The tabularht package

Heiko Oberdiek*
<heiko.oberdiek at googlemail.com>

2016/05/16 v2.6

Abstract

This package defines some environments that adds a height specification to tabular and array.

Contents

1 Usage **2**
1.1 Option vlines . . . 2
1.2 Limitations . . . 3
1.3 Compatibility . . . 3
1.4 Examples . . . 3
1.4.1 Example 1 . . . 3
1.4.2 Example 2 . . . 3

2 Implementation **4**
2.1 Environments . . . 4
2.2 Options . . . 6
2.3 Option vlines, driver independent stuff . . . 7
2.4 Driver pdftex . . . 7
2.5 DVI drivers . . . 11

3 Installation **13**
3.1 Download . . . 13
3.2 Bundle installation . . . 13
3.3 Package installation . . . 14
3.4 Refresh file name databases . . . 14
3.5 Some details for the interested . . . 14

4 Catalogue **15**

5 History **15**
[2005/09/22 v1.0] . . . 15
[2005/10/16 v2.0] . . . 15
[2005/10/18 v2.1] . . . 15
[2006/02/20 v2.2] . . . 16
[2006/12/22 v2.3] . . . 16
[2007/03/21 v2.4] . . . 16
[2007/04/11 v2.5] . . . 16
[2016/05/16 v2.6] . . . 16

6 Index **16**

*Please report any issues at https://github.com/ho-tex/oberdiek/issues

1 Usage

```
\usepackage{tabularht}
```

The package provides the following environments that extend the tabular/array environment by a height specification as first argument:

- `tabularht`, `tabularht*`
- `arrayht`
- `tabularhtx` (if package tabularx is loaded)

The height argument allows a length specification, package calc is supported if used. This means, the tabular will have the specified height. You can also use the prefixes `to=` and `spread=`. `to=` is the default, `spread=` means, the natural height of the tabular box is changed by the length after `spread=`.

Examples:

```
\begin{tabularht}{1in}          → height is 1in
\begin{tabularht}{to=1in}       → height is 1in
\begin{tabularht}{spread=0pt} → natural height, same as \begin{tabular}
\begin{tabularht}{spread=1in} → natural height increased by 1in
```

Hint: See also package tabularkv, it provides an interface, where most parameters for the environments can be given by key-value pairs.

`\interrowspace{...}`

Adds space between table rows. It is essentially the same as `\noalign{\vspace{...}}`.

`\interrowfill`

Short for `\interrowspace{\fill}`

`\interrowstart ... \interrowstop`

Marker commands, useful for option vlines.

1.1 Option vlines

Warning: This stuff is experimental.

Vertical lines are interrupted, if space is inserted in `\noalign`, `\interrowspace`, `\addlinespace` (booktabs), between double `\hline`s. This option tries to detect and add the vertical lines. The lines in a tabular with tabularht support (environments of this package) are numbered from left to right. The gap that is controlled by `\interrowspace` or inbetween `\interrowstart` and `\interrowstop` is then filled with the detected vertical lines.

If only a limited selection of the lines should be drawn, the commands know an optional argument with a list of line numbers, e.g.

```
\begin{tabularht}{50mm}{|1|1|}
  Hello & World\\
  \interrowfill[1,3]
  Foo & Bar
\end{tabularht}
```

There are three lines, but the middle line is not drawn in the gap between the first and second row. Zero can be used to suppress all lines:

```
\interrowspace[0]{10mm}
```

The syntax of the commands with the optional argument with the line number list ⟨*list*⟩. ⟨*list*⟩ is a comma separated list of numbers, ⟨*height*⟩ means the height specification described above with the optional prefixes `to=` or `spread=`.

```
\interrowspace [⟨list⟩] {⟨height⟩}
\interrowfill [⟨list⟩]
\interrowstart [⟨list⟩] ... \interrowstop
```

Option vlines is driver dependent and uses ε-TEX features.

pdftex: pdfTEX in PDF mode. Here the positions of the lines are written with the help of the `\pdfsavepos` feature into the `.aux` file(s). Therefore you need two LaTeX runs to get the lines.

dvips: Here, PostScript's currentpoint it used to get the line positions. The lines are then drawn at the end of the page. Thus one LATEX/dvips run is sufficient for this option.

