Displaying page layout variables

Kent McPherson a.o.*

2020-07-25

> This file is maintained by the LaTeX Project team.
> Bug reports can be opened (category `tools`) at
> https://latex-project.org/bugs.html.

1 Introduction

This LaTeX 2_ε package is a reimplementation of `layout.sty` by Kent McPherson. It defines the command `\layout` which produces an overview of the layout of the current document. The command `\layout*` recomputes the values it uses to produce the overview.

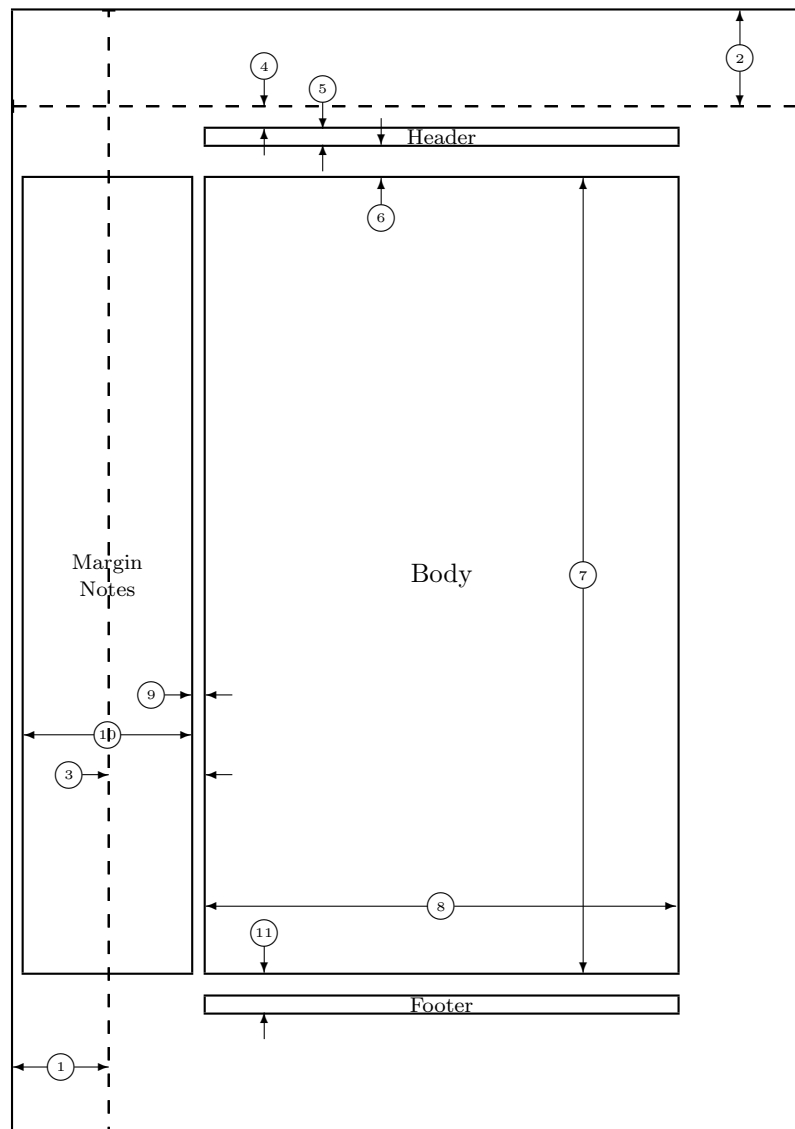The figure on the next page shows the output of the `\layout` command for this document.

2 The implementation

This package prints a figure to illustrate the layout that is implemented by the document class. In the figure several words appear. They are stored in control sequences to be able to select a different language.

```
 1 ⟨*package⟩
 2 \DeclareOption{dutch}{%
 3   \def\Headertext{Kopregel}
 4   \def\Bodytext{Broodtekst}
 5   \def\Footertext{Voetregel}
 6   \def\MarginNotestext{Marge\\Notities}
 7   \def\oneinchtext{een inch}
 8   \def\notshown{niet getoond}
 9   }
10 \DeclareOption{german}{%
11   \def\Headertext{Kopfzeile}
12   \def\Bodytext{Haupttext}
13   \def\Footertext{Fu{\ss}zeile}
14   \def\MarginNotestext{Rand-\\ notizen}
15   \def\oneinchtext{ein Zoll}
16   \def\notshown{ohne Abbildung}
17   }
18 \DeclareOption{ngerman}{\ExecuteOptions{german}}
```

