The tabularborder package *

Johannes Fink

April 15, 2010

Abstract

This package changes the space for horizontal lines at the left side and the right side of a `tabular` (but not `array`) to zero space.

Contents

1 Introduction

Usually, the `tabular` environment adds a space of `\tabcolsep` on the left side and on the right side of the text. For this reason a `\hline`, or a `\toprule`, etc. will exceed the text by the total length of 2 `\tabcolsep`.

col 1	col 2	col 3	col4
multicol 1-2		multicol 3-4	

*This document corresponds to tabularborder v1.0a, dated 2010/04/14.

```
\begin{tabular}{llll}
\toprule
col 1 & col 2 & col 3 & col4\\
\multicolumn{2}{l}{multicol 1-2}
& \multicolumn{2}{l}{multicol 3-4}\\
\bottomrule
\end{tabular}
```

If some publishers, or authors, respectively, do not like these additional length of the horizontal lines, the `tabular` must be reformatted.

col 1	col 2	col 3	col4
multicol 1-2		multicol 3-4	

```
\begin{tabular}{@{}llll@{}}
\toprule
col 1 & col 2 & col 3 & col4\\
\multicolumn{2}{@{}l}{multicol 1-2}
& \multicolumn{2}{l@{}}{multicol 3-4}\\
\bottomrule
\end{tabular}
```

Formatting is done with `@{}`, as shown above. Note that the `\multicolumn` commands must be formatted separately. In a longer project, such as a thesis or book, this reformatting may be a time consuming and a tedious task. In particular, if the supervisor of a thesis work changes his opinion about the layout several times in the course of correcting.

This package changes the leading and trailing spaces automatically into zero without the need of inserting the `@{}`.

col 1	col 2	col 3	col4
multicol 1-2		multicol 3-4	

Of course, with such an arrangement, outer vertical lines do not make sense, and actually, they are displaced if tried anyway. By the way, we are using here the package `booktabs`.

2 Usage

Here are some examples:

2.1 Basic Example

col 1	col 2	col 3	col4
multicol 1-2		multicol 3-4	

Source code:

```
\begin{tabular}{llll}
\toprule
col 1 & col 2 & col 3 & col4\\
\multicolumn{2}{l}{multicol 1-2}
& \multicolumn{2}{l}{multicol 3-4}\\
\bottomrule
\end{tabular}
```

2.2 Odd Example

Vertical lines look odd here:

col 1	col 2	col 3	col4
multicol 1-2		multicol 3-4	

Source code:

```
\begin{tabular}{|ll|ll|}
\hline
col 1 & col 2 & col 3 & col4\\
\multicolumn{2}{l}{multicol 1-2}
& \multicolumn{2}{l}{multicol 3-4}\\
\hline
\end{tabular}
```

2.3 Improved Example

But we may switch off the `tabularborder` package with `\tboff`.

col 1	col 2	col 3	col4
multicol 1-2		multicol 3-4	

Source code:

```
\begin{tabular}{|ll|ll|}
\hline
col 1 & col 2 & col 3 & col4\\
\multicolumn{2}{|l|}{multicol 1-2}
& \multicolumn{2}{l|}{multicol 3-4}\\
\hline
\end{tabular}
```

But we had still to manipulate the `\multicolumn` command. And now switch on again with `\tbon`.

2.4 Sophisticated Example

Article	Price/[USD]
Beer	3.40
Whisky	8.20

Source code:

```
\begin{tabular}{>{\sf}l|r@{}l}
\toprule
Article & \multicolumn{2}{c}{Price/[USD]} \\
\midrule
Beer      & \hskip4ex 3&.40\\
Whisky    &           8&.20 \\
\bottomrule
\end{tabular}
```

2.5 Example Using Helpers

Article	Remark
Beer	Note that this article can be sold only in special shops
Whisky	—

Source code:

```
\begin{tabular}
{b{0.5\tabcolwidthii\hangindent2ex}%
b{0.5\tabcolwidthii\hangindent2ex}}
\toprule
Article & Remark \\
\midrule
Beer      & Note that this article
        can be sold only
        in special shops\\
Whisky    &  ---          \\
\bottomrule
\end{tabular}
```

2.6 Example Using Helpers

Article	Code	Subcode	Remark
Beer	B	12	Note that this article can be sold only in special shops
Whisky	W	10	—

Source code:

```
\begin{tabular}
{b{0.25\tabcolwidthiv\hangindent2ex}%
 b{0.25\tabcolwidthiv\hangindent2ex}%
 b{0.25\tabcolwidthiv\hangindent2ex}%
 b{0.25\tabcolwidthiv\hangindent2ex}}%
\toprule
Article &\hskip-2ex Code & \hskip-4ex Subcode & Remark\\
\midrule
Beer    &B&12  & Note that this article
        can be sold only
        in special shops\\
Whisky    &W&10&  ---          \\
\bottomrule
\end{tabular}
```

