

Automatic booktabs rules for tabular/tabular*

Moritz R. Schäfer

2026/06/26 v1.1.1

Contents

1	Introduction	1
1.1	Example	1
1.2	Notes	2
2	Interface and Implementation	2
2.1	Environments	3
2.2	Public commands	3
2.3	Options	4
3	Internal Helpers	4

1 Introduction

The `booktabstabular` package provides wrapper environments around `tabular` and `tabular*` that insert `\toprule` at the beginning, `\bottomrule` at the end, and locally remap the legacy `\hline` and `\cline` commands to their booktabs equivalents `\midrule` and `\cmidrule`. It also provides commands and options to temporarily override the standard `tabular` and `tabular*` environments so that existing documents can opt into this behaviour without rewriting each table.

1.1 Example

A typical opt-in workflow is:

```
\usepackage[override=global]{booktabstabular}

\begin{tabular}{lll}
  First  & Second & Third \\
  \hline % mapped to \midrule
  Alpha  & Beta   & Gamma
\end{tabular}
```

Alternatively, the wrapper environments can be used directly:

```

\begin{booktabstabular}{lll}
  First & Second & Third \\
\hline
  Alpha & Beta & Gamma
\end{booktabstabular}

```

Both should yield the following output:

First	Second	Third
Alpha	Beta	Gamma

 As you can

see, `\toprule` and `\bottomrule` are automatically inserted, and `\hline` is remapped to `\midrule`, so the table is typeset according to booktabs guidelines. The same applies to `tabular*` when using the `booktabstabular*` environment or the overridden `tabular*` name.

```

\begin{booktabstabular*}{\textwidth}{l@{\extracolsep{\fill}}l}
  First & Second \\
\hline
  Alpha & Beta
\end{booktabstabular*}

```

which produces the slightly wider table

First	Second
Alpha	Beta

.

1.2 Notes

This package deliberately does not insert any intermediate rule such as `\midrule` automatically; that remains the author's responsibility.

The end wrapper inserts `\bottomrule` at the legal alignment boundary, so the final row may be written either with or without a trailing `\\`.

The package does not attempt to remap the legacy `\hline` and `\cline` outside of the wrapper environments, so they will continue to work as normal in documents that only use the wrappers for a subset of tables.

The package affects only `tabular` and `tabular*`. It does not modify `array`, `longtable`, or any other alignment environment.

2 Interface and Implementation

Since the package is rather small, we will present the interface and implementation together in the following subsections.

```

1 <*package>
2 \RequirePackage{booktabs}
3 \ProvidesExplPackage
4   {booktabstabular}
5   {2026-06-26}
6   {1.1.1}
7   {tabular/tabular* wrappers with booktabs rules}

```

2.1 Environments

`origtabular` (*env.*) The `origtabular` and `origtabular*` environments are copies of the original `tabular` and `tabular*` environments. They are used internally but also made available for users who want to call the original versions if they have overridden the standard names.

```
8 \NewEnvironmentCopy { origtabular } { tabular }
9 \NewEnvironmentCopy { origtabular* } { tabular* }
```

`booktabstabular` (*env.*) The `booktabstabular` environment is a drop-in wrapper for `tabular`. It accepts the same arguments as `tabular`: an optional position specifier `[t]` or `[b]`, followed by the column specification.

It calls the copied `begin` macro directly rather than nesting a separate environment, which keeps LaTeX's environment stack balanced when the custom end code is used.

```
10 \NewDocumentEnvironment { booktabstabular } { o m }
11 {
12     \__booktabstabular_begin:n
13     {
14         \IfNoValueTF { #1 }
15         { \origtabular { #2 } }
16         { \origtabular [ #1 ] { #2 } }
17     }
18 }
19 { \__booktabstabular_end:n { \endorigtabular } }
```

`booktabstabular*` (*env.*) The `booktabstabular*` environment is a drop-in wrapper for `tabular*`. It accepts the width, an optional position specifier, and the column specification.

```
20 \NewDocumentEnvironment { booktabstabular* } { m o m }
21 {
22     \__booktabstabular_begin:n
23     {
24         \IfNoValueTF { #2 }
25         { \use:c { origtabular* } { #1 } { #3 } }
26         { \use:c { origtabular* } { #1 } [ #2 ] { #3 } }
27     }
28 }
29 { \__booktabstabular_end:n { \use:c { endorigtabular* } } }
```