---

*Converted for LaTeX 2_ε by Johannes Braams and modified by Hideo Umeki

Header

Margin
Notes

Body

Footer

1	one inch + \hoffset	2	one inch + \voffset
3	\oddsidemargin = 73pt	4	\topmargin = 17pt
5	\headheight = 12pt	6	\headsep = 25pt
7	\textheight = 598pt	8	\textwidth = 355pt
9	\marginparsep = 11pt	10	\marginparwidth = 126pt
11	\footskip = 30pt		\marginparpush = 0pt (not shown)
	\hoffset = 0pt		\voffset = 0pt
	\paperwidth = 597pt		\paperheight = 845pt

```
19 \DeclareOption{english}{%
20   \def\Headertext{Header}
21   \def\Bodytext{Body}
22   \def\Footertext{Footer}
23   \def\MarginNotestext{Margin\\Notes}
24   \def\oneinchtext{one inch}
25   \def\notshown{not shown}
26   }
27 \DeclareOption{french}{%
28   \def\Headertext{Ent\^{e}te}
29   \def\Bodytext{Corps}
30   \def\Footertext{Pied de page}
31   \def\MarginNotestext{Marge\\Notes}
32   \def\oneinchtext{un pouce}
33   \def\notshown{non affich\'{e}}
34   }
35 \DeclareOption{francais}{\ExecuteOptions{french}}
36 \DeclareOption{spanish}{%
37   \def\Headertext{Encabezamiento}
38   \def\Bodytext{Cuerpo}
39   \def\Footertext{Pie de p\'agina}
40   \def\MarginNotestext{Notas\\ Marginales}
41   \def\oneinchtext{una pulgada}
42   \def\notshown{no mostradas}
43   }
44 \DeclareOption{portuguese}{%
45   \def\Headertext{Cabe\c{c}alho}
46   \def\Bodytext{Corpo}
47   \def\Footertext{Rodap\'e}
48   \def\MarginNotestext{Notas\\ Marginais}
49   \def\oneinchtext{uma polegada}
50   \def\notshown{n\~ao mostradas}
51   }
52 \DeclareOption{brazilian}{%
53   \def\Headertext{Cabe\c{c}alho}
54   \def\Bodytext{Corpo}
55   \def\Footertext{Rodap\'e}
56   \def\MarginNotestext{Notas\\ Marginais}
57   \def\oneinchtext{uma polegada}
58   \def\notshown{n\~ao mostradas}
59   }
60 \DeclareOption{italian}{%
61   \def\Headertext{Testatina}
62   \def\Bodytext{Corpo}
63   \def\Footertext{Piedino}
64   \def\MarginNotestext{Note\\ Marginali}
65   \def\oneinchtext{un pollice}
66   \def\notshown{non mostrato}
67   }
68 \DeclareOption{japanese}{%
69   \def\Headertext{□}
70   \def\Bodytext{□□□□}
71   \def\Footertext{□}
72   \def\MarginNotestext{□\\□}
```

```
73   \def\oneinchtext{1␣␣␣}
74   \def\notshown{␣␣␣}
75   }
```

This package has an option verbose. Using it will make the command `\layout` type some of the parameters on the terminal.

```
76 \DeclareOption{verbose}{\let\LayOuttype\typeout}
77 \DeclareOption{silent}{\let\LayOuttype\@gobble}
```

The normal behaviour of this package when showing the values of the parameters is to truncate them. However, if you want to see the real parameter values you can use the option reals to get that effect.

```
78 \def\lay@value{}
79 \DeclareOption{integers}{%
80   \renewcommand*{\lay@value}[2]{%
81     \expandafter\number\csname #1@#2\endcsname pt}}
82 \DeclareOption{reals}{%
83   \renewcommand*{\lay@value}[2]{\the\csname #2\endcsname}}
```

The default language is English, the default mode is silent and the default way of showing parameter values is to use integers.

```
84 \ExecuteOptions{english,silent,integers}
85 \ProcessOptions
```

\LayOutbs Define `\LayOutbs` to produce a backslash. We use a definition which also works with OT1 fonts.

```
86 \newcommand\LayOutbs{}
87 \chardef\LayOutbs`\\
```

\ConvertToCount This macro stores the value of a *length* register in a *count* register.

```
88 \def\ConvertToCount#1#2{%
```

First copy the value

```
89   #1=#2
```

Then divide it by 65536.

```
90   \divide #1 by 65536}
```

The result of this is that the *count* register holds the value of the *length* register in points.

