

A L^AT_EX Style for Typesetting a Three-Dimensional Product Box*

Gerd Neugebauer

Im Lerchelsbühl 5

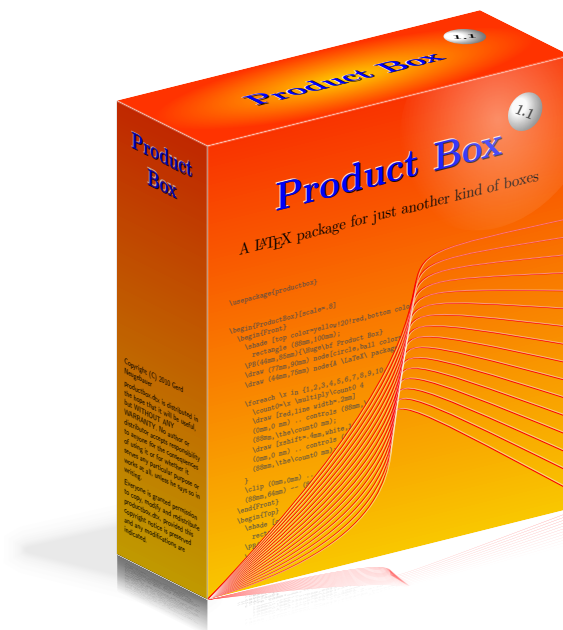
64521 Groß-Gerau (Germany)

Email: gene@gerd-neugebauer.de

Documentation date: 2010/12/29

Abstract

The package `productbox` provides a style file for typesetting a three-dimensional product box. This product box can be rendered as it is standing on a surface and some light is shed onto it. Alternatively it can be typeset as a wireframe to be cut out and glued together. This will lead to a physical product box.



*This file documents `productbox.dtx` version 1.1 (from revision 8333) as of 2010/12/29.

Contents

1	Introduction	3
2	The User Interface	3
2.1	The Simplified User Interface	4
2.2	The Extended User Interface	7
2.3	Settings and Options of the Main Environment	9
3	Tips and Tricks	17
3.1	Adjusting the Paper for the Fold Rendering	17
4	Known Problems	17
5	The Documentation Driver	19
6	The Implementation	19
6.1	The Version Information	19
6.2	Getting Started	20
6.3	Option Declarations	20
6.3.1	General Parameters	20
6.3.2	Box Style Parameters	20
6.3.3	Parameters for the 3D Rendering	21
6.3.4	Parameters for the Fold Rendering	21
6.3.5	Parameters for the Simplified Interface	21
6.4	Storage for the Faces	21
6.5	Settings	22
6.6	The Main Environment	22
6.7	Box Styles	25
6.7.1	Box Style <code>front</code>	25
6.7.2	Box Style <code>back</code>	26
6.7.3	Box Style <code>left</code>	26
6.7.4	Box Style <code>right</code>	26
6.7.5	Box Style <code>top</code>	26
6.7.6	Box Style <code>bottom</code>	26
6.7.7	Box Style <code>fold</code>	27
6.7.8	Box Style <code>threeD</code>	30

1 Introduction

Humans are tied to the physical world. Even in the age of software it is desirable to have a physical representation for it. In the commercial world the software is sold in product boxes containing a CD or DVD and printed documentation.

But even for open source software having a product box provides a means to associate a physical object with the more or less virtual entity “software”.

There are many programs around devoted to producing layouts for product boxes. Each graphics program can be used for this purpose as well. Nevertheless the integration of material from the $\text{T}_{\text{E}}\text{X}$ world is not that easy.

The $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ package `productbox` is an attempt to use the possibilities for $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ and some packages to provide a means for typesetting the faces of a product box and assemble them into a three-dimensional image.

Note that this “image” is in fact a PDF object. It can be scaled without loss of quality¹. It is also possible to extract the text from the product box by cut and paste in an appropriate PDF reader.

2 The User Interface

The $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ style `productbox` is based on `TikZ` at [Tan08] least in version 2.0. It is best used with a $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ variant which is able to produce PDF. If no proper tool chain is used then some of the effects should not be used.

`ProductBox`

The environment `ProductBox` provides the central means for producing a product box. The contents is used to define the appearance of the faces. Finally the selected type of output is produced.

The content of the environment consists mainly of the definition of the six faces. Not all of them need to be defined. If one face is not defined then it appears as an empty rectangle of white color.

Thus you usually want to define the faces. Two approaches are provided to define the content of the faces. The simplified interface does not really require any knowledge of the underlying `TikZ` package. Just some knowledge of $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ is sufficient. The extended interface opens the full power to the user. Both interface types can be freely intermixed. The tow kinds of interfaces are described in section 2.1 and 2.2.

The environment `ProductBox` can be controlled with a number of optional parameters. Those parameters are described in section 2.3.

¹unless some pixel images have been included

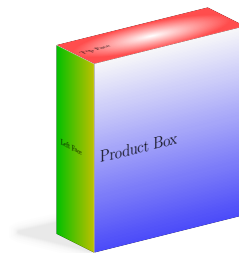
2.1 The Simplified User Interface

The simplified user interface is meant for someone not familiar with *TikZ*. It encapsulates nearly anything and relies on just some basic *L^AT_EX* experience. As a downside the functionality is restricted. Any fancy artwork on the box background can hardly be achieved.

The following example shows a complete – even rather useless – definition of a product box with the simplified user interface.

```
\begin{ProductBox}
  \begin{FrontFace}[bottom color=white!30!blue,top color=white]
    \Huge Product Box
  \end{FrontFace}
  \begin{BackFace}[top color=yellow!30!red,bottom color=white]
    \large Back Face
  \end{BackFace}
  \begin{TopFace}[outer color=white!30!red,inner color=white]
    \large Top Face
  \end{TopFace}
  \begin{BottomFace}[outer color=red,inner color=black]
    \large Bottom Face
  \end{BottomFace}
  \begin{LeftFace}[left color=green,right color=yellow]
    \large Left Face
  \end{LeftFace}
  \begin{RightFace}[left color=white,right color=black]
    \large Right Face
  \end{RightFace}
\end{ProductBox}
```

This code renders as



The content of the environment `ProductBox` is simply evaluated. It can contain any code you like. Useful for the production of a product box are some inner environments. They are called `FrontFace`, `BackFace`, `LeftFace`, `RightFace`, `TopFace`, and `BottomFace`. They are defined inside the main environment only. Those environments can be used to specify the contents of the respective faces of the box.

Note that in the three-dimensional rendering will show at most three of the faces. It does not hurt to define all of them, even if they are not shown at all.

`FrontFace` The environment `FrontFace` is used to define the content of the front face. The

environment processes its contents inside a `minipage` of the default width 88 mm reduced by the left and right separator width (`faceSep`).

The minipage is centered vertically on the face. Usually anything extending the default height of 100 mm is clipped.

```
\begin{FrontFace}
...
\end{FrontFace}
```

BackFace The environment `BackFace` is used to define the content of the back face. The environment processes its contents inside a `minipage` of the default width 88 mm reduced by the left and right separator width (`faceSep`).

The minipage is centered vertically on the face. Usually anything extending the default height of 100 mm is clipped.

```
\begin{BackFace}
...
\end{BackFace}
```

LeftFace The environment `LeftFace` is used to define the content of the left face. The environment processes its contents inside a `minipage` of the default width 100 mm reduced by the left and right separator width (`faceSep`).

The minipage is centered vertically on the face. Usually anything extending the default height of 30 mm is clipped.

```
\begin{LeftFace}
...
\end{LeftFace}
```

RightFace The environment `RightFace` is used to define the content of the right face. The environment processes its contents inside a `minipage` of the default width 100 mm reduced by the left and right separator width (`faceSep`).

The minipage is centered vertically on the face. Usually anything extending the default height of 30 mm is clipped.

```
\begin{RightFace}
...
\end{RightFace}
```

TopFace The environment `TopFace` is used to define the content of the top face. The environment processes its contents inside a `minipage` of the default width 88 mm reduced by the left and right separator width (`faceSep`).

The minipage is centered vertically on the face. Usually anything extending the default height of 30 mm is clipped.

```
\begin{TopFace}
...
\end{TopFace}
```

BottomFace The environment `BottomFace` is used to define the content of the bottom face.

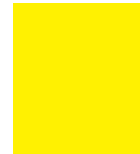
The environment processes its contents inside a `minipage` of the default width 88 mm reduced by the left and right separator width (`faceSep`).

