insert a manual break to prevent this.

In general, `enlargepage` should be used at the final stage of a document for fine-tuning only.

/tcb/enlargepage flexible=⟨*length*⟩ (no default, initially `0pt`)

This allows an automated page enlargement for up to ⟨*length*⟩. The algorithm can use this to avoid breaking a box, if there is anough room after enlargement. Also, the *last* partial box of a break sequence may be enlarged to avoid further breaking.

Note that this potential enlargement is *additive* to settings of `/tcb/enlargepage` → P. 308. But `/tcb/enlargepage flexible` overwrites settings of `/tcb/pad before break*` → P. 310 or `/tcb/pad at break*` → P. 310.

```
% The following setting hinders orphan lines for the last partial box
\tcbset{enlargepage flexible=\baselineskip}
```

N 2014-12-15 **/tcb/compress page=**⟨*option*⟩ (default `all`, initially `baselineskip`)

This option controls the space management on the page which contains the unbroken box or the first part of a *break sequence*. Feasible ⟨*option*⟩ values are:

- `all` (default value): All shrinkable glue on the page is potentially used for the unbroken box or the first part of a *break sequence*. Thus, all vertical spaces on the page will potentially be reduced to their minimal values.
- `baselineskip` (initial value): Shrinkable glue up to one `\baselineskip` on the page is potentially used for the unbroken box or the first part of a *break sequence*.
- `none`: The break algorithm respects the target size of the given glue values on the page. This was the inital value before version `3.34`.

Note that the box *content* is not influenced by this option.

/tcb/shrink break goal=⟨*length*⟩ (no default, initially `0pt`)

This is an emergency parameter if the break algorithm produces unpleasant breaks. It shrinks the goal height of the current box part by ⟨*length*⟩ which may result in smaller boxes. Never use negative values. *Usually, this option will never be needed at all.*

15.4 Option Keys for the Break Appearance

/tcb/toprule at break=⟨*length*⟩ (no default, initially 0.5mm)

Sets the line width of the top rule to ⟨*length*⟩ *if* the box is /tcb/breakable → P. 306. In this case, it is applied to *middle* and *last* parts in a break sequence. Note that /tcb/toprule → P. 34 overwrites this value if used afterwards.

/tcb/bottomrule at break=⟨*length*⟩ (no default, initially 0.5mm)

Sets the line width of the bottom rule to ⟨*length*⟩ *if* the box is /tcb/breakable → P. 306. In this case, it is applied to *first* and *middle* parts in a break sequence. Note that /tcb/bottomrule → P. 34 overwrites this value if used afterwards.

/tcb/topsep at break=⟨*length*⟩ (no default, initially 0mm)

Additional vertical space of ⟨*length*⟩ which is added at the top of *middle* and *last* parts in a break sequence. In general, it is not advisable to change this value if these parts start with a rule or a title.

/tcb/bottomsep at break=⟨*length*⟩ (no default, initially 0mm)

Additional vertical space of ⟨*length*⟩ which is added at the bottom of *first* and *middle* parts in a break sequence. In general, it is not advisable to change this value if these parts end with a rule.

/tcb/pad before break=⟨*length*⟩ (style, no default, initially 3.5mm)

Sets the total amount of vertical space after the text content and before the break point to ⟨*length*⟩. This style sets /tcb/toprule at break to 0pt and changes /tcb/topsep at break as required. In general, it is not advisable to change this value if the *middle* and *last* parts in a break sequence start with a rule or a title.

/tcb/pad before break*=⟨*length*⟩ (style, no default)

Sets /tcb/pad before break to ⟨*length*⟩ and /tcb/enlargepage flexible → P. 309 to an appropriate value such that empty closing frames are avoided.

/tcb/pad after break=⟨*length*⟩ (style, no default, initially 3.5mm)

Sets the total amount of vertical space after the break point and before the text content to ⟨*length*⟩. This style sets /tcb/bottomrule at break to 0pt and changes /tcb/bottomsep at break as required. In general, it is not advisable to change this value if the *first* and *middle* parts in a break sequence end with a rule.

/tcb/pad at break=⟨*length*⟩ (style, no default, initially 3.5mm)

Abbreviation for setting ⟨*length*⟩ to /tcb/pad before break and /tcb/pad after break.

/tcb/pad at break*=⟨*length*⟩ (style, no default)

Sets /tcb/pad at break to ⟨*length*⟩ and /tcb/enlargepage flexible → P. 309 to an appropriate value such that empty closing frames are avoided.

```
% \usepackage{lipsum}  % preamble
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced jigsaw,breakable,pad at break*=0mm,
  title={For this box, the pad space at the break point is set to 0mm}]
  \lipsum[1-2]
\end{tcolorbox}
```

For this box, the pad space at the break point is set to 0mm

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.
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.

`/tcb/pad at break`→P. 310 or `/tcb/pad at break*`→P. 310 should be used as very last option in an option list, because they adapt other settings.

Also see `/tcb/enlarge top at break by`→P. 81 and `/tcb/enlarge bottom at break by`→P. 81.

`/tcb/height fixed for=`⟨*part*⟩ (no default, initially `none`)

When certain amount of space is available for a partial box of a break sequence, the partial box typically is smaller than this space (depending on the box content). For given ⟨*part*⟩(s), the height can be set to all available space.

- `none`: Every partial `tcolorbox` is set with its natural height.
- `first`: The *first* partial box is set to a height which matches the available space.
- `middle`: All *middle* partial boxes are set to a height which matches the available space.
- `last`: The *last* partial box is set to a height which matches the available space.
- `first and middle`: The *first* and all *middle* partial boxes are set to a height which matches the available space.
- `middle and last`: All *middle* partial boxes and the *last* partial box are set to a height which matches the available space.
- `all`: All partial boxes are set to a height which matches the available space.

If the box keeps unbroken, this option is not applied. See `/tcb/height`→P. 51 for setting a fixed height for unbroken boxes. See `/tcb/height fill`→P. 53 for giving unbroken boxes maximum height.

`/tcb/vfill before first=true|false` (default `true`, initially `false`)

Inserts a `\vfill` at the begin of the *first* partial box to move this partial box to the end of the current page. This may be used as an alternative to `/tcb/height fixed for=first` to get justified columns or pages. The `\vfill` is not inserted, if the box gets not actually broken.

15.5 Extra Options for Partial Boxes

N 2015-07-16 **/tcb/extras**={⟨*options*⟩} (no default, initially unset)

Adds `tcolorbox` ⟨*options*⟩ to every box of a *break sequence* after skin settings are done. This is quite late in box processing. Geometry and break settings should *not be used* here, because they will either be ignored or have unexpected negative results. But it is possible to change most colors, skin effects, shadows, borders, frame code, etc. Note that using `/tcb/extras` for every box is very seldom an advantage over setting the options directly. Usually, `/tcb/extras first`, `/tcb/extras middle`, etc. are sensible to apply.

N 2015-07-16 **/tcb/no extras** (style, no default, initially set)

Removes all extras if set before.

N 2015-07-16 **/tcb/extras broken**={⟨*options*⟩} (no default, initially unset)

If the box is set to be `/tcb/breakable` → P. 306 and *is* broken actually, then the ⟨*options*⟩ are added to every box of the *break sequence*. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/extras unbroken**={⟨*options*⟩} (no default, initially unset)

If the box is set to be `/tcb/breakable` → P. 306 but *is not* broken actually or if the box is set to be `/tcb/unbreakable` → P. 307, then the ⟨*options*⟩ are added to the box. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/no extras unbroken** (style, no default, initially set)

Removes the unbroken extras if set before.

N 2015-07-16 **/tcb/extras first**={⟨*options*⟩} (no default, initially unset)

If the box is set to be `/tcb/breakable` → P. 306 and *is* broken actually, then the ⟨*options*⟩ are added to the *first* box of the break sequence. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/no extras first** (style, no default, initially set)

Removes the first extras if set before.

N 2015-07-16 **/tcb/extras middle**={⟨*options*⟩} (no default, initially unset)

If the box is set to be `/tcb/breakable` → P. 306 and *is* broken actually, then the ⟨*options*⟩ are added to every *middle* box (if any) of the break sequence. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/no extras middle** (style, no default, initially set)

Removes the middle extras if set before.

N 2015-07-16 **/tcb/extras last**={⟨*options*⟩} (no default, initially unset)

If the box is set to be `/tcb/breakable` → P. 306 and *is* broken actually, then the ⟨*options*⟩ are added to the *last* box of the break sequence. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/no extras last** (style, no default, initially set)

Removes the last extras if set before.

N 2015-07-16 **/tcb/extras unbroken and first**={⟨*options*⟩} (no default, initially unset)

This is an abbreviation for setting `/tcb/extras unbroken` and `/tcb/extras first` together. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/extras middle and last**={⟨*options*⟩} (no default, initially unset)

This is an abbreviation for setting `/tcb/extras middle` and `/tcb/extras last` together. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/extras unbroken and last**={⟨*options*⟩} (no default, initially unset)

This is an abbreviation for setting `/tcb/extras unbroken` and `/tcb/extras last` together. `/tcb/extras` overwrites this key.

N 2015-07-16 **/tcb/extras first and middle**={⟨*options*⟩} (no default, initially unset)

This is an abbreviation for setting `/tcb/extras first` and `/tcb/extras middle` together. `/tcb/extras` overwrites this key.

```
% \usepackage{lipsum,multicol}
% \usetikzlibrary{decorations.pathmorphing}
% \tcbuselibrary{skins}
\newtcolorbox{mybox}[1][]{
  enhanced,frame hidden,
  colframe=yellow,
  sharp corners,
  colback=green!7,coltitle=blue!50!black,colbacktitle=blue!5,
  center title,
  boxrule=0pt,toprule=1.25mm,bottomrule=1.25mm,
  extras unbroken and first={
    borderline north={0.25mm}{0.5mm}{blue,decoration={zigzag,amplitude=0.5mm},decorate}},
  extras unbroken and last={
    borderline south={0.25mm}{0.5mm}{blue,decoration={zigzag,amplitude=0.5mm},decorate}},
  #1
}

\begin{mybox}[title=My unbroken box]
\lipsum[1]
\end{mybox}

\begin{multicols}{3}
  \begin{mybox}[title=My broken box,
    enforce breakable,% use only breakable in the real world!
    break at=4.2cm,pad at break=2mm,
    height fixed for=first and middle,  ]
  \lipsum[2]
  \end{mybox}
\end{multicols}
```

My unbroken box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My broken box

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.

15.6 Breakable boxes and the `multicol` package

Unbreakable `tcolorbox`es can be used without special care inside a `multicols` environment from the `multicol` package [9].
Since version 3.10, a breakable `tcolorbox` detects, if it is used inside a `multicols` environment. But choosing break points for a breakable box cannot be done by the balancing routine of `multicols`. By default, boxes will break at `\textheight`. To get pleasant results, use the `/tcb/break at` → P. 308 and `/tcb/height fixed for` → P. 311 options.

```
% \usepackage{lipsum,multicol}  % preamble
\small
\begin{multicols}{2}
  \lipsum[1]
  \begin{tcolorbox}[enhanced jigsaw,breakable,size=title,
    colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
    title=My breakable box,pad at break=1mm, break at=7.5cm/0pt ]
  \lipsum[2-4]
  \end{tcolorbox}
  \lipsum[4]
\end{multicols}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My breakable box

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.
Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.
Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

This example is already set inside a `multicols` environment. This time, a *middle* part has full `\textheight`. `/tcb/height fixed for` → P. 311 is used to spread this box part over the full height toi align with neighboring columns.

```
% \usepackage{lipsum,multicol}
\lipsum[1]
\begin{tcolorbox}[enhanced ↩
      ↪jigsaw,breakable,
  size=title,
  colback=red!5!white,
  colframe=red!75!black,
  fonttitle=\bfseries,
  title=My breakable box,
  pad at break=2mm,
  break at=8.2cm/0pt,
  height fixed for=middle ]
\lipsum[2-7]
\end{tcolorbox}
\lipsum[8]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My breakable box

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi

quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

The following example has a `\tcolorbox` which fills the `\multicols` environment completely. Here, `/tcb/height fixed for` → P. 311 is used to give all three columns the full height. Note that the appropriate `/tcb/break at` → P. 308 value is not computed automatically but set manually.

```
% \usepackage{lipsum,multicol}  % preamble
\small
\begin{multicols}{3}
  \begin{tcolorbox}[enhanced jigsaw,breakable,size=small,
    colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
    title=My breakable box,pad at break=2mm,drop fuzzy shadow,
    height fixed for=all, break at=11.4cm ]
  \lipsum[1-3]
  \end{tcolorbox}
\end{multicols}
```

My breakable box

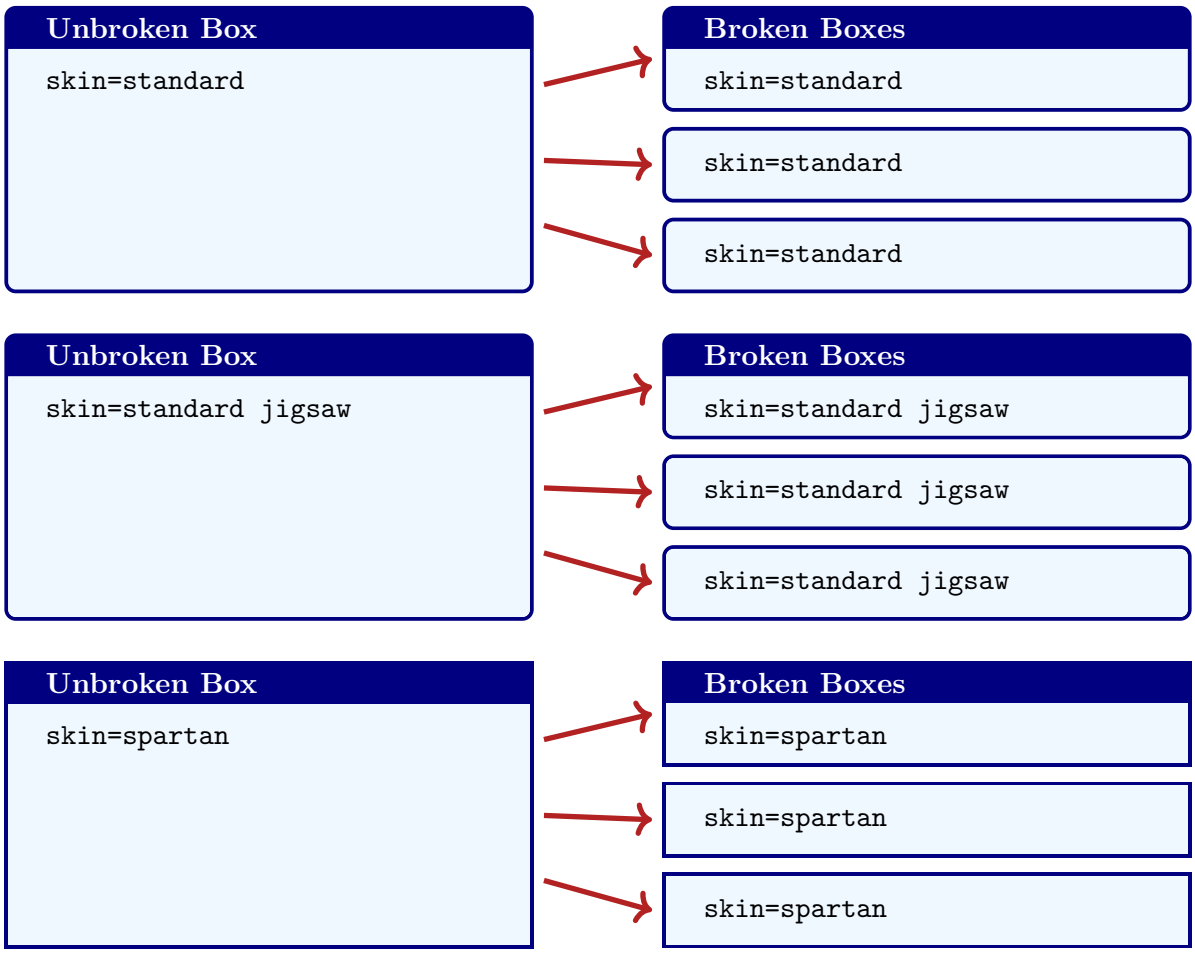
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus 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.

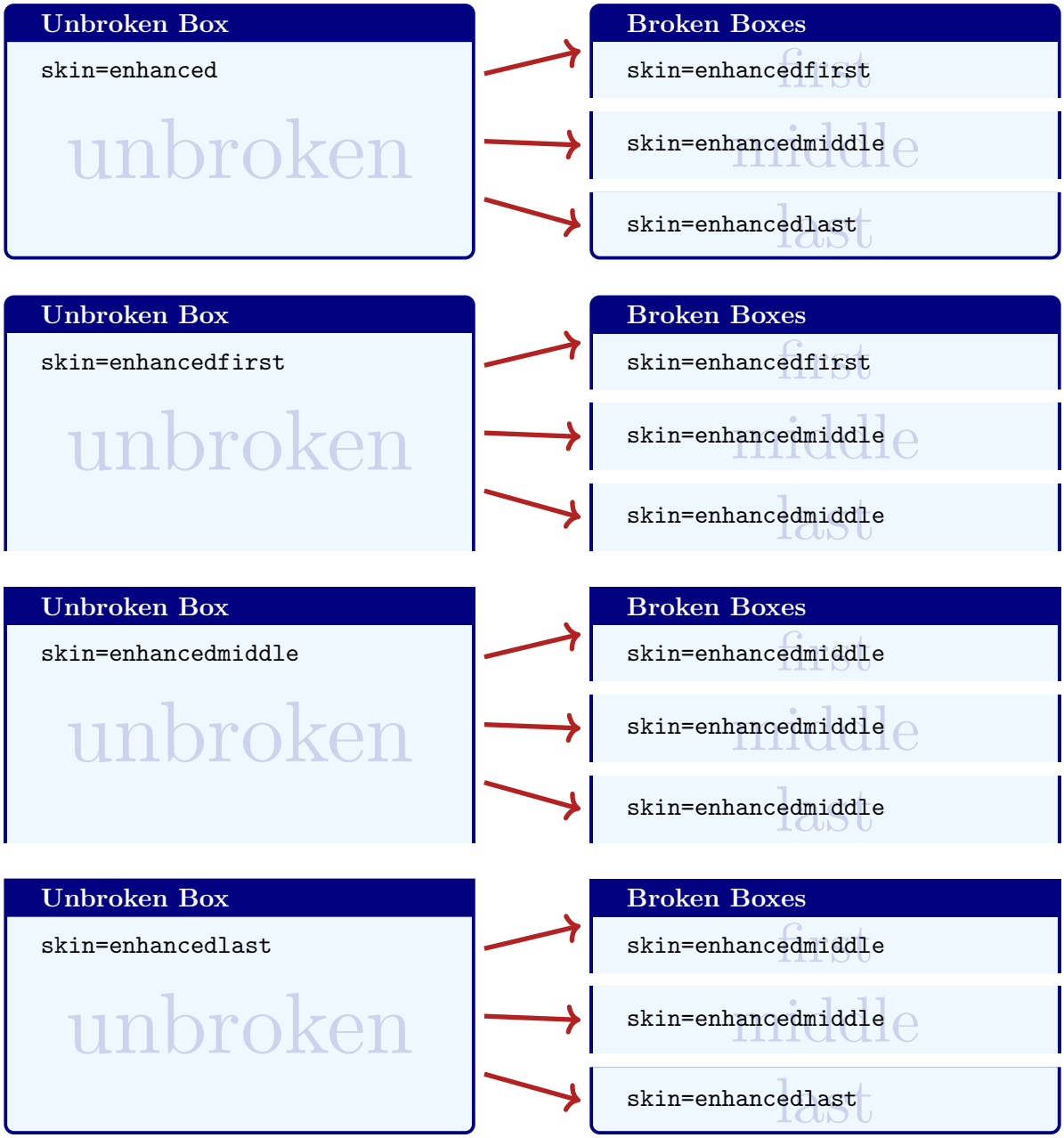
Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit 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.

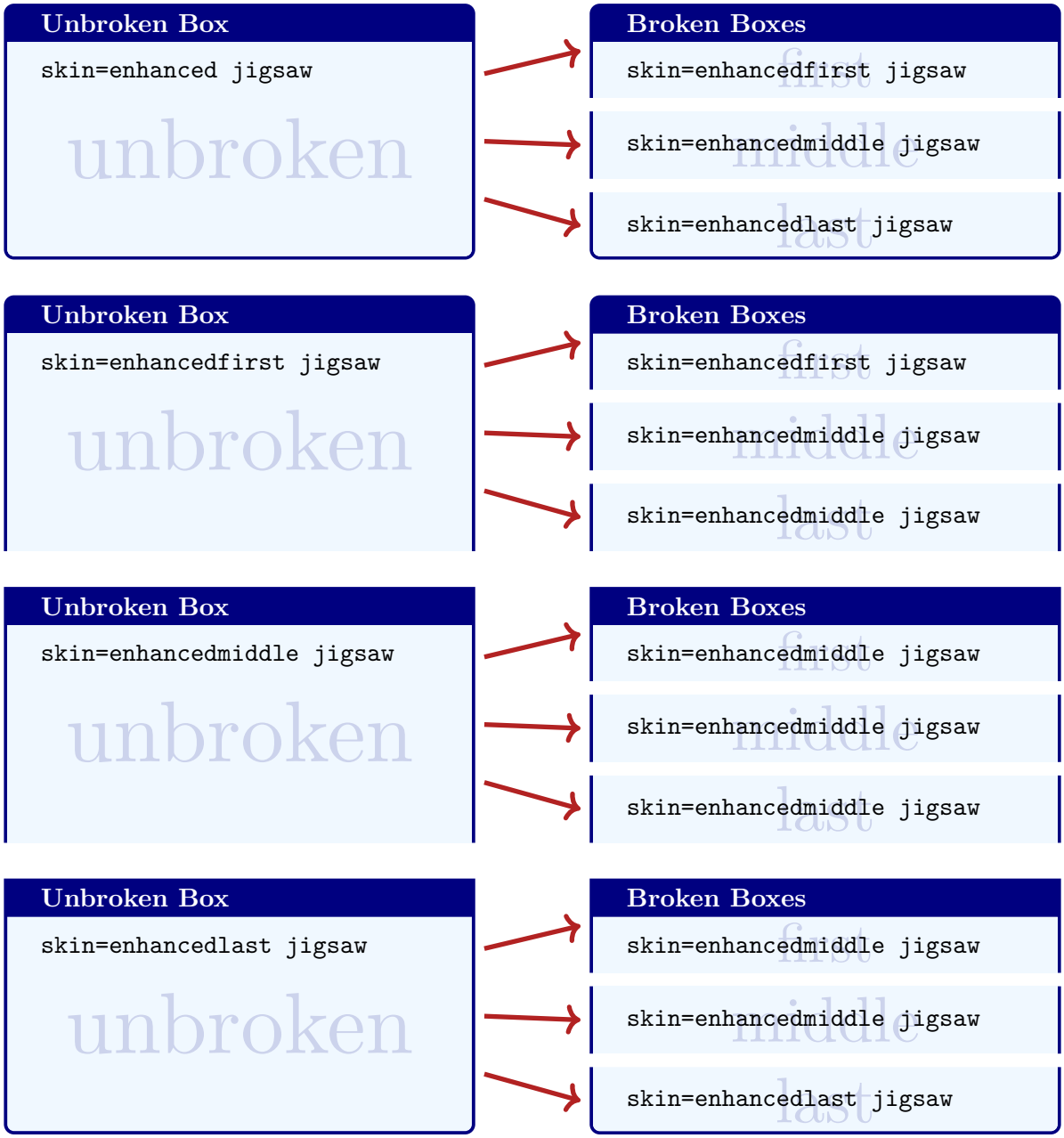
Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

15.7 Break Sequence for the Skins

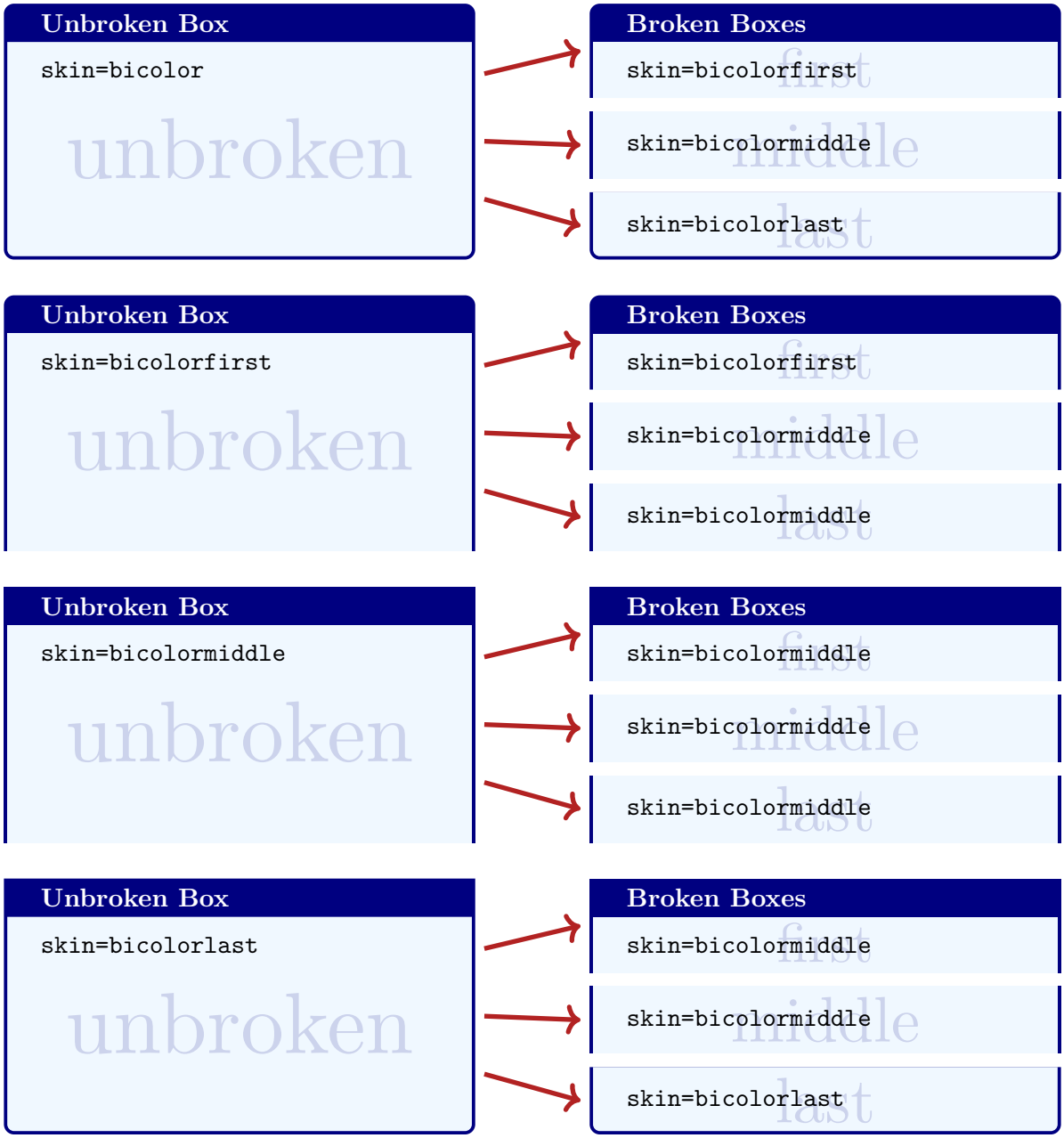
The following diagrams document the *break sequence* for different skins. Depending on the main skin of a `tcolorbox`, the actual skins of the *break sequence* parts are displayed.