2.7 Array Example

The `array` environment is left as before:

$$\overline{x^2+y^2 \quad = \quad z^2}$$

Source code:

```
\[
\begin{array}{lcr}
\toprule
x^2+y^2 &=& z^2\\
\end{array}
\]
```

2.8 Example Using Tabular Star Form

The following does not make sense:

first column	second column
first column	next second column

Source code:

```
\begin{tabular*}{0.5\textwidth}{@{}lr}
\toprule
first column & second column\\
first column & next second column\\
\bottomrule
\end{tabular*}
```

2.9 Nested Tabular Example

A nested tabular:

first column	second column
first column	next second column
[a]	A nested tabular, longer in this field now

Source code:

```
\begin{tabular}{ll}
\toprule
first column & second column\\
first column & next second column\\
\bottomrule
\multicolumn{2}{l}
{\begin{tabular}{ll}
\textsuperscript{a} & A nested tabular, longer in this field now \\
\end{tabular}}\\
\end{tabular}
```

3 Implementation

`\RequirePackage` This package has been tested based on the following package:

```
\RequirePackage{booktabs}
\RequirePackage{array}[2003/12/17 v2.4a Tabular extension package (FMi)]
```

However, the examples given here are working with version `v2.4c` too.

`\tb@ialign` We redefine `\ialign`, `\tabskip` into `-\col@sep`:

```
\newcommand{\tb@ialign}{\everycr{}\tabskip-\col@sep\halign}
```

Here we change definitions so that only the `tabular` is involved.

```
\newcommand{\tb@tbtabarray}{\@ifnextchar[{\tb@array}{\tb@array[c]}}
\def\@tabular{%
  \leavevmode
  \hbox \bgroup $\col@sep\tabcolsep \let\d@llarbegin\begingroup
                                    \let\d@llarend\endgroup
  \@tb@tbtabarray}
```

`\tbon \tboff` The following allows to switch back to the original settings by `\tboff` and reactivate by `\tbon`.

```
\newcommand{\@tb@tbtabarray}{}
\newcommand{\tbon}{\global\let\@tb@tbtabarray\tb@tbtabarray}
\tbon
\newcommand{\tboff}{\global\let\@tb@tbtabarray\@tabarray}
```

`\tb@array` We change the definition of the macro `\@@array` and rename it. Only three changes are needed, but we must place the full macro.

```
\newcommand{\tb@array}{}
\def\tb@array[#1]#2{%
  \@tempdima \ht \strutbox
  \advance \@tempdima by\extrarowheight
  \setbox \@arstrutbox \hbox{\vrule
              \@height \arraystretch \@tempdima
              \@depth \arraystretch \dp \strutbox
              \@width \z@}%
  \begingroup
  \@mkpream{#2}%
```

Here we use `\tb@ialign`, and in the next line we reset the `\tabskip`.

```
  \xdef\@preamble{\noexpand \tb@ialign \@halignto
                  \bgroup \@arstrut \tabskip \z@skip \@preamble
```

In the next line we are switching the `\tabskip` to `-\col@sep`.

```
                           \tabskip-\col@sep \cr}%
```

The rest of the macro is unchanged.

```
  \endgroup
  \@arrayleft
  \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi \fi
  \bgroup
  \let \@sharp ##\let \protect \relax
  \lineskip \z@
  \baselineskip \z@
  \m@th
  \let\\\@arraycr \let\tabularnewline\\\let\par\@empty \@preamble}
```

3.1 Additional Helpers

Here are some additional definitions that facilitate the handling of a tabular. An example is given in section 2.6.

```
\ifdefined\tabcolwidthi\else\newlength{\tabcolwidthi}\fi
\ifdefined\tabcolwidthii\else\newlength{\tabcolwidthii}\fi
\ifdefined\tabcolwidthiii\else\newlength{\tabcolwidthiii}\fi
\ifdefined\tabcolwidthiv\else\newlength{\tabcolwidthiv}\fi
\setlength{\tabcolwidthi}{\textwidth}
\addtolength{\tabcolwidthi}{-0\tabcolsep}
\setlength{\tabcolwidthii}{\textwidth}
\addtolength{\tabcolwidthii}{-2\tabcolsep}
\setlength{\tabcolwidthiii}{\textwidth}
\addtolength{\tabcolwidthiii}{-4\tabcolsep}
\setlength{\tabcolwidthiv}{\textwidth}
\addtolength{\tabcolwidthiv}{-6\tabcolsep}
```