Other drivers:

PostScript drivers: probably possible, an end of page hook would be nice.

VTEX: with GeX (PostScript interpreter) probably possible.

dvipdfm: no idea. The big problem is, how to get the current position?

1.2 Limitations

- Vertical lines are interrupted by `\noalign{\vfill}`.

1.3 Compatibility

- array, delarray, tabularx are supported.
- There can be problems with packages that redefine `\@array` (or `\@@array`, `\@tabarray`) and `\@arrayrule` (for option vlines).
- colortbl: it should at least work, but there isn't support for filling the gaps with color, neither the rules nor the backgrounds.

1.4 Examples

1.4.1 Example 1

```
⟨*example1⟩
\documentclass{article}
\usepackage{tabularht}

\begin{document}
\fbox{%
  \begin{tabularht*}{1in}{4in}{@{}l@{\extracolsep{\fill}}r@{}}%
    upper left corner & upper right corner\\%
    \noalign{\vfill}%
    \multicolumn{2}{@{}c@{}}{bounding box}\\%
    \noalign{\vfill}%
    lower left corner & lower right corner\\%
  \end{tabularht*}%
}
\end{document}
⟨/example1⟩
```

1.4.2 Example 2

```
⟨*example2⟩
\documentclass{article}
\usepackage{booktabs}
\usepackage[dvips,vlines]{tabularht}

\begin{document}

\begin{tabularht}{spread=0pt}{|l|l|}
  \hline
  First&Line\\%
  \hline
\interrowstart
  \addlinespace[10mm]%
\interrowstop
  \hline
  Second&Line\\%
\interrowstart
  \hline
  \hline
\interrowstop
  Third&Line\\%
  \hline
\interrowspace{10mm}
  \hline
  Fourth&Line\\%
  \hline
\end{tabularht}

\end{document}
⟨/example2⟩
```

2 Implementation

```
⟨*package⟩
```

Package identification.

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{tabularht}%
  [2016/05/16 v2.6 Tabular with height specified (HO)]
```

2.1 Environments

```
\let\@toarrayheight\@empty
\let\tabH@array@init\@empty

\toks@={%
  \begingroup
    \long\def\x#1\vcenter\fi\fi\bgroup#2\@sharp#3#4\@nil{%
      \endgroup
      \gdef\@array[##1]##2{%
        \tabH@array@init
        #1%
        \vcenter\fi\fi
        \@toarrayheight
        \bgroup
        \let\@toarrayheight\@empty
        #2\@sharp###3#4%
      }%
    }%
  \expandafter\x\@array[#1]{#2}\@nil % hash-ok
}
\edef\tabH@patch@array{\the\toks@}
\def\tabH@patch@@array{%
  \ifx\@array\@@array
```

```
    \def\reserved@a{\let\@@array\@array}%
  \else
    \let\reserved@a\relax
  \fi
  \tabH@patch@array
  \reserved@a
}
\tabH@patch@@array

\@ifpackageloaded{array}{}{%
  \AtBeginDocument{%
    \@ifpackageloaded{array}{%
      \tabH@patch@@array
    }{}%
  }%
}

\def\tabH@setheight#1{%
  \tabH@@setheight#1==\@nil
}
\def\tabH@@setheight#1=#2=#3\@nil{%
  \ifx\\#2#3\\%
    \setlength{\dimen@}{#1}%
    \edef\@toarrayheight{to\the\dimen@}%
  \else
    \edef\tabH@temp{\zap@space#1 \@empty}%
    \ifx\tabH@temp\tabH@to
    \else
      \ifx\tabH@temp\tabH@spread
      \else
        \PackageError{tabularht}{%
          Unknown height specifier %
          `\expandafter\strip@prefix\meaning\tabH@temp'%
        }{%
          The height dimension for tabular height can be prefixed%
          \MessageBreak
          with `to=' or `spread=', default is `to='.%
        }%
        \let\tabH@temp\tabH@to
      \fi
    \fi
    \setlength{\dimen@}{#2}%
    \edef\@toarrayheight{\tabH@temp\the\dimen@}%
  \fi
}
\def\tabH@to{to}
\def\tabH@spread{spread}
```

First argument is the height of the table, then the orignal arguments for tabular follow.