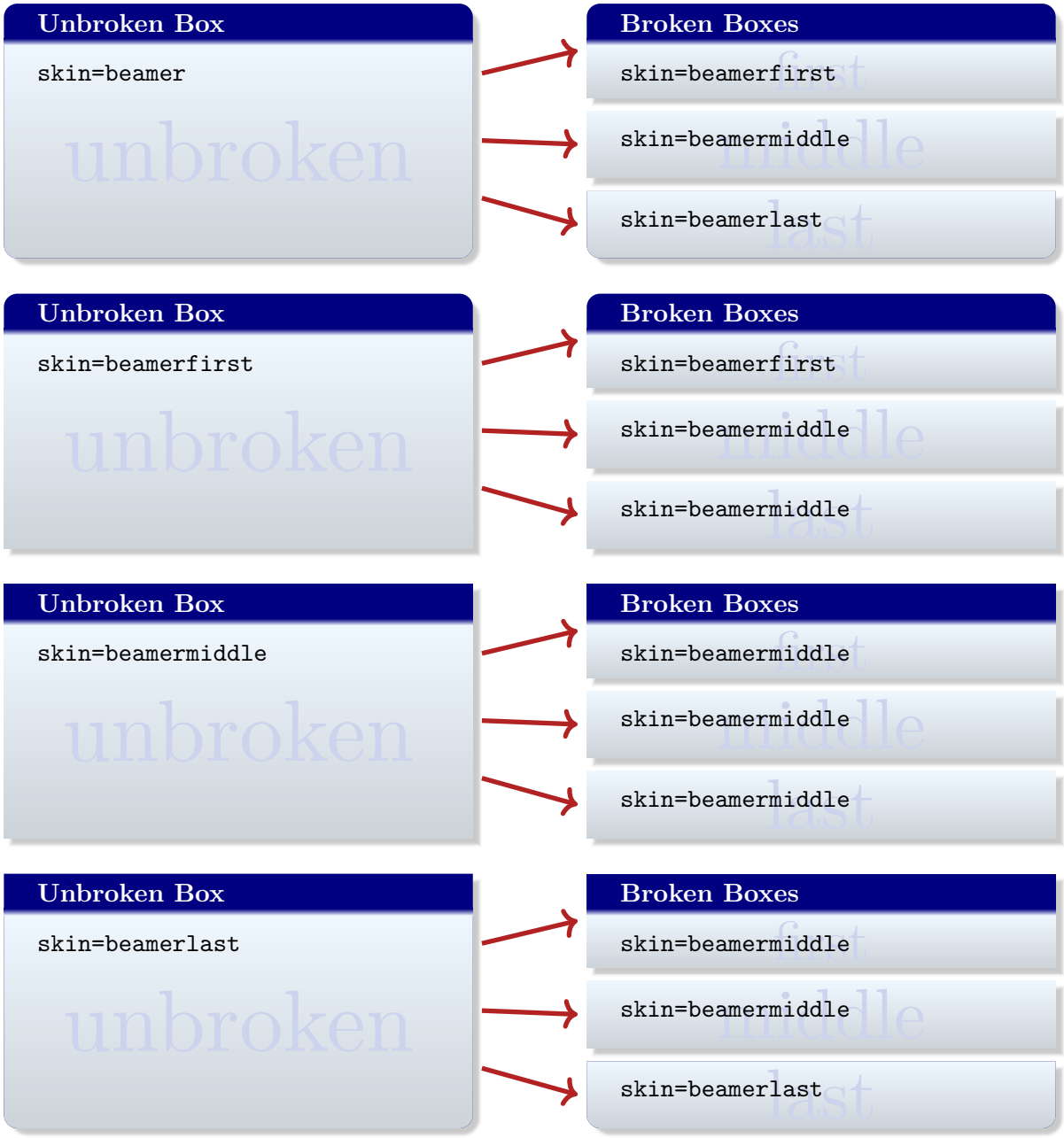


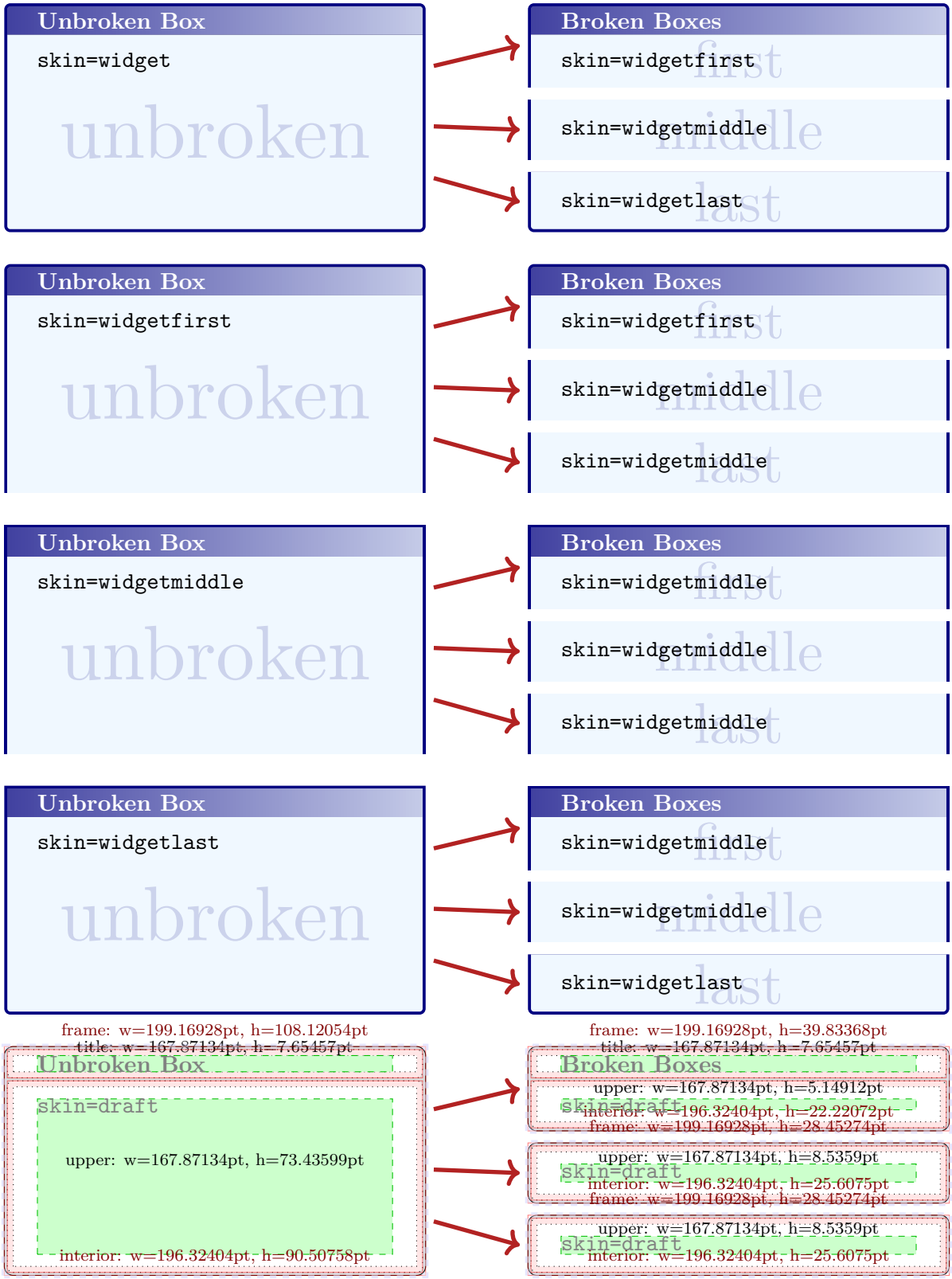

Unbroken Box
skin=enhanced
unbroken
Broken Boxes
skin=enhancedfirst
first
skin=enhancedmiddle
middle
skin=enhancedlast
last
Unbroken Box
skin=enhancedfirst
unbroken
Broken Boxes
skin=enhancedfirst
first
skin=enhancedmiddle
middle
skin=enhancedmiddle
last
Unbroken Box
skin=enhancedmiddle
unbroken
Broken Boxes
skin=enhancedmiddle
first
skin=enhancedmiddle
middle
skin=enhancedmiddle
last
Unbroken Box
skin=enhancedlast
unbroken
Broken Boxes
skin=enhancedmiddle
first
skin=enhancedmiddle
middle
skin=enhancedlast
last

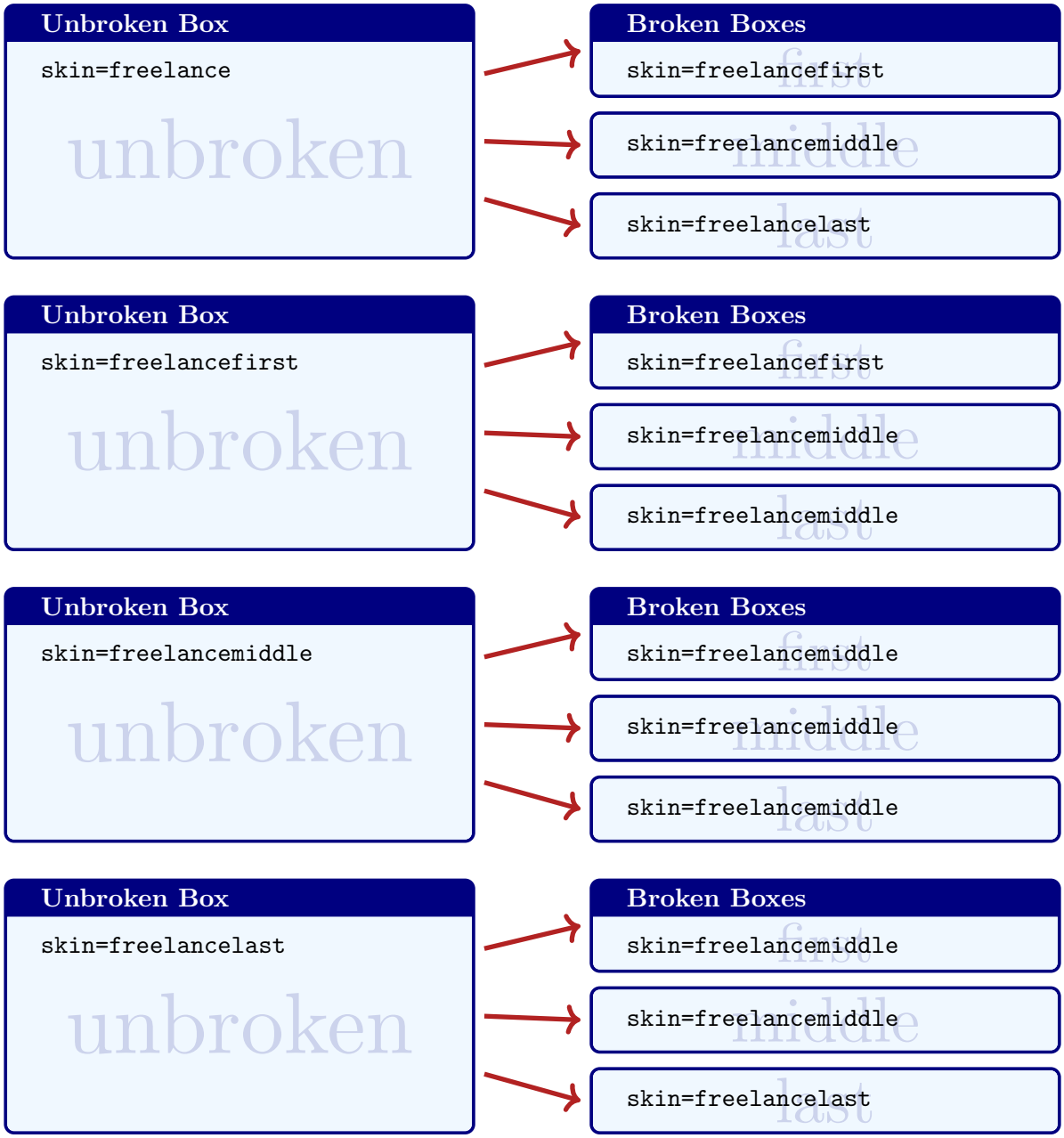

Unbroken Box
skin=enhanced jigsaw
unbroken
Broken Boxes
skin=enhancedfirst jigsaw
first
skin=enhancedmiddle jigsaw
middle
skin=enhancedlast jigsaw
last
Unbroken Box
skin=enhancedfirst jigsaw
unbroken
Broken Boxes
skin=enhancedfirst jigsaw
first
skin=enhancedmiddle jigsaw
middle
skin=enhancedmiddle jigsaw
last
Unbroken Box
skin=enhancedmiddle jigsaw
unbroken
Broken Boxes
skin=enhancedmiddle jigsaw
first
skin=enhancedmiddle jigsaw
middle
skin=enhancedmiddle jigsaw
last
Unbroken Box
skin=enhancedlast jigsaw
unbroken
Broken Boxes
skin=enhancedmiddle jigsaw
first
skin=enhancedmiddle jigsaw
middle
skin=enhancedlast jigsaw
last

skin=empty
unbroken
skin=emptyfirst
first
skin=emptymiddle
middle
skin=emptylast
last
skin=emptyfirst
unbroken
skin=emptyfirst
first
skin=emptymiddle
middle
skin=emptymiddle
last
skin=emptymiddle
unbroken
skin=emptymiddle
first
skin=emptymiddle
middle
skin=emptymiddle
last
skin=emptylast
unbroken
skin=emptymiddle
first
skin=emptymiddle
middle
skin=emptylast
last


Unbroken Box
skin=bicolor
unbroken
Broken Boxes
skin=bicolorfirst
first
skin=bicolormiddle
middle
skin=bicolorlast
last
Unbroken Box
skin=bicolorfirst
unbroken
Broken Boxes
skin=bicolorfirst
first
skin=bicolormiddle
middle
skin=bicolormiddle
last
Unbroken Box
skin=bicolormiddle
unbroken
Broken Boxes
skin=bicolormiddle
first
skin=bicolormiddle
middle
skin=bicolormiddle
last
Unbroken Box
skin=bicolorlast
unbroken
Broken Boxes
skin=bicolormiddle
first
skin=bicolormiddle
middle
skin=bicolorlast
last


Unbroken Box
skin=beamer
unbroken
Broken Boxes
skin=beamerfirst
first
skin=beamermiddle
middle
skin=beamerlast
last
Unbroken Box
skin=beamerfirst
unbroken
Broken Boxes
skin=beamerfirst
first
skin=beamermiddle
middle
skin=beamermiddle
last
Unbroken Box
skin=beamermiddle
unbroken
Broken Boxes
skin=beamermiddle
first
skin=beamermiddle
middle
skin=beamermiddle
last
Unbroken Box
skin=beamerlast
unbroken
Broken Boxes
skin=beamermiddle
first
skin=beamermiddle
middle
skin=beamerlast
last


Unbroken Box
skin=widget
unbroken
Broken Boxes
skin=widgetfirst
first
skin=widgetmiddle
middle
skin=widgetlast
last
Unbroken Box
skin=widgetfirst
unbroken
Broken Boxes
skin=widgetfirst
first
skin=widgetmiddle
middle
skin=widgetmiddle
last
Unbroken Box
skin=widgetmiddle
unbroken
Broken Boxes
skin=widgetmiddle
first
skin=widgetmiddle
middle
skin=widgetmiddle
last
Unbroken Box
skin=widgetlast
unbroken
Broken Boxes
skin=widgetmiddle
first
skin=widgetmiddle
middle
skin=widgetlast
last
frame: w=199.16928pt, h=108.12054pt
title: w=167.87134pt, h=7.65457pt
Unbroken Box
skin=draft
upper: w=167.87134pt, h=73.43599pt
interior: w=196.32404pt, h=90.50758pt
frame: w=199.16928pt, h=39.83368pt
title: w=167.87134pt, h=7.65457pt
Broken Boxes
upper: w=167.87134pt, h=5.14912pt
skin=draft
interior: w=196.32404pt, h=22.22072pt
frame: w=199.16928pt, h=28.45274pt
upper: w=167.87134pt, h=8.5359pt
skin=draft
interior: w=196.32404pt, h=25.6075pt
frame: w=199.16928pt, h=28.45274pt
upper: w=167.87134pt, h=8.5359pt
skin=draft
interior: w=196.32404pt, h=25.6075pt


Unbroken Box
skin=freelance
unbroken
Broken Boxes
skin=freelancefirst
first
skin=freelancemiddle
middle
skin=freelancelast
last
Unbroken Box
skin=freelancefirst
unbroken
Broken Boxes
skin=freelancefirst
first
skin=freelancemiddle
middle
skin=freelancemiddle
last
Unbroken Box
skin=freelancemiddle
unbroken
Broken Boxes
skin=freelancemiddle
first
skin=freelancemiddle
middle
skin=freelancemiddle
last
Unbroken Box
skin=freelancelast
unbroken
Broken Boxes
skin=freelancemiddle
first
skin=freelancemiddle
middle
skin=freelancelast
last

15.8 Break by Hand (Faked Break)

See Section 15.6 on page 314 for *real* column breaks.

Since the appearance of broken boxes is done by skins, it is quite easy to 'fake a break'. For this, you actually don't need the breakable library at all.

```
\tcbset{enhanced,equal height group=fakedbreak,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/3,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}[title=My broken box,skin=enhancedfirst]
This is a box which breaks from one column to another
\end{tcolorbox}\hfill
\begin{tcolorbox}[skin=enhancedmiddle]
column. I am sorry to say that this is a trick.
Nevertheless, you may use this trick for your
\end{tcolorbox}\hfill
\begin{tcolorbox}[skin=enhancedlast]
own purposes.
\end{tcolorbox}
```

My broken box

This is a box which breaks from one column to another

column. I am sorry to say that this is a trick. Nevertheless, you may use this trick for your

own purposes.

16 Library magazine

The main purpose of this library is to store a `tcolorbox` into an array of box registers for later usage.

If the `tcolorbox` is not breakable, there is not much add-on compared to usual TEX/LATEX box storage and usage (and you do not really need this library for that use case). For a breakable `tcolorbox`, this library allows to capture all partial boxes into a sequence of registers. The partial boxes can be used anywhere in arbitrary order.

Example Article

This is an example for an article which starts right here and is continued to the following pages. The body text for the article is written inside a single `tcolorbox`. This box is split into parts using the tools from this section,

— continued on page 329 —

The name of this library indicates *magazine* in the sense of storage, but also in the sense of a journal where an article often is *continued on page x*. An example for this kind of application is given throughout this section starting on the right hand side. The creation of this library was motivated by Ulrike Fischer and Steven B. Segletes.

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{magazine}
```

This also loads the library breakable, see Section 15 on page 304.

The box register operations of this library are global. TEX grouping will not clear the registers when leaving the current group. Also be aware that extensive use of large box arrays may eat up TEX's available memory and registers.

16.1 Creation and Resetting of Box Arrays

N 2015-07-13 `\newboxarray{⟨name⟩}`

This creates a new box array called ⟨*name*⟩. There already is a box array available with name `default` which can be used directly. Note that the creation is a global operation.

```
\newboxarray{myarray}
```

N 2015-07-13 `\boxarrayreset[⟨name⟩]`

Resets the size counter of a box array ⟨*name*⟩ to zero. If ⟨*name*⟩ is not provided, `default` is used as name. Use this or `/tcb/reset box array` before you apply `/tcb/store to box array` → P. 327. Otherwise, all boxes would be appended to the already existing boxes. This command does not clear box registers.

```
\boxarrayreset            % resets 'default'
\boxarrayreset{myarray}   % resets 'myarray'
```

N 2015-07-13 `/tcb/reset box array=⟨name⟩` (default `default`, initially unset)

Resets the size counter of a box array ⟨*name*⟩ to zero. Use this or `\boxarrayreset` (which does the same) before you apply `/tcb/store to box array` → P. 327.

```
\tcbset{
  reset box array,          % resets 'default'
  reset box array=myarray,  % resets 'myarray'
}
```

N 2015-07-13 **\boxarrayclear[⟨name⟩]**

Works like \boxarrayreset→ P. 326 to reset the size counter of a box array ⟨*name*⟩ to zero. Additionally, all allocated box registers of the box array are cleared of their content. Note that the allocated box registers stay allocated. So, this may be useful to clear memory, but not to free registers for other applications. If \consumeboxarray→ P. 330 or \consumetcboxarray→ P. 330 was used to apply the stored boxes, there is no advantage in using \boxarrayclear.

```
\boxarrayclear            % clears 'default'
\boxarrayclear{myarray}   % clears 'myarray'
```

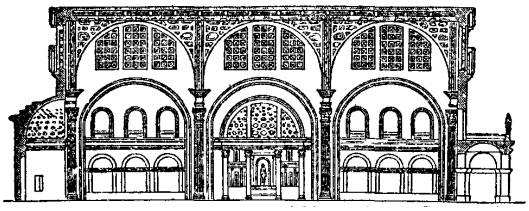
16.2 Storing Content

N 2015-07-13 **/tcb/store to box array=⟨name⟩** (default `default`, initially unset)

Stores a `tcolorbox` or all parts of a break sequence of a `tcolorbox` into a box array ⟨*name*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. Otherwise, the box array has to be created beforehand with \newboxarray→ P. 326. Note that the box has to be /tcb/breakable→ P. 306, if the box shall break into several parts. Typically, manual break points are additionally defined by /tcb/break at→ P. 308. Otherwise, the box parts will have a length of about \textheight. For most use cases, a /tcb/reset box array→ P. 326 should be applied to reset the box array counter.

```
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced jigsaw,size=fbox,width=4cm,
  colback=yellow!10,colframe=yellow!10!black,
  enforce breakable,% use only breakable in the real world!
  break at=7cm/4cm,
  height fixed for=all,
  watermark text=\arabic{tcbbreakpart},
  reset box array,
  store to box array
  ]
\lipsum[1]
\end{tcolorbox}

\useboxarray{1}\hfill
\begin{tabular}[b]{cc}
\multicolumn{2}{c}{\includegraphics[width=7cm]{Basilica_5.png}}\\
\useboxarray{2} & \useboxarray{3}
\end{tabular}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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 rhon-



cus 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.

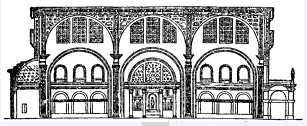
If the first box part should fill the rest of the available space of the current page, you can use `\pagegoal-\pagetotal` minus some distance for the first element of `/tcb/break at` → P. 308. You may want to have some additional distance to the preceeding text.

```
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,breakable,
  reset box array,
  store to box array,
  break at=\pagegoal-\pagetotal-5mm/0pt,
  height fixed for=first and middle]
\lipsum[1-15]
\end{tcolorbox}%
%
\consumetcboxarray{1}{blanker,before=\par\vfill\noindent}
```

```
\begin{tcolorbox}[blanker,width=4cm,
  fontupper=\footnotesize,
  enforce breakable,% use only breakable in the real world!
  break at=4cm,
  height fixed for=all,
  watermark text=\arabic{tcbbreakpart},
  reset box array,
  store to box array
  ]
\includegraphics[width=\linewidth]{Basilica_5.png}\par
\lipsum[1-2]
\end{tcolorbox}

\begin{tcbitemize}[raster columns=3,raster equal height,
  size=small,halign=center,sharp corners,colback=blue!5]
\tcbitem\consumeboxarray{5}
\tcbitem\consumeboxarray{6}
\tcbitem\consumeboxarray{1}
\tcbitem\consumeboxarray{2}
\tcbitem\consumeboxarray{3}
\tcbitem\consumeboxarray{4}
\end{tcbitemize}
```

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.



Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam

arcu libero, nonummy eget, consectetuer 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. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor

N 2015-07-13 **/tcb/reset and store to box array=⟨name⟩** (style, default `default`, initially unset)

Combination of `/tcb/reset box array` → P. 326 and `/tcb/store to box array` → P. 327.

N 2015-07-13 **/tcb/do not store to box array** (style, no default, initially set)

Disables the `/tcb/store to box array` → P. 327 option, if set before.

N 2015-07-13 `\begin{boxarraystore}{⟨name⟩}`
⟨*environment content*⟩
`\end{boxarraystore}`

Stores the environment content into a box array ⟨*name*⟩. This corresponds to the standard LaTeX environment `lrbox`, but the storage operation is global. As long as `\boxarrayreset` → P. 326 is not used, every new `boxarraystore` adds a further box to the array.

```
\boxarrayreset
\begin{boxarraystore}{default}\fbox{Mary}\end{boxarraystore}
\begin{boxarraystore}{default}\fbox{Had}\end{boxarraystore}
\begin{boxarraystore}{default}\fbox{a}\end{boxarraystore}
\begin{boxarraystore}{default}\fbox{Little}\end{boxarraystore}
\begin{boxarraystore}{default}\fbox{Lamb}\end{boxarraystore}
\useboxarray{5}\useboxarray{4}\useboxarray{3}\useboxarray{2}\useboxarray{1}\hfill
\useboxarray{1}\useboxarray{5}
```

Lamb | Little | a | Had | Mary Mary | Lamb

16.3 Retrieving Content

N 2015-07-13 `\boxarraygetsize[⟨name⟩]{⟨macro⟩}`

Stores the current size of a box array ⟨*name*⟩ into a given ⟨*macro*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used.

```
\boxarraygetsize{\mysize}
Current size of the default box array:
\mysize.
```

Current size of the default box array: 5.

— continued from page 326 —

namely `/tcb/reset and store to box array` → P. 329 with a new box array `myarticle` which was created by `\newboxarray{myarticle}`.

The resulting parts are distributed throughout this Section 16 on page 326 using `\consumetcboxarray` → P. 330 at

— continued on page 330 —

N 2015-07-13 `\useboxarray[⟨name⟩]{⟨index⟩}`

Typesets the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. It is silently ignored, if the ⟨*index*⟩ is out of range. Note that `\useboxarray` corresponds to the standard `\usebox` macro, respectively, `\copy`.

```
\boxarraygetsize{\mysize}
\foreach \n in  {1,...,\mysize} { \useboxarray{\n} }
```

Mary | Had | a | Little | Lamb

N 2015-07-13 `\usetcboxarray[⟨name⟩]{⟨index⟩}{⟨options⟩}`

Typesets the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ using `\useboxarray` → P. 329 as content of a `\tcbox` → P. 13. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. It is silently ignored, if the ⟨*index*⟩ is out of range. The `\tcbox` → P. 13 can be customized by `tcolorbox` ⟨*options*⟩.

```
\boxarraygetsize{\mysize}
\foreach \n in  {1,...,\mysize} { \usetcboxarray{\n}{on line,colframe=yellow,
  colback=yellow!10} }
```

Mary Had a Little Lamb

N 2015-07-13 `\consumeboxarray[⟨name⟩]{⟨index⟩}`

Typesets the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. It is silently ignored, if the ⟨*index*⟩ is out of range. In contrast to `\useboxarray` → P. 329, `\consumeboxarray` corresponds to the standard `\box` macro, i.e. after typesetting the box register is cleared and cannot be used again.

```
\boxarraygetsize{\mysize}
First run: \foreach \n in  {1,...,\mysize} { \consumeboxarray{\n} }
\par
Second run: \foreach \n in  {1,...,\mysize} { \consumeboxarray{\n} }
```

First run: Mary Had a Little Lamb
Second run:

N 2015-07-13 `\consumetcboxarray[⟨name⟩]{⟨index⟩}{⟨options⟩}`

Typesets the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ using `\consumeboxarray` as content of a `\tcbox` → P. 13. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. It is silently ignored, if the ⟨*index*⟩ is out of range. The `\tcbox` → P. 13 can be customized by `tcolorbox` ⟨*options*⟩. After typesetting the box register is cleared and cannot be used again.

— *continued from page 329* —

the appropriate places you see. The linking texts like *continued on page x* are created by `/tcb/finish` → P. 171 commands for the embedding `\tcbox` → P. 13. To label the box parts, `/tcb/phantomlabel` → P. 93 is used.

These quite small partial boxes are

— *continued on page 333* —

```
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced jigsaw,size=fbox,width=6cm,
  colback=yellow!10,colframe=yellow!10!black,
  enforce breakable,% use only breakable in the real world!
  break at=5cm,
  watermark text=\arabic{tcbbreakpart},
  reset and store to box array
  ]
\lipsum[1]
\end{tcolorbox}

\consumeboxarray{2} \hfill \consumeboxarray{1} \hfill \consumeboxarray{1}
```

lus 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, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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 tel-

N 2015-07-13 **\boxarraygetbox**[⟨*name*⟩]{⟨*macro*⟩}{⟨*index*⟩}

Assigns the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ to a ⟨*macro*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. If the ⟨*index*⟩ is out of range, the ⟨*macro*⟩ will be undefined.

```
\tcbox[size=small,colframe=blue!20,colback=yellow!5,on line,
  reset and store to box array]{Test}

\boxarraygetsize{\mysize} Array size: \mysize

\boxarraygetbox{\mybox}{1}
Box width: \the\wd\mybox
\quad\usebox{\mybox}
```

Array size: 1
Box width: 30.35799pt Test

16.4 Box Dimensions

N 2015-07-13 **\boxarraygetwidth**[⟨*name*⟩]{⟨*macro*⟩}{⟨*index*⟩}

Assigns the width of the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ to a ⟨*macro*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. If the ⟨*index*⟩ is out of range, the ⟨*macro*⟩ will be set to `0pt`.

```
\tcbox[size=small,colframe=blue!20,colback=yellow!5,on line,
  reset and store to box array]{Test}

\begin{tabular}{ll}
\useboxarray{1} & width of box 1: \boxarraygetwidth{\mylen}{1} \mylen\\
\useboxarray{2} & width of box 2: \boxarraygetwidth{\mylen}{2} \mylen
\end{tabular}
```

Test width of box 1: 30.35799pt
width of box 2: 0pt

N 2015-07-13 **\boxarraygetheight**[⟨*name*⟩]{⟨*macro*⟩}{⟨*index*⟩}

Assigns the height of the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ to a ⟨*macro*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. If the ⟨*index*⟩ is out of range, the ⟨*macro*⟩ will be set to `0pt`.

```
\tcbox[size=small,colframe=blue!20,colback=yellow!5,on line,
  reset and store to box array]{Test}

\begin{tabular}{ll}
\useboxarray{1} & height of box 1: \boxarraygetheight{\mylen}{1} \mylen\\
\useboxarray{2} & height of box 2: \boxarraygetheight{\mylen}{2} \mylen
\end{tabular}
```

Test height of box 1: 9.89883pt
height of box 2: 0pt

N 2015-07-13 **\boxarraygetdepth**[⟨*name*⟩]{⟨*macro*⟩}{⟨*index*⟩}

Assigns the depth of the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ to a ⟨*macro*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. If the ⟨*index*⟩ is out of range, the ⟨*macro*⟩ will be set to `0pt`.

```
\tcbox[size=small,colframe=blue!20,colback=yellow!5,on line,
  reset and store to box array]{Test}

\begin{tabular}{ll}
\useboxarray{1} & depth of box 1: \boxarraygetdepth{\mylen}{1} \mylen\\
\useboxarray{2} & depth of box 2: \boxarraygetdepth{\mylen}{2} \mylen
\end{tabular}
```

Test depth of box 1: 3.69884pt
depth of box 2: 0pt

N 2015-07-13 **\boxarraygettotalheight**[⟨*name*⟩]{⟨*macro*⟩}{⟨*index*⟩}

Assigns the total height of the box with the given ⟨*index*⟩ number from the box array ⟨*name*⟩ to a ⟨*macro*⟩. If no ⟨*name*⟩ is given, the already existing `default` box array is used. It is considered an error, if a not existing box array ⟨*name*⟩ is used. If the ⟨*index*⟩ is out of range, the ⟨*macro*⟩ will be set to `0pt`.