The minipage is centered vertically on the face. Usually anything extending the default height of 30 mm is clipped.

```
\begin{BottomFace}
...
\end{BottomFace}
```

Any of the face defining environments described above can take an optional argument. This argument is used to specify the background. In the simplest case you just have one background color. This is specified with the keyword `color`.

```
\begin{ProductBox}
  \begin{FrontFace}
    [color=yellow]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}
```



Colors in `TikZ` are either one of the named colors or a composition of those colors. The notation `red!60!blue` denotes the color by mixing 60% red and 40% blue.

```
\begin{ProductBox}
  \begin{FrontFace}
    [color=red!60!blue]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}
```



By mixing in black or white you can come to a lighter or darker color.

```
\begin{ProductBox}
  \begin{FrontFace}
    [color=red!20!white]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}
```



A fading from top to bottom can be specified with two colors named `top color` and `bottom color`.

```
\begin{ProductBox}
  \begin{FrontFace}
    [top color=red!50!blue,bottom color=yellow]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}
```



A fading from left to right can be specified with two colors named `left color` and `right color`.

```

\begin{ProductBox}
  \begin{FrontFace}
    [left color=red!50!blue,right color=yellow]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}

```

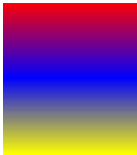


The parameter `middle color` can be used in horizontal or vertical fading to specifying the color in the middle. Note thta it has to be specified after the other colors!

```

\begin{ProductBox}
  \begin{FrontFace}
    [top color=red,bottom color=yellow,middle color=blue]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}

```

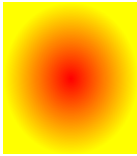


A circular fading can be specified with the color names `inner color` and `outer color`.

```

\begin{ProductBox}
  \begin{FrontFace}
    [inner color=red,outer color=yellow]
    ...
  \end{FrontFace}
  ...
\end{ProductBox}

```



2.2 The Extended User Interface

The extended user interface allows you to use all features of `TikZ`. For this purpose another set of face defining environments is provided which process their content in a `tikzpicture` environment.

The following example shows a complete example of a product box with the extended user interface. This example is used below to demonstrate the effect of the differnt options.

```

\begin{ProductBox}\sf
\begin{Front}
\fill [top color=white!30!blue,bottom color=white]
rectangle (88mm,100mm);
\fill [bottom color=white!40!blue,top color=white!90!blue]
(0mm,0mm) -- (44mm,90mm) -- (88mm,0mm) -- cycle;
\draw (44mm,40mm) node{\huge Product Box};
\end{Front}
\begin{Back}
\fill [top color=yellow!30!red,bottom color=white]
rectangle (88mm,100mm);
\fill [top color=white,bottom color=blue]
(0mm,0mm) -- (88mm,100mm) -- (0mm,100mm) --cycle;
\draw (44mm,50mm) node{\large Back};
\end{Back}
\begin{Top}
\fill [outer color=white!30!red,inner color=white]
rectangle (88mm,30mm);
\draw[white,thick] (0mm,5mm) -- (88mm,25mm);
\draw (44mm,15mm) node{\large Top};
\end{Top}
\begin{Bottom}
\fill [outer color=red!30!white,inner color=black]
rectangle (88mm,30mm);
\draw[white] (44mm,15mm) node{\large Bottom};
\end{Bottom}
\begin{Left}
\fill [left color=green,right color=yellow]
rectangle (30mm,100mm);
\draw (15mm,50mm) node{\large Left};
\end{Left}
\begin{Right}
\fill [top color=green,bottom color=yellow]
rectangle (30mm,100mm);
\draw (15mm,50mm) node{\large Right};
\end{Right}
\end{ProductBox}

```

The content of the environment is simply expanded. It may contain any code you like – except an `ProductBox` environment. Useful for the production of a product box are some inner environments. They are called `Front`, `Back`, `Left`, `Right`, `Top`, and `Bottom`. They are defined inside the main environment only. Those environments can be used to specify the contents of the respective faces of the box.

Note that in the three-dimensional rendering will show at most three of the faces. It does not hurt to define all of them, even if they are not shown at all.

Front The environment `Front` is used to define the content of the front face. The environment processes its contents inside a `tikzpicture` of the default size 88 mm×100 mm. Usually anything outside of this range is clipped.

```

\begin{Front}
...
\end{Front}

```

Back The environment `Back` is used to define the content of the back face. The

environment processes its contents inside a `tikzpicture` of the default size 88 mm×100 mm. Usually anything outside of this range is clipped.

```
\begin{Back}  
...  
\end{Back}
```

`Left` The environment `Left` is used to define the content of the left face, i.e. the face left to the front page. The environment processes its contents inside a `tikzpicture` of the default size 30 mm×100 mm. Usually anything outside of this range is clipped.

```
\begin{Left}  
...  
\end{Left}
```

`Right` The environment `Right` is used to define the content of the right face, i.e. the face right to the front page. The environment processes its contents inside a `tikzpicture` of the default size 30 mm×100 mm. Usually anything outside of this range is clipped.

```
\begin{Right}  
...  
\end{Right}
```

`Top` The environment `Top` is used to define the content of the top face. The environment processes its contents inside a `tikzpicture` of the default size 88 mm×30 mm. Usually anything outside of this range is clipped.

```
\begin{Top}  
...  
\end{Top}
```

`Bottom` The environment `Bottom` is used to define the content of the top face. The environment processes its contents inside a `tikzpicture` of the default size 88 mm×30 mm. Usually anything outside of this range is clipped.

```
\begin{Bottom}  
...  
\end{Bottom}
```

2.3 Settings and Options of the Main Environment

The environment `ProductBox` can take some options to influence the appearance of the product box. Those options are comma separated.

```
\begin{ProductBox}[shape=3d]
```

The settings are local to the main environment. If an option is not set then the fallback from the global settings are used.

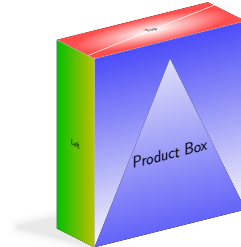
`\ProductBoxSet` The macro `\ProductBoxSet` modifies the global setting of the product box style. The arguments are the same as the optional arguments of the environment `ProductBox` – but enclosed in braces instead of brackets.

```
\ProductBoxSet{shape=3d}
```

The following options can be used to influence the result of the product box.

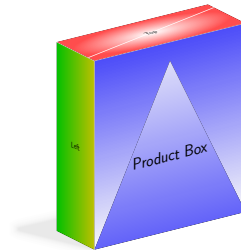
style The box style determines, how the box is drawn. Several rendering functions are provided to produce different effects. Any value is accepted. Unknown box styles will lead to an error message.

```
\begin{ProductBox}[style=3D]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```



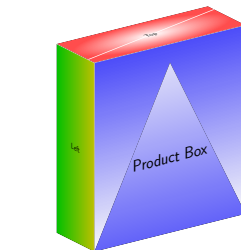
3D The box style 3D is the default. It produces a three-dimensional view of the box. The option 3D is an abbreviation for `style=3D`. It can also be written as `3d` or `threeD`.

```
\begin{ProductBox}
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```



shadow The option `shadow` controls the drawing of the drop shadow in the 3D rendering. It is a boolean value taking the values `true` and `false`. The default value is `true`. The option `shadow` is the abbreviation for `shadow=true`.

```
\begin{ProductBox}[shadow=false]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```

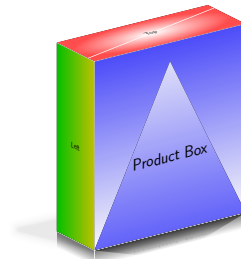


mirror The option `mirror` controls the rendering of the mirror effect in the 3D rendering. It is a boolean value taking the values `true` and `false`. The default value is `false`. The option `mirror` is the abbreviation for `mirror=true`.

```

\begin{ProductBox}[mirror=true]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



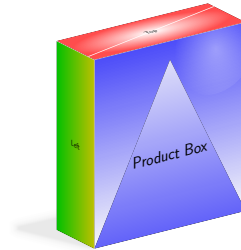
`flare` The option `flare` controls the rendering of the flare effect in the 3D rendering. The flare is a circular reflection of the light source in the upper right corner of the front face. The option is a boolean value taking the values `true` and `false`. The default value is `false`.

The option `flare` is the abbreviation for `flare=true`.

```

\begin{ProductBox}[flare=true]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```

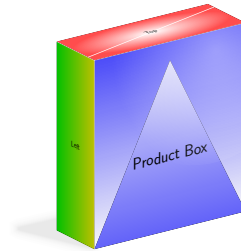


`flareDiameter` The option `flareDiameter` takes a dimension which defines the diameter of the flare effect in the 3D rendering. The default value is 24mm.

```

\begin{ProductBox}[flare=true,
  flareDiameter=60mm]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



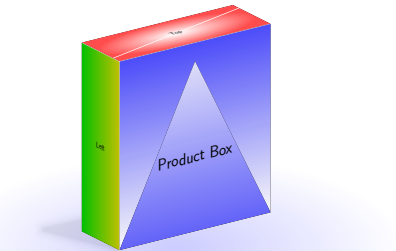
Internally the 3D rendering uses a `tikzpicture`. You can expand your own code either at the beginning or at the end of this environment. This can be achieved by overwriting a macro.

`ProductBoxThreeDStartHook` The macro `ProductBoxThreeDStartHook` contains code to be expanded at the beginning of the 3D rendering. Initially it is defined as empty.

```

\renewcommand\ProductBoxThreeDStartHook{
  \filldraw[yscale=.4,
    shift={(30mm,4mm)},
    color=white,
    inner color=white!70!blue,
    outer color=white]
    circle (120mm);
}
\begin{ProductBox}
...
\end{ProductBox}

```



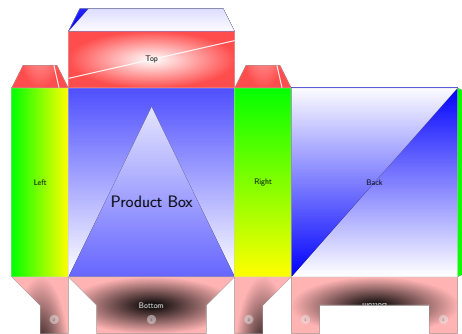
`ProductBoxThreeDEndHook` The macro `ProductBoxThreeDEndHook` contains code to be expanded at the end of the 3D rendering. Initially it is defined as empty.