\SetToHalf Small macros used in computing positions.
\SetToQuart

```
91 \def\SetToHalf#1#2{#1=#2\relax\divide#1by\tw@}
92 \def\SetToQuart#1#2{#1=#2\relax\divide#1by4}
```

\Identify A small macro used in identifying dimensions.

```
93 \def\Identify#1{%
94   \put(\PositionX,\PositionY){\circle{20}}
95   \put(\PositionX,\PositionY){\makebox(0,0){\tiny #1}}
96 }
```

\InsideHArrow This macro is used to produce two horizontal arrows inside a box. The argument gives the width of the box.

```
97 \def\InsideHArrow#1{{%
98   \ArrowLength = #1
```

```
99    \divide\ArrowLength by \tw@
100   \advance\ArrowLength by -10
101   \advance\PositionX by -10
102   \ifnum\ArrowLength<\z@
103     \put(\PositionX,\PositionY){\vector(1,0){-\ArrowLength}}
104     \advance\PositionX by 20
105     \put(\PositionX,\PositionY){\vector(-1,0){-\ArrowLength}}
106   \else
107     \put(\PositionX,\PositionY){\vector(-1,0){\ArrowLength}}
108     \advance\PositionX by 20
109     \put(\PositionX,\PositionY){\vector(+1,0){\ArrowLength}}
110   \fi
111 }}
```

\InsideVArrow This macro is used to produce two vertical arrows inside a box. The argument gives the height of the box.

```
112 \def\InsideVArrow#1{{%
113   \ArrowLength = #1
114   \divide\ArrowLength by \tw@
115   \advance\ArrowLength by -10
116   \advance\PositionY by -10
117   \put(\PositionX,\PositionY){\vector(0,-1){\ArrowLength}}
118   \advance\PositionY by 20
119   \put(\PositionX,\PositionY){\vector(0,+1){\ArrowLength}}
120 }}
```

\OutsideHArrow This macro is used to produce two horizontal arrows to delimit a length. The first argument is the position for the right arrow, the second argument gives the length and the third specifies the length of the arrows.

```
121 \def\OutsideHArrow#1#2#3{{%
122   \PositionX = #1
123   \advance\PositionX by #3
124   \put(\PositionX,\PositionY){\vector(-1,0){#3}}
125   \PositionX = #1 \advance\PositionX-#2
126   \advance\PositionX by -#3
127   \put(\PositionX,\PositionY){\vector(+1,0){#3}}
128 }}
```

\OutsideVArrow This macro is used to produce two vertical arrows to delimit a length. The first argument is the position for the lower arrow, the second argument gives the length and the third and fourth specify the lenghts of the lower and upper arrow.

```
129 \def\OutsideVArrow#1#2#3#4{{%
130   \PositionY = #1
131   \advance\PositionY by -#3
132   \put(\PositionX,\PositionY){\vector(0,+1){#3}}
133   \PositionY = #1
134   \advance\PositionY#2
135   \advance\PositionY#4
136   \put(\PositionX,\PositionY){\vector(0,-1){#4}}
137 }}
```

\Show Macro used in the table that shows the setting of the parameters.

```
138 \def\Show#1#2{\LayOutbs #2 = \lay@value{#1}{#2}}
```

\Type   Macro used to show a setting of a parameter on the terminal.

```
139 \def\Type#1#2{%
140   \LayOuttype{#2 = \lay@value{#1}{#2}}}
```

\oneinch   A constant, giving the length of an inch in points (approximately)

```
141 \newcount\oneinch
142 \oneinch=72
```

Because the overview of the layout is produced in a figure environment we need to allocate a number of counters that are used to store the values of various dimensions.