— continued from page 330 —

for demonstration purposes. With the tools of this section, a magazine type document could be created, but this still needs a lot of manual control.

```
\boxarrayreset
\tcbox[size=small,colframe=blue!20,colback=yellow!5,on line,
  store to box array]{Test}

\begin{tabular}{ll}
\useboxarray{1} & total height of box 1: \boxarraygettotalheight{\mylen}{1} \mylen\\
\useboxarray{2} & total height of box 2: \boxarraygettotalheight{\mylen}{2} \mylen
\end{tabular}
```

Test

total height of box 1:	13.59767pt
total height of box 2:	0pt

17 Library fitting

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{fitting}
```

17.1 Macros of the Library

\tcboxfit[⟨*options*⟩]{⟨*box content*⟩}

Creates a colored box where the given ⟨*box content*⟩ is fitted to the width and height of the box. A `tcboxfit` has to have a fixed height. If no fixed height is given, a square box is constructed. In principle, most ⟨*options*⟩ for a `tcolorbox` → P. 11 can be used for `\tcboxfit` with some restrictions. A `\tcboxfit` cannot have a lower part and cannot be broken.

```
% \usepackage{lipsum} \tcbuselibrary{raster}
\tcbset{colframe=blue!50!black,colback=red!10!white,
    boxsep=0pt,top=1mm,bottom=1mm,left=1mm,right=1mm,
    fit algorithm=hybrid*,raster equal skip=1mm}
\begin{tcbraster}[raster columns=3,raster valign=bottom]
  \tcboxfit[height=8cm]{\lipsum[1]}
  \tcboxfit[height=4cm]{\lipsum[1]}
  \tcboxfit[height=2cm]{\lipsum[1]}
\end{tcbraster}
\begin{tcbraster}[colback=green!10!white,boxsep=1mm]
  \tcboxfit[height=4cm]{\lipsum[2]}
  \tcboxfit[height=4cm,title=With a title]{\lipsum[2]}
\end{tcbraster}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

With a title

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.

See Section 19.6 on page 367 for more elaborate methods to create new commands.

\newtcboxfit[⟨*init options*⟩]{\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Creates a new macro \⟨*name*⟩ based on \tcboxfit → P. 334. Basically, \newtcboxfit operates like \newcommand. The new macro \⟨*name*⟩ optionally takes ⟨*number*⟩+1 arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying tcboxfit. The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

```
\newtcboxfit{\mybox}{colback=red!5!white,
  colframe=red!75!black,width=4cm,
  height=1.5cm,halign=center}

\mybox{This is my own box.}\par
\mybox{This is my own box with more text
       to be written.}
```

This is my own box.

This is my own box with more text to be written.

```
% \usepackage{lipsum}
\newtcboxfit{\mybox}[2]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  boxsep=1mm,left=0mm,right=0mm,top=0mm,
  bottom=0mm,halign=center,valign=center,
  nobeforeafter,width=#1,height=#2}

\mybox{2.5cm}{1cm}{First box}%
\mybox{2.5cm}{1cm}{Second box with more text}\\
\mybox{5cm}{2cm}{Third box with text}\\
\mybox{5cm}{3cm}{\lipsum[1]}
```

First box

Second box with more text

Third box with text

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
% \usepackage{lipsum}
\newtcboxfit{\mybox}[2][]{colback=red!5!white,
  colframe=red!75!black,
  width=#2,height=#2/3*2,#1}

\mybox[colback=yellow]{5cm}%
  {\lipsum[2]}
```

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.

\renewtcboxfit[⟨*init options*⟩]{\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like \newtcboxfit, but based on \renewcommand instead of \newcommand. An existing macro is redefined.

\tcbfontsize{⟨*factor*⟩}

Selects a font size inside a tcolorbox which is scaled with the given ⟨*factor*⟩ relative to \tcbfitdim.

```
\tcbset{colback=red!5!white,colframe=red!75!black}
\begin{tcolorbox}[fit basedim=10pt]
  {\tcbfontsize{0.25} Very tiny,}\\
  {\tcbfontsize{0.5} Small,}\\
  {\tcbfontsize{1} Normal,}\\
  {\tcbfontsize{2} Large,}\\
  {\tcbfontsize{4} Huge.}
\end{tcolorbox}
```

Very tiny,
Small,
Normal,
Large,
Huge.

17.2 Option Keys of the Library

The font size for the content of a box with fixed width and fixed height can be adjusted automatically. This is called the *fitbox capture mode*. Note that the fit control algorithm constructs a series of versions for the box and selects the 'best'. Therefore, the compilation time is quite longer than for a normal box. The algorithm will fail, if a different selected font size does not change the overall size of the box content. The `\tcboxfit` → P. 334 macro uses this algorithm by default.

The fit control keys are only applicable to unbreakable boxes without a lower part. The box content should not change counters.

/tcb/fit (style, initially unset)

Sets the `/tcb/capture` → P. 89 mode to `fitbox`, i. e. enables the font size adjustment algorithm. Thereby, a `tcolorbox` → P. 11 acts like `\tcboxfit` → P. 334 where the given ⟨*box content*⟩ is fitted to the width and height of the box. Therefore, the box has to have a fixed height. If no fixed height is given, a square box is constructed. The font dimension **`\tcbfitdim`** can also be used to adjust the margins of the box since a box with a tiny font may not need large margins. The number of constructed boxes is saved to the macro **`\tcbfitsteps`** for analysis.

```
% \usepackage{lipsum}
% \tcbuselibrary{skins}
\newtcolorbox{fitting}[2][]{fit,height=#2,boxsep=1pt,valign=center,opacityupper=0.5,
  top=0.4\tcbfitdim,bottom=0.4\tcbfitdim,left=0.75\tcbfitdim,right=0.75\tcbfitdim,
  enhanced,watermark text={\tcbfitsteps},colframe=blue!75!black,colback=white,#1}

\begin{fitting}{4cm}
\lipsum[1]
\end{fitting}

\begin{fitting}{2cm}
\lipsum[2]
\end{fitting}

\begin{fitting}{1cm}
\lipsum[3]
\end{fitting}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

/tcb/fit to=⟨*width*⟩ and ⟨*height*⟩ (style, initially unset)

Shortcut for using /tcb/fit → P. 336 and setting the ⟨*width*⟩ and ⟨*height*⟩ values separately.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[fit to=3cm and 2cm]
  This box content is fitted to the given
  dimensions.
\end{tcolorbox}
```

This box content is fitted to the given dimensions.

/tcb/fit to height=⟨*height*⟩ (style, initially unset)

Shortcut for using /tcb/fit → P. 336 and setting the ⟨*height*⟩ value separately.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[fit to height=2cm]
  This box content is fitted to the given
  height.
\end{tcolorbox}
```

This box content is fitted to the given height.

/tcb/fit basedim=⟨*length*⟩ (no default, initially 10pt)

Sets the starting font dimension for the font size adjustment algorithm to ⟨*length*⟩. The algorithm never enlarges this dimension.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[fit to=4cm and 2cm]
  Too few words for the box.
\end{tcolorbox}

\begin{tcolorbox}[fit to=4cm and 2cm,
  fit basedim=50pt]
  Enough words for the box.
\end{tcolorbox}
```

Too few words for the box.

Enough words for the box.

/tcb/fit skip=⟨*real value*⟩ (no default, initially 1.2)

Sets the skip value of the selected font to ⟨*real value*⟩ times \tcbfitdim.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,
  colframe=red!75!black,left=1mm,
  right=1mm,boxsep=0mm}

\begin{tcolorbox}[fit to=5cm and 4cm,
  fit skip=1.0  ]
  \lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/fit fontsize macros (style, initially unset)

Redefines the standard LaTeX font size macros `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, and `\Huge`, to set font sizes relative to the current `\tcbfitdim`. Note that the display skip values for mathematical formulas are respected by the redefined macros.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,
  colframe=red!75!black,left=1mm,
  right=1mm,boxsep=0mm}

\begin{tcolorbox}[fit to height=4cm]
  {\Large\bfseries This text is
              not adapted:\par}
  \lipsum[2]
\end{tcolorbox}

\begin{tcolorbox}[fit to height=4cm,
  fit fontsize macros ]
  {\Large\bfseries This text is adapted:\par}
  \lipsum[2]
\end{tcolorbox}
```

This text is not adapted:
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.

This text is adapted:
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.

```
\tcbset{colback=red!5!white,
  colframe=red!75!black,left=1mm,
  right=1mm,boxsep=0mm}

\let\realHuge=\Huge

\begin{tcolorbox}[fit basedim=7pt,
  fontupper=\normalsize,
  fit fontsize macros]
The relative relative font size macros
are also usable without the
\textit{fit} algorithm.\par
{\Huge Adapted Huge} ---
{\realHuge Original Huge}
\end{tcolorbox}
```

The relative relative font size macros are also usable without the *fit* algorithm.
Adapted Huge — Original Huge

```
\tcbset{size=fbox,colback=red!5!white,
  colframe=red!75!black}

\tcboxfit[height=5cm,
  fit fontsize macros,
  fonttitle=\normalsize\bfseries,
  title=Adapted title]
{\lipsum[2]}
```

Adapted title
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.

/tcb/fit height plus=⟨*dimension*⟩ (no default, initially `0pt`)

The box is allowed to enlarge the fixed height up to the given ⟨*dimension*⟩, before a font size fit is applied. An optional `/tcb/fit width plus` is tried after the height adaption.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,width=3cm,height=3cm,nobeforeafter}

\begin{tcolorbox}[fit]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit,fit height plus=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit]
\lipsum[2]
\end{tcolorbox}
\begin{tcolorbox}[fit,fit height plus=1cm]
\lipsum[2]
\end{tcolorbox}
```

This is a tcolorbox.

This is a tcolorbox.

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.

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.

/tcb/fit width plus=⟨*dimension*⟩ (no default, initially `0pt`)

The box is allowed to enlarge the fixed width up to the given ⟨*dimension*⟩, before a font size fit is applied. An optional `/tcb/fit height plus` is tried before the width adaption.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,width=3cm,height=3cm,nobeforeafter}

\begin{tcolorbox}[fit]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit,fit width plus=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit]
\lipsum[2]
\end{tcolorbox}
\begin{tcolorbox}[fit,fit width plus=1cm]
\lipsum[2]
\end{tcolorbox}
```

This is a tcolorbox.

This is a tcolorbox.

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.

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.

Typically but not necessarily, the optional title of a `tcolorbox` is not part of the fit operation. If a `/tcb/fit width plus` is applied, the title is also adapted to the new width. If counters are increased inside the title text, they may be increased more than one time. To avoid this, you are encouraged to use `/tcb/phantom` → P. 93 or `/tcb/step and label` → P. 93 to set counters or use automatic numbering, see Subsection 5.1 from page 97.

`/tcb/fit width from=`⟨*min*⟩ `to` ⟨*max*⟩ (style, no default)

Sets the box width to ⟨*min*⟩ and allows the width to grow up to ⟨*max*⟩.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,height=4cm}

\begin{tcolorbox}[fit,width=\linewidth/2]
\lipsum[2]
\end{tcolorbox}\par
\begin{tcolorbox}[fit width from=\linewidth/2 to \linewidth]
\lipsum[2]
\end{tcolorbox}\par
```

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.

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.

/tcb/fit height from=⟨*min*⟩ **to** ⟨*max*⟩ (style, no default)

Sets the box height to ⟨*min*⟩ and allows the height to grow up to ⟨*max*⟩.

```
% \usepackage{lipsum}
\newtcolorbox{mybox}{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,
  bottom=1mm,right=1mm,boxsep=0mm,width=4cm,nobeforeafter,
  fit height from=1cm to 8cm}

\begin{mybox}
This is a tcolorbox.
\end{mybox}
\begin{mybox}
This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.
\end{mybox}
\begin{mybox}
\lipsum[2]
\end{mybox}
```

This is a tcolorbox.

This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.

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.

`/tcb/fit algorithm=`⟨*name*⟩ (no default, initially `fontsize`)

Sets the algorithm for the fitting process *after* optionally width and height are adapted. Feasible values for ⟨*name*⟩ are:

- `fontsize` (initial): The algorithm is a bisection method that adapts the font size until certain stop conditions are fulfilled. This is the most time-consuming method but it is robust and gives pleasant results.

  The used font has to be freely scalable for this method! Other content than text is not scaled down. The aspect ratio is fully garanteed.

N 2014-10-29

- `fontsize*`: First, the `fontsize` algorithm is applied. If the font was scaled down and the resulting height is too small, the box is squeezed to fit the area.

  The used font has to be freely scalable for this method! Other content than text may be slightly rescaled. The aspect ratio cannot be fully garanteed.

- `areasize`: The algorithm calculates the area size for the text without scaling the font. The text box is shaped for the needed aspect ratio in one or two steps. Finally, it is scaled down with a standard `\resizebox` macro.

  The used font has not to be scalable. Every box content is scaled down. The aspect ratio cannot be fully garanteed.

N 2014-10-29

- `areasize*`: The `areasize` algorithm is applied, but if the content was scaled down and the resulting height is too small, the box is squeezed to fit the area.

  The used font has not to be scalable. Every box content is scaled down. The aspect ratio cannot be fully garanteed.

- `hybrid`: First, this algorithm estimates the needed font size in one or two steps. Then an `areasize` fitting as above is a applied.

  The used font has to be freely scalable for this method! Other content than text may be slightly rescaled. The aspect ratio cannot be fully garanteed.

N 2014-10-29

- `hybrid*`: First, this algorithm estimates the needed font size in one or two steps. Then an `areasize*` fitting as above is a applied.

  The used font has to be freely scalable for this method! Other content than text may be slightly rescaled. The aspect ratio cannot be fully garanteed.

- `squeeze`: The text box is brutally scaled down to fit.

  The aspect ratio is very likely to be horrible. You should not use this method for final documents.

```
% \usepackage{lipsum}
\newtcboxfit{mybox}[1]{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,
  bottom=1mm,right=1mm,boxsep=0mm,width=3.5cm,height=7cm,nobeforeafter,
  before upper=\textcolor{blue}{\rule{5mm}{5mm}}\ ,
  enhanced,watermark text={\tcbfitsteps},
  fonttitle=\bfseries,adjusted title=#1,fit algorithm=#1}

\mybox{fontsize}{\lipsum[2]}\hfill
\mybox{hybrid}{\lipsum[2]}\hfill
\mybox{areasize}{\lipsum[2]}\hfill
\mybox{squeeze}{\lipsum[2]}

Quality \dotfill versus \dotfill Speed
```

fontsize

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.

hybrid

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.

areasize

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.

squeeze

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.

Quality . . . . . versus . . . . . Speed

```
% \usepackage{lipsum}
\newtcboxfit{mybox}[2]{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,
  size=tight,width=7.2cm,height=5cm,nobeforeafter,
  before upper=\textcolor{blue}{\rule{5mm}{5mm}}\ ,
  enhanced,fonttitle=\bfseries,adjusted title=#2,fit algorithm=#1}

\mybox{hybrid}{hybrid (possible gap at end)}{\lipsum[1]}\hfill
\mybox{hybrid*}{hybrid* (no gap but possibly squeezed)}{\lipsum[1]}
```

hybrid (possible gap at end)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

hybrid* (no gap but possibly squeezed)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

The following options set control parameters for the fit algorithm. Mainly, they apply to the `fontsize` variant, see `/tcb/fit algorithm` → P. 342. The options should be seen as experimental and are likely to change in future versions, if necessary.

/tcb/fit maxstep=⟨*number*⟩ (no default, initially `20`)

Sets the maximal step size for the font size adjustment algorithm. In normal situations, the algorithm stops before reaching the intial value of 20 steps. If the box content does not shrink, this value prevents an endless loop.

/tcb/fit maxfontdiff=⟨*dimension*⟩ (no default, initially `0.1pt`)

The algorithm stops, if the font size is determined within a deviation of ⟨*dimension*⟩.

/tcb/fit maxfontdiffgap=⟨*dimension*⟩ (no default, initially `1pt`)

The algorithm stops, if the number of lines is determined and the font size is determined within a deviation of ⟨*dimension*⟩.

/tcb/fit maxwidthdiff=⟨*dimension*⟩ (no default, initially `1pt`)

The algorithm stops, if the (optionally) flexible box width is determined within a deviation of ⟨*dimension*⟩.

/tcb/fit maxwidthdiffgap=⟨*dimension*⟩ (no default, initially `10pt`)

The algorithm stops, if the number of lines is determined and the (optionally) flexible box width is determined within a deviation of ⟨*dimension*⟩.

/tcb/fit warning=⟨*value*⟩ (no default, initially `off`)

Typically, the fit control algorithm constructs several auxiliary boxes to determine the optimal one. If not switched off, the construction of the auxiliary boxes may produce many `hbox` warnings. This option key changes the `\hbadness` value.

- `off`: Most of `'Underfull \hbox'` and `'Overfull \hbox'` warnings are switched off (including the ones for the finally used box).
- `on`: All warnings for all auxiliary boxes are displayed.
- `final`: Only warnings for the finally used box are displayed. Note that an additional box has to be contructed for theses messages.

18 Library hooks

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{hooks}
```

For the skin related options, the library skins has to be loaded separately.

18.1 Concept of Hooks

A hook is a placeholder in some LaTeX code where additional code can be added. For example, the LaTeX macro `\AtBeginDocument` adds code to a hook which is placed at the beginning of every document.

Several option keys of `tcolorbox` allow providing some code which is added to specific places of a colored box. For example, `/tcb/before upper` → P. 58 places code before the content of the upper part. A following usage of this key overwrites any prior settings.

The library hooks extends `/tcb/before upper` → P. 58 and several more existing keys to 'hookable' versions, e. g. `/tcb/before upper app` → P. 346 and `/tcb/before upper pre` → P. 346. The 'hookable' keys don't overwrite prior settings but either *app*end or *pre*pend the newly given code to the existing code.

The general naming convention (with some small exceptions) is:

- ⟨*option key*⟩ `app`: works like ⟨*option key*⟩ but *app*ends its code to the existing code.
- ⟨*option key*⟩ `pre`: works like ⟨*option key*⟩ but *pre*pends its code to the existing code.

If the original ⟨*option key*⟩ is used (again), all code will be overwritten. Therefore, the order of the option key usage is crucial.

```
% \usepackage{array,tabularx}
\newcolumntype{Y}{>{\raggedleft\arraybackslash}X}% see tabularx
\tcbset{enhanced,fonttitle=\bfseries\large,fontupper=\normalsize\sffamily,
  colback=yellow!10!white,colframe=red!50!black,colbacktitle=Salmon!30!white,
  coltitle=black,center title,
  tabularx={X||Y|Y|Y|Y||Y},% this sets 'before upper' and 'after upper'
  before upper app={Group & One & Two & Three & Four & Sum\\\hline\hline}  }

\begin{tcolorbox}[title=My table]
Red   & 1000.00 & 2000.00 &  3000.00 &  4000.00 & 10000.00\\\hline
Green & 2000.00 & 3000.00 &  4000.00 &  5000.00 & 14000.00\\\hline
Blue  & 3000.00 & 4000.00 &  5000.00 &  6000.00 & 18000.00\\\hline\hline
Sum   & 6000.00 & 9000.00 & 12000.00 & 15000.00 & 42000.00
\end{tcolorbox}
```

My table

Group	One	Two	Three	Four	Sum
Red	1000.00	2000.00	3000.00	4000.00	10000.00
Green	2000.00	3000.00	4000.00	5000.00	14000.00
Blue	3000.00	4000.00	5000.00	6000.00	18000.00
Sum	6000.00	9000.00	12000.00	15000.00	42000.00

18.2 Box Content Additions

The following option keys extend the options given in Subsection 4.11 from page 58.

/tcb/before title app=⟨*code*⟩ (no default)
Appends the given ⟨*code*⟩ to /tcb/before title →P. 58 *after* the color and font settings and *before* the content of the title.

/tcb/before title pre=⟨*code*⟩ (no default)
Prepends the given ⟨*code*⟩ to /tcb/before title →P. 58 *after* the color and font settings and *before* the content of the title.

/tcb/after title app=⟨*code*⟩ (no default)
Appends the given ⟨*code*⟩ to /tcb/after title →P. 58 *after* the content of the title.

/tcb/after title pre=⟨*code*⟩ (no default)
Prepends the given ⟨*code*⟩ to /tcb/after title →P. 58 *after* the content of the title.

/tcb/before upper app=⟨*code*⟩ (no default)
Appends the given ⟨*code*⟩ to /tcb/before upper →P. 58 *after* the color and font settings and *before* the content of the upper part.

/tcb/before upper pre=⟨*code*⟩ (no default)
Prepends the given ⟨*code*⟩ to /tcb/before upper →P. 58 *after* the color and font settings and *before* the content of the upper part.

/tcb/after upper app=⟨*code*⟩ (no default)
Appends the given ⟨*code*⟩ to /tcb/after upper →P. 59 *after* the content of the upper part.

/tcb/after upper pre=⟨*code*⟩ (no default)
Prepends the given ⟨*code*⟩ to /tcb/after upper →P. 59 *after* the content of the upper part.

```
% \tcbuselibrary{theorems}
\tcbset{ams align,%  this sets 'before upper' and 'after upper'
  colback=yellow!10!white,colframe=red!50!black,
  before upper app={\frac{2}{\sqrt{2}}&=\sqrt{2}.\\},
  after upper pre={\\\sin\left(\frac{\pi}{2}\right)&=1.},
}

\begin{tcolorbox}
  \sum\limits_{n=1}^{\infty} \frac{1}{n} &= \infty.\\
  \int x^2 ~\text{d}x &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\frac{2}{\sqrt{2}} = \sqrt{2}. \tag{22}$$

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{23}$$

$$\int x^2 \,\mathrm{d}x = \frac{1}{3}x^3 + c. \tag{24}$$

$$\sin\left(\frac{\pi}{2}\right) = 1. \tag{25}$$

`/tcb/before lower app=`⟨*code*⟩ (no default)

Appends the given ⟨*code*⟩ to `/tcb/before lower`→P. 59 *after* the color and font settings and *before* the content of the lower part.

`/tcb/before lower pre=`⟨*code*⟩ (no default)

Prepends the given ⟨*code*⟩ to `/tcb/before lower`→P. 59 *after* the color and font settings and *before* the content of the lower part.

`/tcb/after lower app=`⟨*code*⟩ (no default)

Appends the given ⟨*code*⟩ to `/tcb/after lower`→P. 59 *after* the content of the lower part.

`/tcb/after lower pre=`⟨*code*⟩ (no default)

Prepends the given ⟨*code*⟩ to `/tcb/after lower`→P. 59 *after* the content of the lower part.

18.3 Embedding into the Surroundings

The following option keys extend the options given in Subsection 4.15 from page 76.

The 'hookable' versions are usable inside the document. In the preamble, they can only be used after explicit setting of `/tcb/before`→P. 76 and `/tcb/after`→P. 76 or by e. g. `/tcb/parskip`→P. 76.

`/tcb/before app=`⟨*code*⟩ (no default)

Appends the given ⟨*code*⟩ to `/tcb/before`→P. 76 before the colored box.

`/tcb/before pre=`⟨*code*⟩ (no default)

Prepends the given ⟨*code*⟩ to `/tcb/before`→P. 76 before the colored box.

`/tcb/after app=`⟨*code*⟩ (no default)

Appends the given ⟨*code*⟩ to `/tcb/after`→P. 76 after the colored box.

`/tcb/after pre=`⟨*code*⟩ (no default)

Prepends the given ⟨*code*⟩ to `/tcb/after`→P. 76 after the colored box.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,before app={The box follows:\\[4pt]},
  after app={This is the end.}]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

The box follows:

My title

This is a **tcolorbox**.

This is the end.

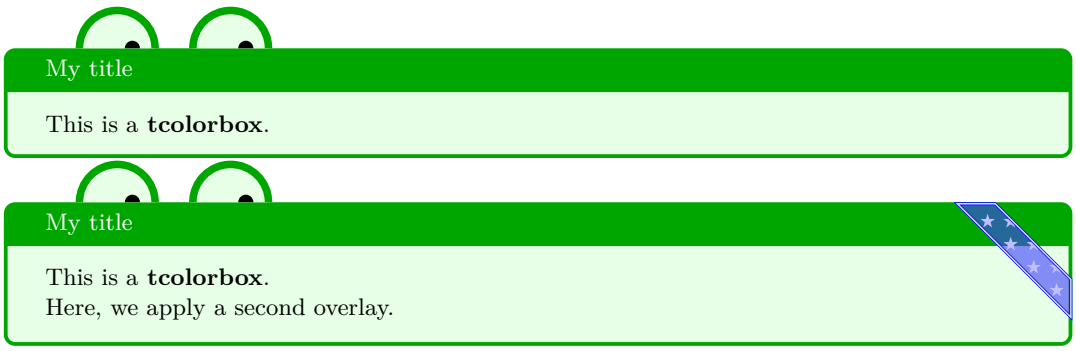
18.4 Overlays

The following option keys extend the options given in Subsection 4.12 from page 64.

/tcb/overlay app=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to /tcb/overlay→ P. 64.

```
% \usetikzlibrary{patterns} % preamble
% \tcbuselibrary{skins} % preamble
\tcbset{frogbox/.style={enhanced,colback=green!10,colframe=green!65!black,
  enlarge top by=5.5mm,
  overlay={\foreach \x in {2cm,3.5cm} {
    \begin{scope}[shift={([xshift=\x]frame.north west)}]
      \path[draw=green!65!black,fill=green!10,line width=1mm] (0,0) arc (0:180:5mm);
      \path[fill=black] (-0.2,0) arc (0:180:1mm);
    \end{scope}}}]}}
\tcbset{ribbon/.style={overlay app={%
  \path[fill=blue!75!white,draw=blue,double=white!85!blue,
    preaction={opacity=0.6,fill=blue!75!white},
    line width=0.1mm,double distance=0.2mm,
    pattern=fivepointed stars,pattern color=white!75!blue]
    ([xshift=-0.2mm,yshift=-1.02cm]frame.north east)
    -- ++(-1,1) -- ++(-0.5,0) -- ++(1.5,-1.5) -- cycle;}}}

\begin{tcolorbox}[frogbox,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}

\begin{tcolorbox}[frogbox,ribbon,title=My title]
This is a \textbf{tcolorbox}.\par
Here, we apply a second overlay.
\end{tcolorbox}
```



/tcb/overlay pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to /tcb/overlay→ P. 64.

/tcb/overlay unbroken app=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to /tcb/overlay unbroken→ P. 65.

/tcb/overlay unbroken pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to /tcb/overlay unbroken→ P. 65.

/tcb/overlay first app=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to /tcb/overlay first→ P. 65.

/tcb/overlay first pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to /tcb/overlay first→ P. 65.

`/tcb/overlay middle app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay middle`→ P. 65.

`/tcb/overlay middle pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay middle`→ P. 65.

`/tcb/overlay last app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay last`→ P. 65.

`/tcb/overlay last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay last`→ P. 65.

`/tcb/overlay broken app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay broken`→ P. 65.

`/tcb/overlay broken pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay broken`→ P. 65.

`/tcb/overlay unbroken and first app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay unbroken and first`→ P. 65.

`/tcb/overlay unbroken and first pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay unbroken and first`→ P. 65.

`/tcb/overlay middle and last app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay middle and last`→ P. 65.

`/tcb/overlay middle and last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay middle and last`→ P. 65.

`/tcb/overlay unbroken and last app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay unbroken and last`→ P. 65.

`/tcb/overlay unbroken and last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay unbroken and last`→ P. 65.

N 2014-09-19 `/tcb/overlay first and middle app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/overlay first and middle`→ P. 65.

N 2014-09-19 `/tcb/overlay first and middle pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay first and middle`→ P. 65.

18.5 Watermarks

The following option keys extend the options given in Subsection 9.3 from page 139.

Watermarks are special overlays. The LIB hooks library allows the combination of several watermarks and overlays.

/tcb/watermark text app=⟨*text*⟩ (no default)

Appends a /tcb/watermark text → P. 139 to the colored box.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark graphics=Basilica_5.png,
  watermark opacity=0.25,
  watermark text app=Basilica,watermark color=Navy
  ]
\lipsum[1-2]
\tcblower
This example uses a public domain picture from\\
\url{http://commons.wikimedia.org/wiki/File:Basilica_5.png}
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

This example uses a public domain picture from
http://commons.wikimedia.org/wiki/File:Basilica_5.png

/tcb/watermark text pre=⟨*text*⟩ (no default)

Prepends a /tcb/watermark text → P. 139 to the colored box.

/tcb/watermark text app on=⟨*part*⟩ **is** ⟨*text*⟩ (no default)

Appends a /tcb/watermark text on → P. 139 the named ⟨*part*⟩ of a break sequence.

/tcb/watermark text pre on=⟨*part*⟩ **is** ⟨*text*⟩ (no default)

Prepends a /tcb/watermark text on → P. 139 the named ⟨*part*⟩ of a break sequence.

/tcb/watermark graphics app=⟨*file name*⟩ (no default)

Appends a /tcb/watermark graphics →P. 140 referenced by ⟨*file name*⟩ to the colored box.

/tcb/watermark graphics pre=⟨*file name*⟩ (no default)

Prepends a /tcb/watermark graphics →P. 140 referenced by ⟨*file name*⟩ to the colored box.

/tcb/watermark graphics app on=⟨*part*⟩ **is** ⟨*file name*⟩ (no default)

Appends a /tcb/watermark graphics on →P. 140 the named ⟨*part*⟩ of a break sequence. The picture is referenced by ⟨*file name*⟩.

/tcb/watermark graphics pre on=⟨*part*⟩ **is** ⟨*file name*⟩ (no default)

Prepends a /tcb/watermark graphics on →P. 140 the named ⟨*part*⟩ of a break sequence. The picture is referenced by ⟨*file name*⟩.

/tcb/watermark tikz app=⟨*graphical code*⟩ (no default)

Appends a /tcb/watermark tikz →P. 141 with the given **tikz** ⟨*graphical code*⟩ to the colored box.

/tcb/watermark tikz pre=⟨*graphical code*⟩ (no default)

Prepends a /tcb/watermark tikz →P. 141 with the given **tikz** ⟨*graphical code*⟩ to the colored box.