`fold` The option `fold` switches to the box style for rendering a complete wire frame with all faces in place. The option `fold` is an abbreviation for `style=fold`.

```

\begin{ProductBox}[fold]
  \begin{Front}
  ...
  \end{Front}
  \begin{Left}
  ...
  \end{Left}
  ...
\end{ProductBox}

```



The top face is printed on the left and right top ear as well. This should avoid a break in the pattern when the box is partially opened. The same principle applies for the glue ear on the right side and the bottom.

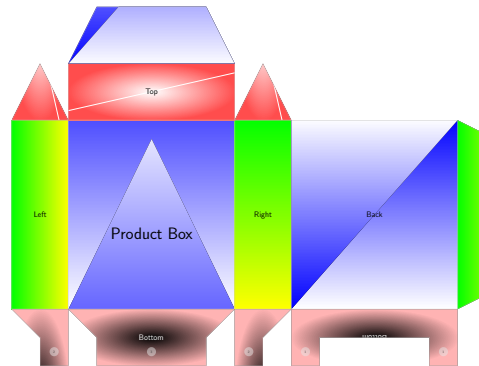
The bottom is formed in a way to maximize stability without the need to glue. In addition numbers are printed on the parts of the bottom indicating the sequence in which the parts should be closed.

`earSize` The option `earSize` takes a dimension which defines the width of the ears in the fold rendering. The ears around the top are this wide. The width of the glueing ear is half of this size. The default value is 12 mm.

```

\begin{ProductBox}[fold,earSize=30mm]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



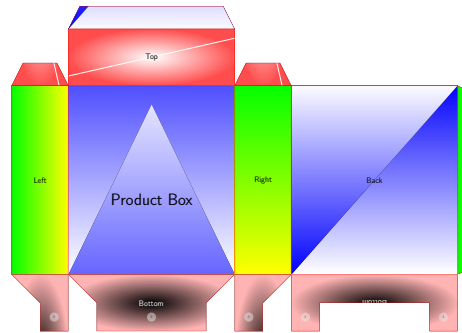
Note that the ear size must not exceed the width of the left face, the width of the front face, and the height of the box. Otherwise funny effects in the ears will happen.

`foldLine` The option `foldLine` takes a specification for the line surrounding the fold drawing. Usually you want to simply use a color like “gray” or “red”. The default is a kind of gray.

```

\begin{ProductBox}[fold,foldLine=red]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```

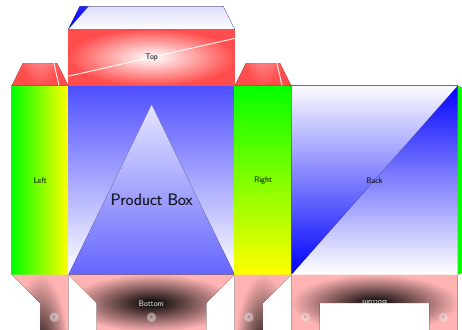


`foldOpacity` The option `foldOpacity` takes a fraction for the opacity of the line surrounding the fold drawing. The default is 0.5. If you want to let the fold lines disappear then use a value of 1.

```

\begin{ProductBox}[fold,foldOpacity=0]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

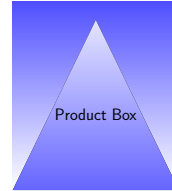
```



`front` The option `front` switches to the box style for rendering the front face only. The

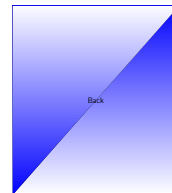
option `front` is an abbreviation for `style=front`.

```
\begin{ProductBox}[front]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```



`back` The option `back` switches to the box style for rendering the back face only. The option `back` is an abbreviation for `style=back`.

```
\begin{ProductBox}[back]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```



`left` The option `left` switches to the box style for rendering the left face only. The option `left` is an abbreviation for `style=left`.

```
\begin{ProductBox}[left]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```



`right` The option `right` switches to the box style for rendering the right face only. The option `right` is an abbreviation for `style=right`.

```
\begin{ProductBox}[right]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}
```



`top` The option `top` switches to the box style for rendering the top face only. The option `top` is an abbreviation for `style=top`.

```

\begin{ProductBox}[top]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



`bottom` The option `bottom` switches to the box style for rendering the face only. The option `bottom` is an abbreviation for `style=bottom`.

```

\begin{ProductBox}[bottom]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



`empty` The option `empty` switches to the box style for not rendering the box at all. The option `empty` is an abbreviation for `style=empty`.

```

\begin{ProductBox}[empty]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

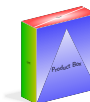
```

`scale` The option `scale` controls the scaling of the whole rendering. It is a number where 1. represents the original size.²

```

\begin{ProductBox}[scale=.1]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



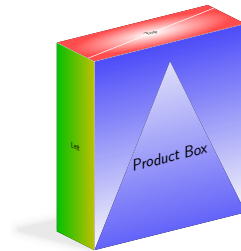
`view` The option `view` takes a name of a view definition and activates the appropriate settings. A few views are predefined. The default value is 1.

²The examples on the right side are normally rendered with a scale of 0.25.

```

\begin{ProductBox}[view=1]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

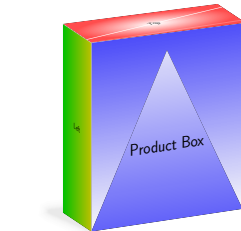
```



```

\begin{ProductBox}[view=2]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

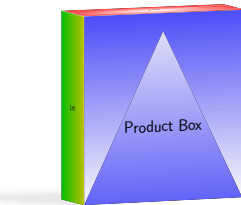
```



```

\begin{ProductBox}[view=3]
  \begin{Front}
    ...
  \end{Front}
  \begin{Left}
    ...
  \end{Left}
  ...
\end{ProductBox}

```



- edgeColor** The option `edgeColor` takes a color specification for highlighting the inner edges in the 3D rendering.
- faceSep** The option `faceSep` takes a dimension denoting the additional separating white-space between the outer border and the minipage in the simplified interface.
- width** The option `width` takes a dimension denoting the width of the box. The default value is 88 mm.
- height** The option `height` takes a dimension denoting the height of the box. The default value is 100 mm.
- depth** The option `depth` takes a dimension denoting the depth of the box. The default value is 30 mm.
- clean** The option `clean` controls the cleaning of the stored faces upon entering the main environment. If it is turned off then the previously defined faces are still present and do not need to be defined again. It is a boolean value taking the values `true` and `false`. The default value is `true`.

This option can be used to typeset the same product box with different parameters. For this purpose the main environment `ProductBox` is left empty and the option `clean=false` is added.
- clip** The option `clip` controls the clipping of the faces to their defined size. If it is

turned off then the faces can be oversized leading to strange effects. It is a boolean value taking the values `true` and `false`. The default value is `true`.

3 Tipps and Tricks

3.1 Adjusting the Paper for the Fold Rendering

When you produce the fold rendering it is usually meant to be cut out and glued together. This means that the normal rules for the paper layout are not relevant. Instead you want to use the complete page for printing the product box. Below an example is shown how this goal can be achieved.

```
\documentclass{report}
\usepackage[a4paper,
            landscape,
            left=5mm,
            right=5mm,
            top=5mm,
            bottom=5mm]{geometry}
\usepackage{productbox}
% load any required packages here
\pagestyle{empty}
\begin{document}
\noindent
\begin{ProductBox}[style=fold]
% include any definitions for the faces here
\end{ProductBox}
\end{document}
```

The class for typesetting this example is `report`. This can be changed to suit your needs. For instance if you are used to a document class with other macros and environments predefined you can just use it instead.

The example above uses the package `geometry` [Ume07] to get rid of any predefined page layout. Some parameters (marked in red) can be adjusted. First of all is the paper definition. Here the value `a4paper` is used. If you want to print onto paper of a different size just use an appropriate short name like `letterpaper` or `a3paper`. See the documentation of the `geometry` package for a complete list of values.

The values `left`, `right`, `top`, and `bottom` denote the margins left on the respective outer side of the paper. They are set to 5 mm in this example to cope with the problem that some printers are not able to fill the complete page. They might need a small non-printable area at the borders. You can experiment and adjust those values to whatever suits your printer.

4 Known Problems

This section lists some issues which might lead to undesirable results.

Nested ProductBox environments. The definition of the environment uses some global storage. As a consequence the environment `ProductBox` can not be used inside the definition of a face. For instance if you want to show a product box on a side of another product box can lead to this problem.

In this case you can simply store the inner product box in a box register (with `setbox`) and use this box register instead of a direct rendering. This will overcome the restriction.

Free selection of the point of view. The 3D rendering is rather limited in the possibilities of selecting the view. Arbitrary rotation about any axis is not implemented (yet). This is on the which list for a future release already.

References

- [Tan08] Till Tantau. *TikZ & PGF: Manual for Version 2.0*, February 2008. <http://sourceforge.net/projects/pgf>.
- [Ume07] Hideo Umeki. *The geometry package*, July 2007.

5 The Documentation Driver

The documentation driver changes `productbox.dtx` into a self-extracting documentation. Thus it is possible to run \LaTeX on `productbox.dtx` to produce the documentation.