\cnt@paperwidth   The dimensions of the paper
\cnt@paperheight

```
143 \newcount\cnt@paperwidth
144 \newcount\cnt@paperheight
145 \ConvertToCount\cnt@paperwidth\paperwidth
146 \ConvertToCount\cnt@paperheight\paperheight
```

\cnt@hoffset   the offsets,
\cnt@voffset

```
147 \newcount\cnt@hoffset
148 \newcount\cnt@voffset
149 \ConvertToCount\cnt@hoffset\hoffset
150 \ConvertToCount\cnt@voffset\voffset
```

\cnt@textheight   dimensions of the text area,
\cnt@textwidth

```
151 \newcount\cnt@textheight
152 \newcount\cnt@textwidth
```

\cnt@topmargin   margins,
\cnt@oddsidemargin
\cnt@evensidemargin

```
153 \newcount\cnt@topmargin
154 \newcount\cnt@oddsidemargin
155 \newcount\cnt@evensidemargin
```

\cnt@headheight   dimensions of the running heads,
\cnt@headsep

```
156 \newcount\cnt@headheight
157 \newcount\cnt@headsep
```

\cnt@marginparsep   marginal paragraphs,
\cnt@marginparwidth
\cnt@marginparpush

```
158 \newcount\cnt@marginparsep
159 \newcount\cnt@marginparwidth
160 \newcount\cnt@marginparpush
```

\cnt@footskip   the distance between the running footers and the text,

```
161 \newcount\cnt@footskip
```

and the height of the footers, which is needed here to display a box, but which isn't used by LaTeX.

\fheight

```
162 \newcount\fheight
163 \fheight=12
```

Apart from integer representations of the page layout parameters we also need registers to store reference values in.

\ref@top   The position of the top of the 'printable area' is one inch below the top of the paper by default. The value of \ref@top is relative to the lower left corner of the picture environment that will be used.

```
164 \newcount\ref@top
165 \ref@top=\cnt@paperheight \advance\ref@top by -\oneinch
```

\ref@hoffset
\ref@voffset   For the offsets,

```
166 \newcount\ref@hoffset
167 \newcount\ref@voffset
```

The \hoffset and \voffset values are added to the default offset of one inch.

```
168 \ref@hoffset=\cnt@hoffset  \advance\cnt@hoffset by \oneinch
169 \ref@voffset=\cnt@voffset
```

\cnt@voffset is converted to be relative to the origin of the picture.

```
170 \cnt@voffset=\ref@top
171 \advance\cnt@voffset by -\ref@voffset
```

\ref@head   and the text areas, running heads,

```
172 \newcount\ref@head
```

\ref@body   body of the text

```
173 \newcount\ref@body
```

\ref@foot   and running footers.

```
174 \newcount\ref@foot
```

\ref@margin
\ref@marginwidth
\ref@marginpar   These are different for even and odd pages, so they are computed by \layout.

```
175 \newcount\ref@margin
176 \newcount\ref@marginwidth
177 \newcount\ref@marginpar
```

The following are a number of scratch registers, used in the positioning of the various pices of the picture.

```
178 \newcount\Interval
179 \newcount\ExtraYPos
180 \newcount\PositionX
181 \newcount\PositionY
182 \newcount\ArrowLength
```

\lay@getvalues   All values that might change during the document are computed by calling the macro \lay@getvalues. By default this macro is executed at \begin{document}.

```
183 \def\lay@getvalues{%
184   \ConvertToCount\cnt@textheight\textheight
185   \ConvertToCount\cnt@textwidth\textwidth
186   \ConvertToCount\cnt@topmargin\topmargin
187   \ConvertToCount\cnt@oddsidemargin\oddsidemargin
188   \ConvertToCount\cnt@evensidemargin\evensidemargin
189   \ConvertToCount\cnt@headheight\headheight
190   \ConvertToCount\cnt@headsep\headsep
```

```
191  \ConvertToCount\cnt@marginparsep\marginparsep
192  \ConvertToCount\cnt@marginparwidth\marginparwidth
193  \ConvertToCount\cnt@marginparpush\marginparpush
194  \ConvertToCount\cnt@footskip\footskip
195  \ref@head=\ref@top
196    \advance\ref@head by -\ref@voffset
197    \advance\ref@head by -\cnt@topmargin
198    \advance\ref@head by -\cnt@headheight
199  \ref@body=\ref@head
200    \advance\ref@body by -\cnt@headsep
201    \advance\ref@body by -\cnt@textheight
202  \ref@foot=\ref@body
203    \advance\ref@foot by -\cnt@footskip
204  }
205  \AtBeginDocument{\lay@getvalues}
```