```
% \usepackage{tikz}
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  watermark color=Navy,watermark opacity=0.25,
  smiley/.style={watermark tikz pre={%
    \path[fill=yellow,draw=yellow!75!red] (0,0) circle (1cm);
    \fill[red] (45:5mm) circle (1mm);
    \fill[red] (135:5mm) circle (1mm);
    \draw[line width=1mm,red] (215:5mm) arc (215:325:5mm);}}}

\begin{tcolorbox}[enhanced,title=My title, watermark text=Watermark,
  smiley]
\lipsum[1-2]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

Watermark

/tcb/watermark tikz app on=⟨*part*⟩ **is** ⟨*graphical code*⟩ (no default)

Appends a /tcb/watermark tikz on →P. 141 the named ⟨*part*⟩ of a break sequence.

/tcb/watermark tikz pre on=⟨*part*⟩ **is** ⟨*graphical code*⟩ (no default)

Prepends a /tcb/watermark tikz on →P. 141 the named ⟨*part*⟩ of a break sequence.

18.6 Underlays

The following option keys extend the options given in Section 9.8 on page 169. There are no `app` type keys since underlays are stackable by default.

`/tcb/underlay pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay`→P. 169.

`/tcb/underlay unbroken pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay unbroken`→P. 170.

`/tcb/underlay first pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay first`→P. 170.

`/tcb/underlay middle pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay middle`→P. 170.

`/tcb/underlay last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay last`→P. 170.

`/tcb/underlay boxed title pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay boxed title`→P. 170.

`/tcb/underlay broken pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay broken`→P. 170.

`/tcb/underlay unbroken and first pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay unbroken and first`→P. 170.

`/tcb/underlay middle and last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay middle and last`→P. 170.

`/tcb/underlay unbroken and last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay unbroken and last`→P. 170.

N 2014-09-19 `/tcb/underlay first and middle pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/underlay first and middle`→P. 170.

18.7 Finishes

The following option keys extend the options given in Section 9.9 on page 171. There are no `app` type keys since finishes are stackable by default.

`/tcb/finish pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish`→P. 171.

`/tcb/finish unbroken pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish unbroken`→P. 172.

`/tcb/finish first pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish first`→P. 172.

`/tcb/finish middle pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish middle`→P. 172.

`/tcb/finish last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish last`→P. 172.

`/tcb/finish broken pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish broken`→P. 172.

`/tcb/finish unbroken and first pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish unbroken and first`→P. 172.

`/tcb/finish middle and last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish middle and last`→P. 172.

`/tcb/finish unbroken and last pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish unbroken and last`→P. 172.

N 2014-09-19 `/tcb/finish first and middle pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/finish first and middle`→P. 172.

18.8 Skin Code

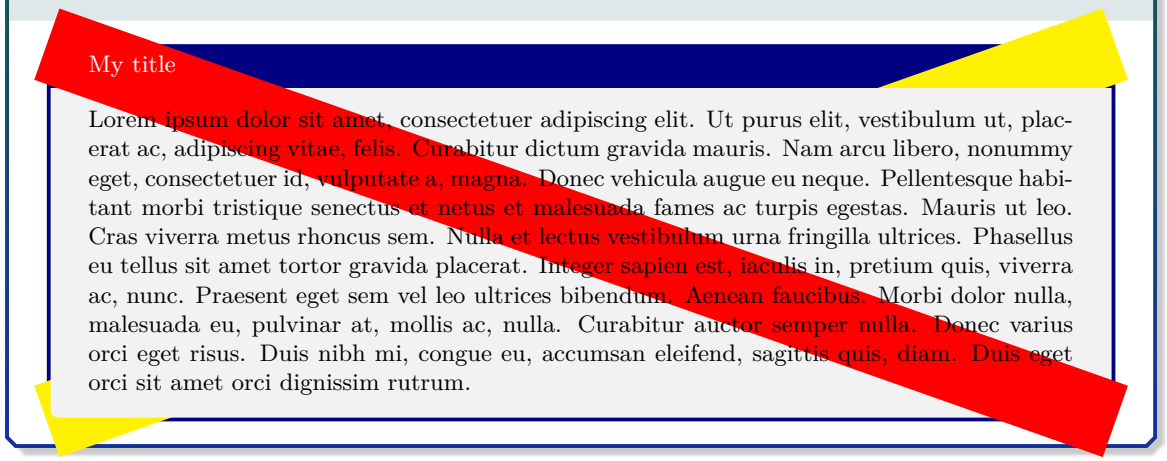
The following option keys extend the options given in Subsection 8.2 from page 116.

`/tcb/frame code app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/frame code`→P. 116.

`/tcb/frame code pre=`⟨*graphical code*⟩ (no default)
Prepends the given ⟨*graphical code*⟩ to `/tcb/frame code`→P. 116.

`/tcb/interior titled code app=`⟨*graphical code*⟩ (no default)
Appends the given ⟨*graphical code*⟩ to `/tcb/interior titled code`→P. 116.

```
\begin{tcolorbox}[title=My title,enhanced,colframe=Navy,
  frame code app={\draw[yellow,line width=1cm] (
    frame.south west)--(frame.north east);},
  interior titled code app={\draw[red,line width=1cm]
    (frame.north west)--(frame.south east);},
  ]
\lipsum[1]
\end{tcolorbox}
```



/tcb/interior titled code pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to **/tcb/interior titled code** →P. 116.

/tcb/interior code app=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to **/tcb/interior code** →P. 117.

/tcb/interior code pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to **/tcb/interior code** →P. 117.

/tcb/segmentation code app=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to **/tcb/segmentation code** →P. 117.

/tcb/segmentation code pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to **/tcb/segmentation code** →P. 117.

/tcb/title code app=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to **/tcb/title code** →P. 118.

/tcb/title code pre=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to **/tcb/title code** →P. 118.

18.9 Extras

The following option keys extend the options given in Section 15.5 on page 312. There are no app type keys since extras are stackable by default.

N 2015-07-16 **/tcb/extras pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras→P. 312.

N 2015-07-16 **/tcb/extras unbroken pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras unbroken→P. 312.

N 2015-07-16 **/tcb/extras first pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras first→P. 312.

N 2015-07-16 **/tcb/extras middle pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras middle→P. 312.

N 2015-07-16 **/tcb/extras last pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras last→P. 312.

N 2015-07-16 **/tcb/extras broken pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras broken→P. 312.

N 2015-07-16 **/tcb/extras unbroken and first pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras unbroken and first→P. 312.

N 2015-07-16 **/tcb/extras middle and last pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras middle and last→P. 312.

N 2015-07-16 **/tcb/extras unbroken and last pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras unbroken and last→P. 312.

N 2015-07-16 **/tcb/extras first and middle pre={⟨*options*⟩}** (no default)

Prepends the given ⟨*options*⟩ to /tcb/extras first and middle→P. 312.

19 Library xparse

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{xparse}
```

This also loads the package `xparse` [11].

The purpose of this library is to give comfortable access to the powerful document command production with `xparse` for `tcolorbox`. See the `xparse` package documentation [11] for details about the argument ⟨*specification*⟩ used in this section.

19.1 Option Keys

/tcb/verbatim (style, no value)

Sets options for a *verbatim* style `\tcbox` → P. 13. Since the indented boxes may contain only very few words, the dimensions are made smaller and `/tcb/nobeforeafter` → P. 76 and `/tcb/tcbox raise base` → P. 91 are set.

```
\DeclareTotalTCBox{\myverb}{ v }{verbatim,
  colframe=red!75!black,colupper=blue}{#1}

\myverb{\textbf} is a \myverb{\LaTeX} command.
```

`\textbf` is a `\LaTeX` command.

/tcb/IfNoValueTF={⟨*argument*⟩}{⟨*true options*⟩}{⟨*false options*⟩} (no default)

Wraps the `\IfNoValueTF` command of `xparse` for option setting. If the ⟨*argument*⟩ has no value, the ⟨*true options*⟩ are set. Otherwise, the ⟨*false options*⟩ are set.

```
\DeclareTColorBox{mybox}{ o }{colframe=red!75!black,
  IfNoValueTF={#1}{colback=red!5!white}{enhanced,interior style image=#1}}

\begin{mybox}
This is a tcolorbox.
\end{mybox}

\begin{mybox}[goldshade.png]
This is a tcolorbox.
\end{mybox}
```

This is a tcolorbox.

This is a tcolorbox.

/tcb/IfValueTF={⟨*argument*⟩}{⟨*true options*⟩}{⟨*false options*⟩} (no default)

Wraps the `\IfValueTF` command of `xparse` for option setting. If the ⟨*argument*⟩ has a value, the ⟨*true options*⟩ are set. Otherwise, the ⟨*false options*⟩ are set.

```
\DeclareTColorBox{mybox}{ o }{colframe=red!75!black,colback=red!5!white,
  IfValueTF={#1}{title={\flqq #1\frqq},fonttitle=\bfseries}{}}

\begin{mybox}
This is a tcolorbox.
\end{mybox}

\begin{mybox}[My title]
This is a tcolorbox.
\end{mybox}
```

This is a tcolorbox.

«My title»

This is a tcolorbox.

/tcb/IfBooleanTF={⟨*argument*⟩}{⟨*true options*⟩}{⟨*false options*⟩} (no default)

Wraps the `\IfBooleanTF` command of `xparse` for option setting. If the ⟨*argument*⟩ is `\BooleanTue`, the ⟨*true options*⟩ are set. If the ⟨*argument*⟩ is `\BooleanFalse`, the ⟨*false options*⟩ are set.

```
\DeclareTColorBox{mybox}{ s }{colframe=red!75!black,
  IfBooleanTF={#1}{colback=yellow!50!red}{colback=red!5!white}}

\begin{mybox}
This is a tcolorbox.
\end{mybox}

\begin{mybox}*
This is a tcolorbox.
\end{mybox}
```

This is a tcolorbox.

This is a tcolorbox.

19.2 Producing tcolorbox Environments and Commands

\DeclareTColorBox[⟨*init options*⟩]{⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Creates a new environment ⟨*name*⟩ based on tcolorbox→ P. 11.

Basically, \DeclareTColorBox operates like \DeclareDocumentEnvironment. This means, the new environment ⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying tcolorbox→ P. 11.

Note that /tcb/savedelimiter→ P. 25 is set to the given ⟨*name*⟩ automatically.

The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

The new environment is always created, irrespective of an already existing environment with the same name.

```
% counter from previous example
\DeclareTColorBox[use counter from=pabox]{mybox}{ O{red} m d"" O{} }
  {enhanced,colframe=#1!75!black,colback=#1!5!white,
   fonttitle=\bfseries,title={\thetcbcounter~#2},
   IfValueTF={#3}{watermark text={#3}}{},#4}

\begin{mybox}{My title}
This is a tcolorbox.
\end{mybox}

\begin{mybox}[blue]{My title}
This is a tcolorbox.
\end{mybox}

\begin{mybox}[green]{My title}"My Watermark"
This is a tcolorbox.
\end{mybox}

\begin{mybox}[yellow]{My title}[colbacktitle=yellow!50!white,coltitle=black]
This is a tcolorbox.
\end{mybox}

\begin{mybox}[purple]{My title}"All together"[coltitle=yellow]
This is a tcolorbox.
\end{mybox}
```

19.1 My title

This is a tcolorbox.

19.2 My title

This is a tcolorbox.

19.3 My title

This is a tcolorbox. My Watermark

19.4 My title

This is a tcolorbox.

19.5 My title

This is a tcolorbox. All together

`\NewTColorBox[⟨init options⟩]{⟨name⟩}{⟨specification⟩}{⟨options⟩}`

Operates like `\DeclareTColorBox` → P. 358, but based on `\NewDocumentEnvironment` instead of `\DeclareDocumentEnvironment`. An error is issued if ⟨*name*⟩ has already been defined.

`\RenewTColorBox[⟨init options⟩]{⟨name⟩}{⟨specification⟩}{⟨options⟩}`

Operates like `\DeclareTColorBox` → P. 358, but based on `\RenewDocumentEnvironment` instead of `\DeclareDocumentEnvironment`. An existing environment is redefined.

`\ProvideTColorBox[⟨init options⟩]{⟨name⟩}{⟨specification⟩}{⟨options⟩}`

Operates like `\DeclareTColorBox` → P. 358, but based on `\ProvideDocumentEnvironment` instead of `\DeclareDocumentEnvironment`. The environment ⟨*name*⟩ is only created if it is not already defined.

\DeclareTotalTColorBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Creates a new command \⟨*name*⟩ based on tcolorbox→P. 11. In contrast to \DeclareTColorBox→P. 358, also the ⟨*content*⟩ of the tcolorbox is specified.

Basically, \DeclareTotalTColorBox operates like \DeclareDocumentCommand. This means, the new command \⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying tcolorbox→P. 11 which is filled with the specified ⟨*content*⟩.

Note that /tcb/savedelimiter→P. 25 is set to the given ⟨*name*⟩ automatically.

The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

The new command is always created, irrespective of an already existing command with the same name.

```
\DeclareTotalTColorBox{\diabox}{ O{} v  m }
  { bicolor,nobeforeafter,equal height group=diabox,width=5.7cm,
    fonttitle=\bfseries\ttfamily,adjusted title={#2},center title,
    colframe=blue!20!black,leftupper=0mm,rightupper=0mm,colback=black!75!white,#1}
  { \tikz\path[fill zoom image={#2}] (0,0) rectangle (\linewidth,4cm);%
    \tcblower#3}

\diabox{blueshade.png}{Created with |GIMP|.\\\url{http://www.gimp.org}}
\diabox{goldshade.png}{Created with |GIMP|.\\\url{http://www.gimp.org}}
```

\NewTotalTColorBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Operates like \DeclareTotalTColorBox, but based on \NewDocumentCommand instead of \DeclareDocumentCommand. An error is issued if \⟨*name*⟩ has already been defined.

\RenewTotalTColorBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Operates like \DeclareTotalTColorBox, but based on \RenewDocumentCommand instead of \DeclareDocumentCommand. An existing command is redefined.

\ProvideTotalTColorBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Operates like \DeclareTotalTColorBox, but based on \ProvideDocumentCommand instead of \DeclareDocumentCommand. The command \⟨*name*⟩ is only created if it is not already defined.

19.3 Producing tcbox Commands

\DeclareTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Creates a new command \⟨*name*⟩ based on \tcbox→ P. 13. Basically, \DeclareTCBox operates like \DeclareDocumentCommand. This means, the new command \⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying \tcbox→ P. 13.

Note that /tcb/savedelimiter→ P. 25 is set to the given ⟨*name*⟩ automatically.

The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

The new command is always created, irrespective of an already existing command with the same name.

```
% counter from previous example
\DeclareTCBox[use counter from=pabox]{\mybox}{ s m s }
{ nobeforeafter,colback=red!5!white,colframe=red!75!black,
  title={#2 (Box \thetcbcounter)},fonttitle=\bfseries,
  IfBooleanTF={#1}{enhanced,drop shadow}{},
  IfBooleanTF={#3}{colbacktitle=red!50!white}{}  }

\mybox{Bird}{This is my first box.}
  \hfill
\mybox*{Tree}{This is my second box.}
  \par\bigskip
\mybox{Bike}*{This is my third box.}
  \hfill
\mybox*{City}*{This is my fourth box.}
```

Bird (Box 19.6)
This is my first box.

Tree (Box 19.7)
This is my second box.

Bike (Box 19.8)
This is my third box.

City (Box 19.9)
This is my fourth box.

\NewTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBox, but based on \NewDocumentCommand instead of \DeclareDocumentCommand. An error is issued if \⟨*name*⟩ has already been defined.

\RenewTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBox, but based on \RenewDocumentCommand instead of \DeclareDocumentCommand. An existing command is redefined.

\ProvideTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBox, but based on \ProvideDocumentCommand instead of \DeclareDocumentCommand. The command \⟨*name*⟩ is only created if it is not already defined.

`\DeclareTotalTCBox[⟨init options⟩]{\⟨name⟩}{⟨specification⟩}{⟨options⟩}{⟨content⟩}`

Creates a new command \⟨*name*⟩ based on `\tcbox` → P. 13. In contrast to `\DeclareTCBox` → P. 361, also the ⟨*content*⟩ of the `tcbox` is specified.

Basically, `\DeclareTotalTCBox` operates like `\DeclareDocumentCommand`. This means, the new command \⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying `\tcbox` → P. 13 which is filled with the specified ⟨*content*⟩.

Note that `/tcb/savedelimiter` → P. 25 is set to the given ⟨*name*⟩ automatically.

The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

The new command is always created, irrespective of an already existing command with the same name.

```
\DeclareTotalTCBox{\myverb}{ O{red} v O{} }
{ fontupper=\ttfamily,nobeforeafter,tcbox raise base,arc=0pt,outer arc=0pt,
  top=0pt,bottom=0pt,left=0mm,right=0mm,
  leftrule=0pt,rightrule=0pt,toprule=0.3mm,bottomrule=0.3mm,boxsep=0.5mm,
  colback=#1!10!white,colframe=#1!50!black,#3}{#2}

To set a word \textbf{bold} in \myverb{\LaTeX}, use
\myverb[green]{\textbf{bold}}. Alternatively, write
\myverb[yellow]{{\bfseries bold}}.
In \myverb[blue]{\LaTeX}[enhanced,fuzzy halo], other font settings are
done in the same way, e.\,g. \myverb{\textit}, \myverb{\itshape}\\
or \myverb[brown]{\texttt}, \myverb[brown]{\ttfamily}.
```

To set a word **bold** in `\LaTeX`, use `\textbf{bold}`. Alternatively, write `{\bfseries bold}`. In `\LaTeX`, other font settings are done in the same way, e. g. `\textit`, `\itshape` or `\texttt`, `\ttfamily`.

The next example uses `\lstinline` from the `listings` package to typeset the verbatim content.

```
% \usepackage{listings} or \tcbuselibrary{listings}
\DeclareTotalTCBox{\commandbox}{ s v }
{verbatim,colupper=white,colback=black!75!white,colframe=black}
{\IfBooleanTF{#1}{\textcolor{red}{\ttfamily\bfseries > }}{}%
  \lstinline[language=command.com,keywordstyle=\color{blue!35!white}\bfseries]^#2^}

\commandbox*{cd "My Documents"} changes to directory \commandbox{My Documents}.

\commandbox*{dir /A} lists the directory content.

\commandbox*{copy example.txt d:\target} copies \commandbox{example.txt} to
  \commandbox{d:\target}.
```

`> cd "My␣Documents"` changes to directory `My Documents`.
`> dir /A` lists the directory content.
`> copy example.txt d:\target` copies `example.txt` to `d:\target`.

\NewTotalTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Operates like \DeclareTotalTCBox→ P. 362, but based on \NewDocumentCommand instead of \DeclareDocumentCommand. An error is issued if \⟨*name*⟩ has already been defined.

\RenewTotalTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Operates like \DeclareTotalTCBox→ P. 362, but based on \RenewDocumentCommand instead of \DeclareDocumentCommand. An existing command is redefined.

\ProvideTotalTCBox[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Operates like \DeclareTotalTCBox→ P. 362, but based on \ProvideDocumentCommand instead of \DeclareDocumentCommand. The command \⟨*name*⟩ is only created if it is not already defined.

\tcboxverb[⟨*options*⟩]{⟨*verbatim box content*⟩}

Creates a colored box based on \tcbox→ P. 13 which is fitted to the width of the given ⟨*verbatim box content*⟩. The underlying \tcbox→ P. 13 is styled with /tcb/verbatim→ P. 356 plus the given ⟨*options*⟩. The difference to \tcbox→ P. 13 is that the ⟨*verbatim box content*⟩ is interpreted *verbatim*. Therefore, \tcboxverb acts similar to \verb.

```
\tcboxverb{\LaTeX}, \tcboxverb[colback=blue!10!white,colupper=blue]{\LaTeX},
\tcboxverb[blank,fuzzy halo]{\LaTeX}, \tcboxverb[beamer]{\LaTeX},
\tcboxverb[enhanced,skin=enhancedmiddle jigsaw,colframe=red]{\LaTeX}.
```

\LaTeX, \LaTeX, \LaTeX, \LaTeX, \LaTeX.

19.4 Producing tcblisting Environments

The following commands need the LIB listings library to be included.

\DeclareTCBListing[⟨*init options*⟩]{⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Creates a new environment ⟨*name*⟩ based on tcblisting→ P. 243.
Basically, \DeclareTCBListing operates like \DeclareDocumentEnvironment. This means, the new environment ⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying tcblisting→ P. 243.
Note that /tcb/savedelimiter→ P. 25 is set to the given ⟨*name*⟩ automatically.
The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.
The new environment is always created, irrespective of an already existing environment with the same name.

```
\DeclareTCBListing{mybox}{ s O{} m }{%
  colback=red!5!white,
  colframe=red!75!black,
  fonttitle=\bfseries,
  IfBooleanTF={#1}
    {listing side text}
    {text side listing},
  title=#3,#2}

\begin{mybox}{Listing Box}
This is my
\LaTeX\ box.
\end{mybox}
\bigskip

\begin{mybox}*{Listing Box}
This is my
\LaTeX\ box.
\end{mybox}
\bigskip

\begin{mybox}[colback=yellow]
  {Listing Box}
This is my
\LaTeX\ box.
\end{mybox}
```

Listing Box

This is my LaTeX box. | This is my \LaTeX\ box.

Listing Box

This is my \LaTeX\ box. | This is my LaTeX box.

Listing Box

This is my LaTeX box. | This is my \LaTeX\ box.

\NewTCBListing[⟨*init options*⟩]{⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBListing, but based on \NewDocumentEnvironment instead of \DeclareDocumentEnvironment. An error is issued if ⟨*name*⟩ has already been defined.

\RenewTCBListing[⟨*init options*⟩]{⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBListing, but based on \RenewDocumentEnvironment instead of \DeclareDocumentEnvironment. An existing environment is redefined.

\ProvideTCBListing[⟨*init options*⟩]{⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBListing, but based on \ProvideDocumentEnvironment instead of \DeclareDocumentEnvironment. The environment ⟨*name*⟩ is only created if it is not already defined.

Caveats of using an environment ending with an optional argument

```
\DeclareTCBListing{mybox}{ O{} }{listing only,#1}

\begin{mybox}[colframe=red]
\good
\end{mybox}

\begin{mybox}[colframe=red]\good\end{mybox}

\begin{mybox}
\good
\end{mybox}

\begin{mybox} \good\end{mybox}

\begin{mybox}\bad!\end{mybox}

\begin{mybox}
[\good]
\end{mybox}

\begin{mybox} [\good]\end{mybox}

\begin{mybox}[\bad!]\end{mybox}
```

19.5 Producing tcbinputlisting Commands

The following commands need the LIB listings library to be included.

\DeclareTCBInputListing[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Creates a new command \⟨*name*⟩ based on \tcbinputlisting → P. 245. Basically, \DeclareTCBInputListing operates like \DeclareDocumentCommand. This means, the new command \⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying \tcbinputlisting → P. 245.
The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.
The new command is always created, irrespective of an already existing command with the same name.

```
% counter from previous example
\DeclareTCBInputListing[use counter from=pabox]{\mylisting}{ O{} O{red} m }{%
  listing file={#3},title=Listing~\thetcbcounter,
  colback=#2!5!white,colframe=#2!50!black,colbacktitle=#2!75!black,
  fonttitle=\bfseries,listing only,#1}

\mylisting[before upper=\textit{This is the included file content:}]
  [blue]{\jobname.tcbtemp}
```

Listing 19.10

This is the included file content:

```
% counter from previous example
\DeclareTCBInputListing[use counter from=pabox]{\mylisting}{ O{}
    O{red} m }{%
  listing file={#3},title=Listing~\thetcbcounter,
  colback=#2!5!white,colframe=#2!50!black,colbacktitle=#2!75!black,
  fonttitle=\bfseries,listing only,#1}

\mylisting[before upper=\textit{This is the included file content:}]
  [blue]{\jobname.tcbtemp}
```

\NewTCBInputListing[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBInputListing, but based on \NewDocumentCommand instead of \DeclareDocumentCommand. An error is issued if \⟨*name*⟩ has already been defined.

\RenewTCBInputListing[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBInputListing, but based on \RenewDocumentCommand instead of \DeclareDocumentCommand. An existing command is redefined.

\ProvideTCBInputListing[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBInputListing, but based on \ProvideDocumentCommand instead of \DeclareDocumentCommand. The command \⟨*name*⟩ is only created if it is not already defined.

19.6 Producing tboxfit Commands

The following commands need the LIB fitting library to be included.

\DeclareTCBoxFit[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

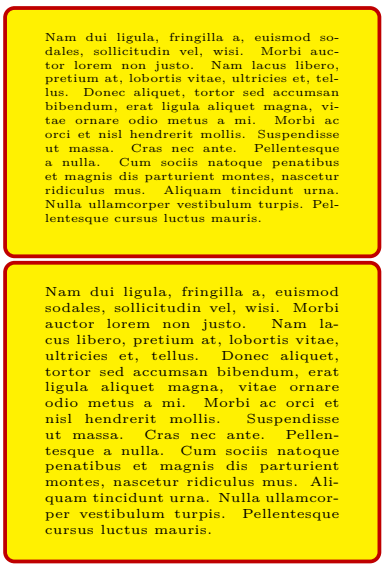
Creates a new command \⟨*name*⟩ based on \tcboxfit → P. 334. Basically, \DeclareTCBoxFit operates like \DeclareDocumentCommand. This means, the new command \⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying \tcboxfit → P. 334.

Note that /tcb/savedelimiter → P. 25 is set to the given ⟨*name*⟩ automatically.

The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

The new command is always created, irrespective of an already existing command with the same name.

```
% \usepackage{lipsum}

\DeclareTCBoxFit{\mybox}{ O{} m o }
 {colback=red!5!white,
  colframe=red!75!black,
  width=#2,height=#2/3*2,
  IfValueTF={#3}{height=#3}{},
  #1}

\mybox[colback=yellow]{5cm}%
  {\lipsum[2]}

\mybox[colback=yellow]{5cm}[4cm]{\lipsum[2]}
```



\NewTCBoxFit[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBoxFit, but based on \NewDocumentCommand instead of \DeclareDocumentCommand. An error is issued if \⟨*name*⟩ has already been defined.

\RenewTCBoxFit[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBoxFit, but based on \RenewDocumentCommand instead of \DeclareDocumentCommand. An existing command is redefined.

\ProvideTCBoxFit[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}

Operates like \DeclareTCBoxFit, but based on \ProvideDocumentCommand instead of \DeclareDocumentCommand. The command \⟨*name*⟩ is only created if it is not already defined.

\DeclareTotalTCBoxFit[⟨*init options*⟩]{\⟨*name*⟩}{⟨*specification*⟩}{⟨*options*⟩}{⟨*content*⟩}

Creates a new command \⟨*name*⟩ based on \tcboxfit→ P. 334. In contrast to \DeclareTCBoxFit→ P. 367, also the ⟨*content*⟩ of the tcboxfit is specified.

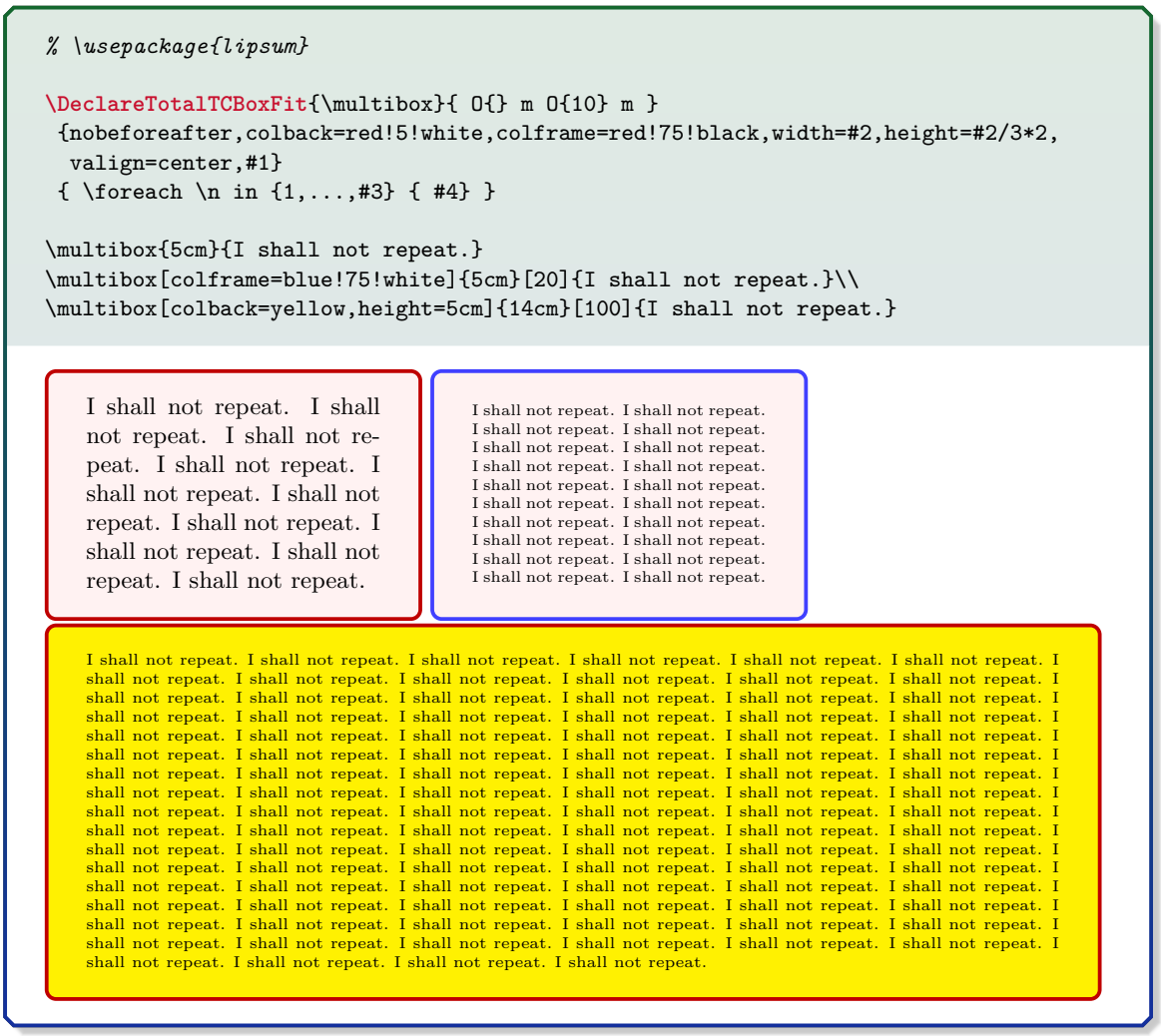
Basically, \DeclareTotalTCBoxFit operates like \DeclareDocumentCommand. This means, the new command \⟨*name*⟩ is constructed with the given argument ⟨*specification*⟩. The ⟨*options*⟩ are given to the underlying \tcboxfit→ P. 334 which is filled with the specified ⟨*content*⟩.

Note that /tcb/savedelimiter→ P. 25 is set to the given ⟨*name*⟩ automatically.

The ⟨*init options*⟩ allow setting up automatic numbering, see Section 5 from page 97.

The new command is always created, irrespective of an already existing command with the same name.