The documentation can be adapted in a file named `productbox.dcf` (documentation configuration). This file can contain instructions for `docstrip`. Especially useful might be the instruction

`\OnlyDescription`

which suppresses the generation of the implementation description.

```
1 <*driver>
2 \documentclass{ltxdoc}
3 \usepackage{productbox}
4 \usepackage[colorlinks,citecolor=blue]{hyperref}
5 \usepackage{graphicx,color}
6 \RecordChanges
7 \EnableCrossrefs
8 \CodelineIndex
9 \definecolor{darkblue}{rgb}{.4,.4,1.}
10 \renewcommand\MacroFont{\tt\footnotesize\color{darkblue}}
11 \parindent=0pt
12 \parskip=1ex plus .5ex minus .25ex
13 \InputIfFileExists{productbox.dcf}{}{}
14 \begin{document}
15 \DocInput{productbox.dtx}
16 \end{document}
17 </driver>
```

6 The Implementation

The implementation contains the code of the style.

6.1 The Version Information

The following lines define the version information for the class file. The information is partially taken from the version control system (Subversion).

```
18 <*style>
19 \begingroup
20 \def\ProductBox@VC$#1: #2 #3$#2}
21 \def\ProductBox@VCdate$#1: #2-#3-#4 #5$#2/#3/#4}
22 \xdef\fileversion{1.1}
23 \xdef\filerevision{\ProductBox@VC$Revision: 8333 $}
24 \xdef\filedate{\ProductBox@VCdate
25   $Date: 2010-12-29 20:58:06 +0100 (Mi, 29 Dez 2010) $}
26 \xdef\filename{productbox.dtx}
27 \endgroup
28 </style>
```

6.2 Getting Started

First we have to determine that the right kind of L^AT_EX is running and identify the style file.

```
29 <*style>
30 \NeedsTeXFormat{LaTeX2e}
31 \ProvidesPackage{productbox}[\filedate Another type of boxes...]
```

A bunch of packages is loaded to form the base of the work herein.

```
32 \RequirePackage{keyval}
33 \RequirePackage{tikz}
34 \usetikzlibrary{calc}
35 \usetikzlibrary{fadings}
```

6.3 Option Declarations

Define the parameters for the keyval package. They are used in the main environment `Product Box` and in the declaration of global options `\ProductBoxSet`.

6.3.1 General Parameters

```
36 \define@key{ProductBox}{scale}{%
37   \def\ProductBox@scale{#1}}
38 \define@key{ProductBox}{width}{%
39   \def\ProductBox@x{#1}}
40 \define@key{ProductBox}{height}{%
41   \def\ProductBox@y{#1}}
42 \define@key{ProductBox}{depth}{%
43   \def\ProductBox@z{#1}}
44 \define@key{ProductBox}{clean}[true]{%
45   \csname ProductBox@clean#1\endcsname}
```

6.3.2 Box Style Parameters

```
46 \define@key{ProductBox}{flat}[true]{%
47   \def\ProductBox@style{flat}}
48 \define@key{ProductBox}{fold}[true]{%
49   \def\ProductBox@style{fold}}
50 \define@key{ProductBox}{3d}[true]{%
51   \def\ProductBox@style{threeD}}
52 \define@key{ProductBox}{3D}[true]{%
53   \def\ProductBox@style{threeD}}
54 \define@key{ProductBox}{threeD}[true]{%
55   \def\ProductBox@style{threeD}}
56 \define@key{ProductBox}{top}[true]{%
57   \def\ProductBox@style{top}}
58 \define@key{ProductBox}{bottom}[true]{%
59   \def\ProductBox@style{bottom}}
60 \define@key{ProductBox}{front}[true]{%
61   \def\ProductBox@style{front}}
62 \define@key{ProductBox}{back}[true]{%
63   \def\ProductBox@style{back}}
```

```

64 \define@key{ProductBox}{left}[true]{%
65   \def\ProductBox@style{left}}
66 \define@key{ProductBox}{right}[true]{%
67   \def\ProductBox@style{right}}
68 \define@key{ProductBox}{empty}[true]{%
69   \def\ProductBox@style{empty}}
70 \define@key{ProductBox}{style}{%
71   \def\ProductBox@style{#1}}

```

6.3.3 Parameters for the 3D Rendering

```

72 \define@key{ProductBox}{shadow}[true]{%
73   \csname ProductBox@shadow#1\endcsname}
74 \define@key{ProductBox}{mirror}[true]{%
75   \csname ProductBox@mirror#1\endcsname}
76 \define@key{ProductBox}{flare}[true]{%
77   \csname ProductBox@flare#1\endcsname}
78 \define@key{ProductBox}{flareDiameter}{%
79   \def\ProductBox@flareDiameter{#1}}
80 \define@key{ProductBox}{edgeColor}{%
81   \def\productBox@edgeColor{#1}}
82 \define@key{ProductBox}{view}{%
83   \@nameuse{ProductBox@View#1}}

```

6.3.4 Parameters for the Fold Rendering

```

84 \define@key{ProductBox}{earSize}{%
85   \def\ProductBox@earSize{#1}}
86 \define@key{ProductBox}{foldLine}{%
87   \def\ProductBox@foldLine{#1}}
88 \define@key{ProductBox}{foldOpacity}{%
89   \def\ProductBox@foldOpacity{#1}}

```

6.3.5 Parameters for the Simplified Interface

```

90 \define@key{ProductBox}{faceSep}{%
91   \def\ProductBox@FaceSep{#1}}

```

6.4 Storage for the Faces

`\ProductBox@Front` The box `\ProductBox@Front` contains the front material.

```
92 \newbox\ProductBox@Front
```

`\ProductBox@Left` The box `\ProductBox@Left` contains the left material.

```
93 \newbox\ProductBox@Left
```

`\ProductBox@Top` The box `\ProductBox@Top` contains the top material.

```
94 \newbox\ProductBox@Top
```

`\ProductBox@Right` The box `\ProductBox@Right` contains the right material.

```
95 \newbox\ProductBox@Right
```

`\ProductBox@Bottom` The box `\ProductBox@Bottom` contains the bottom material.
96 `\newbox\ProductBox@Bottom`

`\ProductBox@Back` The box `\ProductBox@Back` contains the back material.
97 `\newbox\ProductBox@Back`

6.5 Settings

`\ProductBoxSet` The macro `\ProductBoxSet` defines the global parameters used by the environment `ProductBox`. They can be overwritten either within a group or in the optional parameter of the environment.
98 `\newcommand\ProductBoxSet[1]{\setkeys{ProductBox}{#1}}`

6.6 The Main Environment

`\ProductBox@style` The default style is the 3D rendering.
99 `\newcommand\ProductBox@style{threeD}`

`\ProductBox@x` The width of the box.
100 `\newcommand\ProductBox@x{88mm}`

`\ProductBox@y` The height of the box.
101 `\newcommand\ProductBox@y{100mm}`

`\ProductBox@z` The depth of the box.
102 `\newcommand\ProductBox@z{30mm}`

`ifProductBox@clean` Indicator that the boxes for the faces should be cleaned.
103 `\newif\ifProductBox@clean`

`ifProductBox@active` Indicator that we are inside a `ProductBox` environment already.
104 `\newif\ifProductBox@active`

`ifProductBox@clip` The boolean `ProductBox@clip` determines whether or not the additional clipping of the boxes should be enabled.
105 `\newif\ifProductBox@clip \ProductBox@cliptrue`

`\ProductBox@FaceSep` The horizontal separator of the minipage in the face definitions in the simplified interface.

```
106 \newcommand\ProductBox@FaceSep{1em}
```

`ProductBox` This is the central environment provided by this style. In the begin code only the local environments are initialized and the optional parameters are evaluated with the help of the package `keyval`.

```
107 \newenvironment{ProductBox}[1][]{%  
108 \setkeys{ProductBox}{#1}%
```

Next we clean the faces if this is required.

```
109 \ifProductBox@clean  
110 \global\setbox\ProductBox@Front\hbox{}%  
111 \global\setbox\ProductBox@Back\hbox{}%  
112 \global\setbox\ProductBox@Left\hbox{}%  
113 \global\setbox\ProductBox@Right\hbox{}%  
114 \global\setbox\ProductBox@Top\hbox{}%  
115 \global\setbox\ProductBox@Bottom\hbox{}%  
116 \fi
```

Next we define the local environments to make sure that they have the proper definitions within this environment. Since the environment provides an implicit group, the definitions are local to this environment.