```
\newenvironment{tabularht}[1]{%
  \tabH@setheight{#1}%
  \tabular
}{%
  \endtabular
}

\newenvironment{tabularht*}[1]{%
  \tabH@setheight{#1}%
  \@nameuse{tabular*}%
}{%
  \@nameuse{endtabular*}%
}
```

```
\newenvironment{tabularhtx}[1]{%
  \tabH@setheight{#1}%
  \tabularx
}{%
  \endtabularx
}

\newenvironment{arrayht}[1]{%
  \tabH@setheight{#1}%
  \array
}{%
  \endarray
}

\def\interrowspace{%
  \noalign\bgroup
    \tabH@interrowspace
}
\newcommand*{\tabH@interrowspace}[2][]{%
    \tabH@vspace{#1}{#2}%
  \egroup
}
\def\interrowfill{%
  \noalign\bgroup
    \tabH@interrowfill
}
\newcommand*{\tabH@interrowfill}[1][]{%
    \tabH@vspace{#1}{\fill}%
  \egroup
}
\def\tabH@vspace#1#2{%
  \tabH@vspace@start{#1}%
  \vspace{#2}%
  \tabH@vspace@stop
}
\let\tabH@vspace@start\@gobble
\let\tabH@vspace@stop\@empty

\newcommand*{\interrowstart}{%
  \noalign\bgroup
    \tabH@interrowstart
}
\newcommand*{\tabH@interrowstart}[1][]{%
    \tabH@vspace@start{#1}%
  \egroup
}
\newcommand*{\interrowstop}{%
  \noalign{\tabH@vspace@stop}%
}
```

2.2 Options

```
\providecommand*{\tabH@driver}{}

\DeclareOption{vlines}{%
  \let\tabH@temp\relax
}
\DeclareOption{pdftex}{}
\DeclareOption{dvips}{%
  \def\tabH@driver{dvips}%
}
\ProcessOptions*\relax
```

```
\ifx\tabH@temp\relax
\else
  \expandafter\endinput
\fi

\begingroup
  \@ifundefined{eTeXversion}{%
    \PackageError{tabularht}{%
      Option `vlines' requires eTeX%
    }{%
      Use of eTeX is recommended for LaTeX, see ltnews16.%
    }%
    \endgroup
    \endinput
  }{}%
\endgroup
```

2.3 Option vlines, driver independent stuff

```
\begingroup
  \let\@addtoreset\@gobbletwo
  \newcounter{tabH@unique}%
\endgroup
\let\tabH@currenttab\@empty

\def\tabH@array@init{%
  \ifx\@toarrayheight\@empty
    % ignore vertical lines of nested tabular environments
    \let\tabH@currenttab\@empty
  \else
    \stepcounter{tabH@unique}%
    \edef\tabH@currenttab{\the\c@tabH@unique}%
  \fi
}

\renewcommand*{\@arrayrule}{%
  \@addtopreamble{%
    \hskip -.5\arrayrulewidth
    \ifx\tabH@currenttab\@empty
    \else
      \tabH@vrule{\tabH@currenttab}%
    \fi
    \begingroup
      \expandafter\ifx\csname CT@arc@\endcsname\relax
      \else
        \expandafter\CT@arc@
      \fi
      \vline
    \endgroup
    \hskip -.5\arrayrulewidth
  }%
}
\let\tabH@arrayrule\@arrayrule
\AtBeginDocument{%
  \@ifpackageloaded{colortbl}{%
    \let\@arrayrule\tabH@arrayrule
  }{}%
}

\let\tabH@vrule\@gobble
```

2.4 Driver pdftex

```
\RequirePackage{ifpdf}
\ifpdf
  \begingroup
    \@ifundefined{pdfsavepos}{%
      \PackageError{tabularht}{%
        Your pdfTeX is too old%
      }{%
        \string\pdfsavepos\space is missing.%
      }%
      \endgroup
      \csname fi\endcsname
      \endinput
    }{}%

    \let\on@line\@empty
    \PackageInfo{tabularht}{%
      Using driver `pdftex' because of pdfTeX in PDF mode%
    }%
  \endgroup