```
% \usepackage{lipsum}

\DeclareTotalTCBoxFit{\multibox}{ O{} m O{10} m }
 {nobeforeafter,colback=red!5!white,colframe=red!75!black,width=#2,height=#2/3*2,
  valign=center,#1}
 { \foreach \n in {1,...,#3} { #4} }

\multibox{5cm}{I shall not repeat.}
\multibox[colframe=blue!75!white]{5cm}[20]{I shall not repeat.}\\
\multibox[colback=yellow,height=5cm]{14cm}[100]{I shall not repeat.}
```


I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat.
I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat. I shall not repeat.

20 Library LIB external

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{external}
```

The purpose of this library is to support externalization of document snippets like graphics or boxes which can be compiled stand-alone. These snippets are written to external files, compiled and the resulting pdf files are included to the main document as images. The whole procedure saves compilation time, if such a snippet is costly to compile but needs to compile just once or very seldom.

There are very good alternatives to this library. One should consider the `standalone` package or the Ti*k*Z externalization library instead. The LIB external library is something in between and can be seen as poor man variant of the Ti*k*Z externalization library.

The main differences between Ti*k*Z externalization and LIB external are:

- Ti*k*Z `external` compiles the whole original document in a sophisticated way while LIB external uses only the preamble or a part of the preamble of the original document.
- Ti*k*Z `external` can automatically externalize all `tikzpicture` environments while LIB external externalizes marked snippets only.
- Code snippets to be externalized by LIB external are not restricted to `tikzpicture` environments. But these snippets have to be stand-alone without dependencies to the rest of the document.

Why should somebody use LIB external instead of the more powerful Ti*k*Z `external`? One reason could be compilation speed, but the main reason for creating the library at all was that Ti*k*Z `external` tends to choke on complicated documents where the sophisticated mechanism stumbles. Since LIB external does not use the original document body for compilation, this cannot happen.

Source snippets are compiled, if their `md5` checksum has changed. They are not compiled automatically, if option settings are changed or anything outside the snippet is changed. Use `/tcb/external/force remake` → P. 370 to force compilation in this case or simply delete the externalized snippets.

To use the externalization options, the compiler has to be called with the `-shell-escape` permission to authorize potentially dangerous system calls. Be warned that this is a security risk.

20.1 Preparation of a Document for Externalization

The preamble of the main document has to contain the \tcbEXTERNALIZE command. Without this command, no externalization operation will be executed.

N 2015-03-11 **\tcbEXTERNALIZE**

It is mandatory for externalization that this command is used once in the preamble of the main document. Every setting *before* \tcbEXTERNALIZE will also be used for compiling an external snippet. Every setting *after* \tcbEXTERNALIZE will be ignored for compiling an external snippet. Place this command right before \begin{document}, if you are not absolutely sure about another place.
The main document has to look like the following:

```
\documentclass[a4paper]{book}%   for example
\usepackage{...}%                 anything
% ...
% Tpyically, all or the very most settings for the document.

\tcbEXTERNALIZE% Typically, just before \begin{document}

% Additional settings which are ABSOLUTELY irrelevant for the
% stand-alone snippets.
%
\begin{document}
  % The document.
  % This also contains the marked snippets for externalization.
\end{document}
```

During compilation, a /tcb/external/runner file is dynamically created (several times). This is the actual main file for compiling an externalized snippet.

N 2015-03-11 **/tcb/external/runner=⟨*file name*⟩** (no default, initially \jobname_run.tex)

Sets the ⟨*file name*⟩ for dynamically created runner file. This is the actual main file for a document snippet. Typically, the initial setting is not needed to be changed.

```
\tcbset{external/runner=myrunner.tex}
```

N 2015-03-11 **/tcb/external/prefix=⟨*text*⟩** (no default, initially external/)

The ⟨*text*⟩ is prefixed to any /tcb/external/name → P. 372 for an externalization snippet. The initial setting implies saving all snippets into an external/ subdirectory. Depending on the operation system, the subdirectory may have to be created manually once.

```
% Use a 'real' prefix instead of writing into a subdirectory:
\tcbset{external/prefix=ext_}
```

N 2015-03-11 **/tcb/external/externalize=true|false** (default true, initially true)

If set to true, the marked snippets are compiled if necessary. If set to false, the marked snippets are not compiled but included as text. /tcb/external/externalize can only be used after \tcbEXTERNALIZE.

N 2015-03-11 **/tcb/external/force remake=true|false** (default true, initially false)

If set to true, the marked snippets are always compiled. If set to true, the marked snippets are compiled only if necessary. The necessity is given, if a compiled pdf file is missing or the md5 checksum of the source snippet has changed.

N 2015-03-11 **/tcb/external/!** (style)

Shortcut for setting /tcb/external/force remake to true.

N 2015-06-12 **/tcb/external/-** (style)

Shortcut for setting /tcb/external/externalize to false.

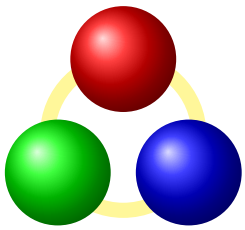
20.2 Marking Externalization Snippets

N 2015-03-11

```
\begin{tcbexternal}[⟨options⟩]{⟨name⟩}
 ⟨environment content⟩
\end{tcbexternal}
```

Marks the environment content as a snippet for externalization. Typically, the content is a `tikzpicture` or something similar. It is important to note that the snippet should not have any dependencies with the rest of the document, e.g. referencing counters or setting counters is not possible. The ⟨*name*⟩ is automatically prefixed with `/tcb/external/prefix` → P. 370. In combination, this has to be a unique file name. It is advised to not use spaces or umlauts for the name. The ⟨*options*⟩ are keys from the `/tcb/external/` key tree.

```
\begin{tcbexternal}{example_tikzpicture}
  \begin{tikzpicture}
    \path[fill=yellow!50!white] (0,0) circle (11mm);
    \path[fill=white] (0,0) circle (9mm);
    \foreach \w/\c in {90/red,210/green,330/blue}
      {\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
  \end{tikzpicture}
\end{tcbexternal}
```



If a `tcolorbox` → P. 11 is externalized, one should use `/tcb/nobeforeafter` → P. 76 for the box. Indention and distances to the text before and after have to be given separately outside the `tcbexternal` environment.

```
\noindent%
\begin{tcbexternal}[minipage]{example_tcolorbox}
  \begin{tcolorbox}[nobeforeafter,enhanced,
      fonttitle=\bfseries,title=Externalized Box,
      colframe=red!50!black,drop fuzzy shadow,
      interior style={fill overzoom image=goldshade.png}]
    This complete tcolorbox is externalized. One cannot use numbered
    boxes here. Note the \texttt{minipage} option which tells the
    current line width to the external snippet.
  \end{tcolorbox}
\end{tcbexternal}
```

Externalized Box

This complete tcolorbox is externalized. One cannot use numbered boxes here. Note the `minipage` option which tells the current line width to the external snippet.

```
\begin{tcolorbox}[nobeforeafter,enhanced,
      fonttitle=\bfseries,title=Externalized Box,
      colframe=blue!50!black,
      interior style={fill overzoom image=blueshade.png}]
  \begin{tcbexternal}[minipage]{example_tcolorbox2}
    \color{white}%
    The interior of the tcolorbox is externalized.
    One can use numbered boxes without problem.
    Note that the text color has to be set for the text manually
    since it is converted into an image.
  \end{tcbexternal}
\end{tcolorbox}
```

Externalized Box

The interior of the tcolorbox is externalized. One can use numbered boxes without problem. Note that the text color has to be set for the text manually since it is converted into an image.

```
\begin{tcbexternal}[minipage]{example_tabularx}
  \newcolumntype{Y}{>{\raggedleft\arraybackslash}X}%
  \begin{tabularx}{\linewidth}{|l||Y|Y|Y|Y||Y|}\hline
    Group & One & Two & Three & Four & Sum\\\hline\hline
    Red & 1000.00 & 2000.00 & 3000.00 & 4000.00 & 10000.00\\\hline
    Green & 2000.00 & 3000.00 & 4000.00 & 5000.00 & 14000.00\\\hline
    Blue & 3000.00 & 4000.00 & 5000.00 & 6000.00 & 18000.00\\\hline\hline
    Sum & 6000.00 & 9000.00 & 12000.00 & 15000.00 & 42000.00\\\hline
  \end{tabularx}
\end{tcbexternal}
```

Group	One	Two	Three	Four	Sum
Red	1000.00	2000.00	3000.00	4000.00	10000.00
Green	2000.00	3000.00	4000.00	5000.00	14000.00
Blue	3000.00	4000.00	5000.00	6000.00	18000.00
Sum	6000.00	9000.00	12000.00	15000.00	42000.00

N 2015-03-11 `/tcb/external/name=`⟨*name*⟩ (no default, initially `unnamed`)

The ⟨*name*⟩ is automatically prefixed with `/tcb/external/prefix` → P. 370. In combination, this has to be a unique file name for externalization. Typically, this key is not used directly but is set indirectly as mandatory parameter, see `tcbexternal` → P. 371.

N 2015-03-11

```
\begin{extcolorbox}[⟨options⟩]{⟨name⟩}[⟨tcolorbox options⟩]
 ⟨environment content⟩
\end{extcolorbox}
```

This is an externalized version of `tcolorbox` → P. 11 created using `\newtcbexternalizetcolorbox` → P. 378:

```
\newtcbexternalizetcolorbox{extcolorbox}{tcolorbox}{}{}
```

⟨*options*⟩ and ⟨*name*⟩ are given to the underlying `tcbexternal` → P. 371 environment, while ⟨*tcolorbox options*⟩ are given to `tcolorbox` → P. 11.

Note that you should not redefine `/tcb/before` → P. 76 and `/tcb/after` → P. 76 inside the ⟨*tcolorbox options*⟩, since the externalized version would not be identical to the non-externalized otherwise.

```
\begin{extcolorbox}[minipage]{example_extcolorbox}
  [ enhanced,colframe=red!50!black,colback=yellow!10,
    fonttitle=\bfseries,drop fuzzy shadow,
    title=My external box ]

  This box is completely externalized.

  \begin{tcolorbox}[colframe=blue,colback=blue!5,before skip=6pt]
  Inner box.
  \end{tcolorbox}
\end{extcolorbox}
```

My external box

This box is completely externalized.

Inner box.

- **Never** externalize numbered boxes.
- **Never** externalize boxes which contain references to other things, e.g. using `\ref` or `\cite`.
- **Never** externalize breakable boxes.

N 2015-03-11

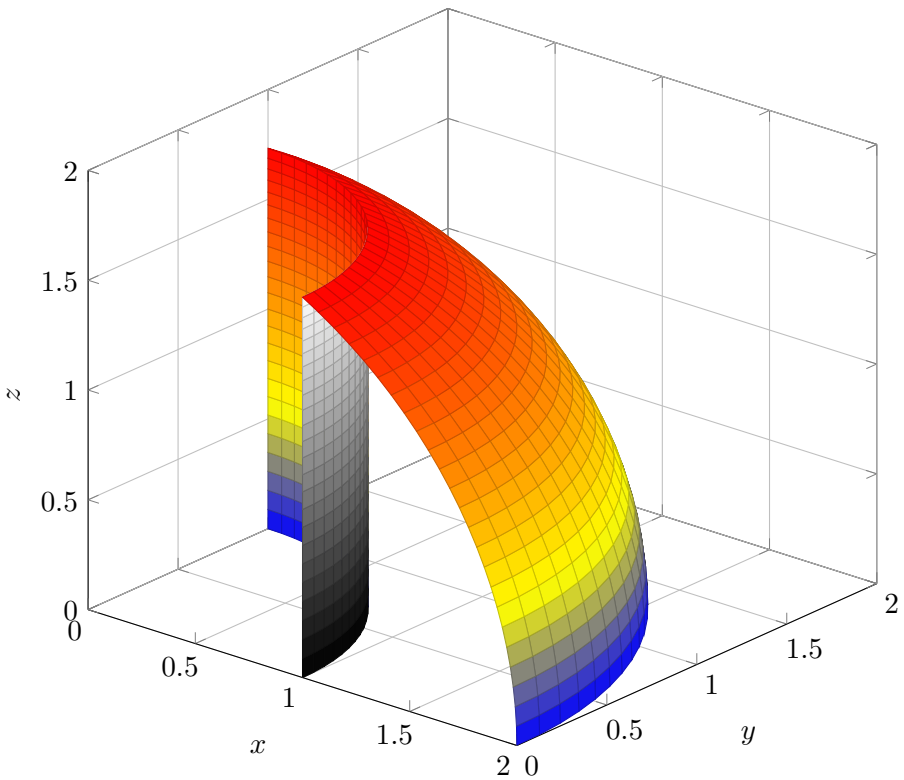
```
\begin{extikzpicture}[⟨options⟩]{⟨name⟩}[⟨tikz options⟩]
 ⟨environment content⟩
\end{extikzpicture}
```

This is an externalized version of `tikzpicture` created using `\newtcbexternalizeenvironment` → P. 378:

```
\newtcbexternalizeenvironment{extikzpicture}{tikzpicture}{}{}{}
```

⟨*options*⟩ and ⟨*name*⟩ are given to the underlying `tcbexternal` → P. 371 environment, while ⟨*tikz options*⟩ are given to `tikzpicture`.

```
\begin{center}
\begin{extikzpicture}[
  preamble={\usepackage{pgfplots}},  % add package for external graph
  input source on error=false,       % do not load source on error
]{example_pgfplots}
  \pgfplotsset{width=12cm}
  \begin{axis}[3d box=background,grid=major,
    xlabel=$x$, ylabel=$y$, zlabel=$z$, view/h=40,
    mesh/interior colormap name=hot,
    colormap/blackwhite,
    z buffer=sort,domain=0:90,y domain=0:60,
    zmin=0,zmax=2,z post scale=1.2,
    ]
  \addplot3[surf,mesh/interior colormap name=blackwhite,
    colormap/hot,] ( {cos(x)},{sin(x)}, {2*sin(y)} );
  \addplot3[surf] ( {2*cos(x)*cos(y)},{2*sin(x)*cos(y)}, {2*sin(y)} );
  \end{axis}
\end{extikzpicture}
\end{center}
```

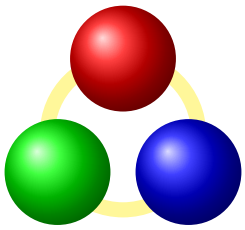


N 2015-03-11 **/tcb/externalize listing=⟨*name*⟩** (style, no default)

The text content of a tcblisting → P. 243 is externalized with the given ⟨*name*⟩. Note that the listing part is not externalized.

```
\begin{tcblisting}{externalize listing=example_listing,
  bicolor,colback=yellow!10,colframe=yellow!50!black,
  colbacklower=white,center lower}
\begin{tikzpicture}
  \path[fill=yellow!50!white] (0,0) circle (11mm);
  \path[fill=white] (0,0) circle (9mm);
  \foreach \w/\c in {90/red,210/green,330/blue}
    {\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{tcblisting}
```

```
\begin{tikzpicture}
  \path[fill=yellow!50!white] (0,0) circle (11mm);
  \path[fill=white] (0,0) circle (9mm);
  \foreach \w/\c in {90/red,210/green,330/blue}
    {\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
```



N 2015-03-11 **/tcb/externalize listing!=⟨*name*⟩** (style, no default)

Combination of /tcb/externalize listing and /tcb/external/force remake → P. 370.

N 2015-03-11 **/tcb/externalize example=⟨*name*⟩** (style, no default)

The text content of a dispExample* → P. 385 is externalized with the given ⟨*name*⟩. Note that the listing part is not externalized.

```
\begin{dispExample*}{sidebyside,externalize example=example_example}
\tikz\path[shading=ball,
  ball color=red] circle (7mm);
\end{dispExample*}
```

```
\tikz\path[shading=ball,
  ball color=red] circle (7mm);
```

N 2015-03-11 **/tcb/externalize example!=⟨*name*⟩** (style, no default)

Combination of /tcb/externalize example and /tcb/external/force remake → P. 370.

20.3 Customization

N 2015-03-11 **/tcb/external/safety=⟨*length*⟩** (no default, initially `2mm`)

The snippet box is surrounded with a safety border with a thickness of ⟨*length*⟩. This border is automatically trimmed during picture inclusion. The reason for this mechanism is to catch box content which extrudes over the bounding box. For example, shadows of a `tcolorbox` are painted outside the bounding box and would be lost otherwise.

N 2015-03-11 **/tcb/external/environment=⟨*env*⟩** (no default, initially unset)

Surrounds the exported snippet text with an environment ⟨*env*⟩ without parameters. Note that this option is ignored for `/tcb/externalize listing`→ P. 375.

N 2015-05-05 **/tcb/external/environment with percent=true|false** (default `true`, initially `true`)

If set to `true`, the `\begin` and `\end` code of `/tcb/external/environment` is appended with a percent sign. For verbatim environments, this option typically has to be se to `false`.

N 2015-03-11 **/tcb/external/minipage=⟨*length*⟩** (default `\linewidth`, initially unset)

Surrounds the exported snippet text with a minipage. The optional ⟨*length*⟩ parameter sets the width of the minipage. Note that the default width is the current line width of the main document. See `tcbexternal`→ P. 371 for examples. Note that this option is ignored for `/tcb/externalize listing`→ P. 375.

N 2015-03-11 **/tcb/external/plain** (no value, initially set)

Removes any text which was set to surround the snippet. This removes the setting of `/tcb/external/minipage`, but is independent of `/tcb/external/safety`.

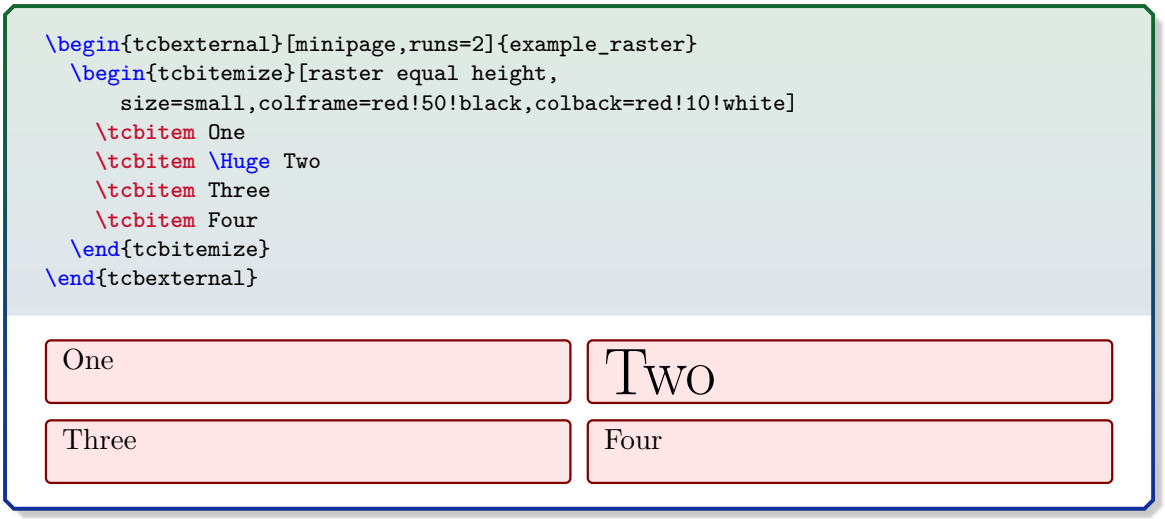
N 2015-03-11 **/tcb/external/compiler=⟨*text*⟩** (no default, initially `pdflatex`)

Sets the name of the compiler for the snippets. Note that this compiler has to support the `\pdfmdfivesum` primitive e.g. using the `pdftexcmds` package. This should work for `xelatex` and `lualatex`.

N 2015-03-11 **/tcb/external/runs=⟨*number*⟩** (no default, initially `1`)

Sets the number of compiler runs for the snippet.

```
\begin{tcbexternal}[minipage,runs=2]{example_raster}
  \begin{tcbitemize}[raster equal height,
      size=small,colframe=red!50!black,colback=red!10!white]
    \tcbitem One
    \tcbitem \Huge Two
    \tcbitem Three
    \tcbitem Four
  \end{tcbitemize}
\end{tcbexternal}
```



N 2015-03-11 **/tcb/external/input source on error=true|false** (default `true`, initially `true`)

If set to `true`, the source code of the snippet is loaded instead of the failed pdf picture. Typically, this will lead to an error stop at the faulty place of the source and such helps detecting the cause. If the source input compiles without error, the document setup may be incorrect, see Section 20.1 on page 370. Maybe, the `external/` subdirectory has to be created manually in this case, see `/tcb/external/prefix`→ P. 370.

If the option is set to `false`, the compilation stops immediately on an error. The log file of the external snippet has to be consulted for error messages in this case.

N 2015-05-05 **/tcb/external/preclass=⟨*code*⟩** (no default, initially unset)

The given ⟨*code*⟩ is added before the snippet document. Typically, this means before `\documentclass`. This is not used for compilation of the main document.

N 2015-05-05 **/tcb/external/PassOptionsToPackage={⟨*options*⟩}{⟨*package*⟩}** (no default, initially unset)

The given ⟨*options*⟩ are passed to the given ⟨*package*⟩ for the snippet document. This is a shortcut for using `/tcb/external/preclass` with `\PassOptionsToPackage`. This not used for compilation of the main document.

N 2015-05-05 **/tcb/external/PassOptionsToClass={⟨*options*⟩}{⟨*class*⟩}** (no default, initially unset)

The given ⟨*options*⟩ are passed to the given ⟨*class*⟩ for the snippet document. This is a shortcut for using `/tcb/external/preclass` with `\PassOptionsToClass`. This not used for compilation of the main document.

N 2015-05-05 **/tcb/external/clear preclass** (no value)

Removes all additional `/tcb/external/preclass` settings.

N 2015-03-11 **/tcb/external/preamble=⟨*code*⟩** (no default, initially unset)

The given ⟨*code*⟩ is added to the preamble of the snippet document. This is not used for compilation of the main document.

N 2015-05-05 **/tcb/external/preamble tcbset=⟨*options*⟩** (no default, initially unset)

The given ⟨*options*⟩ are added as parameter for `\tcbset` → P. 12 to the preamble of the snippet document. This are not used for compilation of the main document.

N 2015-03-16 **/tcb/external/clear preamble** (no value)

Removes all additional `/tcb/external/preamble` settings.

N 2015-03-11 **\tcbifexternal{⟨*true*⟩}{⟨*false*⟩}**

Expands to ⟨*true*⟩, if executed during snippet compilation, and to ⟨*false*⟩, if executed during main document compilation. This can be used *before* `\tcbEXTERNALIZE` → P. 370 to give different setting to snippet and main document.

```
\tcbifexternal{
  \usepackage{onlyforexternal}
}{
  \usepackage{onlyformain}
}
```

N 2015-03-11 **\newtcbexternalizeenvironment**{⟨*newenv*⟩}{⟨*env*⟩}{⟨*options*⟩}{⟨*begin*⟩}{⟨*end*⟩}

Creates a new environment ⟨*newenv*⟩ which is based on tcbexternal^{→ P. 371}. This enviroment takes *at least* one optional parameter and one mandatory parameter. These two parameters are passed to tcbexternal^{→ P. 371}. Further, the given ⟨*options*⟩ are always added to the option list of tcbexternal^{→ P. 371}.
The environment content is externalized and the external snippet is surrounded by an environment ⟨*env*⟩. All further parameters of ⟨*newenv*⟩ are given to ⟨*env*⟩ as parameters. The included image is prepended by ⟨*begin*⟩ and appended by ⟨*end*⟩.
extikzpicture^{→ P. 374} is an example application for \newtcbexternalizeenvironment.

```
\newtcbexternalizeenvironment{extabular}{tabular}{}{\par\centering}{\par}

\begin{extabular}{example_tabular}{|l|p{6cm}|r|}\hline
A & B & C\\\hline
a & This table is externalized as snippet. Obviously,
  this only makes sense for highly complex tables.
& b\\\hline
\end{extabular}
```

A	B	C
a	This table is externalized as snippet. Obviously, this only makes sense for highly complex tables.	b

N 2015-03-11 **\renewtcbexternalizeenvironment**{⟨*newenv*⟩}{⟨*env*⟩}{⟨*options*⟩}{⟨*begin*⟩}{⟨*end*⟩}

Identical to \newtcbexternalizeenvironment, but the environment ⟨*newenv*⟩ is created by \renewenvironment instead of \newenvironment.

N 2015-03-11 **\newtcbexternalizetcolorbox**{⟨*newenv*⟩}{⟨*env*⟩}{⟨*options*⟩}{⟨*begin end options*⟩}

Creates a new environment ⟨*newenv*⟩ which is based on tcbexternal^{→ P. 371}. This enviroment takes *at least* one optional parameter and one mandatory parameter. These two parameters are passed to tcbexternal^{→ P. 371}. Further, the given ⟨*options*⟩ are always added to the option list of tcbexternal^{→ P. 371}.
The environment content is externalized and the external snippet is surrounded by an environment ⟨*env*⟩. All further parameters of ⟨*newenv*⟩ are given to ⟨*env*⟩ as parameters. **In contrast to \newtcbexternalizeenvironment, the environment ⟨*env*⟩ is intended to be based on tcolorbox^{→ P. 11} or tcblisting^{→ P. 243}.**
The ⟨*begin end options*⟩ are options for settings the space before and after the included image using /tcb/before^{→ P. 76}, /tcb/before skip^{→ P. 78}, /tcb/after^{→ P. 76}, or /tcb/after skip^{→ P. 78}.

Use the exact identical values for /tcb/before^{→ P. 76} and /tcb/after^{→ P. 76} inside ⟨*begin end options*⟩ as they where used for definition of ⟨*env*⟩! Otherwise, externalized and non-externalized version will have different spacings.

extcolorbox^{→ P. 373} is an example application for\newtcbexternalizetcolorbox.

Definition in the preamble:

```
\newtcblisting{myownlisting}[2][]{
  enhanced,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  colbacktitle=red!50!yellow,before skip=6pt,after skip=6pt,
  title={#2},#1}

\newtcbexternalizetcolorbox{exmyownlisting}{myownlisting}{minipage}%
  {before skip=6pt,after skip=6pt}% same values as for mylisting
```

```
\begin{exmyownlisting}{example_mylisting}% <- name for the external file
  {My externalized example box}
This is my \LaTeX\ box.
\end{exmyownlisting}
```

My externalized example box

This is my \LaTeX\ box.

This is my LaTeX box.

N 2015-03-11 **\renewtcbexternalizetcolorbox**{⟨*newenv*⟩}{⟨*env*⟩}{⟨*options*⟩}{⟨*begin end options*⟩}

Identical to \newtcbexternalizetcolorbox→ P. 378, but the environment ⟨*newenv*⟩ is created by \renewenvironment instead of \newenvironment.

20.4 Troubleshooting and FAQ

- **I use the default settings, but the external subdirectory is not created.**
  Depending on operating system and compiler, an external subdirectory is automatically created or not. If not, create such a directory manually or add the following to your document:

  ```
  \immediate\write18{mkdir external}
  ```

  or

  ```
  \immediate\write18{mkdir -p external}
  ```

  If the combination of /tcb/external/prefix→ P. 370 and chosen snippet name points to another subdirectory than external, this has to be adapted.

- **I use the minted package and I get a cache directory for every externalized snippet.**
  To avoid this problem, there are several ways.

  – If you do not need minted inside the snippet code, you may use \usepackage{minted} *after* \tcbEXTERNALIZE→ P. 370 or use \tcbifexternal→ P. 377 to switch minted off for the external code. If minted is already included by another package, add the following to your preamble:

  ```
  \tcbset{external/PassOptionsToPackage={draft}{minted}}
  ```

  – If minted is needed for the snippet code, caching can be switched off by adding the following to your preamble:

  ```
  \tcbset{external/PassOptionsToPackage={cache=false}{minted}}
  ```

  Alternatively, the cachedir option of minted may be used to redirect the cache.

21 Library documentation

This library has the single purpose to support LaTeX package documentations like this one. Actually, the visual nature follows the approach from Till Tantau's pgf [20] documentation. Typically, this library is assumed to be used in conjunction with the class ltxdoc or alike.