To suppress any error messages about environments which are already defined the start macros are reset to undefined.

```
117 \ifProductBox@active  
118 \errmessage{Trying to use an environment ProductBox inside the  
119 environment ProductBox. This is not allowed.}%  
120 \else  
121 \ProductBox@activetrue  
122 \fi  
123 \let\Front\undefined \let\endFront\undefined  
124 \let\Back\undefined \let\endBack\undefined  
125 \let\Left\undefined \let\endLeft\undefined  
126 \let\Right\undefined \let\endRight\undefined  
127 \let\Top\undefined \let\endTop\undefined  
128 \let\Bottom\undefined \let\endBottom\undefined  
129 \let\FrontFace\undefined \let\endFrontFace\undefined  
130 \let\BackFace\undefined \let\endBackFace\undefined  
131 \let\LeftFace\undefined \let\endLeftFace\undefined  
132 \let\RightFace\undefined \let\endRightFace\undefined  
133 \let\TopFace\undefined \let\endTopFace\undefined  
134 \let\BottomFace\undefined \let\endBottomFace\undefined  
135 \newenvironment{Front}{\ProductBox@Start\ProductBox@Front  
136 (\ProductBox@x,\ProductBox@y)}{\ProductBox@End}%  
137 \newenvironment{Back}{\ProductBox@Start\ProductBox@Back  
138 (\ProductBox@x,\ProductBox@y)}{\ProductBox@End}%  
139 \newenvironment{Left}{\ProductBox@Start\ProductBox@Left  
140 (\ProductBox@z,\ProductBox@y)}{\ProductBox@End}%  
141 \newenvironment{Right}{\ProductBox@Start\ProductBox@Right  
142 (\ProductBox@z,\ProductBox@y)}{\ProductBox@End}%
```

```

143 \newenvironment{Top}{\ProductBox@Start\ProductBox@Top
144   (\ProductBox@x,\ProductBox@z)}{\ProductBox@End}%
145 \newenvironment{Bottom}{\ProductBox@Start\ProductBox@Bottom(
146   \ProductBox@x,\ProductBox@z)}{\ProductBox@End}%
147 \newenvironment{FrontFace}[1][white]{\ProductBox@StartFace\ProductBox@Front
148   {\ProductBox@x}{\ProductBox@y}{##1}}{\ProductBox@EndFace}%
149 \newenvironment{BackFace}[1][white]{\ProductBox@StartFace\ProductBox@Back
150   {\ProductBox@x}{\ProductBox@y}{##1}}{\ProductBox@EndFace}%
151 \newenvironment{LeftFace}[1][white]{\ProductBox@StartFace\ProductBox@Left
152   {\ProductBox@z}{\ProductBox@y}{##1}}{\ProductBox@EndFace}%
153 \newenvironment{RightFace}[1][white]{\ProductBox@StartFace\ProductBox@Right
154   {\ProductBox@z}{\ProductBox@y}{##1}}{\ProductBox@EndFace}%
155 \newenvironment{TopFace}[1][white]{\ProductBox@StartFace\ProductBox@Top
156   {\ProductBox@x}{\ProductBox@z}{##1}}{\ProductBox@EndFace}%
157 \newenvironment{BottomFace}[1][white]{\ProductBox@StartFace\ProductBox@Bottom
158   {\ProductBox@x}{\ProductBox@z}{##1}}{\ProductBox@EndFace}%
159 \ignorespacesafterend
160 }{%

```

Check that the box style is defined or issue an appropriate error message.

```

161 \@ifundefined{ProductBox@style@\ProductBox@style}{%
162   \errmessage{Box style '\ProductBox@style' for ProductBox is unknown}%
163 }{%

```

The main activity is performed in the end code. Since the flexibility of the environment is one of its design goals, the expansion of the macro `\ProductBox@style` is used to invoke the macro stored in it. Optionally it is enclosed in a `\scalebox` macro to perform the scaling. Thus the implementations of the box styles do not need to care about scaling at all.

```

164   \@ifundefined{ProductBox@scale}{%
165     \@nameuse{ProductBox@style@\ProductBox@style}}{%
166     \scalebox{\ProductBox@scale}%
167       {\@nameuse{ProductBox@style@\ProductBox@style}}
168   }}%
169 \ProductBox@activefalse
170 \ignorespacesafterend
171 }

```

`\ProductBox@Start` The macro `\ProductBox@Start` starts the environment storing a face.

```

172 \def\ProductBox@Start#1(#2){%
173   \global\setbox#1\hbox\bgroup\begin{tikzpicture}%
174     \ifProductBox@clip \clip rectangle (#2); \fi
175   }%

```

`\ProductBox@End` The macro `\ProductBox@End` ends the environment storing a face.

```

176 \newcommand\ProductBox@End{\end{tikzpicture}\egroup\ignorespacesafterend}

```

`\ProductBox@StartFace` The macro `\ProductBox@StartFace` starts the environment storing a face in a minipage.

```

177 \newcommand\ProductBox@StartFace[4]{%

```



```

178 \let\ProductBox@@box#1%
179 \def\ProductBox@@w{#2}%
180 \def\ProductBox@@h{#3}%
181 \def\ProductBox@@args{[#4]}%
182 \@tempdima=#2
183 \advance\@tempdima-\ProductBox@FaceSep
184 \advance\@tempdima-\ProductBox@FaceSep
185 \setbox#1\hbox\bgroup\begin{minipage}{\the\@tempdima}%
186 \ignorespaces
187 }

```

`\ProductBox@EndFace` This macro ends a face definition and stores the result in the internal box `\ProductBox@@box`.

```

188 \newcommand\ProductBox@EndFace{\end{minipage}\egroup
189 \global\setbox\ProductBox@@box=\hbox{\begin{tikzpicture}
190 \ifProductBox@clip \clip rectangle (\ProductBox@@w,\ProductBox@@h); \fi
191 \expandafter\fill \ProductBox@@args rectangle(\ProductBox@@w,\ProductBox@@h);
192 \draw (.5*\ProductBox@@w,.5*\ProductBox@@h)
193 node[inner sep=0pt] {\box\ProductBox@@box};
194 \end{tikzpicture}}\ignorespacesafterend}

```

6.7 Box Styles

`\ProductBox@style@empty` This macro defines the style *empty* for a product box. It simply does nothing.

```

195 \newcommand\ProductBox@style@empty{}

```

`\ProductBox@style@flat` This macro defines the style *flat* for a product box.

```

196 \newcommand\ProductBox@style@flat{%
197 \par\noindent
198 \hspace*{\ProductBox@z}\kern1mm \copy\ProductBox@Top\[[1ex]
199 \copy\ProductBox@Left\kern1mm \copy\ProductBox@Front\kern1mm
200 \copy\ProductBox@Right\kern1mm \copy\ProductBox@Back\[[1ex]
201 \hspace*{\ProductBox@z}\kern1mm \copy\ProductBox@Bottom\par
202 }

```

6.7.1 Box Style front

`\ProductBox@style@front` This macro defines the style *front* for a product box. Only this one face of the box is shown.

```

203 \newcommand\ProductBox@style@front{%
204 \copy\ProductBox@Front
205 }

```

6.7.2 Box Style back

`\ProductBox@style@back` This macro defines the style *back* for a product box. Only this one face of the box is shown.

```
206 \newcommand\ProductBox@style@back{%class
207   \copy\ProductBox@Back
208 }
```

6.7.3 Box Style left

`\ProductBox@style@left` This macro defines the style *left* for a product box. Only this one face of the box is shown.

```
209 \newcommand\ProductBox@style@left{%
210   \copy\ProductBox@Left
211 }
```

6.7.4 Box Style right

`\ProductBox@style@right` This macro defines the style *right* for a product box. Only this one face of the box is shown.

```
212 \newcommand\ProductBox@style@right{%
213   \copy\ProductBox@Right
214 }
```

6.7.5 Box Style top

`\ProductBox@style@top` This macro defines the style *top* for a product box. Only this one face of the box is shown.

```
215 \newcommand\ProductBox@style@top{%
216   \copy\ProductBox@Top
217 }
```

6.7.6 Box Style bottom

`\ProductBox@style@bottom` This macro defines the style *bottom* for a product box. Only this one face of the box is shown.

```
218 \newcommand\ProductBox@style@bottom{%
219   \copy\ProductBox@Bottom
220 }
```

6.7.7 Box Style fold

`\ProductBox@earSize` Parameter for the size of the ears.

```
221 \newcommand\ProductBox@earSize{12mm}
```

`\ProductBox@foldLine` The color of the additional lines in the fold rendering.

```
222 \newcommand\ProductBox@foldLine{white!80!black}
```

`\ProductBox@foldOpacity` The color of the additional lines in the fold rendering.

```
223 \newcommand\ProductBox@foldOpacity{.5}
```

`\ProductBox@style@fold` This macro defines the style *fold* for a product box.

```
224 \newcommand\ProductBox@style@fold{%
225   \begin{tikzpicture}[sw/.style={anchor=south west,
226                               inner sep=0pt},
227                       se/.style={anchor=south east,
228                               inner sep=0pt},
229                       nw/.style={anchor=north west,
230                               inner sep=0pt},
231                       num/.style={circle,
232                               fill=white!90!black,
233                               fill opacity=.5,
234                               font=\tiny\bfseries\sffamily}]
```

left face

```
235   \begin{scope}
236     \clip rectangle (\ProductBox@z,\ProductBox@y);
237     \draw node[sw]{\copy\ProductBox@Left};
238     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
239       rectangle (\ProductBox@z,\ProductBox@y);
240   \end{scope}
```

front face

```
241   \begin{scope}[xshift=\ProductBox@z]
242     \clip rectangle (\ProductBox@x,\ProductBox@y);
243     \draw node[sw]{\copy\ProductBox@Front};
244     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
245       rectangle (\ProductBox@x,\ProductBox@y);
246   \end{scope}
```

right face

```
247   \begin{scope}[xshift=\ProductBox@z+\ProductBox@x]
248     \clip rectangle (\ProductBox@z,\ProductBox@y);
249     \draw node[sw]{\copy\ProductBox@Right};
250     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
251       rectangle (\ProductBox@z,\ProductBox@y);
252   \end{scope}
```