  \protected\def\tabH@vrule#1{%
    \if@filesw
      \pdfsavepos
      \protected@write\@auxout{%
        \let\tabH@lastxpos\relax
      }{%
        \tabH@aux@vrule{#1}{\tabH@lastxpos}%
      }%
    \fi
  }%

  \def\tabH@lastxpos{\the\pdflastxpos}%
  \def\tabH@lastypos{\the\pdflastypos}%

  % The .aux file contains three commands:
  % \tabH@aux@vrule{tabular id}{x position}
  % \tabH@aux@vstart{tabular id}{row id}{x position}{y position}
  % \tabH@aux@vstop{y position}
  %
  \AtBeginDocument{%
    % The .aux files are read the first time before
    % \AtBeginDocument and later at \end{document}.
    % \tabH@aux@done is a marker to distinguish
    % between these two readings. Only in the first
    % case we need the \tabH@aux@... commands.
    \let\tabH@aux@done\@empty
    \if@filesw
      \immediate\write\@mainaux{%
        \@percentchar\@percentchar BeginProlog: tabularht%
      }%
      % items in the aux file are executed,
      % if tabularht is loaded
      % and during the aux file read at \begin{document} only
      \immediate\write\@mainaux{%
        \detokenize{%
          % the \tabH@aux@... commands are needed only if
          % tabularht is loaded with driver pdftex.
          \@ifundefined{tabH@aux@vrule}\@secondoftwo\@firstofone
          {%
            % disable commands except for the first .aux files reading
            \@ifundefined{tabH@aux@done}\@gobble\@firstofone
          }%
```

```
      {%
        \let\tabH@aux@vrule\@gobbletwo
        \let\tabH@aux@vstart\@gobblefour
        \let\tabH@aux@vstop\@gobble
      }%
    }%
  }%
  \immediate\write\@mainaux{%
    \@percentchar\@percentchar EndProlog: tabularht%
  }%
 \fi
}%

% the x positions of vrules are stored in
% \tabH@<tabcount>list with distinct values
\protected\def\tabH@aux@vrule#1#2{%
 \@ifundefined{tabH@#1list}{%
  \expandafter\xdef\csname tabH@#1list\endcsname{%
   \noexpand\do{#2}%
  }%
 }{%
  \begingroup
   \def\x{#2}%
   \let\y\@undefined
   \let\do\tabH@do@add
   \expandafter\xdef\csname tabH@#1list\endcsname{%
    \csname tabH@#1list\endcsname\@empty
    \ifx\y\@undefined
     \noexpand\do{\x}%
    \fi
   }%
  \endgroup
 }%
}%
\def\tabH@do@add#1{%
 \ifx\y\@undefined
  \ifnum#1<\x\space
  \else
   \expandafter\ifx\csname y\endcsname\relax\fi
   \ifnum#1>\x\space
    \noexpand\do{\x}%
   \fi
  \fi
 \fi
 \noexpand\do{#1}%
}%

\def\tabH@vspace@start#1{%
 \if@filesw
  \stepcounter{tabH@unique}%
  \edef\tabH@currentrow{\the\c@tabH@unique}%
  \pdfsavepos
  \protected@write\@auxout{%
   \let\tabH@lastxpos\relax
   \let\tabH@lastypos\relax
  }{%
   \tabH@aux@vstart{\tabH@currenttab}{\tabH@currentrow}%
                   {\tabH@lastxpos}{\tabH@lastypos}%
  }%
 \fi
 \begingroup
  \edef\a{tabH@\tabH@currenttab row\tabH@currentrow}%
```