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{documentation}
```

This also loads the library listings, see Section 13 on page 242, the library skins, see Section 9 on page 124, the library xparse , see Section 19 on page 356, and a bunch of packages, namely pifont, marvosym, makeidx, marginnote, refcount, and hyperref.

N 2014-09-19

The package makeidx is loaded only, if \printindex is *not* already defined. Therefore, one can include an alternative to makeidx like imakeidx *before* the library documentation is used.

The package marginnote is loaded only, if \marginnote is *not* already defined.

For UTF-8 support, load:

```
\tcbuselibrary{listingsutf8,documentation}
```

For minted [14] support, load:

```
\tcbuselibrary{documentation,minted}
\tcbset{listing engine=minted}
```

21.1 Macros of the Library

\begin{docCommand}[⟨*options*⟩]{⟨*name*⟩}{⟨*parameters*⟩}
⟨*command description*⟩
\end{docCommand}

Documents a LaTeX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. The given ⟨*options*⟩ are set with \tcbset→ P. 12. This macro takes mandatory or optional ⟨*parameters*⟩. It is automatically indexed and can be referenced with \refCom→ P. 387{⟨*name*⟩}.

```
\begin{docCommand}{foomakedocSubKey}{\marg{name}\marg{key path}}
  Creates a new environment \meta{name} based on \refEnv{docKey} for the
  documentation of keys with the given \meta{key path}.
\end{docCommand}
```

\foomakedocSubKey{⟨*name*⟩}{⟨*key path*⟩}
Creates a new environment ⟨*name*⟩ based on docKey→ P. 381 for the documentation of keys with the given ⟨*key path*⟩.

```
\begin{docCommand}[color definition=blue]{foomakedocSubKey*}%
    {\marg{name}\marg{key path}}
  Creates a new environment \meta{name} based on \refEnv{docKey} for the
  documentation of keys with the given \meta{key path}.
\end{docCommand}
```

\foomakedocSubKey*{⟨*name*⟩}{⟨*key path*⟩}
Creates a new environment ⟨*name*⟩ based on docKey→ P. 381 for the documentation of keys with the given ⟨*key path*⟩.

```
\begin{docCommand*}[⟨options⟩]{⟨name⟩}{⟨parameters⟩}
 ⟨command description⟩
\end{docCommand*}
```

Identical to docCommand→ P. 380, but without index entry.

```
\begin{docEnvironment}[⟨options⟩]{⟨name⟩}{⟨parameters⟩}
 ⟨environment description⟩
\end{docEnvironment}
```

Documents a LaTeX environment with given ⟨*name*⟩. The given ⟨*options*⟩ are set with \tcbset→ P. 12. This environment takes mandatory or optional ⟨*parameters*⟩. It is automatically indexed and can be referenced with \refEnv→ P. 387{⟨*name*⟩}.

```
\begin{docEnvironment}{foocolorbox}{\oarg{options}}
  This is the main environment to create an accentuated colored text box with
  rounded corners and, optionally, two parts.
\end{docEnvironment}
```

```
\begin{foocolorbox}[⟨options⟩]
 ⟨environment content⟩
\end{foocolorbox}
```

This is the main environment to create an accentuated colored text box with rounded corners and, optionally, two parts.

```
\begin{docEnvironment}%
    [doclang/environment content=My content text]%
    {foocolorbox*}{\oarg{options}}
  This is the main environment to create an accentuated colored text box with
  rounded corners and, optionally, two parts.
\end{docEnvironment}
```

```
\begin{foocolorbox*}[⟨options⟩]
 ⟨My content text⟩
\end{foocolorbox*}
```

This is the main environment to create an accentuated colored text box with rounded corners and, optionally, two parts.

```
\begin{docEnvironment*}[⟨options⟩]{⟨name⟩}{⟨parameters⟩}
 ⟨environment description⟩
\end{docEnvironment*}
```

Identical to docEnvironment, but without index entry.

U 2014-09-29

```
\begin{docKey}[⟨key path⟩][⟨options⟩]{⟨name⟩}{⟨parameters⟩}{⟨description⟩}
 ⟨key description⟩
\end{docKey}
```

Documents a key with given ⟨*name*⟩ and an optional ⟨*key path*⟩. The given ⟨*options*⟩ are set with \tcbset→ P. 12. This key takes mandatory or optional ⟨*parameters*⟩ as value with a short ⟨*description*⟩. It is automatically indexed and can be referenced with \refKey→ P. 388{⟨*name*⟩}.

```
\begin{docKey}[foo]{footitle}{=\meta{text}}{no default, initially empty}
  Creates a heading line with \meta{text} as content.
\end{docKey}
```

/foo/footitle=⟨*text*⟩ (no default, initially empty)

Creates a heading line with ⟨*text*⟩ as content.

U 2014-09-29 `\begin{docKey*}[⟨key path⟩][⟨options⟩]{⟨name⟩}{⟨parameters⟩}{⟨description⟩}`
`⟨key description⟩`
`\end{docKey*}`

Identical to `docKey` → P. 381, but without index entry.

`\docValue{⟨name⟩}`

Documents a value with given ⟨*name*⟩. Typically, this is a value for a key. This value is automatically indexed.

```
A feasible value for \refKey{/foo/footitle} is \docValue{foovalue}.
```

A feasible value for `/foo/footitle` → P. 381 is `foovalue`.

`\docValue*{⟨name⟩}`

Identical to `\docValue`, but without index entry.

U 2014-10-09 `\docAuxCommand{⟨name⟩}`

Documents an auxiliary or minor LaTeX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. This macro is automatically indexed.

```
The macro \docAuxCommand{fooaux} holds some interesting data.
```

The macro `\fooaux` holds some interesting data.

U 2014-10-09 `\docAuxCommand*{⟨name⟩}`

Identical to `\docAuxCommand`, but without index entry.

`\docAuxEnvironment{⟨name⟩}`

Documents an auxiliary or minor LaTeX environment with given ⟨*name*⟩. This macro is automatically indexed.

```
The environment \docAuxEnvironment{fooauxenv} holds some interesting data.
```

The environment `fooauxenv` holds some interesting data.

`\docAuxEnvironment*[⟨key path⟩]{⟨name⟩}`

Identical to `\docAuxEnvironment`, but without index entry.

`\docAuxKey[⟨key path⟩]{⟨name⟩}`

Documents an auxiliary key with given ⟨*name*⟩ and an optional ⟨*key path*⟩. It is automatically indexed.

```
The key \docAuxKey[foo]{fooaux} holds some interesting data.
```

The key `/foo/fooaux` holds some interesting data.

`\docAuxKey*[⟨key path⟩]{⟨name⟩}`

Identical to `\docAuxKey`, but without index entry.

N 2015-01-08 **\docCounter{⟨*name*⟩}**

Documents a counter with given ⟨*name*⟩. The counter is automatically indexed.

```
The counter \docCounter{foocounter} can be used for computation.
```

The counter foocounter can be used for computation.

N 2015-01-08 **\docCounter*{⟨*name*⟩}**

Identical to \docCounter, but without index entry.

N 2015-01-08 **\docLength{⟨*name*⟩}**

Documents a counter with given ⟨*name*⟩. The counter is automatically indexed.

```
The length \docLength{foolength} can be used for computation.
```

The length \foolength can be used for computation.

N 2015-01-08 **\docLength*{⟨*name*⟩}**

Identical to \docLength, but without index entry.

\docColor{⟨*name*⟩}

Documents a color with given ⟨*name*⟩. The color is automatically indexed.

```
The color \docColor{foocolor} is available.
```

The color foocolor is available.

\docColor*{⟨*name*⟩}

Identical to \docColor, but without index entry.

\cs{⟨*name*⟩}

Macro from `ltxdoc` [3] to typeset a command word ⟨*name*⟩ where the backslash is prefixed. The library overwrites the original macro.

```
This is a \cs{foocommand}.
```

This is a `\foocommand`.

\meta{⟨*text*⟩}

Macro from `doc` [8] to typeset a meta ⟨*text*⟩. The library overwrites the original macro.

```
This is a \meta{text}.
```

This is a ⟨*text*⟩.

\marg{⟨*text*⟩}

Macro from `ltxdoc` [3] to typeset a ⟨*text*⟩ with curly brackets as a mandatory argument. The library overwrites the original macro.

```
This is a mandatory \marg{argument}.
```

This is a mandatory {⟨*argument*⟩}.

\oarg{⟨*text*⟩}

Macro from `ltxdoc` [3] to typeset a ⟨*text*⟩ with square brackets as an optional argument. The library overwrites the original macro.

```
This is an optional \oarg{argument}.
```

This is an optional [⟨*argument*⟩].

\brackets{⟨*text*⟩}

Sets the given ⟨*text*⟩ with curly brackets.

```
Here we use \brackets{some text}.
```

Here we use {some text}.

U 2014-10-10 `\begin{dispExample}`
⟨*environment content*⟩
`\end{dispExample}`

Creates a colored box based on a `tcolorbox` → P. 11. It displays the environment content as source code in the upper part and as compiled text in the lower part of the box. The appearance is controlled by `/tcb/documentation listing style` → P. 390 and the style `/tcb/docexample` → P. 390. It may be changed by redefining this style.

```
\begin{dispExample}
This is a \LaTeX\ example.
\end{dispExample}
```

This is a \LaTeX\ example.

This is a LaTeX example.

U 2014-10-10 `\begin{dispExample*}`{⟨*options*⟩}
⟨*environment content*⟩
`\end{dispExample*}`

The starred version of `dispExample` takes `tcolorbox` → P. 11 ⟨*options*⟩ as parameter. These ⟨*options*⟩ are executed after `/tcb/docexample` → P. 390.

```
\begin{dispExample*}{sidebyside}
This is a \LaTeX\ example.
\end{dispExample*}
```

This is a \LaTeX\ example.

This is a LaTeX example.

```
\begin{dispListing}
 ⟨environment content⟩
\end{dispListing}
```

Creates a colored box based on a `tcolorbox` → P. 11. It displays the environment content as source code. The appearance is controlled by `/tcb/documentation listing style` → P. 390 and the style `/tcb/docexample` → P. 390. It may be changed by redefining this style.

```
\begin{dispListing}
This is a \LaTeX\ example.
\end{dispListing}
```

```
This is a \LaTeX\ example.
```

```
\begin{dispListing*}{⟨options⟩}
 ⟨environment content⟩
\end{dispListing*}
```

The starred version of `dispListing` takes `tcolorbox` → P. 11 ⟨*options*⟩ as parameter. These ⟨*options*⟩ are executed after `/tcb/docexample` → P. 390.

```
\begin{dispListing*}{title=My listing}
This is a \LaTeX\ example.
\end{dispListing*}
```

My listing

```
This is a \LaTeX\ example.
```

```
\begin{absquote}
 ⟨environment content⟩
\end{absquote}
```

Used to typeset an abstract as quoted and small text.

```
\begin{absquote}
|tcolorbox| provides an environment for colored and framed text boxes with a
heading line. Optionally, such a box can be split in an upper and a lower part.
\end{absquote}
```

> `tcolorbox` provides an environment for colored and framed text boxes with a heading line. Optionally, such a box can be split in an upper and a lower part.

\tcbmakedocSubKey{⟨name⟩}{⟨key path⟩}

Creates a new environment ⟨*name*⟩ based on docKey→ P. 381 for the documentation of keys with the given ⟨*key path*⟩ as default. The new environment ⟨*name*⟩ takes the same parameters as docKey→ P. 381 itself. A second starred environment ⟨*name*⟩ is also created, which is identical to ⟨*name*⟩ but without index entry.

```
\tcbmakedocSubKey{docFooKey}{foo}

\begin{docFooKey}{foodummy}{=\meta{nothing}}{no default, initially empty}
Some key.
\end{docFooKey}

\begin{docFooKey*}{foo another dummy}{=\meta{nothing}}{no default, initially empty}
Some key (not indexed).
\end{docFooKey*}
```

/foo/foodummy=⟨*nothing*⟩ (no default, initially empty)
Some key.

/foo/foo another dummy=⟨*nothing*⟩ (no default, initially empty)
Some key (not indexed).

\refCom{⟨name⟩}

References a documented LaTeX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. The page reference is suppressed if it links to the same page.

```
We have created \refCom{foomakedocSubKey} as an example.
```

We have created \foomakedocSubKey→ P. 380 as an example.

\refCom*{⟨name⟩}

References a documented LaTeX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. There is no page reference.

```
We have created \refCom*{foomakedocSubKey} as an example.
```

We have created \foomakedocSubKey as an example.

\refEnv{⟨name⟩}

References a documented LaTeX environment with given ⟨*name*⟩. The page reference is suppressed if it links to the same page.

```
We have created \refEnv{foocolorbox} as an example.
```

We have created foocolorbox→ P. 381 as an example.

\refEnv*{⟨name⟩}

References a documented LaTeX environment with given ⟨*name*⟩. There is no page reference.

```
We have created \refEnv*{foocolorbox} as an example.
```

We have created foocolorbox as an example.

\refKey{⟨name⟩}

References a documented key with given ⟨*name*⟩ where ⟨*name*⟩ is the full path name of the key. The page reference is suppressed if it links to the same page.

```
We have created \refKey{/foo/footitle} as an example.
```

We have created `/foo/footitle` → P. 381 as an example.

\refKey*{⟨name⟩}

References a documented key with given ⟨*name*⟩ where ⟨*name*⟩ is the full path name of the key. There is no page reference.

```
We have created \refKey*{/foo/footitle} as an example.
```

We have created `/foo/footitle` as an example.

\refAux{⟨name⟩}

References some auxiliary environment, key, value, or color. The hyperlink color is used, but there is no real link.

```
Some pages back, one can see \refAux{/foo/footitle} as an example.
```

Some pages back, one can see `/foo/footitle` as an example.

\refAuxcs{⟨name⟩}

References some auxiliary macro ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. The hyperlink color is used, but there is no real link.

```
Some pages back, one can see \refAuxcs{fooaux} as an example.
```

Some pages back, one can see `\fooaux` as an example.

\colDef{⟨text⟩}

Sets ⟨*text*⟩ with the command color, see `/tcb/color command` → P. 392.

```
This is my \colDef{text}.
```

This is my text.

\colOpt{⟨text⟩}

Sets ⟨*text*⟩ with the option color, see `/tcb/color option` → P. 392.

```
This is my \colOpt{text}.
```

This is my text.

N 2014-09-19 **\tcbdocmarginnote**[⟨*options*⟩]{⟨*text*⟩}

Creates a `tcolorbox` note with the given ⟨*text*⟩ inside the margin using the `marginnote` package. The style of the `tcolorbox` is predefined and can be altered by `/tcb/doc marginnote` → P. 396 and the given ⟨*options*⟩.

```
Some text\tcbdocmarginnote{Note A}
which is commented by a note inside the margin.
Alternatively to |\tcbdocmarginnote|, you can always use
|\marginnote| with a |tcolorbox| directly.\par
This is further text%
\tcbdocmarginnote[colframe=blue!50!white,colback=blue!5!white]{Note B}
with another note.
```

Note A

Note B

Some text which is commented by a note inside the margin. Alternatively to `\tcbdocmarginnote`, you can always use `\marginnote` with a `tcolorbox` directly.
This is further text with another note.

N 2014-09-19 **\tcbdocnew**{⟨*date*⟩}

Auxiliary macro which typesets the `/tcb/doclang/new` → P. 393 text with the given ⟨*date*⟩. It may be redefined for customization.

New: 1978-02-09

```
\tcbdocnew{1981-10-29}.
% Next one is displayed in the margin:
\tcbdocmarginnote{\tcbdocnew{1978-02-09}}
```

New: 1981-10-29.

N 2014-09-19 **\tcbdocupdated**{⟨*date*⟩}

Auxiliary macro which typesets the `/tcb/doclang/updated` → P. 393 text with the given ⟨*date*⟩. It may be redefined for customization.

```
\tcbdocupdated{2014-09-19}.
```

Updated: 2014-09-19.

21.2 Option Keys of the Library

U 2015-03-16 **/tcb/docexample** (style, no value)

Sets the style for dispExample[→P. 385] and dispListing[→P. 386] with the colors ExampleBack and ExampleFrame. To change the appearance of the examples, this style can be redefined.

```
% Predefined style:
\tcbset{
  docexample/.style={colframe=ExampleFrame,colback=ExampleBack,
    before skip=\medskipamount,after skip=\medskipamount,
    fontlower=\footnotesize}
}
```

/tcb/documentation listing options=⟨*key list*⟩ (no default, initially style=tcbdocumentation)

Sets the options from the package listings [6]. They are used inside dispExample[→P. 385] and dispListing[→P. 386] to typeset the listings. Note that this is not identical to the key /tcb/listing options[→P. 249] which is used for 'normal' listings.
Used for /tcb/listing engine[→P. 254]=listings only.

/tcb/documentation listing style=⟨*listing style*⟩ (no default, initially tcbdocumentation)

Abbreviation for documentation listing options={style=...}. This key sets a ⟨*style*⟩ for the listings package, see [6]. Note that this is not identical to the key /tcb/listing style[→P. 249] which is used for 'normal' listings.
Used for /tcb/listing engine[→P. 254]=listings only.

/tcb/documentation minted style=⟨*key list*⟩ (no default, initially unset)

Sets a ⟨*style*⟩ known to Pygments [12] for the package minted [14], if used. Note that this is not identical to the key /tcb/minted style[→P. 253] which is used for 'normal' listings.
Used for /tcb/listing engine[→P. 254]=minted only.

/tcb/documentation minted options=⟨*minted style*⟩ (no default, initially tabsize=2,fontsize=\small)

Sets the options from the package minted [14] which are used during typesetting of the listing, if used. Note that this is not identical to the key /tcb/minted options[→P. 252] which is used for 'normal' listings.
Used for /tcb/listing engine[→P. 254]=minted only.

The following two keys are deprecated and without function (v3.50 and above). Use /tcb/before[→P. 76] and /tcb/after[→P. 76] with appropriate values instead. Also see /tcb/docexample.

U 2015-03-16 **/tcb/before example=**⟨*macros*⟩ (no default, initially empty)

Sets the ⟨*macros*⟩ which are executed before dispExample[→P. 385] and dispListing[→P. 386] additional to /tcb/before[→P. 76].

deprecated

/tcb/after example=⟨*macros*⟩ (no default, initially empty)

Sets the ⟨*macros*⟩ which are executed after dispExample[→P. 385] and dispListing[→P. 386] additional to /tcb/after[→P. 76].

deprecated

N 2015-01-09 **/tcb/index command=**⟨*macro*⟩ (no default, initially `\index`)

Replaces the internally used `\index` macro by the given ⟨*macro*⟩. The ⟨*macro*⟩ has to take one mandatory argument like `\index`. This option is mutually exclusive with `/tcb/index command name`.

```
\tcbset{index command=\myindexcommand}
```

N 2015-01-09 **/tcb/index command name=**⟨*name*⟩ (no default, initially unset)

Replaces the internally used `\index` macro by `\index[`⟨*name*⟩`]`, i.e. `\index{...}` is replaced by `\index[`⟨*name*⟩`]{...}`. This option is intended to be used with `imakeidx` and is mutually exclusive with `/tcb/index command`.

```
\tcbset{index command name=mydoc}
```

/tcb/index format=⟨*format*⟩ (no default, initially `pgf`)

Determines the basic ⟨*format*⟩ of the generated index. Feasible values are:

- `pgfsection`: The index is formatted like in the `pgf` documentation (as a section).
- `pgfchapter`: The index is formatted like in the `pgf` documentation (as a chapter).
- `pgf`: Alias for `pgfsection`.
- `doc`: The index is assumed to be formatted by `doc` or `ltxdoc`. The usage of `makeindex` with `-s gind.ist` is assumed. The package `hypdoc` has to be loaded *before* `tcolorbox`.
- `off`: The index is not formatted by `tcolorbox`. Use this, if the index is formatted by other package like `imakeidx`.

/tcb/index actual=⟨*character*⟩ (no default, initially `@`)

Sets the character for 'actual' in automatic indexing.

/tcb/index quote=⟨*character*⟩ (no default, initially `"`)

Sets the character for 'quote' in automatic indexing.

/tcb/index level=⟨*character*⟩ (no default, initially `!`)

Sets the character for 'level' in automatic indexing.

/tcb/index default settings (style, no value)

Sets the `makeindex` default values for `/tcb/index actual`, `/tcb/index quote`, and `/tcb/index level`.

/tcb/index german settings (style, no value)

Sets the `makeindex` values recommended for German language texts. This is identical to setting the following:

```
\tcbset{index actual={=},index quote={!},index level={>}}
```

`/tcb/index annotate=true|false` (default `true`, initially `true`)

If set to `true`, the index entries are annotated with short descriptions given by `/tcb/doclang/environment` → P. 393, `/tcb/doclang/key` → P. 393, and others.

`/tcb/index colorize=true|false` (default `true`, initially `false`)

If set to `true`, the index entries colorized according to the color settings given by `/tcb/color environment`, `/tcb/color key`, and others.

`/tcb/color command=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by macro definitions.

`/tcb/color environment=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by environment definitions.

`/tcb/color key=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by key definitions.

`/tcb/color value=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by value definitions.

N 2015-01-08 `/tcb/color counter=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by counter definitions.

N 2015-01-08 `/tcb/color length=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by length definitions.

`/tcb/color color=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by color definitions.

`/tcb/color definition=`⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color for `/tcb/color command`, `/tcb/color environment`, `/tcb/color key`, `/tcb/color value`, `/tcb/color counter`, `/tcb/color length`, and `/tcb/color color`.

`/tcb/color option=`⟨*color*⟩ (no default, initially `Option`)

Sets the color used for optional arguments.

`/tcb/color hyperlink=`⟨*color*⟩ (no default, initially `Hyperlink`)

Sets the color for all hyper-links, i. e. all internal and external links.

The following keys are provided for language specific settings. The English language is predefined.

`/tcb/english language` (style, no value)
Sets all language specific settings to English.

`/tcb/doclang/color=`⟨*text*⟩ (no default, initially `color`)
Text used in the index for colors.

`/tcb/doclang/colors=`⟨*text*⟩ (no default, initially `Colors`)
Heading text in the index for colors.

N 2015-01-08 `/tcb/doclang/counter=`⟨*text*⟩ (no default, initially `counter`)
Text used in the index for counters.

N 2015-01-08 `/tcb/doclang/counters=`⟨*text*⟩ (no default, initially `Counters`)
Heading text in the index for counters.

`/tcb/doclang/environment=`⟨*text*⟩ (no default, initially `environment`)
Text used in the index for environments.

`/tcb/doclang/environments=`⟨*text*⟩ (no default, initially `Environments`)
Heading text in the index for environments.

`/tcb/doclang/environment content=`⟨*text*⟩ (no default, initially `environment content`)
Text used in `docEnvironment` → P. 381.

`/tcb/doclang/index=`⟨*text*⟩ (no default, initially `Index`)
Heading text for the index.

`/tcb/doclang/key=`⟨*text*⟩ (no default, initially `key`)
Text used in the index for keys.

`/tcb/doclang/keys=`⟨*text*⟩ (no default, initially `Keys`)
Heading text used in the index for keys.

N 2015-01-08 `/tcb/doclang/length=`⟨*text*⟩ (no default, initially `length`)
Text used in the index for lengths.

N 2015-01-08 `/tcb/doclang/lengths=`⟨*text*⟩ (no default, initially `Lengths`)
Heading text in the index for lengths.

N 2014-09-19 `/tcb/doclang/new=`⟨*text*⟩ (no default, initially `New`)
Announcement text for new content.

`/tcb/doclang/pageshort=`⟨*text*⟩ (no default, initially `P.`)
Short text for page references.

N 2014-09-19 `/tcb/doclang/updated=`⟨*text*⟩ (no default, initially `Updated`)
Announcement text for updated content.

`/tcb/doclang/value=`⟨*text*⟩ (no default, initially `value`)
Text used in the index for values.

`/tcb/doclang/values=`⟨*text*⟩ (no default, initially `Values`)
Heading text in the index for values.

/tcb/doc left=⟨*length*⟩ (no default, initially 2em)

Sets the left hand offset of the documentation texts from docCommand → P. 380, docEnvironment → P. 381, docKey → P. 381, etc, to ⟨*length*⟩.

```
\begin{docCommand*}[doc left=2cm,doc left indent=-2cm]{myCommandA}{\marg{argment}}
  This is the documentation of \refCom{myCommandA} which takes one \meta{argument}.
  \refCom{myCommandA} does some funny things with its \meta{argument}.
\end{docCommand*}
```

\myCommandA{⟨*argment*⟩}

This is the documentation of \myCommandA which takes one ⟨*argument*⟩. \myCommandA does some funny things with its ⟨*argument*⟩.

/tcb/doc right=⟨*length*⟩ (no default, initially 0em)

Sets the right hand offset of the documentation texts from docCommand → P. 380, docEnvironment → P. 381, docKey → P. 381, etc, to ⟨*length*⟩.

```
\begin{docCommand*}[doc right=2cm]{myCommandB}{\marg{argment}}
  This is the documentation of \refCom{myCommandB} which takes one \meta{argument}.
  \refCom{myCommandB} does some funny things with its \meta{argument}.
\end{docCommand*}
```

\myCommandB{⟨*argment*⟩}

This is the documentation of \myCommandB which takes one ⟨*argument*⟩. \myCommandB does some funny things with its ⟨*argument*⟩.

/tcb/doc left indent=⟨*length*⟩ (no default, initially -2em)

Sets the left hand indent of documentation heads from docCommand → P. 380, docEnvironment → P. 381, docKey → P. 381, etc, to ⟨*length*⟩.

```
\begin{docCommand*}[doc left indent=2cm]{myCommandC}{\marg{argment}}
  This is the documentation of \refCom{myCommandC} which takes one \meta{argument}.
  \refCom{myCommandC} does some funny things with its \meta{argument}.
\end{docCommand*}
```

\myCommandC{⟨*argment*⟩}

This is the documentation of \myCommandC which takes one ⟨*argument*⟩. \myCommandC does some funny things with its ⟨*argument*⟩.

/tcb/doc right indent=⟨*length*⟩ (no default, initially 0pt)

Sets the right hand indent of documentation heads from docCommand → P. 380, docEnvironment → P. 381, docKey → P. 381, etc, to ⟨*length*⟩.

```
\begin{docCommand*}[doc right indent=-10mm,doc right=10mm,
    doc description=test value]{myCommandD}{\marg{argment}}
  This is the documentation of \refCom{myCommandD} which takes one \meta{argument}.
  \refCom{myCommandD} does some funny things with its \meta{argument}.
\end{docCommand*}
```

\myCommandD{⟨*argment*⟩} (test value)

This is the documentation of \myCommandD which takes one ⟨*argument*⟩. \myCommandD does some funny things with its ⟨*argument*⟩.

The head lines of the main documentation environments docCommand→ P. 380, docEnvironment→ P. 381, docKey→ P. 381, etc, are set inside tcolorboxes. Options to these tcolorboxes can be given using the following keys.

/tcb/doc head command=⟨*options*⟩ (no default, initially empty)

Sets ⟨*options*⟩ for the head line of docCommand→ P. 380 and docCommand*→ P. 381.

```
\tcbset{doc head command={interior style={fill,left color=red!20!white,
  right color=blue!20!white}}}

\begin{docCommand*}{myCommandE}{\marg{argment}}
  This is the documentation of \refCom{myCommandE} which takes one \meta{argument}.
  \refCom{myCommandE} does some funny things with its \meta{argument}.
\end{docCommand*}
```

\myCommandE{⟨*argment*⟩}

This is the documentation of \myCommandE which takes one ⟨*argument*⟩. \myCommandE does some funny things with its ⟨*argument*⟩.

/tcb/doc head environment=⟨*options*⟩ (no default, initially empty)

Sets ⟨*options*⟩ for the head line of docEnvironment→ P. 381 and docEnvironment*→ P. 381.

```
\tcbset{doc head environment={beamer,boxsep=2pt,arc=2pt,colback=green!20!white,
  after=\par\smallskip}}

\begin{docEnvironment*}{myEnvironment}{\marg{argment}}
  This is the documentation of \refEnv{myEnvironment} which
  takes one \meta{argument}.
\end{docEnvironment*}
```

\begin{myEnvironment}{⟨*argment*⟩}
⟨*environment content*⟩
\end{myEnvironment}

This is the documentation of myEnvironment which takes one ⟨*argument*⟩.

/tcb/doc head key=⟨*options*⟩ (no default, initially empty)

Sets ⟨*options*⟩ for the head line of docKey→ P. 381 and docKey*→ P. 382.

```
\tcbset{doc head key={boxsep=4pt,arc=4pt,boxrule=0.6pt,
  frame style=fill,interior style=fill,colframe=green!50!black}}

\begin{docKey*}{/foo/myKey}{}{no value}
  This is the documentation of \refKey{/foo/myKey}.
\end{docKey*}
```

/foo/myKey (no value)

This is the documentation of /foo/myKey.

/tcb/doc head=⟨*options*⟩ (no default, initially empty)

Shortcut for setting the same ⟨*options*⟩ for /tcb/doc head command, /tcb/doc head environment, and /tcb/doc head key.

/tcb/doc description=⟨*text*⟩ (no default, initially empty)

Sets a (short!) additional description ⟨*text*⟩ for docCommand→ P. 380 or docEnvironment→ P. 381. Such a description is mandatory for docKey→ P. 381.

```
\begin{docCommand*}[doc description=my description]{myCommandF}{\marg{argment}}
  This is the documentation of \refCom{myCommandF} which takes one \meta{argument}.
  \refCom{myCommandF} does some funny things with its \meta{argument}.
\end{docCommand*}
```

\myCommandF{⟨*argment*⟩} (my description)

This is the documentation of \myCommandF which takes one ⟨*argument*⟩. \myCommandF does some funny things with its ⟨*argument*⟩.

Note that the description ⟨*text*⟩ may overlap with the text on the left hand side if too long. Linebreaks can be used inside the ⟨*text*⟩.

/tcb/doc into index=true|false (default true, initially true)

If set to false, no index entries are written for the main documentation environments. The same effect is achieved by using e. g. docCommand*→ P. 381 instead of docCommand→ P. 380.

N 2014-09-19 **/tcb/doc marginnote=⟨*options*⟩** (no default, initially empty)

Sets style ⟨*options*⟩ for the displayed box of the \tcbdocmarginnote→ P. 389 command.