\computevalues The command `\layout` makes the picture and table that display the current set-
\layout tings of the layout parameters.
\layout*
```
206  \newcommand\layout{%
207    \@ifstar{\lay@getvalues\lay@xlayout}{\lay@xlayout}}
208  \def\lay@xlayout{%
209    \lay@layout
210    \if@twoside
211      \lay@layout
212    \fi}
```

\lay@layout The internal macro `\lay@layout` does all the dirty work.
```
213  \newcommand\lay@layout{%
214    \thispagestyle{empty}
```

The actions of `\layout` depend on the pagestyle.
```
215    \if@twoside
216      \ifodd\count\z@
```

Here we deal with an odd page in the twosided case.
```
217        \typeout{Two-sided document style, odd page.}
```

So we compute `\ref@marginwidth`, `\ref@marginpar` and `\ref@margin`.
```
218        \ref@marginwidth=\cnt@oddsidemargin
219        \ref@marginpar=\oneinch
220        \advance\ref@marginpar by \ref@hoffset
221        \advance\ref@marginpar by \cnt@oddsidemargin
222        \ref@margin\ref@marginpar
223        \if@reversemargin
224          \advance\ref@marginpar by -\cnt@marginparsep
225          \advance\ref@marginpar by -\cnt@marginparwidth
226        \else
227          \advance\ref@marginpar by \cnt@textwidth
228          \advance\ref@marginpar by \cnt@marginparsep
229        \fi
230      \else
```

Here we deal with an even page in the twosided case.
```
231    \typeout{Two-sided document style, even page.}
```

So we compute `\ref@marginwidth`, `\ref@marginpar` and `\ref@margin`.

```
232      \ref@marginwidth=\cnt@evensidemargin
233      \ref@marginpar=\oneinch
234      \advance\ref@marginpar by \ref@hoffset
235      \advance\ref@marginpar by \cnt@evensidemargin
236      \ref@margin\ref@marginpar
237      \if@reversemargin
238        \advance\ref@marginpar by \cnt@textwidth
239        \advance\ref@marginpar by \cnt@marginparsep
240      \else
241        \advance\ref@marginpar by -\cnt@marginparsep
242        \advance\ref@marginpar by -\cnt@marginparwidth
243      \fi
244    \fi
245  \else
```

Finally we the case for single sided printing.

```
246      \typeout{One-sided document style.}
247      \ref@marginwidth=\cnt@oddsidemargin
248      \ref@marginpar=\oneinch
249      \advance\ref@marginpar by \ref@hoffset
250      \advance\ref@marginpar by \cnt@oddsidemargin
251      \ref@margin\ref@marginpar
252      \if@reversemargin
253        \advance\ref@marginpar by -\cnt@marginparsep
254        \advance\ref@marginpar by -\cnt@marginparwidth
255      \else
256        \advance\ref@marginpar by \cnt@textwidth
257        \advance\ref@marginpar by \cnt@marginparsep
258      \fi
259  \fi
```

Now we begin the picture environment; dividing all the lengths by two is done by setting `\unitlength` to `0.5pt`

```
260    \setlength{\unitlength}{.5pt}
261    \begin{picture}(\cnt@paperwidth,\cnt@paperheight)
262      \centering
263      \thicklines
```

First we have the pagebox and reference lines,

```
264      \put(0,0){\framebox(\cnt@paperwidth,\cnt@paperheight){\mbox{}}}
265      \put(0,\cnt@voffset){\dashbox{10}(\cnt@paperwidth,0){\mbox{}}}
266      \put(\cnt@hoffset,0){\dashbox{10}(0,\cnt@paperheight){\mbox{}}}
```

then the header,

```
267      \put(\ref@margin,\ref@head){%
268        \framebox(\cnt@textwidth,\cnt@headheight)%
269          {\footnotesize\Headertext}}
```

the body of the text area,

```
270      \put(\ref@margin,\ref@body){%
271        \framebox(\cnt@textwidth,\cnt@textheight){\Bodytext}}
```

the footer

```
272     \put(\ref@margin,\ref@foot){%
273        \framebox(\cnt@textwidth,\fheight){\footnotesize\Footertext}}
```

and the space for marginal notes.

```
274     \put(\ref@marginpar,\ref@body){%
275        \framebox(\cnt@marginparwidth,\cnt@textheight)%
276               {\footnotesize\shortstack{\MarginNotestext}}}
```