back face

```
253 \begin{scope}[xshift=\ProductBox@x+2*\ProductBox@z]
254 \clip rectangle (\ProductBox@x,\ProductBox@y);
255 \draw node[sw]{\copy\ProductBox@Back};
256 \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
257 rectangle (\ProductBox@x,\ProductBox@y);
258 \end{scope}
```

top face

```
259 \begin{scope}[xshift=\ProductBox@z,yshift=\ProductBox@y]
260 \clip rectangle (\ProductBox@x,\ProductBox@z);
261 \draw node[sw]{\copy\ProductBox@Top};
262 \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
263 rectangle (\ProductBox@x,\ProductBox@z);
264 \end{scope}
```

bottom ear 3 front

```
265 \begin{scope}[xshift=\ProductBox@z]
266 \clip
267 (0mm,0mm) -- (.5*\ProductBox@z,-.5*\ProductBox@z) --
268 (.5*\ProductBox@z,-\ProductBox@z) --
269 (\ProductBox@x-.5*\ProductBox@z,-\ProductBox@z) --
270 (\ProductBox@x-.5*\ProductBox@z,-.5*\ProductBox@z) --
271 (\ProductBox@x,0mm) -- cycle;
272 \draw (\ProductBox@x,-\ProductBox@z)
273 node[se]{\copy\ProductBox@Bottom};
274 \draw [thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
275 (0mm,0mm) -- (.5*\ProductBox@z,-.5*\ProductBox@z) --
276 (.5*\ProductBox@z,-\ProductBox@z) --
277 (\ProductBox@x-.5*\ProductBox@z,-\ProductBox@z) --
278 (\ProductBox@x-.5*\ProductBox@z,-.5*\ProductBox@z) --
279 (\ProductBox@x,0mm) -- cycle;
280 \draw (.5*\ProductBox@x,-.75*\ProductBox@z) node[num]{3};
281 \end{scope}
```

bottom ear 1 back

```
282 \begin{scope}[xshift=\ProductBox@x+2*\ProductBox@z]
283 \clip
284 (0mm,0mm) -- (0mm,-\ProductBox@z) --
285 (.5*\ProductBox@z,-\ProductBox@z) --
286 (.5*\ProductBox@z,-.5*\ProductBox@z) --
287 (\ProductBox@x-.5*\ProductBox@z,-.5*\ProductBox@z) --
288 (\ProductBox@x-.5*\ProductBox@z,-\ProductBox@z) --
289 (\ProductBox@x,-\ProductBox@z) -- (\ProductBox@x,0mm) --
290 cycle;
291 \draw node[se,rotate=180]{\copy\ProductBox@Bottom};
292 \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
293 (0mm,0mm) -- (0mm,-\ProductBox@z) --
294 (.5*\ProductBox@z,-\ProductBox@z) --
295 (.5*\ProductBox@z,-.5*\ProductBox@z) --
296 (\ProductBox@x-.5*\ProductBox@z,-.5*\ProductBox@z) --
297 (\ProductBox@x-.5*\ProductBox@z,-\ProductBox@z) --
298 (\ProductBox@x,-\ProductBox@z) -- (\ProductBox@x,0mm) --
299 cycle;
300 \draw (.25*\ProductBox@z,-.75*\ProductBox@z) node[num]{1};
```

```

301     \draw (\ProductBox@x-.25*\ProductBox@z,-.75*\ProductBox@z)
302           node[num]{1};
303   \end{scope}

bottom ear 2 right

304   \begin{scope}[xshift=\ProductBox@z+\ProductBox@x]
305     \clip
306       (0mm,0mm) -- (\ProductBox@z,0mm) --
307       (.5*\ProductBox@z,-.5*\ProductBox@z) --
308       (.5*\ProductBox@z,-\ProductBox@z) -- (0mm,-\ProductBox@z) --
309       cycle;
310     \draw (\ProductBox@z,0mm)
311           node[se,rotate=90]{\copy\ProductBox@Bottom};
312     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
313       (0mm,0mm) -- (\ProductBox@z,0mm) --
314       (.5*\ProductBox@z,-.5*\ProductBox@z) --
315       (.5*\ProductBox@z,-\ProductBox@z) -- (0mm,-\ProductBox@z) --
316       cycle;
317     \draw (.25*\ProductBox@z,-.75*\ProductBox@z) node[num]{2};
318   \end{scope}

bottom ear 2 left

319   \begin{scope}
320     \clip
321       (0mm,0mm) -- (\ProductBox@z,0mm) --
322       (\ProductBox@z,-\ProductBox@z) --
323       (.5*\ProductBox@z,-\ProductBox@z) --
324       (.5*\ProductBox@z,-.5*\ProductBox@z) -- cycle;
325     \draw
326       node[sw,rotate=270]{\copy\ProductBox@Bottom};
327     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
328       (0mm,0mm) -- (\ProductBox@z,0mm) --
329       (\ProductBox@z,-\ProductBox@z) --
330       (.5*\ProductBox@z,-\ProductBox@z) --
331       (.5*\ProductBox@z,-.5*\ProductBox@z) -- cycle;
332     \draw (.75*\ProductBox@z,-.75*\ProductBox@z) node[num]{2};
333   \end{scope}

glue ear

334   \begin{scope}[xshift=2*\ProductBox@x+2*\ProductBox@z]
335     \clip
336       (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize/4) --
337       (\ProductBox@earSize/2,\ProductBox@y-\ProductBox@earSize/4) --
338       (0mm,\ProductBox@y) -- cycle;
339     \draw node[sw]{\copy\ProductBox@Left};
340     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
341       (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize/4) --
342       (\ProductBox@earSize/2,\ProductBox@y-\ProductBox@earSize/4) --
343       (0mm,\ProductBox@y) -- cycle;
344   \end{scope}

top ear left

345   \begin{scope}[yshift=\ProductBox@y]
346     \clip
347       (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize) --

```

```

348     (\ProductBox@z-\ProductBox@earSize/2,\ProductBox@earSize) --
349     (\ProductBox@z,0mm) -- cycle;
350     \draw node[nw,rotate=90]{\copy\ProductBox@Top};
351     \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
352     (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize) --
353     (\ProductBox@z-\ProductBox@earSize/2,\ProductBox@earSize) --
354     (\ProductBox@z,0mm) -- cycle;
355     \end{scope}

```

top ear right

```

356     \begin{scope}[xshift=\ProductBox@x+\ProductBox@z,
357                 yshift=\ProductBox@y]
358         \clip
359         (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize) --
360         (\ProductBox@z-\ProductBox@earSize/2,\ProductBox@earSize) --
361         (\ProductBox@z,0mm) -- cycle;
362         \draw node[se,rotate=270]{\copy\ProductBox@Top};
363         \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
364         (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize) --
365         (\ProductBox@z-\ProductBox@earSize/2,\ProductBox@earSize) --
366         (\ProductBox@z,0mm) -- cycle;
367     \end{scope}

```

top ear back

```

368     \begin{scope}[xshift=\ProductBox@z,
369                 yshift=\ProductBox@y+\ProductBox@z]
370         \clip
371         (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize) --
372         (\ProductBox@x-\ProductBox@earSize/2,\ProductBox@earSize) --
373         (\ProductBox@x,0mm) -- cycle;
374         \draw (0mm,\ProductBox@y)
375         node[se,rotate=180]{\copy\ProductBox@Back};
376         \draw[thin,opacity=\ProductBox@foldOpacity,\ProductBox@foldLine]
377         (0mm,0mm) -- (\ProductBox@earSize/2,\ProductBox@earSize) --
378         (\ProductBox@x-\ProductBox@earSize/2,\ProductBox@earSize) --
379         (\ProductBox@x,0mm) -- cycle;
380     \end{scope}
381 \end{tikzpicture}}

```

6.7.8 Box Style threeD

`\ProductBoxThreeDStartHook` A hook to add some code at the beginning.

```
382 \newcommand\ProductBoxThreeDStartHook{}
```

`\ProductBoxThreeDEndHook` A hook to add some code at the end.

```
383 \newcommand\ProductBoxThreeDEndHook{}
```

`\ifProductBox@shadow` The boolean `ProductBox@shadow` determines whether or not the shadow in the 3D rendering is shown.

```
384 \newif\ifProductBox@shadow \ProductBox@shadowtrue
```

`\ifProductBox@mirror` The boolean `ProductBox@mirror` determines whether or not the mirror effect in the 3D rendering is shown.

```
385 \newif\ifProductBox@mirror
```

`\ifProductBox@flare` The boolean `ProductBox@flare` determines whether or not the flare effect in the 3D rendering is shown.

```
386 \newif\ifProductBox@flare
```

`\ProductBox@edgeColor` The edge color in the 3D rendering.

```
387 \newcommand\ProductBox@edgeColor{white}
```

`\ProductBox@flareDiameter` The diameter of the flare.

```
388 \newcommand\ProductBox@flareDiameter{24mm}
```

The angle to rotate the box about the z axis.

```
389 \newcommand\ProductBox@angleZ{8}
```

The angle to rotate the box about the x axis.

```
390 \newcommand\ProductBox@angleX{10}
```