```
\tcbset{doc marginnote={colframe=blue!50!white,colback=blue!5!white}}%
This is some text\tcbdocmarginnote{Note A}
which is commented by a note inside the margin.
```

Note A

This is some text which is commented by a note inside the margin.

N 2014-09-19 **/tcb/doc new=⟨*date*⟩** (style, no default)

Adds a a marginnote with a 'New: ⟨*data*⟩' message at the beginning of the upper box part. The intended use is inside the option list of docCommand→ P. 380, docEnvironment→ P. 381, etc.

```
\begin{docCommand}[doc new=2000-01-01]{foosomething}{\marg{text}}
Some command for something.
\end{docCommand}
```

New: 2000-01-01

\foosomething{⟨*text*⟩}

Some command for something.

N 2014-09-19 **/tcb/doc updated=⟨*date*⟩** (style, no default)

Adds a marginnote with a 'Updated: ⟨*data*⟩' message at the beginning of the upper box part. See /tcb/doc new.

N 2014-09-19 **/tcb/doc new and updated={⟨*new date*⟩}{⟨*update date*⟩}** (style, no default)

Adds a marginnote with 'New: ⟨*new date*⟩' and 'Updated: ⟨*update data*⟩' messages at the beginning of the upper box part. See /tcb/doc new.

21.3 Predefined Colors of the Library

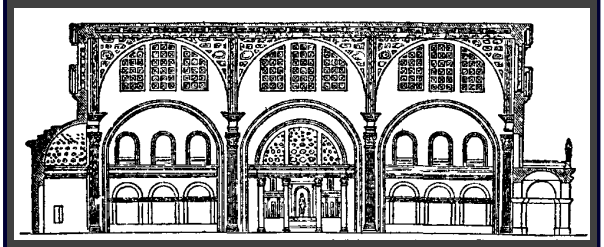
The following colors are predefined. They are used as default colors in some library commands.

Option , Definition , ExampleFrame , ExampleBack , Hyperlink .

A Picture Credits

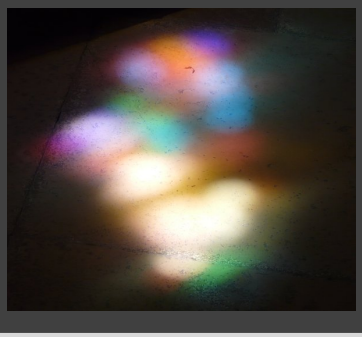
The following pictures were used inside this documentation.

Basilica_5.png



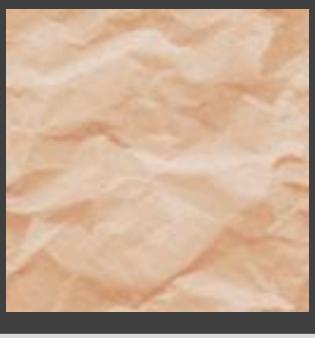
http://commons.wikimedia.org/wiki/File:Basilica_5.png

lichtspiel.jpg



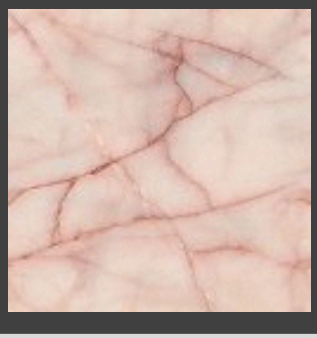
Photograph taken by Thomas F. Sturm.

crinklepaper.png



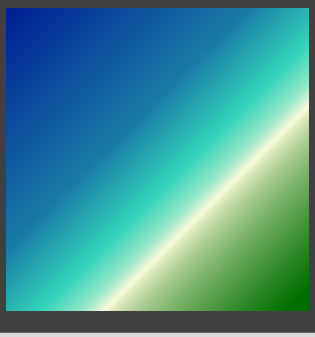
Created with GIMP.
http://www.gimp.org

pink_marble.png



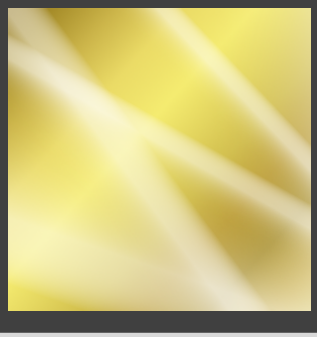
Created with GIMP.
http://www.gimp.org

blueshade.png



Created with GIMP.
http://www.gimp.org

goldshade.png



Created with GIMP.
http://www.gimp.org

References

[1] Donald Arseneau. *The varwidth package*. Sept. 26, 2011.
mirrors.ctan.org/macros/latex/contrib/varwidth/varwidth-doc.pdf.

[2] Peter Breitenlohner. *The ε-TEX manual*. Version 2. Feb. 1998.
http://ftp.gwdg.de/pub/ctan/systems/e-tex/v2/doc/etex_man.pdf.

[3] David Carlisle. *The file ltxdoc.dtx for use with LaTeX 2ε*. Nov. 11, 2007.
http://mirror.ctan.org/macros/latex/base/ltxdoc.dtx.

[4] David Carlisle. *The tabularx package*. May 13, 2014.
http://mirror.ctan.org/macros/latex/required/tools/tabularx.pdf.

[5] Toby Cubitt. *The cleveref Package*. Mar. 22, 2013.
http://mirror.ctan.org/macros/latex/contrib/cleveref/cleveref.pdf.

[6] Carsten Heinz, Brooks Moses, and Jobst Hoffmann. *The Listings Package*. Version 1.5b. Aug. 26, 2013.
http://mirror.ctan.org/macros/latex/contrib/listings/listings.pdf.

[7] Philipp Lehman. *The etoolbox Package*. Jan. 3, 2011.
http://mirror.ctan.org/macros/latex/contrib/etoolbox/etoolbox.pdf.

[8] Frank Mittelbach. *The doc and shortvrb Packages*. June 30, 2011.
http://mirror.ctan.org/macros/latex/base/doc.dtx.

[9] Frank Mittelbach. *An environment for multicolumn output*. June 21, 2014.
http://mirror.ctan.org/macros/latex/required/tools/multicol.pdf.

[10] Heiko Oberdiek. *The listingsutf8 package*. Nov. 10, 2011.
http://mirror.ctan.org/macros/latex/contrib/oberdiek/listingsutf8.pdf.

[11] The LaTeX3 Project. *The xparse Package*. Dec. 31, 2013.
http://mirror.ctan.org/macros/latex/contrib/l3packages/xparse.pdf.

[12] *Pygments. Python syntax highlighter*. 2013.
http://pygments.org/.

[13] Will Robertson. *A couple of things involving environments*. Apr. 1, 2013.
http://mirror.ctan.org/macros/latex/contrib/environ/environ.pdf.

[14] Konrad Rudolph. *The minted package. Highlighted source code in LaTeX*. Sept. 17, 2011.
http://mirror.ctan.org/macros/latex/contrib/minted/minted.pdf.

[15] Rainer Schöpf, Bernd Raichle, and Chris Rowley. *A New Implementation of LaTeX's verbatim and verbatim* Environments*. Mar. 12, 2001.
http://mirror.ctan.org/macros/latex/required/tools/verbatim.pdf.

[16] Thomas F. Sturm. *Mathematik 1 (B.Eng.)* Vorlesungs-Skriptum. Neubiberg: Universität der Bundeswehr München, 2007.
http://www.unibw.de/bw/sturm.

[17] Thomas F. Sturm. *Mathematik 2 (B.Eng.)* Vorlesungs-Skriptum. Neubiberg: Universität der Bundeswehr München, 2008.
http://www.unibw.de/bw/sturm.

[18] Thomas F. Sturm. *Höhere Mathematik: Fortgeschrittene mathematische Methoden (M.Eng.)* Vorlesungs-Skriptum. Neubiberg: Universität der Bundeswehr München, 2010.
http://www.unibw.de/bw/sturm.

[19] Thomas F. Sturm. *LaTeX – Einführung in das Textsatzsystem*. 10th edition. RRZN-Handbücher. Hannover: Regionales Rechenzentrum für Niedersachsen (RRZN), Mar. 2014.
http://www.rrzn.uni-hannover.de/buch.html?&titel=latex.

[20] Till Tantau. *The TikZ and PGF Packages. Manual for version 3.0.0*. Dec. 20, 2013.
http://sourceforge.net/projects/pgf/.

[21] Kresten Krab Thorub, Frank Jensen, and Chris Rowley. *The calc package – Infix notation arithmetic in LaTeX*. Aug. 22, 2007.
http://mirror.ctan.org/macros/latex/required/tools/.

Index

- key, 370
! key, 370

absquote environment, 386
add to height key, 53
add to list key, 94
add to width key, 33
adjust text key, 17
adjusted title key, 17
adjusted title after break key, 307
after key, 76
after app key, 347
after example key, 390
after lower key, 59
after lower app key, 347
after lower pre key, 347
after pre key, 347
after skip key, 78
after title key, 58
after title app key, 346
after title pre key, 346
after upper key, 59
after upper app key, 346
after upper pre key, 346
all key, 9
all value, 46, 47, 239, 309, 311
ams align key, 292
ams align lower key, 292
ams align upper key, 292
ams align* key, 292
ams align* lower key, 292
ams align* upper key, 292
ams equation key, 291
ams equation lower key, 291
ams equation upper key, 291
ams equation* key, 291
ams equation* lower key, 291
ams equation* upper key, 291
ams gather key, 293
ams gather lower key, 293
ams gather upper key, 293
ams gather* key, 293
ams gather* lower key, 293
ams gather* upper key, 293
ams nodisplayskip key, 294
ams nodisplayskip lower key, 294
ams nodisplayskip upper key, 294
arc key, 35
arc is angular key, 37
arc is curved key, 37
areasize value, 342
areasize* value, 342
at begin tikz key, 166
at begin tikz reset key, 166
at end tikz key, 166
at end tikz reset key, 166
attach boxed title to bottom center key, 131
attach boxed title to bottom left key, 131
attach boxed title to bottom right key, 131
attach boxed title to top center key, 131
attach boxed title to top left key, 131
attach boxed title to top right key, 131
attach title key, 19
attach title to upper key, 19
auto value, 92
auto counter key, 97
auto limited value, 92
auto outer arc key, 37
autoparskip key, 76

base value, 77
baseline key, 77
baselineskip value, 309
beamer key, 196
beamer Skin, 196
beamerfirst Skin, 198
beamerlast Skin, 200
beamermiddle Skin, 199
bean arc key, 36
before key, 76
before app key, 347
before example key, 390
before lower key, 59
before lower app key, 347
before lower pre key, 347
before nobreak key, 79
before pre key, 347
before skip key, 78
before title key, 58
before title app key, 346
before title pre key, 346
before upper key, 58
before upper app key, 346
before upper pre key, 346
beforeafter skip key, 78
bicolor key, 191
bicolor Skin, 191
bicolorfirst Skin, 193
bicolorlast Skin, 195
bicolormiddle Skin, 194
blank key, 180
blanker key, 206
blend before title key, 102
blend before title code key, 103
blend into key, 101
borderline key, 151
borderline east key, 154
borderline horizontal key, 155
borderline north key, 154
borderline south key, 154

borderline vertical key, 155
borderline west key, 154
bottom key, 41
bottom value, 32, 71, 72, 77, 238
bottom seam value, 71, 72
bottomrule key, 34
bottomrule at break key, 310
bottomsep at break key, 310
bottomtitle key, 41
box align key, 77
\boxarrayclear, 327
\boxarraygetbox, 331
\boxarraygetdepth, 332
\boxarraygetheight, 332
\boxarraygetsize, 329
\boxarraygettotalheight, 333
\boxarraygetwidth, 332
\boxarrayreset, 326
boxarraystore environment, 329
boxed title style key, 133
boxrule key, 35
boxsep key, 37
\brackets, 384
break value, 296
break at key, 308
breakable key, 8, 306
broken value, 139–141

capture key, 89
center value, 29, 32, 71, 72, 77, 238
center lower key, 31
center seam value, 71, 72
center title key, 31
center upper key, 31
change value, 296
change apart value, 296
change break value, 296
change standard value, 295
check odd page key, 94
circular arc key, 36
clear preamble key, 377
clear preclass key, 377
clip lower key, 150
clip title key, 149
clip upper key, 149
clip watermark key, 145
code key, 96
colback key, 26
colbacklower key, 192
colbacktitle key, 26
\colDef, 388
colframe key, 26
collower key, 27
colon value, 102
colon hang value, 102
\colOpt, 388
color key, 393
color color key, 392
color command key, 392
color counter key, 392
color definition key, 392
color environment key, 392
color hyperlink key, 392
color key key, 392
color length key, 392
color option key, 392
color value key, 392
Colors
    Definition, 396
    ExampleBack, 396
    ExampleFrame, 396
    foocolor, 383
    Hyperlink, 396
    Option, 396
colors key, 393
coltext key, 27
coltitle key, 27
colupper key, 27
comment key, 255
comment above listing key, 261
comment above* listing key, 261
comment and listing key, 257
comment only key, 255
comment outside listing key, 259
comment side listing key, 259
comment style key, 257
compilable listing key, 265
compiler key, 376
compress page key, 309
\consumeboxarray, 330
\consumetcboxarray, 330
counter key, 393
Counters
    foocounter, 383
counters key, 393
Crefname key, 100
crefname key, 100
\cs, 384

dash value, 102
dash hang value, 102
\DeclareTCBInputListing, 366
\DeclareTCBListing, 364
\DeclareTCBox, 361
\DeclareTCBoxFit, 367
\DeclareTColorBox, 358
\DeclareTotalTCBox, 362
\DeclareTotalTCBoxFit, 368
\DeclareTotalTColorBox, 360
Definition color, 396
description color key, 286
description delimiters key, 286
description delimiters none key, 286
description delimiters parenthesis key, 286
description font key, 287
description formatter key, 287
detach title key, 19
dispExample environment, 385
dispExample* environment, 385

dispListing environment, 386
dispListing* environment, 386
do not store to box array key, 329
doc value, 391
doc description key, 396
doc head key, 395
doc head command key, 395
doc head environment key, 395
doc head key key, 395
doc into index key, 396
doc left key, 394
doc left indent key, 394
doc marginnote key, 396
doc new key, 396
doc new and updated key, 396
doc right key, 394
doc right indent key, 394
doc updated key, 396
\docAuxCommand, 382
\docAuxCommand*, 382
\docAuxEnvironment, 382
\docAuxEnvironment*, 382
\docAuxKey, 382
\docAuxKey*, 382
\docColor, 383
\docColor*, 383
docCommand environment, 380
docCommand* environment, 381
\docCounter, 383
\docCounter*, 383
docEnvironment environment, 381
docEnvironment* environment, 381
docexample key, 390
docKey environment, 381
docKey* environment, 382
\docLength, 383
\docLength*, 383
documentation key, 9
documentation listing options key, 390
documentation listing style key, 390
documentation minted options key, 390
documentation minted style key, 390
\docValue, 382
\docValue*, 382
downhill value, 46, 47
draft key, 215
draft Skin, 215
draftmode key, 175
drop fuzzy midday shadow key, 157
drop fuzzy shadow key, 156
drop fuzzy shadow east key, 160
drop fuzzy shadow north key, 160
drop fuzzy shadow northeast key, 160
drop fuzzy shadow northwest key, 160
drop fuzzy shadow south key, 159
drop fuzzy shadow southeast key, 159
drop fuzzy shadow southwest key, 159
drop fuzzy shadow west key, 159
drop large lifted shadow key, 161
drop lifted shadow key, 161
drop midday shadow key, 156
drop shadow key, 156
drop shadow east key, 159
drop shadow north key, 158
drop shadow northeast key, 159
drop shadow northwest key, 158
drop shadow south key, 158
drop shadow southeast key, 158
drop shadow southwest key, 158
drop shadow west key, 158
drop small lifted shadow key, 161

east value, 46, 47
empty key, 205
empty Skin, 205
empty value, 113, 114
emptyfirst Skin, 207
emptylast Skin, 209
emptymiddle Skin, 208
enforce breakable key, 307
english language key, 393
enhanced key, 178
enhanced Skin, 178
enhanced jigsaw key, 185
enhanced jigsaw Skin, 185
enhanced standard key, 180
enhancedfirst Skin, 182
enhancedfirst jigsaw Skin, 186
enhancedlast Skin, 184
enhancedlast jigsaw Skin, 190
enhancedmiddle Skin, 183
enhancedmiddle jigsaw Skin, 187
enlarge bottom at break by key, 81
enlarge bottom by key, 81
enlarge bottom finally by key, 80
enlarge by key, 82
enlarge left by key, 81
enlarge right by key, 81
enlarge top at break by key, 81
enlarge top by key, 81
enlarge top initially by key, 80
enlargepage key, 308
enlargepage flexible key, 309
environment key, 376, 393
environment content key, 393
environment with percent key, 376
Environments
    absquote, 386
    boxarraystore, 329
    dispExample, 385
    dispExample*, 385
    dispListing, 386
    dispListing*, 386
    docCommand, 380
    docCommand*, 381
    docEnvironment, 381
    docEnvironment*, 381
    docKey, 381
    docKey*, 382

extcolorbox, 373
extikzpicture, 374
fooauxenv, 382
foocolorbox, 381
foocolorbox*, 381
tcbclipframe, 146
tcbclipinterior, 148
tcbcliptitle, 148
tcbexternal, 371
tcbinvclipframe, 147
tcbitemize, 234
tcblisting, 243
tcboutputlisting, 245
tcbraster, 233
tcbverbatimwrite, 105
tcbwritetemp, 105
tcolorbox, 11
environments key, 393
equal height group key, 56
evenpage value, 44, 83
every box key, 86
every box on higher layers key, 87
every box on layer n key, 87
every float key, 69
every listing line key, 250
every listing line* key, 250
ExampleBack color, 396
ExampleFrame color, 396
extcolorbox environment, 373
extend freelance key, 217
extend freelancefirst key, 217
extend freelancelast key, 217
extend freelancemiddle key, 217
external key, 9, 95
externalize key, 370
externalize example key, 375
externalize example! key, 375
externalize listing key, 375
externalize listing! key, 375
extikzpicture environment, 374
extras key, 312
extras broken key, 312
extras broken pre key, 355
extras first key, 312
extras first and middle key, 312
extras first and middle pre key, 355
extras first pre key, 355
extras last key, 312
extras last pre key, 355
extras middle key, 312
extras middle and last key, 312
extras middle and last pre key, 355
extras middle pre key, 355
extras pre key, 355
extras unbroken key, 312
extras unbroken and first key, 312
extras unbroken and first pre key, 355
extras unbroken and last key, 312
extras unbroken and last pre key, 355
extras unbroken pre key, 355
extrude bottom by key, 85
extrude by key, 85
extrude left by key, 84
extrude right by key, 84
extrude top by key, 85

false value, 76, 79, 240, 306
fbox value, 42
figures value, 101
fill image opacity key, 228
fill image options key, 228
fill image scale key, 228
fill overzoom image key, 224
fill overzoom image* key, 224
fill overzoom picture key, 224
fill plain image key, 222
fill plain image* key, 222
fill plain picture key, 222
fill shrink image key, 226
fill shrink image* key, 226
fill shrink picture key, 226
fill stretch image key, 223
fill stretch image* key, 223
fill stretch picture key, 223
fill tile image key, 227
fill tile image* key, 227
fill tile picture key, 227
fill tile picture* key, 227
fill zoom image key, 225
fill zoom image* key, 225
fill zoom picture key, 225
final value, 344
finish key, 171
finish broken key, 172
finish broken pre key, 353
finish first key, 172
finish first and middle key, 172
finish first and middle pre key, 353
finish first pre key, 353
finish last key, 172
finish last pre key, 353
finish middle key, 172
finish middle and last key, 172
finish middle and last pre key, 353
finish middle pre key, 353
finish pre key, 353
finish unbroken key, 172
finish unbroken and first key, 172
finish unbroken and first pre key, 353
finish unbroken and last key, 172
finish unbroken and last pre key, 353
finish unbroken pre key, 353
first value, 139–141, 311
first and middle value, 139, 311
fit key, 336
fit algorithm key, 342
fit basedim key, 337
fit fontsize macros key, 338
fit height from key, 341

fit height plus key, 339
fit maxfontdiff key, 344
fit maxfontdiffgap key, 344
fit maxstep key, 344
fit maxwidthdiff key, 344
fit maxwidthdiffgap key, 344
fit skip key, 337
fit to key, 337
fit to height key, 337
fit warning key, 344
fit width from key, 340
fit width plus key, 339
fitbox value, 89
fitting key, 8
float key, 69
float* key, 69
floatplacement key, 69
flush center value, 29, 31
flush left value, 29, 31
flush right value, 29, 31
flushleft lower key, 31
flushleft title key, 31
flushleft upper key, 31
flushright lower key, 31
flushright title key, 31
flushright upper key, 31
fontlower key, 28
fontsize value, 342
fontsize* value, 342
fonttitle key, 28
fontupper key, 28
\fooaux, 382
fooaux key, 382
fooauxenv environment, 382
foocolor color, 383
foocolorbox environment, 381
foocolorbox* environment, 381
foocounter counter, 383
foodummy key, 387
\foolength length, 383
\foomakedocSubKey, 380
\foomakedocSubKey*, 380
\foosomething, 396
footitle key, 381
foovalue value, 382
force remake key, 370
forced value, 44, 83
forced center value, 92
forced left value, 92
forced right value, 92
forces nobeforeafter key, 76
frame code key, 116
frame code app key, 353
frame code pre key, 353
frame empty key, 116
frame engine key, 113
frame hidden key, 125
frame style key, 124
frame style image key, 124
frame style tile key, 125
freelance key, 217
freelance Skin, 217
freelance value, 113, 114
freelancefirst Skin, 217
freelancelast Skin, 217
freelancemiddle Skin, 217
fuzzy halo key, 157
fuzzy shadow key, 163

geometry nodes key, 115
graphical environment key, 112
graphics directory key, 221
graphics options key, 221
graphics pages key, 221
grow to left by key, 82
grow to right by key, 82

halign key, 29
halign lower key, 30
halign title key, 31
halign upper key, 29
halo key, 157
hbox key, 89
hbox value, 89
hbox boxed title key, 137
height key, 51
height fill key, 53
height fixed for key, 311
height from key, 52
height plus key, 51
highlight math key, 290
highlight math style key, 290
hooks key, 8
hybrid value, 342
hybrid* value, 342
Hyperlink color, 396
hyphenationfix key, 91

IfBooleanTF key, 357
IfNoValueTF key, 356
IfValueTF key, 357
ignore nobreak key, 79
ignored value, 23
image comment key, 255
\imagename, 219
\imagepage, 220
index key, 393
index actual key, 391
index annotate key, 392
index colorize key, 392
index command key, 391
index command name key, 391
index default settings key, 391
index format key, 391
index german settings key, 391
index level key, 391
index quote key, 391
input source on error key, 376
interior code key, 117

interior code app key, 354
interior code pre key, 354
interior empty key, 117
interior engine key, 114
interior hidden key, 126
interior style key, 125
interior style image key, 126
interior style tile key, 126
interior titled code key, 116
interior titled code app key, 353
interior titled code pre key, 354
interior titled empty key, 116
interior titled engine key, 113
invisible key, 21
invisible value, 21, 23

justify value, 29

key key, 393
Keys
  /foo/
    fooaux, 382
    foodummy, 387
    footitle, 381
  /tcb/
    add to height, 53
    add to list, 94
    add to width, 33
    adjust text, 17
    adjusted title, 17
    adjusted title after break, 307
    after, 76
    after app, 347
    after example, 390
    after lower, 59
    after lower app, 347
    after lower pre, 347
    after pre, 347
    after skip, 78
    after title, 58
    after title app, 346
    after title pre, 346
    after upper, 59
    after upper app, 346
    after upper pre, 346
    ams align, 292
    ams align lower, 292
    ams align upper, 292
    ams align*, 292
    ams align* lower, 292
    ams align* upper, 292
    ams equation, 291
    ams equation lower, 291
    ams equation upper, 291
    ams equation*, 291
    ams equation* lower, 291
    ams equation* upper, 291
    ams gather, 293
    ams gather lower, 293
    ams gather upper, 293
    ams gather*, 293
    ams gather* lower, 293
    ams gather* upper, 293
    ams nodisplayskip, 294
    ams nodisplayskip lower, 294
    ams nodisplayskip upper, 294
    arc, 35
    arc is angular, 37
    arc is curved, 37
    at begin tikz, 166
    at begin tikz reset, 166
    at end tikz, 166
    at end tikz reset, 166
    attach boxed title to bottom center, 131
    attach boxed title to bottom left, 131
    attach boxed title to bottom right, 131
    attach boxed title to top center, 131
    attach boxed title to top left, 131
    attach boxed title to top right, 131
    attach title, 19
    attach title to upper, 19
    auto outer arc, 37
    autoparskip, 76
    baseline, 77
    beamer, 196
    bean arc, 36
    before, 76
    before app, 347
    before example, 390
    before lower, 59
    before lower app, 347
    before lower pre, 347
    before nobreak, 79
    before pre, 347
    before skip, 78
    before title, 58
    before title app, 346
    before title pre, 346
    before upper, 58
    before upper app, 346
    before upper pre, 346
    beforeafter skip, 78
    bicolor, 191
    blank, 180
    blanker, 206
    blend before title, 102
    blend before title code, 103
    borderline, 151
    borderline east, 154
    borderline horizontal, 155
    borderline north, 154
    borderline south, 154
    borderline vertical, 155
    borderline west, 154

bottom, 41
bottomrule, 34
bottomrule at break, 310
bottomsep at break, 310
bottomtitle, 41
box align, 77
boxed title style, 133
boxrule, 35
boxsep, 37
break at, 308
breakable, 306
capture, 89
center lower, 31
center title, 31
center upper, 31
check odd page, 94
circular arc, 36
clip lower, 150
clip title, 149
clip upper, 149
clip watermark, 145
code, 96
colback, 26
colbacklower, 192
colbacktitle, 26
colframe, 26
collower, 27
color color, 392
color command, 392
color counter, 392
color definition, 392
color environment, 392
color hyperlink, 392
color key, 392
color length, 392
color option, 392
color value, 392
coltext, 27
coltitle, 27
colupper, 27
comment, 255
comment above listing, 261
comment above* listing, 261
comment and listing, 257
comment only, 255
comment outside listing, 259
comment side listing, 259
comment style, 257
compilable listing, 265
compress page, 309
description color, 286
description delimiters, 286
description delimiters none, 286
description delimiters parenthesis, 286
description font, 287
description formatter, 287
detach title, 19
do not store to box array, 329
doc description, 396
doc head, 395
doc head command, 395
doc head environment, 395
doc head key, 395
doc into index, 396
doc left, 394
doc left indent, 394
doc marginnote, 396
doc new, 396
doc new and updated, 396
doc right, 394
doc right indent, 394
doc updated, 396
docexample, 390
documentation listing options, 390
documentation listing style, 390
documentation minted options, 390
documentation minted style, 390
draft, 215
draftmode, 175
drop fuzzy midday shadow, 157
drop fuzzy shadow, 156
drop fuzzy shadow east, 160
drop fuzzy shadow north, 160
drop fuzzy shadow northeast, 160
drop fuzzy shadow northwest, 160
drop fuzzy shadow south, 159
drop fuzzy shadow southeast, 159
drop fuzzy shadow southwest, 159
drop fuzzy shadow west, 159
drop large lifted shadow, 161
drop lifted shadow, 161
drop midday shadow, 156
drop shadow, 156
drop shadow east, 159
drop shadow north, 158
drop shadow northeast, 159
drop shadow northwest, 158
drop shadow south, 158
drop shadow southeast, 158
drop shadow southwest, 158
drop shadow west, 158
drop small lifted shadow, 161
empty, 205
enforce breakable, 307
english language, 393
enhanced, 178
enhanced jigsaw, 185
enhanced standard, 180
enlarge bottom at break by, 81
enlarge bottom by, 81
enlarge bottom finally by, 80
enlarge by, 82
enlarge left by, 81
enlarge right by, 81
enlarge top at break by, 81
enlarge top by, 81
enlarge top initially by, 80

enlargepage, 308
enlargepage flexible, 309
equal height group, 56
every box, 86
every box on higher layers, 87
every box on layer n, 87
every float, 69
every listing line, 250
every listing line*, 250
extend freelance, 217
extend freelancefirst, 217
extend freelancelast, 217
extend freelancemiddle, 217
external, 95
externalize example, 375
externalize example!, 375
externalize listing, 375
externalize listing!, 375
extras, 312
extras broken, 312
extras broken pre, 355
extras first, 312
extras first and middle, 312
extras first and middle pre, 355
extras first pre, 355
extras last, 312
extras last pre, 355
extras middle, 312
extras middle and last, 312
extras middle and last pre, 355
extras middle pre, 355
extras pre, 355
extras unbroken, 312
extras unbroken and first, 312
extras unbroken and first pre, 355
extras unbroken and last, 312
extras unbroken and last pre, 355
extras unbroken pre, 355
extrude bottom by, 85
extrude by, 85
extrude left by, 84
extrude right by, 84
extrude top by, 85
finish, 171
finish broken, 172
finish broken pre, 353
finish first, 172
finish first and middle, 172
finish first and middle pre, 353
finish first pre, 353
finish last, 172
finish last pre, 353
finish middle, 172
finish middle and last, 172
finish middle and last pre, 353
finish middle pre, 353
finish pre, 353
finish unbroken, 172
finish unbroken and first, 172
finish unbroken and first pre, 353
finish unbroken and last, 172
finish unbroken and last pre, 353
finish unbroken pre, 353
fit, 336
fit algorithm, 342
fit basedim, 337
fit fontsize macros, 338
fit height from, 341
fit height plus, 339
fit maxfontdiff, 344
fit maxfontdiffgap, 344
fit maxstep, 344
fit maxwidthdiff, 344
fit maxwidthdiffgap, 344
fit skip, 337
fit to, 337
fit to height, 337
fit warning, 344
fit width from, 340
fit width plus, 339
float, 69
float*, 69
floatplacement, 69
flushleft lower, 31
flushleft title, 31
flushleft upper, 31
flushright lower, 31
flushright title, 31
flushright upper, 31
fontlower, 28
fonttitle, 28
fontupper, 28
forces nobeforeafter, 76
frame code, 116
frame code app, 353
frame code pre, 353
frame empty, 116
frame engine, 113
frame hidden, 125
frame style, 124
frame style image, 124
frame style tile, 125
freelance, 217
fuzzy halo, 157
fuzzy shadow, 163