Then we start putting in 'arrows' to mark the various parameters. From here we use `\thinlines`.

```
277     \thinlines
```

`\PositionX` and `\PositionY` will be the coordinates of the center of the arrow displaying `\textwidth`.

```
278     \SetToHalf\PositionX\cnt@textwidth
279     \advance\PositionX by \ref@margin
```

The arrow should be a bit above the bottom of the 'body box'.

```
280     \PositionY = \ref@body
281     \advance\PositionY by 50
```

An identifying number is put here, in a circle.

```
282     \Identify{8}
```

Then the arrow is drawn.

```
283     \InsideHArrow\cnt@textwidth
```

Now the `\textheight`

```
284     \SetToHalf\PositionY\cnt@textheight
285     \advance\PositionY by \ref@body
```

The x-position of the arrow is at 4/5 of the width of the 'body box'.

```
286     \PositionX = \cnt@textwidth
287     \divide\PositionX by 5
288     \multiply \PositionX by 4
289     \advance\PositionX by \ref@margin
```

An identifying number is put here, in a circle.

```
290     \Identify{7}
291     \InsideVArrow\cnt@textheight
```

The `\hoffset`,

```
292     \PositionY = 50
293     \SetToHalf\PositionX\cnt@hoffset
294     \Identify{1}
295     \InsideHArrow\cnt@hoffset
```

The width of the margin.

```
296     \SetToQuart\PositionY\cnt@textheight
297     \advance\PositionY by \ref@body
298     \ifnum\ref@marginwidth > 0
299        \OutsideHArrow\ref@margin\ref@marginwidth{20}
300        \PositionX = \cnt@hoffset
301     \else
302        \OutsideHArrow\cnt@hoffset{-\ref@marginwidth}{20}
303        \PositionX = \ref@margin
```

```
304      \fi
305      \advance\PositionX by -30
306      \Identify{3}
```

the \marginparwidth,

```
307      \SetToQuart\PositionY\cnt@textheight
308      \advance\PositionY by \ref@body
```

This arrow has to be bit below the one for the \oddsidemargin or \evensidemargin.

```
309      \advance\PositionY by 30
310      \SetToHalf\PositionX\cnt@marginparwidth
311      \advance\PositionX by \ref@marginpar
312      \Identify{10}
313      \InsideHArrow\cnt@marginparwidth
```

The \marginparsep, this depends on single or double sided printing.

```
314      \advance\PositionY by 30
315      \if@twoside
```

Twosided mode, reversemargin;

```
316        \if@reversemargin
317          \ifodd\count\z@
318            \OutsideHArrow\ref@margin\cnt@marginparsep{20}
319            \PositionX = \ref@margin
320          \else
321            \OutsideHArrow\ref@marginpar\cnt@marginparsep{20}
322            \PositionX = \ref@marginpar
323          \fi
324        \else
```

Not reversemargin;

```
325        \ifodd\count\z@
326          \OutsideHArrow\ref@marginpar\cnt@marginparsep{20}
327          \PositionX = \ref@marginpar
328        \else
329          \OutsideHArrow\ref@margin\cnt@marginparsep{20}
330          \PositionX = \ref@margin
331        \fi
332      \fi
333    \else
```

Single sided mode.

```
334      \if@reversemargin
335        \OutsideHArrow\ref@margin\cnt@marginparsep{20}
336        \PositionX = \ref@margin
337      \else
338        \OutsideHArrow\ref@marginpar\cnt@marginparsep{20}
339        \PositionX = \ref@marginpar
340      \fi
341    \fi
342    \advance\PositionX by -\cnt@marginparsep
343    \advance\PositionX by -30
344    \Identify{9}
```

Identify the `\footskip`. The arrow will be located on 1/8th of the `\textwidth`.

```
345     \PositionX = \cnt@textwidth
346     \divide\PositionX by 8
347     \advance\PositionX by \ref@margin
348     \OutsideVArrow\ref@foot\cnt@footskip{20}{20}
349     \PositionY = \ref@foot
350     \advance\PositionY by \cnt@footskip
351     \advance\PositionY by 30
352     \Identify{11}
```

Identify the `\voffset`. The arrow will be located a bit to the left of the edge of the paper.

```
353     \PositionX = \cnt@paperwidth
354     \advance\PositionX by -50
355     \PositionY = \cnt@paperheight
356     \ExtraYPos = \PositionY
357     \advance\ExtraYPos by -\cnt@voffset
358     \advance\PositionY by \cnt@voffset
359     \divide\PositionY by \tw@
360     \Identify{2}
361     \InsideVArrow\ExtraYPos
```

Identify `\topmargin`, `\headheight` and `\headsep`.