```
    \expandafter\let\expandafter\x\csname\a x\endcsname
    \ifx\x\relax
    \else
      \expandafter\let\expandafter\y\csname\a y\endcsname
      \expandafter\let\expandafter\l
          \csname tabH@\tabH@currenttab list\endcsname
      \ifx\l\relax
      \else
        \def\f{#1}%
        \ifx\f\@empty
          \let\do\tabH@do@set
        \else
          \count@=\z@
          \let\do\tabH@do@filter
        \fi
        \setbox\z@=\hbox{\l}%
        \wd\z@=\z@
        \dp\z@=\z@
        \copy\z@
      \fi
    \fi
  \endgroup
}%
\def\tabH@vspace@stop{%
  \if@filesw
    \pdfsavepos
    \protected@write\@auxout{%
      \let\tabH@lastypos\relax
    }{%
      \tabH@aux@vstop{\tabH@lastypos}%
    }%
  \fi
}%
\def\tabH@do@set#1{%
  \hbox to \z@{%
    \hskip \dimexpr #1sp - \x sp\relax
    \vrule \@width\arrayrulewidth
          \@depth\dimexpr \y sp\relax
    \hss
  }%
}%
\def\tabH@do@filter{%
  \@tempswafalse
  \advance\count@\@ne
  \@for\e:=\f\do{%
    \ifnum\e=\count@
      \@tempswatrue
    \fi
  }%
  \if@tempswa
    \expandafter\tabH@do@set
  \else
    \expandafter\@gobble
  \fi
}%

\protected\def\tabH@aux@vstart#1#2#3#4{%
  \def\tabH@current@vstart{{#1}{#2}{#3}{#4}}%
}%
\protected\def\tabH@aux@vstop{%
  \expandafter\tabH@aux@v\tabH@current@vstart
}%
```

```
  \def\tabH@aux@v#1#2#3#4#5{%
    \expandafter\gdef\csname tabH@#1row#2x\endcsname{#3}%
    \expandafter\xdef\csname tabH@#1row#2y\endcsname{%
      \the\numexpr #4 - #5\relax
    }%
  }%

  \csname fi\endcsname
  \endinput

\fi
```

2.5 DVI drivers

```
\ifx\tabH@driver\@empty
  \PackageError{tabularht}{%
    Missing DVI driver, option `vlines' disabled%
  }{%
    Supported DVI drivers: dvips.%
  }%
  \expandafter\endinput
\fi

\def\tabH@driver@dvips{%
  \def\tabH@literalps##1{\special{ps:SDict begin ##1 end}}%
  \def\tabH@headerps##1{\special{! ##1}}%
}

\@onelevel@sanitize\tabH@driver
\@ifundefined{tabH@driver@\tabH@driver}{%
  \PackageError{tabularht}{%
    Unsupported driver `\tabH@driver'%
  }{%
    Supported DVI drivers: dvips.%
  }%
  \endinput
}{}

\begingroup
  \let\on@line\@empty
  \PackageInfo{tabularht}{%
    Using driver `\tabH@driver'%
  }%
\endgroup
\csname tabH@driver@\tabH@driver\endcsname

\protected\def\tabH@vrule#1#2\vrule#3\arrayrulewidth{%
  #2% \fi or empty
  % hack to get rid of maxdrift rounding of dvips,
  % thus simulate a large motion
  \kern1in\relax
  \tabH@literalps{%
    #1 tabH.vrule %
    Resolution neg 0 translate%
  }%
  \vrule#3\arrayrulewidth
  \tabH@literalps{Resolution 0 translate}%
  \kern-1in\relax
}

\def\tabH@vspace@start#1{%
  \begingroup
    \let\y\@empty
```

```
    \@for\x:=#1\do{%
      \ifx\y\@empty
        \edef\y{\x}%
      \else
        \edef\y{\y\space\x}%
      \fi
    }%
    \tabH@literalps{\tabH@currenttab[\y]currentpoint exch pop}%
  \endgroup
}
\def\tabH@vspace@stop{%
  \tabH@literalps{%
    currentpoint exch pop %
    \number\dimexpr\arrayrulewidth\relax\space
    tabH.vspace%
  }%
}