geometry nodes, 115
graphical environment, 112
graphics directory, 221
graphics options, 221
graphics pages, 221
grow to left by, 82
grow to right by, 82
halign, 29
halign lower, 30
halign title, 31
halign upper, 29
halo, 157
hbox, 89

hbox boxed title, 137
height, 51
height fill, 53
height fixed for, 311
height from, 52
height plus, 51
highlight math, 290
highlight math style, 290
hyphenationfix, 91
IfBooleanTF, 357
IfNoValueTF, 356
IfValueTF, 357
ignore nobreak, 79
image comment, 255
index actual, 391
index annotate, 392
index colorize, 392
index command, 391
index command name, 391
index default settings, 391
index format, 391
index german settings, 391
index level, 391
index quote, 391
interior code, 117
interior code app, 354
interior code pre, 354
interior empty, 117
interior engine, 114
interior hidden, 126
interior style, 125
interior style image, 126
interior style tile, 126
interior titled code, 116
interior titled code app, 353
interior titled code pre, 354
interior titled empty, 116
interior titled engine, 113
invisible, 21
label, 93
label type, 93
left, 38
left skip, 79
lefthand ratio, 74
lefthand width, 73
leftlower, 38
leftright skip, 79
leftrule, 34
lefttitle, 38
leftupper, 38
lifted shadow, 164
lines before break, 307
list entry, 94
list text, 94
listing above comment, 261
listing above text, 260
listing above* comment, 261
listing above* text, 260
listing and comment, 257
listing and text, 254
listing engine, 254
listing file, 254
listing inputencoding, 250
listing only, 254
listing options, 249
listing outside comment, 259
listing outside text, 258
listing remove caption, 250
listing side comment, 259
listing side text, 258
listing style, 249
listing utf8, 251
lower separated, 24
lowerbox, 23
marker, 187
math, 290
math lower, 290
math upper, 290
middle, 41
minimum for equal height group, 57
minipage, 89
minipage boxed title, 137
minipage boxed title*, 137
minted language, 252
minted options, 252
minted style, 253
natural height, 51
no borderline, 153
no extras, 312
no extras first, 312
no extras last, 312
no extras middle, 312
no extras unbroken, 312
no finish, 172
no finish first, 172
no finish last, 172
no finish middle, 172
no finish unbroken, 172
no label type, 93
no listing options, 249
no overlay, 65
no process, 265
no recording, 106
no shadow, 156
no underlay, 169
no underlay boxed title, 170
no underlay first, 170
no underlay last, 170
no underlay middle, 170
no underlay unbroken, 170
no watermark, 141
nobeforeafter, 76
nofloat, 69
noparskip, 76
nophantom, 93
notitle, 17
notitle after break, 307
octogon arc, 36

on line, 91
only, 96
opacityback, 49
opacitybacktitle, 49
opacityfill, 49
opacityframe, 49
opacitylower, 50
opacitytext, 50
opacitytitle, 50
opacityupper, 50
outer arc, 37
overlay, 64
overlay app, 348
overlay broken, 65
overlay broken app, 349
overlay broken pre, 349
overlay first, 65
overlay first and middle, 65
overlay first and middle app, 349
overlay first and middle pre, 349
overlay first app, 348
overlay first pre, 348
overlay last, 65
overlay last app, 349
overlay last pre, 349
overlay middle, 65
overlay middle and last, 65
overlay middle and last app, 349
overlay middle and last pre, 349
overlay middle app, 349
overlay middle pre, 349
overlay pre, 348
overlay unbroken, 65
overlay unbroken and first, 65
overlay unbroken and first app, 349
overlay unbroken and first pre, 349
overlay unbroken and last, 65
overlay unbroken and last app, 349
overlay unbroken and last pre, 349
overlay unbroken app, 348
overlay unbroken pre, 348
oversize, 43
pad after break, 310
pad at break, 310
pad at break*, 310
pad before break, 310
pad before break*, 310
parbox, 90
parskip, 76
pdf comment, 256
pdf extension, 257
phantom, 93
phantomlabel, 93
process code, 265
raster after skip, 236
raster before skip, 236
raster column n, 240
raster column skip, 237
raster columns, 235
raster equal height, 239
raster equal height group, 239
raster equal skip, 236
raster even column, 240
raster even number, 241
raster even row, 241
raster every box, 240
raster force size, 240
raster halign, 238
raster height, 236
raster left skip, 237
raster number n, 241
raster odd column, 240
raster odd number, 241
raster odd row, 240
raster reset, 240
raster right skip, 237
raster row m, 241
raster row m column n, 241
raster row skip, 237
raster rows, 235
raster valign, 238
raster width, 235
record, 106
remake, 95
remember, 167
remember as, 168
reset, 95
reset and store to box array, 329
reset box array, 326
right, 39
right skip, 79
righthand ratio, 74
righthand width, 73
rightlower, 40
rightrule, 34
righttitle, 39
rightupper, 39
rotate, 167
rounded corners, 47
run arara, 267
run biber, 267
run bibtex, 267
run dvips, 267
run latex, 267
run lualatex, 267
run makeindex, 267
run pdflatex, 265
run ps2pdf, 267
run system command, 265
run xelatex, 267
savedelimiter, 25
savelowerto, 23
saveto, 22
scale, 167
segmentation code, 117
segmentation code app, 354
segmentation code pre, 354
segmentation empty, 117

segmentation engine, 114
segmentation hidden, 127
segmentation style, 127
separator sign, 285
separator sign colon, 285
separator sign dash, 285
separator sign none, 285
shadow, 162
sharp corners, 46
sharpish corners, 47
shield externalize, 95
show bounding box, 153
shrink break goal, 309
shrink tight, 84
sidebyside, 71
sidebyside align, 71
sidebyside gap, 73
size, 42
skin, 112
skin first, 112
skin first is subskin of, 119
skin last, 112
skin last is subskin of, 119
skin middle, 112
skin middle is subskin of, 119
smart shadow arc, 164
space, 54
space to both, 55
space to lower, 54
space to upper, 54
spartan, 214
split, 55
square, 54
squeezed title, 18
squeezed title*, 18
standard, 176
standard jigsaw, 177
step, 93
step and label, 93
store to box array, 327
subtitle style, 20
tabularx, 61
tabularx*, 61
tcbimage comment, 256
tcbox raise, 91
tcbox raise base, 91
tcbox width, 92
tempfile, 91
terminator sign, 287
terminator sign colon, 288
terminator sign dash, 288
terminator sign none, 288
text above listing, 260
text above* listing, 260
text and listing, 254
text fill, 60
text height, 52
text only, 255
text outside listing, 258
text side listing, 258
text width, 33
theorem, 289
theorem name, 289
theorem name and number, 288
theorem number and name, 289
theorem style, 295
tikz, 166
tikz lower, 62
tikz reset, 166
tikz upper, 62
tikznode, 63
tikznode boxed title, 138
tikznode lower, 63
tikznode upper, 63
title, 17
title after break, 307
title code, 118
title code app, 354
title code pre, 354
title empty, 118
title engine, 114
title filled, 26
title hidden, 128
title style, 127
title style image, 128
title style tile, 128
titlerule, 35
titlerule style, 129
toggle enlargement, 83
toggle left and right, 44
top, 40
toprule, 34
toprule at break, 310
topsep at break, 310
toptitle, 40
unbreakable, 307
underlay, 169
underlay boxed title, 170
underlay boxed title pre, 352
underlay broken, 170
underlay broken pre, 352
underlay first, 170
underlay first and middle, 170
underlay first and middle pre, 352
underlay first pre, 352
underlay last, 170
underlay last pre, 352
underlay middle, 170
underlay middle and last, 170
underlay middle and last pre, 352
underlay middle pre, 352
underlay pre, 352
underlay unbroken, 170
underlay unbroken and first, 170
underlay unbroken and first pre, 352
underlay unbroken and last, 170
underlay unbroken and last pre, 352

underlay unbroken pre, 352
upperbox, 21
valign, 32
valign lower, 32
valign scale limit, 32
valign upper, 32
varwidth boxed title, 138
varwidth boxed title*, 138
varwidth upper, 63
verbatim, 356
vfill before first, 311
visible, 21
watermark color, 144
watermark graphics, 140
watermark graphics app, 351
watermark graphics app on, 351
watermark graphics on, 140
watermark graphics pre, 351
watermark graphics pre on, 351
watermark opacity, 142
watermark overzoom, 143
watermark shrink, 143
watermark stretch, 144
watermark text, 139
watermark text app, 350
watermark text app on, 350
watermark text on, 139
watermark text pre, 350
watermark text pre on, 350
watermark tikz, 141
watermark tikz app, 351
watermark tikz app on, 351
watermark tikz on, 141
watermark tikz pre, 351
watermark tikz pre on, 351
watermark zoom, 142
widget, 201
width, 33
/tcb/boxtitle/
xshift, 132
yshift, 132
yshift*, 132
yshifttext, 132
/tcb/doclang/
color, 393
colors, 393
counter, 393
counters, 393
environment, 393
environment content, 393
environments, 393
index, 393
key, 393
keys, 393
length, 393
lengths, 393
new, 393
pageshort, 393
updated, 393
value, 393
values, 393
/tcb/external/
-, 370
!, 370
clear preamble, 377
clear preclass, 377
compiler, 376
environment, 376
environment with percent, 376
externalize, 370
force remake, 370
input source on error, 376
minipage, 376
name, 372
PassOptionsToClass, 377
PassOptionsToPackage, 377
plain, 376
preamble, 377
preamble tcbset, 377
preclass, 377
prefix, 370
runner, 370
runs, 376
safety, 376
/tcb/library/
all, 9
breakable, 8
documentation, 9
external, 9
fitting, 8
hooks, 8
listings, 8
listingsutf8, 8
magazine, 8
many, 9
minted, 8
most, 9
raster, 8
skins, 8
theorems, 8
xparse, 8
/tcb/new/
auto counter, 97
blend into, 101
Crefname, 100
crefname, 100
list inside, 104
list type, 104
no counter, 98
number format, 99
number freestyle, 99
number within, 99
use counter, 98
use counter from, 98
use counter*, 98
/tikz/
fill image opacity, 228
fill image options, 228

fill image scale, 228
fill overzoom image, 224
fill overzoom image*, 224
fill overzoom picture, 224
fill plain image, 222
fill plain image*, 222
fill plain picture, 222
fill shrink image, 226
fill shrink image*, 226
fill shrink picture, 226
fill stretch image, 223
fill stretch image*, 223
fill stretch picture, 223
fill tile image, 227
fill tile image*, 227
fill tile picture, 227
fill tile picture*, 227
fill zoom image, 225
fill zoom image*, 225
fill zoom picture, 225
tcb fill frame, 130
tcb fill interior, 130
tcb fill title, 130
keys key, 393

label key, 93
label type key, 93
last value, 139–141, 311
left key, 38
left value, 29, 238
left skip key, 79
lefthand ratio key, 74
lefthand width key, 73
leftlower key, 38
leftright skip key, 79
leftrule key, 34
lefttitle key, 38
leftupper key, 38
length key, 393
Lengths
\foolength, 383
lengths key, 393
lifted shadow key, 164
lines before break key, 307
list entry key, 94
list inside key, 104
list text key, 94
list type key, 104
listing above comment key, 261
listing above text key, 260
listing above* comment key, 261
listing above* text key, 260
listing and comment key, 257
listing and text key, 254
listing engine key, 254
listing file key, 254
listing inputencoding key, 250
listing only key, 254
listing options key, 249
listing outside comment key, 259
listing outside text key, 258
listing remove caption key, 250
listing side comment key, 259
listing side text key, 258
listing style key, 249
listing utf8 key, 251
listings key, 8
listings value, 101, 254
listingsutf8 key, 8
lower separated key, 24
lowerbox key, 23

magazine key, 8
many key, 9
\marg, 384
margin value, 297
margin apart value, 297
margin break value, 297
marker key, 187
math key, 290
math lower key, 290
math upper key, 290
maximum value, 53
\meta, 384
middle key, 41
middle value, 139–141, 311
middle and last value, 139–141, 311
minimal value, 42
minimum center value, 92
minimum for equal height group key, 57
minimum left value, 92
minimum right value, 92
minipage key, 89, 376
minipage value, 89, 234
minipage boxed title key, 137
minipage boxed title* key, 137
minted key, 8
minted value, 254
minted language key, 252
minted options key, 252
minted style key, 253
most key, 9

name key, 372
natural height key, 51
new key, 393
\newboxarray, 326
\newtcbexternalizeenvironment, 378
\newtcbexternalizetcolorbox, 378
\NewTCBInputListing, 366
\newtcbinputlisting, 248
\NewTCBListing, 364
\newtcblisting, 246
\NewTCBox, 361
\newtcbox, 15
\NewTCBoxFit, 367
\newtcboxfit, 335
\newtcbtheorem, 281
\NewTColorBox, 359
\newtcolorbox, 14

\NewTotalTCBox, 363
\NewTotalTCBoxFit, 368
\NewTotalTColorBox, 360
no borderline key, 153
no counter key, 98
no extras key, 312
no extras first key, 312
no extras last key, 312
no extras middle key, 312
no extras unbroken key, 312
no finish key, 172
no finish first key, 172
no finish last key, 172
no finish middle key, 172
no finish unbroken key, 172
no label type key, 93
no listing options key, 249
no overlay key, 65
no process key, 265
no recording key, 106
no shadow key, 156
no underlay key, 169
no underlay boxed title key, 170
no underlay first key, 170
no underlay last key, 170
no underlay middle key, 170
no underlay unbroken key, 170
no watermark key, 141
nobeforeafter key, 76
nofloat key, 69
none value, 44, 83, 239, 309, 311
noparskip key, 76
nophantom key, 93
normal value, 42
north value, 46, 47
northeast value, 46, 47
northwest value, 46, 47
notitle key, 17
notitle after break key, 307
number format key, 99
number freestyle key, 99
number within key, 99

\oarg, 384
octogon arc key, 36
off value, 344, 391
on value, 344
on line key, 91
only key, 96
opacityback key, 49
opacitybacktitle key, 49
opacityfill key, 49
opacityframe key, 49
opacitylower key, 50
opacitytext key, 50
opacitytitle key, 50
opacityupper key, 50
Option color, 396
outer arc key, 37
overlay key, 64
overlay app key, 348
overlay broken key, 65
overlay broken app key, 349
overlay broken pre key, 349
overlay first key, 65
overlay first and middle key, 65
overlay first and middle app key, 349
overlay first and middle pre key, 349
overlay first app key, 348
overlay first pre key, 348
overlay last key, 65
overlay last app key, 349
overlay last pre key, 349
overlay middle key, 65
overlay middle and last key, 65
overlay middle and last app key, 349
overlay middle and last pre key, 349
overlay middle app key, 349
overlay middle pre key, 349
overlay pre key, 348
overlay unbroken key, 65
overlay unbroken and first key, 65
overlay unbroken and first app key, 349
overlay unbroken and first pre key, 349
overlay unbroken and last key, 65
overlay unbroken and last app key, 349
overlay unbroken and last pre key, 349
overlay unbroken app key, 348
overlay unbroken pre key, 348
oversize key, 43

pad after break key, 310
pad at break key, 310
pad at break* key, 310
pad before break key, 310
pad before break* key, 310
pageshort key, 393
parbox key, 90
parskip key, 76
PassOptionsToClass key, 377
PassOptionsToPackage key, 377
path value, 113, 114
pathfirst value, 113, 114
pathfirstjigsaw value, 113
pathjigsaw value, 113
pathlast value, 113, 114
pathlastjigsaw value, 113
pathmiddle value, 113, 114
pathmiddlejigsaw value, 113
pdf comment key, 256
pdf extension key, 257
\pdfpages, 221
pgf value, 391
pgfchapter value, 391
pgfsection value, 391
phantom key, 93
phantomlabel key, 93
plain key, 376
plain value, 295
plain apart value, 296

preamble key, 377
preamble tcbset key, 377
preclass key, 377
prefix key, 370
process code key, 265
\ProvideTCBInputListing, 366
\ProvideTCBListing, 364
\ProvideTCBox, 361
\ProvideTCBoxFit, 367
\ProvideTColorBox, 359
\ProvideTotalTCBox, 363
\ProvideTotalTCBoxFit, 368
\ProvideTotalTColorBox, 360

raster key, 8
raster after skip key, 236
raster before skip key, 236
raster column n key, 240
raster column skip key, 237
raster columns key, 235
raster equal height key, 239
raster equal height group key, 239
raster equal skip key, 236
raster even column key, 240
raster even number key, 241
raster even row key, 241
raster every box key, 240
raster force size key, 240
raster halign key, 238
raster height key, 236
raster left skip key, 237
raster number n key, 241
raster odd column key, 240
raster odd number key, 241
raster odd row key, 240
raster reset key, 240
raster right skip key, 237
raster row m key, 241
raster row m column n key, 241
raster row skip key, 237
raster rows key, 235
raster valign key, 238
raster width key, 235
record key, 106
\refAux, 388
\refAuxcs, 388
\refCom, 387
\refCom*, 387
\refEnv, 387
\refEnv*, 387
\refKey, 388
\refKey*, 388
remake key, 95
remember key, 167
remember as key, 168
\renewtcbexternalizeenvironment, 378
\renewtcbexternalizetcolorbox, 379
\RenewTCBInputListing, 366
\renewtcbinputlisting, 248
\RenewTCBListing, 364
\renewtcblisting, 247
\RenewTCBox, 361
\renewtcbox, 15
\RenewTCBoxFit, 367
\renewtcboxfit, 335
\renewtcbtheorem, 282
\RenewTColorBox, 359
\renewtcolorbox, 14
\RenewTotalTCBox, 363
\RenewTotalTCBoxFit, 368
\RenewTotalTColorBox, 360
reset key, 95
reset and store to box array key, 329
reset box array key, 326
right key, 39
right value, 29, 238
right skip key, 79
righthand ratio key, 74
righthand width key, 73
rightlower key, 40
rightrule key, 34
righttitle key, 39
rightupper key, 39
rotate key, 167
rounded corners key, 47
rows value, 239
run arara key, 267
run biber key, 267
run bibtex key, 267
run dvips key, 267
run latex key, 267
run lualatex key, 267
run makeindex key, 267
run pdflatex key, 265
run ps2pdf key, 267
run system command key, 265
run xelatex key, 267
runner key, 370
runs key, 376

safety key, 376
savedelimiter key, 25
savelowerto key, 23
saveto key, 22
scale key, 167
scale value, 32
scale* value, 32
segmentation code key, 117
segmentation code app key, 354
segmentation code pre key, 354
segmentation empty key, 117
segmentation engine key, 114
segmentation hidden key, 127
segmentation style key, 127
separator sign key, 285
separator sign colon key, 285
separator sign dash key, 285
separator sign none key, 285
shadow key, 162
sharp corners key, 46

sharpish corners key, 47
shield externalize key, 95
show bounding box key, 153
shrink break goal key, 309
shrink tight key, 84
sidebyside key, 71
sidebyside align key, 71
sidebyside gap key, 73
size key, 42
skin key, 112
skin first key, 112
skin first is subskin of key, 119
skin last key, 112
skin last is subskin of key, 119
skin middle key, 112
skin middle is subskin of key, 119
Skins
    beamer, 196
    beamerfirst, 198
    beamerlast, 200
    beamermiddle, 199
    bicolor, 191
    bicolorfirst, 193
    bicolorlast, 195
    bicolormiddle, 194
    draft, 215
    empty, 205
    emptyfirst, 207
    emptylast, 209
    emptymiddle, 208
    enhanced, 178
    enhanced jigsaw, 185
    enhancedfirst, 182
    enhancedfirst jigsaw, 186
    enhancedlast, 184
    enhancedlast jigsaw, 190
    enhancedmiddle, 183
    enhancedmiddle jigsaw, 187
    freelance, 217
    freelancefirst, 217
    freelancelast, 217
    freelancemiddle, 217
    spartan, 214
    standard, 176
    standard jigsaw, 177
    widget, 201
    widgetfirst, 202
    widgetlast, 204
    widgetmiddle, 203
skins key, 8
small value, 42
smart shadow arc key, 164
south value, 46, 47
southeast value, 46, 47
southwest value, 46, 47
space key, 54
space to both key, 55
space to lower key, 54
space to upper key, 54
spartan key, 214
spartan Skin, 214
spartan value, 113, 114
split key, 55
square key, 54
squeeze value, 342
squeezed title key, 18
squeezed title* key, 18
standard key, 176
standard Skin, 176
standard value, 113, 114, 295
standard jigsaw key, 177
standard jigsaw Skin, 177
step key, 93
step and label key, 93
store to box array key, 327
subtitle style key, 20

tables value, 101
tabularx key, 61
tabularx* key, 61
tcb fill frame key, 130
tcb fill interior key, 130
tcb fill title key, 130
tcbclipframe environment, 146
tcbclipinterior environment, 148
tcbcliptitle environment, 148
\tcbcontinuedraftmode, 175
\tcbcounter, 97
\tcbdocmarginnote, 389
\tcbdocnew, 389
\tcbdocupdated, 389
tcbexternal environment, 371
\tcbEXTERNALIZE, 370
\tcbfitdim, 335–338
\tcbfitsteps, 336
\tcbfontsize, 335
\tcbhighmath, 283
\tcbifexternal, 377
tcbimage comment key, 256
\tcbincludegraphics, 218
\tcbincludepdf, 220
\tcbinputlisting, 245
\tcbinputrecords, 106
\tcbinterruptdraftmode, 175
tcbinvclipframe environment, 147
\tcbitem, 234
tcbitemize environment, 234
\tcbline, 181
\tcbline*, 181
tcblisting environment, 243
\tcblistof, 104
\tcblower, 11
\tcbmakedocSubKey, 387
\tcbmaketheorem, 282
tcboutputlisting environment, 245
\tcbox, 13
tcbox raise key, 91
tcbox raise base key, 91
tcbox width key, 92

\tcboxedtitleheight, 132
\tcboxedtitlewidth, 132
\tcboxfit, 334
\tcboxmath, 283
\tcboxverb, 363
\tcbpatcharcangular, 229
\tcbpatcharcround, 229
tcbraster environment, 233
\tcbrecord, 106
\tcbset, 12
\tcbsetforeverylayer, 12
\tcbsetmacrotoheightofnode, 230
\tcbsetmacrotowidthofnode, 230
\tcbsetmanagedlayer, 87
\tcbsettoheightofnode, 230
\tcbsettowidthofnode, 230
\tcbstartdraftmode, 175
\tcbstartrecording, 106
\tcbstopdraftmode, 175
\tcbstoprecording, 106
\tcbsubskin, 119
\tcbsubtitle, 20
\tcbtitle, 19
\tcbtitletext, 19
\tcbuselibrary, 8
\tcbuselistinglisting, 245
\tcbuselistingtext, 245
\tcbusetemp, 105
\tcbusetemplisting, 245
tcbverbatimwrite environment, 105
tcbwritetemp environment, 105
tcolorbox environment, 11
\tcolorboxenvironment, 16
tempfile key, 91
terminator sign key, 287
terminator sign colon key, 288
terminator sign dash key, 288
terminator sign none key, 288
text above listing key, 260
text above* listing key, 260
text and listing key, 254
text fill key, 60
text height key, 52
text only key, 255
text outside listing key, 258
text side listing key, 258
text width key, 33
theorem key, 289
theorem name key, 289
theorem name and number key, 288
theorem number and name key, 289
theorem style key, 295
theorems key, 8
\thetcbcounter, 97, 98
\thetcbrasternum, 233
tight value, 42
tikz key, 166
tikz lower key, 62
tikz reset key, 166
tikz upper key, 62
tikznode key, 63
tikznode boxed title key, 138
tikznode lower key, 63
tikznode upper key, 63
title key, 17
title value, 42
title after break key, 307
title code key, 118
title code app key, 354
title code pre key, 354
title empty key, 118
title engine key, 114
title filled key, 26
title hidden key, 128
title style key, 127
title style image key, 128
title style tile key, 128
titlerule key, 35
titlerule style key, 129
toggle enlargement key, 83
toggle left and right key, 44
top key, 40
top value, 32, 71, 72, 77, 238
top seam value, 71, 72
toprule key, 34
toprule at break key, 310
topsep at break key, 310
toptitle key, 40
true value, 79, 306

unbreakable key, 307
unbroken value, 139–141
unbroken and first value, 139–141
underlay key, 169
underlay boxed title key, 170
underlay boxed title pre key, 352
underlay broken key, 170
underlay broken pre key, 352
underlay first key, 170
underlay first and middle key, 170
underlay first and middle pre key, 352
underlay first pre key, 352
underlay last key, 170
underlay last pre key, 352
underlay middle key, 170
underlay middle and last key, 170
underlay middle and last pre key, 352
underlay middle pre key, 352
underlay pre key, 352
underlay unbroken key, 170
underlay unbroken and first key, 170
underlay unbroken and first pre key, 352
underlay unbroken and last key, 170
underlay unbroken and last pre key, 352
underlay unbroken pre key, 352
unlimited value, 305, 306
updated key, 393
uphill value, 46, 47
upperbox key, 21

use counter key, 98
use counter from key, 98
use counter* key, 98
\useboxarray, 329
\usetcboxarray, 330

valign key, 32
valign lower key, 32
valign scale limit key, 32
valign upper key, 32
value key, 393
Values
    all, 46, 47, 239, 309, 311
    areasize, 342
    areasize*, 342
    auto, 92
    auto limited, 92
    base, 77
    baselineskip, 309
    bottom, 32, 71, 72, 77, 238
    bottom seam, 71, 72
    break, 296
    broken, 139–141
    center, 29, 32, 71, 72, 77, 238
    center seam, 71, 72
    change, 296
    change apart, 296
    change break, 296
    change standard, 295
    colon, 102
    colon hang, 102
    dash, 102
    dash hang, 102
    doc, 391
    downhill, 46, 47
    east, 46, 47
    empty, 113, 114
    evenpage, 44, 83
    false, 76, 79, 240, 306
    fbox, 42
    figures, 101
    final, 344
    first, 139–141, 311
    first and middle, 139, 311
    fitbox, 89
    flush center, 29, 31
    flush left, 29, 31
    flush right, 29, 31
    fontsize, 342
    fontsize*, 342
    foovalue, 382
    forced, 44, 83
    forced center, 92
    forced left, 92
    forced right, 92
    freelance, 113, 114
    hbox, 89
    hybrid, 342
    hybrid*, 342
    ignored, 23
    invisible, 21, 23
    justify, 29
    last, 139–141, 311
    left, 29, 238
    listings, 101, 254
    margin, 297
    margin apart, 297
    margin break, 297
    maximum, 53
    middle, 139–141, 311
    middle and last, 139–141, 311
    minimal, 42
    minimum center, 92
    minimum left, 92
    minimum right, 92
    minipage, 89, 234
    minted, 254
    none, 44, 83, 239, 309, 311
    normal, 42
    north, 46, 47
    northeast, 46, 47
    northwest, 46, 47
    off, 344, 391
    on, 344
    path, 113, 114
    pathfirst, 113, 114
    pathfirstjigsaw, 113
    pathjigsaw, 113
    pathlast, 113, 114
    pathlastjigsaw, 113
    pathmiddle, 113, 114
    pathmiddlejigsaw, 113
    pgf, 391
    pgfchapter, 391
    pgfsection, 391
    plain, 295
    plain apart, 296
    right, 29, 238
    rows, 239
    scale, 32
    scale*, 32
    small, 42
    south, 46, 47
    southeast, 46, 47
    southwest, 46, 47
    spartan, 113, 114
    squeeze, 342
    standard, 113, 114, 295
    tables, 101
    tight, 42
    title, 42
    top, 32, 71, 72, 77, 238
    top seam, 71, 72
    true, 79, 306
    unbroken, 139–141
    unbroken and first, 139–141
    unlimited, 305, 306
    uphill, 46, 47
    visible, 21, 23

west, 46, 47
values key, 393
varwidth boxed title key, 138
varwidth boxed title* key, 138
varwidth upper key, 63
verbatim key, 356
vfill before first key, 311
visible key, 21
visible value, 21, 23

watermark color key, 144
watermark graphics key, 140
watermark graphics app key, 351
watermark graphics app on key, 351
watermark graphics on key, 140
watermark graphics pre key, 351
watermark graphics pre on key, 351
watermark opacity key, 142
watermark overzoom key, 143
watermark shrink key, 143
watermark stretch key, 144
watermark text key, 139
watermark text app key, 350
watermark text app on key, 350
watermark text on key, 139
watermark text pre key, 350
watermark text pre on key, 350
watermark tikz key, 141
watermark tikz app key, 351
watermark tikz app on key, 351
watermark tikz on key, 141
watermark tikz pre key, 351
watermark tikz pre on key, 351
watermark zoom key, 142
west value, 46, 47
widget key, 201
widget Skin, 201
widgetfirst Skin, 202
widgetlast Skin, 204
widgetmiddle Skin, 203
width key, 33

xparse key, 8
xshift key, 132

yshift key, 132
yshift* key, 132
yshifttext key, 132