The arrows will be located on 1/8th of the `\textwidth`, with intervals of the same size, stored in `\Interval`.

```
362     \Interval = \cnt@textwidth
363     \divide\Interval by 8
364     \PositionX = \ref@margin
365     \advance\PositionX by \Interval
```

First the `\topmargin`. If `\topmargin` has a positive value, the arrow is upward. Otherwise, it is downward. The number label is always placed at the base of the arrow.

```
366     \ifnum\cnt@topmargin > \z@
367       \ExtraYPos = \ref@head
368       \advance\ExtraYPos\cnt@headheight
369       \OutsideVArrow\ExtraYPos\cnt@topmargin{20}{20}
370       \PositionY = \ExtraYPos
371       \advance\PositionY by \cnt@topmargin
372     \else
373       \ExtraYPos = \cnt@voffset
374       \OutsideVArrow\ExtraYPos{-\cnt@topmargin}{20}{20}
375       \PositionY = \ExtraYPos
376       \advance\PositionY by -\cnt@topmargin
377     \fi
378     \advance\PositionY by 30
379     \Identify{4}
380     \advance\PositionX by \Interval
```

Then the `\headheight`

```
381     \OutsideVArrow\ref@head\cnt@headheight{20}{20}
382     \PositionY = \ref@head
383     \advance\PositionY by \cnt@headheight
384     \advance\PositionY by 30
385     \Identify{5}
```

```
386     \advance\PositionX by \Interval
```

and finally the `\headsep`

```
387     \ExtraYPos=\ref@body
388     \advance\ExtraYPos\cnt@textheight
389     \OutsideVArrow\ExtraYPos\cnt@headsep{20}{20}
390     \PositionY = \ref@body
391     \advance\PositionY by \cnt@textheight
392     \advance\PositionY by -30
393     \Identify{6}
```

Here we can end the picture environment and insert a little space.

```
394   \end{picture}
395
396   \medskip
```

Below the picture we put a table to show the actual values of the parameters. Note that fractional points are truncated, i.e., `72.27pt` is displayed as `72pt`

The table is typeset inside a box with a depth of 0 to always keep it on the same page as the picture.

```
397   \vtop to 0pt{%
398     \@minipagerestore\footnotesize\ttfamily
399     \begin{tabular}{@{}rl@{\hspace{20pt}}rl}
400       1 & \oneinchtext\ + \LayOutbs\texttt{hoffset}
401         & 2 & \oneinchtext\ + \LayOutbs\texttt{voffset} \\
402       3 & \if@twoside
403           \ifodd\count\z@ \Show{cnt}{oddsidemargin}
404           \else \Show{cnt}{evensidemargin}
405           \fi
406         \else
407           \Show{cnt}{oddsidemargin}
408         \fi                    & 4 & \Show{cnt}{topmargin} \\
409       5 & \Show{cnt}{headheight} & 6 & \Show{cnt}{headsep} \\
410       7 & \Show{cnt}{textheight} & 8 & \Show{cnt}{textwidth} \\
411       9 & \Show{cnt}{marginparsep}&10& \Show{cnt}{marginparwidth} \\
412       11& \Show{cnt}{footskip}  &  & \Show{cnt}{marginparpush}
413        \rlap{(\notshown)}\\
414         & \Show{ref}{hoffset}   &  & \Show{ref}{voffset} \\
415         & \Show{cnt}{paperwidth} &  & \Show{cnt}{paperheight} \\
416
417   \end{tabular}\vss}
```

When the option verbose was used the following lines will show dimensions on the terminal.

```
418   \Type{ref}{hoffset}
419   \Type{ref}{voffset}
420   \Type{cnt}{textheight}
421   \Type{cnt}{textwidth}
```

Finally we start a new page.

```
422   \newpage
423 }
424 ⟨/package⟩
```