\tabH@headerps{%
  userdict begin%
    /tabH.list 10 dict def%
    /tabH.job [] def %
  end%
  /tabH.vrule{%
    10 string cvs cvn dup tabH.list exch known{%
      tabH.list exch dup [ exch tabH.list exch get %
      currentpoint pop round exch true exch{%
        % tabH.list key [ ... x true i
        % tabH.list key [ ... false i
        exch{%
          % ... [ ... x i
          2 copy lt{false}{%
            2 copy eq{pop false}{exch true}ifelse%
          }ifelse%
        }{false}ifelse%
      }forall %
      pop%
      ]put%
    }{%
      tabH.list exch[currentpoint pop round]put%
    }ifelse%
  }bind def%
  % <tab num> <cols array> <ytop> <ybottom> <rulewidth[sp]>
  /tabH.vspace{%
    userdict begin %
      10 dict dup begin %
        exch 65536 div Resolution mul 72.27 div %
        % dvips uses a poor man's ceil function
        % see dopage.c before "drawrule": (int)(... + 0.9999999)
        0.9999999 add truncate%
        /rulewidth exch def %
        exch/ybottom exch def %
        exch/ytop exch def %
        exch/cols exch def %
        exch/tabkey exch 10 string cvs cvn def %
      end%
      /tabH.job exch[exch userdict/tabH.job get aload pop]def %
    end%
  }bind def %
  % Now we do the work at the end of the page.
  % Unhappily "eop-hook" cannot be used, because "eop"
  % executes "restore" before, so that all data are lost.
```

```
559  TeXDict begin%
560    /eop%
561    [%
562      {%
563        tabH.job{%
564          begin%
565          /colarray %
566            tabH.list tabkey known{tabH.list tabkey get}{[]}ifelse %
567          def %
568          cols length 0 eq not{%
569            /colarray[%
570              cols{1 sub %
571                dup 0 lt{pop}{%
572                  dup colarray length ge{pop}{%
573                    colarray exch get%
574                  }ifelse%
575                }ifelse%
576              }forall%
577            ]def%
578          }if %
579          colarray{%
580            % (rulewidth) == rulewidth == % debug
581            Resolution sub %
582            ytop rulewidth ytop ybottom sub v%
583          }forall %
584        end%
585      }forall%
586      % tabH.list{== ==}forall % debug
587    }bind aload pop %
588    TeXDict /eop get aload pop%
589  ]cvx def %
590  end%
591 }
```

592 ⟨/package⟩

3 Installation

3.1 Download

Package. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/oberdiek/tabularht.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/tabularht.pdf Documentation.

Bundle. All the packages of the bundle 'oberdiek' are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/oberdiek.tds.zip

TDS refers to the standard "A Directory Structure for TEX Files" (CTAN:tds/tds.pdf). Directories with texmf in their name are usually organized this way.

3.2 Bundle installation

Unpacking. Unpack the oberdiek.tds.zip in the TDS tree (also known as texmf tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

[1] http://ctan.org/pkg/tabularht

Script installation. Check the directory TDS:scripts/oberdiek/ for scripts that need further installation steps. Package attachfile2 comes with the Perl script pdfatfi.pl that should be installed in such a way that it can be called as pdfatfi. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

3.3 Package installation

Unpacking. The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain TEX:

```
tex tabularht.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as texmf tree):

```
tabularht.sty            → tex/latex/oberdiek/tabularht.sty
tabularht.pdf            → doc/latex/oberdiek/tabularht.pdf
tabularht-example1.tex → doc/latex/oberdiek/tabularht-example1.tex
tabularht-example2.tex → doc/latex/oberdiek/tabularht-example2.tex
tabularht.dtx            → source/latex/oberdiek/tabularht.dtx
```

If you have a docstrip.cfg that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

3.4 Refresh file name databases

If your TEX distribution (teTEX, mikTEX, ...) relies on file name databases, you must refresh these. For example, teTEX users run texhash or mktexlsr.

3.5 Some details for the interested

Unpacking with LATEX. The .dtx chooses its action depending on the format:

plain TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{tabularht.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the .dtx or the .drv to generate the documentation. The process can be configured by the configuration file ltxdoc.cfg. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLATEX:

```
pdflatex tabularht.dtx
makeindex -s gind.ist tabularht.idx
pdflatex tabularht.dtx
makeindex -s gind.ist tabularht.idx
pdflatex tabularht.dtx
```

