

Semi-Manual Grid Setting Using `gridset` *

Markus Kohm

2020-02-12 v0.3

Abstract

Grid setting—also known as strict in-register setting—is something, that should be done for a lot of documents but is not easy using L^AT_EX. Package `gridset` helps to get the information needed for grid setting. It does not implement auto grid setting, but there is a command `\vskipnextgrid`, that moves to the next grid position. This may be enough under some circumstances. In other circumstances it may fail. So `gridset` is only one more step for grid setting not a complete solution.

Important Note: This package should have been never released, because it was only a very quick implementation of an idea. You should not use it for any productive purpose. It has been made for testing only. I would prefer to retire it from any distribution. Nevertheless I know few persons using the package. So it will be still there but without any support!

Contents

1 User Manual	1
2 Implementation	3
3 Example	8

1 User Manual

`\gridinterval` This macro contains a number without unit! The number is the distance between two grid lines in unit ‘scale points’ (sp). You may set it so another value using, e.g.

```
\newlength{\name{your length}}
\setlength{\name{your length}}{new length value}
\newcounter{\name{your counter}}
```

*This is an alpha version! Don’t use it! Only test it! There’s no support and everything may change!

	<code>\setcounter{\<name of your counter>}{\<name of your length>} \edef\gridinterval{\<name of your length>}</code>
<code>\gridbase</code>	This macro contains an integer number, that represents the y-coordinate of the upper start of the grid. If you want to change it, just save a position and <code>\edef</code> the <code>\gridbase</code> to the y-pos of that position.
	Most time you don't need to change <code>\gridinterval</code> and <code>\gridposition</code> , because they are initialized to a base line grid at start of first page. Because of this, it doesn't matter, that changing them is not really user friendly.
<code>\SavePos</code>	<code>\SavePos{\<unique name>}</code> saves informations about the current position to the aux-file. These informations are read at next L ^A T _E X run and may be used (see <code>\the...</code> commands below) then. The <code>\<unique name></code> has to be a position name, that is unique for all saved position informations of the current document. Note that the command has been renamed from <code>\savepos</code> to <code>\SavePos</code> in release 0.2, because L ^A T _E X since 0.85 uses <code>\savepos</code> as a new primitive. On other engines the old name <code>\savepos</code> is still available.
<code>\vskipnextgrid</code>	This command moves to the next grid position. To achieve this, a position information is saved at this and used at next L ^A T _E X run. The used name of the position information is <code>vb!\<number of skip></code> . <code>\<number of skip></code> is the number of the current <code>\vskipnextgrid</code> usage. Counter <code>gridcnt</code> is used to number the usage of <code>\vskipnextgrid</code> .
<code>\thegridinfo</code>	<code>\thegridinfo{\<name>}</code> outputs <ul style="list-style-type: none"> • arabic page number of the named position, • grid base, that was valid saving the information of the named position, • grid interval, that was valid saving the information of the named position, • x-coordinate of the named position, • y-coordinate of the named position.
	The coordinates and intervals are numbers without units. The unit is 'scale points' (sp).
<code>\theposinfo</code>	<code>\theposinfo{\<name>}</code> outputs <ul style="list-style-type: none"> • y-coordinate of the named position, • grid line number (first is 0) of the next grid position, • offset of the next grid position from grid base, • distance to the next grid position.
	The coordinates, offsets and distances are numbers without units. The unit is 'scale points' (sp).
<code>\theypos</code>	<code>\theypos{\<name>}</code> outputs the y-coordinate of the named position.

2 Implementation

`\gridset@luaorpdf` We need some locals because LuaTeX changed the names of several primitives inherited from PDFTeX.

```
1 \newcommand*{\gridset@luaorpdf}[1]{%
2   \expandafter\newcommand\csname gridset@#1\endcsname{}%
3   \ifcsname pdf#1\endcsname
4     \expandafter\let\csname gridset@#1\expandafter\endcsname
5       \csname pdf#1\endcsname
6   \else
7     \ifcsname #1\endcsname
8       \expandafter\let\csname gridset@#1\expandafter\endcsname
9         \csname #1\endcsname
10    \else
11      \PackageError{gridset}{%
12        neither \expandafter\string\csname #1\endcsname\space
13        nor \xpandafter\string\csname pdf#1\endcsname\space
14        defined%
15      }{This package needs either PDFTeX or LuaTeX or XeTeX.}%
16    \fi
17  \fi
18 }
19 \gridset@luaorpdf{pageheight}
20 \gridset@luaorpdf{pagewidth}
21 \gridset@luaorpdf{savepos}
22 \gridset@luaorpdf{lastxpos}
23 \gridset@luaorpdf{lastypos}
```