`\ProductBox@style@threeD` This macro defines the style *threeD* for a product box. The faces are placed such the illusion of a three-dimensional box appears. In addition a shadow is shown if not disabled.

```
391 \newcommand\ProductBox@style@threeD{
392   \begin{tikzpicture}[sw/.style={anchor=south west,
393     inner sep=Opt}]
394     \ProductBoxThreeDStartHook
395     \ifProductBox@mirror \ProductBox@threeD@mirror \fi
396     \ifProductBox@shadow \ProductBox@threeD@shadow \fi
397
398     \begin{scope}
399       \fill[white,
400         yslant=\ProductBox@p@front@yslant,
401         xscale=\ProductBox@p@front@xscale,
402         yscale=\ProductBox@p@front@yscale,
403         anchor=south west]
404         rectangle (\ProductBox@x,\ProductBox@y);
405     \draw
406       node[sw,
407         yslant=\ProductBox@p@front@yslant,
408         xscale=\ProductBox@p@front@xscale,
409         yscale=\ProductBox@p@front@yscale]
410       {\copy\ProductBox@Front};
411     \fill[black,
412       fill opacity=.025,
413       yslant=\ProductBox@p@front@yslant,
414       xscale=\ProductBox@p@front@xscale,
```

```

415         yscale=\ProductBox@p@front@yscale,
416         anchor=south west]
417     rectangle (\ProductBox@x,\ProductBox@y);
418 \ifProductBox@flare
419     \begin{scope}[
420         yslant=\ProductBox@p@front@yslant,
421         xscale=\ProductBox@p@front@xscale,
422         yscale=\ProductBox@p@front@yscale]
423         \clip rectangle(\ProductBox@x,\ProductBox@y);
424         \ProductBox@threeD@flare
425     \end{scope}
426     \fi
427 \end{scope}
428 \begin{scope}[shift={(-\ProductBox@p@left@xscale*\ProductBox@z,
429         \ProductBox@p@left@yslant
430         *\ProductBox@p@left@xscale
431         *\ProductBox@z)}]
432     \fill[white,
433         yslant=-\ProductBox@p@left@yslant,
434         xscale=\ProductBox@p@left@xscale,
435         yscale=\ProductBox@p@left@yscale,
436         anchor=south west]
437         rectangle (\ProductBox@z,\ProductBox@y);
438     \draw
439         node[sw,
440             yslant=-\ProductBox@p@left@yslant,
441             xscale=\ProductBox@p@left@xscale,
442             yscale=\ProductBox@p@left@yscale]
443             {\copy\ProductBox@Left};
444     \fill[black,
445         fill opacity=.25,
446         yslant=-\ProductBox@p@left@yslant,
447         xscale=\ProductBox@p@left@xscale,
448         yscale=\ProductBox@p@left@yscale,
449         anchor=south west]
450         rectangle (\ProductBox@z,\ProductBox@y);
451 \end{scope}
452 \begin{scope}[yshift=\ProductBox@y]
453     \fill [white,
454         anchor=south west,
455         xslant=-\ProductBox@p@top@xslant,
456         xscale=\ProductBox@p@top@xscale,
457         yslant=\ProductBox@p@top@yslant,
458         yscale=\ProductBox@p@top@yscale]
459         rectangle (\ProductBox@x,\ProductBox@z);
460     \draw
461         node[sw,
462             xslant=-\ProductBox@p@top@xslant,
463             xscale=\ProductBox@p@top@xscale,
464             yslant=\ProductBox@p@top@yslant,
465             yscale=\ProductBox@p@top@yscale]
466             {\copy\ProductBox@Top};
467     \fill [black,
468         fill opacity=.0025,
469         anchor=south west,
470         xslant=-\ProductBox@p@top@xslant,
471         xscale=\ProductBox@p@top@xscale,

```



```

472         yslant=\ProductBox@p@top@yslant,
473         yscale=\ProductBox@p@top@yscale]
474     rectangle (\ProductBox@x,\ProductBox@z);
475 \end{scope}
476 \begin{scope}[shift={(-\ProductBox@p@left@xscale*\ProductBox@z,
477 \ProductBox@p@left@yslant*\ProductBox@p@left@xscale*\ProductBox@z)}]
478     \draw [\ProductBox@edgeColor,
479         line width=.5mm,
480         draw opacity=.25,
481         yslant=-\ProductBox@p@left@yslant,
482         xscale=\ProductBox@p@left@xscale]
483         (\ProductBox@z,0mm) --
484         (\ProductBox@z,\ProductBox@y) --
485         (0mm,\ProductBox@y);
486 \end{scope}
487 \draw [\ProductBox@edgeColor,
488     line width=.5mm,
489     draw opacity=.25,
490     yslant=\ProductBox@p@front@yslant,
491     xscale=\ProductBox@p@front@xscale,
492     anchor=south west]
493     (0mm,\ProductBox@y) -- (\ProductBox@x,\ProductBox@y);
494 \ProductBoxThreeDEndHook
495 \end{tikzpicture}}

```

Parameters and views.

```

496 \newcommand\ProductBox@setFrontParams[3]{%
497     \def\ProductBox@p@front@yslant{#1}%
498     \def\ProductBox@p@front@xscale{#2}%
499     \def\ProductBox@p@front@yscale{#3}%
500 }
501 \newcommand\ProductBox@setLeftParams[3]{%
502     \def\ProductBox@p@left@yslant{#1}%
503     \def\ProductBox@p@left@xscale{#2}%
504     \def\ProductBox@p@left@yscale{#3}%
505 }
506 \newcommand\ProductBox@setTopParams[4]{%
507     \def\ProductBox@p@top@xslant{#1}%
508     \def\ProductBox@p@top@yslant{#2}%
509     \def\ProductBox@p@top@xscale{#3}%
510     \def\ProductBox@p@top@yscale{#4}%
511 }
512 \newcommand\ProductBox@setMirrorParams[1]{%
513     \def\ProductBox@p@mirror@yscale{#1}%
514 }
515 \@namedef{ProductBox@View@1}{%
516     \ProductBox@setFrontParams{.25}{.9090909}{1}%
517     \ProductBox@setLeftParams{.5}{.6666666}{1}%
518     \ProductBox@setTopParams{2}{.2265}{1.363}{.3333333}%
519     \ProductBox@setMirrorParams{.4}}
520 \@namedef{ProductBox@View@2}{%
521     \ProductBox@setFrontParams{.15}{.9090909}{1}%
522     \ProductBox@setLeftParams{.6666666}{.5}{1}%
523     \ProductBox@setTopParams{1.5}{.12}{1.11}{.3333333}%
524     \ProductBox@setMirrorParams{.4}}
525 \@namedef{ProductBox@View@3}{%

```

```

526 \ProductBox@setFrontParams{.05}{.95}{1}%
527 \ProductBox@setLeftParams{.25}{.4}{1}%
528 \ProductBox@setTopParams{4}{.04}{1.133333}{.1}%
529 \ProductBox@setMirrorParams{.4}}
530 \@nameuse{ProductBox@View@1}

```

`\ProductBox@threeD@shadow` This macro defines the code to produce the shadow effect for the 3D rendering.

```

531 \newcommand\ProductBox@threeD@shadow{
532   \begin{scope}
533     \foreach \x in {.5,1,1.5,2,2.5,3,4,5,6,8,10} {
534       \filldraw [black,
535                 line width=\x mm,
536                 rounded corners=2mm,
537                 opacity=.01,
538                 shift={(-1mm,1mm)}]
539         (0mm,0mm) --
540         (-\ProductBox@p@top@xslant
541          * \ProductBox@p@left@xscale
542          * \ProductBox@z,
543          \ProductBox@p@top@yscale
544          * \ProductBox@z) --
545         (0mm,
546          \ProductBox@p@top@yscale * \ProductBox@z +
547          \ProductBox@p@top@xslant
548          * \ProductBox@p@left@xscale
549          * \ProductBox@p@top@yslant
550          * \ProductBox@z) --
551         cycle;
552     }
553   \end{scope}
554 }

```

The following fading is used for the mirror effect in the 3D rendering.

```

555 \tikzfading[name=ProductBoxFade,
556             top color=transparent!100,
557             bottom color=transparent!50,
558             middle color=transparent!100]

```

`\ProductBox@threeD@mirror` This macro defines the code to produce the mirror effect.

```

559 \newcommand\ProductBox@threeD@mirror{%
560   \fill
561     node[anchor=south west,
562          inner sep=0pt,
563          yslant=\ProductBox@p@front@yslant,
564          xscale=\ProductBox@p@front@xscale,
565          yscale=-\ProductBox@p@mirror@yscale]
566     {\begin{tikzpicture}
567       \clip rectangle (\ProductBox@x,
568                       \ProductBox@y * \ProductBox@p@mirror@yscale);
569       \fill node [scope fading=ProductBoxFade,
570                 yscale=\ProductBox@p@mirror@yscale,
571                 inner sep=0pt]{\copy\ProductBox@Front};

```

```

572         \end{tikzpicture}};
573 \fill[xshift=-\ProductBox@p@left@xscale*\ProductBox@z,
574       yshift=\ProductBox@p@top@yscale*\ProductBox@z]
575   node[anchor=south west,
576        inner sep=0pt,
577        yslant=-\ProductBox@p@left@yslant,
578        xscale=\ProductBox@p@left@xscale,
579        yscale=-\ProductBox@p@mirror@yscale]
580   {\begin{tikzpicture}
581     \clip rectangle (\ProductBox@z,
582                    \ProductBox@y*\ProductBox@p@mirror@yscale);
583     \fill node [scope fading=ProductBoxFade,
584               yscale=\ProductBox@p@mirror@yscale,
585               inner sep=0pt]{\copy\ProductBox@Left};
586   \end{tikzpicture}};
588 }

```

`\ProductBox@threeD@flare` This macro defines the code to produce the flare effect. The flare is achieved with overlaying a partially transparent and fading circle of white color.

```

589 \newcommand\ProductBox@threeD@flare{
590   \fill [white,path fading=ProductBoxFlare]
591     (.8*\ProductBox@x,.9*\ProductBox@y)
592     circle(\ProductBox@flareDiameter);
593 }

```

The following fading is used for the flare effect in the 3D rendering.

```

594 \tikzfading[name=ProductBoxFlare,
595            inner color=transparent!60,
596            outer color=transparent!100]

```

Finally we define some variant names.

```

597 \@namedef{ProductBox@style@3D}{\ProductBox@style@threeD}
598 \@namedef{ProductBox@style@3d}{\ProductBox@style@threeD}
599 </style>

```

That's all.