4 Catalogue

The following XML file can be used as source for the TEX Catalogue. The elements caption and description are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is tabularht.xml.

```
⟨*catalogue⟩
<?xml version='1.0' encoding='us-ascii'?>
<!DOCTYPE entry SYSTEM 'catalogue.dtd'>
<entry datestamp='$Date$' modifier='$Author$' id='tabularht'>
  <name>tabularht</name>
  <caption>Tabular environments with height specified.</caption>
  <authorref id='auth:oberdiek'/>
  <copyright owner='Heiko Oberdiek' year='2005-2007'/>
  <license type='lppl1.3'/>
  <version number='2.6'/>
  <description>
    The tabularht package defines some environments that add a height
    specification to tabular and array environments.  The default set
    of new environments take a value for their height in the first
    argument: defined environments are: <tt>tabularht</tt>,
    <tt>tabularht*</tt> and <tt>arrayht</tt>.  If package
    <xref refid='tabularx'>tabularx</xref> is also loaded,
    the package also defines environments <tt>tabularxht</tt> and
    <tt>tabularxht*</tt>.
    <p/>
    The places where stretching is to happen are signalled by<br/>
    <tt>\noalign{\vfill}</tt><br/>
    immediately after the <tt>\\</tt> that ends a row of the table or
    array.
    <p/>
    The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
  </description>
  <documentation details='Package documentation'
      href='ctan:/macros/latex/contrib/oberdiek/tabularht.pdf'/>
  <ctan file='true' path='/macros/latex/contrib/oberdiek/tabularht.dtx'/>
  <miktex location='oberdiek'/>
  <texlive location='oberdiek'/>
  <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
</entry>
⟨/catalogue⟩
```

5 History

[2005/09/22 v1.0]

- First public version.

[2005/10/16 v2.0]

- Height specification allows to=... or spread=..., default is to=.
- Option vlines added, drivers pdftex and dvips.
- \interrowspace, \interrowfil, and \interrowstart...\interrowstop added.

[2005/10/18 v2.1]

- Fix for package colortbl, but the colors of colortbl remain unsupported.

[2006/02/20 v2.2]

- Code is not changed.
- DTX framework.

[2006/12/22 v2.3]

- Documentation fix.
- Fix in code of option vlines.

[2007/03/21 v2.4]

- Fix: Counter `tabh@unique` must not be changed by `\include`.

[2007/04/11 v2.5]

- Line ends sanitized.

[2016/05/16 v2.6]

- Documentation updates.

6 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; plain numbers refer to the code lines where the entry is used.

Symbols

\@@array 72, 73
\@addtopreamble 227
\@addtoreset 211
\@array 58, 68, 72, 73
\@arrayrule 226, 243, 246
\@auxout 274, 365, 401
\@depth 412
\@empty 51, 52, 64, 98, 170, 214, 217, 219, 229, 265, 296, 339, 384, 448, 473, 496, 498
\@firstofone 308, 311
\@for 419, 497
\@gobble 169, 250, 311, 316, 427
\@gobblefour 315
\@gobbletwo 211, 314
\@ifpackageloaded 82, 84, 245
\@ifundefined 200, 254, 308, 311, 329, 463
\@mainaux 298, 304, 320
\@nameuse 129, 131
\@ne 418
\@nil 56, 68, 91, 93
\@onelevel@sanitize 462
\@percentchar 299, 321
\@secondoftwo 308
\@sharp 56, 65
\@tempswafalse 417
\@tempswatrue 421
\@toarrayheight . 51, 62, 64, 96, 115, 217
\@undefined 336, 340, 348
\@width 411
\\ 8, 10, 12, 26, 32, 37, 41, 94, 615

A

\a 374, 375, 378
\addlinespace 29
\advance 418
\array 143
\arrayrulewidth 228, 240, 411, 480, 489, 510
\AtBeginDocument . . 83, 244, 290, 292

B

\begin 5, 7, 22, 24, 303

C

\c@tabH@unique 222, 363
\copy 393
\count@ 387, 418, 420
\csname 234, 261, 330, 338, 339, 351, 375, 378, 380, 438, 439, 444, 478
\CT@arc@ 236

D

\DeclareOption 185, 188, 189
\detokenize 305
\dimen@ 95, 96, 114, 115
\dimexpr 410, 412, 510
\do 331, 337, 341, 353, 357, 385, 388, 419, 497
\documentclass 2, 18
\dp 392

E

\e 419, 420
\end 13, 15, 43, 45, 292
\endarray 145
\endcsname 234, 261, 330, 338, 339, 351, 375, 378, 380, 438, 439, 444, 478
\endinput . 196, 207, 262, 445, 454, 469
\endtabular 124
\endtabularx 138
\extracolsep 7