`\gridbase` These contain the grid information. `\gridbase` is a integer number representing the absolute y coordinate of the upper end of the grid relative to the same reference point `\pdfsavepos` uses. `\gridinterval` is a integer number representing the distance of two grid lines. The unit is ‘scaled point’ (sp) both time.

```
24 \newcommand*{\gridbase}{}%
25 \newcommand*{\gridinterval}{}%
```

`\gridbase` and `\gridinterval` need to be initialized at the start of the first page (fixme: shouldn’t this be done at the start of every page?). We use this occasion to also initialize `\pdfpageheight` and `\pdfpagewidth` if this hasn’t been done already.

```
26 \AtBeginDocument{%
27   \ifdim\gridset@pageheight=\z@
28     \gridset@pageheight=\paperheight
29   \fi
30   \ifdim\gridset@pagewidth=\z@
31     \gridset@pagewidth=\paperwidth
32   \fi
33   \begingroup
34     \tempdima=\dimexpr \gridset@pageheight - \topmargin - 1in
35       - \headheight - \headsep
```

```

36      - \topskip \relax
37      \tempcnta=\tempdima
38      \xdef\gridbase{\the\tempcnta}%
39      \tempcnta=\baselineskip
40      \xdef\gridinterval{\the\tempcnta}%
41      \endgroup
42 }

```

\savepos Save current position on the page to the **aux**-file. The argument is a unique name
\SavePos for the position. The saved informations are:

- the name of the position,
- the arabic page number of the page with the position,
- the grid base, that was valid for this position,
- the grid interval, that was valid for this position,
- the x-coordinate of the absolute position,
- the y-coordinate of the absolute position.

```

43 \newcommand*{\SavePos}[1]{%
44   \begingroup
45     \gridset@savepos
46     \protected@write\auxout{}{%
47       \protect\newpos{\#1}{\the\count\z@\{\gridbase\}\{\gridinterval\}\%
48         \noexpand\number\gridset@lastxpos
49       }%
50       \noexpand\number\gridset@lastypos
51     }%
52   }%
53   \endgroup
54 }
55 \ifx\savepos\gridset@savepos
56   \PackageInfo{\gridset}{LuaTeX detected.\MessageBreak
57   Note, \gridset command is \string\SavePos\MessageBreak
58   but not \string\savepos, which is\MessageBreak
59   a LuaTeX primitive
60   }%
61 \else
62   \PackageInfo{\gridset}{\string\savepos\space defined as an alias of
63   \string\SavePos}%
64   \newcommand*{\savepos}{\SavePos}%
65 \fi

```

\newpos This is the command, that has been written to the **aux**-file. Reading the **aux**-file it defines several position dependant macros to store the position information. Reading the **aux**-file while **\begindocument** a double definition test is done. Reading the **aux**-file while **\enddocument** a test is done, if the position has been changed

and notes the user about needed additional L^AT_EX runs. (fixme: shouldn't the test be done with the x- and the y-coordinate instead of the vskip only?) The defined macros are:

```
\pos@<position name>@page the arabic page number of the position
\pos@<position name>@base the valid grid base while saving the position
\pos@<position name>@interval the valid grid interval while saving the position
\pos@<position name>@x the x-coordinate of the position
\pos@<position name>@y the y-coordinate of the position
\pos@<position name>@line the number of the next grid line for the position
(first grid line has number 0)
\pos@<position name>@offset distance of the next grid line from the grid base
\pos@<position name>@vskip distance to the next grid line for the position
```

All values are integers. The unit to all values is 'scaled points' (sp). See \pdfsavepos at the pdfT_EX user manual for more information.

```
66 \newcommand*\newpos[6]{%
67   \grid@unique@test{#1}{#2}%
68   \expandafter\global\@namedef{pos@#1@page}{#2}%
69   \expandafter\global\@namedef{pos@#1@base}{#3}%
70   \expandafter\global\@namedef{pos@#1@interval}{#4}%
71   \expandafter\global\@namedef{pos@#1@x}{#5}%
72   \expandafter\global\@namedef{pos@#1@y}{#6}%
73   \begingroup
74     \tempcnta=\numexpr \nameuse{pos@#1@base} - \nameuse{pos@#1@y}\relax
75     \tempcnta=\numexpr \tempcnta + \nameuse{pos@#1@interval} - 1\relax
76     \divide\tempcnta by\nameuse{pos@#1@interval}\relax
77     \expandafter\xdef\csname pos@#1@line\endcsname{\the\tempcnta}%
78     \tempcnta=\numexpr \tempcnta * \nameuse{pos@#1@interval}\relax
79     \expandafter\xdef\csname pos@#1@offset\endcsname{\the\tempcnta}%
80     \tempcnta=\numexpr \nameuse{pos@#1@y}%
81     - ( \nameuse{pos@#1@base} - \tempcnta )\relax
82     \expandafter\let\expandafter\tempa\csname pos@#1@vskip\endcsname%
83     \expandafter\xdef\csname pos@#1@vskip\endcsname{\the\tempcnta}%
84     \expandafter\ifx\csname pos@#1@vskip\endcsname@\tempa\else
85       \grid@ReRunMessage
86     \fi
87   \endgroup
88 }
```