2.2 Public commands

`\overridetabular` After calling `\overridetabular`, the public `tabular` and `tabular*` environments are rebound to the wrapper environments provided by this package. Note that many classes, including the standard classes, use a `tabular` in unexpected places, *i.e.*, to typeset the list of authors. If you don't want that to be affected, you should use the `override` package option.

```
30 \NewDocumentCommand \overridetabular {}
31 {
32     \RenewEnvironmentCopy { tabular } { booktabstabular }
33     \RenewEnvironmentCopy { tabular* } { booktabstabular* }
34 }
```

`\restoretabular` The command `\restoretabular` restores the original definitions of `tabular` and `tabular*`.

```

35 \NewDocumentCommand \restoretabular {}
36 {
37   \RenewEnvironmentCopy { tabular } { origtabular }
38   \RenewEnvironmentCopy { tabular* } { origtabular* }
39 }

```

2.3 Options

We offer an option to automatically override the standard `tabular` and `tabular*` environments.

override The `override` option selects how `tabular` and `tabular*` are rebound. The full syntax is `override=none|table|global`.

none Do not override the standard environments.

global This has almost the same effect as calling `\overridetabular` after loading the package. The difference is that the package option automatically disables the override around `\maketitle` to avoid affecting the title page, while the command does not.

table Override `tabular` and `tabular*` only inside `table` and `table*` environments. This is the safest and **recommended** option.

```

40 \str_new:N \l__booktabstabular_override_str
41 \DeclareKeys [ booktabstabular ]
42 {
43   override .choices:nn =
44     { none , table , global }
45     { \str_set_eq:NN \l__booktabstabular_override_str \l_keys_choice_str } ,
46   override .initial:n = none ,

```

overridetabular The `overridetabular` option is a legacy alias for `override=global`, and is provided for backwards compatibility.

```

47   overridetabular .meta:n      = { override = global } ,
48 }
49
50 \ProcessKeyOptions [ booktabstabular ]

```

3 Internal Helpers

`__booktabstabular_rulesetup:` Called at the beginning of the wrapper environments. Maps legacy horizontal rules to booktabs-style rules inside the wrappers and inserts `\toprule` at the beginning of the table. Note that we need to open a group here to keep the redefinitions local to the table. This group needs to be closed in the end code.

```

51 \cs_new_protected:Nn \__booktabstabular_begin:n
52 {
53   \group_begin:
54   \cs_set_eq:NN \hline \midrule
55   \cs_set_eq:NN \cline \cmidrule
56   #1

```

```

57 \toprule
58 }

```

(End of definition for _booktabstabular_rulesetup:.)

`_booktabstabular_end:n` Called at the end of the wrapper environments. It ends the current alignment row, inserts `\bottomrule` at the legal alignment boundary, then delegates the remaining close sequence to the saved original end code. This avoids copying kernel or package internals such as tagging sockets. Close the group that was opened in the begin code afterwards.

```

59 \cs_new_protected:Nn \_booktabstabular_end:n
60 {
61   \crr
62   \bottomrule
63   #1
64   \group_end:
65 }

```

(End of definition for _booktabstabular_end:n.)

`_booktabstabular_apply_override:` Applies the override according to the value of the override option.

```

66 \cs_new_protected:Nn \_booktabstabular_apply_override:
67 {
68   \str_case:Vn \l__booktabstabular_override_str
69   {
70     { none } { }
71     { global } { \_booktabstabular_activate_global: }
72     { table } { \_booktabstabular_activate_float: }
73   }
74 }

```

(End of definition for _booktabstabular_apply_override:.)

`_booktabstabular_activate_global:` The function called if the override option is set to global.

```

75 \cs_new_protected:Nn \_booktabstabular_activate_global:
76 {
77   \hook_gput_code:nnn { cmd/maketitle/before } { . } { \restoretabular }
78   \hook_gput_code:nnn { cmd/maketitle/after } { . } { \overridetabular }
79   \overridetabular
80 }

```

(End of definition for _booktabstabular_activate_global:.)

`_booktabstabular_activate_float:` The function called if the override option is set to table.

```

81 \cs_new_protected:Nn \_booktabstabular_activate_float:
82 {
83   \hook_gput_code:nnn { env/table/begin } { . } { \overridetabular }
84   \hook_gput_code:nnn { env/table*/begin } { . } { \overridetabular }
85 }

```

(End of definition for _booktabstabular_activate_float:.)

Finally, we call the function at the end of the package to apply the override if requested.

```

86 \_booktabstabular_apply_override:
87 </package>

```