F

\f 383, 384, 419
\fbox 6
\fill 7, 161

G

\gdef 58, 438

H

\hbox 390, 409
\hline 25, 27, 31, 34, 35, 38, 40, 42
\hskip 228, 240, 410
\hss 413

I

\if@filesw 272, 297, 361, 399
\if@tempswa 424
\ifnum 349, 352, 420
\ifpdf 252
\ifx 72, 94, 99, 101, 194, 217, 229, 234, 340, 348, 351, 376, 381, 384, 448, 498
\immediate 298, 304, 320
\interrowfill *2*, 156
\interrowspace *2*, 39, 148
\interrowstart *2*, 28, 33, 172
\interrowstop 30, 36, 180

K

\kern 484, 491

L

\l 379, 381, 390

M

\meaning 105
\MessageBreak 108
\multicolumn 10

N

\NeedsTeXFormat 48
\newcommand . . 152, 160, 172, 176, 180
\newcounter 212
\newenvironment . . . 120, 127, 134, 141
\noalign . . 9, 11, 149, 157, 173, 181, 614
\number 510
\numexpr 440

O

\on@line 265, 473

P

\PackageError . . 103, 201, 255, 449, 464
\PackageInfo 266, 474
\pdflastxpos 282
\pdflastypos 283
\pdfsavepos 258, 273, 364, 400
\ProcessOptions 192
\protected 271, 328, 431, 434, 480
\protected@write 274, 365, 401
\providecommand 183
\ProvidesPackage 49

R

\renewcommand 226
\RequirePackage 251
\reserved@a 73, 75, 78

S

\setbox 390
\setlength 95, 114
\space 258, 349, 352, 501, 510
\special 458, 459
\stepcounter 221, 362
\strip@prefix 105

T

\tabH@ 327
\tabH@@setheight 91, 93
\tabH@array@init 52, 59, 216
\tabH@arrayrule 243, 246
\tabH@aux@ 295, 306
\tabH@aux@done 293, 296
\tabH@aux@v 435, 437
\tabH@aux@vrule . . 277, 286, 314, 328
\tabH@aux@vstart . . 287, 315, 369, 431
\tabH@aux@vstop . . 288, 316, 404, 434
\tabH@current@vstart 432, 435
\tabH@currentrow 363, 369, 374
\tabH@currenttab 214, 219, 222, 229, 231, 369, 374, 380, 504
\tabH@do@add 337, 347
\tabH@do@filter 388, 416
\tabH@do@set 385, 408, 425
\tabH@driver 183, 190, 448, 462, 463, 465, 475, 478
\tabH@driver@dvips 457
\tabH@headerps 459, 515
\tabH@interrowfill 158, 160
\tabH@interrowspace 150, 152
\tabH@interrowstart 174, 176
\tabH@lastxpos . 275, 277, 282, 366, 370
\tabH@lastypos . 283, 367, 370, 402, 404
\tabH@literalps . 458, 485, 490, 504, 508
\tabH@patch@@array 71, 80, 85
\tabH@patch@array 70, 77
\tabH@setheight . 90, 121, 128, 135, 142
\tabH@spread 101, 119
\tabH@temp 98, 99, 101, 105, 111, 115, 186, 194
\tabH@to 99, 111, 118
\tabH@vrule 231, 250, 271, 480
\tabH@vspace 153, 161, 164

\tabH@vspace@start 165, 169, 177, 360, 494
\tabH@vspace@stop 167, 170, 181, 398, 507
\tabular 122
\tabularx 136
\the . 70, 96, 115, 222, 282, 283, 363, 440
\toks@ 54, 70

U

\usepackage 3, 19, 20

V

\vcenter 56, 61
\vfill 9, 11, 614
\vline 238
\vrule 411, 480, 489
\vspace 166

W

\wd 391
\write 298, 304, 320

X

\x 56, 68, 335, 341, 349, 352, 353, 375, 376, 410, 497, 499, 501

Y

\y 336, 340, 348, 378, 412, 496, 498, 499, 501, 504

Z

\z@ 387, 390, 391, 392, 393, 409
\zap@space 98