\grid@unique@test A very simple test to warn if a position name isn't unique.

```
89 \newcommand*\grid@unique@test[2]{%
90   \expandafter\ifx\csname pos@#1@page\endcsname\relax\else
```

```

91      \PackageError{gridset}{position '#1' is not unique.\@gobble}{%
92          You have used the position name '#1' you are using on page
93          '#2'\MessageBreak
94          already on page '\csname pos@\#1@page\endcsname'.\MessageBreak
95          You should stop processing, remove the aux-files and correct the
96          names.\MessageBreak
97          If you'd continue, this will result in grid position
98          failures,\MessageBreak
99          that won't be reported!}%
100     \fi
101 }
102 \AtBeginDocument{%
103     \global\let\grid@unique@test\gobble
104 }

\grid@ReRunMessage The change test will be done for each \newpos but one user information at the
end of the document should be enough. So we use a message macro, that destroys
itself after first usage.
105 \newcommand*\grid@ReRunMessage{}
106 \AtBeginDocument{%
107     \renewcommand*\grid@ReRunMessage{%
108         \PackageWarningNoLine{gridset}{Grid position labels may have
109             changed.\MessageBreak
110             Rerun to get grid positions right}%
111         \global\let\grid@ReRunMessage\relax
112     }%
113 }

\vskipnextgrid Move to next grid position. The counter gridcnt is used to give every move to
gridcnt position a unique position name. The names are 'vp!number of the move to
position'. You may use this to get informations e.g. about the last move to
position.
114 \newcounter{gridcnt}
115 \newcommand*{\vskipnextgrid}{%
116     \begingroup
117         \stepcounter{gridcnt}\edef@\tempa{vp!\thegridcnt}%
118         \ifvmode
119             \leavevmode\SavePos{\@tempa}%
120             \expandafter\ifx\csname pos@\@tempa @vskip\endcsname\relax
121             \else
122                 \expandafter\ifnum \csname pos@\@tempa @vskip\endcsname =\z@\else
123                     \PackageInfo{gridset}{%
124                         vmode \string\vskip\csname pos@\@tempa @vskip\endcsname sp%
125                     }%

```

```

126      \vskip -\parskip\vskip -\baselineskip
127      \expandafter\vskip\csname pos@\@tempa @vskip\endcsname sp\relax
128      \fi
129      \fi
130  \else
```

\pdfsavepos in horizontal mode is a problem too, because we have to enter the vertical mode to do vertical skips. Because of this, the remark is the same like the vertical mode remark.

```

131      \parskip=\z@
132      \SavePos{vp!}{\thegridcnt}%
133      \expandafter\ifx\csname pos@\@tempa @vskip\endcsname\relax
134      \else
135          \expandafter\ifnum \csname pos@\@tempa @vskip\endcsname =\z@\else
136              \PackageInfo{gridset}{%
137                  hmode \string\vskip\csname pos@\@tempa @vskip\endcsname sp%
138              }%
139          \vskip \dimexpr -\baselineskip
140                  + \csname pos@\@tempa @vskip\endcsname sp\relax
```