Change History

1.0				
	General: First public release.	1		
1.1				
	General: Simplified user interface added.	1		
	\ProductBox@style@empty: New box style “empty”.	25		
	\ProductBox@style@threeD: Fix: The flare is also transformed and clipped.	31		
			\ProductBox@threeD@mirror: Code rewritten to make full use of transparency.	34
			ProductBox: Do not clean the faces upon request.	23
			Issue an error message for nested ProductBox environments. . .	23
			Simplified user interface added.	23
			Suppress error messages.	23

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

B	P
<code>\Back</code> 124	<code>ProductBox</code> (environment) <u>107</u>
<code>\BackFace</code> 130	<code>\ProductBox@args</code> 181, 191
<code>\bfseries</code> 234	<code>\ProductBox@box</code> 178, 189, 193
<code>\Bottom</code> 128	<code>\ProductBox@ch</code> 180, 190–192
<code>\BottomFace</code> 134	<code>\ProductBox@cw</code> 179, 190–192
	<code>\ProductBox@activefalse</code> 169
	<code>\ProductBox@activetrue</code> 121
E	<code>\ProductBox@angleX</code> 390
<code>\endBack</code> 124	<code>\ProductBox@angleZ</code> 389
<code>\endBackFace</code> 130	<code>\ProductBox@Back</code> <u>97</u> ,
<code>\endBottom</code> 128	111, 137, 149, 200, 207, 255, 375
<code>\endBottomFace</code> 134	<code>\ProductBox@Bottom</code> <u>96</u> , 115, 145,
<code>\endFront</code> 123	157, 201, 219, 273, 291, 311, 326
<code>\endFrontFace</code> 129	<code>\ProductBox@cliptrue</code> 105
<code>\endLeft</code> 125	<code>\ProductBox@earSize</code>
<code>\endLeftFace</code> 131 85, <u>221</u> , 336, 337, 341,
<code>\endRight</code> 126	342, 347, 348, 352, 353, 359,
<code>\endRightFace</code> 132	360, 364, 365, 371, 372, 377, 378
<code>\endTop</code> 127	<code>\ProductBox@edgeColor</code> <u>387</u> , 478, 487
<code>\endTopFace</code> 133	<code>\productBox@edgeColor</code> 81
environments:	<code>\ProductBox@End</code>
<code>ProductBox</code> <u>107</u> 136, 138, 140, 142, 144, 146, <u>176</u>
	<code>\ProductBox@endFace</code>
F 148, 150, 152, 154, 156, 158, <u>188</u>
<code>\filedate</code> 24, 31	<code>\ProductBox@FaceSep</code> <u>91</u> , <u>106</u> , 183, 184
<code>\filename</code> 26	<code>\ProductBox@flareDiameter</code> 79, <u>388</u> , 592
<code>\filerevision</code> 23	<code>\ProductBox@foldLine</code> 87, <u>222</u> ,
<code>\fileversion</code> 22	238, 244, 250, 256, 262, 274,
<code>\footnotesize</code> 10	292, 312, 327, 340, 351, 363, 376
<code>\Front</code> 123	<code>\ProductBox@foldOpacity</code> 89, <u>223</u> ,
<code>\FrontFace</code> 129	238, 244, 250, 256, 262, 274,
	292, 312, 327, 340, 351, 363, 376
I	<code>\ProductBox@Front</code> <u>92</u> , 110,
<code>\ifProductBox@active</code> 104, <u>104</u> , 117	135, 147, 199, 204, 243, 410, 571
<code>\ifProductBox@clean</code> 103, <u>103</u> , 109	<code>\ProductBox@Left</code> <u>93</u> , 112, 139,
<code>\ifProductBox@clip</code> 105, <u>105</u> , 174, 190	151, 199, 210, 237, 339, 443, 585
<code>\ifProductBox@flare</code> <u>386</u> , 418	<code>\ProductBox@p@front@xscale</code>
<code>\ifProductBox@mirror</code> <u>385</u> , 395 401, 408, 414, 421, 491, 498, 564
<code>\ifProductBox@shadow</code> <u>384</u> , 396	<code>\ProductBox@p@front@yscale</code>
 402, 409, 415, 422, 499
L	<code>\ProductBox@p@front@yslant</code>
<code>\Left</code> 125 400, 407, 413, 420, 490, 497, 563
<code>\LeftFace</code> 131	

\ProductBox@p@left@xscale	\ProductBox@VC	20, 23
. 428, 430, 434, 441, 447, 476,	\ProductBox@VCdate	21, 24
477, 482, 503, 541, 548, 573, 578	\ProductBox@x	
\ProductBox@p@left@yscale 39, <u>100</u> , 136, 138, 144, 146,	
. 435, 442, 448, 504	148, 150, 156, 158, 242, 245,	
\ProductBox@p@left@yslant . . 429,	247, 253, 254, 257, 260, 263,	
433, 440, 446, 477, 481, 502, 577	269–272, 277–280, 282, 287–	
\ProductBox@p@mirror@yscale	289, 296–298, 301, 304, 334,	
. 513, 565, 568, 570, 579, 582, 584	356, 372, 373, 378, 379, 404,	
\ProductBox@p@top@xscale	417, 423, 459, 474, 493, 567, 591	
. 456, 463, 471, 509	\ProductBox@y 41, <u>101</u> ,	
\ProductBox@p@top@xslant	136, 138, 140, 142, 148, 150,	
. 455, 462, 470, 507, 540, 547	152, 154, 236, 239, 242, 245,	
\ProductBox@p@top@yscale	248, 251, 254, 257, 259, 337,	
. 458, 465, 473, 510, 543, 546, 574	338, 342, 343, 345, 357, 369,	
\ProductBox@p@top@yslant	374, 404, 417, 423, 437, 450,	
. 457, 464, 472, 508, 549	452, 484, 485, 493, 568, 582, 591	
\ProductBox@Right	\ProductBox@z 43, <u>102</u> , 140,	
. . <u>95</u> , 113, 141, 153, 200, 213, 249	142, 144, 146, 152, 154, 156,	
\ProductBox@scale 37, 166	158, 198, 201, 236, 239, 241,	
\ProductBox@setFrontParams	247, 248, 251, 253, 259, 260,	
. 496, 516, 521, 526	263, 265, 267–270, 272, 275–	
\ProductBox@setLeftParams	278, 280, 282, 284–289, 293–	
. 501, 517, 522, 527	298, 300, 301, 304, 306–308,	
\ProductBox@setMirrorParams	310, 313–315, 317, 321–324,	
. 512, 519, 524, 529	328–332, 334, 348, 349, 353,	
\ProductBox@setTopParams	354, 356, 360, 361, 365, 366,	
. 506, 518, 523, 528	368, 369, 428, 431, 437, 450,	
\ProductBox@shadowtrue 384	459, 474, 476, 477, 483, 484,	
\ProductBox@Start	542, 544, 546, 550, 573, 574, 581	
. 135, 137, 139, 141, 143, 145, <u>172</u>	\ProductBoxSet <u>98</u>	
\ProductBox@StartFace	\ProductBoxThreeDEndHook . . . <u>383</u> , 494	
. 147, 149, 151, 153, 155, 157, <u>177</u>	\ProductBoxThreeDStartHook . <u>382</u> , 394	
\ProductBox@style 47, 49,		
51, 53, 55, 57, 59, 61, 63, 65,		
67, 69, 71, <u>99</u> , 161, 162, 165, 167		
\ProductBox@style@back <u>206</u>		
\ProductBox@style@bottom <u>218</u>		
\ProductBox@style@empty <u>195</u>		
\ProductBox@style@flat <u>196</u>		
\ProductBox@style@fold <u>224</u>		
\ProductBox@style@front <u>203</u>		
\ProductBox@style@left <u>209</u>		
\ProductBox@style@right <u>212</u>		
\ProductBox@style@threeD <u>391</u> , 597, 598		
\ProductBox@style@top <u>215</u>		
\ProductBox@threeD@flare . . . 424, <u>589</u>		
\ProductBox@threeD@mirror . . 395, <u>559</u>		
\ProductBox@threeD@shadow . . 396, <u>531</u>		
\ProductBox@Top <u>94</u> , 114, 143,		
155, 198, 216, 261, 350, 362, 466		
	R	
	\Right 126	
	\RightFace 132	
	S	
	\sffamily 234	
	T	
	\tiny 234	
	\Top 127	
	\TopFace 133	
	U	
	\undefined 123–134	
	X	
	\x 533, 535	