In twocolumn mode we have to take care that in the second column we reduce the horizontal movement by the width of the first column plus the column separation. Note: This may fail, because the column information may be wrong outside the output routine. Maybe we should add this information to \SavePos or correct the x-pos there.

```

141      \leavevmode
142      \if@twoside
143          \expandafter\ifodd\csname pos@\@tempa @page\endcsname\relax
144              \hskip \dimexpr -1in - \oddsidemargin - \parindent
145              \if@twocolumn\if@firstcolumn\else
146                  - \columnwidth - \columnsep
147              \fi\fi
148              + \csname pos@\@tempa @x\endcsname sp\relax
149      \else
150          \hskip \dimexpr -1in - \evensidemargin - \parindent
151          \if@twocolumn\if@firstcolumn\else
152              - \columnwidth - \columnsep
153          \fi\fi
154          + \csname pos@\@tempa @x\endcsname sp\relax
155      \fi
156      \else
157          \hskip \dimexpr -1in - \oddsidemargin - \parindent
158          \if@twocolumn\if@firstcolumn\else
159              - \columnwidth - \columnsep
160          \fi\fi
161          + \csname pos@\@tempa @x\endcsname sp\relax
162      \fi
163      \fi
164      \fi
165  \fi
```

```

166   \endgroup
167 }

(fixme: A better solution would be to first move and then set the position. But
that solution needs some more tests and maybe some more ideas, because after
moving the position is on grid and so the saved x-pos would be on grid.)
```

```

\thegridinfo Some informations about the grid (valid for a position) or the position.
\theuserinfo 168 \newcommand*\thegridinfo}[1]{%
\theypos 169   page=\@nameuse{pos@#1@page},
170   base=\@nameuse{pos@#1@base},
171   interval=\@nameuse{pos@#1@interval},
172   x=\@nameuse{pos@#1@x},
173   y=\@nameuse{pos@#1@y}%
174 }
175 \newcommand*\theuserinfo}[1]{%
176   y=\@nameuse{pos@#1@y},
177   gridline=\@nameuse{pos@#1@line},
178   gridoffset=\@nameuse{pos@#1@offset},
179   movedown=\@nameuse{pos@#1@vskip}%
180 }
181 \newcommand*\theypos}[1]{\@nameuse{pos@#1@y}}
```

3 Example

You may try the following example document. You have to do several L^AT_EX runs until no new rerun warning occurs.

```

182 \documentclass[a4paper,12pt]{article}
183 \usepackage{gridset}
184 \usepackage{blindtext}
185 \raggedbottom
186
187 \pagestyle{myheadings}
188
189 \begin{document}
190 \markright{gridbase=\gridbase, gridinterval=\gridinterval\ without move down}%
191 \newcounter{Zeile}%
192 \makeatletter
193 \@whilenum \value{Zeile}<40\do {%
194   \stepcounter{Zeile}%
195   \theZeile. Zeile:
196   \SavePos{\thepage.\theZeile}\thegridinfo{\thepage.\theZeile}\par
197 }%
198 \makeatother
199 \clearpage
200 \setcounter{Zeile}{0}
201 \makeatletter
202 \@whilenum \value{Zeile}<20\do {%
203   \stepcounter{Zeile}%
204 }
```

```

204   \theZeile. Zeile:
205   \SavePos{\thepage.\theZeile}\theinfo{\thepage.\theZeile}\par
206 }%
207 \makeatother
208 \clearpage
209 \parskip=.5\baselineskip
210 \setcounter{Zeile}{0}
211 \makeatletter
212 \@whilenum {\value{Zeile}<20}\do {%
213   \stepcounter{Zeile}%
214   \theZeile. Zeile:
215   \SavePos{\thepage.\theZeile}\theinfo{\thepage.\theZeile}\par
216 }%
217 \makeatother
218 \clearpage
219 \markright{gridbase=\gridbase, gridinterval=\gridinterval\ with real move down
220 at vmode}%
221 \parskip=.5\baselineskip
222 \setcounter{Zeile}{0}
223 \makeatletter
224 \@whilenum {\value{Zeile}<25}\do {%
225   \stepcounter{Zeile}%
226   \vskipnextgrid\theZeile. Zeile: \theinfo{vp!\thegridcnt}\par
227 }%
228 \makeatother
229 \clearpage
230 \markright{gridbase=\gridbase, gridinterval=\gridinterval\ with real move down
231 at hmode}%
232 \parskip=.5\baselineskip
233 \setcounter{Zeile}{0}
234 \makeatletter
235 \@whilenum {\value{Zeile}<25}\do {%
236   \stepcounter{Zeile}%
237   \theZeile. Zeile: \vskipnextgrid\theinfo{vp!\thegridcnt}\par
238 }%
239 \makeatother
240 \clearpage
241 \parskip=0pt
242 \blindtext
243 \begin{itemize}
244 \item Test
245 \item Test
246 \end{itemize}
247 \vskipnextgrid\theinfo{vp!\thegridcnt}\blindtext
248
249 \end{document}

```

Change History

v0.2	
\SavePos: new name	4
prepared for LuaTeX 0.85	4
\gridbase: prepared for	
LuaTeX 0.85	3
\gridinterval: prepared for	
LuaTeX 0.85	3
\gridset@lastxpos: new internal	
because of LuaTeX 0.85	3
\gridset@lastypos: new internal	
because of LuaTeX 0.85	3
\gridset@pageheight: new	
internal because of	
LuaTeX 0.85	3
\gridset@pagewidth: new internal	
because of LuaTeX 0.85	3
\gridset@savepos: new internal	
because of LuaTeX 0.85	3
\savepos: macro renamed	4
\gridset@luaorpdf: new internal	
because of LuaTeX 0.85	3
v0.3	
\gridcnt: support for twocolumn	
mode	7